



UNIVERSIDAD AUTÓNOMA METROPOLITANA

Maestría en Economía

Campo de conocimiento: Empresas Finanzas e Innovación

**"Innovación en empresas basadas en software libre:
cuatro estudios de caso."**

Trabajo Terminal que presenta: Graciela Zárate García

Asesor: Jordy Micheli Thirión

México DF., a 15 de Marzo de 2012

AGRADECIMIENTOS

Este trabajo está dedicado a mis padres Ciria y Luis que inspiraron este trabajo mediante su ejemplo de disciplina y constancia.

Quiero agradecer a mi asesor Jordy Micheli por escuchar mis inquietudes de manera paciente y ayudarme a convertirlas en esta investigación.

Igualmente expreso mi agradecimiento a Daniel Ceballos presidente de la Asociación Mexicana Empresarial de Software Libre A.C (AMESOL) por su enorme apoyo en la realización de este trabajo. Así como a los empresarios entrevistados que respondieron de manera entusiasta mis preguntas.

También debo mencionar a Miguel Solís, Edgar Buenrostro y Rubén Oliver quienes me proporcionaron información valiosa para enriquecer esta investigación. De igual manera a los revisores que con sus comentarios me permitieron mejorar el documento.

Agradezco a todas aquéllas personas que me acompañaron en la realización de este trabajo; los amigos que con sus palabras me impulsaron cuando tuve dudas, a los compañeros de la maestría que compartieron conmigo dos años de estudio, desvelos y muchas satisfacciones, a los profesores de los cuáles tuve la fortuna de aprender muchas cosas.

Finalmente al Consejo Nacional de Ciencia y Tecnología así como a la Universidad Autónoma Metropolitana por la oportunidad que me brindaron de realizar esta Maestría.

Introducción.....	1
1. Innovación.....	4
1.1 Definición de innovación.....	5
1.2 Modelos de innovación.....	7
1.2.1 El papel del conocimiento en la innovación.....	9
1.2.1.1 <i>Sticky information</i>	11
1.3 Innovación y gestión del conocimiento.....	12
1.4 Innovación motivada por el usuario.....	14
1.4.1 <i>Democratizing innovation</i>	15
1.4.2 Usuarios líderes.....	15
1.4.3 Motivación de los usuarios.....	18
2. Características del software libre.....	21
2.1 Diferencia entre Software libre y <i>Open Source Software</i>	25
2.2 Forma de producción del FLOSS.....	29
2.3 Motivación de los programadores.....	32
2.4 Liderazgo en el software libre.....	36
2.5 Modelos de negocio.....	37
2.6 Gestión de la propiedad intelectual.....	44
2.7 La industria del software en México.....	46
3. Estudios de Caso.....	49
3.1. Perfil del empresario.....	49
3.2 Perfil de la empresa.....	49
3.3 Motivación para participar en software libre.....	55
3.3.1 Motivación de los empresarios.....	55
3.3.2 Motivación de los colaboradores.....	56
3.4 Proceso de innovación.....	57
3.5 Relaciones con los usuarios.....	62

3.6 Protección del conocimiento.....	63
3.7 Formas de comunicación e intercambio de información.....	64
4. Conclusiones.....	65
Bibliografía.....	72
Anexos.....	75

Índice de Figuras

Figura 1. Modelo 1 de Schumpeter de la innovación empresarial.....	7
Figura 2. Red de interacción del software abierto.....	25
Figura 3. Business Canvas Model.....	40
Figura 4. Proceso de respuesta a las oportunidades de negocio para la empresa Roja	60
Figura 5. Proceso de respuesta a las oportunidades de negocio para la empresa Verde	60
Figura 6. Proceso de respuesta a las oportunidades de negocio para la empresa Amarilla.....	61
Figura 7. Proceso de respuesta a las oportunidades de negocio para la empresa Azul.....	61

Índice de Tablas

Tabla 1. Modelos de negocio de software libre.	41
Tabla 2. Estrategias de negocio de las empresas entrevistadas.....	50
Tabla 3. Áreas que conforman las empresas.....	52
Tabla 4. Formas de motivar a los colaboradores.....	54
Tabla 5. Motivación de los colaboradores a innovar.....	57
Tabla 6. Tipos de clientes para la empresa Verde.....	62
Tabla 7. Formas de protección del conocimiento.....	63

RESUMEN

La innovación es un proceso de vital importancia para las organizaciones ya que constituye una forma de creación de ventaja competitiva y generación de riqueza. En este entorno han surgido empresas que utilizan el software libre como base para la generación de beneficios económicos. El presente trabajo muestra mediante cuatro estudios de caso que las empresas que operan bajo el esquema de software libre, están motivadas por razones ideológicas así como económicas y que establecen estrechas relaciones de colaboración con su entorno, ya sean clientes o competidores, compartiendo gran parte del conocimiento generado, mientras al interior mantienen estructuras jerárquicas tradicionales.

Palabras clave: innovación, software libre, usuarios líderes, producción por pares, motivación.

ABSTRACT

Innovation is a vital process for organizations as it is a way of creating competitive advantage and wealth. In this environment, companies have emerged using free software as the basis for generating economic benefits. The present work shows through four case studies that companies that operate under the scheme of free software are motivated by ideological and economic reasons and establish close working relationships with its environment, customers and competitors, by sharing much of the knowledge generated, while kept inside them traditional hierarchical structures.

Key words: *innovation, free software, lead users, peer production, motivation.*

Introducción

La importancia de la innovación es reconocida debido a que constituye una fuente en la creación de ventajas. Las empresas tienen altos incentivos para innovar con objeto de diferenciarse de sus competidores, lo cual según Porter (1996) les otorgará una ventaja competitiva. Actualmente los ciclos de vida de los productos se han acelerado en comparación con lo que ocurría hace cincuenta años y en algunos sectores la innovación constante es necesaria para sobrevivir en el mercado. Ejemplo de ello es la industria del software que presenta un dinamismo creciente, por lo tanto muchos países se han planteado la generación de software como estrategia de posicionamiento mundial. Esta industria dentro de las Tecnologías de la información presenta algunas características que la diferencian de otros sectores.

Algunas de esas particularidades son:

1. La [Industria del software] ISW es una industria transversal. Eso significa que sus productos y servicios son susceptibles de ser utilizados en otras cadenas productivas.
2. La industria presenta también un proceso de estratificación asociado a una diferenciación entre servicios de alto valor agregado y otros de menos nivel.
3. En la producción de software es notorio el impacto de factores que determinan la necesidad de distintos niveles de proximidad geográfica entre las empresas que prestan los servicios (ya sea que colaboren entre sí o no), y entre ellas y los usuarios de sus servicios.

(Arechavala, Montoya, Rodríguez y Sampedro, 2010, pp 30-31).

La generación e intercambio del conocimiento juegan un papel importante en el surgimiento de la innovación. El caso de la industria del software presenta una peculiaridad debido a que "... las empresas de software presentan procesos de aprendizaje más centrados en el cliente y los competidores (Isaksen y Holmes, 2006)." (Citado por Díaz en Hualde, 2010, p 105). Si bien se ha considerado que el aprendizaje organizacional proviene de diversas fuentes, existe la idea generalizada de que la innovación tiende a ocurrir con mayor frecuencia en los

entornos científicos y académicos, o dentro de los organismos de investigación y desarrollo (I+D) al interior de las empresas. Por lo tanto, resulta interesante, entender de qué manera las empresas internalizan la información así como los conocimientos que obtienen de sus clientes y los transforman en innovación.

“El *Open Source Software* (OSS) es un producto público producido de manera privada y cuyo código fuente puede ser descargado de internet y utilizado gratuitamente. En teoría, no debería haber muchos productos comerciales de software libre y su desarrollo debería ser más bien limitado, como estrategia del OSS por ir contra la lógica económica mediante la no explotación del marco legal para generar utilidades de mercado de la misma forma que los productos propietarios. Sin embargo, el número de proyectos OSS (tanto voluntarios como comerciales) aumenta continuamente, y va desde los centrados en pequeñas utilidades y controladores de dispositivo a los dirigidos a desarrollar paquetes grandes y complejos, tales como Apache, *Open Office*, y MySQL. El OSS ha demostrado ser una opción viable de innovación y producción de software: algunos proyectos de software libre capturan importantes cuotas de mercado de sus competidores comerciales, introduciendo nuevas características y un rendimiento superior.” (Harrison y Koski, 2010, p1).

El objetivo de este trabajo es analizar la forma en que las empresas innovan en la industria del software, específicamente software libre. Así como saber, mediante estudios de caso realizados a cuatro empresas, de qué forma se da la estrecha relación existente entre las empresas con sus clientes y competidores.

La hipótesis que se propone es que se presentan estrechas interacciones con sus usuarios caracterizados por la ideología del software abierto que promueve el libre intercambio de conocimiento, por lo que sus modelos de negocio y formas de producción se basan en la forma en que comparten dicho conocimiento.

El primer capítulo explora el concepto de innovación, en que consiste y proporciona algunos modelos. También en este capítulo se aborda el concepto de usuario líder que es necesario para comprender la forma en que el software libre

opera, así como las motivaciones de los usuarios para involucrarse en proyectos abiertos.

El capítulo dos aborda una descripción que permite entender en que consiste el software libre, sus diferencias con el software propietario así como las diferentes concepciones que se dan al interior del mismo movimiento de software libre. Una de las características de este tipo de software es su forma de distribución la cual se regula mediante distintos modelos de licencia establecidos por los propios participantes del sector, por lo tanto también se hace mención a las licencias más comunes para comprender como funcionan. Por otro lado, el surgimiento del software libre ha generado modelos de negocio distintos a los utilizados por el software propietario que obtiene sus beneficios económicos del licenciamiento del propio software, en contraposición el software libre vende servicios alrededor del código, obteniendo beneficios de las actividades relacionadas. De manera enunciativa se enlistan diversas formas de comercialización que los desarrolladores presentan y utilizando estos modelos se realiza el análisis de los estudios de caso.

En el capítulo tres se muestran los resultados de las entrevistas realizadas a cuatro empresas de la industria del software contactadas a través de la Asociación Mexicana de Software libre. Con estas entrevistas se buscó conocer la forma en que operan las empresas, cómo se relacionan con sus clientes, competidores y empleados así como la forma en que innovan en su interior. Los resultados son analizados de acuerdo al marco establecido en los capítulos uno y dos de manera que podemos analizar el comportamiento de las empresas y compararlo con la teoría propuesta. Finalmente se muestran las conclusiones a partir del planteamiento teórico y de las entrevistas realizadas; así como las entrevistas y el anexo metodológico.

1. Innovación

Fagerberg (2003) menciona que existen muchas diferencias entre los conceptos de invención e innovación, para que un invento se convierta en innovación hacen falta, entre otras cosas, que existan tecnologías complementarias así como un mercado capaz de adoptarlo, como ejemplo Fagerberg menciona el avión ideado por Leonardo Da Vinci, el cual al no contar con las condiciones antes mencionadas no se convirtió en innovación y quedó como un mero invento. Asimismo, tanto la invención como la innovación son procesos continuos. “Por lo tanto, lo que consideramos como una innovación individual es a menudo el resultado de un largo proceso de participación de muchas innovaciones interrelacionadas. Esta es una de las razones por las que muchos estudiantes de tecnología e innovación encuentran natural aplicar una perspectiva de sistemas en lugar de centrarse exclusivamente sobre invenciones individuales e innovaciones” (Fagerberg, 2003, p 4). En este trabajo de igual manera se abordará el fenómeno de la innovación de manera sistémica identificando a los agentes involucrados así como las relaciones que se dan entre ellos para que este fenómeno ocurra. Sin embargo, el énfasis se dará en observar las relaciones de las empresas con sus usuarios, los cuales están constituidos por sus clientes, colaboradores e incluso competidores.

De igual manera la innovación ha sido considerada de manera errónea cómo un proceso aleatorio, que ocurre de manera sorpresiva dentro la de la cabeza de algún genio, sin embargo, esta idea ha sido modificada, ya que la innovación ahora se considera un proceso que puede llevarse a cabo de manera intencional y planeada sistemáticamente, así lo menciona Peter Drucker:

“La mayoría de las innovaciones, sin embargo, especialmente las exitosas, son resultado de una búsqueda decidida y consciente de oportunidades de innovación, las cuales se encuentran en sólo pocas situaciones.

Cuatro áreas de oportunidad de este tipo existen en una compañía o industria:

- ocurrencias inesperadas

- incongruencias
- necesidades de proceso
- cambios en el mercado y la industria

Tres fuentes adicionales de oportunidad existen fuera de la compañía en su ambiente social e intelectual:

- cambios demográficos
- cambios en la percepción
- nuevo conocimiento

Verdaderamente, estas fuentes se superponen, a pesar de lo distintas que puedan ser en la naturaleza de su riesgo, dificultad y complejidad, y el potencial de innovación puede bien residir en más de un área a la vez. Pero juntas, ellas cuentan para la gran mayoría de todas las oportunidades de innovación.” (Drucker, 1998, p4)

La innovación sistemática es utilizada con objeto de generar nuevos desarrollos que satisfagan las necesidades de los clientes, creen un nuevo producto y generen desarrollo tecnológico así como beneficio económico y social.

1.1. Definición de innovación

La innovación se relaciona con la originalidad, con la invención, también se dice que alguien es innovador si hace las cosas de manera no convencional. Por ello resulta frecuente confundir una invención con innovación. Morris (2003) ofrece una definición bastante útil y clara acerca de lo que constituye una innovación

“... la palabra innovación se refiere a un atributo, un proceso y un resultado. Es un proceso que ocurre en algún lugar de la compañía o quizás en la mente de alguien. El resultado en cualquier caso puede ser una intuición, una idea, un producto, una estrategia, quizás un nuevo proceso de negocio... pero lo que sea que sea, una de las cualidades que distinguirá la nueva cosa es su grado de novedad, este grado de novedad se refiere a su diferenciación, originalidad, quizás utilidad y lo más importante a su valor.

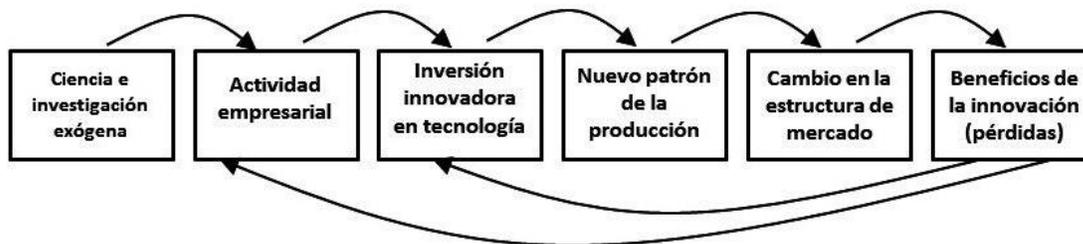
Para ser considerada una innovación en los negocios, el resultado debe ser el aumento del valor en forma de funciones nuevas o mejoradas, reducción de costos, un aumento de precio (bueno para el vendedor), una disminución del precio (bueno para el comprador), más margen para el vendedor, o alguna combinación de estos.” (Morris, 2003, p 7).

