



**UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO**  
PROGRAMA DE MAESTRÍA Y DOCTORADO EN INGENIERÍA  
INGENIERÍA DE SISTEMAS – PLANEACIÓN

ESTRUCTURAS PARA EL DESARROLLO DE PROYECTOS DE SOFTWARE CON  
ENFOQUE EN LOS STAKEHOLDERS

TESIS  
QUE PARA OPTAR POR EL GRADO DE:  
MAESTRO EN INGENIERÍA

PRESENTA:  
ING. JORGE LUIS CANO CONDE

DR. BENITO SÁNCHEZ LARA  
FACULTAD DE INGENIERÍA  
PROGRAMA DE MAESTRÍA Y DOCTORADO EN ING. DE SISTEMAS

MÉXICO, D. F. SEPTIEMBRE 2015

**JURADO ASIGNADO:**

Presidente: M.I. Arturo Fuentes Zenón  
Secretario: Dr. Tomás Bautista Godínez  
Vocal: Dr. Benito Sánchez Lara  
1 er. Suplente: Dr. José Emilio Navas López  
2 do. Suplente: M. en I. Ricardo Rubén Franco Díaz

**Ciudad Universitaria, México, D.F. 2015**

**TUTOR DE TESIS:**

**DR. BENITO SÁNCHEZ LARA**

---

**FIRMA**

# Índice

## Resumen

## Introducción

### Capítulo 1

#### **Antecedentes y problemática en el desarrollo de proyectos de software.....11**

1.1 Empresas desarrolladoras de software: Importancia económica y datos relevantes del mercado del software en México.....	11
1.2 Las organizaciones y los proyectos de software.....	13
1.3 Integración del modelo de madurez de capacidades (CMMI) como proceso de mejora en los proyectos de software.....	17
1.4 Problemática en el desarrollo de proyectos de software .....	21
1.5 Planteamiento del problema y objetivo de la tesis.....	23
1.6 Justificación y alcance.....	24

### Capítulo 2

#### **Revisión de factores asociados al éxito y fracaso de los proyectos de software desde una perspectiva de stakeholders.....26**

#### **2.1 Desarrollo de software.....27**

2.1.1 Definición de proyecto de sistema o software.....	27
2.1.2 Tipos de desarrollo de software.....	29
2.1.2.1 Software Libre.....	29
2.1.2.2 Software Privado.....	30
2.1.3 Estudios recientes en evaluación y desarrollo de proyectos de software.....	30
2.1.3.1 Perfiles de éxito / fracaso de los proyectos de software.....	30
2.1.3.2 Casos de estudio referentes a proyectos de software.....	35
2.1.4 Factores críticos de éxito.....	37
2.1.4.1 Los factores críticos de éxito en la implantación de sistemas de información o software en organizaciones mexicanas.....	37

#### **2.2 Los stakeholders.....38**

2.2.1 Teoría y definición de stakeholders.....	38
2.2.2 Desarrollo de software y los stakeholders.....	42
2.2.3 Ejemplo de participación de los stakeholders: El gobierno corporativo.....	44

#### **2.3 Metodologías de proyectos con un enfoque en stakeholders.....45**

2.3.1 Modelo integrado de madurez y capacidades (CMMI).....	45
2.3.2 Fundamentos de Gestión de Proyectos (PMBOK) y la identificación de	

stakeholders.....	48
<b>2.4 La Estructura Organizacional.....</b>	<b>52</b>
2.4.1 Clases de estructura organizacional .....	52
2.4.2 Factores clave para diseñar la estructura de la organización.....	55
2.4.3 Criterios para diseñar una estructura organizacional de empresas.....	57
2.4.4 Capas de análisis estratégico para diseñar una estructura organizacional orientada en stakeholders.....	58
2.4.4.1 El Interés.....	59
2.4.4.2 La Identidad.....	59
2.4.4.3 La relación entre los individuos.....	59
2.4.4.4 La transmisión de conocimiento.....	60
<b>2.5 El enfoque de planeación y de sistemas.....</b>	<b>60</b>
2.5.1 Planeación interactiva.....	60
2.5.2 Planeación transactiva.....	62
2.5.3 La responsabilidad social corporativa como marco conductor del desarrollo de proyectos de software.....	64
<b>2.6 Estrategia de investigación.....</b>	<b>65</b>
2.6.1 Estrategia.....	66
<b>Capítulo 3</b>	
<b>Análisis, diseño y validación de estructuras y mecanismos de gestión de stakeholders asociados a proyectos de software.....</b>	<b>68</b>
<b>3.1 Identificación de stakeholders en proyectos de software.....</b>	<b>68</b>
3.1.1 Software libre.....	68
3.1.2 Software privado.....	71
3.1.3 Análisis comparativo.....	73
<b>3.2 Dimensiones de análisis: interés, identidad, membresía, relación, (coordinación e integración).....</b>	<b>76</b>
3.2.1 Análisis funcional basado en el interés de los stakeholders.....	79
3.2.2 Análisis funcional basado en la identidad de los stakeholders.....	82
3.2.3 Análisis funcional basado en la relación de los stakeholders.....	84
3.2.4 Análisis funcional basado en la membresía de los stakeholders.....	86
3.2.5 Efectos del traslape de interés, identidad, relación y membrecía en la estructura stakeholders de proyectos de software.....	86
<b>3.3 Mecanismos de gestión y estructuras de stakeholders para el desarrollo de proyectos de software.....</b>	<b>89</b>
<b>3.4 Modelos estructurales de stakeholders.....</b>	<b>93</b>
3.4.1 Modelo estructural de stakeholders basado en el interés.....	93

3.4.2 Modelo estructural de stakeholders basado en la identidad.....	94
3.4.3 Modelo estructural de stakeholders basado en la relación.....	96
3.4.4 Modelo estructural de stakeholders basado en la membresía.....	98
<b>3.5 Validación de los modelos estructurales.....</b>	<b>100</b>
3.5.1 Herramienta de orientación a stakeholders.....	101
3.5.1.1 Identificación y análisis de stakeholders.....	101
3.5.1.2 Priorización.....	102
3.5.1.3 Gestión de las relaciones de los stakeholders.....	103
3.5.1.4 Organización e integración de stakeholders por medio de los modelos estructurales de interés, identidad, relación y membresía.....	104
3.5.1.5 Análisis de stakeholders.....	105
3.5.2 Proceso de validación.....	106
3.5.3 Discusión de resultados específicos relacionados a la herramienta de orientación a stakeholders.....	110
<b>Capítulo 4</b>	
<b>Conclusiones.....</b>	<b>113</b>
<b>Anexo A.....</b>	<b>120</b>
<b>Anexo B.....</b>	<b>122</b>
<b>Anexo C.....</b>	<b>123</b>
<b>Anexo D.....</b>	<b>124</b>
<b>Anexo E.....</b>	<b>125</b>
<b>Anexo F.....</b>	<b>128</b>
<b>Referencias.....</b>	<b>131</b>

## Índice de tablas

<b>Tabla 1</b> Mayores empresas en el mercado del software y las tecnologías de la información (TIC).....	12
<b>Tabla 2</b> Empresas establecidas en México.....	13
<b>Tabla 3</b> Certificaciones en la industria de software y TIC.....	13
<b>Tabla 4</b> Factores de éxito de los proyectos de software.....	32
<b>Tabla 5</b> Problemas comunes en la construcción de software.....	33
<b>Tabla 6</b> Factores que afectan a los proyectos de Software.....	34
<b>Tabla 7</b> Porcentaje de administradores de proyectos de TI que cree que hay más fallas en los proyectos ahora que hace cinco y hasta diez años.....	34
<b>Tabla 8</b> Proyectos TI relevantes para el estudio del <i>Standish Group</i> y detección de stakeholders involucrados.....	35
<b>Tabla 9</b> Niveles de participación de Stakeholders en el gobierno corporativo.....	44
<b>Tabla 10</b> Niveles y caracterización del modelo CMMI .....	45
<b>Tabla 11</b> Clases de estructuras.....	53
<b>Tabla 12</b> Factores que afectan las estructuras organizacionales.....	55
<b>Tabla 13</b> Factores que se toman en cuenta para la construcción de estructuras organizacionales.....	57
<b>Tabla 14.</b> Grupos de interés internos.....	69
<b>Tabla 15</b> Externos que financian o participan en el proceso de desarrollo.....	70
<b>Tabla 16</b> Externos ajenos a la financiación o proceso de desarrollo.....	70
<b>Tabla 17</b> Stakeholders externos ajenos a la financiación o proceso de desarrollo.....	71
<b>Tabla 18</b> Stakeholders internos (comprometidos).....	72
<b>Tabla 19</b> Stakeholders ajenos al proceso de desarrollo (involucrados).....	72
<b>Tabla 20</b> Análisis comparativo entre tipos de software.....	74
<b>Tabla 21</b> Especialistas en proyectos de software involucrados en el proceso de validación.....	106
<b>Tabla 22</b> Percepción de variables de los especialistas entrevistados de sus diferentes proyectos asociados.....	107
<b>Tabla 23</b> Información cuantitativa resultado de las respuestas dadas por los especialistas.....	109

## Índice de Figuras

<b>Figura 1</b> Gestión de Proyectos de Software.....	15
<b>Figura 2</b> Proceso de Administración de Proyecto y stakeholders involucrados.....	18
<b>Figura 3</b> Proceso de Ingeniería y stakeholders involucrados.....	19
<b>Figura 4</b> Tres aspectos de la Teoría de los Stakeholders.....	40
<b>Figura 5</b> Contrastando el modelo de la organización y la teoría de los stakeholders.....	40
<b>Figura 6</b> Tipología de los stakeholders: uno dos y tres atributos.....	42
<b>Figura 7</b> Esquema de ubicación de stakeholders internos y externos en las empresas desarrolladoras de software.....	50
<b>Figura 8</b> Niveles de estudio de los stakeholders dentro de un proyecto de software.....	78
<b>Figura 9</b> Condición primaria para que exista algún tipo de influencia sobre los stakeholders en la empresa de software.....	80
<b>Figura 10</b> La identidad y la solidaridad originada como marco de acción de stakeholders.....	83
<b>Figura 11</b> Modelo de movilización entre stakeholders basado en los vínculos a través del tiempo.....	84
<b>Figura 12</b> Variable de membresía como transmisión de conocimiento, experiencias y roles entre grupos de stakeholders (membresía traslapada).....	86
<b>Figura 13</b> Variables encontradas en el análisis de stakeholders en proyectos de software y su traslape.....	88
<b>Figura 14</b> Modelo estructural de stakeholders de proyectos de software libre.....	91
<b>Figura 15</b> Modelo estructural de stakeholders de proyectos de software privado.....	92
<b>Figura 16</b> Modelo estructural de stakeholders de un proyecto de software basado en el interés.....	93
<b>Figura 17</b> Modelo estructural de stakeholders de un proyecto de software basada en su identidad.....	95
<b>Figura 18</b> Estructura de stakeholders de un proyecto de software basada en la relación.....	96
<b>Figura 19</b> Estructura de stakeholders de un proyecto de software basado en la membresía.....	99

## RESUMEN

Una de las industrias tecnológicas con mayor crecimiento a nivel mundial según los reportes presentados por la Secretaría de Economía en México en 2014 es la industria del desarrollo de software. Dicha industria es cada vez más competitiva y enfrenta retos en innovación, calidad, tiempo de desarrollo de sus proyectos entre otros. Nuevos procedimientos, metodologías, técnicas y estrategias buscan seguir patrones de diseño apegados a los principios organizacionales de las entidades propietarias de software sin considerar en su mayoría estrategias orientadas a identificar y satisfacer las necesidades de los interesados o también denominados *stakeholders*, lo que puede provocar insatisfacciones que pueden impactar en el cumplimiento de las tareas y responsabilidades internas de la organización poco orientadas al cliente y a los usuarios del producto y por consiguiente fracasos en los proyectos. Contribuciones científicas como por ejemplo la teoría de stakeholders, metodologías como lo es el Modelo Integrado de Madurez y Capacidades (CMMI) y enfoques de administración de proyectos y empresas como los fundamentos de gestión de proyectos (PMBOK) o la Responsabilidad Social Corporativa (RSC), toman en cuenta aspectos fundamentales en el estudio e inclusión de stakeholders en diferentes disciplinas sin analizar metodológicamente dicha orientación a favor de los proyectos en cuestión ni el beneficio de incluir actividades orientadas a dichos grupos de interés. Por lo que es necesario medir el papel de los stakeholders en proyectos de software para proponer diversos medios estratégicos y recursos funcionales los cuales consideren los esfuerzos realizados por dichos grupos de interés y por consiguiente sean orientados al éxito de los proyectos en término ya sea del costo, esfuerzo y/o tiempo. Por tal motivo, se realizó un análisis sistémico tomando como base de estudio a proyectos de software comercial con un enfoque de planeación interactiva orientado al análisis de estructuras de stakeholders y se evaluaron los diferentes escenarios que pueden ayudar a crear recursos efectivos en apoyo al éxito de proyectos de software teniendo como resultado una la orientación planeada de stakeholders en proyectos de software.

**Palabras clave:** Stakeholders, proyectos de software, estructuras de stakeholders.

## INTRODUCCIÓN

Diversos son los retos que enfrentan las empresas en la actualidad, cuyo principio fundamental es el de ofrecer mejores servicios a sus clientes, aumentar sus ganancias, creación de valor económico en su entorno y enfrentar la competencia. Existen rubros empresariales que compiten por mejorar las expectativas y experiencias de sus clientes por medio de mejoras en sus procesos y procedimientos. Uno de ellos es el referente a la industria de desarrollo de software, caracterizada por ofrecer propuestas innovadoras por medio del desarrollo de proyectos enfocados a mejorar procesos tácticos y estratégicos de las empresas y de las actividades humanas para solucionar diversos problemas funcionales, satisfacción de necesidades, oportunidades de negocio, recursos de entretenimiento entre otros.

Las necesidades y los requerimientos cambian y solo aquellos proyectos que logren reunir las características necesarias enfocadas a la satisfacción de dichos requerimientos son los que tendrán una mayor rentabilidad, credibilidad social y por consiguiente un mayor éxito y relevancia por parte de los usuarios. Ejemplo de ello son las empresas de consultoría en tecnologías de la información las cuales requieren de procedimientos para brindar mejores experiencias a sus clientes por medio de aplicaciones web, móviles, de escritorio o personalizadas así como software para control de maquinaria en el caso de las empresas de manufactura, software para instrumentos médicos entre otros, lo que implica mayores esfuerzos organizacionales y de recursos humanos capacitados en diversas áreas multidisciplinarias.

Una de las problemáticas surgidas para lograr dichos objetivos es referente a las dificultades que enfrentan los proyectos de software en materia de organización, tiempos de desarrollo, cambios en los requerimientos, trabajo extra por parte de sus implicados para lograr los objetivos del proyecto, mayores inversiones y por consiguiente insatisfacción en sus clientes cuyas consecuencias se ven reflejadas en la cancelación de proyectos, prolongación de periodos de desarrollo, fuga de personal, desmotivación, recontractación de recursos humanos y renovación de licencias en uso de herramientas informáticas lo que implica una mayor inversión empresarial, reestructuración interna y mayores esfuerzos de crecimiento económico.

Surge la propuesta de realizar un análisis de stakeholders enfocado al desarrollo de proyectos de software tomando en cuenta las insatisfacciones que se pueden presentar por los interesados, la falta de un enfoque compartido hacia los stakeholders y por consiguiente, las repercusiones hacia los proyectos siguiendo una estrategia orientada en el impacto de la estructura funcional de stakeholders, con necesidades, intereses y hábitos propios, en términos de sus funciones, responsabilidades, roles y actividades y donde se puedan encontrar prácticas enfocadas a la resolución los requerimientos técnicos y metodológicos en el desarrollo de proyectos y a la vez contemplar actividades orientadas a stakeholders.

Los resultados de esta investigación están basados en marcos estratégicos que contemplan las razones por las cuales se justifican las acciones de los stakeholders dentro de las empresas como lo son sus intereses, su identidad propia y hacia el trabajo, su forma de relacionarse y la transmisión de conocimientos y habilidades entre rangos de responsabilidad

donde los stakeholders pueden ser beneficiados sin desviar la atención en el desarrollo de los proyectos. Se considera que dichos marcos de análisis junto con el análisis económico de la industria de software, el aporte de la teoría de stakeholders, de las principales metodologías de administración y desarrollo de software utilizadas por las organizaciones y de las prácticas estratégicas de la responsabilidad social corporativa (RSC) como marco conductor de valores empresariales, pueden servir como base metodológica y aporte innovador en proyectos de software, muchas veces limitados por tiempos de desarrollo y carentes de orientación a stakeholders, para así, hacer una propuesta de orientación a stakeholders planeado por medio del enfoque sistémico de stakeholders y proyectos de software así como del aporte del proceso de planeación interactiva como marco para la coordinación e integración y por consiguiente, explicar las razones que provocan la movilización de los stakeholders orientadas al éxito de los proyectos de software y poder proponer recursos metodológicamente diseñados para ser utilizados en organizaciones carentes de un enfoque hacia sus stakeholders.

## ANTECEDENTES Y PROBLEMÁTICA EN EL DESARROLLO DE PROYECTOS DE SOFTWARE

### 1.1 EMPRESAS DESARROLLADORAS DE SOFTWARE: IMPORTANCIA ECONÓMICA Y DATOS RELEVANTES DEL MERCADO DEL SOFTWARE EN MÉXICO

Hoy en día, las personas experimentan diversos cambios, reflejados en su manera de vivir, sus gustos y aficiones. Durante los últimos años, la humanidad ha experimentado la evolución de muchos de los procesos simples de vida como por ejemplo la comunicación entre las personas, el uso de dispositivos que facilitan actividades cotidianas así como la creación de nuevas prácticas y procedimientos dentro de las empresas. Sin duda son las nuevas tecnologías una de las principales causas de diversos cambios experimentados a través de la historia cuya definición se ve reflejada en el desarrollo, uso y empleo de cualquier objeto, método u organización en servicio de la humanidad para así poder entablar una relación cordial entre el medio ambiente y la humanidad. En la actualidad, el uso de dispositivos tecnológicos como los gadgets, computadoras, aparatos móviles, tabletas, etc. se ha convertido en una actividad tan cotidiana como el consumo de alimentos básicos ya que sin el uso de dichos dispositivos no se podrían realizar la mayoría de actividades que dependen de su uso como por ejemplo el procesamiento de datos provenientes de personas, empresas, actividades o financieras.

El software es un tipo de tecnología informática que ha evolucionado a través del tiempo gracias a la aportación de científicos y profesionales en tecnologías de la información (TI) y forma parte de los cambios tecnológicos mencionados con anterioridad y están reflejados dentro del sistema de procesamiento y funcionamiento de la gran mayoría de dispositivos electrónicos actuales. El software es un programa de computadora que trabaja a base de órdenes que se ejecutan en los dispositivos para procesar información mucho más eficiente que la de un ser humano. Cuando un diseño y construcción de programas es desarrollado para realizar una tarea en específico se le puede llamar aplicación. Existen aplicaciones que pueden satisfacer distintas necesidades intelectuales, laborales e incluso emocionales cuya función principal es el procesamiento de la información resultante de dichos procesos como por ejemplo aplicaciones para medir el tiempo, para escribir electrónicamente, procesar números, etc. El enfoque de estudio del presente trabajo pretende adentrarse en los antecedentes del uso y desarrollo del software en las empresas y con los usuarios finales (personas que utilizan las aplicaciones).

El software se encuentra en la categoría de tecnologías de la información y comunicación (TIC) la cual evoluciona a través del tiempo e impacta en el desarrollo de proyectos de software ya que el uso de diversas herramientas informáticas puede cambiar y las metodologías usadas para implementar dichas herramientas también. La industria de las TIC crece a niveles notables, se tiene registrado que el mercado de servicios de tecnologías de la información (TI) durante el año 2014 fue de 2.22 billones de dólares, cifra que implica un incremento de 6.2% con respecto al año anterior la cual fue de 1.6% (TI Forrester Research, 2013). Estados Unidos es el país con mayor participación en el gasto global de servicios de TI (40% del total) seguidos

de Canadá, India y Filipinas. Respecto al software se tiene registrado que el gasto global durante el 2014 fue de 568,000 millones de dólares seguido por el outsourcing de TI con 442,000 millones, la consultoría de TI y los servicios de integración con 421,000 millones, el equipamiento informático con 416,000 millones, y los equipos de comunicaciones con 373,000 millones de dólares, según el informe de Forrester. Estados Unidos es el país que más invierte en éste tipo de tecnología (alrededor de un 40% del valor global) seguido de Japón, Alemania, Reino Unido y Francia.

Otra de las actividades dentro del desarrollo de software que han tenido crecimiento durante los últimos años es el denominado “Cómputo en la nube” la cual es una práctica que ofrece o utiliza programas y recursos de cómputo desde el internet y cuya inversión registrada oscila entre los 68,000 millones de dólares reemplazando al desarrollo de software tradicional (*Secretaría de economía, 2013*). Existen distintas empresas cuyo negocio se centra en el mercado de las tecnologías de la información y en especial las de procesamiento de la información, como lo son las del software, empresas de diseño, servicios en TI, servicios en capacitación, mantenimiento, operación y soporte de programas, Call centers, etc. Diversas son las empresas que apuestan por el negocio de las tecnologías de la información como lo son las que presentan mayor inversión en investigación y desarrollo (I + D), las que ofrecen servicios de TI y las que tienen mayores ingresos sea por la comercialización o explotación de herramientas de software.

**Tabla 1** Mayores empresas en el mercado del software y las tecnologías de la información TIC (*Secretaría de economía, 2013*)

EMPRESAS CON MAYOR INVERSIÓN EN I + D	EMPRESAS DE SOFTWARE POR VOLÚMENES DE INGRESOS	EMPRESAS EN SERVICIOS DE TI (20%)
Microsoft	Microsoft	HP
Oracle	IBM	IBM
SAP	Oracle	Accenture
CA Technologies	-	Fujitsu
Simantec	-	CSC (Computer Software College)

En el caso de **México** los servicios de TI presentaron un valor de mercado de 3, 988.26 millones de dólares durante el año 2010 lo que representa un crecimiento de 15 % respecto al año anterior. Por su parte, la industria del software en México presentó un valor de mercado de 1,546.14 millones de dólares en el mismo año creciendo a un 12% con respecto al año anterior. Ambos rubros, el de servicios de TI como el de software crecerán a una tasa de 11% anual (*Business Monitor, 2015*). Además. México cuenta con más de 25 clúster de TI a lo largo del país que agrupan a más de 700 compañías en ésta industria ya que academias, iniciativas privadas y gobierno han desarrollado 24 parques tecnológicos a lo largo y ancho del país.

El crecimiento de la industria de las tecnologías de la información crece año con año. Si bien los datos reportados anteriormente pertenecen al estudio realizado por la Secretaría de Economía durante el año 2010. Las tendencias consideran un crecimiento anual promedio. Para el año 2015, se espera un crecimiento del 6.3% en el mercado mexicano, lo que significa más del promedio de crecimiento del mismo rubro en América Latina. Dicho aumento es consecuencia del crecimiento de la clase media promedio (40 % de la población para el año 2030), así como el aumento de expansión de dicho mercado debido a las inversiones hechas en la región latinoamericana. Según los reportes presentados en el año 2015, existen tendencias de crecimiento llamado “Software as a service”, el cual se menciona que tendrá consecuencias no solo en el mercado de software sino también en el de hardware mientras que los servicios de TI crecerán 6.5% y el del software un 11% respecto al año anterior (*Intertational Data Consulting IDC, 2015*).

**Tabla 2** Empresas establecidas en México TIC (*Secretaría de Economía, 2013*).

Posicionamiento en México	Empresa
1	Microsoft
2	Accenture
3	IBM
4	Softteck
5	Neoris
6	Infosys

Las certificaciones que pueden tener las empresas cuyo rubro es el software, existen algunas que garantizan mayor calidad en los productos de software y que son prueba de la existencia de buenas prácticas en organizaciones de ésta naturaleza como por ejemplo:

**Tabla 3** Certificaciones en la industria de software y TIC (*Secretaría de economía, 2013*)

Relevancia en empresas de software	Siglas en inglés	Significado
1	CMMI	Capability Maturity Model Integration
2	MOPROSOFT	Modelo de procesos para la industria del Software
3	CMM	Certification of Meeting Management

## 1.2 LAS ORGANIZACIONES Y LOS PROYECTOS DE SOFTWARE

La razón por la cual se define al software y el beneficio que aporta a la teoría de stakeholders, considerando su validez en las organizaciones de desarrollo e innovación tecnológica, nos conduce a realizar un análisis riguroso sobre los límites de dichas organizaciones. Para Luhmann (1998), desde la Teoría de los Sistemas Sociales la aplicación de la Teoría General de Sistemas se centra en la Teoría de la Comunicación como un componente característico de los sistemas sociales. Es posible entender las organizaciones por medio del análisis de la

transferencia de la información entre los elementos de un mismo sistema o con su entorno. Es en este contexto donde algunas organizaciones tradicionales funcionan como sistemas cerrados, es decir, que están cerrados al mundo exterior, y toda la interacción y el conocimiento se transmite dentro de él, así, el cliente (y su información) no es parte del sistema sino del entorno es por ello que por motivos de dicho trabajo, el hecho de buscar la posibilidad de que los clientes fueran parte de un sistema cada vez más abierto e inclusivo es relevante. Es importante aclarar que la Teoría de los Sistemas Sociales, planteada por Luhmann (1998), no parte de un principio social sino de un principio operacional dentro del sistema, de la forma en que funciona la sociedad simulando procesos internos de dicho sistema y no de las relaciones interpersonales como por ejemplo, formas de convivencia, psicología humana, usos y/o costumbres, lo que limita la participación integral de la sociedad. Del anterior argumento se fundamentan la mayoría de las atribuciones que se le dan al hombre como excluido del sistema en la organización (los stakeholders, al ser nombrados como aquellos grupos de interés, y por lo tanto, reconociendo su potencial dentro de la organización, son una forma de inclusión de algunos elementos externos al sistema en la organización).

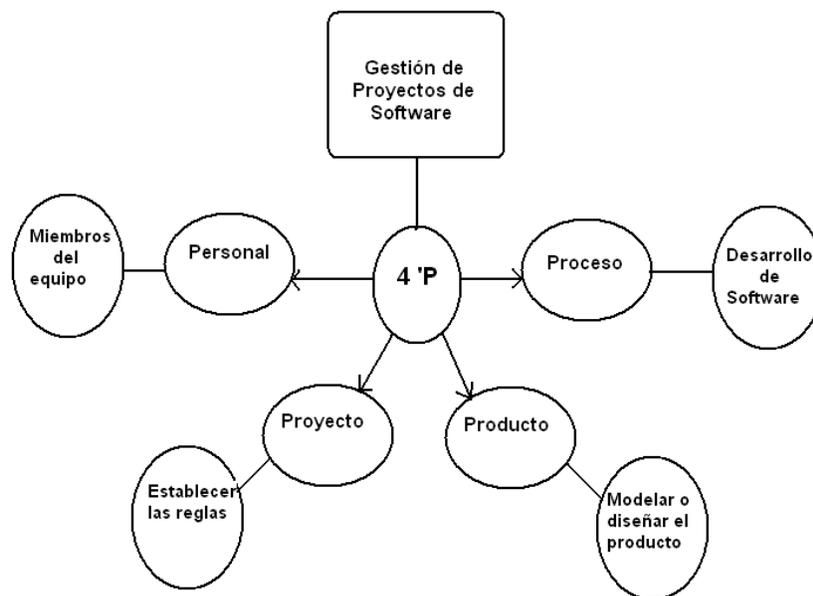
En el campo de los sistemas de información y comunicación, la forma en que interactúan los elementos del sistema se ve reflejada en como la transferencia en forma de bits o de datos, por medios que se constituyen con redes computacionales compuestas por componentes físicos, lógicos y que por medio de protocolos de comunicación logran la transferencia de la información de un lugar a otro en forma de bits, almacenada y procesada en un sistema de información, en este caso todo el proceso y funcionamiento de la organización se fundamenta en la información que posee o que puede obtener. El cliente, dentro de ésta definición, está representado por todas sus características, preferencias y necesidades manifestadas en la solución de un problema organizacional, la optimización de procesos tanto internos como externos así como de la información en datos recolectados en cada visita y transacción realizada por medio de un sitio Web o cualquier tipo de transacción por medios electrónicos donde sea plenamente identificado como individuo. Por tal motivo, es imposible excluir al hombre del sistema de información y comunicación, debido a que no actúa como vínculo con otros sistemas, sino que es fundamental para que el mismo funcione, se desarrolle y alcance sus objetivos. Sin las preferencias del cliente registradas por medio de los diversos mecanismos de información y comunicación, no sería posible construir un modelo lógico para la existencia, desarrollo y funcionamiento de los proyectos de software.

Debido al análisis anterior y como parte de entender la relación de los stakeholders dentro de los proyectos de software, es conveniente explicar el proceso de desarrollo de dichos proyectos que por medio de diversas metodologías como CMMI o ISO (que se explicarán más adelante), pueden llegar a culminarse de manera exitosa ya sea en términos de costo, tiempo o esfuerzo, o bien fracasar dependiendo de la naturaleza con la que fueron concebidos.

Un proyecto de sistemas computacionales o software es el proceso para la creación de un sistema de información, el cual encierra un conjunto de actividades, una de las cuales es la estimación, cuya importancia dentro del desarrollo de un proyecto es muy importante. Existen técnicas útiles para la estimación de costos de tiempo y dado que la estimación es la base de todas las demás actividades de planificación del proyecto, servirá como guía para una buena ingeniería de sistemas y software (*Brown, 2010*).

Al estimar el tiempo de desarrollo, se toma en cuenta no solo el procedimiento técnico a utilizar en el proyecto, si no que se toma en cuenta los recursos, los costos y la planificación. El tamaño del proyecto en términos de código desarrollado y procesamiento de información, es otro factor importante que puede afectar las estimaciones. A medida que dicho tamaño aumenta, crece rápidamente la interdependencia entre varios elementos del software. El objetivo de planificar un proyecto de software es el de proporcionar un marco de trabajo que permita al gestor hacer estimaciones razonables de recursos, costos y planificación temporal. Estas estimaciones se hacen dentro de un marco de tiempo limitado al comienzo de un proyecto de software, y deberían actualizarse regularmente a medida que progresa el proyecto. La figura 1 muestra las partes más esenciales del proceso de gestión de proyectos de software, partiendo de cuatro ejes principales: 1) Personal, que son los miembros del equipo de trabajo, 2) Proyecto que es donde se establecen las reglas, 3) Producto, donde se modelan y diseñan los diferentes componentes y 4) Proceso, donde se hace el desarrollo de Software.

**Figura 1** Gestión de Proyectos de Software (*Calidad, 2011*).



Dentro de las tareas de planificación se encuentran:

1. **Ámbito del Software:** Se deben evaluar la función y el rendimiento que se asigna al software.  
El Ámbito se define como un pre-requisito para la estimación.
2. **Recursos:** Se estiman los recursos requeridos para acometer el esfuerzo del desarrollo de Software.

Cada recurso queda especificado mediante cuatro características: Descripción del recurso, informes de disponibilidad, fecha cronológica en la que se requiere el recurso y tiempo durante el que será aplicado el recurso.

Los diferentes recursos son:

- **Recursos Humanos:** La cantidad de personas requeridas para el desarrollo de un proyecto de software sólo puede ser determinado después de hacer una estimación del esfuerzo del desarrollo.
- **Recursos o componentes del software reutilizables:** Cualquier estudio sobre recursos de software estaría incompleto sin estudiar la reutilización, esto es la creación y reutilización de bloques de construcción de software.
- **Recursos de Entorno:** En el entorno es donde se apoya el proyecto de Software, llamado a menudo entorno de Ingeniería de Software el cual incorpora Hardware y Software.

En cuanto a los modelos de estimación de Proyectos de Software existen:

- **Los Modelos Empíricos:** Donde los datos que soportan la mayoría de los modelos de estimación obtienen una muestra limitada de proyectos. Por esta razón, el modelo de estimación no es adecuado para todas las clases de software.
- **El modelo COCOMO:** Es un modelo constructivo de costos y se divide principalmente en tres partes:
  1. Modelo I: El modelo COCOMO básico calcula el esfuerzo y el costo del desarrollo de software en función del tamaño del programa, expresando en las líneas estimadas.
  2. Modelo II: El modelo COCOMO intermedio calcula el esfuerzo del desarrollo del software en función del tamaño del programa y de un conjunto de conductores de costos que incluyen la evaluación subjetiva del producto, del hardware, del personal y de los atributos del proyecto.
  3. Modelo III: El modelo COCOMO avanzado el cual incorpora todas las características de la versión intermedia y lleva a cabo una evaluación del impacto de los conductores de costos en cada caso (análisis, diseño, etc.) del proceso de ingeniería de software.

Las métricas para la productividad y calidad del Software se refieren a un amplio rango de medidas para el software de computadoras dentro del contexto de la planificación del proyecto del software. Las métricas de calidad pueden ser aplicadas a organizaciones, procesos y productos las cuales directamente afectan a la estimación de los costos.

Existen diferentes tipos de métricas:

- **Métricas de Productividad:** Se centran en el rendimiento del proceso de la ingeniería de software.
- **Métricas de Calidad:** Proporcionan una indicación de cómo se ajusta el software a los requerimientos implícitos y explícitos del cliente.
- **Métricas Técnicas:** Se centran en el carácter técnico del software más que en el proceso, a través del cual el software ha sido desarrollado.
- **Métricas orientadas al tamaño:** Son utilizadas para obtener medidas directas del resultado y la calidad de la ingeniería de Software.

- **Métricas orientadas a la función:** Son medidas del software y del proceso por el cual se desarrollará; se centran en la funcionalidad o utilidad del programa (Puntos de Función)
- **Métricas orientadas a la persona:** Son métricas que consiguen información sobre la forma en que la gente desarrolla software de computadora y sobre el punto de vista humano de la efectividad de las herramientas y métodos.

### 1.3 INTEGRACIÓN DEL MODELO DE MADUREZ DE CAPACIDADES (CMMI) COMO PROCESO DE MEJORA EN LOS PROYECTOS DE SOFTWARE

Existen diversos factores que pueden influir en el éxito o el fracaso de proyectos de software (éxito o fracaso en términos de costo, tiempo y esfuerzo en el desarrollo de proyectos de software) como por ejemplo la asignación de recursos por parte de las organizaciones para la realización de nuevos proyectos de software, las barreras políticas y económicas, la relación entre organizaciones y las partes interesadas, la cultura organizacional, los valores éticos y morales, así como los intereses de los stakeholders entre otros. Tomando en consideración la relación entre las organizaciones y los stakeholders, existe la necesidad de interacción mutua entre los diversos grupos de interés considerando que dichas organizaciones priorizan las actividades más importantes, de entre las cuales están la responsabilidad social colectiva, la generación de valor y utilidades para los accionistas, esto sin dejar a un lado la sostenibilidad de la empresa y su supervivencia a largo plazo (*Suárez, 2009*).

Para poder alcanzar estos objetivos, se hace necesaria la presencia de diversos agentes económicos que aportan recursos a las empresas de software con el fin de contribuir a los fines de la misma, y a cambio, recibir una compensación (desarrollo de proyectos de tecnologías de la información y comunicación) que satisfaga sus necesidades e intereses y que, además, incentive a los agentes a seguir participando en la organización. En ese sentido, podemos decir que los agentes tienen en cierto modo derecho a participar en la distribución de la riqueza generada en las organizaciones como consecuencia de su operación en la economía, recibiendo de esta manera beneficios que van más allá de los acordados con las organizaciones, y que en general se orientan hacia el aumento en la calidad de vida y el desarrollo sostenible, incluyendo mejoras en los aspectos económico, social y ambiental.

Los stakeholders participantes en el desarrollo de proyectos de software, que garantizan la satisfacción de las necesidades y los intereses de los clientes, utilizan diversas metodologías y métodos para mejorar y optimizar los recursos que propicien el éxito de los mismos. Un ejemplo de ello es la integración del Modelo de Madurez de Capacidades (CMMI por sus siglas en inglés) el cual tiene la función de considerar a los stakeholders remarcables dentro del proyecto en cuestión y mejorar procesos metodológicos y técnicos, proporcionando a las organizaciones recursos para aumentar la eficacia de dichos procesos cuyas mejores prácticas se publican en los documentos llamados modelos.

La implementación de CMMI dentro de los proyectos de software puede ser aplicada en proyectos llevados a cabo en fábricas de software o también llamadas áreas de diseño de software en las Consultorías de TI, a proyectos internos o externos o a empresas enfocadas a proveer soluciones en tecnología Web, como el desarrollo de portales, aplicaciones de

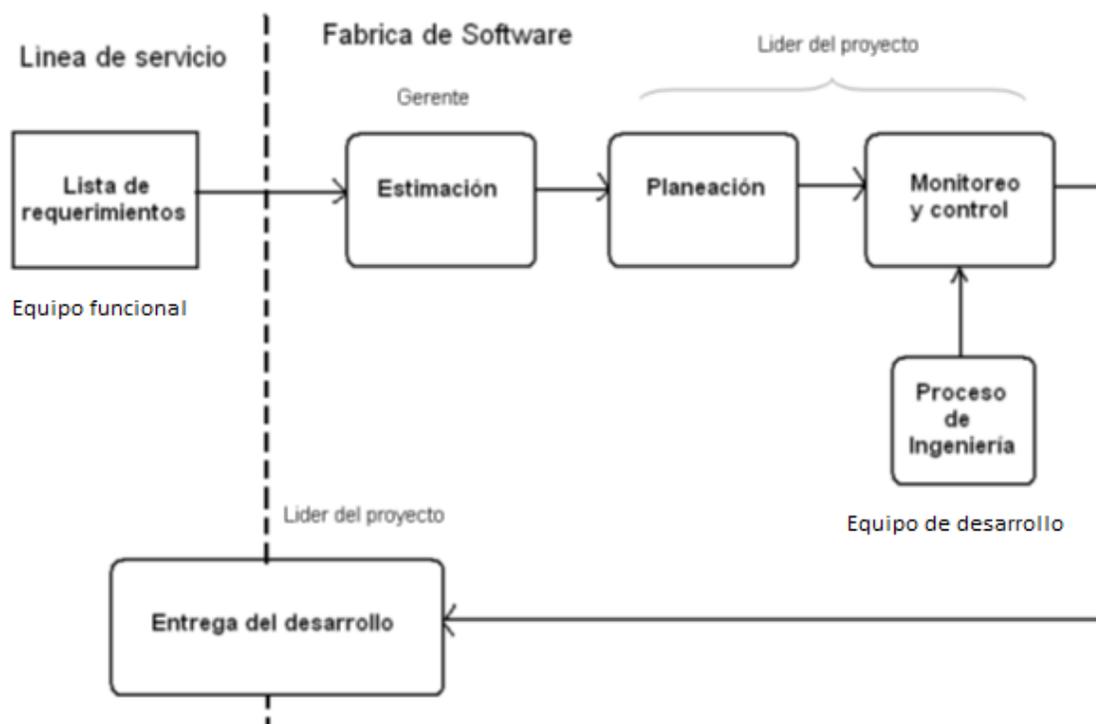
comercio electrónico, administradores de contenidos, aplicaciones administrativas, financieras, de atención a clientes y operativas, sistemas inalámbricos, etc.

El proceso de desarrollo de un proyecto de software puede variar dependiendo de la empresa o fábrica de software a fin. El presente estudio de fases de desarrollo de software y el posterior estudio de las situaciones referentes a proyectos de software con los stakeholders, podrán ayudarnos a entender los puntos en los que podríamos intervenir para mejorar dicho proceso. Considerando los procedimientos establecidos por la metodología CMMI: *“Los proyectos de software que siguen el proceso de CMMI tienen etapas las cuales se les asigna un responsable que las lleve a cabo. Este responsable tiene regularmente un equipo que colabora con él para hacer las actividades correspondientes de dicha etapa. Los documentos que surjan de cada una de las etapas están controlados con el proceso de CMMI y deben formar parte de la evidencia que la fábrica y el cliente requiere”* (Vargas de la Torre, 2009).

Las etapas del proceso de administración de proyectos de software se muestran en la figura 2 con la finalidad de conocer los procesos con los que la fábrica debe opear después de la aprobación del proyecto siguiendo la metodología de CMMI.

### Proceso general de definición

**Figura 2** Proceso de administración de proyecto y stakeholders involucrados (Deloitte México, 2012)



Al cliente, al **Aceptar la Propuesta**, se le indica la fecha de **Levantamiento de Requerimientos** a detalle, en donde se hace la toma de información relevante, tipos de gráficos a utilizar, y con dicha información se realiza un documento llamado especificación

funcional (EF) y uno de control de riesgos que sirven para tener los detalles del proyecto. La lista de requerimientos resultante se transmite al **Gerente** de la fábrica de software para llevar a cabo el proceso de **Estimación** donde se calcula el esfuerzo y el costo del desarrollo de software en función del tamaño del programa, expresando en las líneas estimadas, presupuestos, recursos, etc.

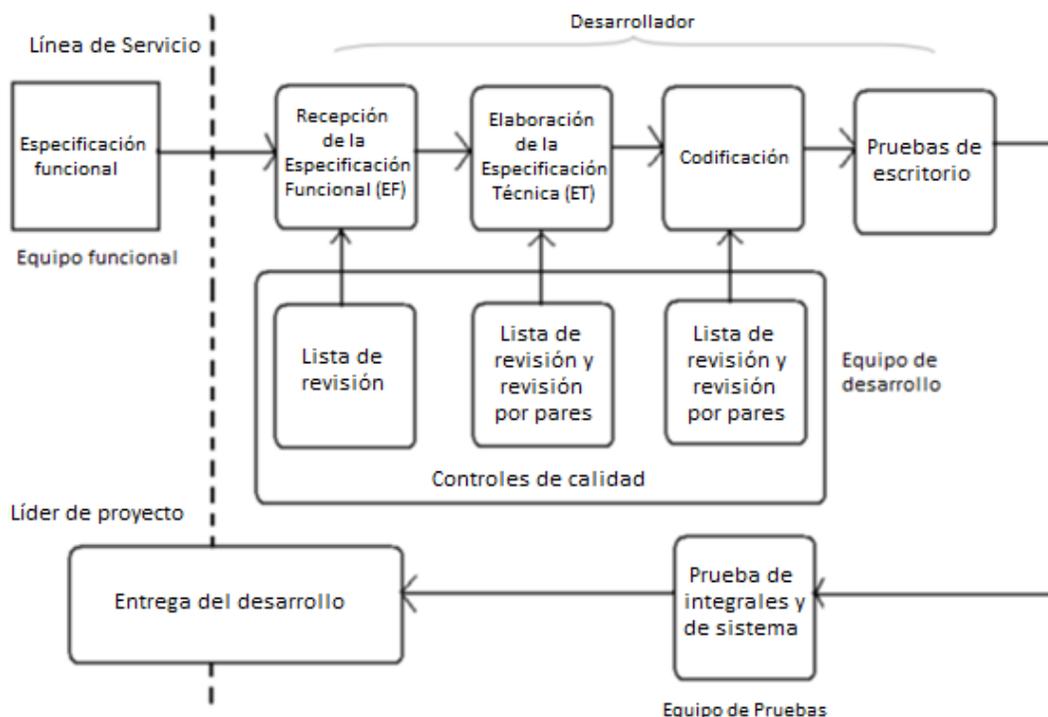
Con dicha información el **Líder de Proyecto** puede desarrollar la **Planeación** correspondiente o un **Plan de trabajo** general de principio a fin del proyecto con el que se realizan las revisiones en las fechas indicadas en este. Es complicado encontrar un proyecto de desarrollo de software en donde se sepa en dónde se va a iniciar y en dónde va a terminar exactamente. La mayoría de los proyectos no son lineales (proyectos donde se tiene planeado claramente el inicio y el fin del proyecto), sino cíclicos, es decir, el cliente quiere ver avances en el desarrollo de su sistema y conforme lo va viendo va teniendo nuevos requerimientos, lo que finalmente hace que el proyecto se convierta en uno denominado cíclico.