En el caso del software, las innovaciones están vinculadas también al desarrollo del hardware, el cambio de los lenguajes de programación está vinculado a los cambios en los medios de proceso y almacenamiento. En el caso del software definir una innovación en los términos antes expuestos resulta complicado, más aún en el caso del software libre. Como se mencionó anteriormente, la construcción del software se da de manera modular, de forma que las contribuciones particulares pueden tener distintos alcances, pueden ser radicalmente innovadoras, o una simple mejora a lo ya existente, a ello hay que sumar el impacto comercial de la innovación. Tal y como Morris menciona, es importante el valor que aporta al productor o el usuario. Para el uso del presente trabajo se considerará innovación cualquier construcción que permita a los empresarios obtener beneficios económicos derivados ya sea de la venta del software en sí, o de la prestación de los servicios relacionados, lo cual se observa con más detalle en la sección 2.4 que habla de modelos de negocio. Sin embargo, se considera el tiempo invertido en desarrollar dicha construcción como parte del proceso de innovación, debido a que son las pruebas de ensayo y error las que permiten obtener las soluciones que permitirán obtener beneficios económicos. Hay que considerar también la facilidad para replicar el software lo que permite múltiples aplicaciones de un código y reduce considerablemente los costos de producción, los cuáles no están directamente vinculados a la cantidad de aplicaciones en que se utilice. Así, un código generado para un proyecto específico puede servir para otros proyectos generando una renta mayor, o puede estar limitado a un proyecto único.

1.2. Modelos de innovación

La innovación es un concepto que ha ido evolucionando a lo largo del tiempo y se ha visto afectada también por el desarrollo tecnológico en tanto concepto como en la forma de llevarse a cabo. Diversos autores han hecho esfuerzos por mostrar cómo ocurre y quiénes son los agentes que participan en ella. En primer lugar podemos mencionar a Schumpeter (1912) y su modelo en el cuál reconoce un papel preponderante al emprendedor, el cual motivado por los beneficios económicos actúa como agente catalizador de la dinámica de innovación.

Figura 1. Modelo 1 de Schumpeter de la innovación empresarial



Fuente: Aleixandre. *Las estrategias para la innovación tecnológica en Castilla y León*. (2002). P 58.

Según este modelo el origen de la innovación se encuentra en la ciencia y tecnología que no forman parte de la empresa. Cabe recordar que en el tiempo en que este modelo fue desarrollado, era poco común que las empresas tuvieran departamentos de investigación y desarrollo en su interior. El conocimiento era un bien exógeno a la empresa. Posteriormente se han desarrollado modelos más complejos que buscan integrar las diversas variables y mayores participantes, tanto dentro como fuera de la empresa. El modelo más simple es el denominado modelo lineal “En este modelo, uno hace investigación, la investigación entonces lleva al desarrollo, el desarrollo a la producción, y la producción al marketing” (Kline y Rosenberg en Landau & Rosenberg, 1986, p 285). Otra forma de describirlo la proporcionan Escorsa y Valls “...el proceso empieza con la investigación básica, pasa por la investigación aplicada y el desarrollo tecnológico

y acaba con el marketing y el lanzamiento al mercado de la novedad.” (Escorsa y Valls, 2003, p 26). Estos componentes los explican los mismos autores de manera más detallada.

“La investigación básica comprende todos aquellos trabajos originales que tienen como objetivo adquirir conocimientos científicos nuevos sobre los fundamentos de los fenómenos y hechos observables. Dentro de este tipo de trabajo se analizan propiedades, estructuras y relaciones y su objetivo consiste en formular hipótesis, teorías y leyes. Los resultados se publican en revistas bastante especializadas y no pretenden lograr ningún objetivo lucrativo en concreto.

La investigación aplicada consiste en trabajos originales que tienen como objetivo adquirir conocimientos científicos nuevos, pero orientados a un objetivo práctico determinado. Está ligada a la investigación básica a causa de que utiliza posibles resultados de la investigación básica y estudia métodos y medios nuevos para lograr un objetivo concreto. Los resultados que se obtienen son los productos determinados una gama de productos nuevos o, incluso, un número limitado de operaciones, métodos y sistemas. Los resultados son susceptibles de ser patentados.

El desarrollo tecnológico abarca la utilización de distintos conocimientos científicos para la producción de materiales, dispositivos, procedimientos, sistemas o servicios nuevos o mejoras substanciales. Realiza trabajos sistemáticos basados en conocimientos existentes, procedentes de la investigación aplicada o de la experiencia práctica. Su primer objetivo consiste en lanzar al mercado una novedad o una mejora concreta. Para poder ensayar, normalmente se hacen pruebas con un prototipo o una planta piloto; actualmente, sin embargo, se tiende de forma creciente a la simulación por ordenador.” (Escorsa y Valls, 2003, p 23 y 24).

Esta idea recuerda una línea de ensamblaje, ya que se trata de un proceso encadenado de etapas sucesivas y donde cada parte del proceso es independiente del anterior. Los sucesivos participantes agregan información o materiales y generan un producto terminado. Sin embargo esta visión de la innovación deja de lado fenómenos donde los participantes se encuentran jugando

papeles que supuestamente no les corresponderían. Desde esta perspectiva los clientes son meros consumidores a la expectativa de los productos que las empresas crean, a los cuáles se les convence de los beneficios de un producto nuevo a través de herramientas de marketing. Otra de las implicaciones de este modelo es que lo que los productores desarrollan satisface del todo los requerimientos de los clientes. Lo cual en realidad bastante difícil ya que no se conocen en su totalidad las necesidades de los clientes, además de que intentar satisfacer todos sus requerimientos puede resultar costoso. Por otra parte se desconocen también los usos alternativos que los usuarios pudieran hacer del producto, sobre todo al tratarse de productos innovadores. Existen casos de productos que fueron creados con un fin, pero los usuarios les encontraron un uso para el cual resultaban más idóneos, por ejemplo los pañuelos desechables fueron creados para desmaquillar y los usuarios los utilizaron para sonarse la nariz, que es el uso convencional ahora.

1.2.1. El papel del conocimiento en la innovación

El conocimiento forma parte de la tecnología de la empresa, como lo reconocen Bell y Pavitt citados por Yoguel (2000). Sin embargo, este conocimiento no es homogéneo y tiene características específicas que afectan su valor y la facilidad con que se transmite. Tomando la clasificación hecha por Lundvall y Jhonson (1994) Enrique Medellín la presenta así:

- *Saber qué (Know-what)* se refiere al conocimiento acerca de los “hechos”. Es un conocimiento cercano a lo que se denomina normalmente información.
- *Saber por qué (Know-why)* se refiere al conocimiento acerca de los principios y leyes del movimiento en la naturaleza, la mente humana y la sociedad, y el cual es la base del desarrollo tecnológico y de avances en productos y procesos en ciertas áreas de base científica, tales como la química, la eléctrica y la electrónica. La producción y reproducción de este tipo de conocimientos se distribuye a menudo en organizaciones especializadas, tales como laboratorios de investigación y universidades.
- *Saber cómo (Know-how)* se refiere a las habilidades o a la capacidad para hacer algo. Es una forma típica de conocimiento desarrollado y mantenido

dentro de los límites de una empresa individual. Una de las razones más importantes para la formación de redes es la necesidad que tienen las empresas de compartir y combinar elementos de este tipo de saberes.

- *Saber quién (Know-who)* se refiere a la información acerca de quién sabe algo y quién sabe hacerlo. Involucra la formación de relaciones sociales especiales que permiten el acceso a expertos y el uso de sus conocimientos de manera eficiente.

A estos tipos Medellín agrega dos más propuestos en un trabajo de 2003:

- *Saber cuándo (Know-when)* que se refiere a la capacidad de actuación oportuna para la toma de decisiones y el logro de resultados. Implica experiencia en la práctica de formulación y ejecución de programas y proyectos, el conocimiento del ciclo de vida de los productos y servicios, así como de las condiciones culturales de la empresa y su entorno.
- *Saber dónde (Know-where)* que se refiere a la capacidad para determinar el lugar o espacio más adecuado para realizar algo. No es suficiente contar con información sobre estructuras de costos, condiciones sociales y políticas, características del sitio, etc., sino que es necesario además determinar cuál es la mejor combinación de recursos para tener éxito en ese espacio específico y en un momento dado. Exige conocimiento sobre los nuevos modelos de negocio que replantea, por ejemplo, la forma de administración de las cadenas de suministro, el desarrollo de nuevas estrategias de integración vertical y horizontal y la participación en *clusters* regionales. (Medellín, 2008, p 60, 61 y 62)

Los dos primeros tipos (*Know-what* y *know-why*) se clasifican a su vez dentro del conocimiento explícito o codificado (Yoguel, 2000), cuya principal característica es que al estar codificado es transferible, mientras que el tácito no. El tácito, integrado por los otros tipos, se caracteriza por “La importancia que adquieren los elementos tácitos del conocimiento en el nuevo contexto para el éxito a largo plazo de los agentes, que se manifiesta en la capacidad de adaptarse al cambio (flexibilidad) y en la capacidad de imponer cambios (innovación)...” (Yoguel, 2000, p 111). Este conocimiento además es difícilmente transferible.

1.2.1.1. *Sticky information*

La interacción con los usuarios no es un proceso sencillo debido a que las empresas cuentan con sus propias estructuras, códigos, lenguaje y prácticas los cuales conforman su conocimiento, el cual es reconocido como una fuente de ventaja competitiva (Yoguel, 2000). Por lo tanto, surge un problema al determinar la cantidad de conocimiento interno que se comparte con los usuarios para generar las innovaciones esperadas, de igual forma resulta complejo determinar la forma en se hará este intercambio para que sea efectivo. Del lado del usuario existen otras limitantes. Según la tipología antes expuesta, el conocimiento que poseen los usuarios es básicamente tácito y difícil de transferir a los productores por ello que existe información que se pierde en el intercambio. Von Hippel denominó a este fenómeno *sticky information*. Mientras más complejos y específicos sean los requerimientos de los usuarios la información será más difícil de transmitir.

“Una consecuencia importante de la rigidez de la información es que da lugar a las asimetrías de información que no se pueden borrar fácilmente o de forma barata. Diferentes usuarios y fabricantes tendrán distintas cantidades de información, y puede resultar costoso adquirir la información que necesitan, pero no tienen. Como resultado cada innovador tenderá a desarrollar las innovaciones basadas en la información pegajosa que ya tiene, porque ese es el curso de acción más barato (Arora y Gambardella, 1994; Von Hippel, 1994). En el caso específico del desarrollo de productos, esto significa que los usuarios, como clase, tienden a desarrollar innovaciones que se basan en gran medida en su propia información acerca de la necesidad y el contexto de uso. Del mismo modo, los fabricantes, como clase, tienden a desarrollar innovaciones que se basan en gran medida en los tipos de información de la solución en el cual se especializan” (Von Hippel, 2005, p 70).

Se ha buscado obtener este conocimiento mediante herramientas como: entrevistas, encuestas y algunas más sofisticadas tales como el *Quality Function Deployment* (QFD) o la Teoría para resolver problemas de inventiva (TRIZ), sin embargo, una gran parte del conocimiento de los usuarios sigue permaneciendo

inalcanzable para las empresas. Por ello resulta tan importante la participación del usuario en sectores como el software. De acuerdo a los resultados de Von Hippel, las contribuciones de los usuarios contienen mayor novedad que las realizadas por los productores, pero las de estos últimos son mejores en cuanto a su funcionalidad y eficacia. Para el caso del software libre hay que recordar que se requiere tener conocimientos técnicos para poder participar en su construcción, ello permite que sus usuarios-colaboradores posean información que podría ubicarse en ambos lados de la ecuación tanto como usuarios o productores.

1.3. Innovación y gestión del conocimiento

“Gestión del conocimiento es un proceso sistémico para adquirir, organizar y comunicar conocimientos tácitos y explícitos de forma que todos los empleados puedan usarlos para ser más efectivos y productivos en su trabajo (Alavi y Leidner, 1997)

Gestión del conocimiento es el proceso que continuamente asegura el desarrollo y aplicación de todo tipo de conocimientos pertinentes en una empresa, con objeto de mejorar su capacidad de resolución de problemas y así contribuir a la sostenibilidad de sus ventajas competitivas (Andreu y Sieber, 1999).

La gestión del conocimiento es un proceso sistemático e integrador de coordinación de las actividades de adquisición, creación, almacenaje y difusión del conocimiento por individuos y grupos con objeto de conseguir los objetivos de la organización (Rastogi, 2000).”(Escorsa y Valls, 2003, p 46).

Escorsa y Valls mencionan tres razones por las cuales es importante innovar.

1. El progreso técnico
2. La internacionalización de la economía
3. La desmasificación de los mercados

Llevar a cabo este proceso de innovación requiere inversión de recursos y un esfuerzo consciente por parte de la empresa, en la teoría ha surgido el término gestión de la innovación para hacer referencia a la forma en que las empresas

realizan las distintas actividades encaminadas a realizar innovación. Citando a Escorsa y Valls tenemos

“... la *gestión de la innovación*, que incluye la gestión de la I+D, pero añadiéndole otros aspectos como el lanzamiento de los nuevos productos o el estudio de las razones de su éxito o fracaso, que no figuran normalmente en el área de la gestión de la I+D” (Escorsa y Valls, 2003, p 46).

“En un entorno competitivo, sobre todo en el mercado de las TIC’s, las empresas se ven en la necesidad de diseñar estrategias para abordar el tema de la innovación, aun cuando no estén conscientes de ello, o adopten una actitud más bien pasiva y esperen a tener mayor información para tomar mejores decisiones. Un buen ejemplo de las distintas estrategias que utilizan las empresas es la clasificación que otorga Christopher Freeman (1974), de la Universidad de Sussex:

- a) *Ofensiva*: pretende conseguir ser el líder técnico, colocándose a la cabeza de los competidores en la introducción de nuevos productos.
- b) *Defensiva*: no quiere asumir el gran riesgo de ser el primero en innovar pero tampoco se quiere quedar atrás. Espera sacar ventaja de los errores de los otros y mejorar sus diseños.
- c) *Imitativa*: se contenta con ir detrás de los líderes en las tecnologías establecidas ya hace tiempo. Suele adquirir licencias y *know-how*.
- d) *Dependiente*: comporta la aceptación de un papel satélite o subordinado. Los nuevos productos responden a iniciativas y especificaciones externas, por ejemplo, de la empresa matriz.
- e) *Tradicional*: el mercado no suele pedir cambios en los productos. La empresa se limita a la adopción de las innovaciones de proceso generadas desde fuera de la empresa.
- f) *Oportunista*: identifica una nueva oportunidad o un diseño complejo que le permite prosperar en algún nicho.

Urban y Hauser (1980), que distinguen, como hacen otros autores, entre estrategias reactivas y proactivas.