Es importante mencionar que el Líder del Proyecto además de llevar a cabo una Planeación adecuada al proyecto de Software, debe llevar a cabo actividades de **Monitoreo y Control**, las cuales por medio del proceso de CMMI, ayudan a tener un registro de las actividades correspondientes al **Proceso de Ingeniería** o de **Construcción**. El plan de trabajo generado, al ser aceptado por el cliente, indica claramente la fecha de inicio para que se respeten las fechas que se manejan en el documento y que servirán como compromiso formal de los tiempos comprometidos en los mismos, iniciando así la etapa de **Construcción** del proyecto.

### Proceso general de Ingeniería

Seguido del proceso general de Definición comienza el Proceso General de Ingeniería de la Fábrica de Software.

**Figura 3** Proceso de Ingeniería y stakeholders involucrados (Deloitte México, 2012)



## *Del diagrama*

En la figura 3 se muestra la fase de ingeniería, en donde, a partir del proceso general de administración de proyecto que se realizó, se hace uso de la Especificación Funcional proveniente de la lista de requerimientos inicial y desarrollada en conjunto entre el Gerente y el Líder de Proyecto. En este proceso, el **Desarrollador** cobra vital importancia dentro de muchas fases del mismo ya que recibe la **Especificación Funcional (EF)** y a partir de ese momento entra en un proceso de **Control de Calidad** siguiendo con el proceso de CMMI. El Desarrollador tiene la obligación de realizar un formato de revisión para controlar y registrar si la EF cuenta con los requisitos mínimos establecidos. Una vez revisada la EF se procede a la elaboración de la **Especificación Técnica (ET)** la cual es un formato que contiene los requerimientos funcionales traducidos a especificaciones técnicas o entendibles técnicamente para el desarrollador de software.

De igual manera se realiza una revisión en forma de lista (check list) de revisión y al mismo tiempo una revisión entre colegas (peer review) del mismo documento. El peer review es una técnica para asegurarse que otra persona o desarrollador entienda e interprete el check list de la misma manera en que el responsable principal lo interpretó.

Una vez teniendo la especificación funcional y la especificación técnica, el desarrollador puede comenzar con el proceso de **Codificación** el cual al igual que la EF y ET se realiza una lista de revisión y una revisión entre colegas. El desarrollador tiene la responsabilidad de llevar a cabo las **Pruebas de Escritorio**, las cuales son pruebas individuales del desarrollo codificado. La dinámica de las pruebas de escritorio puede ser exclusiva del desarrollador pero el resultado de dichas pruebas debe coincidir a lo establecido y demandado por las EF.

Cuando los diferentes miembros del equipo de trabajo terminan de desarrollar los componentes correspondientes, se integran en el sistema y se llevan a cabo las pruebas de funcionalidad por medio de un encargado de ejecutar las pruebas (**tester**). Estas pruebas se dividen en:

- **Pruebas de ensamble**

El equipo de desarrollo prueba en conjunto sus piezas de código.

- **Pruebas de fábrica**

El Líder de Proyecto prueba la funcionalidad del sistema de acuerdo a los módulos que se le ofrecieron al cliente.

- **Pruebas de usuario**

Pruebas finales por el cliente. Se vuelven muy confiables al haberse llevado previamente las pruebas de ensamble y las de fábrica.

Una vez terminado el proceso de pruebas el Líder de proyecto puede entregar el desarrollo de software, ya sea a los gerentes o en conjunto para el cliente solicitante.

## 1.4 PROBLEMÁTICA EN EL DESARROLLO DE PROYECTOS DE SOFTWARE

Se ha mencionado en múltiples ocasiones que en las organizaciones encargadas en desarrollar sistemas de información buscan brindar las mejores experiencias a sus clientes por medio de la implementación de procesos de mejora, así como la implementación de mejores estándares de calidad, pudiendo quizá arriesgar procesos secundarios que si bien no afectan directamente a la culminación de los proyectos, pueden ser las causas de que un proyecto de software fracase de manera rotunda pese a los esfuerzos realizados en la satisfacción del cliente.

Tomando en cuenta los diversos estándares de calidad que las empresas en software utilizan para mejorar sus procesos y por consiguiente una mayor satisfacción del cliente, se puede mencionar por ejemplo el proceso de mejora denominado modelo de madurez y capacidad integrado (CMMI por sus siglas en inglés), el cual es un modelo sistemático de actividades que deben ser cumplidas por los miembros de la organización para mejorar los procesos señalados por dicha metodología. Debido al cumplimiento de las actividades marcadas, se presentan diversos inconvenientes como lo son, la priorización de las actividades fundamentales que debieran tener mayor peso en el desarrollo de un proyecto pero que no se les da la prioridad adecuada debido a que dicha organización debe cumplir al pie de la letra los pasos del proceso de mejora implementado o bien que las personas involucradas en el proyecto presentan deficiencias en cuanto a la estructura de roles, funciones, responsabilidades, etc.

Existen riesgos latentes en relación a las organizaciones, en este caso, desarrolladoras de software que tienen sus orígenes en los grupos de personas interesadas en realización y culminación de más de una actividad o proceso referente a la organización. Por un lado se puede decir que la *empresa* depende del **estado**, para que este le brinde seguridad y condiciones políticas, económicas y sociales adecuadas para el desarrollo y la supervivencia de la organización, pero el estado depende de la organización como fuente de recursos que alimenten el erario y que le permitan sostener sus gastos de funcionamiento principalmente; de los **proveedores**, porque ellos le suministran insumos vitales para la producción de sus bienes o prestación de servicios, y la estabilidad de éstos depende en gran parte de la capacidad de la empresa para cumplir con los pagos y demás compromisos adquiridos; de los **clientes**, porque estos consumen los bienes o servicios producidos y son su fuente principal de ingresos, a la vez que se genera dependencia por la capacidad de la empresa para seguir operando y satisfacer las necesidades de los **consumidores**; de los **acreedores** y **sector financiero**, porque estos le proporcionan capital financiero y plazos de pago para el desarrollo de sus actividades y mantenimiento de liquidez; asimismo asumen el riesgo y la incertidumbre por la capacidad de la empresa de cumplir con el pago del principal y de los intereses; de los **empleados**, porque estos aportan trabajo, capacidades, conocimiento y experiencia que son un instrumento de gran relevancia para el desarrollo del objeto de las empresas (los proyectos de software), pero los empleados asumen el riesgo de que la empresa no sea viable económicamente y por tanto no tenga liquidez para realizar los pagos de nómina y demás compensaciones que le permitan la satisfacción de sus necesidades básicas y el mantenimiento de cierto nivel de vida; y de los **accionistas potenciales**, quienes pueden convertirse en fuentes futuras de recursos, a la vez que estos asumen el riesgo por la buena gestión organizacional que puedan realizar los **gerentes** y que va ligada al precio de las acciones y la generación de beneficios que puedan ser distribuidos como dividendos; sin embargo, los **accionistas** tienen la capacidad de disminuir y gestionar su riesgo a través de la

diversificación (Suárez, 2009).

Debido a la complejidad presentada anteriormente se puede decir que existen factores en el proceso de desarrollo de software que tienen que ver con los grupos de interés asociados a los proyectos y su impacto referente a la supervivencia de empresas de software, al igual que sobre el desarrollo de la sociedad y de la economía. Los intereses que los grupos de interés o stakeholders tienen en la empresa estarán ligados a los aportes que estos realicen a la misma para que pueda desarrollar su actividad económica; por consiguiente, se hace necesario referenciar la forma de regulación en estas relaciones y la manera en que se equilibran los intereses de los diversos stakeholders y de la organización. En relación a los intereses de los stakeholders y de las organizaciones, existe una gran variedad de causas por las cuales dichos intereses pueden llegar a afectar al desarrollo de un proyecto de software como por ejemplo la conformación de los contratos; los elementos del mismo, el alcance en las acciones de cada una de las partes, los incentivos, principalmente para la gerencia, entre otros.

Como podemos percibir, un contrato entre dos partes es algo complejo, principalmente por la divergencia de intereses entre los participantes. El enfoque de los stakeholders, que ha llegado a denominarse teoría de la agencia ampliada, por la gran variedad de agentes económicos con quienes tienen relación las empresas y sus accionistas, puede aumentar la complejidad de este problema, debido a que, como se ha explicado en líneas anteriores, la dependencia es mutua, los intereses son divergentes y, por tanto, deben existir los contratos para equilibrar los mismos y para permitir alinear las actuaciones de los diversos stakeholders y orientarlas hacia la consecución de los fines de la empresa desarrolladora de software, ubicando la generación de valor y utilidades como los aspectos más relevante.

Los contratos entre los stakeholders y las organizaciones pueden llegar a ser una barrera entre la implementación de buenas prácticas, metodologías ágiles y novedosas, herramientas tecnológicas y cambios dentro de la organización. Los contratos establecen los recursos que recibe la organización para que pueda operar y los derechos sobre los mismos a favor principalmente de los diversos stakeholders. En ese sentido, los intereses entre organización y stakeholders son mutuos; por lo general se presentarán conflictos de intereses entre las partes y los contratos cumplen el papel de equilibrar intereses, se convierten en un instrumento de cooperación y buscan asegurar el cumplimiento de los compromisos adquiridos libremente entre las partes. Vale la pena destacar que una vez se han repartido las diferentes compensaciones para la diversidad de agentes (derechos incluidos en los contratos o acuerdos), existirán unos residuales que van a ser disputados por los grupos de interés; pero la ventaja sobre los mismos la tendrán los administradores (gerencia), quienes tienen mayor conocimiento y disponibilidad de información y por consiguiente pueden ser un factor de cambio que priorice la potencialización de la implantación de sistemas de información y comunicación que sirvan para la culminación exitosa de los proyectos de software (Suárez, 2009).

Como resultado de la identificación de acciones de los stakeholders asociados a los proyectos de software se considera que los factores relacionados al éxito o fracaso de los proyectos de software pueden visualizarse en dos diferentes enfoques:

1. Stakeholders externos: Gobierno, accionistas, proveedores, inversionistas, etc.
2. Stakeholders internos: Gerentes, Trabajadores, clientes, consumidores, propietarios, etc.

Los stakeholders *externos* tienen una relación de compromiso con el proyecto de software, generalmente aportan valor monetario relacionado con intereses externos o bien globales entre otros stakeholders externos, también son caracterizados por presentar intereses sociales y económicos tanto nacionales como internacionales.

Los stakeholders *internos* presentan un nivel de responsabilidad superior en el diseño, construcción y desarrollo del proyecto en cuestión por lo que sus intereses están directamente relacionados con el desempeño de su trabajo, su conocimiento invertido y en la relación entre otros stakeholders internos de la organización.

Debido al nivel de responsabilidad entre los stakeholders internos y externos, surgen diferentes cuestionamientos referentes a la diversidad de intereses y al impacto que puedan tener en el desarrollo de software. Por un lado el enfoque hacia los stakeholders externos relaciona conflictos enmarcados con actividades políticas y gerenciales impactando el desarrollo de los proyectos desde un punto de vista superior al del enfoque de stakeholders interno donde los conflictos se relacionan mayoritariamente con prácticas internas, metodologías, valores, uso de conocimientos técnicos y forma de relacionarse.

Por lo tanto se considera que un estudio adecuado de las relaciones existentes entre los stakeholders, los problemas que pueden surgir en el proceso de interacción así como los contratos y beneficios que tengan en la organización son clave para desarrollar un análisis que pueda ayudarnos a entender cómo es que los stakeholders actúan y pueden ayudar a mejorar el proceso de desarrollo de software ya que es importante estudiar la influencia de los stakeholders en proyectos de software para determinar las principales dificultades que se pueden presentar durante el desarrollo de proyectos de software, resultado de diversos factores funcionales, técnicos y estructurales y cuyos análisis puede ayudar a entender cómo se puede llegar al planteamiento, desarrollo y culminación de un proyecto de software de manera exitosa.

## **1.5 PLANTEAMIENTO DEL PROBLEMA Y OBJETIVO DE LA TESIS**

Como resultado de la problemática presentada en la sección anterior, el planteamiento del problema de esta tesis es el siguiente:

Se considera que la estructura funcional y técnica de stakeholders dentro de las empresas desarrolladoras de software tiene un impacto en el éxito y el fracaso de sus proyectos de software si sus intereses, ideologías, forma de comunicación, valores y actividades están orientadas a beneficiar dichos proyectos pero por el contrario si dichos esfuerzos son orientados a satisfacer intereses distintos a los del proyecto como formas de comunicación deficientes, carencia de relaciones entre stakeholders, practicas internas carentes de metodología, grupos de poder, entre otros, el proyecto puede no culminar de la mejor manera aumentando las probabilidades de fracaso.

El **objetivo** general del presente trabajo es el siguiente:

Realizar un análisis de estructuras de stakeholders en empresas desarrolladoras de software por medio de estructuras asociadas a sus roles, funciones y responsabilidades sobre el éxito (costo, tiempo, satisfacción, etc.) de los proyectos a partir de elementos clave para proponer mecanismos de control así como acciones planeadas de mejora.

Los objetivos específicos del presente trabajo son los siguientes:

- 1) Identificar y analizar la estructura de los stakeholders (Roles, funciones, responsabilidades, intereses, etc.) en los proyectos de software.
- 2) Valorar el impacto de la(s) estructura(s) en los proyectos de desarrollo de software por medio de la identificación de factores que influyen en su éxito o fracaso.
- 3) Proponer mecanismos de control que fortalezcan a los factores críticos de éxito (FCE).

## **1.6 JUSTIFICACIÓN Y ALCANCE**

Hoy en día, las organizaciones que tienen como actividad comercial primaria el desarrollo de tecnologías de la información y comunicación se encuentran cada vez más interesadas en la satisfacción del cliente, generar valor y utilidades para sus accionistas, esto sin dejar a un lado la sostenibilidad de la empresa y su supervivencia a largo plazo.

Dentro del estudio de la gestión de proyectos de software existe una amplia variedad de factores que son clave para la culminación y éxito de los mismos como lo son:

- Estimación de proyecto
- Tiempo de desarrollo
- Herramientas utilizadas
- Aprovechamiento de la información interna de la empresa
- Aprovechamiento de la experiencia
- Personas involucradas con el proyecto
- Contratos entre organizaciones y stakeholders
- Barreras comerciales y políticas
- Cultura organizacional

Tanto las empresas desarrolladoras de proyectos de software como a sus clientes (organizaciones, estado, negocios, personas, etc.) están interesadas en potencializar sus procesos y por consiguiente puedan generar valor y una retroalimentación a sus actividades y su vida en general.

Es importante que dentro de dicha satisfacción se reduzcan al máximo las pérdidas generadas durante el proceso de desarrollo de proyectos de software que entre otras puede incluir:

- Perdida de dinero invertido.
- Aumento del re trabajo
- Mala calidad en los desarrollos
- Empeoramiento en la calidad de vida de las partes involucradas.
- Retrasos
- Penalizaciones por parte del cliente a las organizaciones desarrolladoras de software
- Penalizaciones a los trabajadores encargados del desarrollo de los proyectos de software
- Mala imagen de las organizaciones
- Disminución de la responsabilidad social
- Cancelación del Proyecto

Por lo que el desarrollo de este trabajo ayudaría a plantear una propuesta para detectar y analizar los principales factores por los cuales un proyecto de software puede cumplir su objetivo principal por lo cual es diseñado así como ayudar a conocer y entender la implantación de tecnologías de la información y comunicación mediante un análisis adecuado de sus stakeholders.

## CAPITULO 2

# REVISIÓN DE FACTORES ASOCIADOS AL ÉXITO Y FRACASO DE LOS PROYECTOS DE SOFTWARE DESDE UNA PERSPECTIVA DE STAKEHOLDERS

Para poder determinar la importancia de los stakeholders en proyectos de software y encontrar la relación que existe entre ellos con el objetivo de determinar los factores que repercuten en el éxito o fracaso de éste tipo particular de proyecto, es necesario realizar un estudio teórico que sirva como marco de diseño para una propuesta metodológica orientada a proyectos de software por medio del análisis de sus stakeholders.

Como primer acercamiento, es necesario definir y analizar a los proyectos de software por medio de la problemática surgida en su desarrollo, perfiles y casos de estudio para encontrar aquellos factores de éxito que establezcan lineamientos útiles para el presente trabajo. Por otro lado es necesario definir el término de stakeholders tomando en cuenta la influencia que puedan tener en las organizaciones y por consiguiente en el desarrollo de software. Es necesario también ejemplificar el enfoque a stakeholders que se presenta en proyectos actuales para destacar su importancia y de igual manera analizar las diferentes herramientas metodológicas que existen en la actualidad usadas por las organizaciones desarrolladoras de software para mejorar sus procesos organizacionales.

Es importante también analizar los principales componentes por los que se puede diseñar una estructura organizacional usando el enfoque de stakeholders y proyectos de software para encontrar los mecanismos organizacionales que puedan ayudar a impactar de forma más efectiva a dichas organizaciones. Una vez, habiendo analizado a los proyectos de software, los stakeholders y las estructuras organizacionales, se recurre a enfocar esfuerzos en orientar el presente trabajo al uso de los principios sistémicos, de planeación u organizacionales que puedan servir como un marco de diseño para futuras implementaciones, por lo que se hace uso de un enfoque interactivo de planeación el cual está orientado a determinar diferentes acciones organizacionales, entre ellas, el enfoque a stakeholders.

Para poder contar con estrategias innovadoras, es necesario hacer uso de tendencias corporativas y organizacionales para entender la manera en que los esfuerzos realizados en la orientación a stakeholder puedan tener efectos en la cultura organizacional y el cambio por lo que se recurre a analizar a la Responsabilidad Social Corporativa (RSC) como marco conductor de cambio organizacional. Por último, se plantea la estrategia de investigación como resultado de la revisión de distintos factores en el desarrollo de software con relación a los stakeholders que pueda orientar los esfuerzos a crear una propuesta capaz de plantear recursos útiles para una gestión adecuada de stakeholders en proyectos de software.

## 2.1 DESARROLLO DE SOFTWARE

### 2.1.1 DEFINICIÓN DE PROYECTO DE SISTEMA O SOFTWARE

El software o programa de computadora se puede entender como una serie de órdenes lógicas escritas en un lenguaje técnico diseñado para poder ser leído y ejecutado de forma secuencial por dispositivos electrónicos. Tiene por objetivo principal, procesar información proveniente de distintas fuentes para ejecutar diversas acciones como como por ejemplo el análisis matemático, financiero, social, entre otros. El desarrollo de software es una actividad donde se involucra la ingeniería; que es el “estudio y aplicación, por especialistas, de las diversas ramas de la tecnología” (*Real academia española de la lengua, 2014*). El desarrollo de sistemas de software más que la programación de computadoras, es un proceso que involucra a la ingeniería de software que comprende todos los aspectos de la producción de software desde las etapas iniciales de la especificación del sistema (*Somerville, 2005, p.6*).

Desarrollar un software significa construirlo simplemente mediante su descripción, lo que hace a la actividad de desarrollo de software tener el título de ingeniería. Por lo que un producto de software no se limita solo a programas de computadora sino a documentación asociada a éste en las distintas etapas que interviene desde su concepción, análisis, diseño, implementación, pruebas y mantenimiento. El inicio de un proyecto, cualquiera que sea, es la parte medular del mismo y es un factor importante a considerar. Si la naturaleza del proyecto es en términos tecnológicos, específicamente en proyectos de desarrollo de software, se deben tomar en cuenta las etapas de dicho proyecto. Se dice que el inicio de un proyecto de software no es la programación. Gran parte de la suerte de un proyecto se juega antes del comienzo del mismo por diversas razones, las cuales propician que no se culmine de manera exitosa. Factores en términos de negociación, organizacionales y de venta condicionan el proceso de desarrollo produciéndose únicamente circunstancias iniciales: un equipo de proyecto motivado y de calidad, un cliente y usuarios que facilitan las tareas, pueden paliar las pérdidas del proyecto, tangibles (en dinero), como intangibles (pérdida de credibilidad con el cliente ante un proyecto de ejecución deficiente).

Aparte de la variable de venta del proyecto que puede afectar seriamente la realización del mismo, están la elección de la tecnología, ya sea por el cliente o por el proveedor y si ésta cumple con los requerimientos en tiempo, costo y resultados la cual puede determinar diversos retrasos si la elección no fue llevada a cabo de la mejor manera provocando re-trabajos en la mayoría de las actividades de desarrollo. Por otro lado en términos del desarrollo del software, la arquitectura es de vital importancia. El software no estaría completo si no existiera una especificación de requerimiento o un diseño de arquitectura y la elección de la(s) persona(s) encargadas del diseño arquitectónico del software cobra vital relevancia. Si la metodología es ágil o, de menos interactiva incremental, las tareas de análisis también son de gran importancia, aunque desarrollos iterativos y evolutivos dan un mayor margen para la corrección de deficiencias en la captura e interpretación de los requisitos y en el modelado de la solución.

Si el proceso de análisis de requerimientos provocó consecuencias negativas en etapas subsecuentes, en muchos casos, cuando el producto ha llegado a producción, existe la probabilidad de fracaso en la puesta en marcha del sistema y un esfuerzo económico importante para resolver esta circunstancia, mediante mantenimientos correctivos y evolutivos, que podrían ser incluso no asumibles si la deuda en términos técnicos o de construcción del desarrollo es elevada.

Si el diseño del producto carece de una estructura funcional, existe la probabilidad de que se incrementen las consecuencias en el resultado final del proyecto de software. Por tal motivo, se destaca que además de contar con programadores talentosos, disponer de analistas eficientes, entre otros factores humanos, permite que el trabajo grupal tenga mejores resultados y además sea útil el esfuerzo. De ahí la importancia de realizar la etapa de pruebas desde el principio y no al final como se ejecuta en la gran mayoría de procedimientos internos de empresas de software.

Una de las actividades primordiales en términos de la culminación del proyecto de software tiene que ver con las actividades destinadas a la estimación del proyecto, es decir, visualizar de manera rápida el futuro y aceptar el grado de incertidumbre que se obtenga por medio de diversas prácticas ya sea ejecutadas por el administrador de proyecto o bien por herramientas informáticas diseñadas para dicho fin. Aunque la estimación es una actividad cognitiva resultado de la experiencia profesional más que científica, es una actividad importante que no debe llevarse a cabo de forma descuidada. Existen técnicas útiles para estimar los costos de tiempo en los proyectos. Dado que la estimación es la base de todas las demás actividades de planificación del proyecto, sirve como guía para una buena ingeniería de sistemas y software.

Al estimar tomamos en cuenta no solo los procedimientos técnicos a utilizar durante la realización del proyecto, si no que se toma en cuenta los recursos, costos y planificación. El tamaño del proyecto es otro factor importante que puede afectar la precisión de las estimaciones. A medida que el tamaño aumenta, crece rápidamente la interdependencia entre varios elementos del software. La disponibilidad de la información histórica es otro elemento que determina el riesgo de la estimación, por lo que la comunicación que se establezca entre las partes interesadas tanto en el proceso como en la venta del desarrollo de software irá cobrando importancia a medida en que avanza la culminación del mismo.

Por lo tanto, el hecho de no realizarse de manera adecuada las etapas anteriores, el proyecto habrá fracasado antes de que culmine la fecha de entrega, a pesar de que se piense que la parte fundamental es la construcción de software. Es por ello que es necesario entender y desarrollar una definición propia de **éxito** en proyectos de desarrollo de software rescatando los factores más importantes del análisis anterior. Se puede decir que el éxito de un proyecto de software consiste en tener un proyecto a tiempo en costo y expectativas de ambas partes, con el cliente asociado satisfecho por el alcance, funcionalidad y servicio y que el proveedor (organización encargada en el desarrollo de software) obtenga la remuneración económica esperada además de recomendaciones, más oportunidades de proyectos, prestigio aprendizaje, etc. Por lo que además de incluir una buena administración del mismo, se deben considerar por consiguiente, cuatro factores de éxito de proyectos de software que resumen los requerimientos mencionados (Cavazos, 2008):

- 1.- Recursos (Capacidad y experiencia)
- 2.- Metodología (Forma de trabajo)
- 3.- Tecnología (Plataforma tecnológica)
- 4.- Negociación (Tiempo y presupuesto)

## 2.1.2 TIPOS DE DESARROLLO DE SOFTWARE

### 2.1.2.1 SOFTWARE LIBRE

El software libre es un desarrollo que presenta características especiales y que lo diferencian de otros tipos de desarrollo como lo son: las personas que lo desarrollan, las organizaciones que lo apoyan, los intereses por los cuales es desarrollado, el fin con el que es diseñado, los clientes a los cuales va dirigido entre otros.

Al igual que distintos tipos de desarrollo de software, el desarrollo de software libre se puede diferenciar por las siguientes características:

- Supone la creación privada conocimiento libremente disponible para toda la sociedad.
- Genera valor económico (no incluido mayoritariamente en estados financieros)
- Se identifican grupos de interés e indicadores importantes para la responsabilidad social empresarial (RSE).
- La RSE ayuda a analizar esta estructura ya que se enfoca en actividades dirigidas a generar valor para la empresa, como lo son el análisis interno como la mejora de proyectos de software en términos de tiempo, costo y/o beneficio así como externo como la inclusión y unificación de grupos de stakeholders en dichos proyectos,
- El software libre maneja un entorno colaborativo en el que usuarios y desarrolladores de software son contribuyentes y receptores sin limitaciones ya que las mejoras de todos permite a la industria del software e industrias asociadas a evolucionar, desarrollarse así como mejorarse a gran velocidad

Diversos son los beneficios que pueden ser remarcados además de que se considera notable el hecho de que existen menores costos de adquisición y mantenimiento al ser un producto que por su naturaleza presenta dichos beneficios. También permite reducir el riesgo jurídico asociado al código, transferencia de tecnología a la industria local, estabilidad y seguridad de los sistemas entre otros muchos. Diversas son las instituciones quienes lo utilizan, de entre ellas se pueden identificar:

- 1.- Instituciones académicas
- 2.- Administraciones públicas
- 3.- Empresas
- 4.- Personas individuales.

Se hace una connotación especial, para fines de éste estudio, al rubro de las empresas, las cuales están ubicados los desarrollos de software con herramientas de software libre o privado, ya que dichas empresas producen, en la búsqueda de su beneficio, mayor valor que el que capturan sus actividades. Se estima que al menos un 80% del sector de desarrollo de software emplea algún producto de software libre (Daffa, 2010) y que su contribución a la economía en la Unión Europea es de 450 mil millones de euros anuales por reutilización de código además, entre otros beneficios, se mejora la eficiencia y productividad por reinversión de los ahorros generados y una de las ventajas por demás importantes y que contribuyen a la justificación de dicho estudio es que se reduce la tasa de **fracasos** en proyectos.

### **2.1.2.2 SOFTWARE PRIVADO**

El desarrollo de software privado es la tecnología desarrollada por empresas de software cuyo dominio del producto final es de carácter privado al tener derechos de desarrollo y código fuente. El software privado es empleado por diversas instancias empresariales, de gobierno, instituciones nacionales e internacionales que buscan, entre otros fines, modificar sus procesos adicionando sistemas de información a las actividades que realizan siendo los proyectos de software de los recursos más empleados para administrar la información.

El también llamado software comercial se considera una concepción opuesta al software libre, no por el hecho de tener que pagar dinero para adquirirlo o utilizarlo, sino porque sus productores, a diferencia de los programas denominados “libres”, deciden licenciar sus soluciones conforme a leyes y principios que no contemplan la garantía de ninguna de las libertades anteriores (Macías, 2013).

Dentro del desarrollo de software privado, mayoritariamente realizado por empresas e instituciones privadas, existen ciertas ventajas competitivas que repercuten al igual que las ventajas propiamente del software, es decir, en los equipos de desarrollo o en el ambiente empresarial y de trabajo:

1. Cambio de los requisitos en fases avanzadas
2. Utilización de metodología adecuada
3. El producto tiene una deuda técnica inapropiada
4. Los stakeholders no están educados no solo en la propia realidad del desarrollo de software sino de la propia realidad de los cambios en las organizaciones y en consecuencia de sus personas y de sus procesos

El hecho de que la construcción de software es una actividad altamente volátil (*experimenta muchos cambios*) da como resultado que el enfoque tomado por los equipos de trabajo responsables del proyecto deba ser totalmente orientado al desarrollo de software, lo que hace notar el contexto en que se desarrolla el mismo, es decir, teniendo en cuenta que las condiciones pueden cambiar.

### **2.1.3 ESTUDIOS RECIENTES EN EVALUACIÓN Y DESARROLLO DE PROYECTOS DE SOFTWARE**

#### **2.1.3.1 PERFILES DE ÉXITO / FRACASO DE LOS PROYECTOS DE SOFTWARE**

En los Estados Unidos, se gasta más de 250 billones de dólares cada año en el desarrollo de aplicaciones TI en aproximadamente 175,000 proyectos. El costo promedio de un proyecto de software en empresas grandes es de \$ 2,322,000 dólares; para una empresa mediana es de \$ 1,331,000 dólares y para una empresa pequeña es de \$ 434,000 dólares. Se estipula que una gran parte de dichos proyectos no culminará con éxito. Los proyectos de desarrollo en software atraviesan por una crisis lo cual hace que las pérdidas en tiempo, dinero y recursos humanos sean cuantiosas (Guglielmo, 2013).

En 1986, Alfred Spector, presidente de la Transarc Corporation, realiza un estudio de comparación entre los proyectos de construcción de “bridges” o puentes y proyectos de desarrollo de software con la premisa: Los puentes son usualmente construidos “on-time” y “on-budget” es decir, en tiempo y presupuesto real así como de presentar la característica de ser resistentes y no caer ante los movimientos terrestres. Por otro lado, el software tiene la característica de nunca ser ni “on-time” ni “on-budget”, por el contrario, presenta menos resistencia a los cambios y puede caer ante los movimientos de información.

Una de las principales razones por las cuales los proyectos de construcción de puentes están a tiempo, tienen presupuesto y además son resistentes es el detalle en que son diseñados. El diseño de los proyectos de construcción es uno solo y además es flexible a los cambios de especificaciones. Sin embargo, en un mundo donde la rapidez con la que el ambiente de negocios está cambiando, los diseños únicos no responden a lo que el mundo de los negocios demanda día con día. Es por eso que se debe usar cada vez modelos de diseño más flexibles, razón por la cual los proyectos de software no culminan o no son exitosos.

Existen otras razones por las cuales los proyectos de construcción de puentes son más exitosos que los proyectos de software, los primeros tienen más de 3,000 años de experiencia. Cuando un puente se cae, normalmente se investigan las razones de la falla e inmediatamente se realiza un reporte con los detalles de la misma. Esto no sucede en la industria informática, por el contrario, si existe una falla, generalmente es ignorada o racionalizada.

Entendido el éxito de proyectos como el cumplimiento del alcance, plazos, costo y calidad, el departamento de defensa de los Estados Unidos (DoD), funda el *Software Engineering Institute* (SEI) en la Universidad de Carnegie Mellon, en la idea estudiar esta situación y proponer soluciones. En particular, The Standish Group señala que solo un **16.2%** de los proyectos de IT terminan exitosamente (*The Standish Group, 2013*). Este grupo se apoya en un estudio que trabajó una muestra de 365 encuestados relacionados con organizaciones de tamaños grande, mediano y pequeño de los principales segmentos de la industria en los Estados Unidos. En este estudio se cubrieron más de 10,000 proyectos.

El estudio al cual se dirige The Standish Group está enfocado a identificar:

- El alcance de las fallas en los proyectos de software.
- Los factores relevantes que provocan el fracaso de los proyectos de software.
- Los componentes clave que pueden reducir las fallas en los proyectos de software.

Las investigaciones de The Standish Group muestran un asombroso 31.1% de proyectos que serán cancelados antes de que sean completados. Estudios realizados por la misma institución en años más recientes indican que un 52.7% de los proyectos costarán 189% más de su costo promedio estimado. El costo de estas fallas y del trabajo extra generado son solo la parte inicial de los problemas que se pueden generar en etapas futuras. Los costos de oportunidad perdidos pueden ser incalculables, pero pueden ser estimados en trillones de dólares.

The Standish Group estima que en 1995, compañías americanas y agencias gubernamentales gastarían 81 billones de dólares en la cancelación de proyectos de software. Dichas organizaciones pagarían un extra de 3.59 billones de dólares por proyectos de software que fueron completados de manera exitosa, pero excederán su tiempo estimado de desarrollo.

Por otro lado existe el caso de aquellos proyectos de software que fueron exitosos (16.2%) y que cumplieron con características como estar a tiempo y tuvieron los recursos necesarios para su implementación. En las empresas grandes, tal factor no es mucho mejor ya que solo el 9% de sus proyectos terminaron a tiempo. Y a pesar de que estos proyectos hayan terminado a tiempo, muchos de ellos no son sino la sombra de las especificaciones funcionales para los cuales fueron pensados. Proyectos terminados por las mayores empresas americanas presentan aproximadamente un 42% de sus componentes y características originales de diseño. Compañías pequeñas por el contrario presentan mejores resultados. Un total de 78.4% de sus proyectos de software fueron exitosamente terminados con al menos 74.2% de sus características y funciones concebidas originalmente.

Estos datos pueden ser desalentadores, y de hecho, 48% de las implementaciones IT en dicho estudio presentan una mayor cantidad de fallas en comparación de los proyectos registrados cinco años atrás. La razón por la cual los proyectos no son exitosos es sin duda uno de los aspectos a considerar. El Standish Group realizó encuestas a administradores de proyectos de TI para saber sus opiniones acerca del por qué los proyectos tienen éxito. Las tres principales razones por las cuales un proyecto tendrá éxito son **la participación de los stakeholders, el apoyo a la gestión ejecutiva y a una clara y concisa definición de requerimientos**. Existen otros criterios de éxito, pero con los tres elementos mencionados, la probabilidad de éxito son mucho mejores. Sin ellos, las probabilidades de fracaso podrían incrementar dramáticamente. Según el estudio realizado por el Standish Group en el 2013, los factores de éxito o fracaso están definidos por diversos factores relacionados con la participación de los diferentes grupos de interés, toma de decisiones y generación de estrategias.

**Tabla 4** Factores de éxito de los proyectos de software (*The Standish Group, 2013*).

<b>Factores de éxito de los Proyectos</b>	<b>% de Respuestas</b>
1.- Participación de los usuarios	15.90%
2.- Apoyo a la gestión ejecutiva	13.90%
3.- Clara y concisa definición de requerimientos	13.00%
4.- Planeación adecuada	9.60%
5.- Expectativas realistas	8.20%
6.- Hitos en proyectos pequeños	7.70%
7.- Competencias del personal	7.20%
8.- Propietarios	5.30%
9.- Visión y objetivos claros	2.90%
10.- Trabajo duro, Personal enfocado	2.40%
Otros	13.90%

La tabla 4 muestra una serie de factores que pueden ayudar a definir la posibilidad de estudiar el éxito o fracaso de un proyecto partiendo del hecho de que para el estudio realizado el factor de la participación de los usuarios (15.90 %) es el factor de mayor relevancia para determinar el éxito de un proyecto de Software hasta el trabajo duro del personal enfocado (2.40%). Se puede observar que en la mayoría de ellos existe la presencia de diversos grupos de stakeholders como **los usuarios, gerentes, ingenieros funcionales, planeadores, administrativos, propietarios etc.** Los cuales forman parte de los llamados stakeholders del proyecto en cuestión. Por otro lado, se les preguntó a los participantes acerca de los problemas más comunes en la construcción, diseño y desarrollo de software los cuales son:

**Tabla 5** Problemas comunes en la construcción de software (*The Standish Group, 2013*).

<b>Problemas más comunes en la construcción de software</b>	<b>% de Respuestas</b>
1.- Deficiencia en la información ingresada en el sistema por parte de los usuarios.	12.80%
2.- Requisitos y especificaciones incompletas	12.30%
3.- Cambio de requisitos y de especificaciones	11.80%
4.- Falta de apoyo ejecutivo	7.50%
5.- Incompetencia tecnológica	7.00%
6.- Falta de recursos	6.40%
7.- Expectativas poco realistas	5.90%
8.- Objetivos poco claros	5.30%
9.- Plazos poco realistas	4.30%
10.- Nuevas tecnologías	3.70%
11.- Otros	23.00%

Los resultados de la tabla 5 engloban los problemas que más se presentan durante el ciclo de desarrollo de software los cuales pueden ser considerados como factores de análisis para el presente estudio ya que se pueden identificar, de entre los factores técnicos y de recursos materiales, a los stakeholders asociados a cada problema.

En la siguiente tabla se muestran las distintas opiniones de las personas encuestadas acerca de la razón principal por la cual los proyectos son afectados en tiempo, costo y satisfacción. Dentro de los factores se puede apreciar que los grupos de stakeholders en el proyecto son clave para llevarse a cabo como por ejemplo **ingenieros funcionales, usuarios, administradores, gerentes y planeadores.**

**Tabla 6** Factores que afectan a los proyectos de Software (Standish Group, 2013)

<b>Factores que afectan a los Proyectos</b>	<b>% de Respuestas</b>
1.- Requisitos incompletos	13.10%
2.- Carencia de participación por parte de los usuarios	12.40%
3.- Escasez de recursos	10.60%
4.- Expectativas poco realistas	9.90%
5.- Apoyo ejecutivo deficiente	9.30%
6.- Cambio de requerimientos y especificaciones	8.70%
7.- Carencia de planeación	8.10%
8.- No necesitar de nada por mucho tiempo	7.50%
9.- Poca administración de proyectos en TI	6.20%
10.- Ignorancia tecnológica	4.30%
Otros	9.90%

Otro resultado importante arrojado por la encuesta es el alto porcentaje de administradores de proyectos de TI que cree que hay más fallas en los proyectos ahora que hace cinco y hasta diez años atrás. Este hecho, a pesar de los factores tecnológicos como uso de nuevas tecnologías, metodologías y patrones de desarrollo de software, ha ido cambiando a través del tiempo:

**Tabla 7** Porcentaje de administradores de proyectos de TI que cree que hay más fallas en los proyectos ahora que hace cinco y hasta diez años (Standish Group, 2013).

	<b>que hace 5 años</b>	<b>que hace 10 años</b>
Significativamente más fracasos	27.00%	17.00%
Más fracasos	21.00%	29.00%
Sin cambios	11.00%	23.00%
Pocos fracasos	19.00%	23.00%
Significativamente pocos fracasos	22.00%	8.00%

### 2.1.3.2 CASOS DE ESTUDIO REFERENTES A PROYECTOS DE SOFTWARE

El fracaso o éxito en los proyectos de software dependen de diversos factores y para esclarecer y analizar cada una de sus causas, The Standish Group llevó a cabo observaciones a dos casos famosos del tipo de proyectos CANCELADOS y dos casos del tipo de proyecto EXITOSO. Los proyectos que fueron contemplados como parte del análisis y que tuvieron mayor relevancia son los siguientes:

**Tabla 8** Proyectos TI relevantes para el estudio del Standish Group y detección de stakeholders involucrados (Standish Group, 2013).

Proyecto	Caso	Hallazgos
<b>California DMV:</b> Rehabilitación de la licencia de conducir y proceso de solicitud de registro	Departamento de Vehículos Motorizados de California (DMV, 1987). En 1993, después de \$ 45 millones de dólares ya habían sido gastados, el proyecto fue cancelado.	<ul style="list-style-type: none"> <li>- <b>No era apoyado por la dirección ejecutiva.</b></li> <li>- <b>No contó con la participación de usuarios.</b></li> <li>- <b>Tuvo una mala planificación.</b></li> <li>- Especificaciones de diseño pobre y objetivos poco claros.</li> <li>- <b>No tenía el apoyo de información por parte de personal de gestión del Estado.</b></li> </ul>
<b>American Airlines:</b> Proyecto de reserva de servicio de Hotel	American Airlines (1994) estableció un pleito con Budget Rent-A-Car, Marriott y Hilton Corp. Hoteles después de la suma de \$165 millones por la renta de coches CONFIRM y por consiguiente el proyecto de sistema de reserva se derrumbó.	<ul style="list-style-type: none"> <li>- Habían demasiados cocineros y la sopa en mal estado consecuencia arrojada del uso del sistema de reserva.</li> <li>- <b>La dirección ejecutiva no sólo apoyó el proyecto, eran los directores activos de proyectos.</b></li> <li>- Causas importantes incluyen una declaración incompleta de los requisitos</li> <li>- <b>Falta de participación de los usuarios.</b></li> <li>- Cambio constante de los requisitos y especificaciones.</li> </ul>
<b>Hyatt Hotels:</b>	Hyatt comprobaba que en vez de hacerse check-in se hacía un check-out, la disponibilidad de servicios	<ul style="list-style-type: none"> <li>- <b>Participación de los usuarios.</b></li> <li>- <b>Apoyo a la gestión ejecutiva.</b></li> <li>- Una declaración clara de los requisitos.</li> </ul>

Proyecto	Caso	Hallazgos
Sistema de reservación de servicios vía telefónica.	mejoró. Este nuevo fue un ejemplo de bajo presupuesto, con características adicionales por lo que fue catalogado de EXITOSO. Por sólo \$ 15 millones en efectivo.	<ul style="list-style-type: none"> <li>- <b>Una adecuada planificación</b></li> <li>- Proyecto y los hitos pequeños.</li> <li>- Utilización de sistemas modernos y abiertos para software con una base de datos Infomix.</li> <li>- Utilización de monitor de transacciones.</li> <li>- Hardware basado en UNIX</li> </ul>
<b>Banco Itamarati:</b> Sistema de administración bancaria.	Crecimiento del beneficio neto anual de 51% y se trasladó de un lugar 15 <sup>a</sup> al 47 <sup>a</sup> en el sector bancario brasileño por el uso de su propio sistema de gestión bancaria.	<ul style="list-style-type: none"> <li>- <b>Visión clara de los objetivos específicos documentados.</b></li> <li>- <b>Nivel de arriba hacia abajo de la participación permitió a Banco Itamarati a mantener el rumbo.</b></li> <li>- Producción en resultados medibles.</li> <li>- <b>Incremento de labores a lo largo del período de planificación/implementación.</b></li> <li>- <b>Estrategia empresarial.</b></li> <li>- Uso de GRIP monitor de OLTP de Itautec como una herramienta básica para la integración de componentes de software.</li> <li>- Uno de los primeros bancos de implementar una arquitectura cliente-servidor que maximiza el potencial de esta arquitectura.</li> <li>- <b>El liderazgo ejecutivo.</b></li> <li>- <b>Un plan bien comunicado</b></li> <li>- <b>Un equipo diverso y experto.</b></li> </ul>

De la tabla 5 se observan resaltados de los hallazgos que se asocian con los stakeholders implicados en cada proyecto, se puede observar que existen evidencias de que el éxito o fracaso de los proyectos de software están influenciados en su mayoría por las acciones que ejercen los stakeholders. Debido a los resultados obtenidos del estudio, encuestas y análisis en diferentes proyectos de desarrollo e implementación de software se concluye que existen evidencias de que los proyectos de software presentan dificultades durante el proceso de construcción las cuales son originadas por diversos factores entre los cuales está la influencia de distintos grupos de stakeholders en los proyectos y la forma en que están estructurados, la toma de decisiones y la selección del personal que cumpla con las características necesarias para el éxito o fracaso de un proyecto de software.