Estrategias reactivas: responden a las demandas del mercado y a las actividades de los competidores.

- a) *Sensible a los clientes*. La empresa reacciona a las peticiones de innovaciones de sus clientes.
- b) *Imitativa*. La empresa reacciona a los nuevos productos imitándolos.
- c) *Segunda pero mejor (second-but-better)* La empresa desarrolla y mejora las innovaciones de los competidores.
- d) *Defensiva*. Reacciona ante las innovaciones de los competidores modificando sus productos, más que innovando con productos nuevos.

Estrategias proactivas: prevén y se anticipan al mercado y los cambios tecnológicos.

- a) *Basada en la I+D (R&D based)*. Las innovaciones están basadas en la I+D.
- b) *Emprendedora*. La actividad innovadora presenta alto riesgo y es oportunista, aunque no necesariamente en los aspectos tecnológicos.
- c) *Adquisitiva*. Las innovaciones son llevadas a término mediante la compra de nuevos productos o compañías.
- d) *Basada en el marketing (marketing-based)*. Marketing tiene la iniciativa de desarrollar y lanzar productos innovadores.” (Escorsa y Valls, 2003, p 81 y 82).

1.4. Innovación motivada por el usuario

La literatura reconoce la influencia de los usuarios en la mejora de los productos buscando adaptarlos a sus necesidades particulares. Von Hippel (2005) menciona que las necesidades de los usuarios son heterogéneas y que en función de sus requerimientos, tendrán incentivos a pagar por lo que ellos exactamente quieren. En el caso del software se trata de un producto destinado a satisfacer necesidades muy concretas, ello podría motivar el involucramiento del consumidor en el desarrollo de una solución que se ajuste a sus necesidades específicas. También es importante considerar el uso que se le dará al software ya que un mismo cliente tiene requerimientos con demandas de especificación distintas. En caso de que el cliente sea una empresa, quizás pueda utilizar un software estandarizado para la realización de aquellas tareas rutinarias y menos especializadas, pero en el caso de soluciones del proceso de producción, o de actividades que constituyen la actividad medular de la empresa, sus requerimientos serán muy específicos, y

mientras mejor diseñadas estén las soluciones de software serán más eficientes. Así por ejemplo, para una empresa dedicada a la compraventa de juguetes, quizás su programa de contabilidad puede ser genérico, no así para una empresa que se dedica a la consultoría contable. En un caso no se trata de la actividad primordial de la organización, mientras que para la otra el tener un sistema cuidadosamente diseñado puede constituir una ventaja competitiva.

1.4.1. Democratizing innovation

Von Hippel acuñó el término *Democratizing innovation* para referirse a los casos en que la innovación no es exclusiva de los productores, aquéllos que obtienen beneficio de la venta de un producto, sino que también involucra una participación activa de los consumidores ya sea que se trate de personas o empresas (Von Hippel, 2005). Este fenómeno también provoca, desde su punto de vista, que aumente el bienestar social. Para el caso específico del software el ejemplo más claro de este comportamiento lo constituye el software libre (*Free Software*) o el software abierto (*Open Source Software*) uno de cuyos máximos exponentes es el sistema operativo Linux. El corazón de este sistema operativo fue generado sin ser terminado totalmente por Linus Torvalds quien hizo público el código para que los usuarios lo modificaran y terminaran, en un proceso de mejora constante en el cual existen muchos participantes trabajando en distintas partes del mismo. Frente a esta forma de producción grupal surge un nuevo problema: de qué manera se obtienen beneficios por participar en la producción del software libre. Este problema se analiza en el capítulo dos donde se describe la forma en que operan las empresas que se basan en software libre.

1.4.2. Usuarios líderes.

Von Hippel reconoce que no todos los usuarios se involucran en los procesos de innovación, aquéllos que lo hacen son denominados usuarios líderes (*lead users*). Para ilustrar la innovación centrada en el usuario “El modelo centrado en la manufactura encaja en algunos campos y condiciones. Sin embargo, un creciente cuerpo de trabajos empíricos muestra que los usuarios son los primeros en

desarrollar muchos y quizás la mayoría de los nuevos productos industriales y de consumo. El cambio de curso de la innovación hacia los usuarios tiene muchas cualidades atractivas. Se está volviendo progresivamente más fácil para muchos usuarios obtener precisamente lo que ellos quieren diseñándolo ellos mismos. Y la innovación hecha por usuarios parece incrementar el bienestar social. Al mismo tiempo, el cambio de las actividades de desarrollo del producto de los productores a los usuarios es doloroso y difícil para muchos productores” (Von Hippel, 2005, p 2). Cuando el autor menciona que la innovación centrada en la manufactura encaja en ciertos campos y condiciones, esto conlleva implícitamente la idea de que la innovación basada en el usuario ocurre en ciertos entornos con características específicas que permiten que ocurra.

La importancia de los usuarios en la innovación va más allá de reducir los costos de agencia generados cuando los productores intentan comprender las necesidades de los primeros.

“Estudios empíricos muestran que muchos usuarios –del 10 a cerca del 40 por ciento- se involucran en desarrollar o modificar productos. Las correlaciones encontradas entre la innovación por usuarios y el status de usuario líder con altamente significantes y los efectos muy grandes. Un número de estudios ha mostrado que muchas de las innovaciones reportadas por usuarios líderes se juzgan ser comercialmente atractivas y/o han sido realmente comercializadas por los productores” (Von Hippel, 2005, p 4).

Esto se explicaría debido a que las necesidades manifestadas por los usuarios son compartidas por otros miembros de la población que las presentan aunque no las manifiesten, de esta manera los usuarios líderes se convierten en un poseedor de información valiosa, que precisamente por conocer necesidades ocultas, conoce también una forma de satisfacerlas mejor que la que pueda proponer el productor, hay que recordar el fenómeno de la *sticky information*. Mientras más pegajosa sea dicha información más difícil será para el productor acceder a ella y al usuario hacérsela saber.

Este fenómeno también es reconocido por Bell y Pavitt

“El progreso técnico es generado por ambos: productores y usuarios de la tecnología incorporada en el capital. Como se mencionó antes, las empresas que hacen bienes de capital no son los únicos creadores y originadores de nuevos procesos. En realidad, los usuarios de bienes de capital frecuentemente juegan roles activos y creativos al cambiar la tecnología que ellos usan. Algunos juegan roles dominantes o incluso exclusivos al desarrollar nuevas máquinas, y muchos más interactúan con los productores de maquinaria y compañías de ingeniería al hacer modificaciones incrementales o mejoras a las tecnologías de producción. Y así, estos cambios continúan a lo largo de la vida operativa de un proyecto.” (Bell y Pavitt, 1995, p75).

Sin embargo, como Von Hippel observó para que un usuario se involucre en el proceso de innovación se requieren ciertas condiciones. En la consideración de un usuario líder existen dos características que lo definen:

“(1) Ellos están a la vanguardia de una tendencia importante en el mercado, y por eso están experimentando actualmente las necesidades que serán experimentadas más tarde por muchos usuarios en ese mercado. (2) anticipan relativamente altos beneficios de obtener una solución a sus necesidades, y por lo tanto podrían innovar.” (Von Hippel, 2005, p 22)

“Las dos características definitorias de los usuarios líderes y la probabilidad de que ellos desarrollen productos nuevos o modificados se han encontrado altamente relacionados (Morrison et al, 2004). Además, se ha encontrado que mientras más alta sea la intensidad de las características de usuario líder presentadas por un innovador más grande será la atracción comercial de la innovación que el usuario líder desarrolle (Franke and Von Hippel, 2003a).” (Von Hippel, 2005, p 4).

Esta es una razón más para involucrar a los usuarios líderes en el proceso de innovación, pues tendrá más posibilidades de obtener éxito comercial.

También Von Hippel en un trabajo previo habla de la importancia de estos usuarios:

“El contexto particular de nuestra prueba... fue el campo de rápida evolución de los equipos de diseño asistido por computadora utilizados para diseñar tarjetas

de circuitos (PC-CAD). Aquí encontramos que podíamos identificar a un subgrupo de usuarios que denominamos usuarios líderes. Encontramos que la actividad de innovación se concentró dentro de este grupo como se había predicho: el 87% de los encuestados en el grupo de usuarios líderes construyó su propio sistema PC-CAD frente a sólo el 1% de los que no eran usuarios líderes. También se encontró que los conceptos de producto basados en las innovaciones de esos usuarios líderes fueron preferidos por todos los usuarios y por lo tanto tenían promesa comercial desde el punto de vista de los productores de PC-CAD.” (Von Hippel, 1988, p 7)

Aquí podemos ver la importancia que poseen los usuarios líderes, así como la información que poseen. Los usuarios líderes fungen como intérpretes de las necesidades del resto de los usuarios, y al tener la capacidad de satisfacer dichas necesidades debido a sus capacidades resuelven un problema que muy probablemente los productores desconocen.

1.4.3. Motivación de los usuarios.

Los usuarios líderes saben lo que necesitan pero ¿por qué se involucrarían en la innovación? En respuesta a esta pregunta Von Hippel indica que “Los usuarios podrían innovar si ellos quieren algo que no está disponible en el mercado y tiene la capacidad y voluntad de pagar por su desarrollo” (Von Hippel, 2005, p 5). El costo puede estar representado por erogaciones monetarias, tiempo invertido, conocimiento, etc. Esta es una de las características fundamentales de los desarrolladores de software abierto, ya que este movimiento surge como respuesta a la necesidad de adaptar el software a las necesidades específicas de los usuarios. Los cuáles además cuentan con el conocimiento técnico necesario para generar el software que satisface dichas necesidades. Von Hippel también hace mención de los costos sociales en términos de Pareto que se experimentarían si más de un usuario buscar solucionar el mismo problema de manera independiente. Para el caso del software abierto este problema desaparece dada la velocidad con que se publican los hallazgos a través de los distintos medios de comunicación de los desarrolladores (Raymond, 2001).

Además de satisfacer necesidades específicas también el obtener placer juega un papel importante en la participación de los usuarios

“... muchos factores pueden conducir a los usuarios a innovar antes que comprar. En ambos casos el de empresas usuarias y de los usuarios individuales que innovan, los costos de agencia juegan un papel importante. En el caso de los usuarios individuales innovadores, el disfrutar del proceso de innovación puede también ser importante.” (Von Hippel, 2005, p 6). “El placer como motivación puede aplicar al desarrollo de aplicaciones comerciales útiles también. Estudios de las motivaciones de los contribuidores voluntarios de código ampliamente utilizado en productos de software ha mostrado que esos individuos también están fuertemente motivados a innovar por la alegría y el aprendizaje que encuentran en su trabajo (Hertel et al. 2003; Lakhani and Wolf 2005 en Von Hippel, 2005, p 8).

Otra de las causas por las cuales los usuarios se involucran en innovación es la facilidad de transferencia de la información. Utilizando nuevamente el concepto de *sticky information*

“Los desarrolladores de productos necesitan dos tipos de información para tener éxito en su trabajo: necesidad y contexto del uso de la información (generada por los usuarios) y una solución de información genérica (frecuentemente inicialmente generada por los productores especializados en un particular tipo de solución). ... Ambas necesidades de información y solución de información son frecuentemente muy “pegajosas” –esto es, que resulta muy costosa de mover del sitio donde la información fue generada a otros sitios. Como resultado, los usuarios generalmente tienen un modelo más detallado y exacto de sus necesidades que los productores, mientras que los productores tienen un mejor modelo de aproximación a la solución que la que tienen los usuarios. Cuando la información es pegajosa, los innovadores tienden a confiar ampliamente en la información que tienen almacenada.” (Von Hippel, 2005, p 8).

Para resumir, la innovación es un proceso que otorga una ventaja competitiva a las empresas. En este proceso el conocimiento juega un papel muy importante, para conocer las necesidades de los usuarios, así como para desarrollar la forma de solucionarlos, la investigación y desarrollo siguen siendo muy importantes. La

información tiene diferentes grados de complejidad que hacen que sea más fácil o difícil intercambiarla entre los distintos participantes en el proceso, mientras más difícil sea de intercambiar, más difícil será satisfacer las necesidades de los usuarios, a esta información Von Hippel la ha denominado *sticky information* o “información pegajosa” para ilustrar lo complicado de transferirla. Como respuesta a esta clase de problemas ha surgido algo que Von Hippel llama *democratizing innovation*, que consiste en la participación de otros actores distintos a los productores en el proceso de innovación, se refiere específicamente a los usuarios, los cuáles además reúnen características específicas y son denominados “usuarios líderes”. Dichos usuarios actúan como intérpretes de las necesidades de los otros usuarios y también sugieren o proporcionan soluciones específicas que pueden ser adoptadas por los productores. Presentando además la ventaja de que las propuestas ideadas por estos usuarios tienen mayores posibilidades de éxito que las generadas sin tomarlos en cuenta, ya sea únicamente por la empresa, o por instituciones externas de investigación y desarrollo. Para que estos usuarios participen las motivaciones son variadas y van desde la solución a sus problemas específicos, el placer hasta lo costoso que les puede resultar hacer llegar a los productores la información requerida para satisfacer sus necesidades.

A partir de estas características de los usuarios y la innovación centrada en ellos, el siguiente capítulo se enfoca en describir la forma en que funciona el software libre, cuyos usuarios presentan las características descritas en este capítulo. Además de que esta industria también tiene condiciones que facilitan y permiten la colaboración entre los usuarios debido a las características del producto generado, así como las formas de producción e intercambio de la información.

2. Características del software libre.

Después de analizar la innovación procedemos a explicar en qué consiste el software libre así como cuáles son las peculiaridades que presenta y que permiten que la innovación en su interior se dé de una forma no convencional, la cual está modificando no solamente la forma de producción sino también de comercialización de este producto o servicio. En principio hay que explicar qué es el software:

“... el término *software* es usado para describir las instrucciones digitales e información operativa que están contenidas en programas para guiar a las máquinas – especialmente computadoras– en la implementación de las operaciones deseadas (por ejemplo procesamiento de datos o interacción con periféricos). El software es creado de un código fuente que consiste en secuencias de afirmaciones y declaraciones escritas en lenguajes especiales de programación de computadora. El código fuente normalmente es convertido en un formato digital ejecutable usando programas compiladores especiales. Los usuarios finales generalmente obtienen sólo los archivos ejecutables; el código fuente no necesariamente será revelado a los usuarios finales.” (Lippoldt y Stryszowsky, 2009, p 33).

Además, el software presenta dos características que es necesario tomar en cuenta para comprender su forma de producción; la primera es su carácter incorpóreo: el software es un bien no físico que se almacena en diferentes soportes físicos, y cuyo costo de reproducción, una vez creado, es relativamente bajo, además de la facilidad con que se puede copiar y compartir; la segunda característica es su complementariedad con el hardware: para que el software sea útil se requiere un hardware en el cual ejecutarlo, ambos mercados se encuentran estrechamente vinculados así como el desarrollo tecnológico de dichos campos. Las características mencionadas influyen asimismo en la forma de innovar dentro del software, el software posee un desarrollo acumulativo debido a que su naturaleza incorpórea permite un fácil intercambio, reúso o modificación del mismo. Además presenta ciclos de vida más cortos que otros bienes manufacturados (Lippoldt y Stryszowsky, 2009). Estas características son compartidas por cualquier software, independientemente de la forma en que sea

producido o el hardware al que esté destinado. Existen dos principales tipos de software desde la perspectiva económica: el propietario, cuyas ganancias se basan en la venta de licencias y se caracteriza por impedir el acceso a su código fuente y, el software libre o abierto, que es el objeto de estudio de este trabajo.

“Un programa de fuentes abiertas, además, permite que de manera legal, como usuario, puedas estudiarlo y modificarlo, permite distribuir copias... si los usuarios pueden leer, modificar y redistribuir el código fuente de un programa, éste va a evolucionar, a desarrollarse, y a mejorar a una velocidad vertiginosa.” (Guía básica del Software de fuentes abiertas, 2008, p 9)

La misma guía proporciona diez características del software abierto:

- 1) La estabilidad de su sistema operativo Los sistemas operativos cerrados o propietarios, son como cajas secretas, cuyos posibles errores han sido revisados, pero esta revisión la ha realizado un grupo reducido de personas en un período limitado de tiempo. Los sistemas operativos abiertos son como cajas públicas, revisadas por un gran número de expertos de forma continua, desde que se pusieron en circulación.
- 2) Su seguridad. Cualquier persona experta podría detectar virus accediendo al código, pues es de conocimiento público y, en su caso, corregir estos problemas.
- 3) Su coste. Aunque por lo general el software de fuentes abiertas suele tener un menor coste de licencia con respecto al Software privativo (a veces incluso gratuito), el verdadero ahorro en costes se produce en la producción de dicho software. El código abierto permite a los programadores disminuir el coste de producción, ya que cuando desarrollan programas nuevos pueden aprovechar y reutilizar software ya existente, completo o incluso partes de él, en lugar de comenzar de cero la construcción del programa.
- 4) La libertad para modificarlo. Entre las razones por las que las empresas y otras instituciones están optando por el software de fuentes abiertas destaca la necesidad de adaptación. En pocas ocasiones, un paquete de Software privativo se adapta a todas las necesidades de una organización.

- 5) Su sencillez de instalación. En la actualidad las instalaciones de software abierto son, por lo general, tan sencillas de instalar como cualquier otra opción de Software privativo.
- 6) Los estándares abiertos. Los estándares informáticos permiten a los programadores entender cómo se comunica nuestro ordenador con el software que utilizamos y cómo gestiona, maneja y almacena los datos asociados. Los formatos abiertos disponen de una documentación pública que define con exactitud todos los aspectos del estándar, de forma que cualquier experto podría diseñar un programa para gestionar dichos datos.
- 7) Las ventajas a nivel jurídico. Ello se traduce en una serie de ventajas legales que dependerán del tipo de licencia bajo la cual se haya licenciado el software pero que, a nivel general, reconocen, el derecho a distribuir copias, el derecho de uso del software sin restricciones de finalidad, o la posibilidad de hacer públicas las mejoras, entre otros.
- 8) Su capacidad para favorecer la innovación colectiva. El análisis, las revisiones y las mejoras producidas en los sistemas de fuentes abiertas para mejorar su rendimiento, son llevadas a cabo por multitud de personas en todo el mundo, fruto de las necesidades y las situaciones más diversas, generando innovación de forma continua.
- 9) Los beneficios para la industria local. Se dice que el software de fuentes abiertas garantiza a las empresas la independencia del proveedor. Esto es así debido a la disponibilidad del código fuente. En el Software privativo van apareciendo nuevas versiones cada cierto tiempo que mejoran o añaden funcionalidades. Estas actualizaciones son consecuencia de un procedimiento centralizado en la empresa fabricante, derivado de un análisis económico previo hecho por dicha empresa, y sometido siempre a la rentabilidad futura de las modificaciones. En el software abierto existe también un proceso constante de actualización, derivado del contacto entre los usuarios y los desarrolladores a nivel global y local, así como del trabajo de aquellos que tienen un mayor contacto con la comunidad, discutiendo y pactando nuevas aportaciones, o subsanando errores.

- 10) Posibilidad de diferentes lenguas. En el mundo del Software privativo, si una lengua no dispone de herramientas y aplicaciones tales como correctores ortográficos, diccionarios, etc..., su posible desarrollo se decide de forma única en elementos de carácter comercial. A veces, este tipo de traducciones no resulta rentable comercialmente, y por tanto, ciertas lenguas quedan para siempre marginadas para el uso de dichas aplicaciones.

(Guía básica de software de fuentes abiertas, 2008)

El software libre se constituye por la comunidad de usuarios y desarrolladores que participan en la creación, mantenimiento, y uso del software libre.

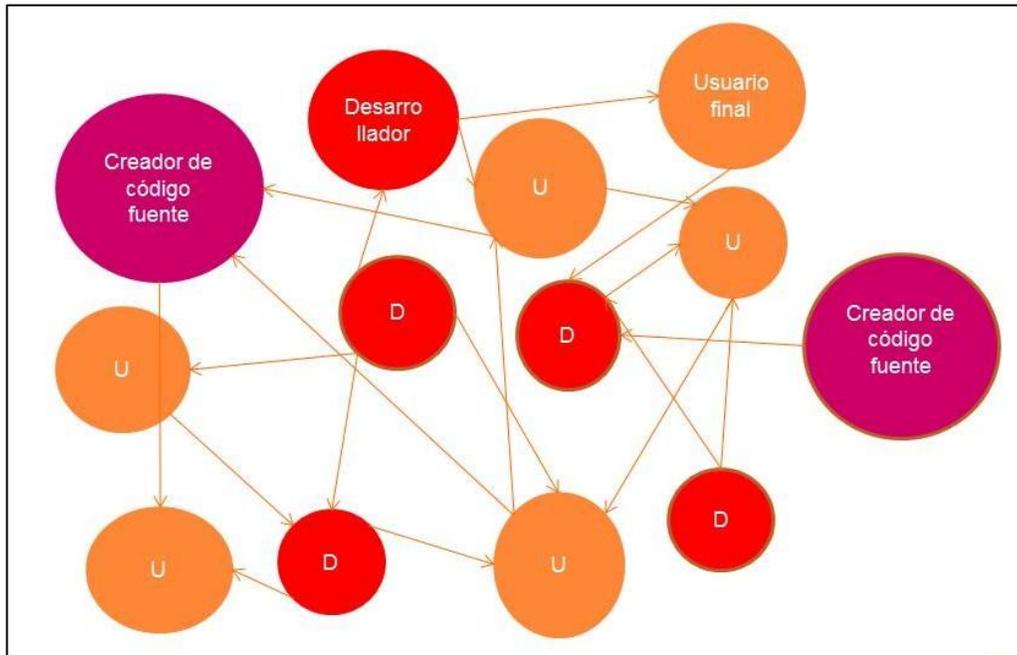
“En definitiva, podemos decir que esta comunidad se compone de dos grandes grupos de personas o instituciones:

Usuarios: son las personas o entidades que usan los programas y cuya misión es realizar tests, demandar mejoras de las funcionalidades, o informar de errores de funcionamiento del software.

Desarrolladores: son las personas o entidades que producen programas o sus modificaciones, y cuya misión es mejorar el software de fuentes abiertas tomando en consideración las aportaciones de los usuarios. Hay que tener en cuenta que todos somos usuarios y que cuando hablamos de desarrolladores, no hablamos solamente de los programadores, sino de cualquier persona que produzca cualquier tipo de medio, material o recurso que contribuya a mejorar el software de fuentes abiertas. Se puede ser desarrollador, por tanto, aportando sonidos, imágenes, iconos, desarrollando documentación y manuales, o traduciéndolos a cualquier idioma, diseñando plantillas o cualquier elemento, en general, que se pueda relacionar con un software para mejorarlo.” (Guía básica de software de fuentes abiertas, 2008, pp 22-23)

Así tenemos un esquema de red con interacciones entre todos los participantes sin tener jerarquías específicas, donde quienes desarrollan un proyecto pueden ser usuarios de otro o viceversa.

Figura 2. Red de interacción del software abierto



Fuente: Elaboración propia con información de Benckler (2002), Lerner y Tirole (2000) y la *Guía básica de software de fuentes abiertas* (2008)

2.1. Diferencia entre Software libre y *Open Source Software*

En la construcción del software libre existen dos concepciones distintas acerca del funcionamiento de este modelo, estas concepciones han sido desarrolladas por los pioneros en el desarrollo de programas mediante esta modalidad. Esta diferencia en la concepción implica la forma en que se genera, distribuye y se obtienen beneficios de su explotación, estas dos concepciones se pueden agrupar bajo los términos de Software libre y Software de código abierto.

Software Libre. Movimiento iniciado por Richard Stallman

“En 1983 el anunció el proyecto para desarrollar el sistema operativo GNU, un sistema operativo parecido a Unix destinado a ser enteramente software libre, y ha sido desde entonces el líder del proyecto. Con ese anuncio Richard también lanzó el movimiento de Software libre. En Octubre de 1985 inició la *Free Software Foundation*.” (Página de la *Free Software Foundation*, 2011)

La definición que el sitio de la *Free Software Foundation* (FSF, Fundación para el Software Libre) proporciona del software libre incluye también la concepción que tienen acerca de las razones por las que existe.

“El software libre es software que te da como usuario la libertad de compartirlo, estudiarlo y modificarlo. Nosotros lo llamamos software libre debido a que el usuario es libre.

Usar software libre es hacer una elección política y ética afirmando el derecho de aprender, y compartir lo que aprendemos con otros. El software libre ha convertido en el fundamento de una sociedad de aprendizaje donde podemos compartir nuestro conocimiento en una forma que otros pueden acumular y disfrutar.” (Lee, 2010)

“El software libre es un asunto de libertad del usuario de correr, copiar, distribuir, estudiar, cambiar o mejorar el software. Más precisamente, significa que los usuarios del programa tienen cuatro libertades esenciales:

- La libertad de correr el programa, con cualquier propósito (libertad 0).
- La libertad de estudiar cómo funciona el programa, y cambiarlo como cambia tu cómputo si quieres (libertad 1). Acceso al código fuente es una condición previa para esto.
- La libertad para redistribuir copias de manera que puedas ayudar a tu vecino (libertad 2).
- La libertad de distribuir copias de tus versiones modificadas a otros (libertad 3).
- Haciendo esto tú puedes dar a la comunidad entera la oportunidad de beneficiarse de tus cambios. El acceso al código fuente es una condición previa para esto.

Un programa es software libre si los usuarios tienen todas estas libertades.”
(Extraído de la página de la *Free Software Foundation*, 2011)

Por otro lado el software de código abierto u *Open Source Software* (OSS) se define como:

“El código abierto no significa principalmente acceso al código fuente. Los términos de distribución del software abierto deben cumplir con los siguientes criterios:

1. Libre redistribución. La licencia no debe restringir a nadie de vender o entregar el software como un componente de una distribución mayor que contiene programas de diferentes fuentes. La licencia no deberá solicitar regalías u otros pagos por dicha venta.
2. Código fuente. El programa debe incluir el código fuente, y debe permitir la distribución en código fuente así como la forma compilada. Donde alguna forma de un producto no está distribuida con código fuente, debe haber una manera bien publicitada de obtener el código fuente por no más que un costo de reproducción razonable preferentemente, la descarga vía internet sin cargo. El código fuente debe ser la forma preferida en que un programador modificará el programa. El código fuente deliberadamente confuso no está permitido. Formas intermedias como el producto de un preprocesador o traductor no están permitidas.
3. Trabajos derivados. La licencia debe permitir modificaciones y trabajos derivados, y debe permitir que estos se distribuyan bajo los mismos términos que la licencia del software original.
4. Integridad del código fuente del autor. La licencia puede restringir el código fuente de ser distribuido en forma modificada solo si la licencia permite la distribución de “parches” con el código fuente para el propósito de modificar el programa en tiempo de compilación. La licencia debe permitir explícitamente la distribución de software construido a partir de código fuente modificado. La licencia puede requerir que los trabajos derivados lleven un nombre diferente o número de versión del software original.
5. No discriminación contra personas o grupos. La licencia no debe discriminar contra cualquier persona o grupo de personas.
6. No discriminación contra campos de esfuerzo. La licencia no debe restringir a cualquiera de hacer uso del programa en un específico campo de esfuerzo. Por ejemplo, no puede restringir el programa de ser usado en negocios, de ser usado para investigación genética.
7. Distribución de licencia. Los derechos asociados al programa deben aplicarse a todos aquellos a quienes se redistribuya el programa sin la necesidad de pedir una licencia adicional para estas partes.

8. La licencia no debe ser específica de un producto. Los derechos asociados al programa no deben depender de que el programa sea parte de una distribución particular de software. Si el programa se extrae de esa distribución y usado o distribuido dentro de los términos de la licencia del programa, todas las partes a las que el programa se redistribuya deben tener los mismos derechos que los que se conceden en relación con la distribución de software original.
9. La licencia no debe restringir otro software. La licencia no debe poner restricciones en otro software que es distribuido en conjunto con el software licenciado. Por ejemplo, la licencia no debe insistir en que todos los otros programas distribuidos por el mismo medio deben ser software de código abierto.
10. La licencia debe ser tecnológicamente neutral. Ninguna disposición de la licencia puede basarse en una tecnología o estilo de interfaz. (Extraído de la página oficial de la *Open Source Initiative*)

Con el fin de salvar las diferencias entre ambas concepciones y tener un solo término que identifique al software no propietario ha surgido el término *Free Libre Open Source Software* (FLOSS), el cual engloba los dos puntos de vista y resulta útil al referirse a este tipo de software. Como se puede observar la característica de libre difusión es muy importante en el FLOSS. De acuerdo a Von Hippel esto incrementa la eficiencia en la producción del bien

“La eficiencia social de un sistema en el que las innovaciones individuales son desarrolladas por usuarios individuales se incrementa si los usuarios de alguna manera difunden lo que ellos han desarrollado a otros. La investigación empírica muestra que los usuarios frecuentemente alcanzan amplia difusión de maneras inesperadas: ellos frecuentemente “revelan libremente” lo que ellos han desarrollado. Cuando decimos que un innovador libremente revela información acerca de un producto o servicio que ha desarrollado, queremos decir que todos los derechos de propiedad intelectual de esa información son voluntariamente dados por el innovador, y todas las partes interesadas tienen acceso dado a esta –la información se convierte en un bien público” (Von Hippel, 2005, p 9).