## 2.1.4 FACTORES CRÍTICOS DE ÉXITO

Los factores críticos de éxito (FCE) son algunas áreas de la actividad de la organización en la cual se requiere que el director de la misma organización consiga resultados favorables con el fin de conseguir las metas que se ha trazado (*Ramírez, 2009*). Debido a que estas áreas de actividad son críticas, el director debe contar con la información apropiada para determinar que eventos se están desarrollando de la mejor manera en cada una de dichas áreas (Bullen y Rockart, 1981, p.3). Los FCE son tan importantes que requieren que el director de la organización les dedique atención constante y cuidadosa. El estado de desempeño en cada una de estas áreas críticas relacionadas a los problemas detectados, debe ser medido constantemente y la información que se genera de dichas mediciones debe ser transmitida a los otros miembros encargados de su administración (Bullen y Rockart, 1981, p.13).

### 2.1.4.1 FACTORES CRÍTICOS DE ÉXITO EN LA IMPLANTACIÓN DE SISTEMAS DE INFORMACIÓN O SOFTWARE EN ORGANIZACIONES MEXICANAS.

Uno de los procesos más importantes del desarrollo de Software es la etapa de implantación del sistema construido dentro de las organizaciones y si bien no cubre la totalidad del análisis de los factores críticos de éxito del desarrollo de Sistemas, si da una pauta para analizar una de las fases intermedias entre el desarrollador y el cliente: La etapa de IMPLANTACIÓN.

García-Sánchez y Pérez-Bernal (2007) estudiaron los factores críticos de éxito para la implantación de SI/TIC en organizaciones mexicanas lo cual es parte del desarrollo de proyectos de Software; el caso particular estudiado por estos investigadores fue el de los sistemas para la planeación de los recursos de la empresa o sistemas ERP. Ellos encontraron que se presentan tres grupos de factores que son los siguientes: **humanos, tecnológicos y organizacionales.**

Cada uno de estos grupos se compone como sigue (el orden en que se listan los factores críticos de éxito corresponde al orden de importancia que resultó de la investigación de García-Sánchez y Pérez-Bernal (2007)):

- |  |  |
|--|--|
| 1. Factores humanos                      | 7. Gerencia de proyecto                      |
| 2. Composición del equipo de trabajo     | 8. Selección del sistema ERP                 |
| 3. Comunicación                          | 9. Entrenamiento y soporte a usuarios        |
| 4. Líder de proyecto                     | 10. Pruebas y solución de problemas          |
| 5. Participación de los usuarios finales | 11. Facilitación de cambios en la estructura |
| 6. Factores tecnológicos                 |  |

- |   |  |
|---|--|
| <p>organizacional,<br/>en los “sistemas<br/>legado” e<br/>infraestructura</p> <p>12. Factores<br/>organizacionales</p> <p>13. Apoyo de la alta<br/>gerencia</p> | <p>14. Reingeniería de<br/>procesos de<br/>negocio</p> <p>15. Contar con<br/>consultores<br/>externos</p> <p>16. Plan de gestión<br/>el cambio</p> |
|---|--|

Enunciado de la visión y plan de negocio adecuado. Esta estructura permite que el director y toda la organización encaminen sus esfuerzos y recursos hacia la toma de decisiones que permitan alcanzar el resultado mejor integrado de acuerdo a la disponibilidad de recursos en la organización. Por ejemplo, si se detecta que la organización es débil con respecto a un grupo de factores (por decir factores humanos) entonces se puede enfatizar las estrategias, esfuerzos y acciones para el reforzamiento de dicha sección en búsqueda de un balance adecuado con los otros dos grupos.

## **2.2 LOS STAKEHOLDERS**

### **2.2.1 TEORÍA Y DEFINICIÓN DE STAKEHOLDERS**

Para entender la problemática entorno a la incursión de los stakeholders en el desarrollo de proyectos de software, es necesario entender la definición, las bases científicas y los principios de la teoría de los stakeholders y la relación que guarda con los procesos de planeación, dirección y generación de riqueza de una empresa.

La teoría de los stakeholders es reconocida como la teoría de los grupos interesados que intenta analizar la gestión empresarial e incluye los diferentes intereses generados en torno a una organización, con desarrollos científicos de carácter normativo instrumental y descriptivo (Jones y Wicks, 1999).

Freeman (1984) define a los stakeholders como: “cualquier persona o grupo de personas que pueda afectar el desempeño de la empresa o que sea afectado por el logro de los objetivos de la organización” (p.46, citada por Mitchell et al; 1997, p. 856). Según la definición de Freeman son stakeholders los empleados, proveedores, clientes, los competidores, el Estado, los directivos, los accionistas y los ciudadanos que están alrededor de la organización de forma directa o indirecta. Siguiendo a Clarkson (1995), a partir de sus investigaciones sobre la evolución del concepto de Responsabilidad Social Corporativa desde los años setenta, y su análisis empírico sobre la ejecución de programas y políticas en las organizaciones para el manejo de la responsabilidad social, este re-conceptualiza en niveles a los stakeholders entre primarios o secundarios, según el grado de relación con la organización. Para Clarkson, los stakeholders son:

*Personas o grupos de personas que tienen, o reclaman, propiedad, derechos, o intereses en una organización y sus actividades, pasadas, presentes, o futuras. Estos derechos reclamados o intereses son el resultado de transacciones con, o acciones tomadas por, la organización y pueden ser legales o morales, individuales o colectivas.*

*Stakeholders con similares intereses, reclamos o derechos, pueden clasificarse como pertenecientes al mismo grupo: empleados, accionistas, clientes y así sucesivamente (p. 106)*

Esta última definición es también de carácter descriptivo, con más evolución teórica que la propuesta por Freeman (1984). Además, incluye las posibles acciones que pueden tomar los stakeholders de una organización, siendo claramente una relación de carácter unidimensional explícitamente por la superioridad de la organización. Para Clarkson (1995) existen diferentes tipos de stakeholders:

### **1) Los grupos primarios de stakeholders**

Son aquéllos sin los cuales la organización no puede sobrevivir (**en el caso de los proyectos de desarrollo de software: Stakeholders internos: Clientes internos, patrocinador del proyecto, equipo central del proyecto, proveedores de recursos funcionales, supervisor del administrador del proyecto, grupos de apoyo. Así como los stakeholders externos: Clientes externos, grupos de usuarios, proveedores, contratistas y consultores**), donde se encuentran los accionistas e inversionistas, empleados, clientes y proveedores, los gobiernos, las comunidades y los mercados, cuyas leyes y reglamentos deben ser obedecidos y para los cuales los impuestos y otras obligaciones deben ser cumplidos y en donde existe un alto grado de interdependencia entre la organización y sus grupos primarios de interesados, y por tanto la insatisfacción de los stakeholders primarios puede generar graves problemas para que la organización pueda continuar.

Es importante el efecto que tiene la sociedad sobre la organización que es necesario incluirla dentro el grupo primario de stakeholders. Sus derechos, objetivos, expectativas y responsabilidades afectan la supervivencia de la compañía, y por esto el directivo deberá, para lograr el éxito de la organización, crear la suficiente riqueza, valor y satisfacción para aquellos que pertenezcan a cada grupo primario de stakeholders. La no atención a un grupo primario de interesados o su no reconocimiento puede generar el fracaso de la organización.

### **2) Los grupos secundarios de stakeholders**

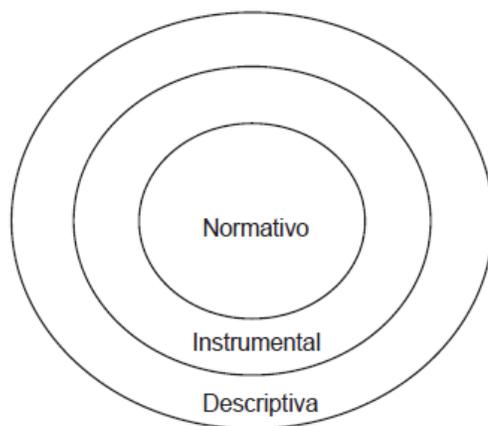
Aunque no tienen el mismo impacto que los grupos primarios, pueden oponerse a las políticas o programas que una organización adopta para cumplir con sus responsabilidades, o para satisfacer las necesidades y expectativas de sus principales grupos de interesados, lo que genera situaciones que la organización debe resolver, pero que le permiten funcionar y sobrevivir mientras lo hace (Clarkson, 1995).

Esta diferenciación entre grupos primarios y secundarios de stakeholders es vital para la definición de estrategias por parte de los directivos, al gestionar o administrar las relaciones con cada grupo de stakeholders, según sea su influencia en la organización. A partir de esta diferenciación propuesta por Clarkson (1995), se han realizado varias aplicaciones para intentar identificar los grupos de stakeholders de una organización y su grado de influencia, desarrolladas entre otros autores por Rowley (1997), Waddock y Graves (1997), Ogden y

Watson (1999), Scott y Lane (2000) o Schneider (2002).

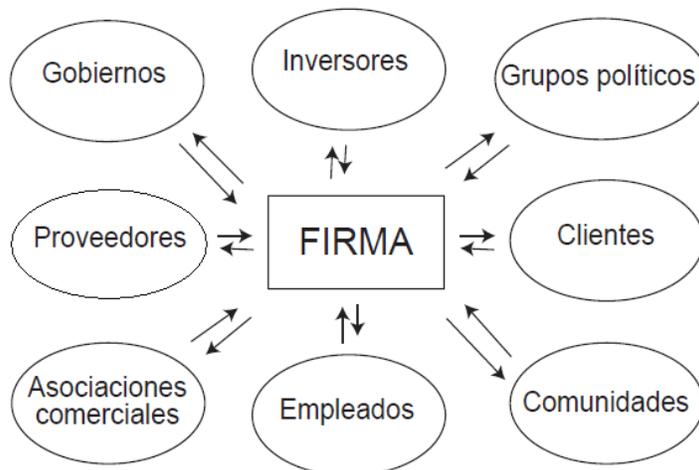
Ahora bien, dentro de las prácticas instrumentales de la Teoría de los Stakeholders, es posible encontrar diferentes modelos que proponen *el cómo debería la organización identificar y/o administrar las relaciones con sus stakeholders*. Un ejemplo relevante es el modelo de Donaldson y Preston (1995), que centran su análisis en el directivo de la organización y proponen cómo es posible identificar y actuar sobre los distintos stakeholders y sus demandas. Este modelo incluye tres (3) tesis importantes: lo descriptivo, lo normativo y lo instrumental (Figura 4). El aspecto descriptivo explica la organización y la relación con sus stakeholders; el instrumental determina el funcionamiento de las relaciones entre la organización y sus stakeholders; y el normativo define conceptualmente a los stakeholders y su representación.

**Figura 4** Tres aspectos de la Teoría de los Stakeholders (Donaldson, Preston, 1995, p. 74).



En la figura 5, se observan las entradas y salidas del modelo de los stakeholders, definidos como todas las personas o grupos con legítimos intereses que participan en una organización para obtener beneficios y donde no prima la prioridad de una serie de intereses y beneficios en detrimento de otro (Donaldson y Preston, 1995).

**Figura 5** Contrastando el modelo de la organización y la teoría de los stakeholders (Donaldson, Preston, 1995, p. 69).



En la figura 5 se identifican las relaciones entre la organización y sus stakeholders con flechas en ambos sentidos o direcciones, donde todos los stakeholders están representados de igual forma y tamaño, además están situadas a la misma distancia de la organización que está en el centro del gráfico. Con este esquema, Donaldson y Preston (1995) intentan proponer teóricamente una cierta igualdad como stakeholders de una misma organización.

Pero también proponen que debe ser el directivo quien defina la prioridad asignada a cada grupo de stakeholder, teniendo en cuenta el grado de dependencia de la organización con cada grupo de stakeholder. Donaldson y Preston (1995) plantean la validez de la Teoría de los Stakeholders en otros tipos de organizaciones como los gobiernos y los paradigmas sociales y resaltan la importancia de analizarla no solo de forma teórica, sino práctica, e incluyen elementos tan importantes como los derechos de propiedad. En una línea de acción muy similar aparece el modelo de Mitchell *et al.* (1997), que a diferencia de Donaldson y Preston (1995), proponen relaciones más dinámicas entre la organización y sus stakeholders. Incluyendo el poder y la urgencia como componentes de la relación y desarrollando aún más el concepto de legitimidad propuesto muy superficialmente por estos autores.

Para Mitchell *et al.*, teniendo en cuenta la legitimidad, el poder y la urgencia, los directivos tienen la facultad de no actuar de igual manera con todos los stakeholders. La prioridad que asignen los directivos a las demandas de sus stakeholders estará definida para todos o algunos de los siguientes atributos: el poder de la influencia del stakeholder; la legitimidad de la relación del stakeholder y la urgencia de la demanda del stakeholder (el tiempo). Partiendo de estos tres atributos y de sus posibles combinaciones, Mitchell *et al.* Proponen las clases y los tipos de stakeholders que pueden existir (Figura 5).

Analizando las posibles combinaciones de uno, de dos o de los tres atributos de los stakeholders, los autores plantearon tres clases con algunos tipos específicos de stakeholders, así:

**1.- Latentes**, incluyen:

- a) Durmientes: con poder sin legitimidad ni urgencia
- b) Discrecionales: con legitimidad, pero sin urgencia y sin poder
- c) Demandantes: con la urgencia, sin poder, ni legitimidad.

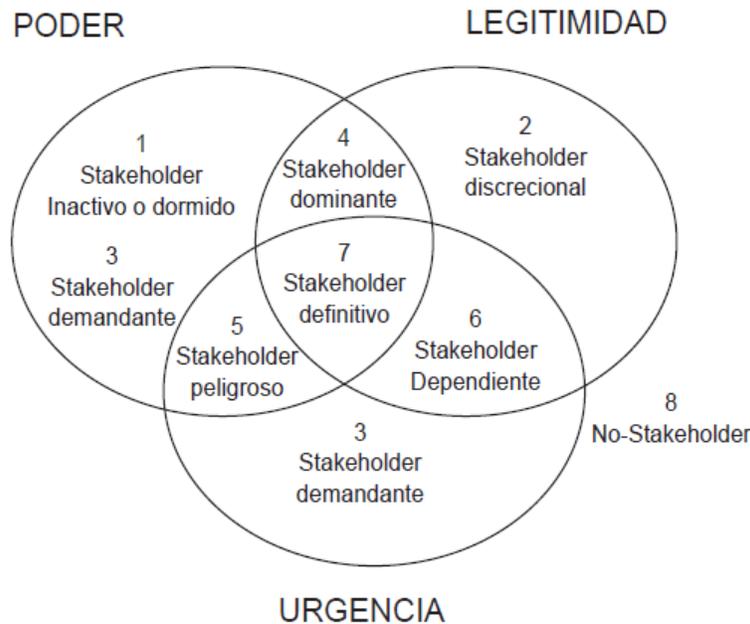
**2.- Expectantes**, incluyen:

- a) Dominantes: con poder y legitimidad, sin urgencia
- b) Peligrosos: urgencia y poder sin legitimidad
- c) Dependientes: con urgencia y legitimidad, sin poder.

**3.- Definitivos**, que tienen los tres atributos, es visto cuando al adquirir el atributo faltante se presenta ante los directivos.

Este modelo permite relacionar el objetivo de la organización con las demandas de los tipos de stakeholders que adquieren mayor importancia para la empresa.

**Figura 6** Tipología de los stakeholders: uno, dos y tres atributos (*Mirchell, Agle, Wood, 1997, p. 872*).



Aunque las corrientes descriptivas y normativas de la Teoría de los Stakeholders son en ocasiones vistas como contrarias, conviven en el desarrollo científico de la teoría y llegan a la definición de debates teóricos continuos, secuenciales y concurrentes, lo que ha permitido una gran producción científica tanto teórica como empírica en el área.

Es por ello que la gestión de los stakeholders puede generar beneficios para la organización y llegar a convertirse en una ventaja competitiva. La información proveniente de los stakeholders puede permitir el desarrollo de nuevos productos y estrategias para conocer y entender como potencializar la implantación de herramientas, metodologías, tecnologías y prácticas en el desarrollo de proyectos de software, que dependerán en parte del reconocimiento de las diferencias existentes entre los diversos grupos de stakeholders (Harrison y Freeman, 1999).

## 2.2.2 DESARROLLO DE SOFTWARE Y LOS STAKEHOLDERS

Stakeholder es el término en inglés que define a los interesados en un proyecto de desarrollo de software para este estudio y de cualquier tipo de proyecto.

La definición de Stakeholder ha ido cambiando a lo largo del tiempo. Lo que antes se conocía como la jerarquización de una organización ha ido cambiando para transformarse en un término que tiene un mayor impacto en el desarrollo de las organizaciones, y por consiguiente de los proyectos de diferentes índoles que se pretendan realizar. Un interesado son todas aquellas **personas** que participando directa o indirectamente en la realización de los trabajos se ven afectadas por el resultado de los mismos.

Con esta definición, las personas y departamentos involucrados pueden ser muchos. Se puede llegar a trabajar con todos, sin embargo lo más interesante, aunque eso lo determinarán las características del proyecto, es seleccionar de entre ese dominio, aquellos en los que realmente el impacto del desarrollo sea sensible, ya sea porque participan en el mismo, porque gestionan el proceso o procesos que se informatiza, porque exista una trascendencia económica que interesa gestionar y controlar o porque son destinatarios de la solución hasta los que pueden verse condicionados por la misma.

La identificación de los **stakeholders** resulta por tanto esencial en un proyecto de desarrollo de software, ya que es necesario que de alguna u otra forma tengan participación en el mismo definiendo el rol que deben tener en la ejecución de los trabajos y determinando las expectativas que tienen respecto a los resultados del proyecto, las cuales condicionarán los trabajos a realizar y servirán como punto de comparación a la hora de evaluar la calidad de los trabajos.

Debido a que es imperativo conocer quiénes son los stakeholders que son impactados positiva o negativamente por los proyectos de desarrollo de software y como resultado de los estudios realizados en distintos casos de estudio donde su impacto repercute en el éxito y/o fracaso del proyecto es necesario listar a los stakeholders relevantes y de los cuales es necesario conocer diferentes niveles de datos para poder manejar apropiadamente las relaciones con ellos y establecer un plan de comunicación.

Dentro de la clasificación de proyectos de desarrollo en software encontramos, como se había mencionado anteriormente, los proyectos de desarrollo libre y privado. Los grupos de interés que participan en el proceso de desarrollo de software privado y que es necesario analizar en el presente trabajo, se pueden definir como aquellos grupos que teniendo como interés específico el desarrollo de proyectos de software para potenciar el accionar de una organización, cumplen y actúan con distintos códigos de comunicación y dirección que los identifican y que hacen que un proyecto de desarrollo en software pueda llegar a culminar de manera exitosa pero por el contrario también impidiendo que cumpla con sus condiciones por las cuales fue solicitado por el cliente, es decir, que no cumpla con sus requerimientos esenciales.

Existen ventajas en el desarrollo de software privado que pueden repercutir en el momento de distinguir los diferentes grupos de interés en el desarrollo de software las cuales son:

1. Propiedad y decisión de uso de software por parte de la empresa
2. Soporte para todo tipo de hardware
3. Mejor acabado de la mayoría de las aplicaciones
4. Aplicaciones número uno son de carácter propietario o privadas
5. El ocio para ordenadores personales está destinado al mercado propietario
6. Menor necesidad de técnicos especializados

Según la cuarta edición del PMBOK, lo que se necesita saber de los stakeholders responde a las siguientes preguntas: **¿Quién es el stakeholder por su nombre?, ¿Cuál es la naturaleza de su interés en el proyecto? ¿Personal o profesional?, ¿Qué esperan los stakeholders del administrador del proyecto?, ¿Qué esperan los stakeholders? Y ¿Cuáles son las prioridades de vigilancia de los interesados? (PMBOK, 2013).**

## 2.2.3 EJEMPLO DE PARTICIPACIÓN DE LOS STAKEHOLDERS: EL GOBIERNO CORPORATIVO.

La AECA (2007) señala que la participación de los stakeholders en el gobierno corporativo “no implica eliminar el papel e intervención de los accionistas, más bien se trata de construir redes de intereses y aplicar estrategias de colaboración, reforzando la supervisión mutua de cada stakeholder sobre las actuaciones de los demás”, para lo cual proponen una cartera de mecanismos de gobierno para la empresa pluralista que identifica los distintos niveles de participación de los grupos de interés:

**Tabla 9** Niveles de participación de Stakeholders en el gobierno corporativo (AECA, 2007)

Nivel	Meta	Enfoques de la relación
Permanecer pasivo	No hay meta ni relación.	La compañía no se relaciona con sus <b>stakeholders</b> . Las inquietudes de los stakeholders se plantean a través de protestas, cartas, Internet, etc.
Hacer un seguimiento	Hacer un seguimiento de las opiniones de los stakeholders	Seguimiento en los medios y en Internet. Informes de segunda mano de otros stakeholders (entrevistas puntuales).
Informar	Informar o educar a los stakeholders	Boletines, cartas, folletos, informes y sitios en Internet. Discursos, conferencias y presentaciones públicas. Acceso a las instalaciones y recorridos. Guías de presentación y exposiciones. Comunicados y conferencias de prensa, publicidad en los medios.
Realizar transacciones	Trabajar juntos en una relación contractual en la que un socio dirige los objetivos y provee los fondos.	Alianzas entre el sector público y el privado e iniciativas de financiación privada, subsidios, marketing de causas, lobby.
Consultar	Obtener información y opiniones de stakeholders para fundamentar las decisiones internas.	Encuestas, grupos de enfoque, evaluación de ambientes de trabajo, reuniones personales y públicas, laborales, foros de asesoramientos, foros on-line, sondeos de opinión.
Participar	Trabajar directamente con los stakeholders para asegurar la comprensión y consideración de sus inquietudes en los procesos de toma de decisiones.	Foros múltiples de stakeholders, paneles de asesoramiento, procesos de creación de consenso, procesos participativos de toma de decisiones.
Colaborar	Asociarse o convocar a una red de stakeholders para desarrollar consensos y	Proyectos conjuntos, iniciativas voluntarias de dos o más stakeholders.

	planes de acción conjunta.	
Delegar	Delegar las decisiones sobre un tema determinado a los stakeholders.	Gobierno democrático de los stakeholders (por ejemplo miembros, accionistas, integrantes de comités especiales, etc.)

## 2.3 METODOLOGÍAS DE PROYECTOS CON UN ENFOQUE EN STAKEHOLDERS

### 2.3.1 MODELO INTEGRADO DE MADUREZ Y CAPACIDADES (CMMI)

La integración de modelos de madurez de capacidades (CMMI por sus siglas en inglés) es un modelo para la mejora y evaluación de procesos para el desarrollo, mantenimiento y operación de sistemas de software. Dentro de la amplia gama de procesos que lleva a cabo CMMI se encuentra la importancia al entorno **Organizacional**. CMMI está formado por un conjunto de áreas de proceso cuya adecuada aplicación definen un determinado nivel de madurez a una organización.

El hecho de que esté orientado a organizaciones o departamentos dentro de las mismas dedicados a esta actividad no quiere decir que se centre exclusivamente en mejorar el proceso de desarrollo de software, sino también en mejorar el propio funcionamiento de esas organizaciones o esos departamentos mediante la aplicación de los procesos definidos en CMMI y la búsqueda de la mejora continua. No se trata solo, por tanto, de normalizar, homogeneizar, ordenar, racionalizar y armonizar el proceso de desarrollo, sino que una vez se hayan estabilizado estos procesos y haya comenzado el cambio de cultura, el siguiente paso es la mejora continua de los procesos y con ello de la organización.

En particular la siguiente tabla describe cada una de las etapas:

**Tabla 10** Niveles y caracterización del modelo CMMI (Palacios, Porcell, 2012)

NIVEL	DESCRIPCIÓN
1. Inicial	<p>El proceso de software es impredecible, sin control y reactivo. El éxito de los proyectos depende del talento de los individuos.</p> <p>Una organización dedicada al desarrollo de software que se encuentre en este nivel se entiende que no tiene un control real sobre los proyectos en los que está trabajando: no se conoce realmente el grado de ejecución de los mismos, no hay control de plazos, no hay control de los costes económicos, no existe una motivación para la toma de determinadas decisiones, etc.</p> <p>Este nivel se caracteriza, por tanto, por el comportamiento impredecible de los proyectos por la ausencia de procesos definidos o por el no cumplimiento de los mismos (que viene a ser lo mismo) de manera que</p>

NIVEL	DESCRIPCIÓN
	<p>cada equipo de proyecto o incluso cada desarrollador sigue su propia estrategia o metodología, como una decisión personal o de grupo, pero no siguiendo unas directrices a nivel organizativo.</p>
<p>2.- Repetible</p>	<p>Existen procesos básicos de gestión los proyectos (costo, calendario, funcionalidad). Los procesos existentes hacen que se puedan repetir éxitos en proyectos de similares características.</p> <p>CMMI pretende dar un primer paso respecto a la situación de anarquía inicial, en la que no existen procesos o si existen no se aplican o controlan. En este nivel se entiende que para aplicar estrategias de nivel organizativo es necesario crear primero una cultura orientada al proceso y para ello plantea como estrategia que por los menos los mismos se implanten a nivel de proyecto.</p> <p>Bastará, por tanto, con que los procesos se definan a nivel de proyecto, pudiendo ser distintos entre unos y otros. Lo importante es que existan mecanismos de gestión, control y supervisión, que sean distintos entre los diferentes proyectos no es esencial, ya que se habrá conseguido un avance significativo respecto al nivel anterior al existir mecanismos que permiten obtener información del estado del proyecto, de su avance, de sus posibles desviaciones, de los posibles problemas que pueden existir.</p> <p>La adquisición del nivel 2 de madurez no es algo que esté fuera del alcance de muchas empresas de desarrollo de software, eso sí, requiere un cambio cultural en la organización de manera que se pasa de una situación donde prima la falta de control a otra en la que se intenta controlar, pero no como fin sino como medio para poder buscar soluciones a tiempo cuando hay problemas y como medio para mejorar.</p> <p>Se requiere una cierta disciplina y un cambio de enfoque, un pequeño precio a pagar teniendo en cuenta los beneficios que puede aportar una gestión de proyectos de desarrollo de software guiada diferentes procesos.</p>
<p>3.- Definido</p>	<p>Existe un proceso de software documentado y estandarizado dentro de la organización. Todos los proyectos utilizan una versión a medida del proceso.</p> <p>El nivel 3 de CMMI pretende dar un paso más, mediante la definición de procesos de ámbito organizativo en los cuales además existirá un mayor nivel de detalle, dejando un menor margen a la interpretación de la definición de los mismos.</p> <p>Exige un mayor control en el proceso de construcción del producto, ir un paso más allá en las técnicas de gestión de proyectos y dar una mayor formalidad a la mejora continua de los procesos y su</p>

NIVEL	DESCRIPCIÓN
	ejecución.
4.Gestionado	<p>La organización recolecta métricas del proceso software y de los productos desarrollados. Tanto el proceso como los productos se entienden y controlan cuantitativamente.</p> <p>Se utilizan estrategias y herramientas que permitan obtener el desempeño/rendimiento real (presente) de las áreas de proceso implantadas y los productos y servicios que ofrece la organización, así como su comportamiento futuro.</p>
5.Optimizado	Existe una mejora continua del proceso software, basada en la retroalimentación cuantitativa del proceso y en la puesta en práctica de ideas y tecnologías innovadoras.

La metodología CMMI considera, dentro de su accionar, un enfoque específico en los **stakeholders**. Dentro de su proceso de **desarrollo de software** (CMMI-DEV por sus siglas en inglés) existe una división en dos modelos, lo que sería el CMMI-DEV tal cual o el CMMI-DEV + IPPD (*Integrated Product and Process Development* por sus siglas en inglés) (Peralta, 2011).

La diferencia entre ambos modelos se encuentra en la aplicación o no de las siguientes áreas de proceso de nivel 3:

1. Equipos integrados.
2. Ambiente organizacional para la integración.

EL CMMI considera un área específica a equipos integrados la cual hace referencia al equipo de proyecto, el proceso para conformarlo, para seleccionar los roles adecuados, para orquestar la posible colaboración entre diferentes equipos. De esto trata esta área de proceso del nivel 3 de CMMI y por la cual se refleja la participación de los stakeholders en dicho proceso.

Una vez conformado el equipo, lo siguiente es definir las actividades que permiten fomentar una visión compartida de los objetivos a conseguir y del método para conseguirlo (comunicación, trabajo en equipo, etc.), dar a conocer a cada componente su rol y responsabilidades, establecer los procesos necesarios para coordinar el funcionamiento del equipo y por último definir los procedimientos para poder establecer relaciones de colaboración con otros equipos de trabajo (para ello además de existir determinados procesos horizontales, requiere tener un conocimiento de las tareas y proyectos que se están realizando en la organización o en los que colaboran algunos de los interesados). Una vez conformado el equipo lo siguiente es definir las actividades:

- 1.- Fomento de una visión compartida de los objetivos a conseguir y del método para conseguirlo (comunicación, trabajo en equipo, etc.)
- 2.- Dar a conocer a cada componente su rol y responsabilidades
- 3.- Establecer los procesos necesarios para coordinar el funcionamiento del equipo

4.- Definir los procedimientos para poder establecer relaciones de colaboración con otros equipos de trabajo.

La metodología CMMI considera dentro de sus niveles de acción a los stakeholders. Se puede mencionar al nivel 2 el cual contempla la planificación del proyecto. La planificación del proyecto supone la base para la realización del mismo, por lo que es habitual que la mayor parte de las tareas que lo componen se realicen de una manera más o menos formal: Determinar un alcance inicial (aunque sea a alto nivel), hacer un análisis de riesgos, hacer una estimación del costo/esfuerzo, de los recursos en materia de personal (¿quiénes van a participar?, ¿cuándo?, ¿con qué dedicación?, ¿necesitan algún tipo de formación antes de enfrentarse al proyecto?) y materiales (software y hardware), definir el ciclo de vida/metodología a utilizar, proponer un cronograma, definir los hitos, las personas que los aprueban, etc.

Buena parte de la suerte del proyecto se juega al principio, una mala negociación, una mala estimación, en general, una mala planificación puede condicionar negativamente el proyecto, hasta tal punto de que si no hay posibilidad de enmendar la situación a tiempo puede afectar incluso a su viabilidad. No se trata solo, por tanto, de normalizar, homogeneizar, ordenar, racionalizar y armonizar el proceso de desarrollo, sino que una vez se hayan estabilizado estos procesos y haya comenzado el cambio de cultura, el siguiente paso es la mejora continua de los procesos y con ello de la organización.

Conforme vayamos mirando de más arriba a los proyectos nos daremos cuenta de la existencia de los **interesados** en diferentes áreas dentro de la organización y del cliente que participan en diferentes proyectos en los cuales los equipos de proyecto pueden ser distintos o tener pocos componentes en común. Esta circunstancia recomienda igualmente una gestión integrada de **las relaciones con los interesados**, ¿cuántas veces se ha hecho llegar un mensaje contradictorio a un cliente desde personas distintas dentro de un equipo de proyecto o entre integrantes de proyectos distintos? Hay que tener cuidado con estos traspiés ya que merman la credibilidad de los interesados afectados, todavía más grave si el interesado es el cliente (no solo se consigue la confianza de estos obteniendo buenos resultados en el producto, también se consigue a través de las relaciones con los mismos).

También es importante no solo tener controlado lo que se dice a los i, sino también la información que se recibe de ellos. En el nivel 3 de CMMI las áreas de proceso tienen un alcance organizacional por lo que tiene sentido que se trate también de armonizar y homogeneizar determinados tipos de decisiones alejando la misma de interpretaciones o varas de medir personales y traspasándolas a un entorno repetible y más objetivo. Todo esto requiere la definición de esa cultura de empresa y los mecanismos para que la misma sea conocida por los empleados y más allá de eso, para que esté integrada dentro del día a día de los trabajadores. Esa cultura de empresa, deberá tener como aspecto de gran interés el fomento de trabajo en equipo, para ello debe proporcionar y promocionar un ambiente integrado de trabajo (pasar de las buenas palabras a hechos que fomenten y permitan la colaboración).

### **2.3.2 FUNDAMENTOS DE LA GESTIÓN DE PROYECTOS (PMBOK) Y LA IDENTIFICACIÓN DE STAKEHOLDERS**

Aparte de las herramientas de madurez organizacional para empresas desarrolladoras de software como CMMI, existen otras que incluyen en sus procesos el análisis y la detección de

stakeholders, una de ellas es el PMBOK.

Bajo el nombre “*Identificar Stakeholders*”, el Instituto de Administración de Proyectos (PMI por sus siglas en inglés) decidió exaltar esta actividad como un nuevo proceso en la Cuarta Edición del Marco de Conocimiento de la Administración de Proyectos (PMBOK por sus siglas en inglés) (Crow, 2013), ya que es uno de los más importantes para establecer las bases tempranas dirigidas a la posterior planificación, ejecución, así como monitoreo y control de la información y comunicación del proyecto, para alcanzar el éxito en éste.

Este proceso debe hacerse en la fase de inicio del proyecto para que las salidas claves del Registro de Stakeholders y la Estrategia de Administración de los Stakeholders sean usadas asociativamente en el proceso de Gestión de la Comunicación conocido como Manejo de las Expectativas de los Stakeholders.

Es imperativo identificar a todas las personas y organizaciones que serán impactadas por el proyecto y posteriormente documentar la información relevante respecto a sus intereses, participación e impactos sobre el éxito del proyecto. En este ámbito, el PMBOK sugiere usar dos salidas tempranas para la identificación de stakeholders en el proyecto:

1. La primera disponible es la salida proveniente del desarrollo del Acta Constitutiva del Proyecto, la cual puede contener una lista de los clientes, patrocinadores, ejecutivos, equipo del proyecto o entidades que son externas al desarrollo de la organización participante en el proyecto. Aquí es recomendable sostener un encuentro por separado con los personajes identificados en el acta y preguntarles si conocen de otros que puedan figurar como stakeholders.
2. Si el proyecto es el resultado de una actividad de aprovisionamiento o está basado en un contrato establecido, es importante usar los documentos de adquisición para identificar todas las partes dentro del contrato que podrían ser stakeholders claves para el proyecto. Los proveedores que participan en el contrato también podrían ser considerados para la identificación de los interesados en el proyecto. Usar un contrato hace el proceso un poco más fácil, ya que las diferentes interfaces del contrato deberían ser enlistadas.

El acta constitutiva del proyecto y las indicaciones de los documentos de aprovisionamiento de los interesados solamente pueden darse al grupo de **stakeholders**. Por lo tanto, procederemos a definir con mayor precisión otras posibilidades de stakeholders y lo que se necesita para saber sobre éstos. El PMBOK sugiere usar una herramienta de Análisis de Stakeholders y una técnica para recoger información que determine quién de los interesados debería ser considerado a lo largo del proyecto.

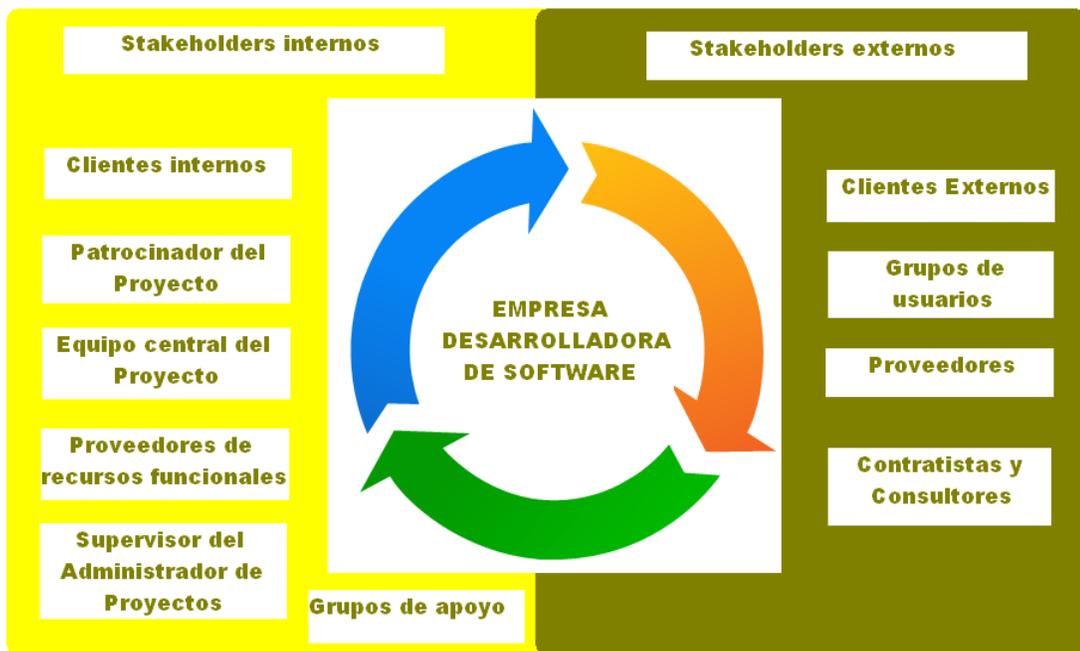
La importancia de analizar a los interesados ayuda a definir el lugar para cada stakeholder, así como sus funciones. El análisis de stakeholders es una herramienta de modelo de clasificación que ayuda identificar y determinar el impacto o apoyo que cada stakeholder podría generar y entonces es utilizado para clasificar a éstos y así precisar la información para la Estrategia de Administración de los Stakeholders, la cual es una de las salidas de este proceso.

**¿Qué necesitamos saber sobre los stakeholders?**, se requiere conocer diferentes niveles de datos acerca de los interesados identificados para poder manejar apropiadamente las relaciones con ellos y establecer un Plan de Comunicaciones. Estos son los factores claves que se deben recoger de los stakeholders:

1. ¿Quién es el stakeholder por su nombre?
2. ¿Cuál es la naturaleza de su interés en el proyecto? ¿Personal o profesional?
3. ¿Qué esperan los stakeholders del administrador del proyecto?;
4. ¿Qué esperar de los stakeholders?
5. ¿Cuáles son las prioridades de vigilancia de los stakeholders?

Por otro lado se analiza la clasificación del desempeño de los stakeholders en grupos internos o externos y los papeles que juegan dentro de un proyecto de acuerdo con la metodología del PMBOK.

**Figura 7** Esquema de ubicación de stakeholders internos y externos en las empresas desarrolladoras de software (Crow, 2013).



**Stakeholders internos:** La mayoría de los stakeholders claves son personas que laboran dentro de la organización sobre la cual se va a desarrollar el proyecto. En este grupo encontramos:

3. **Clientes internos:** Normalmente son personas para quienes el administrador de proyecto está haciendo el trabajo y tienen una necesidad particular en que el proyecto pueda ser dirigido a feliz término. A menudo, los clientes internos pagan por el proyecto y por lo tanto reciben impactos en sus negocios a partir de los entregables del proyecto.
4. **Patrocinador del proyecto:** Normalmente, no es una posición específica dentro de la organización, más bien es un rol jugado en un proyecto. El papel del patrocinador es típicamente un representante de alta jerarquía quien tiene un gran interés en los resultados del proyecto. Este rol puede ser invaluable para el administrador de proyecto (project manager) cuando enfrenta problemas o asuntos que van más allá de su ámbito de influencia. El patrocinador puede facilitar decisiones y contribuir con la asignación de recursos. Un patrocinador puede ser un miembro de la dirección quien tiene un interés

en el éxito o fracaso en el proyecto. Hay que describirles sus roles, así como las expectativas del administrador del proyecto. Después de todo, la dirección de la empresa debería estar muy dispuesta a contribuir con el éxito.

5. **Equipo central del proyecto:** Generalmente están ligados cercanamente para hacer el trabajo. En la mayoría de los casos el equipo principal es un grupo relativamente pequeño compuesto a partir de diferentes departamentos de trabajos necesarios para completar el proyecto.
6. **Proveedores de recursos funcionales:** Asegurar los recursos puede depender del tipo de estructura de la organización que requiere el proyecto. En los proyectos se debe solicitar recursos de otros departamentos, pidiéndoselos al gerente funcional adecuado.
7. **Supervisor del administrador de proyectos:** Simplemente es el jefe del administrador de proyecto y tiene un gran interés en el éxito del proyecto. El administrador del proyecto debe mantenerlo informado en todo momento y protegerlo para que no reciba sorpresas desagradables.
8. **Diferentes grupos de apoyo:** Esos grupos existen dentro de la organización y son los relativos a la parte legal, contabilidad, procesamiento de datos y recursos humanos de la empresa. El papel de éstos hacia el proyecto es más de apoyo que trabajo activo, dependiendo de las necesidades específicas del proyecto. Aquí hay que considerar si uno de esos grupos debería tener un representante en las reuniones del equipo principal.