Esta explicación de Von Hippel encaja perfectamente en la anterior descripción del FLOSS, así se hace evidente que su principal ventaja es también su principal

característica, la libertad de compartir la información contribuye a acelerar los procesos de mejora. El mismo autor agrega:

“Los innovadores frecuentemente revelan libremente debido a que es frecuentemente la mejor o la única opción práctica disponible para ellos. El esconder una innovación como un secreto comercial es improbable que sea exitoso por largo tiempo: demasiados individuos generalmente saben cosas similares, y algunos conservadores de la información ‘secreta’ pueden llegar a perder poco o nada por revelar libremente lo que saben.” (Von Hippel, 2005, p 10).

2.2. Forma de producción del FLOSS.

Como se mencionó en la introducción, la producción del software *open source* se da de manera privada, pero se constituye en un bien público. Yochai Benkler denominó a la forma de producción del software abierto como producción por pares (2002). Raymond (2001) por su parte describió la producción de Linux como producción de estilo bazar.

Benkler distingue tres formas de producción haciendo alusión a los conceptos ideados por Coase, en principio se sitúa la producción gobernada por el mercado, y cuando los costos de transacción son muy altos se presenta la producción organizada al interior de una empresa. Benkler presenta frente a estos dos modelos clásicos la producción por pares o iguales, para mostrar un nuevo tipo de producción carente de jerarquizaciones y el cual es caracterizado por motivaciones y compensaciones distintas.

“Si los problemas de motivación y organización pueden ser resueltos, entonces la producción basada en comunidad tiene dos grandes ventajas sobre empresas y mercados.

Primero, sitúa el punto de la decisión acerca de asignar cualquier persona dada a cualquier conjunto de recursos dado con el individuo. Dada la alta variabilidad entre los individuos y a través del tiempo en términos de talento, experiencia, motivación, enfoque, disponibilidad, etc., la creatividad humana es un recurso especialmente difícil de especificar para contratación eficiente o gestión. Las empresas reconocen esto e intentan resolver este problema creando varios esquemas de compensación

y de recompensas intangibles, como los premios al empleado del mes. Estos esquemas funcionan hasta cierto punto para aliviar las pérdidas de información asociadas con la producción gerencial, pero sólo en la medida en que estos esquemas capturen todas las motivaciones y contribuciones de manera precisa. La producción por pares provee un marco dentro del cual los individuos que tienen la mejor información disponible acerca de su propia aptitud para una tarea pueden auto-identificarse para la tarea. Esto proporciona una ganancia de información sobre las empresas y mercados, pero sólo si el sistema desarrolla algún mecanismo para filtrar juicios erróneos que los agentes hacen acerca de sí mismos. Esta es la razón por la que prácticamente todos los sistemas exitosos de producción por pares tiene un mecanismo robusto para la revisión estadística o entre colegas eliminando las contribuciones de los agentes que se evalúan erróneamente a sí mismos.

Las ganancias de asignación de la producción por pares son tan importantes como las ganancias de información. La creatividad humana no puede asumirse como un interruptor de encendido y apagado de idoneidad para un trabajo, como los modelos simples de producción industrial pueden tratar al trabajo. Uno no puede decir en el contexto de la información que 'esta persona pasa los requisitos de idoneidad para jalar de esta palanca todo el día' e ignorar la variabilidad más allá de este hecho. Es más probable que la variabilidad en la productividad será mayor para diferentes personas con cualquier conjunto de recursos dado y colaboradores para cualquier conjunto de proyectos dado. Describo esta diversidad como la probabilidad de que cualquier agente tendrá un buen ajuste con un conjunto de recursos y agentes que producen nueva información valiosa o bienes culturales. La producción por pares tiene ventaja sobre empresas y mercados porque permite a grupos más grandes de individuos rastrear grandes grupos de recursos en búsqueda de materiales, proyectos, colaboraciones, y combinaciones de lo que es posible para las empresas o individuos que funcionan en mercados. Los costos de transacción asociados con la propiedad y contrato limitan el acceso de las personas a cada uno, a los recursos y los proyectos cuando la producción está organizada en un modelo de empresa o mercado, pero no cuando está organizada en un modelo de producción por pares. Debido a que el ajuste de las personas a los proyectos y a cualquier otra persona es variable, existen retornos crecientes de escala del

número de personas, recursos, y proyectos capaces de ser combinados.” (Benkler, 2002, pp 7-8)

Así, es posible observar las posibilidades generadas por la modularidad del software recordemos la característica de incorporeidad mencionada por Lippoldt y Stryzowsky. La facilidad con que el software puede ser copiado y la posibilidad de construirlo tomando código de diversas fuentes han permitido que los usuarios se involucren con facilidad a la producción del mismo. También hay que considerar lo que menciona Benkler respecto a la ventaja de permitir a grupos más grandes de individuos rastrear los recursos, proyectos, etc. A través del internet los usuarios-colaboradores del software abierto pueden realizar grandes búsquedas a costos muy bajos, y tienen la posibilidad de elegir la tarea en la cual van a participar en función de sus habilidades, conocimientos y motivaciones. Sin embargo, los proyectos de producción por pares, en este caso el software libre presentan algunos problemas debidos a su naturaleza heterogénea y carente de jerarquías.

“Los años pasados han visto un crecimiento sin precedente de software *open source*. Al mismo tiempo, el movimiento ha enfrentado un número de retos. Resaltaremos dos de estos aquí: la “bifurcación” de los proyectos (el desarrollo de variaciones que compiten) y el desarrollo de productos para usuarios de gama alta.

Un problema que ha surgido en un número de proyectos *open source* es el potencial de los programas para dividirse en diversas variantes.

Otro reto ha sido el aparentemente menor énfasis en documentación y soporte, las interfaces de usuario, y la compatibilidad hacia atrás encontrada en al menos algunos proyectos *open source*. Las características tecnológicas relativas del software desarrollado en ambientes *open source* y tradicionales son asunto de discusión apasionada. Algunos miembros de la comunidad creen que este método de producción domina el desarrollo de software tradicional en todos los aspectos. Pero muchos abogados del *open source* argumentan que el software *open source* tiende a estar orientado a los usuarios más sofisticados. Esta idea es mostrada claramente por un desarrollador *open source*:

[E]n cada ciclo de lanzamiento Microsoft siempre escucha a sus *clientes más ignorantes*. Esta es la clave del embrutecimiento de cada ciclo de lanzamiento de software para seguir atacando la población de computadora no personal. Los desarrolladores de Linux y OS/2, por otro lado, tienden a escuchar a sus *consumidores más inteligentes*... Lo bueno que Microsoft hace al traer computadoras a los no usuarios es superado por la maldición que ellos traen a los usuarios experimentados [Nadeau, 1999].

Ciertamente, la difusión más grande de proyectos *open source* parece ser en lugares donde los usuarios finales son sofisticados, como el servidor Apache instalado por administradores de sistemas. En estos casos, los usuarios están aparentemente más dispuestos a tolerar la falta de documentación detallada o las interfaces fáciles de entender por el usuario a cambio de los ahorros en costos y la posibilidad de modificar el código fuente por ellos mismos.” (Lerner y Tirole, 2000, pp 7 y 8).

Los usuarios como eje de la innovación en el software tienen una importancia fundamental cuando consideramos lo mencionado por Von Hippel respecto al impacto comercial de la innovación realizada por los *lead users*. Esto aunado a lo que Drucker menciona: “...la innovación basada en el conocimiento depende más del mercado que cualquier otra clase de innovación.”(Drucker, 1998, p 7). Si las innovaciones realizadas por usuarios líderes tienen gran aceptación comercial y el software libre es una industria basada en el conocimiento, esto contribuye a explicar el éxito comercial de los proyectos de software libre como Apache, MySQL, etc.

2.3. Motivación de los programadores

Después de conocer cómo se produce el software libre, surge la duda acerca del porqué los programadores querrían participar, si bien es cierto que la producción por pares presenta muchas ventajas estas no son siempre evidentes a los participantes. Sobre todo si consideramos que ellos tienen una visión pequeña de la totalidad de los proyectos. Más allá de las razones ideológicas concernientes a la libertad implícita en los proyectos de software libre, existen otras causas que

explican la participación de cientos de miles de programadores independientes alrededor del mundo.

De acuerdo a Lerner y Tirole (2000)

“Una programadora participa en un proyecto, sea comercial u *open source*, sólo si se deriva un beneficio neto (ampliamente definido) de comprometerse en la actividad. El beneficio neto es igual al beneficio inmediato (el beneficio actual menos los costos actuales) más el beneficio retardado.

Una programadora trabajando en un proyecto de desarrollo de software incurre en una variedad de beneficios inmediatos y costos. Primero, la programadora recibe compensación monetaria si ella está trabajando para una empresa comercial. Segundo, la programadora puede estar corrigiendo un error de programación o personalizando un programa para su propio beneficio (también como, en el caso de un proceso *open source*, para el beneficio de otros.) Tercero, la programadora incurre en un costo de oportunidad de su tiempo. Mientras ella está trabajando en este proyecto no es capaz de involucrarse en otro proyecto de programación. El costo real de este tiempo depende de cuan disfrutable es el trabajo.” (Lerner y Tirole, 2000, p 14)

Esta última característica acerca del disfrute del trabajo está estrechamente relacionada con el placer que se mencionó en el apartado 1.4.3 de la motivación de los usuarios a innovar.

“La recompensa retardada cubre dos distintos, aunque difíciles de distinguir incentivos. El *incentivo concerniente a la carrera* se refiere a futuras ofertas de trabajo, acciones en compañías comerciales basadas en *open source*, o acceso futuro a mercados de *venture capital*. El *incentivo gratificación del ego* se deriva de un deseo de reconocimiento entre pares. Probablemente la mayoría de los programadores responde a ambos incentivos. Existen algunas diferencias entre los dos. La programadora principalmente preocupada por reconocimiento entre pares puede rehuir recompensas monetarias futuras, y podría también indicar su talento a una audiencia ligeramente distinta que aquellos motivados por preocupación por su carrera. Desde una perspectiva económica, sin embargo, los incentivos son similares en la mayoría de los aspectos. Agruparemos el incentivo concerniente a la

carrera y el incentivo de la gratificación del ego bajo un solo encabezado: el *incentivo señalador*.

La teoría económica [por ejemplo Holmström, 1999] sugiere que el incentivo señalador es más fuerte cuando,

- a) Más visible es el desempeño para las audiencias relevantes (pares, mercado de trabajo, comunidad de *venture capital*),
- b) Más alto es el impacto del esfuerzo en el desempeño, y
- c) Más informativo es el desempeño acerca del talento

La primera condición da lugar a lo que los economistas llaman 'complementariedades estratégicas.' Para tener una 'audiencia', los programadores querrán trabajar en proyectos de software que atraerán un gran número de otros programadores. Esto sugiere la posibilidad de equilibrios múltiples. El mismo proyecto puede atraer pocos programadores porque los programadores esperan que otros programadores no estén interesados; o éste podría florecer mientras los programadores (racionalmente) tengan fe en el proyecto.

El mismo punto aplica para bifurcar un proyecto *open source* dado. Los proyectos *open source* en este aspecto son bastante similares a la investigación académica. La última es bien conocida por exhibir modas. Campos son completamente descuidados por años, mientras otros con aparentemente no superior interés intrínseco atraen un gran número de investigadores. Las modas en la academia son mal vistas por su ineficiente impacto en la asignación de investigación. No debería ignorarse, sin embargo, que las modas también tienen beneficios. Una moda puede crear un fuerte incentivo de señalización: los investigadores trabajando en un área popular pueden estar altamente motivados a producir un trabajo de alta calidad, dado que ellos pueden confiar en que una gran audiencia examinará su trabajo. (Lerner y Tirole, 2000, pp 14-16)

Realizando una comparación entre las motivaciones que proporciona el software *open source* y el propietario tenemos:

“Los proyectos comerciales tienen una ventaja sobre las compensaciones actuales debido a que la naturaleza propietaria del código genera ingreso. Esto hace que valga la pena para las compañías privadas ofrecer salarios. Esta aseveración es el

viejo argumento económico de que los beneficios fomentan la inversión, que es usado por ejemplo, para justificar la adjudicación de patentes para impulsar la investigación.

En contraste un proyecto *open source* puede disminuir el costo del programador, por dos razones:

- i. 'Efecto *alumni*': Debido a que el código está libremente disponible a todos, puede ser utilizado en escuelas y universidades para propósitos de aprendizaje; por eso resulta familiar a los programadores. Esto reduce el costo de programación para UNIX por ejemplo.
- ii. Beneficios de personalización y corrección de errores de programación: el costo de contribuir a un proyecto *open source* es menor si la actividad brinda un beneficio privado (corrección de errores de programación, personalización) para la programadora y su empresa. Note nuevamente que este factor de reducción de costo está directamente vinculado a la apertura del código origen.

Cambiamos ahora al componente de recompensa retardada (incentivo señalador). En este aspecto también, el proceso *open source* tiene algunos beneficios sobre el acercamiento de código cerrado. Como hemos notado, los incentivos señaladores son más fuertes, mientras más visible el desempeño y más atribuible sea a un individuo dado. Los incentivos señaladores por lo tanto pueden ser más fuertes en el modo de *open source* por tres razones:

- i. Mejor medida del desempeño: los extranjeros pueden observar solamente de manera inexacta la funcionalidad y/o calidad de los elementos individuales de un típico programa comercialmente desarrollado, mientras son incapaces de observar el código origen propietario. En contraste, en un proyecto *open source*, los extranjeros son capaces de ver no sólo cual fue la contribución de un individuo y si ese componente "funcionó" pero también si la tarea fue difícil, si el problema fue dirigido en forma útil, si el código puede ser útil para otras tareas de programación en el futuro, y así sucesivamente.
- ii. Iniciativa completa: La programadora *open source* es su propia jefa y toma completa responsabilidad de éxito de un subproyecto. En una empresa comercial jerárquica, sin embargo, el desempeño de la programadora depende de la

interferencia de su supervisor, consejo, etc. La teoría económica predeciría que el desempeño de la programadora es medido más precisamente en el viejo caso.

iii. Mayor fluidez: podría argumentarse que el mercado de trabajo es más fluido en un ambiente *open source*. Es más probable que los programadores tengan menos idiosincrasia, o empresa-específico, el capital humano que limita el cambio del trabajo propio a un nuevo programa o ambiente de trabajo. (Dado que muchos elementos del código abierto son compartidos a través de proyectos de código abierto, más del conocimiento que ellos han acumulado pueden ser transferidos a un nuevo ambiente).

Estos argumentos teóricos también proveen puntos de vista acerca de quién es más probable que contribuya y que tareas son las que mejor se adaptan a proyectos *open source*. Los usuarios sofisticados obtienen beneficios directos cuando ellos personalizan o corrigen un error de programación en el software *open source*. Una segunda categoría de contribuidores potenciales consiste en individuos con fuertes incentivos señaladores; estos pueden usar el software *open source* como puerta de entrada. Por ejemplo, los procesos *open source* pueden dar a un talentoso administrador de sistema en una universidad (quién es también un usuario) una oportunidad única de mostrar su talento a sus pares, posibles empleadores, y comunidad de *venture capital*.” (Lerner y Tirole, 2000, pp 16-18)

Esta variedad de motivaciones contribuye a las diferencias en la participación de los programadores en proyectos tanto propietarios como *open source*. En función de las expectativas de los programadores, estos se verán más o menos incentivados a participar en determinado proyecto. Además de solucionar un problema de programación propio, se observa el papel importante que juegan tanto el placer como el sentido del logro al explicar la participación de los programadores en proyectos *open source*.