**Stakeholders externos:** Los de este grupo tienen interés intrínseco en el proyecto más, aunque no formen parte de la organización. En este grupo encontramos:

- **Clientes externos:** Se caracterizan típicamente por los contratos.
- **Grupos de usuarios:** Se debe considerar a los grupos de usuarios si el proyecto desarrolla o fabrica un producto que será comercializado y vendido a los consumidores. El administrador del proyecto puede consultar al stakeholder externo acerca de gustos, desagradados, preferencias y elecciones que tal vez su estrategia de marketing ha asilado para el futuro o para un producto producido similarmente.
- **Proveedores:** El proyecto puede requerir materiales que deben ser conseguidos a partir de compañías externas. El administrador del proyecto debe utilizar la lista de proveedores principales en el caso de que la empresa tenga una.
- **Contratistas y consultores:** Al igual que ocurre con los materiales, los cuales son adquiridos a proveedores, el líder del proyecto también puede utilizar tanto a contratistas como consultores para realizar ciertas labores o requerir de algunos servicios. En este caso es recomendable usar un criterio basado en el desempeño y un registro verificable al momento de seleccionar a estos stakeholders.

Del esquema de ubicación de stakeholders propuesto por el PMBOK (Figura 7) se debe mencionar que pueden existir diversos puntos de vista por parte de los administradores del proyecto y en general para ciertos stakeholders en cuanto al manejo y comprensión al momento de implementar la metodología en la administración de algún proyecto. Se debe mencionar que la propuesta exhibe a dos tipos diferentes de clientes: externo e interno, lo que pudiera ser confuso para fines prácticos y metodológicos por dichos grupos de interés. Se debe considerar que el tipo de administración de proyectos de software diferencia dichos tipos de stakeholders para fines prácticos, ya que por ejemplo es necesario precisar que los stakeholders externos (contratos) deben su nombre por razones a que existe la posibilidad de

que en el desarrollo de dichos proyectos se manejan diferentes tipos de contratos externos por parte de aquellos stakeholders que no son denominados como “usuarios”, es por ello que también, dentro de la clasificación externa, existe una diferencia entre los “clientes externos” y los “grupos de usuarios”. Ambos, no deben ser clasificados dentro del mismo grupo ya que los clientes que solicitan el desarrollo del software no son los mismos que los grupos de usuarios al tener características que los diferencian como por ejemplo las experiencias de usuarios, las cuales son registros de uso del software que pueden cambiar el diseño del software.

En el caso de los proveedores, éstos deben ser clasificados como “funcionales” o internos ya que por la naturaleza de las herramientas informáticas que se utilizan dentro del desarrollo de proyectos de software, los recursos que se necesitan mayoritariamente durante dicho proceso, pueden ser solicitadas de manera interna por medio de aquellos stakeholders que tienen roles internos funcionales y técnicos sin la necesidad de consultar proveedores externos los cuales brindan recursos estratégicos diferentes como lo son, capacitaciones en el uso de nuevas herramientas, venta de licencias para el uso de software privado etc. Para el caso de los “proveedores” y “contratistas” éstos se diferencian en tipo de servicio que prestan ya que por ejemplo los proveedores de un proyecto de software específico proporcionarán un servicio único y exclusivo para su cliente a diferencia de los contratistas que para un proyecto de software son agentes externos o “consultores” que buscan proporcionar recursos adecuados y a la medida del problema o situación que pueda surgir durante el desarrollo del proyecto.

Considerando la presente clasificación, hay que colocar, como mínimo, nombre, información de contacto, posición dentro de la organización, rol dentro del proyecto, su expectativa primordial, su principal requerimiento, e influencia potencial en el proyecto o fase durante el ciclo de vida. Por último, se debe incluir una identificación de los stakeholders, ya sean internos o externos. La estrategia de administración de stakeholders define el enfoque que debería ayudar para obtener el apoyo y reducir los impactos negativos de la identificación de los stakeholders. Una forma común de representar este dato es dentro de una matriz de Análisis de Estrategia de los Stakeholders, a fin de que los elementos como: la lista de los stakeholders claves, quienes pueden afectar significativamente al proyecto; el nivel de participación deseado para cada uno de los stakeholders claves identificados y la estrategia potencial para ganar el apoyo o reducir obstáculos.

Puesto que la **Estrategia de Administración de Stakeholders** puede contener algún dato que es subjetivo y podría ser considerado sensible, el administrador del proyecto debe asegurarse de ser discreto al momento de comunicar o compartir los datos sobre los documentos con acceso de lectura o incluido en un documento compartido. El **registro de stakeholders** y la **estrategia de administración de stakeholders** serán insumos esenciales para los procesos de plan de comunicación y manejo de las expectativas de los stakeholders.

## 2.4 LA ESTRUCTURA ORGANIZACIONAL

### 2.4.1 CLASES DE ESTRUCTURA ORGANIZACIONAL

Una vez que ya hemos visto los factores que influyen a la hora de diseñar la estructura de las organizaciones y los diferentes criterios que habitualmente se emplean para ponerla en marcha, se resumen las principales clases de estructura que podemos encontrar hoy en día en las organizaciones.

**Tabla 11** Clases de estructuras (González, 2008).

Clase de estructura	Descripción
<b>Estructura de tipo simple.</b>	Esta es la más primaria y la más sencilla. Se suele producir siempre en las PYMES y empresas de reciente creación. En estas estructuras, existe un empresario que hace las veces de comercial, de contable, de aprovisionamiento, producción. Son estructuras donde no existe jerarquía y los empleados hacen de todo un poco, pero es el empresario la figura vital de esta estructura. Este diseño solo tiene razón de ser al comienzo de la empresa, ya que en cuanto se inicia el crecimiento, se necesita delegar y repartir funciones y responsabilidades.
<b>Estructura funcional</b>	Es el paso lógico tras la estructura simple. El negocio comienza a crecer y se necesita una organización que divida el trabajo y amplíe sus posibilidades. Comienza entonces un proceso de especialización centrado, sobre todo, en las funciones. Dentro de estas funciones, las más clásicas y las básicas son la de mercadotecnia y comercial, operaciones con aprovisionamiento, logística interna y externa, producción, recursos humanos y económico-financiero. Este tipo de estructuras permite una mayor planificación, organización, dirección y control de las actividades que estamos realizando.
<b>Estructura divisional</b>	Esta estructura es típica de organizaciones que están creciendo o que ya poseen un tamaño destacado y que trabajan en diferentes zonas geográficas o productos diversificados. Las divisiones se crean no ya por funciones, sino por propósito, es decir, que son divisiones autónomas con capacidad para tomar sus propias decisiones y que funcionan de forma descentralizada, como centros de beneficios. Se encuentran coordinadas por una sede central que ofrece servicios de staff.
<b>Estructura holding</b>	Es un paso descentralizado más de la estructura divisional. Si la estructura divisional es típica de organizaciones que ofrecen productos-servicios diversificados pero relacionados, la estructura de holding es la típica de empresas que ofrecen productos-servicios diversificados y no relacionados. Esto significa que cada actividad empresarial representa un negocio casi independiente, con sus propias estructuras y que tan solo depende de la matriz mediante una relación de control simple, centrado sobre todo en lo económico-financiero.
	Quizás esta estructura sea una de las más alabadas y criticadas por el mundo de la administración en los últimos tiempos. Por su complejidad en su puesta en marcha, ha vivido grandes fracasos y alguno que otro

Clase de estructura	Descripción
<b>Estructura matricial</b>	<p>éxito.</p> <p>La estructura matricial es propia de empresas que quieren trabajar con la misma profundidad e intensidad que las funcionales, pero con la flexibilidad de las divisionales.</p>
<b>Estructura federal</b>	<p>Digamos que esta estructura intenta tomar lo bueno de la estructura divisional y de la de holding. Se queda en un camino intermedio, ya que las divisiones de la empresa funcionan como casi empresas, pero todavía no llegan a ser holding, puesto que existe un pequeño órgano central de dirección que lidera la estrategia, ofrece servicios económicos financieros y alguna unidad de apoyo que no pueda ser descentralizada.</p>
<b>Estructura en trébol</b>	<p>La llegada del siglo XXI ha traído consigo el despegue definitivo de las nuevas tecnologías, la personalización masiva, la desaparición de las fronteras sectoriales, la obsolescencia inmediata, la valoración del talento y otros factores que obligan a muchos sectores y empresas a reducir su estructura y a convertirse en organizaciones tendentes a la horizontalidad y a la ausencia de jerarquías frente a la verticalidad y jerarquización de la época industrial.</p>
<b>Estructura en red</b>	<p>Si vamos un paso más en ese adelgazamiento empresarial de las estructuras, nos encontramos con las estructuras en red, donde un pequeño núcleo con sus competencias básicas, dirige la internalización del resto de los servicios que ofrece, contratando empresas diversas como si fuera una red, según las necesidades que va teniendo. Este tipo de estructura posee los mismos inconvenientes y ventajas que la estructura en trébol, pero con un riesgo mayor porque todavía hemos reducido más el núcleo del negocio.</p>
<b>Estructuras virtuales</b>	<p>Es el mismo concepto que las de trébol y en red. Un núcleo muy pequeño que se une a otras empresas para producir el producto demandado de forma temporal. La radicalidad de esta estructura es que las alianzas ya son virtuales con empresas de cualquier entorno y lugar y las ventajas y desventajas son las mismas que en las otras estructuras mencionadas.</p>

Solamente una tendencia parece clara, y esta es la cada vez mayor horizontalidad de las organizaciones, con estructuras menos jerárquicas, la potenciación de la autogestión de los equipos, la ampliación del ámbito de control de las personas y la búsqueda de una mayor flexibilidad que aumente la rapidez en la respuesta a las necesidades del cliente.

Visto todos estos aspectos, podemos concluir que **no** existe una única estructura válida para todas las empresas, y que esta será un reflejo de una serie de factores que influyen en el desarrollo de la organización. Para el caso específico de las **empresas desarrolladoras de software** y debido a la estructura presentada en secciones anteriores, los grupos encargados

en el desarrollo de proyectos de éste dominio responden a diversas estructuras, de entre las principales se encuentran la estructura **FUNCIONAL** ya que comienza con procesos de especialización centrado en funciones (empresas de desarrollo de software de bajo y mediano costo). También se encuentran la estructura **DIVISIONAL** ya que al ser el desarrollo de software un rubro de empresas de servicio, trabajan en diferentes zonas geográficas hacia mercados diversificados.

Sin embargo, debido a los cambios tecnológicos y a una constante diversificación económica, se puede ver que las empresas desarrolladoras de Software del siglo XXI responden a una estructura en **TREBOL** ya que éstas responden a la personalización masiva, desaparición de las fronteras sectoriales, la valoración de talento reduciendo jerarquías laborales.

## 2.4.2 FACTORES CLAVE PARA DISEÑAR LA ESTRUCTURA DE UNA ORGANIZACIÓN

La estructura la cual es la manera como la gente se organiza, junto a otras variables, alcanzar nuestros objetivos fijados y en el caso de las organizaciones dedicadas al desarrollo de software, significa además obtener un mayor control en sus procesos y métodos para la realización de proyectos es cuestión.

Se trata de conocer bien nuestro entorno y nuestros recursos y capacidades para así poder ajustar el modelo de estructura adecuado a nuestra circunstancia.

**Tabla 12** Factores que afectan las estructuras organizacionales (Fernández, 2009)

Factor	Descripción
<b>Entorno económico</b>	El <b>entorno económico</b> en el que se va a mover la organización desarrolladora de software no afectará a su escalabilidad.
<b>Política</b>	Nuestra organización en términos de consultoría o desarrollo de software verá limitada su estructura a los condicionantes políticos que encontremos en los lugares en los que nos instalemos (medio rural o urbano).
<b>Globalización</b>	Hoy en día, cualquier compañía de cualquier país puede competir en cualquier lugar. Esto obliga a que las empresas de desarrollo de software, por pequeñas que sean, deban comenzar a plantearse la internacionalización, lo que afecta directamente a la estructura de la organización, ya que comenzará a plantearse la idoneidad de la centralización frente a la localización y la contratación de agentes extranjeros.
	Muchos modelos japoneses fracasaron en Estados Unidos debido a que las culturas, los valores y las normas de cada

Factor	Descripción
<b>Cultura de los países</b>	país son diferentes y las estructuras deben adaptarse a cada entorno. No es posible trasplantar directamente las buenas prácticas de una cultura a otra sin una adaptación.
<b>Desarrollo Tecnológico</b>	En lugares con un alto desarrollo tecnológico, puede existir una tendencia mayor a la horizontalidad de las organizaciones, ya que la tecnología permite la transversalidad y sistemas de información más rápidos que permiten un mejor flujo de información.
<b>Sector objetivo</b>	Una variable fundamental a la hora de diseñar nuestra estructura es el <b>sector en el que nos vamos a mover</b> .
<b>Mercado</b>	No es lo mismo estar en un mercado estable, donde los cambios son escasos, que en un entorno dinámico o en uno turbulento, donde cada día se producen nuevos acontecimientos que provocan movimientos en el interior de la organización.
<b>Proveedores</b>	Dependiendo de la integración que tengamos o que necesitemos, afectará a la estructura.
<b>Distribuidores</b>	Nuestra relación con los <b>distribuidores</b> también definirá la estructura, ya que si potenciamos un trabajo de relación a medio y largo plazo con nuestros distribuidores y minoristas, lo normal es que acabemos creando estructuras con inclusiones de departamentos como el de “trade marketing”, que van a cambiar el enfoque organizativo y que van a provocar alineamientos de objetivos.
<b>Consumidor</b>	Por supuesto, en este siglo XXI donde todas las organizaciones dicen estar orientadas al cliente, el comportamiento y el perfil del <b>consumidor</b> es fundamental
<b>Stakeholders</b>	No debemos olvidar tampoco la influencia de los <b>stakeholders</b> , que han hecho que muchas organizaciones cambien su estructura para darles más servicio y atención.
<b>Factores competitivos clave de nuestro mercado</b>	Tampoco debemos dejar de lado los <b>factores competitivos clave en nuestro mercado</b> , ya que si uno de los factores clave es la rapidez y flexibilidad en el servicio, nuestra estructura deberá adoptar una forma que permita responder a esa necesidad.
	Normalmente, las empresas con un capital más desarrollado, con empleados que actúan como colaboradores

<b>Factor</b>	<b>Descripción</b>
<b>Capital Intelectual</b>	semiautónomos que tienen un gran poder de autogestión, con estructuras flexibles y con un flujo de relaciones muy extendido, tendrán estructuras más horizontales y rápidas en la respuesta, con poca jerarquía.
<b>Procesos e interrelaciones y la cadena de valor</b>	Si estamos en un negocio donde el sistema de valor (relaciones entre todos los agentes del mercado que hacen que yo lleve mi producto al consumidor) es complejo y con multitud de actores, es probable que requiera una estructura que permita trabajar más en profundidad con dichos actores.
<b>La misión, la visión, la cultura empresarial, los objetivos y las estrategias fijadas</b>	Si tenemos un objetivo de crecimiento, nuestra estructura probablemente tenga más niveles jerárquicos y se tornará más compleja.

#### 2.4.3 CRITERIOS PARA DISEÑAR UNA ESTRUCTURA ORGANIZACIONAL DE EMPRESAS

Existen una serie de criterios básicos que se suelen emplear a la hora de diseñar una estructura en la organización.

**Tabla 13** Factores que se toman en cuenta para la construcción de estructuras organizacionales (*Friendle, 2007*).

<b>Criterio</b>	<b>Definición</b>
<b>Funcional</b>	Dependiendo de las funciones que se desempeñen dentro de la empresa, entendiendo estas como una serie de actividades agrupadas que emplean unos recursos humanos y técnicos similares y que sirven para realizar una determinada función, así iremos departamentalizando nuestra organización.
<b>Producto-servicio</b>	Dependiendo de los productos o servicios que ofrezcamos, así iremos organizando la empresa. Tendremos departamentos encargados de un producto con todas las funciones para fabricar el mismo y otro departamento para fabricar otro producto con todas las funciones también.
<b>Cliente</b>	Podemos organizarnos en torno a los clientes a los que servimos, ya sea del canal de distribución o el cliente final al que vendemos el producto o servicio. Existiría un departamento de atención a un determinado cliente donde se desarrollarían todas las funciones.
<b>Zona</b>	Organizaríamos aquí nuestra estructura en función de los mercados geográficos atendidos. Es muy típico de estructuras divisionales, donde una

Criterio	Definición
<b>geográfica</b>	división atiende a una zona del país o del mundo y en esa división se desempeñan todas las funciones.
<b>Células de trabajo</b>	A menudo las organizaciones se estructuran en una combinación mixta de alguno de los anteriores criterios y se le añade una célula de trabajo independiente que es capaz de realizar todas las funciones para fabricar el producto en dicha célula formada por un grupo de trabajadores. Con esto lo que se pretende es no perder las economías de escala pero dar una respuesta más rápida y personalizada a las demandas del cliente. Es muy típico de los sistemas lean como el “ <i>Just in time</i> ”.
<b>Procesos</b>	Por último, está el criterio de organizar la estructura por <b>procesos</b> , donde los departamentos siguen teniendo algo de funcional, pero lo que los une son los procesos por los que se va transformando el producto o servicio.

Los criterios de células de trabajo y procesos suelen afectar sobre todo a la organización desde el punto de vista de la producción, mientras que los criterios de clientes suelen referirse a la parte comercial.

A pesar de todos estos criterios, el primer criterio, el funcional, siempre se mantiene en todas las organizaciones, ya que debe existir una parte de la empresa que se organice por criterios de las funciones que se llevan a cabo.

#### 2.4.4 CAPAS ESTRATÉGICAS DE ANÁLISIS PARA DISEÑAR UNA ESTRUCTURA ORGANIZACIONAL ORIENTADA EN STAKEHOLDERS

Los factores clave, así como los criterios para diseñar la estructura de la organización marcan la pauta para entender el nivel de análisis estratégico a implementar, priorizando para dicho fin las partes que relacionan a los proyectos de software con los stakeholders. Por tal motivo, se pueden remarcar aquellos factores y criterios que puedan proporcionar mayor información hacia el estudio de las estructuras de stakeholders dentro de los proyectos de software.

Si se considera que uno de los factores clave para el diseño de una estructura de cualquier organización son sus **stakeholders**, y que dentro de dicho diseño existe un criterio **funcional** clave que pueda relacionar a los grupos de interés con las diferentes actividades de dicha organización, se puede remarcar que existen variables que pueden maximizar la inclusión de los grupos de stakeholders con el desarrollo de proyectos de software. Dichas variables enmarcan expresiones de satisfacción como los **intereses**, necesidad de inclusión como la **identidad**, formas de comunicación efectivas como la **relación entre los individuos** y formas de aprendizaje como la **transmisión del conocimiento**.

#### **2.4.4.1 EL INTERÉS**

En términos empresariales y de grupos de trabajo, medir los éxitos y fracasos de los proyectos de software puede ser el inicio en el cual se puedan establecer estudios acerca de las acciones para su culminación exitosa o en su caso la mejora de procesos de desarrollo. Uno de los factores que explican las acciones de los individuos es sin duda su comportamiento en términos de acciones ejercidas en los grupos de trabajo asociados. Se dice que el comportamiento de los distintos grupos de interés manifiesta los medios por los cuales los stakeholders intentan influir a una organización (Frooman, 1984).

Existen varios tipos de intereses por los cuales se debe prestar atención y por los cuales se puede explicar la acción de los distintos grupos de interés:

1. Intereses basados en los derechos (morales, legales, etc.)
2. Intereses basados en las propiedades
3. Intereses basados en los títulos legales

Aunque también al analizar cada uno de ellos, es evidente notar que cuando existe el interés, puede existir algún tipo de riesgo con la organización en cuestión para lograr dichos cometidos. Es por eso que tanto los grupos de interés como la organización y en especial sus dirigentes deben tomar en cuenta ambas partes siendo la última quien priorice a sus stakeholders, ya que estarán cuidando su interés de manera permanente lo que podría beneficiar a los proyectos mismos.

#### **2.4.4.2 LA IDENTIDAD**

La identidad es un conjunto lógico de proposiciones que una persona usa para describirse a sí misma o describirse a sí misma con los demás, y está acompañada por el valor de la solidaridad la cual puede actuar en éste caso como una poderosa catarsis para la movilización de los grupos de interés, así como convertir el objetivo de la definición de sí mismos hacia las personas, sus acciones enfocadas a dicha movilización la cual es una mezcla compleja de identidad por parte de un colectivo de personas en sí. (Johanson, Laraña & Gusfield, 2013)

#### **2.4.4.3 LA RELACIÓN ENTRE LOS INDIVIDUOS**

La relación entre los individuos es considerada como parte vital en la construcción de estructuras de stakeholders a medida en que existan acciones dirigidas hacia todos los grupos de interés las cuales puedan mejorar la construcción de proyectos de software. Se puede decir que las relaciones entre los miembros pueden jugar un papel importante. Para que exista una perspectiva en la relación entre grupos de interés, deben considerarse internamente a los vínculos entre los miembros que los conforman. Un vínculo puede existir desde el hecho de relacionarse con miembros de la organización, interacción laboral o tomar nuevas responsabilidades. Así, los líderes pueden tener vínculos de liderazgo entre los desarrolladores de software por medio de las ordenes que deban ejecutar para diseñar las herramientas

informáticas o las instrucciones directivas que un líder deba seguir provenientes de stakeholders superiores a él, como lo es el gerente del proyecto o el propietario del producto.

#### **2.4.4.4 LA TRANSMISIÓN DE CONOCIMIENTO**

Se considera importante determinar la intensidad de los vínculos entre los diferentes miembros de una organización o equipo de trabajo para que exista un cambio en el enfoque comunicativo del equipo. Se pretende aprovechar dichos vínculos para establecer reglas de transmisión del conocimiento que pueda ser al mismo tiempo efectivo y eficaz durante el ciclo de desarrollo del proyecto. Se considera que el hecho de crear un cambio en los roles del proyecto puede ayudar a incrementar los conocimientos de cada stakeholder, conocer mejor el negocio e incrementar las habilidades para futuros cambios de rol y profesionales.

La transmisión contemporánea del conocimiento sucede en el interior de una sociedad y dicha transmisión forma una unidad inseparable, una nueva totalidad en la que las modificaciones sociales cambian las formas de transmisión de conocimiento y los nuevos medios que pueden modificar a la sociedad. Es en éste nivel en el cual se pueden llegar a construir teorías sobre la forma de transmitir el conocimiento y su importancia en la edificación de todo ser humano, es por ello que para poder crear nuevas formas de inclusión dentro de un grupo de trabajo como lo son los grupos de stakeholders dentro de las organizaciones, se deben crear nuevos vínculos que permitan que el conocimiento transmitido es efectivo y benéfico para cualquier tipo de organización (*Riaza, 2002*).

## **2.5 EL ENFOQUE DE PLANEACIÓN Y DE SISTEMAS**

### **2.5.1 PLANEACIÓN INTERACTIVA**

Como parte de los hallazgos encontrados en el enfoque a los stakeholders y a la relación existente con proyectos de software se tiene ahora la necesidad de analizar aquel proceso de planeación que pueda conducir los esfuerzos en la identificación de stakeholders, diseñar una estructura que responda al enfoque a los grupos de interés pero que además brinde soporte a bases fundamentales que puedan diseñarse a la interacción entre stakeholders en el presente trabajo. Dicha forma de interacción está contenida dentro del **proceso de planeación interactiva**.

La planeación interactiva tiene como principios fundamentales que la planeación debe ser continua, holística y participativa y tiene como su elemento más significativo, la idea que las fases del proceso de planeación debe estar centrado en el diseño de “un futuro idealizado”. Saca a relucir la idea de “Planear o ser planeado”. Responde a la idea de las organizaciones dependen de una “coalición” de stakeholders y reafirma la idea de que son “culturas”, así como reconocer que pueden ser designados para promover el conocimiento similar a las funciones del “cerebro”.

Este tipo de planeación propuesta por Ackoff está enfocada a asumir problemas de contexto considerados como complejos-plurales, en el punto en que se toma en cuenta ambas, la complejidad de las organizaciones a las cuales los administradores deben encarar día con día y al pluralismo inherente en dichas organizaciones concebidas como el punto medular del estudio de sistemas. La idea en que uno de las mayores pérdidas de tiempo durante la vida de los planeadores es la manera de cuantificar la calidad de vida por lo que es posible planear de la mejor manera para los otros, puede ser trasladado una vez de que es reconocido el hecho de que las personas deben planear para ellas mismas. Todo lo que se necesita es una metodología de planeación, en el cual las personas puedan usar con ayuda de los planeadores y la cual pueda desarrollar sus propios ideales y valores en conjunto.

Un motivo importante por el cual se propone el uso de este tipo de planeación, es porque dentro de sus principios menciona que los administradores deben dirigirse a servir a las tres capas, desarrollar a todos los stakeholders de la organización y remover cualquier conflicto entre ellos. Si se logra dicho efecto, todos los “stakeholders” de la organización, cualesquiera que sean internos o externos, continuarán siguiendo sus propios intereses a pesar de ello, y la organización permanecerá viable y efectiva. Menciona Ackoff dentro ésta concepción que la planeación debe ser participativa y tiene que ver acerca de permitir a otros planear efectivamente para ellos mismos.

Menciona también que los planeadores del interactivismo (corriente de planeación) no desean regresar al pasado, para dejar las cosas como son, o aceptar un futuro inevitable. Ellos toman en cuenta el pasado, el presente y predicciones acerca del futuro, pero las usan solo como entradas dentro del proceso de planeación el cual es encargado de designar un futuro deseado e inventar maneras de llevarlo a cabo. Ellos creen que el futuro puede ser afectado por lo que la organización y sus **stakeholders** hagan ahora.

Los principios en que se basa la planeación interactiva son los siguientes:

## 1. PARTICIPATIVO:

El proceso de planeación es más importante que el plan producido. Se trata de estar envuelto en el proceso de planeación en el cual los miembros de la organización logran entender a la organización y el rol que pueden tener dentro de ella. El principio de que nadie puede planear por los demás ya que puede ser que se olviden los beneficios principales de la planeación. La segunda idea es la que aquellos que sean afectados por el proceso de planeación deben ser tomados en cuenta en él. (Value-full).

Dicho estado del principio participativo, por el contrario, menciona que todos los **stakeholders** deben participar en uno o en varios estados del proceso de planeación. Para ayudar en el proceso de institucionalización, Ackoff ha realizado un diseño organizacional para la planeación participativa. En dicho diseño los diferentes niveles de la empresa son organizados dentro de los principios de planeación, con responsables de cada nivel siendo miembros de dichos niveles en tres niveles: El propio nivel y el nivel inmediato superior e inferior. En el nivel superior, representativos de los stakeholders externos son representados en el Nivel 1 y, en el nivel más bajo, todos los trabajadores que son miembros de las unidades del Nivel 3. A pesar de eso, este compromiso debe parecer difícil de manejar y puede implicar mucho tiempo al

principio, con algunos administradores estando a cargo quizá de diez líneas, la experiencia de Ackoff dice que los beneficios en términos de actividades coordinadas, integración organizacional y motivación (la que se tiene cuando un administrador debe tener en cada una de sus actividades) son muy considerables.

Una de las preguntas más frecuentes que se pueden hacer dentro del principio participativo es: ¿Cuál es el rol de los planeadores en dicho proceso? Y ¿Cómo reaccionarían los altos administrativos con el hecho de tener que relacionarse con otros stakeholders, especialmente aquellos de niveles más bajos, en el proceso de planeación? Ackoff explica que los planeadores profesionales no son excluidos de ninguna manera del proceso, es simplemente que su rol ha cambiado. **Ellos usan su experiencia ahora no solo para planear para los otros, si no para que los otros planeen por sí mismos.**

En el caso de la inclusión de **stakeholders**, Ackoff considera que puede ser renuente el hecho de permitir una participación completa. Por otro lado, en este caso, si otros grupos pueden ganar aceptación desde el principio como “**consultores**”, es normalmente posible incrementar su participación en el proceso durante el tiempo.

## 2. CONTINUIDAD:

Los valores que tienen los stakeholders de las organizaciones cambiarán durante el tiempo y se necesitarán cambios adecuados en el plan, así como eventos inesperados pueden ocurrir. El plan puede no funcionar como se espera, o los cambios en los ambientes de la organización pueden cambiar la situación en la cual se encuentra ella misma. Ningún plan puede predecir nada antes de que ocurran las cosas, por lo que los planes, sobre el principio de continuidad debe ser constantemente revisado.

## 3. HOLISTICO:

Se debe planear simultáneamente e interdependientemente tomando en cuenta cada nivel y partes del “sistema” tanto como sea posible.

Esto se puede explicar como sigue:

a) Un “principio de **coordinación**”, en el que los estados que unen los mismos niveles deben ser planeados al mismo tiempo y en conjunto, ya que se trata de darle importancia a la **interacción** entre las diferentes unidades más que su acciones interdependientes las cuales pueden originar mayores dificultades.

b) Un “principio de **integración**”, el cual remarca unidades en los diferentes niveles estén contenidos en el plan simultáneamente y juntos, porque las decisiones tomadas en un solo nivel tendrá normalmente efecto en otros niveles al mismo tiempo.

### 2.5.2 PLANEACIÓN TRANSACTIVA

Además del enfoque de planeación interactiva, donde se considera un análisis especial a los stakeholders, es necesario adicionar un enfoque que pueda relacionar de manera efectiva a

dichos grupos de interés para establecer una mayor efectividad en el diseño de una propuesta efectiva orientada al éxito de los proyectos de software por medio de proceso de comunicación efectivos.

La planeación transactiva cambia el conocimiento en acción a través de una secuencia inquebrantable de relaciones interpersonales (*Friedmann, 1987*). Como un estilo particular de planeación, puede ser aplicado a distribuir e innovar. Se incluye dentro del presente análisis de proyectos de software y stakeholders ya que por el presente enfoque de planeación es una respuesta al creciente hueco en la comunicación entre los planeadores técnicos y sus clientes, el cual es uno de los objetivos a considerar dentro del desarrollo de proyectos de software con un enfoque en sus stakeholders. Dicho enfoque de planeación es considerado de igual manera ya que es una respuesta al creciente hueco en la comunicación entre los planeadores técnicos y sus clientes.

Se puede remarcar que dentro del presente proceso de planeación, el planeador trabaja principalmente con el conocimiento procesado abstraído del mundo y manipulado de acuerdo a ciertos postulados de teoría y el método científico; su cliente trabaja principalmente desde el conocimiento personal, actúa directamente desde la experiencia. A pesar de que el conocimiento personal es mucho más rico en contenido y en su habilidad de diferenciar los detalles de la vida diaria, es menos sistematizado y ordenado que el conocimiento procesado. Si el hueco de comunicación entre el planeador y el cliente quiere ser cerrado, se necesita de una serie continua de transacciones verbales personales y primarias, en donde el conocimiento procesado es fusionado con el conocimiento personal y ambos son fusionados en acción (*Friedmann, 1987*).

En la planeación transactiva, se distinguen existen dos niveles de comunicación:

1. Centrado en la persona: Supone una relación a todas las formas de relaciones humanas. Esto es lo que se llama "*la vida del diálogo*".
2. En el asunto de interés: se sustenta en la primera relación de diálogo y no puede ser entendida independientemente.

Por lo tanto, la planeación transactiva es cimentada en el aumento del diálogo, es un proceso del aprendizaje mutuo. Es por ello que para fines de encontrar una manera de justificar la importancia entre el establecimiento de mecanismos que incrementen las formas de diálogo entre los distintos stakeholders en proyectos de software, se hace uso de los principios de dialogo y aprendizaje mutuo que pueda ser considerado en futuras implementaciones dentro del presente trabajo.

### **2.5.3 LA RESPONSABILIDAD SOCIAL CORPORATIVA COMO MARCO CONDUCTOR DEL DESARROLLO DE PROYECTOS DE SOFTWARE**

La razón principal de estudiar, analizar y concluir acerca del uso del software es el de la obtención de recursos teóricos y metodológicos que aporten valor al enfoque a stakeholders, objetivo clave de ésta investigación y por consiguiente, orientar esfuerzos en la mejora de proyectos de software en términos ya sea de costo, tiempo y/o beneficio, además de basar principios y buenas prácticas que siguen las prácticas en el desarrollo de proyectos de software.

Se analiza entonces un recurso orientado a la situación actual del proyecto en términos de funcionamiento y organizacionales tanto interna como externa del desarrollo de software, es decir, las estructuras responsables que hacen funcionar a una empresa por medio del objetivo primordial de ellas: La creación de valor.

Para analizar la creación de valor dentro de la empresa que desarrolla software es vital hacer uso de algunas ventajas que se pueden presentar en el uso de prácticas organizacionales innovadoras, responsables en su estructura y orientadas a la satisfacción de los stakeholders. El desarrollo de software da importancia al stakeholder denominado “propietario del producto”, como por ejemplo en el caso del software libre al usuario y en el del privado el propietario del producto. Podemos definir al propietario del producto y a sus accionistas relacionados junto con las políticas creadas como el gobierno de la empresa, donde se establecen los mecanismos internos del control directivo ya que la adopción de un enfoque de stakeholders en la toma de decisiones es, en última instancia, responsabilidad de los equipos de alta dirección (EAD) de la empresa ( Cennamo et al., 2012; Quazi, 2003; Swanzon, 2008) y atendiendo a la literatura , podría depender de las dinámicas de comportamiento y relaciones entre sus miembros ( Sharma y Sharma, 2011; Tullberg, 2012; Wong, Ormiston y Tetlock 2011).

Entendemos como Gobierno de la empresa a los mecanismos internos de control directivo para que dichos mecanismos de gobierno tengan efecto es necesario recurrir al dimensionamiento de los valores que se tienen en las distintas empresas, instituciones y/o negocios los cuales tienen como actividad principal el desarrollo de proyectos de software.

Cuando se habla de valores de la empresa, lo que define hoy en día su estudio es lo que se conoce como responsabilidad social corporativa (RSC), el cual transforma la fórmula clásica del gobierno centrada en la relación bilateral entre accionistas y directivos en otra multilateral en la que se participan todos los grupos de interés.

La RSC es por tanto un marco donde el presente trabajo se basará para ejercer acciones destinadas a todos los grupos de interés que participan directamente proyecto de software. Es importante mencionar que las organizaciones de software que adopten esta filosofía son aquellas que quieren ejercer un efecto directo en los activos de la empresa, pero no están obligadas a seguirla a menos que las EAD decidan implementarla por lo que es totalmente VOLUNTARIO para dichas organizaciones.

Las acciones que se tomen en las organizaciones orientadas a desarrollar software pueden optar por la RSC si quieren orientar esfuerzos a los grupos de interés ya que modifica el

proceso de toma de decisiones al añadir criterios de eficiencia económica además de la consideración del impacto ambiental y social de las actividades de la empresa.

La empresa de software que decida optar por la implementación de la RSC debe elegir los niveles de responsabilidad que quiere asumir, incorporando en su sistema de objetivos, no solo los propios de los accionistas, sino también el resto de grupos sociales participantes de la misma. Por ello, además de orientar esfuerzos a la culminación de proyectos, debe entender si es bueno orientar actividades encaminadas a mejorar la relación con los grupos de interés y el efecto que puede tener en el desarrollo de cualquier proyecto de software.

Los factores que influyen en la decisión de la adopción en una empresa de la RSC son dos: Legales y políticos. Los que interesan al presente estudio son los factores políticos ya que atienden a grupos de interés (stakeholders) con los que se relaciona la empresa especialmente con aquellos que han sido valorados como más relevantes.

Por lo tanto es imprescindible saber para qué serviría la RSC en este estudio. Una vez que la empresa oriente sus actividades en adoptar filosofías de RSC se propone:

- Una estrategia encaminada a obtener:

1. Nuevas fuentes de financiamiento.
2. Implantar evaluadores de desempeño organizacional que permita la mejora continua y generar nexos de unión con los grupos de interés involucrados en el desarrollo de software.
3. Herramienta interna de gestión

- En el desarrollo de software:

1. Desarrollo de herramientas y métricas para entender el desarrollo colaborativo.
2. Desarrollo de una serie de protocolos para evaluar los procesos de desarrollo centrándose en procesos tales como la madurez, supervivencia o estrategia de la organización que gestione el proyecto pero incorporando también aspectos funcionales y operativos.

Una vez identificados los stakeholders principales en el desarrollo de proyectos de software e implementado actividades de RSC se pretende encontrar los elementos de diseño que ayuden a establecer las bases metodológicas para que los proyectos de software sean beneficiados en términos del desarrollo y de sus stakeholders asociados para así poder identificar fácilmente las áreas que incrementen la eficiencia de los grupos de interés en dichos proyectos.

## **2.6 ESTRATEGIA DE INVESTIGACIÓN**

La complejidad a la que se enfrentan las consultorías en TI, así como desarrolladoras de software y productos tecnológicos impacta en su mayoría dentro de la eficiencia, los alcances, el costo y los beneficios de los proyectos que se llevan a cabo y por lo tanto se presentan dificultades que traspasan a la organización como la relación con los clientes, relación laboral de los equipos de trabajo, relación con los niveles gerenciales, re trabajos, calidad de vida etc.

Existen diversas herramientas que ayudan a optimizar los procesos de diseño, control y organización de proyectos de software las cuales contemplan como una de sus etapas el estudio y la priorización de la estructura organizacional haciendo hincapié en los stakeholders como los son: Metodologías de Madurez Organizacional (Ej. Capability Maturity Model Integration CMMI), metodologías de Administración de Proyectos (Ej. PMBOK) y estándares de calidad y control como lo son ISO-9000, etc; pero a pesar de contar con estrictas reglas de inclusión e importancia de stakeholders en los proyectos en cuestión, no contempla un análisis profundo de los motivos que hacen que los stakeholders actúen para mejorar los proyectos de software, se organicen de acuerdo a las expectativas de los proyectos y se consideren sus necesidades como prioritarias.

Para este estudio, se parte de la idea de que existen diversos factores los cuales afectan directa o indirectamente al éxito de los proyectos (herramientas de trabajo, metodologías, entorno de desarrollo, ambiente laboral, competencia en el mercado) y uno de ellos es el factor organizacional, específicamente, la estructura de stakeholders cuyo análisis es motivado principalmente por el estudio de variables que hacen actuar a los stakeholders y justifican sus acciones como lo son: el interés, la relación, la membresía y la identidad, resultado del estudio de diversas teorías que justifican sus acciones dentro de las organizaciones que desarrollan software.

### **2.6.1 ESTRATEGIA**

Se pretende estudiar diversas estructuras organizacionales que presenten características suficientes para mejorar los proyectos de software en términos de tiempo, esfuerzo, costo, etc. Dichas características pueden ser formas de integración de equipos de stakeholders, relación entre los distintos miembros, necesidades de los stakeholders principales al momento de diseñar, desarrollar y probar una herramienta o sistema de software, funciones principales de los miembros y su impacto en las actividades principales entre otras, cuya implementación en proyectos de software proporcione mejoras en función de sus relaciones y funciones así como establecer indicadores, métricas o métodos para verificar aquellas que ayuden a maximizar el éxito de los proyectos.

Al analizar las estructuras se pretende tipificarlas de tal manera en que se establezcan parámetros diversos como: comportamiento de stakeholders, impacto de sus partes interesadas, áreas críticas y áreas marginales así como priorizar áreas donde los esfuerzos en el desarrollo de proyectos de software presenten mayores retrasos. Para poder determinar las características idóneas se pretende construir estructuras que respondan a la optimización del éxito de los proyectos de software por medio de incluir un análisis por medio de capas donde se puedan diferenciar y clasificar por niveles las variables antes mencionadas.

La estrategia de investigación puede resumirse de la siguiente manera:

- 1.- Identificar el proceso de proyectos de Software.**
- 2.- Clasificar y diferenciar a los principales stakeholders de los proyectos tanto de software privado como libre para obtener conclusiones acerca de sus ventajas y desventajas en el desarrollo de proyectos de software.**
- 3.- Identificar las estructuras idóneas para proponer procesos planeados de mejora en proyectos de software.**

- 4.- Utilizar los procesos de mejora propuestos para diseñar herramientas orientadas a la gestión de stakeholders en proyectos de software.**
- 5.- Validar las herramientas diseñadas por medio de entrevistas a expertos en desarrollo de software, administración de proyectos o liderazgo empresarial.**

# ANÁLISIS, DISEÑO Y VALIDACIÓN DE ESTRUCTURAS Y MECANISMOS DE GESTIÓN DE STAKEHOLDERS ASOCIADOS A PROYECTOS DE SOFTWARE.

Para diferenciar a los stakeholders que participan en los proyectos de desarrollo en software de aquellos que no lo son, uno de los ejercicios más importantes a considerar es su identificación y en especial, tomar en cuenta diversos parámetros los cuales ayuden a discriminar aquellos stakeholders que no tienen relevancia o que no son considerados como un agente influyente en el desarrollo de dichos proyectos.

En el presente capítulo se pretende realizar un estudio de la identificación de estructuras de stakeholders en forma de *sociogramas*, así como determinar las áreas de impacto en las que se pueden diseñar dichas estructuras, divididas en dimensiones de desarrollo que ayuden a entender los principios de acción de los esfuerzos realizados a una posible culminación exitosa de proyectos de software.

También, dentro del análisis de las variables a considerar se pretende remarcar los recursos que se pueden implementar para que las estructuras de stakeholders puedan funcionar como un marco organizativo dentro de las empresas que desarrollan software y de tal manera impactar positivamente en términos de tiempo de desarrollo de los proyectos, el costo de construcción y al beneficio para la empresa y para los clientes del producto en cuestión.

Para poder identificar a los stakeholders de proyectos de software es necesario enfocar el análisis a los dos tipos de desarrollo de software existentes: software libre y privado (comercial). Se desea determinar y analizar los stakeholders de cada tipo debido a que ambos enfoques aportan valor al uso de software, comercialización y orientación a stakeholders y es necesario identificar buenas prácticas dentro de su estructura marco metodológico y de desarrollo de software con el objetivo de aportar recursos que impacten en el análisis de stakeholders y puedan beneficiar a proyectos de software.

### 3.1 IDENTIFICACIÓN DE STAKEHOLDERS EN PROYECTOS DE SOFTWARE

#### 3.1.1 SOFTWARE LIBRE

Las ventajas que puede proporcionar el software libre pueden ser aprovechadas en la presente investigación, por lo que se procede a identificar los principales grupos de interés así como detallar diferentes indicadores que analicen a los stakeholders que participan en desarrollos de software libre y por consiguiente, ayuden a que un proyecto de software aumente sus posibilidades de ser culminado con éxito.

Los stakeholders de un proyecto de software libre pueden diferenciarse según su participación en la empresa, pueden ser internos o externos. Para fines de análisis en el software libre se consideran tres grandes categorías: Internos, externos que financian o participan en el proceso de desarrollo y externos ajenos a la financiación o proceso de desarrollo (García-García,

Alonso de Magdaleno, 2013).

Para la elección e identificación de stakeholders se tomó en cuenta distintos aspectos los cuales ayudan a discriminar aquellos grupos que siendo parte del grupo de stakeholders a la empresa desarrolladora de software libre no repercuten directamente en el desarrollo del mismo, siendo que en éste estudio, es importante diferenciar aquellos que sean más próximos al desarrollo de software.

Los siguientes aspectos son:

1. Grupos de stakeholders similares en cualquier tipo de empresa desarrolladora de software.
2. Grupos de stakeholders asociados al software libre.
3. Indicadores de información relevante para los grupos de stakeholders basado en las características adaptadas de Ostrom (1990, 2005)

Se considera que existen dos grupos de stakeholders: interno y externos que financian o participan en el proceso de desarrollo los cuales responden a las necesidades del estudio ya que retoman directamente a quienes están encargados o son responsables del desarrollo de software.

Los stakeholders internos están caracterizador por tener mayor participación en el proceso de desarrollo. Sus intereses están dirigidos a obtener el mayor beneficio de sus acciones y que éstas sean recomenzadas. Su compromiso es mayor por lo que basan sus esfuerzos en el reconocimiento organizacional. Las acciones o comportamientos que caracterizan a éste grupo de stakeholders son las siguientes:

- 1.- Están involucrados en el desarrollo por lo que sus acciones tienen mayor impacto en el producto final.
- 2.- Están estrechamente relacionados por su nivel, rol y rango de participación.
- 3.- Tienen mayor poder en la toma de decisiones.
- 4.- Se encargan del diseño y desarrollo del producto final.
- 5.- Buscan tener el mayor beneficio del producto final.
- 6.- Pueden comunicarse con otros grupos de stakeholders de manera eficiente.
- 7.- Su comportamiento impacta en la organización.

**Tabla 14** Grupos de interés internos (García, Alonso, 2013).

CATEGORÍA	STAKEHOLDER
INTERNOS	Propietarios de la empresa o socios y miembros de la organización
	Alta dirección de la organización y liderazgo u órganos de gobierno del proyecto
	Empleados de la organización

Los mencionados grupos externos que se consideran en éste estudio repercuten en el desarrollo de software libre por distintas razones las cuales se justifican en el impacto que pueden tener en el uso y desarrollo de herramientas libres.

**Tabla 15** Externos que financian o participan en el proceso de desarrollo (*García, Alonso, 2013*).

CATEGORÍA	STAKEHOLDER
EXTERNOS QUE FINANCIAN O PARTICIPAN EN EL PROCESO DE DESARROLLO	Agentes que incorporan el desarrollo a sus productos de software o hardware.
	Colaboradores independientes ajenos a la organización
	Aliados en el desarrollo tecnológico y Universidades y grupos de investigación.
	Instituciones que apoyan o financian el desarrollo
	Donantes en general

Es importante notar que dentro de la tabla anterior los grupos denominados “Agentes que incorporan el desarrollo a sus productos de software o hardware” y “donantes en general” se pueden considerar como grupos que aprovechan o toman ventaja del software libre para satisfacer sus necesidades de terceros sin proporcionar ayuda ni aportaciones reales al desarrollo de software (readers por su significado en inglés) por lo que la tabla 15 puede ser reducida a tener solo 3 grupos de stakeholders.

Las actividades y comportamientos que caracterizan éste grupo de stakeholders son las siguientes:

- 1.- Buscan que el beneficio del producto final esté orientado a satisfacer una necesidad específica.
- 2.- Buscan obtener patrocinios de otras instituciones con necesidades e intereses similares.
- 3.- Aunque no son propietarios del producto final, aprovechan el conocimiento obtenido resultado del desarrollo del producto final.

**Tabla 16** Externos ajenos a la financiación o proceso de desarrollo (*García, Alonso, 2013*).

CATEGORÍA	STAKEHOLDER
EXTERNOS AJENOS A LA FINANCIACIÓN O PROCESO DE DESARROLLO	Clientes y usuarios finales del desarrollo
	Comunidades de otros desarrollos de código abierto
	Competidores software libre o privativo)
	Administraciones públicas
	Sociedad en general

El grupo considerado como “Externos ajenos a la financiación o proceso de desarrollo” en la tabla 17 agrupa aquellos stakeholders que si bien no participan en el desarrollo de software, están enfocados al producto final y por consiguiente son los que potencian los intereses de algunos grupos internos de desarrollo.

Las acciones características entre los stakeholders externos que financian o participan en el proceso de desarrollo son las siguientes:

1. Enfoque al producto final.
2. Su interés principal es el beneficio del software terminado para satisfacer necesidades particulares sin pagar algún tipo de licencia.
3. La relación entre los distintos stakeholders es limitada.

**Tabla 17** Stakeholders externos ajenos a la financiación o proceso de desarrollo (*García, Alonso, 2013*).

CATEGORÍA	STAKEHOLDER
EXTERNOS QUE FINANCIAN O PARTICIPAN EN EL PROCESO DE DESARROLLO	Colaboradores independientes ajenos a la organización
	Aliados en el desarrollo tecnológico y Universidades y grupos de investigación.
	Instituciones que apoyan o financian el desarrollo

Como se puede observar, el desarrollo de software libre considera diversos grupos de stakeholders y los agrupa en áreas que presentan acciones y comportamientos propios que los identifican y los diferencian entre sí.

### 3.1.2 SOFTWARE PRIVADO

Diversas son las metodologías y procedimientos que ayudan a la identificación de stakeholders de manera formal y reglamentada en proyectos privados, de entre ellas dos de las más utilizadas: PMBOK y SCRUM. Por su parte PMBOK define a los stakeholders como empresas u organizaciones que participan activamente en el proyecto o cuyos intereses pueden verse afectados positivamente o negativamente por la ejecución y terminación del proyecto, es decir que son todos los interesados que pueden ejercer influencia sobre el proyecto, entregables y miembros del equipo. Para dicha metodología, la identificación de stakeholders es una prioridad ya que deben estar satisfechos para que el proyecto sea considerado como exitoso.

La segunda metodología que hace una gestión responsable de stakeholders es la SCRUM la cual junto con la metodología tomada del PMBOK logra identificar a los principales grupos de interés o stakeholders que participan en los proyectos de software de manera directa.

La categorización de los stakeholders que participan en desarrollos de software privados toma en cuenta dos instancias de inclusión y participación: *Stakeholders comprometidos* y *stakeholders involucrados*. Los stakeholders internos o comprometidos son nombrados de tal manera por el grado de participación en el proyecto de desarrollo y las acciones, actividades y decisiones que tomen en conjunto o de manera individual que pueden repercutir directamente en el proyecto (Tabla 18).

Por otro lado, los stakeholders denominados “involucrados” están definidos también por su grado de participación dentro del proyecto, son grupos de interés los cuales tienen sus propios códigos de comportamiento y basan sus decisiones y acciones a medida en que sus necesidades pueden ser cubiertas o bien cumpliendo con requerimientos de la misma empresa sin ser parte directa del proceso de desarrollo (Tabla 18)

*Nota: A diferencia de las tablas de los stakeholders de proyectos libres, las de proyectos privados presentan una columna con un identificador de acuerdo con el stakeholder asociado debido a que es necesario asignar un identificador a cada stakeholder de proyectos de software privado el cual pueda ser usado en el desarrollo de futuras herramientas o mecanismos diseñados para ésta investigación. Además, el objetivo de éste trabajo está orientado al desarrollo de software empresarial por lo que es necesario priorizar la presente clasificación de stakeholders.*

**Tabla 18** Stakeholders internos (comprometidos)

CATEGORÍA	STAKEHOLDER	ID
INTERNOS (COMPROMETIDOS)	Propietario del producto	SP-IC1
	Lider del proyecto	SP-IC2
	Equipo de desarrollo	SP-IC3
	Equipo de pruebas o testers	SP-IC4
	Consultores del proyecto	SP-IC5

**Tabla 19** Stakeholders ajenos al proceso de desarrollo (involucrados)

CATEGORÍA	STAKEHOLDER	ID
AJENOS AL PROCESO DE DESARROLLO (INVOLUCRADOS)	Cientes finales	SP-AD1
	Usuarios finales	SP-AD2
	Gerentes del proyecto	SP-AD3
	Recursos Humanos	SP-AD4

### **3.1.3 ANÁLISIS COMPARATIVO**

El análisis por separado del software libre y privado, así como la identificación de los principales grupos de stakeholders que participan e influyen directamente al desarrollo de un proyecto nos ayuda a discriminar aquellos stakeholders cuya relevancia y participación en el desarrollo del software presenta un nivel de interés inferior o bien no participan directamente en el desarrollo de los proyectos, diferenciar características como por ejemplo, códigos de comportamiento, acciones, intereses, entre otros y que en su conjunto pueden ser parte del beneficio de un proyecto de software bien diseñado.

Se procede a realizar un análisis comparativo de cada tipo de desarrollo (libre y privado) para obtener y dimensionar los beneficios en el uso de buenas prácticas las cuales ayuden a explicar las diferentes acciones de los grupos de interés.

#### **1. COMPONENTE JURÍDICO**

El componente jurídico reflejado en cada tipo de desarrollo sirve para nombrar qué tan accesible y sobre todo permisible en el uso y explotación puede ser dicho desarrollo. El software libre presenta libertad de expresión por lo que los usuarios que desarrollen herramientas libres pueden tener diversos derechos sin incurrir en usos indebidos. Por otro lado el software privado no tiene dicha ventaja ya que al presentar una licencia de carácter privado, el usuario simplemente es limitado al uso y adquisición del producto terminado.

#### **2. CÓDIGO**

El componente de código de los diferentes desarrollos describe la alteración de mecanismos que hace que una herramienta o desarrollo de software funcione tal y para lo que fue diseñada. En el caso del software libre existen actividades que el usuario tiene derecho a realizar, así como algunas que no puede realizar efecto de licenciamientos privados.

#### **3. LICENCIAS**

Las licencias forman parte de todo software ya que por ser un producto de propietario, su explotación y estudio forma parte de las actividades que cualquier usuario puede realizar. Para el caso de las licencias en software libre podemos encontrar dos principales: Copyleft y GPL. En el caso del software privado existen las licencias de tipo comercial las cuales en su mayoría son características de todo tipo de desarrollo que se realiza en instituciones de carácter privado.

#### **4. ESTRATEGIAS**

Las estrategias que se utilizan en los desarrollos de software indican la manera en que son desarrollados los productos en sí. Para el caso de los productos informáticos o de TI, las herramientas con que son desarrollados se relacionan con el tipo de licenciamiento, ya que pueden ser herramientas informáticas libres o privadas de igual manera. Ejemplo, ambientes de desarrollo de software, diseño de bases de datos, etc.

## 5. MODELO DE NEGOCIO

Hablar del modelo de negocio en los desarrollos de software es adentrarse en la manera en que se obtienen las ganancias por los diseños e implementaciones de los mismos. Existen principalmente dos orientaciones que enmarcan cada tipo de desarrollo: Servicios personalizables para los usuarios o productos orientados a una gran cantidad de ellos.

## 6. MONOPOLIO

Las prácticas monopólicas, en el caso del desarrollo de software, sirven para cerrar los mercados a los competidores potenciales que deseen desarrollar productos de software similares a los servicios prestados por los principales productores. El caso del software privado, existe un mayor uso de dichas prácticas como por ejemplo el hecho de licenciar sus productos para que de tal manera el usuario adquiera el producto de manera legal.

En el caso del desarrollo libre, existen los monopolios de producto los cuales son reflejados principalmente en los sistemas operativos (software que controla los recursos básicos de una computadora) y para lo cual son utilizados para la creación y explotación de cualquier tipo de herramienta de software. Esto no implica que exista un monopolio de empresa al ser, por su filosofía de desarrollo, promotor de desarrollo independiente de software.

**Tabla 20** Análisis comparativo entre tipos de software

ANÁLISIS COMPARATIVO		
FACTORES DE COMPARACIÓN	TIPO DE DESARROLLO DE SOFTWARE	
	LIBRE	PRIVADO
COMONENTE JURÍDICO	Sin licenciamiento de soluciones	Licenciamiento de soluciones
CÓDIGO	Ejecución, copia, distribución, modificación y estudio libre	Ejecución, copia, distribución, modificación privada.
LICENCIAS	Tipo Copyleft y GPL: Regulan la redistribución, creación y copia.	Software comercial: Software desarrollado por la empresa con fines de lucro
ESTRATEGIAS	Empleo de tecnologías libres	Empleo de tecnologías propietarias

MONOPOLIO	<p>Monopolio de producto no implica monopolio de empresa.</p> <p>Monopolio implica licencias con ventajas para el software libre. Sana competencia entre las empresas.</p> <p>Usuarios retoman el control.</p>	<p>Monopolio de empresa y de producto. Licencia de producto privado implica monopolio producto lo que implica monopolio de empresa</p>
MODEL DE NEGOCIO	<p>Servicios personalizables para el usuario.</p> <p>Participación de los stakeholders. Valor más importante al usuario y a su opinión. Rentabiliza el conocimiento.</p> <p>Logra excelencia rápidamente. Fácil depuración de errores</p>	<p>Software comoproducto (grandes grupos de usuarios). Integra determinados nichos de negocio.</p> <p>Producción en masa. Monopoliza el mercado.</p> <p>Preinstalación del software</p>

De la comparación anterior entre los dos tipos principales de desarrollo en software, aparte de identificar las características principales de cada uno, sus beneficios, ventajas y desventajas se pueden obtener los siguientes resultados orientados al software privado:

1. Los proyectos de desarrollo privado deben constituirse de principios libres, como por ejemplo que sean orientados al usuario final y a su satisfacción.
2. El software libre brinda la oportunidad al usuario de crear, alterar, promover y estudiar su contenido, lo que puede orientar a acciones y prácticas del desarrollo privado a incluir la participación activa de los usuarios en el desarrollo en cuestión.
3. Las licencias que se utilizan en el desarrollo de software privado es una estrategia de quien es el propietario del producto para generar valor empresarial por lo que también es parte del modelo de negocio.
4. En empresas desarrolladoras de software privado se hace empleo de herramientas de software libre ya que se estima que al menos un 80% del sector de desarrollo de software emplea algún producto de software libre (Driver 2010) y que su contribución a la economía en la Unión Europea es de 450 mil millones de euros anuales por la reutilización de código además de la reducción de la tasa de fracaso de los proyectos de software.

5. El modelo de negocio utilizado en ambos tipos de desarrollo ayuda a entender la naturaleza de las organizaciones TI y al público al cual van dirigidos sus productos.

6. Tanto en el software libre como el privado se manejan diferentes tipos de monopolio lo que marca su identidad en el uso de las herramientas de software, sin embargo, el software libre al ser orientado al usuario final, no existe control monopólico ni empresarial lo que hace que exista un manejo independiente entre producto y empresa. No así en el caso del software privado el cual es orientado a crear monopolios de empresa y producto.

7. El enfoque utilizado por el desarrollo de software libre está orientado a cumplir con las buenas prácticas de la responsabilidad social corporativa (RSC) al enfocar los desarrollos exclusivamente al beneficio del usuario final y no a un grupo de ellos y a un uso responsable de los recursos informáticos. Brinda también la posibilidad de enfocar esfuerzos compartidos dentro de la comunidad de desarrollo de software libre. Por lo que el enfoque a la responsabilidad social aportaría beneficios al desarrollo de software privado para orientar esfuerzos a sus stakeholders y así brindar beneficios a los clientes.

### **3.2 DIMENSIONES DE ANÁLISIS: INTERÉS, IDENTIDAD, MEMBRESÍA, RELACIÓN, (COORDINACIÓN E INTEGRACIÓN)**

Una de las actividades importantes para el diseño de las estructuras de stakeholders que puedan potenciar el éxito de proyectos de software es identificar las variables que afectan en algún nivel las relaciones entre los stakeholders y las cuales pueden explicar la razón por la cual dichos grupos actúen directamente en los proyectos de software.

Para la identificación de las variables se recurrió a realizar una investigación por medio de los factores y criterios para la creación de una **estructura organizacional** mencionados anteriormente la cual evidencia las actividades que tienen que ver directamente con los stakeholders en los proyectos de software, centralizando los esfuerzos en estudiar las razones por las cuales los grupos de interés actúan, se organizan, cumplen ciertas funciones y tratan de satisfacer sus necesidades buscando la unidad y la mejora de los proyectos de software. Se dice que los stakeholders se pueden organizar en grupos los cuales tienen distintos enfoques y pueden llevar a cabo distintas acciones que son motivadas, ya sea por un interés propio o enfocado a la organización.

La razón de comenzar el estudio desde los **intereses de los stakeholders** es debido a que por definición propia, los grupos de interés o stakeholders siempre van a realizar acciones siguiendo precisamente sus intereses. Los intereses no siempre se convierten en acciones lo que produce que se ejecute una acción puede ser también el deseo de expresar una **identidad** así como de **nombramientos o membresías** entre dichos grupos de interés que pueden afectar las acciones de dichos grupos (Rowley, Moldoveanu, 2003).

Las acciones de los stakeholders pueden ser debido a dos cosas principalmente:

- Deseo de expresar su IDENTIDAD
- Deseo de proteger su INTERÉS

Ante estas dos aseveraciones las cuales pretenden ser dirigidas hacia las empresas de desarrollo de software surgen dos cuestionamientos. ¿En qué momento los stakeholders que participan en los proyectos actúan? Y ¿cuál es la relación que hay entre cada grupo de stakeholders?

Por su parte, la **relación** entre grupos de interés no examina la probabilidad de que los stakeholders actúen pero puede explicar entre muchas cosas el nivel histórico de los cambios que han sido llevados en las organizaciones para beneficio de sus activos y la creación de valor.

Siguiendo con la necesidad de relacionar las acciones de los grupos de interés con los proyectos de software de las organizaciones surge el siguiente cuestionamiento: ¿Cómo los stakeholders, dado que son motivados por algo (personal, empresarial, gubernamental, etc.) tratan de influenciar en la organización en la que trabajan?

Para que exista influencia sobre la organización, los grupos de stakeholders pueden pasar por niveles de urgencia resultado de la entrega de proyectos, cambio en los requerimientos del sistema hasta la corrección de errores de diseño. Por otro lado, los mismos grupos de stakeholders pueden presentar un grado de descontento al darse cuenta que sus intereses son amenazados por los intereses de otras entidades o grupos es por ello que cada individuo o grupo que busque mejorar, estará encaminado hacia la protección de los mismos por lo que el análisis de los stakeholders desde el marco de interés es fundamental. Por tal motivo se incluye la variable de relación entre grupos de stakeholders y su contribución al presente estudio para contribuir a un modelo de acciones de stakeholders para tener respuestas más certeras en cuestión a las razones que hacen a los grupos de stakeholders actúen.

La variable de **membresía** tiene como principio fundamental determinar las razones por las que los grupos de stakeholders comparten el conocimiento entre personas con roles similares o rangos distintos. Se tiene la idea de que una administración del conocimiento centrado en la colaboración y benevolencia de los stakeholders puede incrementar su rendimiento ante situaciones adversas como carencia de talento reactivo, sustitución de habilidades de rangos diferentes, crecimiento intelectual y del negocio de stakeholders, beneficio a los proyectos de software entre otros. Se considera adecuado el término de membresía debido a que se desea nombrar aquella situación que tenga beneficios personales a quien la ejecuta, brinde beneficios transmitidos a otros miembros por medio de planes o concesiones, incremente la identidad por parte del equipo de trabajo.

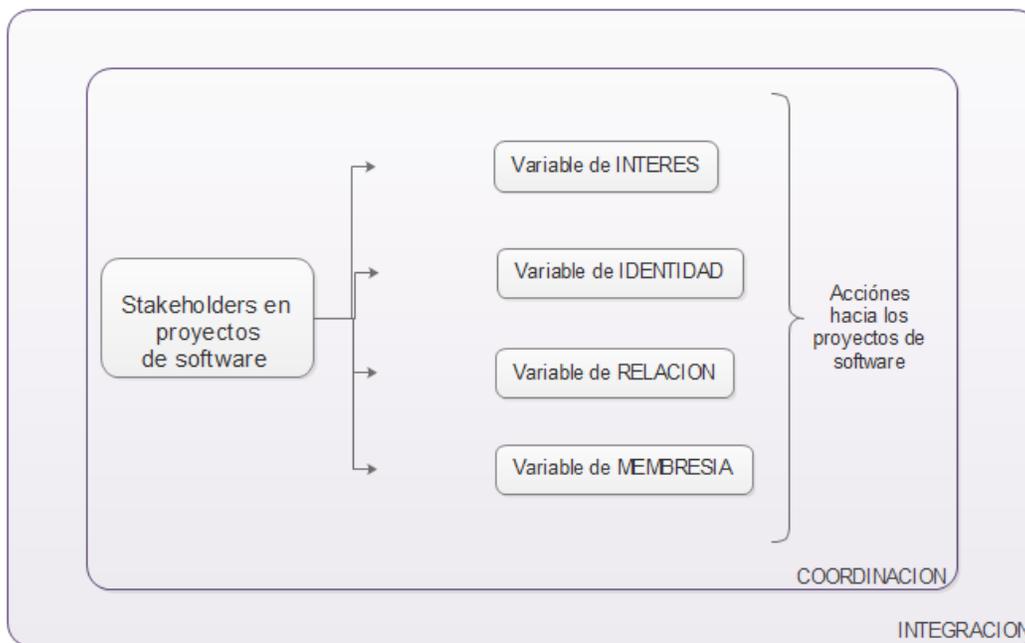
Por último se introduce la variable de **identidad** la cual surge de la necesidad de especificar las razones por las que los intereses de los distintos grupos de interés no son suficientes para explicar sus acciones ya que es importante considerar que la identidad forma parte de los individuos y también de grupos de personas las cuales buscan un lugar dentro de la sociedad y en este caso buscan ser identificados dentro del proceso de desarrollo de software.

La figura 8 representa las etapas de estudio relacionadas por las cuatro variables propuestas por el presente trabajo: Interés, identidad, relación y membresía, así como los marcos estratégicos de coordinación e integración por los cuales se basará su estudio. Dichas variables son incluidas dentro de un marco de interacción basándose en el principio holístico de **la planeación interactiva** el cual representa acciones de coordinación e integración dentro de su proceso de planeación (Ackoff, 1974). **El principio de la coordinación** establece que

ninguna parte de una organización puede planearse con eficiencia si se planea independientemente de las demás unidades del mismo nivel. Así, todas ellas deben planearse simultánea e interdependientemente. **El principio de la integración** establece que la planeación realizada independientemente en cualquier sistema no puede ser tan eficiente como la planeación llevada a cabo interdependientemente en todos los niveles (Ackoff, 1974). Además, dentro del marco de la planeación interactiva existen principios los cuales aportan valor al presente estudio, debido a que cuenta con una estructura funcional orientada a los stakeholders y por consiguiente puede beneficiar al desarrollo de la presente investigación.

Haciendo uso del **principio participativo** de la planeación interactiva se puede enfocar esfuerzos para que los individuos que participan en el proceso de inclusión a stakeholders puedan planear por sí mismos y de manera eficiente haciendo que las actividades que se realicen estén enfocadas a orientar beneficios al trabajo realizado además se considera adecuado dicho principio ya que los profesionales deben proporcionar la motivación, la información, los conocimientos, la comprensión, la prudencia y la imaginación que requieran los demás para planear eficientemente por su propia cuenta. Por otro lado se puede hacer uso del principio de continuidad ya que durante el proceso de orientación e inclusión de stakeholders, se pretende que exista un nivel de monitoreo de las actividades y que éstas tengan seguimiento por parte de las personas responsables de ejecutar dichas acciones. La planeación continua es recomendable cuando perseguimos algo que valuamos, el valor que le asignamos frecuentemente varía a medida que nos acercamos a él. Nuestros valores cambian tanto como los hechos, y sus cambios también requieren una modificación apropiada de los planes (Ackoff, 1974).

**Figura 8** Niveles de estudio de los stakeholders dentro de un proyecto de software



El modelo planteado por la figura 8 marca una pauta en el diseño de los diferentes niveles de inclusión de stakeholders, lo que de ahora en adelante se le llamarán “capas de diseño”. Siguiendo los principios de la planeación interactiva se procede a realizar el análisis de los

distintos niveles de estudio donde en cada uno de ellos se pretende determinar las características que hace que los grupos de stakeholders actúen, se organicen y ejecuten tareas con el objetivo de beneficiar los proyectos de desarrollo en software de una manera planeada.

### **3.2.1 ANÁLISIS FUNCIONAL BASADO EN EL INTERÉS DE LOS STAKEHOLDERS**

El estudio de la perspectiva basada en el interés hace una importante mención a los stakeholders de un proyecto de software, cuyo objetivo es el de identificar las razones por las cuales actúan y ejercen acciones correctivas y de mejora. Aunque la clasificación mencionada con anterioridad es importante para distinguir los tipos de actores que intervienen en dichos procesos, ésta no explica del todo la razón del por qué actúan.

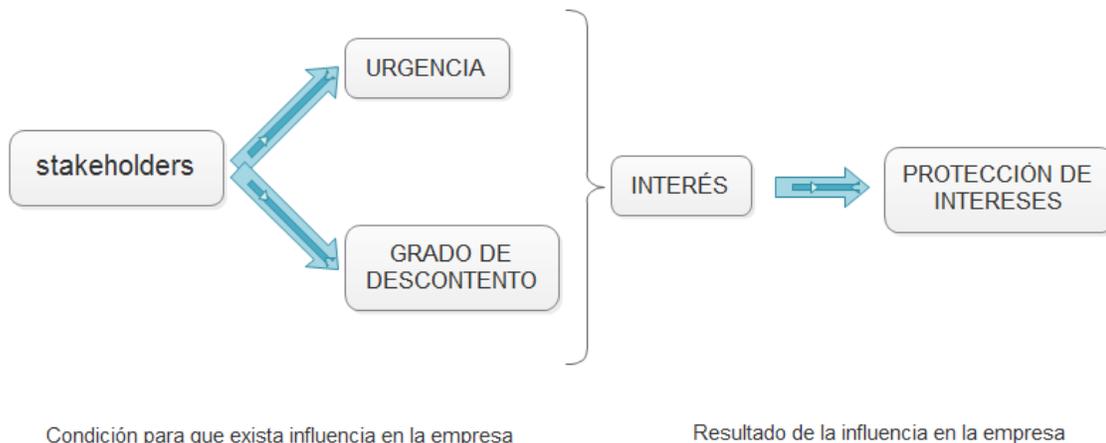
Para que el comportamiento de un individuo dentro de un grupo de interés pueda ser medido, uno de los objetivos a cumplir es el del trabajo de las entidades de alta dirección y administradores, ya que es a partir de las acciones que ejecuten estos grupo de interés en particular donde se pueden ejecutar tareas dirigidas a los grupos de interés de los proyectos. Los administradores deben delimitar cuales son las fortalezas y debilidades de los grupos de trabajo y de los recursos que la propia empresa tenga así como determinar si los stakeholders serán adversarios o cooperarán entre sí por la organización.

Cuando se analizan las acciones enfocadas en los stakeholders surgen cuestionamientos acerca de si los intereses que tiene cada grupo son denominados como urgentes o simplemente responden a un grado de descontento por parte de la situación de trabajo o por la falta de equilibrio entre lo que se quiere y lo que se tiene (Figura 9). Según autores, una urgencia no se categoriza en acción ya que es totalmente innecesario organizar y movilizar recursos para garantizar el resguardo y cumplimiento de intereses al mismo tiempo lo que puede ser problemático, por lo tanto una urgencia puede hacer que un grupo de stakeholders y su interés colectivo tomen acción hacia la empresa y en especial a los proyectos de software de interés, pero no en sí que dicho grupo tenga el deseo ni la habilidad de actuar. Por ello se concluye que el deseo de actuar y que exista algún tipo de acción en algún grupo de interés están separados por el costo de organización de dicha acción.

Es así como la tarea del administrador debe estar enfocado en dar importancia a los intereses de los stakeholders incluyendo acciones orientadas a ese fin dentro de su metodología y esto a su vez beneficie en acciones encaminadas a los distintos proyectos por lo que en base a esta tarea se propone una primera condición para que exista un grado de influencia en la empresa que desarrolla software. Para que exista una influencia en la empresa se dice que los stakeholders asociados a ella deben estar bajo un ambiente de urgencia por parte de agentes internos como pueden ser fechas de entrega, cumplimiento de objetivos, resolución de problemas inesperados etc. Por otro lado surgen condiciones laborales y organizacionales que afectan al cumplimiento de los objetivos internos de la empresa lo que puede provocar un grado de descontento entre sus stakeholders. Ambas condiciones son características de los proyectos de software y en su conjunto pueden provocar acciones reactivas, activando sus intereses los cuales pueden ser: cambio de responsabilidades, mayor capacitación, mayores prestaciones o bien dejar la empresa por lo que aquellos intereses surgidos proceso de esa influencia deben ser protegidos sea por los stakeholders mismos o aquellas entidades de alta

dirección lo que al final provoca un efecto organizacional interno para hacer frente a dichos conflictos.

**Figura 9** Condición primaria para que exista algún tipo de influencia sobre los stakeholders en la empresa de software



Los intereses también tienen medios para ser logrados como todo proceso en ejecución y son los mismos stakeholders los que actúan en dicho proceso para la obtención de un estado final en particular y que además se puede interpretar como el efecto de acción – reacción. Si la organización y en específico su política de reacción ante la satisfacción de los intereses de sus stakeholders, detiene el proceso de una obtención final de intereses, dichas acciones pueden también provocar diversas reacciones procedentes de los grupos de interés que conlleve no precisamente a acciones que favorezcan a los proyectos de software sino más bien a un cambio en las políticas empresariales. Dicho descontento puede relacionar directamente las acciones de los stakeholders, por lo que se observa que, para evitar reacciones que perjudiquen el accionar del proceso de desarrollo, los grupos necesitan tener acceso a recursos que asistan a acciones colaborativas de las organizaciones y que así sientan todo el respaldo del lugar donde trabajan y no el deseo de la insatisfacción de intereses así como de evitar tener el problema de que las personas solo buscan su beneficio individual.

Así, se concluye que la facilitación de recursos por medio de las organizaciones dirigidas a sus grupos de interés repercutirá en sus acciones, es decir, la movilización de organizaciones encaminada al beneficio de los proyectos de software empieza por la accesibilidad de recursos materiales usados para mejorar las tareas de acciones colectivas (dinero, herramientas informáticas, computadoras, etc.) así como de los recursos no materiales que motivan a la participación y facilitan la coordinación (liderazgo, consensos, responsabilidades morales etc.)

Por otro lado, se debe analizar las acciones realizadas por los distintos grupos de stakeholders respondiendo a la perspectiva de interés. Así, debido a la conformación de los distintos grupos dentro de la organización, se sabe que si sus intereses son identificados, éstos pueden estar en riesgo al no ser atendidos o por el contrario si existe una participación en conjunto durante el tiempo, ésta puede repercutir en acciones destinadas a la mejora de algún proyecto por lo que se debe pensar en las acciones que hasta el momento se han realizado para poder entender las acciones realizadas en el pasado.

Las acciones pasadas de los stakeholders reduce el costo de organización, ese es el objetivo perseguido por las organizaciones enfocadas a implementar algún tipo de planeación y para efectos de la presente investigación se recurre a relacionar los recursos de la planeación interactiva para considerar las acciones del pasado sin desviar la atención de acciones presentes y futuras por parte de los planeadores. Dichas acciones en el pasado pudieron haber sido resultado de buenas prácticas al igual que resultado del descontento de los grupos en el pasado. El costo del cual se habla es importante, ya que mide en cantidad el recurso empleado por dicha organización para ejecutar ciertas acciones y en el caso del desarrollo de software no es la excepción ya que a medida en que diversos grupos de stakeholders traten de resolver sus conflictos presentes reducirá su costo en el futuro.

Se considera de igual manera que la cantidad de relaciones pasadas es un factor clave para entender las relaciones presentes, es decir, que el número de relaciones entre otros stakeholders en la organización durante el pasado brinda la posibilidad de conocer a las organizaciones y la manera en que interactuaban sus stakeholders, los recursos comunicativos, si el producto de la comunicación masiva benefició en algún grado a los procesos dentro de la organización y si existió mayor fluidez en la comunicación efectiva para resolver los conflictos del pasado. Además, se considera que mientras el número de relaciones por stakeholders sea mayor, existe la probabilidad de que incremente el interés de dirigir esfuerzos hacia los proyectos de software con mayor rapidez al mismo tiempo que se le brinda importancia a la transmisión del conocimiento.

Por lo tanto podemos decir que los esfuerzos que se hayan hecho en el pasado son importantes para futuras acciones, así como notar que la estructura social, en este caso los usuarios del desarrollo de software, puede también influir en la producción de acciones de dichos grupos, esto ayuda a reducir por consiguiente los costos, ya que se identifican las acciones que en el pasado fueron problemáticas y pueden prevalecer en el presente. Así, mientras se identifiquen las acciones del pasado se podrán identificar también los actores responsables de dichas acciones y los vínculos que los unen. Al identificar los vínculos es claro ver que se pueden realizar distintas actividades:

1. Facilitar la difusión de normas y deseos
2. Mejorar la comunicación
3. Establecer maneras de cómo direccionar esfuerzos en insatisfacciones
4. Actuar como mecanismos de GOBIERNO (para detener oportunismo individuales y oportunistas)
5. Movilizar el descontento más rápidamente
6. Aumentar la fuerza que vincula a grupos de stakeholders y que hace que dichos grupos tomen acción.

Del análisis anterior se puede concluir que el hecho de que los grupos de stakeholders estén relacionados con acciones colectivas en el pasado genera un aumento de probabilidad en la movilización de influencias hacia la organización a comparación de los grupos de stakeholders que no han tenido acciones colaborativas en el pasado. Se concluye también que la cantidad de relaciones entre los grupos de stakeholders incrementa la probabilidad de que los grupos de stakeholders actúen para influir a la organización en mejora de los proyectos de software.

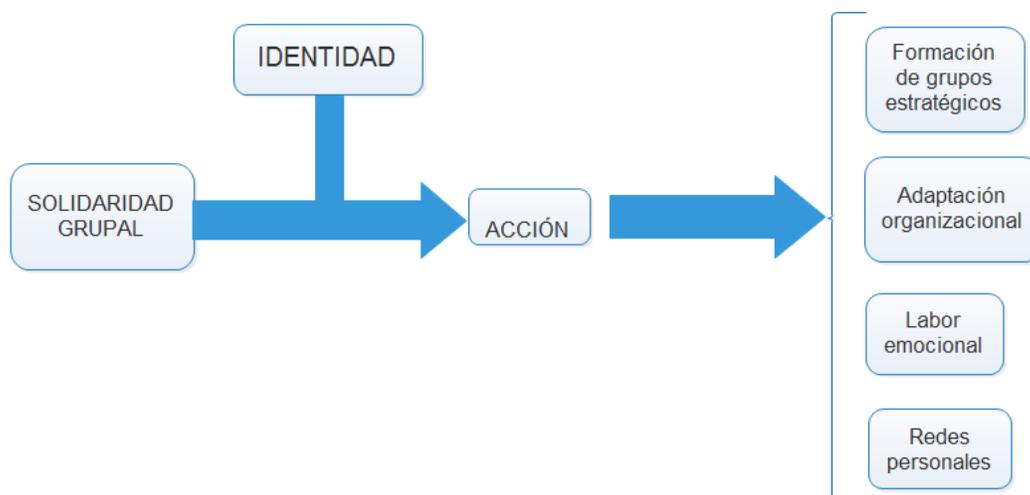
### 3.2.2 ANÁLISIS FUNCIONAL BASADO EN LA IDENTIDAD DE LOS STAKEHOLDERS

A pesar de que el análisis de stakeholders desde la perspectiva de sus intereses es importante para explicar las razones de sus acciones, no se explica de una manera más concreta su movilización en mejora de los proyectos de software dentro de la organización.

Toda organización cuenta con recursos que la hace competitiva, a pesar de ello, los participantes potenciales (grupos representativos de stakeholders) deben conocerse a sí mismos como un colectivo para que exista una unión y posibles soluciones a lo que podemos llamar “movilización del conocimiento” es decir, poner en práctica todos los recursos que existan para lograr acciones en conjunto, entre ellos el conocimiento que cada grupo de stakeholders puede aportar y en especial hacerse notar como única. La identidad colectiva define perfectamente dicha acción, la cual sirve en este caso como una base alternativa para la tan mencionada movilización, desde que se crea un compromiso individual y un sentimiento de solidaridad. Para diversos autores el sentimiento de solidaridad debe surgir de una cultura organizativa por medio de las entidades de alta dirección (EAD) es decir, que si los dirigentes desarrollan una cultura donde exista sentimiento de ayuda y benevolencia, existirá un mejor clima laboral y se establecerán diversos valores todos encaminados a maximizar esfuerzos hacia los proyectos en cuestión (Cabrera Suárez, Déniz Déniz, Santana 2014).

La identidad, al ser una variable importante a considerar y una de las dimensiones de análisis de stakeholders de proyectos de software, puede originar, entre los grupos de interés, un sentimiento de solidaridad grupal que en su conjunto converge en acciones encaminadas a la generación de valor para la empresa (Figura 10) como por ejemplo: La formación de grupos estratégicos donde se incluye ya la participación de los grupos de interés y se busca encontrar a sus representantes, la adaptación organizacional, labor emocional que brinde seguridad y comodidad a los empleados directamente relacionados a los proyectos de software para así crear redes personales.

**Figura 10** La identidad y la solidaridad originada como marco de acción de stakeholders



Los proyectos de software que se desarrollan en las empresas no son ajenos a este enfoque de identidad, ya que es a partir de los grupos de interés donde debe comenzar a surgir los cambios los cuales cualquier organización debe estar preparada de enfrentar. Existen diversos tipos de identidad, expresada por los grupos de interés, la que es importante analizar es sin duda la identidad social de los grupos de interés la cual articula el proceso en el cual un actor le da valor a un significado emocional de pertenecer a su equipo de desarrollo de software por ejemplo.

Las acciones que se originan por parte de los grupos de interés con la perspectiva de identidad y las cuales se trata de estudiar en el presente trabajo es un tópico que es importante a considerar. Como ya se mencionó, la identidad dentro de cualquier estancia organizativa y social provoca diversos beneficios como compartir creencias, la solidaridad la cual genera participación de los distintos grupos como la creación de lazos culturales.

Los proyectos de software son caracterizados por su tamaño, es decir, por el número de personas que lo conforman (Gerente del proyecto, líderes, programadores, equipo de pruebas, etc.). Es preciso indicar que cada grupo de stakeholders al tener responsabilidades diferentes, busca el reconocimiento a su trabajo y ésta es la diferencia más importante. El reconocimiento único, según autores, motiva a una mayor movilización ya que se establece de manera prioritaria que se diferencien los esfuerzos para ser evaluados. Pero qué sucede cuando se busca una misma identidad es decir, que se pretenda que todos los grupos de stakeholders tengan una misma dirección en cuanto a su identidad.

Surge ahora la necesidad de conocer si dicha propuesta (identidad única o traslapada) provoca un beneficio en el desarrollo de proyectos de software y si se beneficia la relación que existe entre cada grupo de interés. Se considera la premisa de que al igual que el interés, la identidad deba ser traslapada pero no se conoce el impacto de considerar a cada grupo de stakeholder con una misma identidad.

Si se considera que cada grupo de stakeholders (GS) presenta una identidad lo cual hace diferenciarse de otros grupos dentro de la misma organización. Existe la posibilidad de que dicha diferencia cree valor en las acciones marcadas por dichos grupos. Se considera entonces que cada grupo se identificará con aquellos miembros cuyos objetivos, pensamientos y acciones sean parecidos pero que además estén orientados hacia el mismo objetivo, el proyecto de software. Los grupos que se caracterizan por tener miembros que dan valor a la identidad en común y tienen sentimientos de solidaridad obtenida de la participación dentro del grupo son motivados a actuar por la misma identidad social ya mencionada, adicionalmente al interés que es generado por el éxito de dicha acción, por lo que se puede afirmar que el grupo de stakeholders no es instrumental si no expresivo, es decir, expresa la identidad del actor en cuestión. Se concluye entonces que aquellos miembros que reúnen dichas características son los mismos miembros del grupo de stakeholders. Es decir, líderes, desarrolladores, gerentes, etc. Cada uno debe tener una identidad única que los diferencia de otros grupos lo que crea mayor valor a sus acciones.

Si los miembros de un mismo grupo de stakeholders le dan valor a la identidad que lo caracteriza, es probable que cualquier miembro dentro del grupo movilice influencias que puedan ser benéficas a los objetivos de la organización. Y para efectos del presente trabajo, el objetivo de las empresas que desarrollan software es el de mejorar sus procesos de

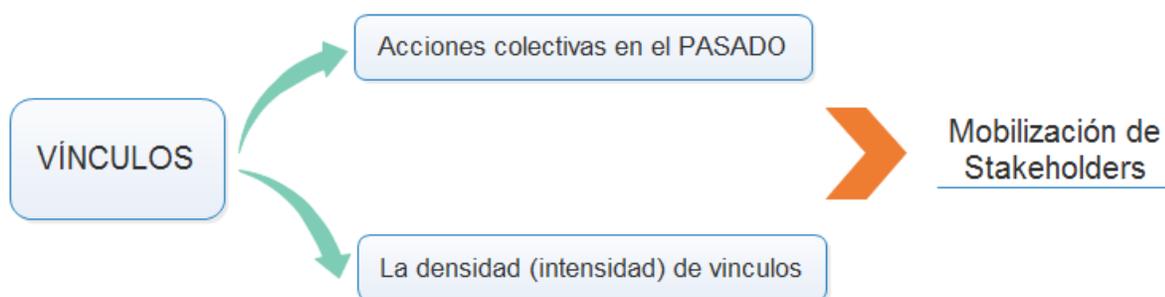
desarrollo y organizacionales ara diseñar productos creativos y útiles para los clientes por lo que se considera importante incluir la identidad única entre grupos de stakeholders dentro de éste estudio.

Así, a partir del análisis anterior se puede decir que la identidad es un medio para definir una posición social relativa la cual hace que las personas que tienen una identidad propia reflejada en el grupo de interés en relación, diferenciarse en sí misma en términos morales de personas que no tienen dicha identidad, lo que evidencia que no se es parte de dicho grupo de interés. Es importante mencionar que el propósito de crear identidad dentro de los grupos de stakeholders es para diferenciarse a sí mismo de otros (otros grupos de interés) ya que un grupo construye su identidad por las acciones que tome. El efecto de actuar de la misma manera que otro grupo de stakeholders puede ser reflejo de que dicho grupo tenga la misma identidad lo que frustra los objetivos del grupo en cuestión al reflejar insatisfacción del trabajo que realiza por lo que es importante la construcción de la identidad única de un grupo distinto del mismo desarrollo de software. Para el caso de la identidad en los grupos de stakeholders todo tipo de acciones que se realicen para mejorar el proyecto en cuestión serán motivadas principalmente por el hecho de que cada grupo tiene una misma identidad pero diferente de la de los demás.