2.4. Liderazgo en el software libre

En el entorno productivo del software abierto donde las contribuciones son voluntarias y están dadas en función de las capacidades así como motivaciones de los colaboradores, el liderazgo adquiere relevancia en la forma de organizar las contribuciones. Desde el punto de vista de la generación del código fuente según Lerner y Tirole el liderazgo cumple las siguientes funciones:

"Aunque el líder es a menudo en el origen un usuario que intenta resolver un programa en particular, el líder con el tiempo lleva a cabo menos y menos programación. El líder debe (a) proporcionar una visión, (b) asegurarse de que el proyecto global se divida en tareas más pequeñas y bien definidas ("módulos") que los individuos puedan enfrentar de forma independiente de otras tareas, (c) atraer a otros programadores, y, por último pero no menos importante, (d) evitar que el proyecto se bifurque o abandone". (Lerner y Tirole, 2000, p 21)

Esta definición de las funciones del líder nos permite comprender su importancia para los proyectos globales open source, sin embargo, en el entorno empresarial de los negocios que utilizan el software libre como base de su modelo de negocios las funciones no son las mismas, en realidad tienen más que ver con la habilidad técnica y conocimiento del negocio.

"La confianza y la familiaridad entre los miembros de los equipos es la clave en la creación y generación del conocimiento y, por ende, en el éxito de los nuevos productos generados en este marco de innovación (Akgün, Byrne, Keskin, Lynn, & Imamoglu, 2005; Lee & Cole, 2003; Powell, Koput, & Mith-Doerr, 1996). En consecuencia, hablamos de un liderazgo que fomente la confianza y motivación entre pares, mitigando la percepción de riesgo. No tanto como consecuencia de la aprobación por parte de los superiores del trabajo de sus subordinados, sino al vincular el liderazgo con una elevada formación y conocimiento técnico relacionado con la temática del proyecto." (González y García, 2010, p 90)

Por lo tanto el liderazgo en este entorno se da en función con la habilidad técnica que posea el líder. Se le reconoce la experiencia y la capacidad para desarrollar soluciones. El conocimiento se constituye en el origen de su autoridad. El papel de experto surge como explicación de dicha autoridad y sus pares al reconocer su habilidad le confieren la misma.

2.5. Modelos de negocio.

Después de establecer que una de las características del software *open source* es que permite una forma distinta de llevar a cabo la producción de software, el siguiente paso es preguntarse de qué manera los participantes de esta industria

obtienen ingresos. Como se mencionó en la introducción, este modelo de producción y distribución lejos de ser un fenómeno aislado continúa creciendo y han surgido en torno al software libre distintas empresas que operan bajo distintos modelos para obtener ingresos utilizando al software libre como insumo o herramienta.

“En otras palabras, los individuos producen sobre bases no propietarias y contribuyen con su producto a un conocimiento ‘colectivo’ que nadie reconoce ‘poseer’, y que cualquiera puede, de hecho es requerido por normas profesionales, tomar y ampliar. Nos apropiamos del valor de nuestras contribuciones usando una variedad de métodos: modelos basados en el servicio mejor que en el producto (enseñar en lugar de regalías por libros), subvenciones gubernamentales y fuentes sin fines de lucro, también como reputación e intangibles similares pero motivaciones inmensamente poderosas plasmadas en premios, títulos, etc. En la excitación de un momento que se siente como una gran transformación, es fácil tener un sistema mixto de producción comercial/propietaria y producción no propietaria por pares – no como un segundo mejor o un recuerdo proveniente de la edad media, sino debido a que en algunas cosas el sistema de producción no propietaria por pares del mundo académico es simplemente mejor.” (Benkler, 2002, p 13)

En principio habrá que comprender en qué consiste un modelo de negocio. Según Ostelwalder y Pigneur (2009) “Un modelo de negocio describe los fundamentos de cómo una organización crea, desarrolla y captura valor.” Es necesario conocer el entorno para poder establecer una estrategia adecuada.

“El contexto de la estrategia de negocios es el mercado en que se actúa, por lo tanto, las discusiones sobre la estrategia deben empezar con referencia al mercado. Hoy los factores más críticos son el cambio acelerado, el aumento de la competencia y de la complejidad” (Morris, 2003 p. 5).

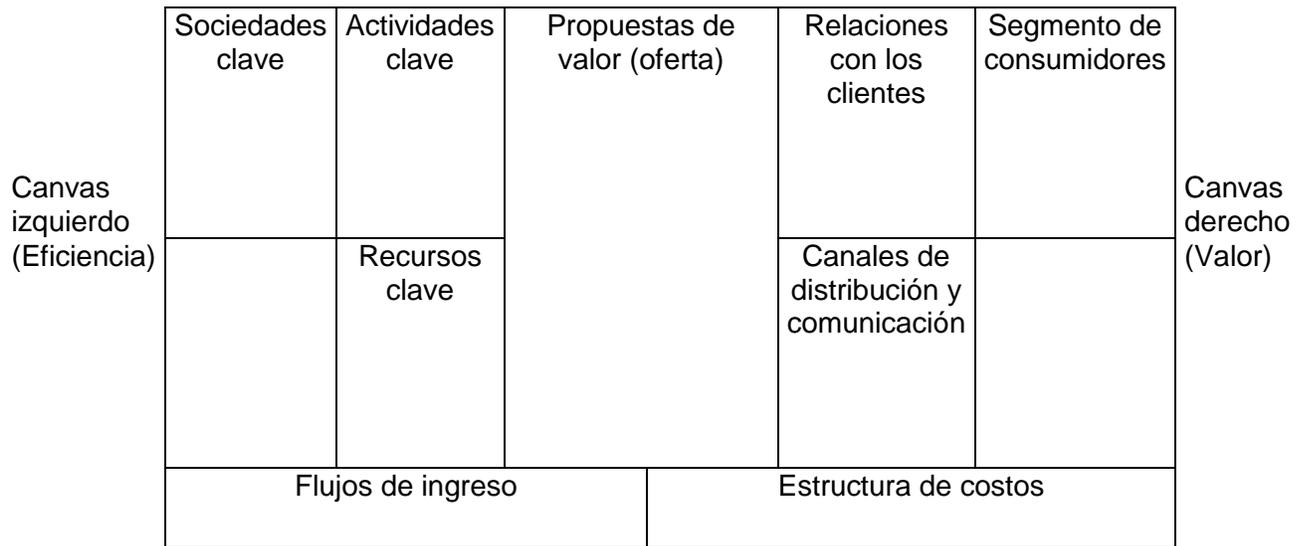
Al desenvolverse en entornos tan dinámicos se torna necesario que las organizaciones tengan un concepto claro de su modelo de negocio, de manera que puedan tomar las decisiones más acertadas para crear ofertas de valor que les permitan permanecer en el mercado.

El modelo que proponen Osterwalder y Pigneur consiste en identificar nueve variables distintas que constituyen el modelo de negocio estas son:

1. Segmento de consumidores. Una organización sirve a uno o varios segmentos de consumidores.
2. Propuestas de valor. Busca resolver los problemas de los consumidores y satisfacer las necesidades de los consumidores con propuestas de valor.
3. Canales. las propuestas de valor son entregadas a los consumidores a través de los canales de comunicación, distribución y ventas.
4. Relaciones con los clientes. Las relaciones con los clientes se establecen y mantienen con cada segmento de consumo.
5. Flujos de ingreso. Resultan de propuestas de valor exitosas ofrecidas a los clientes.
6. Recursos clave. Son los recursos requeridos para ofrecer y entregar los elementos descritos previamente.
7. Actividades clave. A través de determinado número de actividades clave.
8. Sociedades clave. Algunas actividades son subcontratadas y otros recursos son adquiridos fuera de la empresa.
9. Estructura de costos. Los elementos del modelo de negocio resultan en una estructura de costos.

Usando estas variables como bloques construye lo que denominan *Business Canvas Model* que a su vez clasifica estas variables en *canvas* izquierdo enfocado en la eficiencia y *canvas* derecho enfocado en el valor.

Figura 3. Business Canvas Model



Fuente: Tomado de Osterwalder y Pigneur (2009). *Business Model Generation*.

A partir de este lienzo se puede diseñar el modelo de negocio, utilizando como centro del mismo uno o más bloques. Este gráfico es útil para analizar la forma en que está concebido un modelo de negocio e identificar sus ventajas y desventajas, así como lo que hace diferente a una empresa de su competencia.

En el caso específico del software libre Rossi, Gonzales, Romera y Patón (2008) proponen la siguiente clasificación en los modelos de negocio del software libre.

Tabla 1. Modelos de negocio de software libre

Modelos clásicos		
Empresarios	Estrategia	Características
Desarrolladores de software	Cobrar una cuota mínima por el software libre	Cobrar por la distribución de una copia del software libre.
	Licencia dual	Generar dos modelos de licencia una propietaria y una libre.
	Partnership o franquicias	Todo el software para crear una pila de software libre está disponible gratuitamente. El secreto reside en la manera en la que este software es integrado, probado y certificado al igual que una marca comercial
	Liberar la penúltima versión	El software se distribuye bajo una licencia propietaria pero que tenga una fecha de expiración.
Proveedores de servicios	Soporte	Los servicios de soporte tratan de ayudar a resolver determinados problemas con algún producto en vez de entrenar o personalizar.
	Mantenimiento	El mantenimiento de una aplicación consiste en mantener el software instalado en un estado consistente y funcionando. También consiste en instalar y mantener nuevas versiones del producto así como solucionar problemas, etc. No existe diferencia entre software propietario y código abierto en esta materia.
Modelos clásicos		
Empresarios	Estrategia	Características
Proveedores de servicios	Integración	La integración del software consiste en repartir paquetes compuestos de un conjunto de componentes probados e integrados. Este es el trabajo de los editores de las distribuciones de Linux y de los proveedores de software que crean paquetes de elementos de código abierto bajo demanda.

Modelos clásicos		
Empresarios	Estrategia	Características
Proveedores de servicios	Formación	Las ofertas de formación y entrenamiento preparan a los usuarios y futuros administradores en el uso del software. La formación puede estar ofertada con la implantación del software, cuando se actualiza y se introducen nuevas funcionalidades o simplemente para aumentar habilidades de los usuarios.
	Personalización	Para el software no estándar así como para el software estándar personalizable una compañía puede ofrecer el ajuste de éste para cubrir unas necesidades determinadas. La personalización es proporcionada por las compañías expertas en el proceso de compilar e integrar productos de código abierto implantados en el sistema de destino al cual cobran por el servicio.
Modelos emergentes		
Empresarios	Estrategia	Características
Desarrolladores de software	Software propietario funcionando sobre núcleo de software libre	Este modelo es lo mismo de lo llamado "Estrategia del Patrocinador" en la clasificación de John Koenig: la empresa actúa como patrocinadora de un software libre por dar más confianza a los usuarios de la calidad del software, así que más empresas y más usuarios profesionales se acerquen al software y lo usen. Los ingresos son generados de la venta de software por encima de la pila del software.

Modelos emergentes		
Empresarios	Estrategia	Características
Generar ecosistemas de negocio	Creación de redes o consorcios de empresas	Cuando el Software Libre constituía un fenómeno relativamente aislado estas empresas tendían a agruparse en conjuntos relativamente horizontales unidos exclusivamente porque todos ofrecían servicios o desarrollos basados en Software Libre. A medida que estos grupos fueron madurando, estas agrupaciones evolucionaron hacia consorcios centralizados alrededor de un ámbito o tecnología concretos.
	Agrupaciones o redes de empresa	Un ejemplo claro es MORFEO. Fue fundado por Telefónica I+D, y las Universidades Rey Juan Carlos y Politécnica de Madrid, aunque a lo largo de los sus varios años de vida ha ido incorporando PYMES, grandes empresas y centros de investigación.
Software como servicio (SAAS)	Modelo de suscripción	Este modelo de negocio consiste en que el cliente que utiliza el software libre paga por un conjunto de servicios durante un tiempo determinado, a modo de suscripción. Los servicios más comunes que ofrecen las empresas que trabajan con este modelo de negocio son, entre otros, acceso a las actualizaciones, soporte con un tiempo determinado de respuesta, consultoría y formación.
	Software propietario funcionando sobre núcleo de software libre	Este modelo consiste en construir software propietario sobre una plataforma libre o utilizando herramientas libres, y después vender sus servicios, normalmente a través de Internet. Importantes empresas de la red, como son Google o eBay utilizan este modelo de negocio. Ellos cobran por diversos servicios, como anuncios en sus páginas web, servicios de búsqueda para empresas, por venta realizada, etc.

Modelos emergentes		
Empresarios	Estrategia	Características
Software como servicio (SAAS)	Hardware con FLOSS incluido	Nokia decidió confiar en componentes basados en Software Libre, eligiendo Linux como Sistema Operativo y Debian como distribución específica. Nokia asumió el desarrollo de componentes faltantes liberándolos como software libre. También pudo “subcontratar” el desarrollo de ciertas aplicaciones a terceras empresas con mayor experiencia en el dominio específico.
	Publicidad	Este modelo consiste en hacer más visible un software Open Source. Mayormente el software sea conocido y sea usado, mayor serán las probabilidades que coincida con los estándares oficiales.

Fuente: Elaboración propia con información de Rossi et al, (2008). *Modelos de negocios para SW libre*

2.6. Gestión de la propiedad intelectual

En un modelo como el del software de código abierto hablar de propiedad intelectual quizás resulte contradictorio, dada su vocación por compartir la información. La capacidad de apropiación de los beneficios de la innovación ha sido reconocida como un incentivo a innovar, (Teece, 1986), sin embargo en el caso del software libre, este incentivo deja de funcionar. En México los programas de cómputo se protegen mediante la Ley federal del derecho de autor la cual dedica su capítulo cuatro a los programas de computación y bases de datos. De acuerdo al artículo número ciento uno un programa de cómputo: “Se entiende por programa de computación la expresión original en cualquier forma, lenguaje o código, de un conjunto de instrucciones que, con una secuencia, estructura y organización determinada, tiene como propósito que una computadora o dispositivo realice una tarea o función específica.” Esta ley restringe los derechos de los usuarios a obtener copias del programa que hayan adquirido, faculta al poseedor de los derechos el permitir o prohibir la copia, distribución, traducción y en sí cualquier cambio que se realice al programa. En la práctica, para el caso del software libre, se utilizan diferentes licencias creadas por universidades, y

organizaciones como la *Free Software Foundation*. Una explicación bastante clara acerca de estas licencias la proporciona la Guía básica de software de fuentes abiertas (2008)

“El software protegido con *copyleft* confiere unos términos de distribución que aseguran que todas las copias, así como las mejoras y modificaciones que se realicen, sigan siendo software abierto. Esto significa que estas licencias no permiten a terceros restringir ninguna de las libertades originales, y exige que el código fuente siga siendo público en cualquier caso. El otro gran grupo de licencias de software abierto lo constituyen aquellas que no obligan necesariamente al usuario a que sus modificaciones sigan siendo software abierto. Entre ellas las más conocidas son las licencias BSD (Berkeley Software Distribution) y las licencias tipo MPL (Mozilla Public License).

En este caso, el autor mantiene la protección de copyright únicamente para la renuncia de garantía y para requerir la adecuada atribución de la autoría en trabajos derivados, pero permite la libre distribución y modificación. Esta licencia es tan permisiva, que el usuario tiene libertad ilimitada con respecto al software resultante, y puede decidir, incluso, redistribuirlo como no libre. Ambos tipos de licencias siguen diferentes filosofías de desarrollo y distribución, pero cada una con ventajas e inconvenientes: Las licencias de software abierto con *copyleft* (GPL), tienen como finalidad la de perpetuar y mejorar el conjunto de programas libres que reviertan en la comunidad, sin por ello perder el control de la propiedad intelectual. Las licencias de software abierto sin *copyleft* (BSD y MPL), son más prácticas y flexibles, permiten unir y compatibilizar código abierto y cerrado reteniendo, a nivel empresarial, las ventajas competitivas al licenciar el producto final del modo que más le convenga a la empresa desarrolladora.” P 36-37

Con objeto de no confundir al software libre con otras modalidades de distribución la Guía nos explica los siguientes conceptos.

“**El Software de dominio público:** el software de dominio público es aquél en que el autor ha renunciado explícitamente a sus derechos de propiedad intelectual. No hay licencia que lo ampare o restrinja, si bien el código fuente no ha de estar necesariamente disponible.

El Freeware: este término es usado comúnmente para referirse a aplicaciones que se pueden distribuir pero no modificar (por tanto su código fuente no está disponible). A pesar de lo ambiguo del nombre estas aplicaciones no son software libre.

El Shareware: es software del que se permite redistribuir copias, pero por cada copia realmente utilizada, el usuario debe pagar un cargo por licencia. Responde a la filosofía comercial de “probar antes de comprar”; en función del grado de uso así es el pago.” P 37

2.7. La industria del software en México

La importancia de la industria del software en México ha sido reconocida, por lo que se ha creado el Programa de Desarrollo del Sector de Servicios de Tecnologías de Información (PROSOFT 2.0),”el cual busca crear las condiciones necesarias para que México cuente con un sector de TI más competitivo internacionalmente y asegurar su crecimiento en el largo plazo.” (PROSOFT 2.0, 2011)

Esta importancia la muestran los siguientes datos:

“En México, el PIB informático se triplicó prácticamente entre 1994 y 2004. En ese último año representaba alrededor de 4.5 por ciento del PIB total frente a 1.9 por ciento en 1994. Por otro lado, las exportaciones informáticas se multiplicaron por diez entre 1994 y 2004, de un millardo de dólares a casi 11 millardos de dólares, lo que significa que, como porcentaje de las exportaciones manufacturas (sic), pasaron de 2.5 por ciento a 7 por ciento en dicha década.

Segmentos, mercado y empresas de software en México.

1. Industria nacional de software y servicios informáticos.
 2. Producción interna (autoconsumo-in house).
 3. Filiales de las grandes empresas trasnacionales de software empaquetado.
 4. Grandes empresas trasnacionales exportadoras de productos electrónicos.”
- (Hualde, Jaén y Mochi en Hualde, 2010, pp 64-65)

“El mercado de software está compuesto por el software en paquete (aplicaciones; herramientas de software; infraestructura y seguridad) y servicios (desarrollo a la medida; consultoría; integración de aplicaciones empresariales) relacionados con el desarrollo de software que venden las empresas locales (tanto a nivel nacional como internacional), incluidas las importaciones de software.” (Hualde, Jaén y Mochi en Hualde, 2010, p 65)

En referencia al capital humano.

“Un estudio realizado por la Universidad Autónoma Metropolitana (2004) -...- señala que actualmente la oferta de profesionistas, entendida como la matrícula en carreras relacionadas con la industria (informática, computación), supera la demanda.

... La brecha entre egresados y el mercado de trabajo no es un indicador inequívoco de una oferta excesiva de personal calificado, porque se desconocen aspectos específicos respecto a la calidad de los egresados.” (Hualde, Jaén y Mochi en Hualde, 2010, p 78)

En resumen hemos mostrado que el software consiste en un conjunto de instrucciones que permiten la operación de las máquinas, las cuáles poseen un código fuente y diversos módulos que pueden ser ensamblados de acuerdo a las necesidades de los usuarios. De acuerdo a sus características económicas se dividen en software propietario y libre o abierto. Para evitar confusiones en este último concepto se ha adoptado el término FLOSS. Este tipo de software se caracteriza por la libertad que proporciona a los usuarios de acceder al código fuente, de modificarlo, adaptarlo, traducirlo y demás. También se caracteriza por su forma de producción que se ha denominado “producción por pares” y que consiste en que un número indeterminado de programadores participen construyendo distintos proyectos de software libre, aportando sus recursos en función de las motivaciones que tengan. Dichas motivaciones incluyen desde: salarios inmediatos, expectativas de acceso a capital de riesgo, solución de problemas individuales de programación, hasta sentido de logro y obtención de placer al resolver un problema complejo. Dada la importancia de la recompensa a largo plazo, para los programadores resulta fundamental dar a conocer sus logros

con objeto de volverse visibles en la comunidad y acceder a los beneficios esperados por su participación.

También se ha explicado la forma en que se comparte el conocimiento en este sistema de producción. La existencia de distintas licencias, emanadas de la propia comunidad FLOSS, que regulan el uso de las contribuciones hechas por los programadores. Es necesario hacer notar que dado el mecanismo de incentivos que operan en este entorno, es muy importante para los programadores poder adjudicarse la autoría de código o la solución de un problema. Así, a pesar de tener libre acceso al código, aún en los casos en que es gratuito, es importante reconocer al autor del mismo.

Otra de las cosas que caracteriza al software libre es el modelo de negocio bajo el que operan. Dado que basan sus ingresos no en la venta de código, sino en la venta de actividades relacionadas, donde éste juega un papel de herramienta, más no es el producto principal. En el capítulo siguiente se muestran cuatro empresas cada una con modelos de negocio distintos, y todas ellas operan utilizando software libre como parte primordial de sus negocios.

Todas las características mencionadas se buscaron en los estudios de caso que se describen en el siguiente capítulo. Si bien se encontraron numerosas coincidencias con la teoría expuesta hasta ahora, también aparecieron comportamientos que parecen contradecirla. La investigación ha resultado muy útil para comprender como un proceso de índole global ocurre en el entorno local.

3. Estudios de caso

Con objeto de observar cómo da en la práctica el proceso de innovación en las empresas del sector de software libre, se recurrió a entrevistar a cuatro empresas dedicadas a la producción de software abierto a las cuáles denominaremos: **Azul, Verde, Roja y Amarilla**, con objeto de mantener la confidencialidad y facilitar su identificación.

3.1. Perfil del empresario

La primera parte de la entrevista consistió en establecer un perfil de los empresarios así como de las empresas. Hay que considerar que en el caso de la empresa Roja no se entrevistó al dueño sino a un líder de proyecto quien antes tuvo su propia empresa.

En cuanto a la formación académica, tres de ellos tienen estudios universitarios completos.

El dueño de Amarilla posee una maestría en calidad. El dueño de Verde tiene una especialidad no relacionada con el software, dada la complejidad de los conocimientos requeridos para trabajar con software cabría esperar que tuviera una formación académica de tipo ingeniería en sistemas, informática o afín. Sin embargo, este empresario comenta que llevaba mucho tiempo trabajando con software y sabía lo que era necesario para tener una empresa en este sector, dando a entender que el conocimiento necesario se puede obtener de manera alternativa a la formación académica. El dueño de la empresa Azul tiene formación universitaria en ingeniería en sistemas sin concluir. El representante de Roja tiene una licenciatura en Matemáticas. Las edades de los empresarios son: Azul 37, Verde 38, Amarilla 38 y Roja 46 años.

3.2. Perfil de la empresa

Los años de fundación de las empresas son:

Roja- 1994; Azul-1997; Amarilla-2004 y Verde-2004

Todas las empresas se encuentran ubicadas en el Distrito Federal

En 2011 la empresa verde cambio su nombre pero conservó el mismo modelo de negocio. Respecto al modelo de negocio que presentan Amarilla y Roja están especializadas en labores de consultoría en *business intelligence*, Verde en hosting de sitios de internet y Azul aunque trabaja principalmente con programas abiertos de ofimática busca solucionar problemas de diversa índole mediante las herramientas de software ya sea libre o propietario. De esta manera la generación de ingresos de todas las empresas proviene del conjunto de servicios proporcionados a través del software libre más no de la venta del software por sí mismo.

Según los modelos propuestos por Rossi *et al* revisados en el marco teórico, las cuatro empresas se consideran proveedores de servicios y podríamos clasificarlas de la siguiente manera.

Tabla 2. Estrategias de negocio de las empresas entrevistadas

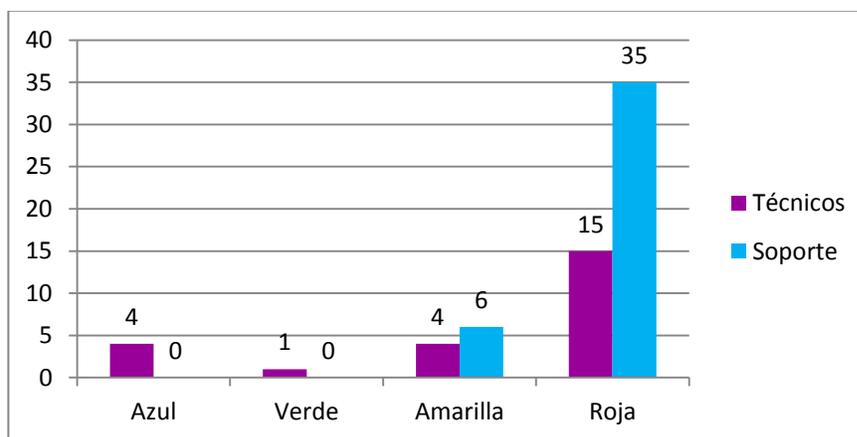
Empresa	Estrategia	Producto principal
Azul	Mantenimiento y formación.	Servicios de ofimática.
Verde	Mantenimiento	Hosting en internet
Roja	Personalización, soporte y formación.	Business Intelligence
Amarilla	Personalización, soporte y formación.	Business Intelligence

Fuente: Elaboración propia con base en las entrevistas (2011).

Respecto a las estrategias de innovación que siguen tenemos de acuerdo a Freeman citado por Escorsa y Valls (2003) todas las empresas tienen una estrategia ofensiva, es decir, buscan colocarse a la cabeza de los competidores introduciendo productos nuevos, en muchos de los casos se trata de productos únicos al estar completamente adaptados a las necesidades de los clientes y también una estrategia oportunista porque todas operan en nichos de mercado muy pequeños. En tanto desde el punto de vista de Urban y Hauser, también citados por Escorsa y Valls (2003), las estrategias serían sensibles a los clientes, imitadoras, defensivas y basadas en I+D. Esto es porque las empresas al compartir el conocimiento pueden beneficiarse de la innovación de las otras, por lo tanto son imitadoras en determinado momento, esto también las vuelve defensivas, ya que tienen los conocimientos necesarios para mejorar lo realizado por las otras. Dada la naturaleza de la industria se requiere mucha investigación y desarrollo, aunque es importante notar que no precisamente se da de manera estructurada contando con un departamento de I+D, salvo en el caso de la empresa Roja.

En el siguiente cuadro se muestra el número de empleados de las cuatro empresas así como la forma en que se distribuyen de acuerdo a su función. Como podemos ver únicamente Amarilla y Roja tienen empleados de soporte, en los casos de Azul y Verde las actividades de soporte, como contabilidad, se subcontratan.

Gráfica 1. Distribución de empleados por empresa y áreas funcionales



Fuente: Elaboración propia con base en las entrevistas (2011).

De igual manera, las empresas Verde y Roja manifestaron que contratan personal temporal en función de los proyectos que estén ejecutando. La empresa Azul está conformada únicamente por socios, de manera que las relaciones laborales no son de manera patrón-empleado, pero se reconoce a un líder que se encarga de las funciones de coordinación.

Las empresas están conformadas por distintas áreas dependiendo de su tamaño así como de su modelo de negocio y los productos o servicios que presta. En la Tabla 3 se muestran las distintas áreas que integran las empresas.

Tabla 3. Áreas que conforman la empresa

Azul	Verde	Amarilla	Roja
Las tareas se asignan en función del conocimiento de los socios.	Un solo empleado, que se encarga de operación, consultoría y soporte.	Dirección	Tiene el área de administración que se encarga de la contabilidad, RH y ventas.
	El dueño se encarga de desarrollo, dirección y consultoría	Contabilidad	El área de desarrollo se ocupa de la parte técnica.
		Recursos Humanos	La dirección la realiza el dueño de la empresa.
		Desarrollo	
		I+D+i	

Fuente: Elaboración propia con base en las entrevistas (2011).

Respecto al liderazgo dentro de las empresas, las respuestas coincidieron en que se trataba de liderazgos centralizados basados en el conocimiento. Las características que los empresarios consideraban necesarias para ser líder son: conocimiento técnico, empatía, trabajo, iniciativa. Destacaron que se trabaja en la cultura del hacker, lidera el que contribuye más al proyecto.

Respecto al tipo de conocimiento que las empresas poseen y generan, tenemos que todas cuentan con el *know-what* y *know-why* ya que el conocimiento técnico

es la base de su operación, es importante tener la capacidad de realizar código, generar manuales, etc., en cuanto al *know-how* y el *know-who* resultan fundamentales en empresas como las entrevistadas que mantienen estrechas relaciones tanto con clientes como con competidores y colaboradores que se encuentran en otra ubicación geográfica. Dado que todas las empresas prestan servicios personalizados y adaptados a las necesidades particulares del cliente, sus procesos internos integrados en el *know-how* se convierten en una ventaja respecto a sus competidores. Respecto al *know-when* y *know-where* los empresarios entrevistados comentaron que era importante prever cuando un desarrollo será obsoleto y adelantarse a las circunstancias, para todos el monitoreo del desarrollo tecnológico a nivel mundial es importante, contar con información actualizada que les permita mantenerse al corriente de los últimos desarrollos es fundamental para poder dar soluciones al mercado de manera rápida y efectiva.

Para las cuatro empresas su principal cliente es el gobierno a distintos niveles. También trabajan para corporativos, industria de la transformación, construcción, la empresa Verde manifiesta tener muchos usuarios individuales, escuelas, y compañías pequeñas.