### 3.2.3 ANÁLISIS FUNCIONAL BASADO EN LA RELACIÓN DE LOS STAKEHOLDERS

Se plantea una propuesta de movilización de stakeholders (Figura 11) de acuerdo al análisis de relaciones entre los stakeholders, retomando el principio de interés el cual relaciona la forma en que se pueden crear vínculos estratégicos entre los miembro de una organización y de un grupo de desarrollo de software específico asociando actividades del presente y del pasado de forma cuantitativa, a la formación de grupos de stakeholders y a la formación de estructuras de liderazgo dentro de los equipos de desarrollo de software.

**Figura 11** Modelo de movilización entre stakeholders basado en los vínculos a través del tiempo



La figura anterior da relevancia a los vínculos que existan entre los stakeholders mediante la asociación de acciones colectivas en el pasado y la intensidad de dichos vínculos. Se considera que el hecho de considerar los factores anteriores puede ayudar a entender y asociar las acciones características que han movilizado a sus stakeholders y que pueden movilizarlos en el presente. Por lo que la idea de recabar información de la historia individual y colectiva de los diferentes grupos de stakeholders de la organización podría ayudar a diseñar

una nueva estructura que responda a las necesidades actuales basándose en la experiencia de los miembros del equipo.

Para analizar las perspectivas de relación es imperativo considerar y comparar las relaciones que han existido en el pasado antes de ejecutar cualquier acción de mejora en el presente, es decir, cómo la empresa ha ido adaptándose a los cambios que demanda el presente sin deteriorar lo que se ha llevado a cabo durante el tiempo de vida de la organización.

Diferentes pueden ser las estructuras de relación que se han creado durante el tiempo, así, si la perspectiva es encaminada a mejorar las relaciones en el presente, se considera que en el pasado, si las relaciones eran débiles o nulas y se ejecutan acciones para mejorar dicha relación en términos cuantitativos y cualitativos a través del tiempo, los grupos en el presente pueden relacionarse mejor, puede existir más interacción entre ellos debido a las diferentes acciones en el pasado.

Los grupos de stakeholders que son relacionados con acciones colectivas en el pasado son más compatibles, además, es más probable que movilicen sus influencias hacia la organización que los grupos de stakeholders que no han tenido acciones colaborativas además si la intensidad de relaciones entre los grupos de stakeholders incrementa, es probable que dichos grupos actúen para influir a la organización.

Por otro lado, si los stakeholders tienen experiencia, aumenta la probabilidad de movilización o acciones destinadas a los proyectos en cuestión pero no se sabe el costo de sus acciones iniciales por lo que se tendrá que incluir acciones por parte del gobierno de la empresa que cuantifiquen dichos esfuerzos.

Del análisis anterior se puede aplicar la misma fórmula de traslape en términos de relación y se observa que de acuerdo con la naturaleza de los proyectos de software, donde existen stakeholders clave en el desarrollo de proyectos, el incremento de relaciones entre los miembros del equipo originará un cambio de relaciones indirectas, lo que puede ayudar a incrementar una red de relaciones capaz de utilizarse como puente en la solución de problemas dentro de la organización. Por otro lado, cuando se pretende entablar comunicación efectiva entre los stakeholders encaminada a resolver cualquier tipo de conflicto, deben establecerse parámetros de conducta, si dichos parámetros de conducta son regulados y gestionados por las entidades de alta dirección o los líderes de los proyectos existe la posibilidad de que las relaciones crezcan de manera controlada y efectiva.

### **3.2.4 ANÁLISIS FUNCIONAL BASADO EN LA MEMBRESÍA DE LOS STAKEHOLDERS**

Dentro del análisis de las distintas variables que pueden ayudar a darle efectividad a la construcción de una estructura de stakeholders en proyectos es necesario evaluar ahora el efecto que pudiera tener el enfoque a los roles que existen dentro de los equipos de trabajo y la ventaja que puede tener la transmisión de sus conocimientos entre otros miembros del equipo. Se pretende que cada rol dentro de un equipo de desarrollo cambie durante el desarrollo del proyecto (traslape de membresía) sin abandonar sus responsabilidades esenciales con fines a mejorar las relaciones unitarias que se fomentan cuando solo existe contacto entre un receptor y un remitente de un proceso interno del equipo de trabajo. Así, el líder del equipo de trabajo será pieza clave dentro de la impartición de dichos cambios dónde también debe actuar como administrador del proyecto y ayudar a las personas que deben

tomar otras responsabilidades. Por su parte los desarrolladores deben cambiar su rol por ejemplo ser parte del equipo de pruebas y viceversa. También los administradores del proyecto deben cambiar su rol a ser líderes cuando el momento lo amerite. Existen roles que no pueden ser cambiados por su nivel estratégico como el propietario del producto o los responsables de recursos humanos.

De acuerdo al análisis anterior, se considera importante determinar la intensidad de los vínculos analizados en la sección anterior para que exista un cambio en el enfoque comunicativo del equipo. Se pretende aprovechar dichos vínculos para establecer reglas de transmisión del conocimiento que pueda ser al mismo tiempo efectivo y eficaz durante el ciclo de desarrollo del proyecto. Se considera que el hecho de crear un cambio en los roles del proyecto puede ayudar a incrementar los conocimientos de cada stakeholders, conocer mejor el negocio e incrementar las habilidades para futuros cambios de rol y profesionales. El nombre propuesto para definir el cambio de roles, transmisión de conocimientos e incremento de habilidades es el de membresía al considerar que se puede tener un beneficio personal, único y transferible.

**Figura 12** Variable de membresía como transmisión de conocimiento, experiencias y roles entre grupos de stakeholders (membresía traslapada)



La variable de membresía considera que el conocimiento de cada grupo de stakeholders producto de experiencias vividas a lo largo del tiempo pueda ser transmitido de manera eficiente beneficiando a los integrantes de los equipos y que el mismo comportamiento sea replicado en cada grupo de stakeholders que tenga similitud técnica es decir que sus actividades estén enfocadas en un mismo objetivo estratégico (Figura 12).

### 3.2.5 EFECTOS DEL TRASLAPE DE INTERÉS, IDENTIDAD, RELACIÓN Y MEMBRESÍA LA ESTRUCTURA DE STAKEHOLDERS DE PROYECTOS DE SOFTWARE

Como los grupos de stakeholders se enfocan a la organización desarrolladora de software y la espera de influenciarse el nivel de importancia de las relaciones entre dicho grupo surge interés especial.

Se dice que de las variables ya mencionadas: interés, identidad, relación y membresía pueden ser analizadas desde un nivel de traslape entre los miembros de los distintos grupos de

stakeholders. De análisis anteriores se puede decir que, por ejemplo, en el caso de la membresía se pueden ocupar diferentes roles dentro del grupo de interés, esto ayudará a maximizar el cambio de funciones dando la posibilidad de experimentar diferentes tareas a diferentes miembros del grupo. Además cabe la posibilidad de aumentar la probabilidad de influenciar a la organización, esto podría balancear los conflictos y diferencias de roles así como los conflictos personales de dichos actores. En el caso de traslape de intereses (entre distintos grupos de stakeholders) puede haber la posibilidad que existan conflictos de intereses, lo que puede afectar sus acciones, por lo tanto el grado de seguimiento de los intereses de un grupo de stakeholders por parte de la organización es el grado en que se hayan intereses similares. Si los grupos de stakeholders tienen los mismos intereses que aquellos de la organización es más probable que la persona participe en la acción.

Así, si los intereses son traslapados, quiere decir que existen intereses en común y si dicho traslape aumenta, es probable que aumente la participación o movilización de los grupos de interés en la organización. Por su parte la probabilidad de que un grupo de stakeholders dentro de un proyecto de software se movilice (actúe) para influenciar a la organización decrecerá a medida que el grado de identidad traslapada con otros grupos aumente.

Por lo tanto se concluye del interés traslapado que la probabilidad de que un grupo de stakeholders se movilice es motivada por el hecho de que se tengan intereses en común y que éste incremente a medida que pase el tiempo.

Por su parte, la variable de identidad es la única variable considerada dentro del estudio la cual no necesita ser traslapada, ya que como se ha mencionado, si la variable es influenciada en distintos grupos de interés, hay la posibilidad de que ésta sea la misma y los integrantes no se sientan identificados con el trabajo que realizan en su grupo. Por tales razones, la probabilidad de que un grupo de stakeholders se movilice para influenciar a la organización puede disminuir a medida en que el grado de identidad traslapada con otros grupos de stakeholders aumente. Por lo tanto y comparando identidad e interés se puede decir que a medida en que aumenta el interés traslapado entre grupos de stakeholders aumentará el grado de movilización de dichos grupos hacia la organización lo que puede ayudar a beneficiar el desarrollo de los proyectos de software.

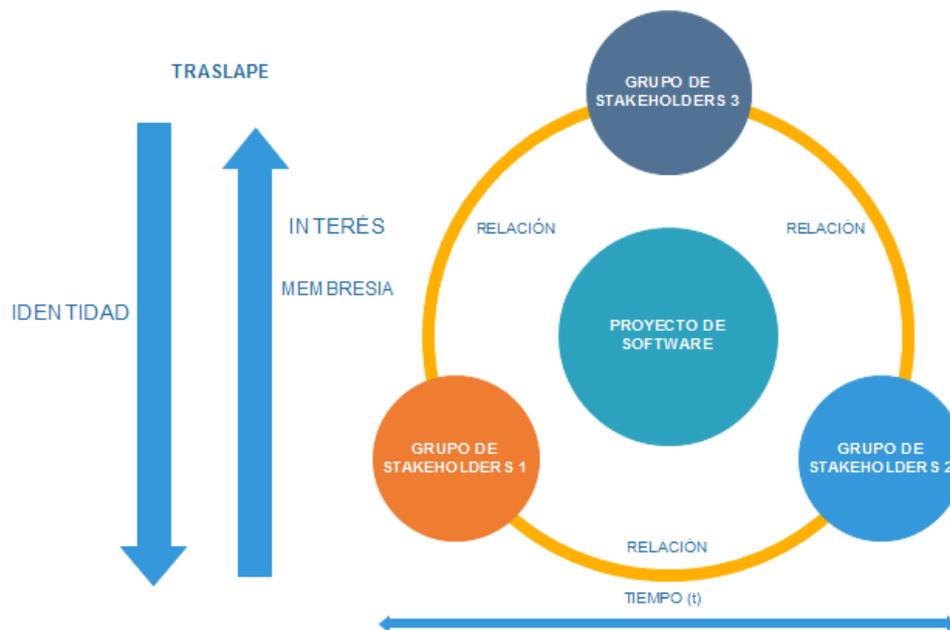
Para que se pueda mejorar la relación entre los grupos de stakeholders se analizó la membresía traslapada en términos de cambio de responsabilidades para desarrollar las habilidades de otros grupos de stakeholders. Con tal hecho, cada grupo de stakeholders experimentará desde puntos distintos, diferentes cambios entre ellos la mejora en las relaciones no solo con su mismo grupo si no con otros. En el caso de la relación traslapada, una vez habiendo analizado tanto el interés traslapado como la identidad traslapada se puede decir que la relación traslapada es obtenida una vez que los mecanismos de mejora en interés e identidad han sido ejecutados.

Por lo tanto, se puede dar paso a la formación de los grupos de stakeholders que respondan a las variables analizadas. Para la formación de estructuras de stakeholders se deben considerar factores que las fortalezcan y que se desarrollen como un marco conductor de proyectos de software como por ejemplo:

- 1- Formación de grupos de stakeholders.
- 2- Diferentes tipos de estructuras de liderazgo y su influencia inicial y subsecuente a un grupo de stakeholders.
- 3- Si el líder de un grupo de stakeholders es motivado por el hecho de expresar la identidad de un grupo, éste estará menos interesado en el costo personal asociado y relativo al beneficio de sus intereses incrementando de igual manera la probabilidad de que el grupo de stakeholders actúe.
- 4- Relacionar interés e identidad con diferentes grupos de stakeholders y sus acciones garantiza más que el hecho de realizar puntualmente las actividades, saber las razones que motivan dichas acciones.
- 5- Traslapar responsabilidades entre distintos grupos de stakeholders puede ayudar a entender la naturaleza de las funciones de cada miembro así como mejorar las relaciones entre ellos.

La conclusión obtenida del análisis anterior es que el traslape del interés, la identidad, relación y membresía entre grupos de stakeholders que participan en el desarrollo de proyectos de software genera la movilización para actuar por el beneficio de dichos proyectos y que la relación que tengan los grupos en términos de solidaridad y confianza beneficia en el presente de la organización junto con las actividades que se hayan planeado en el pasado y las cuales no se centran en darle importancia al costo si no al beneficio que éstas puedan generar en los equipo de desarrollo y por ende a la empresa en cuestión (Figura 13).

**Figura 13** Variables encontradas en el análisis de stakeholders en proyectos de software y su traslape



Por lo que se concluye que si los stakeholders tienen experiencia ganada con el tiempo, pueden tener intereses en común, una identidad propia entre grupos de stakeholders y si se le da importancia a las relaciones para que sean sólidas en el futuro, aumenta la probabilidad de

movilización pero no se sabrá el costo de dichas acciones sin que se vean reflejadas en el diseño de estructuras de stakeholders que respondan a las variables de cambio en la organización dirigidas a mejorar los proyectos de software y orientadas única y exclusivamente en los stakeholders. Es por eso que el gobierno de la empresa debe estar atento a regular éstos privilegios a medida en que se garantice la inclusión y participación de todos los individuos y por consiguiente se minimicen los costos, aumente el beneficio de los proyectos de software y se reduzcan los tiempos de su diseño y desarrollo.

### **3.3 MECANISMOS DE GESTIÓN Y ESTRUCTURAS DE STAKEHOLDERS PARA EL DESARROLLO DE PROYECTOS DE SOFTWARE**

Dentro del análisis de los temas anteriores se centra el estudio de formar estructuras de stakeholders en el desarrollo de proyectos de software que respondan a las variables de interés, identidad, relación y membresía de los grupos de stakeholders que conforman un proyecto de software las cuales actúan directamente como factores de construcción e identificación de estructuras y grupos de stakeholders en el desarrollo de software además de indicar las razones por las cuales producen la movilización de dichos grupos con beneficio de los proyectos.

Como parte de la ubicación de dichas variables se pretende determinar, en un espacio tangible, cómo actúan dichos grupos y las razones que los hacen reaccionar ante sus estímulos es decir de movilización hacia los proyectos de software. Sin embargo, las variables antes mencionadas deben ayudar a construir estructuras que produzcan un cambio en la organización en el punto de vista político y en mejora de la misma.

Se había mencionado el papel que juegan las prácticas dentro de las organizaciones las cuales influyen inicialmente y subsecuentemente a un determinado grupo de stakeholders y resultado del análisis del uso de la responsabilidad social corporativa para saber quién debe de diseñar las políticas que ayuden a cambiar la cultura organizativa de las empresas dedicadas a desarrollar software.

La estructura de stakeholders que se proponga en cualquier dimensión debe considerar a un stakeholder central llamado **stakeholder definitivo** (Tipología de los stakeholders, Cap. 2) siguiendo con el del proyecto de software y las motivaciones a las cuales debe responder son las siguientes:

- Oportunidad de expresarse
- Afiliado a la identidad (organizacional)

Dichas percepciones afirman que el stakeholder definitivo estaría, como ya se había mencionado, menos interesado del costo personal relativo al beneficio de sus intereses por lo que incrementará la posibilidad de que los stakeholders se movilicen. También, dicho stakeholder definitivo debe considerar las siguientes actividades:

1. Ideales para la planeación interactiva para disolver problemas
2. Cambiar el sistema y/o el ambiente en el cual el conflicto dentro de los equipos de desarrollo y el conjunto de problemas interrelacionados están contenidos.

### 3. Idealizar un futuro para buscar el desarrollo de la organización

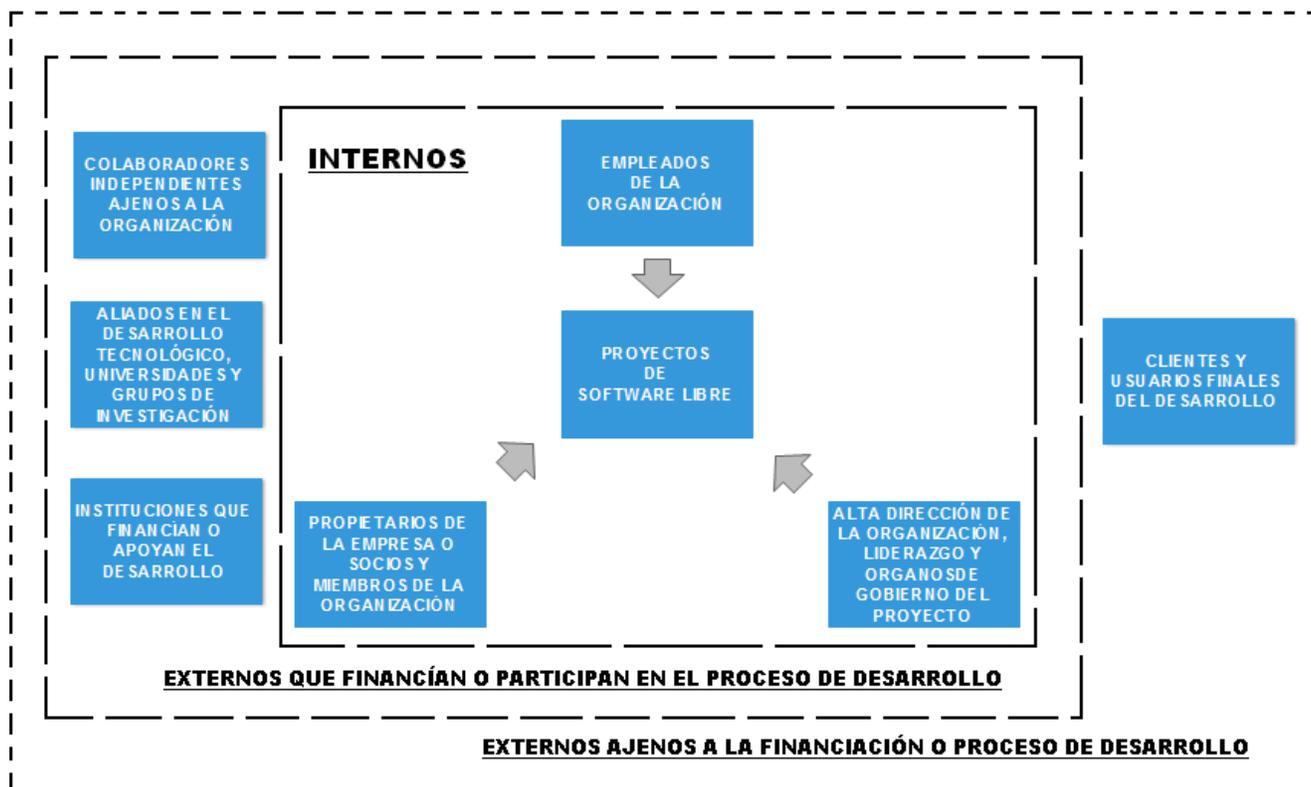
Por otro lado, el motivo por el cual se pretende explicar a los grupos de interés de proyectos de software en términos de su coordinación e integración, es debido a que una vez identificada la variable de tiempo entre el pasado y presente de las acciones destinadas a mejorar los proyectos de software se pretende construir las bases de un futuro idealizado el cual se define como el estudio de las entidades en términos de coordinación e integración (Ackoff, 1974).

Para poder construir las estructuras de stakeholders necesarias para incrementar las acciones destinadas a la mejora de los desarrollos de software es necesario identificar las estructuras que actualmente describen los dos tipos de desarrollos de software analizados en éste estudio: software libre y privado. Ambas estructuras darán un marco de diseño para la(s) estructura(s) resultante(s) que busquen añadir las distintas propiedades y variables representativas de cada uno y para después validar dicho diseño como un posible marco de construcción de estructuras de stakeholders de proyectos de software. En el caso del software libre se han identificado tanto a los stakeholders internos como los stakeholders externos divididos según la sección 3.1.2 referente a la identificación de stakeholders de software libre tomando en consideración aquellos stakeholders que tienen relación directa al desarrollo del proyecto (Figura 14).

Para tal efecto se consideran a los stakeholders internos los que tienen que ver directamente con los desarrollos. Por su parte, los stakeholders externos que financian o participan en el proceso de desarrollo (segunda clasificación de stakeholders de proyectos de software libre) toman importancia aquellos que generan componentes de código que aporta la evolución y/o soporte del proyecto de software y por tanto que benefician a su desarrollo como lo son: El rubro de colaboradores independientes ajenos a la organización, rubro de aliados en el desarrollo tecnológico, universidades y grupos de investigación y el rubro de instituciones que financian o apoyan el desarrollo. También, dentro de la clasificación de stakeholders externos ajenos a la financiación o proceso de desarrollo se encuentran quienes sin participar directamente en el proceso de desarrollo, afectan su proceso por motivos de uso del productos lo que retroalimenta su construcción, re-diseño y evolución. Así, stakeholders dentro del rubro de “clientes y usuarios finales del desarrollo” son los stakeholders importantes.

Dichos rubros, discriminando aquellos grupos de stakeholders que no impactan en el desarrollo ni la construcción del software, hacen posible la estructura de stakeholders siguiente:

**Figura 14** Modelo estructural de stakeholders de proyectos de software libre



En el caso de la estructura de software privado se contemplan dos niveles de análisis, los stakeholders internos también llamados comprometidos ya que se relacionan directamente con el desarrollo de software al tener responsabilidades que pueden repercutir directamente al desarrollo como por ejemplo, que el líder ejecute una orden hacia el equipo de desarrollo y ésta afecte al accionar del proyecto, que el equipo de desarrollo genere el código para un módulo que no fue planeado, etc.

Los stakeholders comprometidos son:

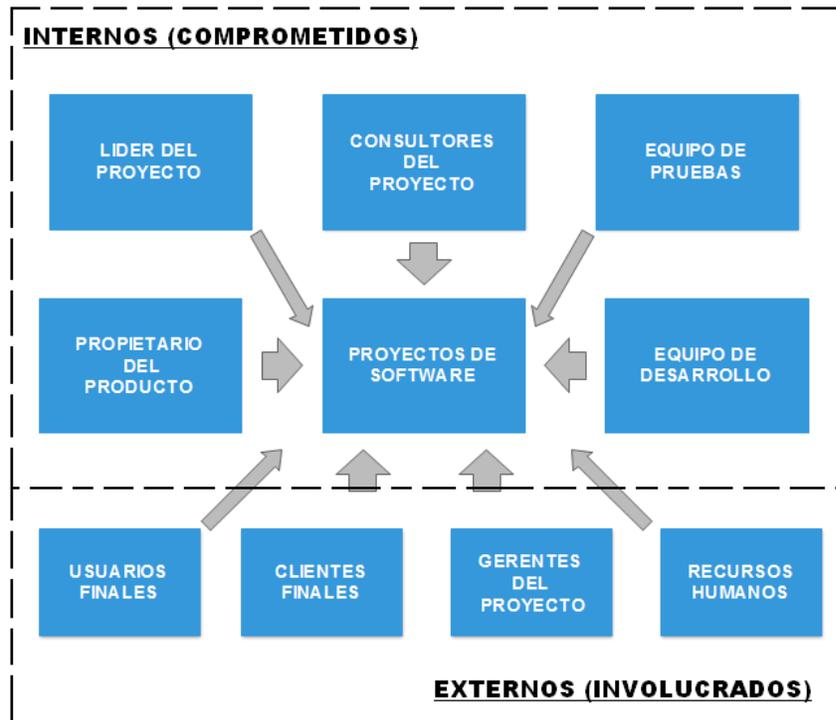
1. Propietario del producto
2. Líder del proyecto
3. Consultores del proyecto
4. Equipo de pruebas
5. Equipo de desarrollo

La segunda división es referente al grupo de stakeholders externo o involucrado el cual tiene un papel menos activo en el desarrollo del software pero que cualquiera de sus decisiones puede repercutir en el diseño y concepción del desarrollo como por ejemplo que los usuarios cambien de hábitos o preferencias y por lo tanto los clientes encargados de prestar los servicios que satisfacen dichas necesidades tengan que cambiar los requerimientos para así repercutir directamente en el desarrollo de software.

Los stakeholders involucrados son los siguientes:

1. Usuarios finales
2. Clientes finales
3. Gerentes del proyecto
4. Recursos Humanos

**Figura 15** Modelo estructural de stakeholders de proyectos de software privado



Es importante mencionar que no se presentan los mismos nombres para un mismo grupo de stakeholder en ambos tipos de desarrollo de software como por ejemplo el equipo de desarrollo, equipo de pruebas, consultores etc. son parte de un mismo grupo pero a diferencia de la clasificación de software libre solo son mencionados como empleados ya que como ya se ha mencionado, su enfoque está menos dirigido en términos organizacionales y más a la sociedad en sí. Ambas estructuras (software libre y privado) han dado un marco de referencia para identificar a los stakeholders que deben ser considerados en el desarrollo de proyectos mayoritariamente privados pero que respondan a características concebidas en proyectos de software libre como su orientación a los stakeholders, bases organizativas más colaborativas, creación de valor empresarial, responsabilidad social corporativa (RSC), gobierno de la empresa por parte de las entidades de alta dirección (EAD), etc.

Por lo tanto, con base a las características identificadas y los distintos niveles de análisis se procede a construir las estructuras de stakeholders que puedan ser un marco conductor para diseñar herramientas que puedan servir para crear una correcta orientación a stakeholders donde las entidades de alta dirección, los líderes de un proyecto de software, clientes y demás stakeholders implicados orienten sus esfuerzos en mejorar el proyecto de software.

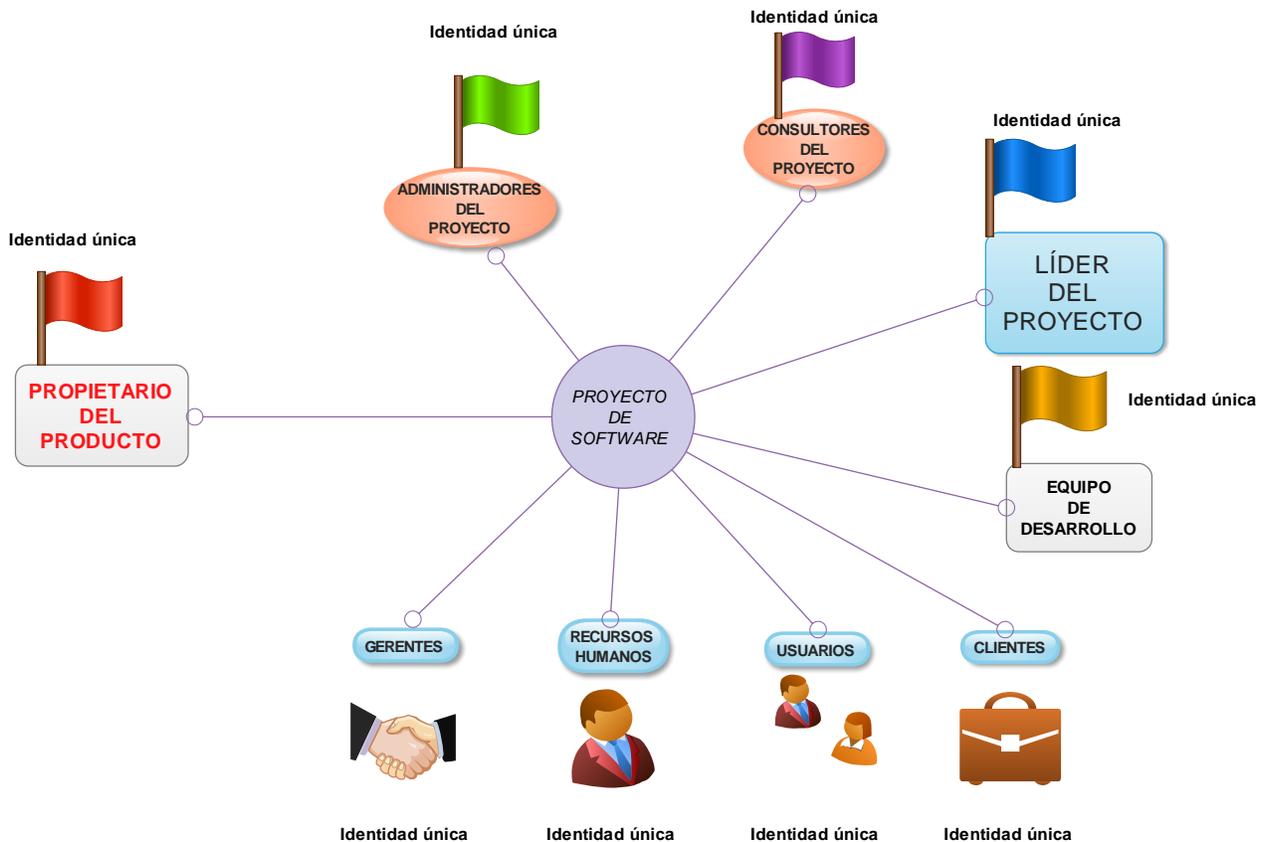
### 3.4 MODELOS ESTRUCTURALES DE STAKEHOLDERS

#### 3.4.1 MODELO ESTRUCTURAL DE STAKEHOLDERS BASADO EN EL INTERÉS

La perspectiva basada en el “interés” nos hace centrar esfuerzos en diseñar la estructura que responda a éste requerimiento. Una vez analizadas las propiedades de dicha variable, se obtienen los stakeholders necesarios para el desarrollo de proyectos privados (objetivo de éste estudio). Debido a que, como ya se ha mencionado antes, el interés entre grupos de stakeholders puede ser considerado de dos maneras: único y traslapado.

La figura 16 muestra la importancia que debe existir entre los grupos de stakeholders tanto comprometidos como involucrados ya que de acuerdo con lo analizado anteriormente, es conveniente construir una estructura de stakeholders orientado en el interés traslapado al ser éste mayoritariamente benéfico al desarrollo de software. Cada grupo debe estar consciente de sus propios intereses pero éstos no deben ser más importantes que los intereses de otros grupos. Si los intereses se comparten desde el propietario del producto hacia los clientes, existe la probabilidad de que cada grupo comparta no solo la relación existente entre ellos si no también los posibles beneficios que pueda originar el producto en cuestión.

**Figura 16** Modelo estructural de stakeholders de un proyecto de software basado en el interés



Como se puede observar en la figura anterior y como resultado del análisis de las entidades de alta dirección (EAD), es el propietario del producto (bandera roja) el que debe manifestar el interés tanto del buen desarrollo del producto de software como de la importancia que deben

tener los diferentes grupos de stakeholders que participan el dicho desarrollo manifestando así el valor de la benevolencia hacia los otros.

Una vez manifestado el interés por parte de las EAD, todos los grupos de stakeholders deben tener conocimiento del **interés compartido (interés traslapado)** como por ejemplo:

1. Satisfacción del cliente en base a requerimientos específicos
2. Construcción y diseño de software como determinada metodología
3. Cambios imprevistos
4. Tiempos de entrega
5. Diseño estructurado y controlado
6. Etc.

Luego entonces, una vez expresado el interés por los distintos grupos, el líder del proyecto (bandera verde) es el stakeholder encargado de manifestar y compartir el interés de manera proactiva y dinámica entre el equipo de desarrollo, las EAD y los clientes por lo que en base a éste análisis, puede considerarse importante el flujo del interés a partir de quien tenga la bandera verde para así ser un orquestador del interés reflejado por los demás grupos de stakeholders.

### 3.4.2 MODELO ESTRUCTURAL DE STAKEHOLDERS BASADO EN LA IDENTIDAD

La “identidad” es la variable identificada en las estructuras de stakeholders para el caso del desarrollo de software que maximiza los beneficios de la construcción y desarrollo de dichos proyectos ya que debido al análisis realizado, los distintos stakeholders deben identificar la existencia de identidad y evaluarla en términos de los siguientes cuestionamientos:

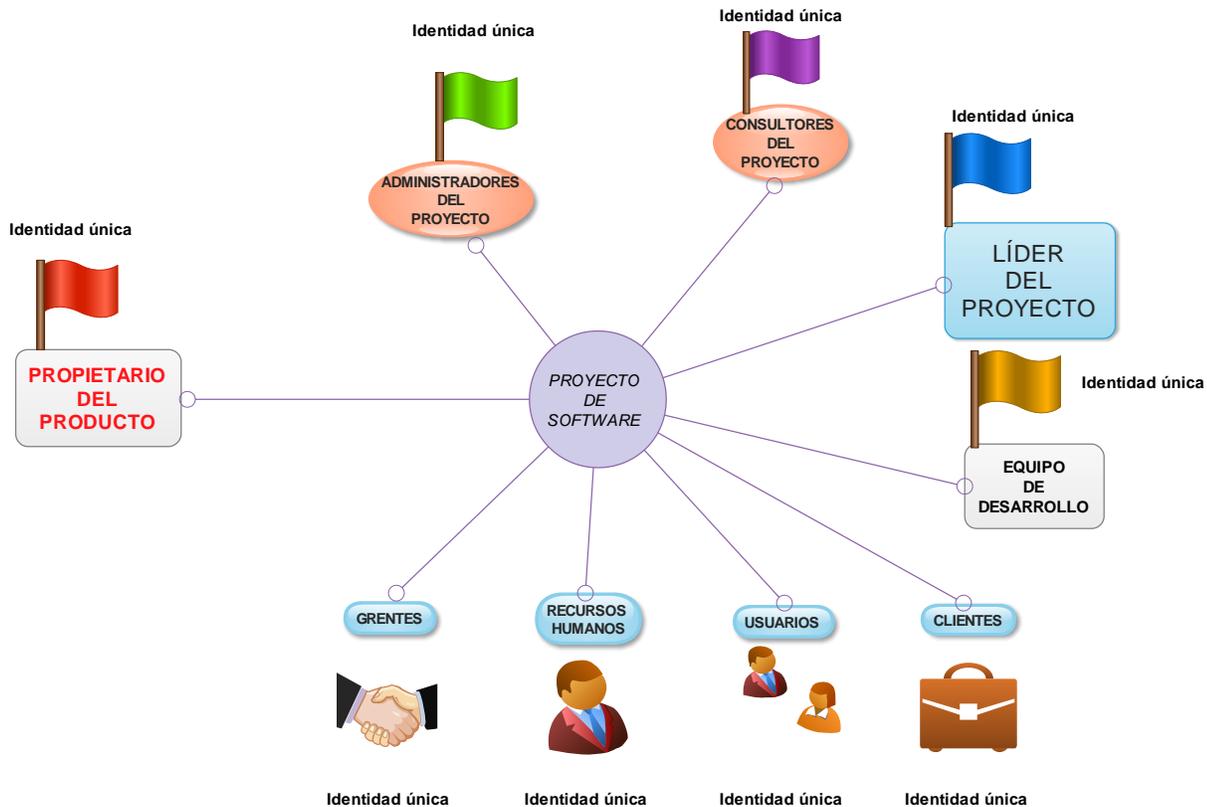
7. ¿Qué es lo que hace único al grupo de stakeholders?
8. ¿Cómo sentirse identificados con el grupo al cual pertenece determinado stakeholder?
9. ¿Es importante tener la misma identidad que otro grupo de stakeholder?
10. ¿Cómo puede ayudar la identidad dentro de un grupo de stakeholders a la construcción de un proyecto de software?

Los cuestionamientos anteriores pueden ser resueltos por medio de la figura 18 la cual muestra la estructura de stakeholders en término de su identidad y confirma el hecho de que cada grupo de stakeholders tiene una identidad única basada principalmente en:

11. Los stakeholders que los componen
12. Formas de trabajar y pensar
13. Herramientas utilizadas
14. Grados de responsabilidad
15. Formas de concebir y resolver los problemas

Lo que hace que cada miembro del grupo sienta único el trabajo que realiza y pueda así ser valorado, sea por los superiores o las EAD. Por lo tanto, la identidad no puede ser traslapada ya que si cada grupo de stakeholders tiene una **identidad única**, existe la probabilidad de que sus esfuerzos orientados al proyecto en cuestión mejoren el desarrollo de los proyectos de software.

**Figura 17** Modelo estructural de stakeholders de un proyecto de software basada en su identidad



En la figura anterior se puede notar que cada grupo de stakeholders debe tener una identidad única la cual responda a la identificación en el grupo al cual se pertenece. Cuando la identidad es única, como ya se ha analizado, es probable que dicho grupo se sienta relativamente más cómodo en el momento de realizar sus labores además entre otros beneficios:

16. Puede sentirse identificado con el grupo de trabajo
17. Incrementa las relaciones de fraternidad y compañerismo
18. Su trabajo es valorado, reconocido y remunerado
19. Orienta sus esfuerzos a la satisfacción propia sin dejar de lado la de los demás
20. Considera que sus conocimientos pueden aportar en el desarrollo del proyecto

Por lo que se concluye que la identidad en los grupos de stakeholders de un proyecto de software debe ser única y no traslapada aumentando la probabilidad de que los stakeholders actúen hacia las actividades referentes al desarrollo del proyecto de software.

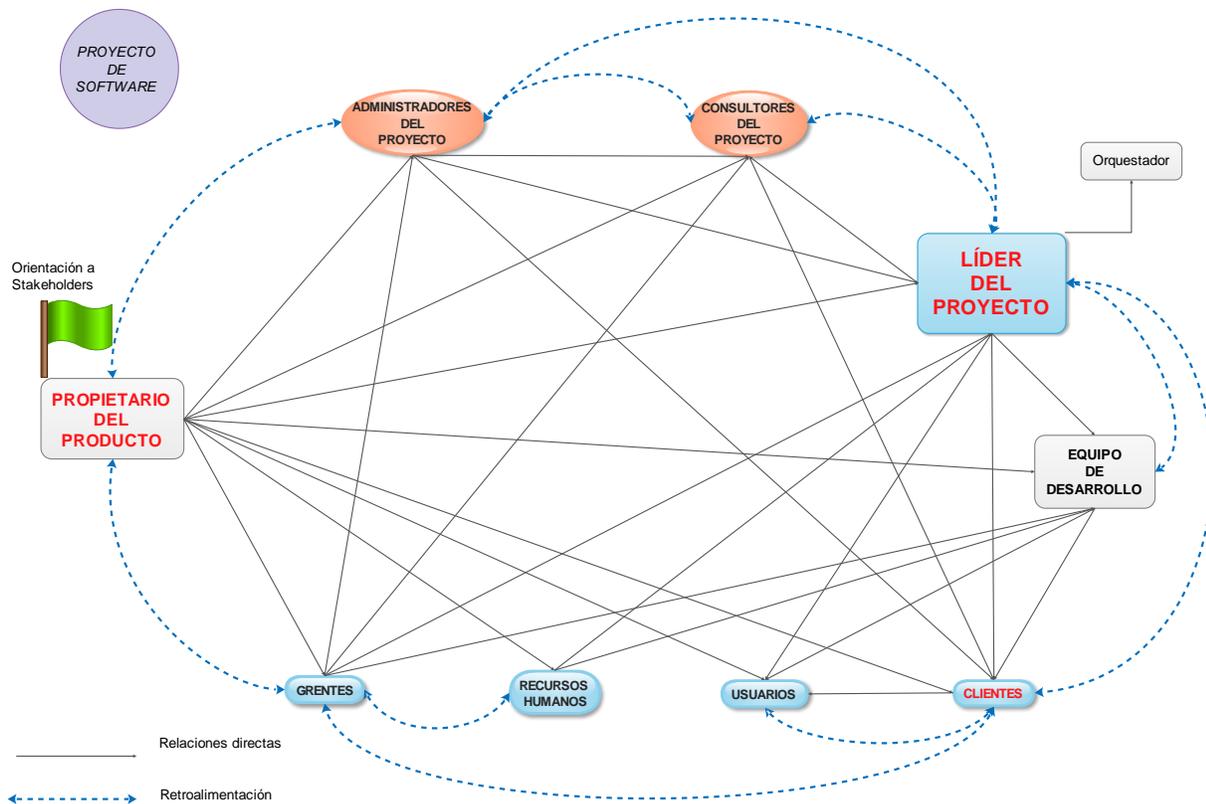
### 3.4.3 MODELO ESTRUCTURAL DE STAKEHOLDERS BASADO EN LA RELACIÓN

La variable de “relación” en el estudio de las estructuras de stakeholders de proyectos de software es fundamental una vez analizadas las variables de “interés” e “identidad” ya que debido a que se han orientado los esfuerzos de éste trabajo a generar valor en los equipos de trabajo en términos de los objetivos que se desean alcanzar y el trabajo realizado, es fundamental crear ambientes de cooperación e integración que maximicen los esfuerzos entre los diferentes grupos hacia los proyectos de software.

Para ello es fundamental marcar las relaciones directas e indirectas que puedan existir entre los grupos de stakeholders una vez siendo identificados de tal manera que se pueda diferenciar a los grupos que no necesitan interacciones entre si y dónde exista un flujo de comunicación efectiva al momento de actuar.

El traslape de intereses y la identidad única hace posible que las relaciones puedan ser traslapadas de la misma manera, es decir, que todas las actividades encaminadas a orientar esfuerzos hacia los grupos de stakeholders incluyen la orientación de mejorar su comunicación e interacción. Por ello la figura 19 enfatiza las relaciones que hay que cuidarse y las que no para así orientar los esfuerzos única y exclusivamente en el éxito de los proyectos de software.

**Figura 18** Estructura de stakeholders de un proyecto de software basada en la relación.



Se puede notar en la imagen anterior dos tipos de relaciones: directas y de retroalimentación. Las relaciones directas se caracterizan por relacionar a los grupos de stakeholders cuyos lazos deben ser fuertes y que muestran más responsabilidades entre sí. Uno de los grupos más representativos en esta clasificación es el propietario del producto ya que, según lo analizado en secciones anteriores, es responsabilidad de las entidades de alta dirección (EAD) la orientación a los stakeholders. Cuando los propietarios del producto fomentan entre los diferentes grupos valores como la comunicación, la participación, la benevolencia y la disposición de los recursos de la empresa, los grupos se considerarán además de únicos, valiosos en la organización por lo que si el propietario del software mantiene estrechos lazos de relación entre la mayoría de los grupos de stakeholders, estos a su vez fomentarán los mismos valores a los grupos que son estratégicamente claves para el desarrollo de software.

La carencia de una relación directa para con un grupo de stakeholders no quiere decir que entre ellos no deba existir relación alguna, si no que el flujo de comunicación entre ellos puede depender de otro(s) grupo(s) de stakeholder(s) sin alterar aquellos que son objetivo. Por otro lado existe las relaciones de retroalimentación, como por ejemplo entre el administrador del proyecto y el cliente la cual teniendo una relación directa entre sí, la relación de retroalimentación que va desde el cliente hacia el líder es el resultado de la gestión adecuada del proyecto como respuesta a un buen uso de mecanismos de relación y comunicación. Los clientes, por un lado pueden notar que se están realizando tareas adecuadas para darle importancia a sus opiniones y consideraciones al momento de obtener los requerimientos para el proyecto y éstos pueden responder al líder del proyecto con determinados privilegios al momento de establecer lineamientos e información relevante que pueda ayudar a agilizar el desarrollo del software.

Es importante notar que además del propietario del proyecto, el administrador del proyecto toma un papel fundamental en éste punto, al ser considerado uno de los stakeholders primordiales. La razón por la cual se considera al líder primordial es porque la naturaleza de la mayoría de proyectos de software requieren hoy en día la participación proactiva de cada uno de sus miembros y muchas veces, los proyectos de software no contarán con un administrador de proyecto capaz de resolver y agilizar las labores de comunicación efectiva entre ellos y a la vez entender la naturaleza técnica de los proyectos. Por lo que se considera que al momento de efectuar una gestión de stakeholders, sean los líderes los que tomen un papel alternativo de administradores el cual accione y cree las herramientas necesarias para interactuar con los clientes como por ejemplo:

1. Entender sus necesidades
2. Ubicar las relaciones existentes
3. Ubicar sus deficiencias
4. Entender sus intereses
5. Mejorar los lazos de comunicación
6. Otros

Otro de los stakeholders clave es el “cliente” ya que es de los grupos que presenta más interacciones con los demás lo que indica que es un stakeholder primordial para éste estudio. La orientación a los clientes puede entonces ser una estrategia que ayude a dimensionar sus intereses y que entienda sus necesidades. Cuando los clientes están satisfechos, es probable que el proyecto de software cumpla con su objetivo para el cual fue concebido y así ser

considerado como exitoso.

Así, cada grupo de stakeholder tiene una relación con otro lo que ayuda a facilitar el flujo de información entre aquellos grupos que no interactuaban con anterioridad por considerarse como no relevantes.

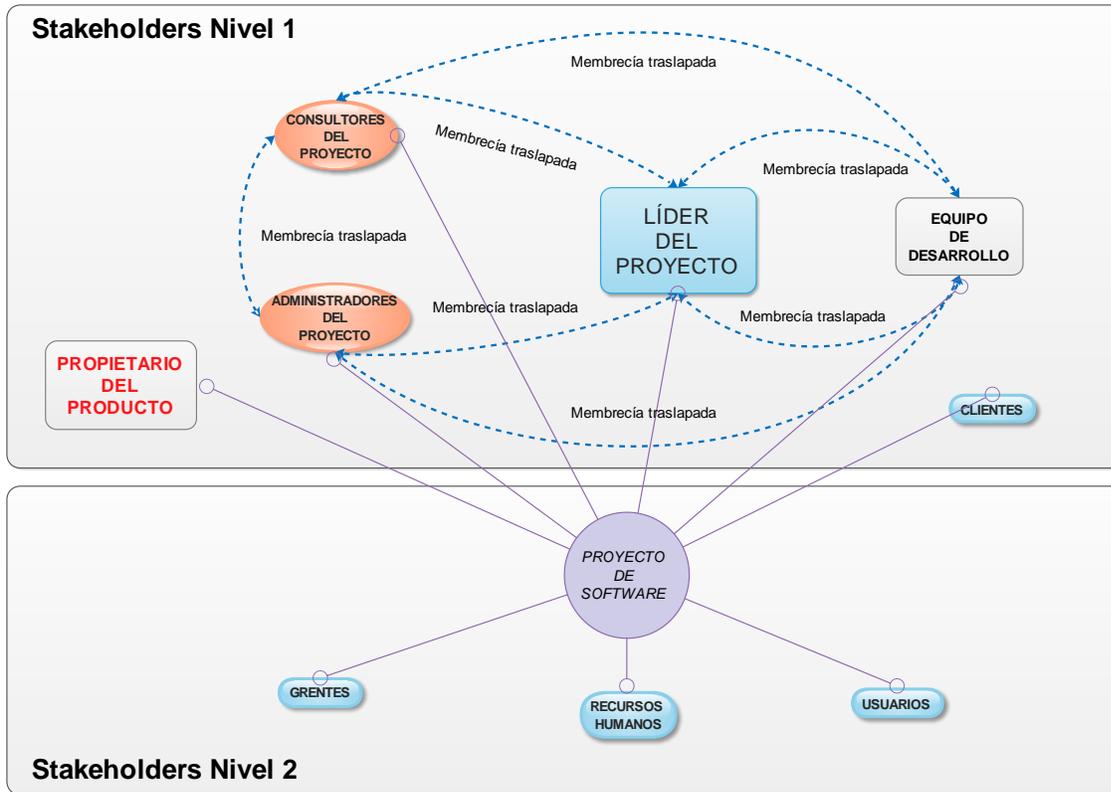
#### **3.4.4 MODELO ESTRUCTURAL DE STAKEHOLDERS BASADO EN LA MEMBRESÍA**

Hasta el momento se han analizado tres variables que maximizan las acciones de los diferentes grupos de stakeholders para el desarrollo de proyectos de software. La variable de relación une de manera integral a todos los grupos de stakeholders identificando quienes deben interactuar entre sí lo que ayuda a determinar un flujo de comunicación efectiva. La comunicación que exista entre los diferentes grupos puede ser analizada desde la perspectiva del traslape, ya que como se ha mencionado antes, el traslape puede maximizar los esfuerzos de acción de los distintos grupos para con los proyectos de software.

La membresía es entonces la característica dentro de los grupos de stakeholders que establece hegemonías y responsabilidades de sus miembros como por ejemplo en el grupo del equipo de desarrollo pueden estar: Los desarrolladores, los diseñadores de páginas web, los analistas, los diseñadores de bases de dato, etc.

Grupos como consultores del proyecto, equipo de desarrollo, clientes y usuarios son ejemplo de distintos niveles de hegemonía. Ya que dentro de los mismos podemos encontrar diferentes membrecías. La figura 20 muestra los grupos de stakeholders que presentan mayores membrecías y de los cuales el traslape entre los mismos incrementa la probabilidad de orientar acciones hacia el desarrollo de los proyectos de software.

**Figura 19** Estructura de stakeholders de un proyecto de software basado en la membresía.



A medida en que se pueden identificar las hegemonías de los distintos grupos de stakeholders podemos hacer dos clasificaciones: Stakeholders de nivel 1 y nivel 2. Ambas clasificaciones se centran en dividir aquellos stakeholders que pueden trasladar sus actividades para así agilizar el proceso de desarrollo de software desde la perspectiva de desarrollo de nuevas habilidades, comunicación entre los grupos de interés y mejora de procesos de desarrollo.

En el nivel 1 se encuentran los stakeholders potenciales los cuales pueden sea trasladar sus funciones y responsabilidades o tener una mayor prioridad dentro del proceso de desarrollo. Los stakeholders de esta clasificación son: Propietario del producto, clientes, administradores de proyecto, consultores, líderes y equipo de desarrollo. Todos excepto el propietario del producto y los clientes pueden trasladar sus responsabilidades, es decir, cambiar su membresía a medida que el proyecto lo permita como por ejemplo que un desarrollador de software pueda tener la capacidad de desarrollar actividades de un líder y viceversa o bien que un consultor pueda aprender de las actividades que desarrolla un administrador de proyecto para así fomentar además del aprendizaje que cada responsabilidad pueda darle, una mejora en la comunicación entre diferentes grupos de stakeholders. Los stakeholders de nivel 2 son: Gerentes, recursos humanos y usuarios y son denominados de tal manera porque éstos no pueden compartir sus responsabilidades entre otros grupos de stakeholders ya que tal hecho podría afectar tanto el flujo de información entre los grupos como el desarrollo mismo de software.

Las estructuras de stakeholders propuestas en el actual trabajo deben cumplir con el objetivo de estudiar las relaciones entre los grupos de stakeholders, y de tal manera diferenciar los distintos niveles de poder que ejerce tanto el propietario del producto, como la influencia y poder de decisión del líder del equipo. Una vez habiendo diferenciado los distintos niveles de poder se recurre a determinar responsables en cada nivel es decir, saber quién debe tomar las decisiones para ejercer los cambios planeados. Es en esta etapa donde el rol del consultor, como persona que analiza y propone mecanismos de control dentro de los equipos de desarrollo y las tecnologías a usar debe actuar como planeador junto con el líder del equipo para diseñar estrategias de diseño e implementación de software que agilicen su desarrollo.

Dicha inclusión de líderes, consultores y administradores como planeadores exclusivos hace una importante contribución a la orientación de stakeholders de éste trabajo ya que puede ser renuente una participación completa en el cual los grupos pueden ganar aceptación desde el principio como “consultores” y es normalmente posible incrementar su participación en el proceso durante el tiempo. La continuidad con estos cargos y con sus funciones respectivamente tendrán también un efecto positivo en la maximización del beneficio del desarrollo de proyectos de software, ya que los valores que tienen los stakeholders dentro de la organización cambiarán durante el tiempo ( cambios como puedan ocurrir) y es necesario tener en cuenta dicho proceso.

Tanto el líder, el administrador de proyecto y el consultor encargado deben hacer uso de su experiencia ahora no solo para planear para los otros si no para que los otros planeen para sí mismos por medio de mecanismos de identificación, inclusión y gestión de stakeholders. Dichas acciones deben estar alineadas con las prioridades de los grupos de interés, el seguimiento de las estructuras de stakeholders propuestas así como la aplicación de mejores prácticas citadas en el presente trabajo.

### **3.5 VALIDACIÓN DE LOS MODELOS ESTRUCTURALES**

Se tienen presentes hasta ahora cuatro dimensiones de análisis en términos de las estructuras de stakeholders para proyectos de software cuyas variables hacen notar la relevancia a la cual deben centrarse los distintos grupos de trabajo, es decir, estructuras orientadas hacia: el *INTERÉS*, la *IDENTIDAD*, la *MEMBRESÍA* y la *RELACIÓN*.

Se han analizado y construido modelos estructurales que responden a las necesidades que los desarrollos de software pueden requerir para incrementar la probabilidad de éxito de los proyectos las cuales son fundamentadas en la orientación exclusiva hacia sus stakeholders. Como resultado a los estudios llevados a cabo en la presente investigación, se tiene ahora la propuesta de diseñar una herramienta que pueda ser aplicada para el diseño de una estructura funcional de orientación a stakeholders cuyo objetivo central es el de valorar y verificar el impacto de dichas estructuras en los proyectos de desarrollo de software y así, poder proponer los distintos mecanismos de control que puedan fortalecer los factores críticos de éxito de los proyectos en cuestión haciendo uso de las cuatro capas obtenidas en el proceso del presente trabajo junto con los aspectos que pueden tener impacto en dichos proyectos. Se propone una herramienta de estructuras funcional de orientación a stakeholders y no una herramienta de diseño de estructuras de stakeholders debido a que la orientación a stakeholders es, en términos funcionales, más factible de analizar por parte de las organizaciones de desarrollo de

software y no simplemente la modificación de las estructuras actuales ya que brinda mayores beneficios a los equipos de trabajo y puede aplicarse en cualquier momento de proceso de desarrollo del software sin modificar la estructura actual.

El objetivo de dicha orientación se centra ahora en construir los medios metodológicos y secuencialmente estructurados de orientación a stakeholders que puedan ser aplicados a la par que el proceso de desarrollo de software y de igual manera ser aplicados en casos donde se creen espacios para la detección de mejoras en términos de diseño y desarrollos de software por medio de recomendaciones y actividades enfocadas a lograr una mayor organización en equipos de desarrollo encargados de recabar la información necesaria al igual que construir el software. Así, se puede ahora contrastar a las estructuras entre sí por medio de criterios de mejora para que puedan ser efectivas en dichos grupos de trabajo, los líderes de los proyectos y todos los stakeholders cuyas acciones están enfocadas a mejorar los proyectos de software y así puedan tener mayor repercusión en los clientes, las empresas que contratan los servicios de consultoría en TI y en especial para la sociedad en general.

### **3.5.1 HERRAMIENTA DE ORIENTACIÓN A STAKEHOLDERS**

A continuación se define el proceso de orientación a stakeholders en proyectos de software por medio de una herramienta funcional que realice el proceso de mejora tanto en tiempo, costo y beneficio de cualquier desarrollo de software que cumpla con características similares como lo son:

- 1.- Tiempo de desarrollo
- 2.- Tamaño de recursos humanos en el equipo de desarrollo.
- 3.- Tamaño de recursos directivos y estratégicos
- 4.- Recursos tecnológicos
- 5.- Nivel de complejidad

La herramienta de orientación a stakeholders en proyectos de software, **MIIR** (por sus características funcionales y específicas: Membrecía, Interés, Identidad y Relación) muestra cinco fases que pueden ser evaluadas por los siguientes criterios de gestión: **Soluciona la identificación de stakeholder, establecimiento de los roles de los stakeholder, análisis de los requisitos de stakeholders, establecimiento del alcance del proyecto de software en términos de los stakeholders, análisis de las características de los stakeholders.**

Dichos criterios han sido determinados por medio de la literatura de la cual éste trabajo se sustenta así como de las distintas herramientas que actualmente toman en consideración al análisis de stakeholders dentro de sus procesos de gestión para las mejoras de los proyectos de software. La MIIR contempla las siguientes etapas:

#### **3.5.1.1 IDENTIFICACIÓN Y ANÁLISIS DE STAKEHOLDERS**

Para esta primera etapa se considera dos actividades importantes: la identificación y la priorización de stakeholders las cuales contemplan en primera instancia una hoja de información relevante o lo que también puede ser los parámetros de entrada de algún sistema

informático de orientación a stakeholders el cual que debe ser llenada por el líder del proyecto o también por el administrador. El objetivo de contar con la hoja de información de stakeholders es que la persona encargada pueda priorizar actividades referentes a los intereses de los stakeholders, establecer roles y funciones de acuerdo al perfil de los stakeholders, así como información que pueda ser de utilidad.

La hoja de gestión de stakeholders deberá contener requerimientos mínimos como lo son: Información personal del stakeholder (Nombre, apodo, mail y teléfono), Información de empresa del stakeholder (División o área, puesto, antigüedad), información del proyecto del stakeholder (Rol, tipo, interés, lenguaje técnico). Es importante indicar que el rol del stakeholder puede estar definido de acuerdo a las tareas realizadas como lo son si es propietario, si es un referente funcional, usuario técnico, usuario principal. Además de todo lo anterior, el tipo de stakeholder se diferenciará de acuerdo al nivel de responsabilidad ya sea si es interesado o comprometido. Existe información adicional que puede incluirse como lo es la influencia sobre, la influencia por, en caso de ausencia, conflictos con, problemas recurrentes, intereses y comentarios (*Ver anexo A*).

La información anterior podría proporcionar al responsable de administrar los stakeholders una herramienta, fácil de controlar pero sobre todo permitiría listar a los stakeholders en su totalidad, controlar la información personal y clave para la empresa, identificar y ubicar conflictos e intereses, identificar grupos de poder, identificar conflictos y problemas y establecer comentarios para ser analizados por parte de los grupos de gestión. Una vez que se ha llevado a cabo el proceso funcional del levantamiento de requisitos en el proyecto de software (actividad técnica del proceso de desarrollo de software mencionada en capítulo 2), se pueden analizar los requisitos técnicos y funcionales que deberían tener los stakeholders por medio de la información ya recabada. El objetivo de llevar esta administración se basa en que los administradores o líderes de proyectos puedan administrar a sus stakeholders en base a los roles que deberían tomar las personas involucradas con los proyectos e iniciar un proceso controlado de orientación a stakeholders.

### **3.5.1.2 PRIORIZACIÓN**

La **priorización** de stakeholders es una tarea que, al igual que la identificación deberá ser llevada por aquellos stakeholders cuyo poder e influencia sean mayores. Para tal aseveración se analiza por medio de un mapa de stakeholders el nivel de relaciones recíprocas haciendo que se visualicen niveles de manera descendente de carácter global a uno local o ascendente del local al global.

Las empresas desarrolladoras de software deberán actuar de manera sistémica para este estudio, es decir que deberán realizar la categorización de stakeholders de manera global, estableciendo jerarquías de: función de grupo, las expectativas, las herramientas de dialogo, etc.

Para establecer una priorización que permitiera enfocar esfuerzos destinados a ciertos grupos de stakeholders se recurre a determinar cuáles son los stakeholders que deben tener mayor priorización para este estudio. Los indicadores utilizados en la sección de identificación de stakeholders dentro del software privado (objetivo de éste trabajo) nos ayudan a ubicar el nivel

de prioridad dentro de cuatro cuadrantes el cual será de ayuda para enfocar los esfuerzos que requieran mejoras tanto en priorización como en acción (*Ver anexo C*).

Siempre se puede aplicar la herramienta cada vez que el encargado de administrar a los stakeholders comience un proyecto o en una fase temprana del mismo para de ésta manera identificar y relacionar grupos de stakeholders que no se hayan considerado así como aquellos que se incluyan en fases posteriores. Una vez habiendo identificado a los grupos de stakeholders se requiere de un segundo nivel de contrastación que el líder de proyecto deberá realizar una identificación de stakeholders clave orientada a mejorar las relaciones de comunicación e inclusión de los mismos y que esté siempre orientada a mejorar cualquier proyecto de software.

Para identificar aquellos grupos de stakeholders finales que serán exclusivamente especiales al desarrollo de éste trabajo, se propone agregar una variable extra al proceso de priorización de stakeholders ya que, debido a la naturaleza de los proyectos de software, se presentan escenarios turbulentos en los cuales la urgencia que implica terminar los proyectos en tiempo y forma establece características únicas de éste tipo de proyectos. Es por eso que se recurre a utilizar una herramienta cuyo nombre se debe al matemático y filósofo John Venn (1881) conocida como Círculos de Venn y adaptada a la identificación de stakeholders por Guerras y Navas (2013) el cual ayudará a identificar aquellos stakeholders que son más relevantes en el caso de los proyectos de software así como tener una herramienta de contraste no solo de los grupos cuya urgencia es más importante si no que se clasifican por sus niveles de poder, legitimidad y urgencia (*Ver anexo D*).

Por medio del uso de los círculos de Venn mostrados en el anexo D, se puede llegar a encontrar los stakeholders que son clave en fases tempranas del proyecto y al igual que la priorización anterior, el responsable de administrar a los stakeholders puede detallar de una manera más precisa por medio de la variable de URGENCIA a los stakeholders asociados al proyecto.

### **3.5.1.3 GESTIÓN DE LAS RELACIONES DE LOS STAKEHOLDERS**

Para poder evidenciar el tipo de relaciones entre los stakeholders antes mencionados, se recurre a realizar una herramienta la cual permita visualizar, tanto a los stakeholders con mayor prioridad, los que necesitan estar satisfechos, los que necesitan estar informados así como los que se necesitan solamente monitorear resultante del análisis de priorización de stakeholders. Dentro de la propuesta de herramienta de gestión de relaciones se podrán establecer parámetros que ayuden a diferenciar el tipo de relaciones que se tienen entre los stakeholders según su clasificación de prioridad. También se podría enmarcar las cuestiones recabadas por el análisis principal y considerarlas para establecer acciones de mejora según el tipo de relación para una vez teniendo en cuenta la información anterior se puedan llevar el seguimiento adecuado según el tipo de relación.

El uso de la herramienta podría ayudar a diferenciar y clasificar las relaciones de los stakeholders para efectuar mejoras en sus actividades. Se puede tener la propuesta de clasificación de stakeholders por medio de los resultados obtenidos de los cuadrantes de priorización vistos anteriormente los cuales se pueden resumir a niveles de clasificación A, B,

C y D según sea el caso. Es importante mencionar que la información que el administrador de stakeholders ingrese a ésta herramienta puede variar, dependiendo del resultado de los estudios anteriores y de los recursos informáticos y de comunicación con que se cuente (*Ver anexo B*).

#### **3.5.1.4 ORGANIZACIÓN E INTEGRACIÓN DE STAKEHOLDERS POR MEDIO DE MODELOS ESTRUCTURALES DE INTERÉS, IDENTIDAD, RELACIÓN Y MEMBRESÍA.**

Una vez realizado el análisis anterior se procede a considerar la estructura de stakeholders diseñada en secciones anteriores siguiendo el tipo y clasificación de relaciones, donde se pueden ubicar aquellos stakeholders que puedan accionar un cambio en las actividades realizadas por los stakeholders. Las estructuras diseñadas pueden organizar a los stakeholders siguiendo las cuatro bases de diseño propuestas: interés, identidad, relación y membresía. Es importante mencionar que el responsable de llevar a cabo la integración y organización de stakeholders pueda visualizar a su equipo de trabajo como un solo conjunto donde se asegure de seguir las relaciones marcadas por las estructuras propuestas. Si bien, el resultado de identificación de stakeholders no será siempre el mismo, debido a la diversidad de stakeholders de los proyectos, si puede orientar esfuerzos a construir una base que cumpla con los requerimientos señalados por dichas estructuras y así pueda incrementar la posibilidad de orientar esfuerzos hacia los proyectos de software (*Ver anexo E*).

Las estructuras propuestas señalan nuevas maneras de organización entre los equipos desarrolladores de software como una propuesta adicional a la integración y organización tradicional de equipos de trabajo en empresas desarrolladoras de software, además que brindan la posibilidad de orientar esfuerzos para mejorar la relación de las cuatro variables propuestas: membresía, interés, identidad y relación. Por otro lado es importante mencionar que se señala a los posibles stakeholders responsables de cada estructura por motivos a dedicar esfuerzos en quienes tienen mayor poder y mayor influencia y por consiguiente mayores posibilidades de incrementar un cambio organizacional orientado a los stakeholders. El nivel de la variable relacionada significa que cada estructura debe ser considerada siguiendo dos principios: Unicidad y Traslape.

La unicidad, como se ha mencionado en secciones anteriores relaciona conceptos de exclusividad que deben cumplir de manera individual cada uno de los stakeholders., es decir que cada acción que se planea, orientada a la estructura, debe ser considerada en ser aplicada por los miembros de manera individual para establecer efectos referentes a la variable asociada. La variable de Traslape indica que las acciones que se deban llevar a cabo deben ser compartidas entre los diferentes stakeholders señalados.

Otro de los objetivos perseguidos en la aplicación de un nuevo modelo de organización es el de reforzar las relaciones existentes entre los stakeholders, ya que se podría identificar al inicio o durante el proyecto las bases para que los stakeholders se relacionen con fines a maximizar su identidad entre el equipo de trabajo y mantener una mejor comunicación.

### 3.5.1.5 ANÁLISIS DE STAKEHOLDERS

Para cada estructura es necesario remarcar aquellas características que puedan ser de utilidad hacia los equipos de desarrollo, no solo por la necesidad de tomarlas en cuenta, si no por crear lineamientos que permitan cambiar las prácticas y actividades llevadas a cabo y fomentadas por parte de la empresa hacia sus trabajadores por tanto es importante definir y crear dichas herramientas que den importancia a las actividades principales. Es por ello que para cada estructura y variable anteriormente definidas se obtienen diferentes criterios que remarcan las actividades principales que deben hacer que los equipos de trabajo que desarrollan software además de incluir las actividades técnicas habituales, tenga una orientación a stakeholders que pueda maximizar el éxito de los proyectos de software.

En la propuesta de la herramienta de análisis de stakeholders se muestran las dimensiones de análisis (interés, identidad, relación y membresía), las cuales para cada una se listan criterios que están orientados a la toma de decisiones por parte de los stakeholders asignados o que pueden ser parte de las actividades enfocadas a cambiar el enfoque de trabajo y orientarlo hacia la mejora de los proyectos de software. Para la columna de los stakeholders asociados se pretende remarcar y hacer un compendio de los diferentes stakeholders que pueden interactuar en cada actividad pudiendo incluir aquellos stakeholders diferentes a los propuestos y que puedan agilizar el proceso de cambio dirigido a los proyectos de software (**Ver anexo F**).

Cabe mencionar que la herramienta de orientación a stakeholders desarrollada en el presente trabajo tiene como objetivo centrar esfuerzos en aquellas actividades que puedan beneficiar a los desarrollos de software poniendo un enfoque especial en los stakeholders y ubicando aquellos que puedan ser clave en la toma de decisiones anticipada cuando se conforman equipos de trabajo o bien cuando un proyecto está en una fase avanzada pero con posibilidades de poderse implementar sin que afecte el tiempo de desarrollo del software.

Realizar un estudio de las percepciones de los grupos de interés acerca de su experiencia de relación con los proyectos de desarrollo de software se puede entonces realizar las siguientes actividades:

1. Posicionar a los diferentes grupos de interés con respecto al grado de conocimiento sobre los la empresa que desarrolla proyectos de software.
2. Analizar el proceso de relación entre los grupos de interés y la empresa de desarrollo de software.
3. Sondear espontáneamente los parámetros o atributos asignables al proyecto de desarrollo.
4. Evaluar de forma sugerida la importancia y satisfacción de los atributos identificados.
5. Identificar los principales puntos fuertes y áreas de mejora de la actividad de la empresa que desarrolla el proyecto de software.

### 3.5.2 PROCESO DE VALIDACIÓN

Para poder validar los instrumentos propuestos en la sección anterior se procedió a realizar una entrevista a especialistas en liderazgo y proyectos de software que pudieran, en primera instancia, explicar el trabajo que desarrollan dentro de sus organizaciones y en específico en proyectos de software así como conocer los principios fundamentales en los que se basan sus roles y responsabilidades pudiendo, por consiguiente, revisar y analizar la estructura de la herramienta, mediante sus propios criterios y experiencia laboral para de tal manera emitir un juicio profesional acerca de los alcances de la misma, su efectividad, uso, eficacia, etc.

Una vez que se establecieron los diferentes lineamientos funcionales para observar los instrumentos y obtener conclusiones acerca de su función, estructura, utilidad y alcance se obtienen conclusiones también acerca de las principales ventajas y desventajas que pudiera presentar. Los especialistas que se consultaron fueron los siguientes:

**Tabla 21** Especialistas en proyectos de software involucrados en el proceso de validación.

Especialista	Empresa	Cargo	Funciones	Lugar de ubicación de la empresa
E1	GNP	Gerente de desarrollo TI.	Dirección y gestión equipos de trabajo, toma de decisiones planeación, estrategia organizacional.	México D.F.
E2	AXXA	Auditor y administrador de proyectos.	GAP análisis, definición de planes de acción con apego al CMMI, ejecución de auditorías, identificación y registro de sugerencias, capacitación, administración de proyectos.	México D. F.
E3	OpenTec	Administrador de proyectos	Dirección y gestión del ciclo de vida de proyectos en OpenTec.	México D.F.
E4	Price Travel	Desarrollador analista TI.	Análisis, diseño y desarrollo de	Cancún, México

			aplicaciones en plataforma Android, IOS.	
--	--	--	--	--

En la tabla anterior, el nombre de cada especialista no es revelado por cuestiones de confidencialidad por parte de las empresas donde laboran, por lo que se les refiere a ellos como E(1, 2, 3, 4).

Las evidencias obtenidas de cada uno de ellos fueron llevadas a cabo de manera personal ya sea en su lugar de trabajo, en centros de encuentro así como haciendo uso de herramientas de comunicación a distancia como Skype (Caso Price Travel).

Una de las necesidades principales, antes de conocer la opinión de los expertos referente a los instrumentos de estructuras para la orientación a stakeholders, es el de conocer la percepción general acerca de las variables propuestas por la presente investigación (identidad, interés, relación y membresía) para poder medir el enfoque que tomaría la entrevista correspondiente (Tabla 22). Del sondeo de percepción se pudo obtener la siguiente información:

**Tabla 22** Percepción de variables de los especialistas entrevistados de sus diferentes proyectos asociados.

	<b>E1</b>	<b>E2</b>	<b>E3</b>	<b>E4</b>
<b>No. promedio de stakeholders por proyecto</b>	8	6	10	6
<b>Nivel de interés</b>	5.5	7.5	7.5	9
<b>Nivel de identidad</b>	7.5	7.5	5.5	7.5
<b>Nivel de relación</b>	5.5	7.5	9	9
<b>Nivel de membresía</b>	3.5	5.5	7.5	7.5

La tabla 22 muestra, en primera instancia, el número de stakeholders promedio de cada proyecto asociado a los especialistas así como la percepción que cada uno tiene en sus equipos de trabajo referente a las variables que contempla el presente trabajo (interés, identidad, relación y membresía). Dichas percepciones fueron evaluadas mediante un sondeo donde los entrevistados dieron una calificación entre cero y diez. La percepción del nivel de interés un 7.38, la del nivel de identidad 7, la del nivel de relación 7.75 y la de la membresía de 6 lo que indica que cada equipo de trabajo es similar en términos del número de sus stakeholders y que la percepción de las diferentes variables por parte de los especialistas es más o menos similar e influye de acuerdo a la situación de los proyectos donde se encuentren, es decir, el nivel de urgencia, la cantidad de proyectos a su cargo y el tamaño. Se puede concluir también que en promedio se percibe al interés, la identidad y la relación de manera similar no siendo el caso de la membresía o transmisión de habilidades y responsabilidades

entre stakeholders al ser un elemento con mayor número de elementos a analizar por considerarse a la transmisión de conocimientos y habilidades como una actividad que depende en su mayoría de la disponibilidad de tiempo de cada proyecto.

Una vez obtenidas las conclusiones generales acerca de las variables en que se basa esta investigación, se hizo muestra de la herramienta a los especialistas evidenciando sus características principales, la razón de su diseño, los stakeholders que contempla y el análisis que se puede obtener con ella. Momento después, los especialistas dieron su opinión respondiendo a distintos cuestionamientos donde se pudiera evidenciar, en base a su experiencia, múltiples factores como opiniones generales, la función, utilidad, alcance, ventajas, desventajas, faltantes y sobrantes. También, considerando que en todas las entrevistas puede surgir información extra que pueda ayudar a que las bases en que está fundamentada dicha herramienta y por consiguiente la investigación, se utiliza dicha información extra para complementar el presente estudio. La tabla 23 muestra la información cuantitativa resultado de las respuestas dadas por los especialistas donde se puede observar la agrupación de dichas respuestas según la variable que se esté midiendo. Para dicho estudio se consideraron categorías que pudieran recaudar mayor información y la menor ambigüedad acerca de la herramienta mostrada por el expositor dichas categorías son las siguientes:

- Opinión general de la herramienta
- Opinión sobre el diseño estructural
- Opinión sobre la función
- Opinión sobre la utilidad
- Opinión sobre el alcance
- Opinión sobre faltantes o sobrantes
- Ventajas / Desventajas

**Tabla 23** Información cuantitativa resultado de las respuestas dadas por los especialistas

CONCLUSIONES GENERALES POR CATEGORÍA
<b>OPINIÓN GENERAL DE LA HERRAMIENTA</b>
Se considera una propuesta efectiva de investigación poco implementada en los proyectos actuales de software. Existe una percepción general de que la mayoría de proyectos son de carácter reactivo y cuentan con poco tiempo de culminación por lo que la idea de orientar esfuerzos a los stakeholders vistos desde las cuatro capas fundamentales (interés, relación, membresía e identidad) maximizaría la probabilidad de éxito. No se tiene una idea muy clara de la forma de aplicación de la herramienta o su método de implementación pero en primera instancia se considera amigable y de fácil entendimiento.
<b>Diseño estructural</b>
Estructuralmente se considera ideal, de buena secuencia, y con cierta lógica. Se considera que es un análisis el cual puede ser usado como una parte de la implementación de un diagnóstico, no se visualiza la parte de presentación de resultados o bien una guía donde se muestre las actividades específicas a realizar en caso de tener carente algún criterio pero se especifica que el alcance de la herramienta esta en indicar criterios basados en las cuatro capas de análisis. Se apega a otras metodologías de administración de proyectos por lo que quien lo aplica podría estar identificado con su uso. Se debe aplicar en algún proyecto para medir su impacto.

<b>Función</b>
Se considera una herramienta que lleva a cabo un análisis exhaustivo de stakeholders y su entorno. Se considera que su aplicación en algún caso de estudio sería la mejor manera de medir su función. Al considerar que la mayoría de los proyectos requieren de acciones reactivas y que existen roles que llevan a cabo ciertas actividades poco centralizadas a ser orientadas hacia los stakeholders, la focalización hacia aquellos stakeholders prioritarios maximiza el valor que la empresa les brinda y no solo el de la satisfacción de sus propios objetivos. Por otro lado considera que los principios pueden ser aplicables a otros tipos de proyectos o agregar valor a otro tipo de metodologías.
<b>Utilidad</b>
La utilidad radica en aquellas organizaciones y por consiguiente proyectos que en realidad quieran considerar a los stakeholders como factor de cambio en el éxito de los proyectos. Lo que conlleva a que no solo sea útil en proyectos de software si no en otro tipo de proyectos. En el caso de proyectos de software se nota mayor utilidad hacia el grupo de stakeholders denominado como desarrolladores ya que son éstos lo que pueden obtener mayor provecho y beneficio de las herramientas y por consiguiente hacia el proyecto en sí pero debe ser responsabilidad de los stakeholders superiores el aplicar los principios para mejorar la cultura organizacional.
<b>Alcance</b>
La herramienta y los principios en los que se basa presentan alcance que tiene que ver con el momento en que es aplicada durante la vida del proyecto. Se considera que no solo puede ser óptima al inicio del proyecto sino también en etapas intermedias del mismo pudiendo corregir acciones orientadas a stakeholders. También, es posible que no solo esté limitada a un área en específico dentro de la organización sino también a donde se requiera llevar a cabo esta administración. Internamente presenta un alcance orientado al análisis de stakeholders más que a un diagnóstico de la situación de los proyectos.
<b>Faltantes o sobrantes</b>
La herramienta cuenta con las bases para ser funcional pero carece de una fase final de presentación de resultados o donde se pueda diferenciar las recomendaciones con las acciones en el plano real. Las acciones mencionadas deberían ser medibles de acuerdo a los estándares de cada organización. La herramienta es considerada estable en términos de sus sobrantes.
<b>Ventajas / Desventajas</b>
Las ventajas que presenta la herramienta tiene que ver con la profesionalización de un área de orientación a stakeholders ya que se puede entre muchas actividades de carácter prioritario, focalizar esfuerzos en términos sociales es decir entre las personas que están más comprometidas en el proceso de desarrollo y que reúnan mayores beneficios a los proyectos y así se pueda dar mayor valor al trabajo lo que puede originar un cambio de valores y estructura organizacional que beneficie a los proyectos. Por lo que las ventajas pueden visualizarse socialmente, hacia el proyecto y hacia la organización. Las desventajas que presenta la herramienta tienen que ver con el tiempo de dedicación a realizar tareas de gestión de stakeholders ya que en su mayoría los proyectos de software están limitados en tiempo y necesitan de mucho esfuerzo humano para llevarse a cabo lo que podría originar una falta de personal capacitado para implementar dichas actividades pudiendo impactar en costos para el proyecto.

### **3.5.3 DISCURSIÓN DE RESULTADOS ESPECÍFICOS RELACIONADOS A LA HERRAMIENTA DE ORIENTACIÓN A STAKEHOLDERS**

El objetivo del diseño de la herramienta de orientación a stakeholders es el de centralizar esfuerzos en crear recursos útiles para los individuos encargados de mejorar el proyecto de software en términos sea de tiempos, esfuerzos y/o costos basándose en la orientación hacia sus grupos de stakeholders por medio de un enfoque sistémico y de planeación partiendo del uso de teorías y técnicas orientadas al análisis, inclusión y participación de stakeholders. También, la presente propuesta debe incluir dentro de su estructura las bases metodológicas de responsabilidad social corporativa como marco conductor en la creación de valor empresarial haciendo uso del enfoque multilateral hacia los stakeholders. La percepción general acerca de la herramienta de orientación a stakeholders, basada en las capas de interés, identidad, relación y membresía es una idea necesaria para activar un cambio de paradigma organizacional dentro de las empresas que desarrollan software comercial.

Hablar de stakeholders dentro de éste tipo de proyectos es por lo general necesario pero muchas veces incómodo según los expertos, ya que el tipo de proyectos a los que va dirigida ésta investigación tienen una característica muy peculiar y que la diferencia de otro tipo de proyectos: El cambio tecnológico, la modificación de requerimientos por parte de los clientes, el tiempo de desarrollo de los proyectos y la falta de cultura organizacional orientada a la satisfacción de los individuos interesados en los proyectos.

Debido a las razones anteriores, se considera que existe un factor relevante a considerar, dicho factor es el tiempo de desarrollo del proyecto y los recursos que se tienen para resolver los problemas asociados. Una administración efectiva de stakeholders orientada a proyectos de software debe contemplar estrategias de implementación en equipos de trabajo apegándose a los requisitos técnicos básicos y de diseño de software por lo que es imperativo considerar, sea al inicio de un proyecto de software o durante una etapa temprana de desarrollo a un stakeholder encargado de realizar dicha gestión al igual que una habilidad por parte de las entidades de alta dirección (EAD) para detectar a la(s) persona(s) que puedan llevar a cabo dicha gestión sin limitar los esfuerzos ni las responsabilidades reales que el proyecto demande.

Es imperativo considerar también que el perfil de los stakeholders considerados marca un sesgo en el análisis y orientación a stakeholders ya que a pesar de los esfuerzos, existen distintos perfiles orientados a metodologías ya contempladas por experiencias laborales anteriores o bien por las mismas personas al momento de reaccionar ante la herramienta. Generalmente, si los proyectos presentan algún tipo de urgencia, los intereses de las personas pueden ser influenciados por factores que no consideran una buena gestión de stakeholders por lo que si no se lleva a cabo una etapa estratégica de identificación de perfiles, puede existir la probabilidad de que no se puedan implementar las herramientas propuestas. Se concluye pues que el marco estratégico de la responsabilidad social corporativa (RSC) brinda la posibilidad de cambiar el esfuerzo por general políticas internas para elegir al personal calificado que pueda aportar mayor valor a la organización utilizado aumentando el esquema de valores internos y beneficiando el nombre de la empresa por su enfoque hacia el talento empresarial.

En términos generales se concluye también que las bases por las cuales se fundamenta la presente investigación reúne los requerimientos necesarios para llevar a cabo un estudio de la

situación actual de equipos de desarrollo de software en términos de orientación a stakeholders estando limitado al análisis de estructuras por medio de la identificación de stakeholders, priorización, fomento de valores, análisis de relaciones y transmisión de conocimiento pero es necesario considerar el estado particular de urgencia de los proyectos y llevar a cabo un análisis de perfiles de stakeholders oportuno para identificar a las personas que pueden agilizar la aplicación de los recursos con una mayor efectividad.

Estructuralmente, la herramienta ofrece cinco etapas de análisis: Identificación y priorización de stakeholders, fomento de valores, compromisos, diseño de estructuras de stakeholders y criterios de evaluación del estado de orientación a stakeholders. Dicha estructura se percibe lógicamente coherente y de fácil entendimiento según los expertos. Al ser una estructura acotada a cumplir las cinco etapas o capas de diseño y limitada al análisis y evaluación del estado de orientación a stakeholders se considera una parte clave del posible diagnóstico que la empresa pueda ejecutar a equipos de desarrollo con el objetivo de medir su aceptación a orientar esfuerzos hacia sus stakeholders, Debido a los límites mencionados la herramienta no contempla una fase de presentación de resultados sino más bien contempla un análisis exhausto de los factores a considerar para el diseño de una guía de stakeholders donde se muestren situaciones orientadas a cumplir metodológicamente los objetivos de construcción de software, trabajo futuro que puede continuar con la presente investigación.

Funcionalmente, se considera que la herramienta cuenta con una estructura funcional y metodológica para ser implementada en proyectos de software, incluso en metodologías de desarrollo de software o bien en administración de proyectos las cuales pueden tener dentro de su estructura la gestión de stakeholders a un nivel de identificación sin contar necesariamente con la estructura y gestión de stakeholders. Se considera que el momento adecuado de implementación según los expertos, debe ser al inicio de cada proyecto o bien en alguna etapa posterior la cual permita controlar las actividades diseñadas para dicho fin y no debe estar limitada a una aplicación única en el tiempo de desarrollo del proyecto, se concluye que, debido a que se pueden agregar stakeholders relevantes en cualquier momento del desarrollo, es conveniente agregar funcionalmente etapas de inclusión de stakeholders nuevos, análisis de riesgos y mitigación. Por último, se considera que la gestión de stakeholders, al ser parte importante de la administración de proyectos, no está limitada a ser aplicada solo en proyectos de software, funcionalmente tiene las características funcionales de ejecución en otros tipos de proyectos, donde exista amplio conocimiento de las situaciones problemáticas de los stakeholders y se pueda llevar a cabo la etapa de identificación adecuada para ejecutar de una manera planeada y sistémica la presente herramienta.

En términos de su utilidad, se concluye que la herramienta debe estar dirigida a los stakeholders que en un principio tengan una motivación y compromisos suficientes para poder orientar esfuerzos hacia la gestión de stakeholders y cuyas responsabilidades sean de un rango mayor. Se menciona que las entidades de alta dirección (EAD) tienen la característica de poseer mayor poder dentro de las organizaciones y como se ha mencionado en numerosas ocasiones, existen stakeholders clave que deben activar el cumplimiento de acciones señaladas por la herramienta. Según los expertos, existe una utilidad especial para el caso de proyectos de software donde existen grupos grupo de stakeholders denominado como desarrolladores ya que son éstos lo que pueden obtener mayor provecho y beneficio de las herramientas y por consiguiente hacia el proyecto en sí pero debe ser responsabilidad de los stakeholders superiores el aplicar los principios para mejorar la cultura organizacional.

Se concluye de igual manera que, el margen de aplicación de la herramienta, como se mencionó anteriormente puede ser en cualquier etapa de desarrollo de software, siendo que el presente trabajo es un análisis del estado de orientación e inclusión a stakeholders y no un diagnóstico en su totalidad ya que, de ser así, existe la posibilidad de obtener distintos resultados al considerar nuevos stakeholders, lo que alteraría el entorno de aplicación y ejecución.

En términos generales, la herramienta fue considerada sin presentar ningún sobrante términos de su estructura, utilidad, función y alcance. Estructuralmente cumple con las metodologías comerciales y los responsables de aplicarla no tendrían ningún inconveniente según su experiencia laboral pero existe la posibilidad de que exista poca claridad para ejecutar las acciones recomendadas por medio de los criterios que brinda la herramienta, Según los expertos, se necesita una etapa final de presentación de resultados o bien una guía completa donde se consideren las acciones a realizar de acuerdo al resultado del análisis hecho por la herramienta. Se concluye que el trabajo está sesgado a un margen de análisis y no presenta una etapa que genere recursos al stakeholders de poder cambiar sus acciones y actitudes por sí sola además de originar una teoría que pueda explicar la movilización de stakeholders con beneficio de los proyectos de software.

Se considera de vital importancia mencionar que una de las ventajas encontradas durante el desarrollo de esta investigación es la de la profesionalización de un área pocas veces tomada en cuenta durante el desarrollo de proyectos de software y considerada mínimamente por las metodologías comerciales actuales, Se considera positivo centrar esfuerzos en aquellos individuos que tienen mayor interés por el proyecto pero además que sean parte de un proceso de análisis donde se evalúen sus niveles de relevancia y sean considerados pieza clave del éxito de los proyectos además de valorar el trabajo que realizan dentro de la organización.

Uno de los hallazgos que pudieron proporcionar mayor valor a la profesionalización de un área enfocada a stakeholders puede considerarse a la vez una desventaja. El tiempo de dedicación a realizar tareas de gestión de stakeholders, en su mayoría los proyectos de software están limitados en tiempo y necesitan de mucho esfuerzo humano para llevarse a cabo lo que podría originar una falta de personal capacitado para implementar dichas actividades pudiendo impactar en costos para el proyecto. Se concluye que se necesita un verdadero esfuerzo por parte de las EAD para lograr los objetivos planteados y esto solo se puede lograr si existe un cambio en la cultura organizacional que pueda dimensionar el valor y los beneficios que pueda ofrecer la orientación a stakeholders para cualquier proyecto de software para que si pudiera existir un costo agregado a la ejecución de los principios y herramienta de orientación a stakeholders, puedan ser compensados por los beneficios que puedan aportarles los propios stakeholders al proyecto.

Una vez analizando las conclusiones por parte de los especialistas y considerando los requerimientos para realizar una gestión efectiva de stakeholders se puede decir que actualmente no se cuenta con una herramienta enfocada a recabar la información necesaria que pueda ayudar a dicha gestión, por lo que uno de los trabajos futuros pueden centrarse en el diseño de una aplicación informática para gestionar la información y gestión de stakeholders pudiendo hacer uso de los recursos presentados en el presente trabajo y así tener como un complemento adicional que ayude a cumplir los objetivos de esta investigación.

## CAPÍTULO 4

### CONCLUSIONES

El análisis llevado a cabo durante el presente trabajo, objetivo de la investigación, fue resultado de una necesidad cuyo principio se basa en el éxito de los proyectos de software en términos ya sea de costos, tiempos y/o esfuerzos con una perspectiva orientada a los stakeholders asociados. La base para dicho estudio tuvo como fundamento principal el análisis de las estructuras contemporáneas de stakeholders relacionadas al tipo de proyecto (software libre y privado) para encontrar diversos comportamientos que pudieran ser clave en la aplicación de una metodología que accionara el interés de las organizaciones en focalizar los esfuerzos en sus stakeholders y por consiguiente, aumentara la probabilidad de mejorar las condiciones en las que se desarrollan dichos proyectos.

Uno de los objetivos del presente trabajo es el de encontrar las ideas y fundamentos estratégicos y funcionales de gestión y orientación a stakeholders en proyectos de software que pudieran soportar ésta propuesta por medio de estructuras de stakeholders dentro de las organizaciones en tecnologías de la información y las comunicaciones, las cuales presentan características que valen la pena señalar y que de igual manera son fundamentales para diseñar la presente propuesta exhaustiva, sistémica y planeada que contemplara las condiciones actuales de los equipos de trabajo como lo son el tamaño del proyecto, los stakeholders relacionados, las metodologías que se ocupan para desarrollar y administrar los proyectos, el ambiente laboral, la cultura organizacional entre otros. Se concluye que la parte fundamental por la cual está basado éste trabajo es la *orientación* a stakeholders de los grupos de desarrollo de proyectos de software y no simplemente del análisis de estructuras de stakeholders (roles, funciones, posiciones) como se planteó originalmente. El motivo del cambio de enfoque es debido a que se identificaron factores relevantes que afectan el diseño de dichas estructuras únicamente a uno más general, es decir, en términos de orientación, dichos factores son los siguientes:

- Se necesita crear un proceso previo de identificación y priorización a stakeholders para poder analizar estructuras de stakeholders en términos de roles, funciones y posiciones.
- Si se planteara de manera única la propuesta de estructuras de stakeholders, se dejaría a un lado el hecho de analizar los roles de cada stakeholder para poder entender comportamientos, similitudes, etc. que ayuden a crear nuevas formas de organización en los equipos de desarrollo.
- Se necesita saber el estado real del proyecto de software para poder evaluar su nivel de urgencia, lo que puede impactar en la organización de estructuras funcionales de los stakeholders asociados.
- Se necesita crear parámetros, criterios o estándares que puedan impactar directa o indirectamente en las actividades cotidianas de los grupos de stakeholders para que de manera estratégica puedan existir acciones planeadas de mejora en los proyectos de software.

Dichos factores fueron la razón de realizar un proceso organizado de orientación a stakeholders haciendo uso de las estructuras analizadas y de los factores asociados su diseño para impactar de una manera más controlada los procesos organizacionales dentro de los proyectos.

A partir de la modificación del enfoque anterior, surge la necesidad de encontrar los límites de análisis que pudieran ayudar a entender las circunstancias en que se realiza el desarrollo de proyectos de software y que pueden impactar en el éxito de los mismos, por lo que por medio del estudio la teoría de stakeholders, herramientas con enfoques a stakeholders y el marco de la responsabilidad social corporativa (RSC) se determinó la necesidad de explicar la movilización de los stakeholders, sus motivantes y su deseo por mejorar el desempeño aplicado en los proyectos cuyos resultados fueron el análisis de dichos límites, producto de una investigación objetiva y con fines en determinar si las estructuras de stakeholders son un factor clave en el éxito de éste tipo de proyectos. Los límites de análisis son llamados "capas" y tienen su origen en principios orientados al accionar de los stakeholders, es decir, de las acciones que provocan su movilización dentro y fuera de una organización con la premisa de agilizar los procesos internos de desarrollo de software. Para incluir dichas capas en un proceso de planeación que pudiera contemplar a los stakeholders sin perder los objetivos estratégicos y tácticos en el desarrollo de software, se planteó en uso de la planeación interactiva la cual brindó, bajo el principio holístico, un diseño que contemplara a las capas en un marco de coordinación para reunir las actividades de cada capa y poder ser vistas como una única asociación a la vez que se pudieran integrar en un marco estratégico de orientación a stakeholders lo que ayudó a entender la importancia de incluir los diferentes elementos analizados dentro de una misma estructura de diseño.

Se llegó a la conclusión de que el proceso de planeación interactiva conviene al diseño de un marco de planeación con enfoque a stakeholders aplicado a proyectos de software y que además existen razones que pueden hacer que los grupos de stakeholders puedan enfocar sus esfuerzos de una manera planeada y organizada como lo son *satisfacción de intereses, mejoramiento de condiciones laborales, crecimiento laboral y profesional, procesos de comunicación*, entre otras, por medio de dichas capas: Interés, identidad, relación y membresía asociando modelos estructurados que relacionan variables asociadas a cada capa centralizando esfuerzos en el éxito de los proyectos de software.

Las cuatro capas de análisis son agregadas al estudio de orientación a stakeholders y lo relacionan directamente al trabajo que desempeñan, es decir, al desarrollo de proyectos, implementación de metodologías, aplicación de conocimientos técnicos y estratégicos así como prácticas organizacionales. El interés tiene que ver con las razones personales y grupales que hacen o provocan una movilización en los grupos de trabajo y sus miembros pero que no siempre es dirigido hacia los objetivos empresariales, si no por el contrario, es dirigido a satisfacer objetivos personales y específicos sin considerar los intereses de los demás. Es por ello que se pensó en proponer una estructura que respondiera a la integración de intereses en los equipos de trabajo y que además fuera compartida para que no solo se cumplan los intereses individuales de los stakeholders si no que éstos respondan a las verdaderas necesidades y requerimientos de los proyectos como lo son, tiempos de aplicación, cumplimiento de calidad y eficiencia, estructura funcional y técnica entre otros. Se concluyó que el interés debe ser compartido y similar en cada integrante del equipo de trabajo ya que existe la probabilidad de que se enfoquen las acciones de manera efectiva y rápida a mejorar

los proyectos de software.

La identidad por el contrario no debe ser compartida al menos entre los equipos de trabajo que realicen un mismo proyecto y si debe ser individual o al menos única dentro de cada equipo. La identidad tiene su base teórica en los valores que puedan tener los individuos y que los hacen tener un nivel de pertenencia al trabajo que realizan, a la organización donde pertenecen o al equipo de trabajo. El análisis de la estructura de identidad responde entonces al nivel de compromiso que se tiene dentro de los equipos de trabajo y que puede hacer más eficiente y enfocada la movilización en crear mejoras en un proyecto de software. Diversos son los valores que pueden reforzar la identidad en un equipo de trabajo, uno de ellos es la benevolencia la cual, a partir de diversos esfuerzos por crear identidad individual puede hacer que el stakeholder incremente su nivel de productividad apoyando a otros y agregando valor al trabajo realizado. Otros valores como la responsabilidad social se agregan al uso de la identidad para éste estudio. La responsabilidad social dentro de la organización o también llamada corporativa (RSC) agrega valor a la cultura organizacional interna y externa de los individuos ya que al ser considerada internamente, debe cumplir con estándares de cumplimiento de objetivos y focalización a stakeholders, en éste caso si dichos esfuerzos resultado de la RSC son centrados en los stakeholders más cercanos al proyecto, puede existir la probabilidad de incrementar valor al trabajo en conjunto.

En tal caso se concluye que la RSC es un enfoque vital en el desarrollo del presente trabajo al contemplar una estructura funcional orientada a los valores empresariales, de entre ellos la identidad, pero que además surjan políticas creadas por las entidades de alta dirección destinadas a ejecutar dichos cambios internos por medio de su enfoque multilateral donde participen los stakeholders. La única limitante de la variable de identidad y del uso de la RSC podría estar sesgada en el carácter voluntario de la adquisición de la RSC dentro de las empresas por lo que debe ser un esfuerzo totalmente estratégico de las EAD para que los resultados del enfoque de identidad puedan ser tangibles y enfocados al desarrollo de proyectos de software.

La relación, es la capa que define los niveles personales de inclusión laboral, emocional, profesional y de interés entre los stakeholders de un proyecto de software. Se consideró su nivel de importancia para el presente estudio por ser un factor muchas veces omitido en los equipos de trabajo y poco fomentado por los líderes y gerentes de proyectos de software. Se concluyó que estudiar el nivel de relación entre los stakeholders de proyectos de software ayudaría a entender y dirigir esfuerzos en accione del pasado y presente con miras de aplicación en escenarios futuros como lo son medios de comunicación entre stakeholders, grupos de poder, estrategias de trabajo y toma de decisiones entre otros. La estructura asociada a la relación entre stakeholders ayudó a delimitar la manera en que pueden interactuar los stakeholders clave para así tener efectos positivos en el desarrollo de los proyectos.

Por último la membresía, cuya relevancia radica en ser una propuesta funcional al uso adecuado del cocimiento y el aprovechamiento que pueden darle los distintos roles de un proyecto hacia y desde los subordinados. La razón por la que la membresía fue considerada y nombrada es para definir toda aquella acción entre los stakeholders de un proyecto donde la transmisión de conocimiento entre los distintos roles que pueden existir en los equipos de trabajo sea una realidad más que una habilidad entre ciertos stakeholders. Se propone que

dicha transmisión de conocimiento puede funcionar como una membresía ya que tiene la bondad de brindar beneficios personales a quien la posee así como darle derechos y por consiguiente virtudes a quien recibe el conocimiento. Se concluye que no existe una profesionalización del conocimiento entre stakeholders de equipos de desarrollo en proyectos de software que reúna el apoyo de distintas partes estratégicas como lo son la administración y planeación de dichos proyectos, el uso efectivo de recursos y relaciones estables dentro de las organizaciones.

La estructura relacionada a la membresía fue diseñada para identificar a los stakeholders que pueden transmitir su conocimiento de manera efectiva sin interrumpir sus responsabilidades y excluye a aquellos stakeholders prioritarios del análisis que no se benefician con el conocimiento de otros pero que si pueden aprovechar su entorno para aprender nuevas cosas. Podemos hasta este punto, justificar las razones por las que las organizaciones deben enfocar sus esfuerzos hacia la administración de stakeholders clave por medio de las cuatro capas señaladas pero se considera también que todas las acciones que se puedan realizar deben cumplir con objetivos específicos que puedan ser tangibles para aquellos stakeholders que sean responsables de aplicar herramientas, actividades y acciones en beneficio de los proyectos. Es por ello que para efectos prácticos se planteó la posibilidad de diseñar una herramienta que pueda ser fácil de usar a la vez que contenga recursos que potencialicen el cumplimiento de los objetivos de orientación a stakeholders.

Se identificaron a los stakeholders dentro de proyectos de software privado y el resultado de ello ayudó a darle importancia a una primera etapa de análisis: La identificación y priorización de stakeholders. Dicha etapa es fundamental en el inicio de la aplicación de metodologías de stakeholders y agrega valor a la presente investigación. La identificación de stakeholders debe ser un recurso que todos los líderes de proyectos o administradores de igual manera pueden utilizar pero a la vez que no presente atrasos en los objetivos técnicos del proyecto relacionado. Se planteó la propuesta de seguimiento a stakeholders donde se pueden evaluar y listar los datos generales de cada individuo y de igual manera de identificar sus intereses dando posibilidad a un primer acercamiento a la capa de interés mencionada anteriormente.

Se concluyó que si los intereses son identificados pueden ser tomados en cuenta pero no precisamente priorizados por las Entidades de Alta Dirección (EAD) por lo que además de llevar a cabo una etapa de identificación de stakeholders y de sus intereses se propuso un estudio posterior en la evaluación de su priorización el cual podría potencializar acciones de mejora que contemplaran dicha priorización y que impactaran directa o indirectamente en el desarrollo de los proyectos. Se pudo observar que existen diversas herramientas para llevar a cabo la etapa de priorización y se consideraron dos por ser el más claro y eficiente dentro de éste estudio: El uso de los círculos de Venn y los cuadrantes de priorización de stakeholders.

Ambas herramientas forman parte de la segunda etapa de análisis dentro de la herramienta y brindan la posibilidad de priorizar a los stakeholders que tuvieran mayor poder e influencia dentro de los equipos de trabajo. En primera instancia la aplicación de los círculos de Venn cuyo objetivo es el de identificar a aquellos stakeholders por medio de tres variables: Poder, urgencia y legitimidad. Se concluyó que a través de estas tres variables se puede diferenciar a aquellos stakeholders que necesitan mayor atención y una gestión más eficiente ya que al ser un ejercicio de reflexión por parte de quien aplica dicha herramienta, los cuestionamientos

surgidos como ¿Qué stakeholder tiene poder y legitimidad pero no tiene urgencia? sirven principalmente para llegar a dicha clasificación. Se observó que a pesar de no considerar clasificaciones de grupos de stakeholders, se tenía cierto conocimiento de los grupos de poder dentro de la organización, su nivel de urgencia y su credibilidad pero pocas veces se daba importancia a dichos factores por considerarlos parte del mismo ambiente de trabajo.

El segundo instrumento para realizar la priorización de stakeholders es el del cuadrante de priorización de stakeholders. Dicha herramienta fue considerada como un segundo nivel de análisis previo al de los círculos de Venn y su justificación de uso radica en reforzar los esfuerzos orientados hacia los stakeholders clave pero además detectar los niveles de enfoque en la satisfacción de objetivos entre grupos de interés. Uno de los resultados obtenidos en el uso de dicho cuadrante está reflejado en la satisfacción del objetivo número dos de la herramienta: la priorización objetiva de stakeholders lo que ayudo a diseñar las estructuras de stakeholders que respondieran a cada una de las capas de análisis y se pudieran agrupar de una forma sistémica y organizada aquellos stakeholders identificados, se ubicaran los stakeholders prioritarios y se ejecutaran los principios teóricos en los que se basa cada capa de análisis. Se pudo observar que una agrupación de los stakeholders cumpliendo los principios propuestos de interés, identidad, relación y membresía ayudaba a visualizar de una manera más eficiente la forma de agrupación de los mismos pudiendo servir como guía para futuras implementaciones.

La herramienta, al contemplar dichas estructuras, además de identificar a los stakeholders prioritarios explica las razones de agrupación y las actividades a tomar en cuenta para cada caso mostrando al implementador una idea aproximada en la que puede reagrupar al equipo de desarrollo siguiendo los principios de las capas ya mencionadas e identificando a los stakeholders que deben accionar los factores de cambio para así, y de una manera controlada, lograr un mayor control en los grupos de poder centrados mayoritariamente en realizar las actividades que satisfagan sus propios intereses sin buscar la satisfacción y cumplimiento del de los demás.

Paso siguiente al de la reagrupación de stakeholders por medio de estructuras orientadas y funcionales se identificó la necesidad de implementar acciones que reforzaran la capa de identidad, entre los integrantes pudiendo identificar acciones destinadas a mejorar escalas de valores asociadas a los stakeholders. Uno de ellos es el valor de la benevolencia, la cual está orientada a las acciones de los individuos con intereses propios pero que buscan la satisfacción de los de los demás por medio de ayuda y cooperación a las actividades empresariales. Se pudo determinar que en efecto, la escala de valores es un factor clave para la concentración de actividades morales y psicológicas que ayudaran al cumplimiento de objetivos por parte de los stakeholders. En el caso de proyectos de software, donde se presentan características peculiares como la urgencia y la culminación de módulos por entregables, pocas veces se considera enriquecer los valores entre los stakeholders que puedan accionar su movilización hacia objetivos más específicos relacionados al proyecto. Se consideró que en principio, dichos valores deben ser fomentados y canalizados desde las entidades de alta dirección (EAD) cuya responsabilidad social genere factores de cambio en la actitud de los miembros de trabajo pero que además pueda diferenciarse del trabajo de otros equipos asociados al mismo proyecto.

Se concluye entonces que dicho fomento de valores aunado al enfoque único de identidad que deben tener los miembros del equipo de trabajo aumenta la probabilidad de movilización de los stakeholders por lo que como resultado de dichas acciones se pudo determinar que es necesario que exista un verdadero compromiso por parte de las EAD hacia sus stakeholders identificados. Se concluyó que es necesario realizar un estudio subsecuente el cual mida la percepción de los grupos de stakeholders referentes al proyecto donde trabajan donde se pueda entre muchas cosas posicionar a los stakeholders respecto al grado de conocimiento que tengan en las distintas actividades, medir como se están relacionando los grupos de stakeholders y el efecto que éstas relaciones tienen con respecto al logro de los objetivos del proyecto para así sondear de manera correcta dichos objetivos asociados a los resultados de interacción de stakeholders así como identificar los principales puntos fuertes y áreas de mejora de la actividad empresarial que desarrolla el proyecto de software.

Como parte final del uso e implementación de la herramienta de orientación a stakeholders, el responsable de aplicarla tiene la posibilidad de aterrizar las conclusiones hechas en etapas anteriores por medio de una serie de criterios divididos por las diferentes capas de análisis donde se brinda la oportunidad de medirlos siguiendo patrones de aplicación a base de recomendaciones así como los stakeholders asociados que puedan accionar dichos cambios. Es importante señalar que el alcance de dicha herramienta y por tanto de la investigación se centra en el análisis de stakeholders partiendo de la identificación, la priorización la interrelación, creación de valores, el compromiso con los stakeholders y establecimiento de criterios orientados a la administración de stakeholders como se mostró en etapas anteriores lo que establece el marco de alcance teórico y funcional de la presente investigación. Si bien es cierto que la herramienta tiene un margen de aplicación establecido por los criterios asociados a cada capa de análisis, no pretende ser un marco que muestre los pasos a seguir para mejorar la gestión de stakeholders efectiva por medio de acciones concretas si no el de proporcionar un marco planeado de acción el cual oriente esfuerzos a incentivar acciones concretas realizadas por los stakeholders y dirigidas a los proyectos de software por medio de las cuatro capas de análisis utilizando el conjunto de herramientas propuestas en el presente trabajo.

Se concluye que los mecanismos anteriores podrían responder importantes cuestionamientos resultado de la administración de stakeholders y buenas prácticas recomendadas anteriormente. De la información que se puede obtener está la del interés financiero o emocional en el proyecto de software, si es positivo o negativo dicho interés, los factores de motivación de stakeholders, el tipo de información del proyecto que se requiere para tener un buen desempeño en el trabajo, la forma en que se quiere recibir la información solicitada, la forma más efectiva de comunicarse entre los stakeholders, la opinión que se tiene acerca del proyecto y si ésta basada en una fuente confiable, las personas que influyen las opiniones en general y aquella del proyecto, si conviene dicha influencia a las personas que generan influencia en stakeholders y por último la forma de ganar su apoyo para el proyecto al igual que la forma de su oposición.

Es importante mencionar que si no existe una buena comunicación fomentada por el líder del proyecto hacia los distintos grupos de interés, lo anterior puede carecer de propiedades y características esenciales para el desarrollo de proyectos en cuestión. De igual manera, se considera que las capas de trabajo reflejadas por medio de estructuras de stakeholders orientadas a proyectos de software aunadas a los diferentes esfuerzos de las EAD apoyados

por la participación de los stakeholders asociados y tomando en cuenta la responsabilidad social corporativa en dicha organización aumentará la probabilidad éxito de los proyectos de software así como de dirigir esfuerzos reflejados en intereses, identidad, formas de relación y transmisión de conocimientos aplicados en la mejora de los proyectos de software una vez que hayan sido considerados de igual manera factores técnicos asociados al desarrollo mismo de los proyectos como lo son la gestión óptima de requisitos para desarrollar el proyecto y una buena ingeniería de requisitos .

Se concluye que un proyecto de software puede tener éxito o no al considerar diversos factores de entre los cuales se propone la gestión y orientación a stakeholders como detonante de cambio organizacional sistémico y planeado de los equipos de trabajo, ya que, al identificar la definición de éxito de proyectos de software planteada por el presente trabajo, donde se contemplan los recursos humanos y los sistemas de trabajo, se priorizan actividades única y exclusivamente a la unión de dichos factores haciendo a la gestión de stakeholders prioritaria dentro del proceso de desarrollo de un proyecto y no solo parte funcional de una metodología de desarrollo de software. Se puede decir que es probable que un desarrollo de software tenga éxito debido a la fusión de factores técnicos de desarrollo de software iniciales como lo son la elicitación de requisitos por los cuales fue concebido y una correcta ingeniería de requisitos y a factores de orientación a stakeholders como lo son las cuatro capas de análisis: interés, identidad, relación y membresía. La ingeniería de requisitos estará definida por medio de la elicitación de los requisitos llevadas a cabo por los stakeholders clave para dicha actividad una vez que se ha identificado el problema a resolver, la identificación correcta de los stakeholders, la formación de estructuras organizacionales basadas en principios dirigidos a la movilización de stakeholders, de la buena comunicación entre los grupos de stakeholders, el fomento de valores y el establecimiento de relaciones entre el equipo de desarrollo y el cliente además de las escalas organizacionales básicas como lo son la toma de decisiones y la elección de la tecnología adecuada y que el presente trabajo considera como una ampliación a la presente investigación al ser factores considerados como parte del sistema pero vistos de manera estratégica.





ANEXO B

FORMATO DE GESTIÓN DE RELACIONES DE STAKEHOLDERS

NIVEL DE PRIORIDAD	TIPO DE RELACIÓN	Stakeholder	PRINCIPALES HERRAMIENTAS DE DIALOGO	CUESTIONES PLANTEADAS POR LOS STAKEHOLDERS	ACCIÓN IMPLANTADA	RESULTADOS	ESTADO DE AVANCE
Gestionar atentamente	A						
Mantener Satisfecho	B						
Mantener informado	C						
Monitorear (mínimo esfuerzo)	B						

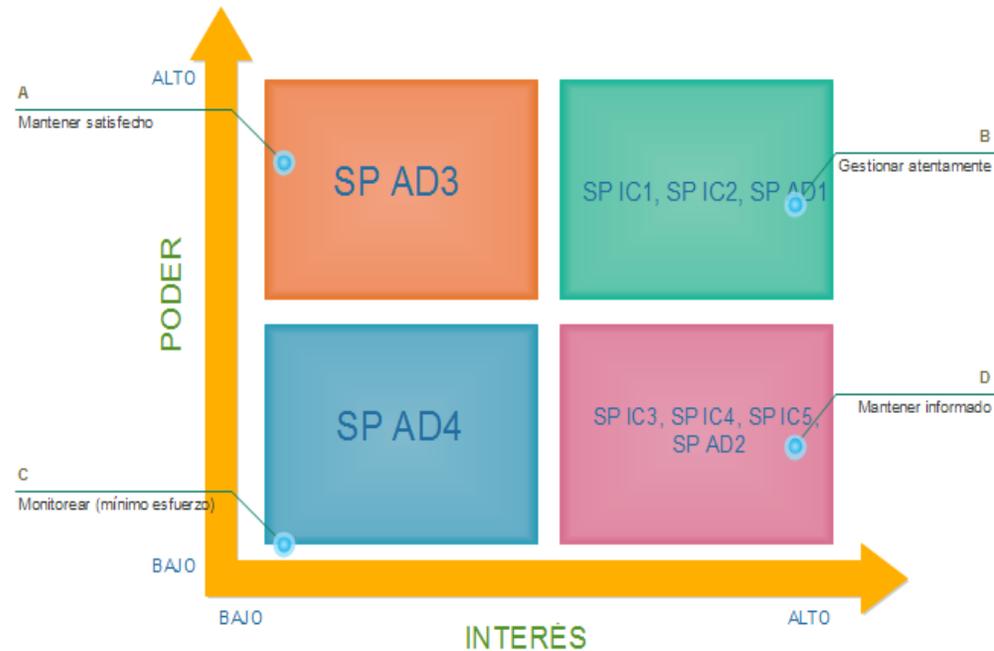
El presente formato es utilizado después de realizar el análisis de priorización de stakeholders donde se definen los niveles de prioridad para ubicar el tipo de relación a utilizar y enfocar esfuerzos al seguimiento de actividades para mejorar las relaciones entre los grupos de stakeholders.

## ANEXO C

### MATRIZ DE PRIORIZACIÓN DE STAKEHOLDERS DE EMPRESAS DESARROLLADORAS DE SOFTWARE

CATEGORÍA	STAKEHOLDER	ID
INTERNOS (COMPROMETIDOS)	Propietario del producto	SP-IC1
	Lider del proyecto	SP-IC2
	Equipo de desarrollo	SP-IC3
	Equipo de pruebas o testers	SP-IC4
	Consultores del proyecto	SP-IC5

CATEGORÍA	STAKEHOLDER	ID
AJENOS AL PROCESO DE DESARROLLO (INVOLUCRADOS)	Cientes finales	SP-AD1
	Usuarios finales	SP-AD2
	Gerentes del proyecto	SP-AD3
	Recursos Humanos	SP-AD4



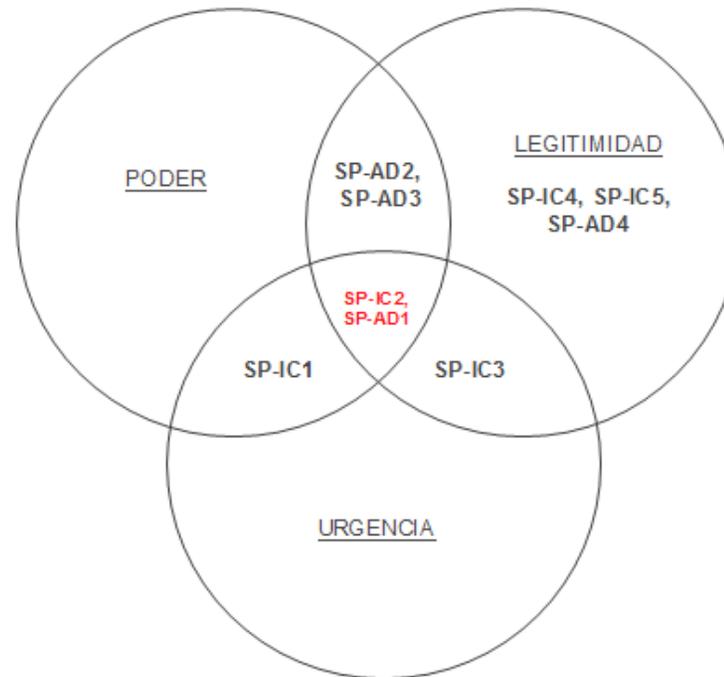
La matriz de priorización de stakeholders usa la clasificación de stakeholders para proyectos privados la cual es dividida en usuarios internos o comprometidos y ajenos al proceso de desarrollo o involucrados. Por medio del identificador pueden ser ubicados en el cuadrante de interés y poder donde a su vez pueden ser clasificados con la clasificación A (mantener satisfecho), B (gestionar atentamente), C (monitorear/mínimo esfuerzo) y D (mantener informado).

## ANEXO D

### CIRCULO DE VENN DE LOS STAKEHOLDERS RELEVANTES DE LOS PROYECTOS DE SOFTWARE

CATEGORÍA	STAKEHOLDER	ID
INTERNOS (COMPROMETIDOS)	Propietario del producto	SP-IC1
	Lider del proyecto	SP-IC2
	Equipo de desarrollo	SP-IC3
	Equipo de pruebas o testers	SP-IC4
	Consultores del proyecto	SP-IC5

CATEGORÍA	STAKEHOLDER	ID
AJENOS AL PROCESO DE DESARROLLO (INVOLUCRADOS)	Cientes finales	SP-AD1
	Usuarios finales	SP-AD2
	Gerentes del proyecto	SP-AD3
	Recursos Humanos	SP-AD4



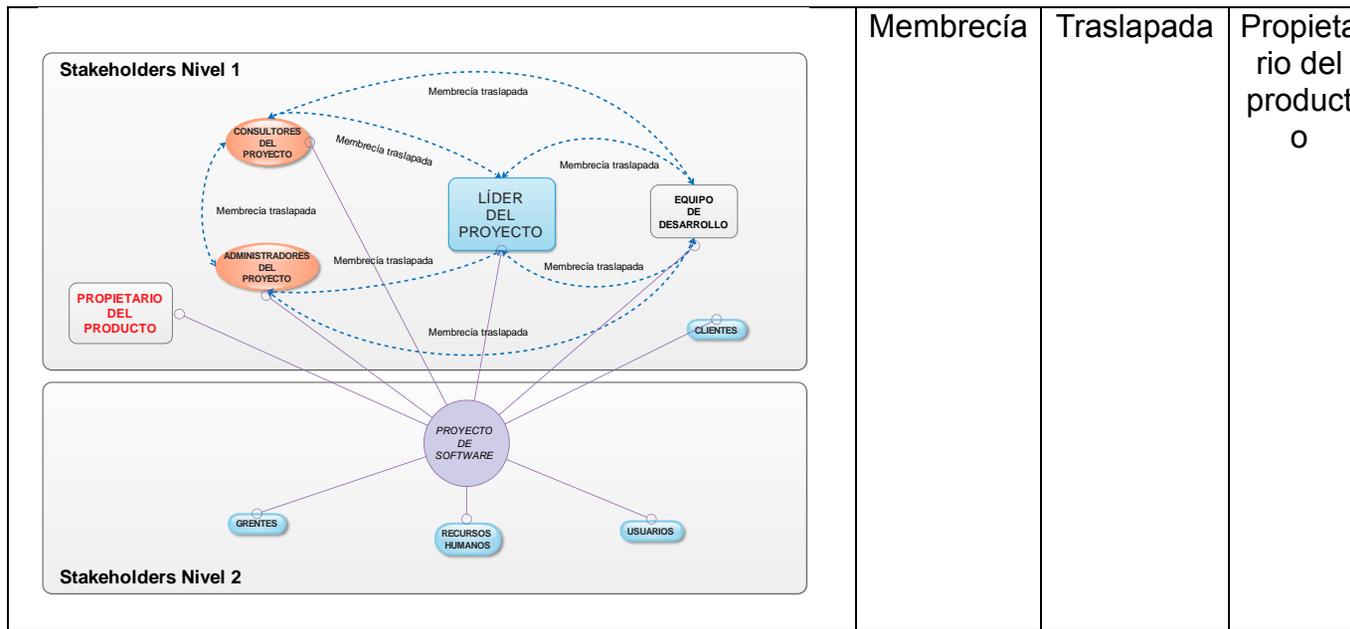
La herramienta anterior (Círculos de Venn) aplicada a las tres variables de stakeholders relevantes (poder, legitimidad, urgencia) ejemplifica el uso de la clasificación de stakeholders para el desarrollo de software privado (internos y ajenos al proceso de desarrollo) los cuales son ubicados dentro de los círculos dando relevancia y significado a los stakeholders que deben ser considerados.

## ANEXO E

### FORMATO DE ORGANIZACIÓN INTERNA DE LA ESTRUCTURA DE STAKEHOLDERS DE UN PROYECTO DE SOFTWARE BASADO EN LAS CAPAS DE INTERÉS, IDENTIDAD, RELACIÓN Y MEMBRESÍA.

ORGANIZACIÓN ESTRUCTURAL DE STAKEHOLDERS EN PROYECTOS DE SOFTWARE			
Estructura	Variable relacionada	Nivel de la variable relacionada	Stakeholders prioritarios
	Interés	Traslapada	Propietario del producto, líder

	<p>Identidad</p>	<p>Única</p>	<p>Propietario del producto, PM, consultores del proyecto, líder, equipo de desarrollo, gerentes, RH, usuarios, clientes</p>
<p>Relaciones directas</p> <p>Retroalimentación</p>	<p>Relación</p>	<p>Traslapada</p>	<p>Propietario del producto, líder, clientes</p>



El formato anterior muestra la forma de organización de stakeholders siguiendo las cuatro variables encontradas en el presente trabajo (interés, identidad, relación y membresía). Se pueden utilizar para establecer principios organizativos que atiendan a las necesidades del proyecto en cuestión así como determinar el dominio de la variable y sus stakeholders asociados.

ANEXO F

CRITERIOS DE ORIENTACIÓN A STAKEHOLDERS DE PROYECTOS DE SOFTWARE

Estructura	Variable	Criterios	Stakeholder asociado
1	INTERÉS	Identificación de los stakeholders	Gerente, LIDER, PM
		Delimitación de fortalezas y debilidades de los stakeholders	Líder
		Nivel de relación entre los stakeholders	Líder, Gerente
		Relevancia de intereses de los stakeholders	Líder, Gerente, RH
		Identificación de descontentos de stakeholders	Líder, Gerente
		Protección de intereses de los stakeholders	Propietario del producto, Gerente, Líder, Equipo de desarrollo, RH
		Clasificación de intereses de los stakeholders	Gerente, Líder
		Acceso a recursos de la empresa	Propietario del producto, Gerente
		La participación de los stakeholders en la toma de decisiones de la empresa	Propietario del producto, Gerente, Líder
		Registro de acciones en el pasado	Gerente, Líder
		Traslape de interés	Propietario del producto, Gerente, Líder, Equipo de desarrollo, Consultores del proyecto, PM, RH, Usuarios, Clientes

2	IDENTIDAD	Identidad individual	Propietario del producto, Gerente, Líder, Equipo de desarrollo, Consultores del proyecto, PM, RH, Usuarios, Clientes
		Identidad colectiva	Propietario del producto, Gerente, Líder, Equipo de desarrollo, Consultores del proyecto, PM, RH, Usuarios, Clientes
		Identificación de valores: Solidaridad, benevolencia, ayuda	Líderes
		Fomento de valores	Propietario del producto, Gerente, Líder, Equipo de desarrollo, Consultores del proyecto, PM, RH, Usuarios, Clientes
		Formación de grupos estratégicos	Líder
		Adaptación organizacional	Gerente, Líder
		Labor emocional	Líder
		Redes personales	Gerente, Líder, Clientes, RH
		Reconocimiento al trabajo	Propietario del producto, Gerente, Líder, Equipo de desarrollo, Clientes, Usuarios, RH
3	RELACION	Registro y clasificación de vínculos	Gerente, Líder
		Registro de acciones colectivas en el pasado	Gerente, Líder
		Registro de acciones colectivas en el presente	Gerente, Líder
		Densidad de vínculos	Líder

		Traslape de relación	Propietario del producto, Gerente, Líder, Equipo de desarrollo, Consultores del proyecto, PM, RH, Usuarios, Clientes
4	MEMBRESÍA	Registro de experiencia en el trabajo de los stakeholders	Gerente, Líder
		Nivel de responsabilidades de los stakeholders	Propietario del producto, Gerente, Líder, Equipo de desarrollo, PM
		Registro de responsabilidades	Líder
		Registro de funciones	Líder
		Traslape de membresía	PM, Administradores del proyecto, Líder, Equipo de desarrollo
		Niveles de poder en los stakeholders	Gerente, Líder, Clientes
		Análisis de la continuidad	Propietario del producto, Gerente, Líder, Clientes

En la propuesta de la herramienta de análisis de stakeholders se muestran las dimensiones de análisis (interés, identidad, relación y membresía) las cuales para cada una se listan criterios que están orientados a la toma de decisiones por parte de los stakeholders asignados o que pueden ser parte de las actividades enfocadas a cambiar el enfoque de trabajo y orientarlo hacia la mejora de los proyectos de software

## REFERENCIAS

- Barrera, A. M. (1999). La responsabilidad social de la gerencia. Monografías de administración Serie Empresa, Economía y Sociedad, 54
- Jones, T. y Wicks, A. (1999). Convergent stakeholder Theory. *The Academy of Management Review*, 24(2), 206-221.
- Freeman, R. (1984). *Strategic Management: A stakeholder approach*. Boston, MA: Pitman.
- Mitchell, R., Agle, B. y Wood, D. (1997). Toward a theory of stakeholder identification and salience: defining the principle of who and what really counts. *Academy of Management Review*, 22,853-886.
- Clarkson, M. (1995). A stakeholder framework for analyzing and evaluating corporate social performance. *Academy of Management Review*, 20(1), 92-117.
- Rowley, T. (1997). Moving beyond Dyadic Ties: A Network Theory of stakeholder Influences. *The Academy of Management Review*, 22(4), 887-910.
- Waddock, S. y Graves, S. (1997). The Corporate Social Performance-Financial Performance Link. *Strategic Management Journal*, 18(4), 303-319.
- Ogden, S. y Watson, R. (1999). Corporate Performance and stakeholder Management: Balancing Shareholder and Customer Interests in the U.K. Privatized Water Industry. *The Academy of Management Journal*, 42(5), 526-538.
- Scott, S. y Lane, V. (2000). A stakeholder Approach to Organizational Identity. *The Academy of Management Review*, 25(1), 43-62.
- Schneider, M. (2002). A stakeholder Model of Organizational Leadership. *Organization Science*, 13(2), 209-220.
- Donaldson, T. y Preston, L. (1995). The stakeholder theory of the corporation: concepts, evidence and implications. *Academy of Management Review*, 20(1), 65-92.
- Mitchell, R., Agle, B. y Wood, D. (1997). Toward a theory of stakeholder identification and salience: defining the principle of who and what really counts. *Academy of Management Review*, 22,853-886.
- Luhmann, N. (1998). *Teoría de la sociedad*. México DF: Triana.
- Timothy J. Rowley y Mihnea M. (2003). When will stakeholders groups act? An interesting and Identity based model of stakeholder group mobilization. *The Academy of Management Review*, 204

García-García, J. y Alonso de Magdaleno, M. (2013). Comunicación de responsabilidad social en el sector del software libre. *Universia Business Review* (2014), 98-124.

Katuska Cabrera, M. Suárez, M. Cruz, D.D. (2013) y Martín Santana, J. El clima familiar y la benevolencia entre miembros del equipo de alta dirección como antecedentes de la orientación hacia los stakeholders en la empresa familiar, *Revista Europea de Dirección y Economía de la Empresa*, 11-21

Friedmann, J. (1987), Transactive Planning, *Planning in the Public Domain: From Knowledge to Action*

Bastels A. Gasto global en TI crecerá 6.2% en 2014, *Forrester Research* (2013).

Reporte de Secretaría de Economía. (2013), Servicios de Tecnologías de la Información y Software. *Secretaría de economía*.

International Data Consulting (IDC) (2015), Predicciones para Latinoamérica 2015, 2015

Brown C. L. *Definición de Proyecto de Software*, 2010

Calidad (Anónimo), Las cuatro P en el desarrollo de software: Personas, Proyecto, Producto y Proceso, 2011.

Suárez J., Relaciones entre organizaciones y Stakeholders: *Necesidad de una interacción mutua entre los distintos grupos de interés*. En CNC Contaduría Pública, 2009

Vargas de la Torre, C.A. , *Diagnóstico de Procesos de Desarrollo de una Fábrica de Software*, UNAM, Facultad de Ingeniería, 2009

Fábrica de Software Deloitte México, *Proceso de Administración de Proyecto*, Fábrica de software Deloitte México, 2012

Suárez Tirado, J., *Relaciones entre organizaciones y Stakeholders: necesidad de una interacción mutua entre los distintos grupos de interés*, CNC, Contaduría Pública, 2009.

Definición de software, *Diccionario de la real academia española de la lengua*, 2014.

Guglielmo C., *El gasto en TI alcanzará 2 billones de dólares en 2013*, Forbes, 2013

The Standish Group, *The Chaos Manifesto, Think Big, Act Small*, CKC, 2013

Ramírez Arias, J.M., *Los modelos de madurez organizacionales y los factores críticos de éxito en la implantación de sistemas de Información*, UNAM, Posgrado de Ingeniería, 2009

Donaldson T. y Preston L., The Stakeholder theory of the corporation: concepts, evidence and implications, *Academy of Management Review*, p. 60 - 74, 1995.

Crow, M.A., *Identificar Stakeholders ¿Por qué molestarse en esto?*, PMBOK, 2013.

Gonzalez, J. C., *Crear una empresa, Plan de Negocios, Tipos de Estructura Organizacional*, 2008

Friendle Business, *Nuevas ideas para nuevos tiempos, ¿Cómo encontrar la estructura ideal en las organizaciones del siglo XXI?*, 2007

Riaza Fernando, *La transmisión del conocimiento*, Ámbitos, 2002

Friedmann, John, *El estilo de la planeación transactiva*, Planeación Transactiva, 1987

Ackoff Russell L., *Principio holístico*, Planeación interactiva, 1974

Palacios H., Porcell N., (2012), *Modelo CMMI*, Obstáculos al implantar el modelo CMMI, (2012).

Peralta Mario L., *Asistente para la evaluación de CMMI-SW*, Proyecto de tesis de magister en software, (2011).

Navas López, J. E.; Guerras Martín, L.A. (2003): *Casos de Dirección Estratégica de La Empresa*", Thomson-Civitas, Madrid 3° edición.

Navas López, J. E.; Guerras Martín, L. A. (2007): *La Dirección Estratégica de La Empresa: Teoría y Aplicaciones*", Thomson-Civitas, Madrid 4° edición.

Cavazos, E. (2008): *Los 4 factores de éxito en un proyecto de TI*, Gravatar, (2008)