Las empresas pertenecen a asociaciones de diversa índole. Las asociaciones a las que manifestaron pertenecer o haber pertenecido fueron:

AMESOL. Asociación Mexicana Empresarial de Software Libre A.C. Los dueños de Azul, Roja y Amarilla fueron fundadores de esta asociación, en la actualidad Amarilla ya no pertenece a ella.

CANACINTRA. Cámara nacional de la industria de la transformación.

RED LATINOAMERICANA DE LA INDUSTRIA DEL SOFTWARE. Integrada por empresas de México, Perú., Brasil y Colombia. Cuenta con financiamiento del Banco Interamericano de Desarrollo. Proporciona apoyos para la certificación de las empresas de software.

Además cabe destacar que el dueño de Amarilla es miembro de la Comisión de ciencia y tecnología del Senado.

Otro de los puntos focales en la industria del software es la motivación para participar, si bien lo que la teoría discute principalmente son las contribuciones de los desarrolladores independientes, me pareció pertinente verificar cómo se motiva a los colaboradores que trabajan para una empresa basada en software abierto. Se le preguntó a los empresarios de qué manera motivaban a sus colaboradores y cuáles eran las cualidades que buscaban en aquéllos que contrataban.

Tabla 4. Formas de motivar a los colaboradores

Importancia	Azul	Verde	Roja	
1	Dinero	Prestigio	Identificando las necesidades de la gente	Reconocimiento
2	Ambiente laboral cordial	Reconocimiento	Ambiente sano de trabajo	Salario
3			Saber que le gusta a la gente	Acciones

Fuente: Elaboración propia con base en las entrevistas (2011).

En cuanto al perfil que buscan en sus colaboradores, Azul, Verde y Amarilla indicaron que lo más importante eran las características de personalidad, iniciativa, creatividad, osadía, tolerancia a trabajar en un proyecto durante mucho tiempo y que al final no funcione, estas empresas indicaron tener programas de capacitación, debido a que en ocasiones los nuevos colaboradores no tienen experiencia en los lenguajes o herramientas utilizados. La empresa Roja dijo que elaboraba un perfil para cada puesto con características técnicas y de personalidad, finalmente los candidatos son entrevistados por quién será su jefe directo para decidir si se le contrata o no. Es interesante observar la importancia

que todas las empresas otorgan a la personalidad más allá de las cualificaciones técnicas, las características que se buscan están relacionadas nuevamente con la cultura del hacker, hay que recordar que para estas empresas el liderazgo es lineal y está estrechamente relacionado al conocimiento.

3.3. Motivación para participar en software libre

Como vimos en los capítulos anteriores, las razones por las que una persona se involucra en proyectos de software libre son diversas, para conocer las razones de los empresarios para crear empresas basadas en este tipo de software se les preguntó directamente la razón por la que habían elegido ese negocio, y también se les preguntó cuáles consideraban ellos eran las motivaciones de sus colaboradores para innovar. En la sección anterior se mostró de qué manera los empresarios motivaban a sus colaboradores al interior de la empresa para tener un buen desempeño. Pero se elaboró una pregunta expresa para saber por qué los colaboradores querrían innovar, las respuestas a estas preguntas se presentan a continuación.

3.3.1. Motivación de los empresarios

Cuando se les preguntó a los empresarios el motivo por el cual habían elegido trabajar con software libre el dueño de Amarilla respondió que era debido a su filosofía, el de Verde dijo que se trataba de algo que ya conocía y que le ofrecía resultados en el menor plazo posible. Los dueños de Azul así como Roja aludieron razones de costo y seguridad debido a que se ahorra el pago de licencias y a que se sabe con certeza cómo funciona el código.

Al preguntarles cuál es la principal ventaja del software libre sobre el propietario las respuestas fueron: bajos costos, control, permite ser mejorado, libertad, seguridad, ajustar la tecnología a los modelos de negocio de los clientes. A partir de estas respuestas podemos deducir que las características más importantes del software libre desde el punto de vista de los empresarios son: bajos costos, libertad y seguridad.

En cuanto a si existía una aplicación idónea del software abierto, las respuestas fueron: es una herramienta, sólo sirve si se sabe usar; está en todos lados, tiene múltiples aplicaciones; técnicamente implica el mismo trabajo desarrollar software abierto que propietario, la diferencia es económica, existen áreas muy críticas para la empresa en que se usa software propietario y finalmente uno de ellos considera que su uso idóneo es en gobierno debido a los bajos costos y la seguridad que proporciona.

De acuerdo a lo referido anteriormente, la iniciativa de los desarrolladores tiene distintos aspectos como son: la carga ideológica que implica el pertenecer al movimiento del software abierto o libre; la motivación dada por el logro; así como razones meramente económicas.

3.3.2. Motivación de los colaboradores

Cuando se les preguntó por las motivaciones de los colaboradores para innovar el dueño de Amarilla respondió que cuando son muy jóvenes buscan sobre todo sentido del logro, aprendizaje y reconocimiento, cuando sus necesidades económicas cambian se enfocan en el salario y recompensas económicas. De manera similar opinó el dueño de Azul. Hay que considerar que esta pregunta está restringida a la motivación de los colaboradores al interior de las organizaciones y no responde a conocer las motivaciones que tienen los participantes independientes u otras empresas que también generan código que es utilizado por las empresas encuestadas. Aun así, es posible confirmar los planteamientos teóricos expuestos por la teoría, (Benckler [2002], Lerner y Tirole [2000], Von Hippel [2005]) respecto a las expectativas y que los motivan para participar en proyectos *open source*.

Tabla 5. Motivación de los colaboradores a innovar

Importancia	Azul	Verde	Roja	Amarilla
1	Sentido de logro	Vanidad	Considera que la creatividad es innata y por lo tanto hay que colocar a las personas en el lugar adecuado.	Sentido de logro/aprendizaje
2	Aprendizaje			Búsqueda de reconocimiento
3	Salario			Salario
	Búsqueda de reconocimiento			

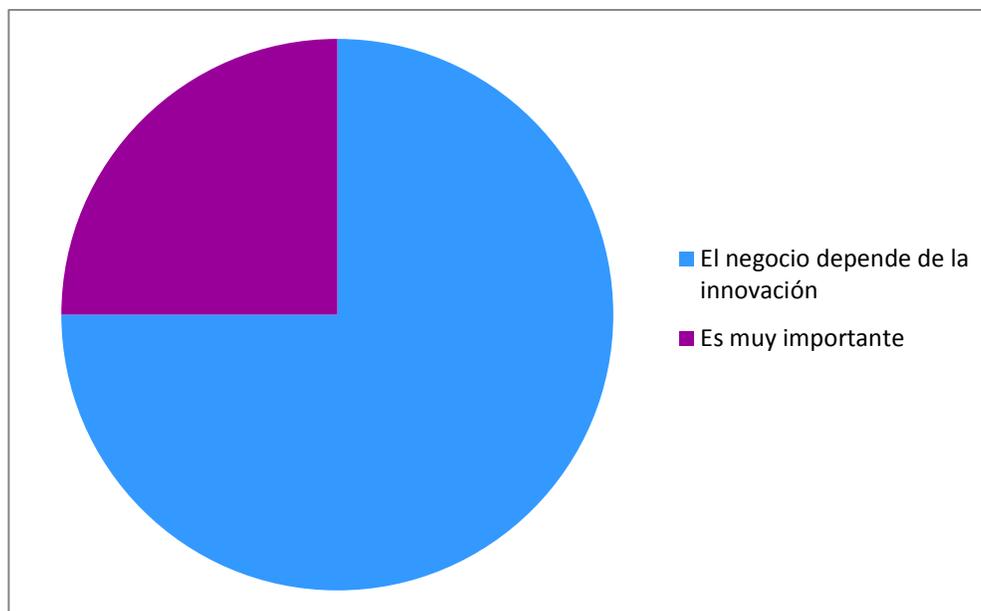
Fuente: Elaboración propia con base en las entrevistas (2011).

El comentario de Roja destaca debido a que una de las ventajas que la teoría reconoce en la producción por pares es la facilidad de asignar los proyectos a las personas más adecuadas. Sin embargo, la afirmación de Roja sugiere que al interior de las empresas que operan de manera jerárquica la asignación de los proyectos se da de manera lineal. Mas ello no desmiente la afirmación teórica, debido a que existen colaboradores externos a las organizaciones que si cumplen con dicha afirmación.

3.4. Proceso de innovación

Respecto a cuán importante es la innovación tres de las cuatro empresas respondieron que su negocio depende de la innovación y la cuarta que es muy importante. En la siguiente gráfica se muestra la distribución de sus respuestas.

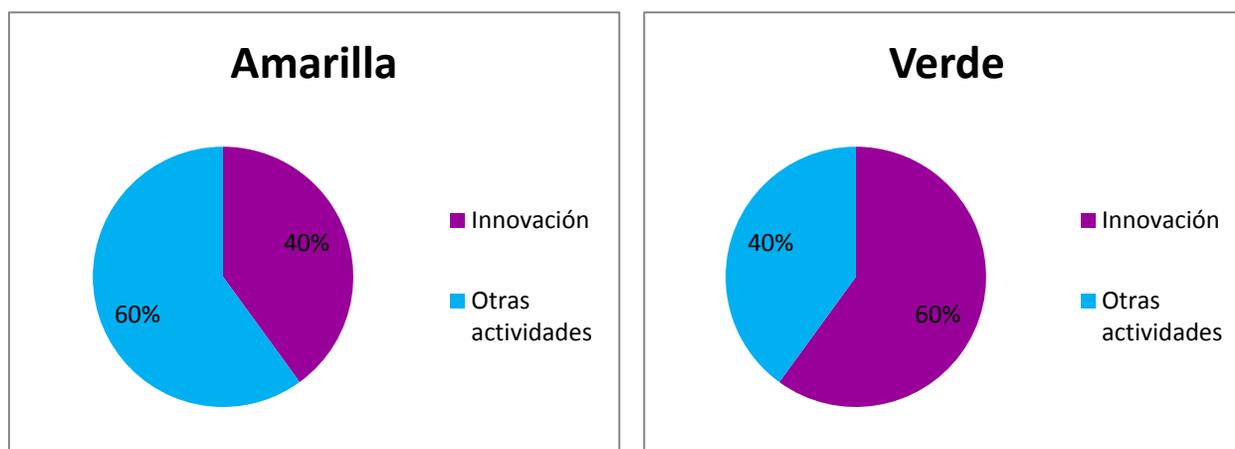
Gráfica 2. Importancia de la innovación para las empresas



Fuente: Elaboración propia con base en las entrevistas (2011).

Dos empresas proporcionaron porcentajes del tiempo dedicado a la innovación respecto al total de actividades.

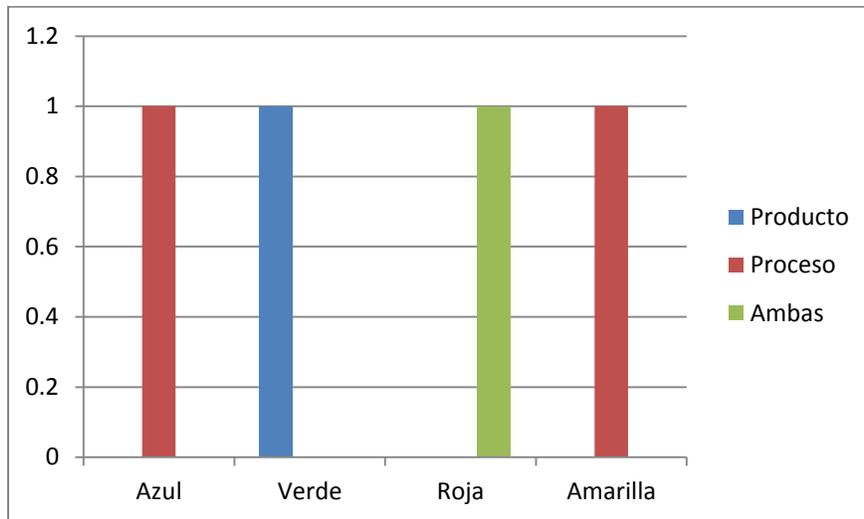
Gráfica 3. Tiempo dedicado a la innovación



Fuente: Elaboración propia con base en las entrevistas (2011).

Respecto a dónde se dan las mayores innovaciones las respuestas fueron:

Gráfica 4. Dónde se dan las mayores innovaciones



Fuente: Elaboración propia con base en las entrevistas (2011).

Respecto al proceso que realizan para innovar, las empresas en general lo identifican con su proceso para desarrollar oportunidades de negocio, lo cual se explica debido a la importancia que se le reconoce como muestra la Gráfica 2.

Para verde la innovación se da usando como base las necesidades de los clientes, y estudiando que es lo último en tecnología, software y aprovechando lo que existe aun cuando esté incompleto o mal hecho, son tecnologías que se pueden mejorar. La empresa Amarilla manifiesta que existe relación con universidades y centros de investigación con los cuáles se establece cooperación, se les solicita el desarrollo y la empresa actúa como supervisora.

En Azul, Amarilla, y Verde el responsable de la innovación es el dueño del negocio, en el caso de Roja, manifiesta que se buscan oportunidades de negocio y que para cada proyecto se asigna un líder que se encarga de desarrollarlo.

Las cuatro empresas manifestaron que identifican las oportunidades de negocio a través de redes de negocio, de las asociaciones a las que pertenecen, de sus

clientes e incluso de sus competidores. Para desarrollar las oportunidades de negocio identificadas.

La empresa Roja sigue el siguiente proceso.

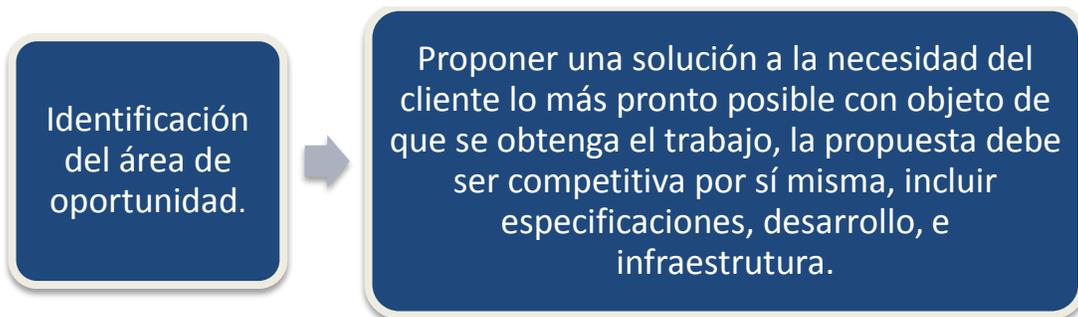
Figura 4. Proceso de respuesta a las oportunidades de negocio para la empresa Roja



Fuente: Elaboración propia con base en las entrevistas (2011).

Para Verde enfocada en *hosting* lo importante es el tiempo.

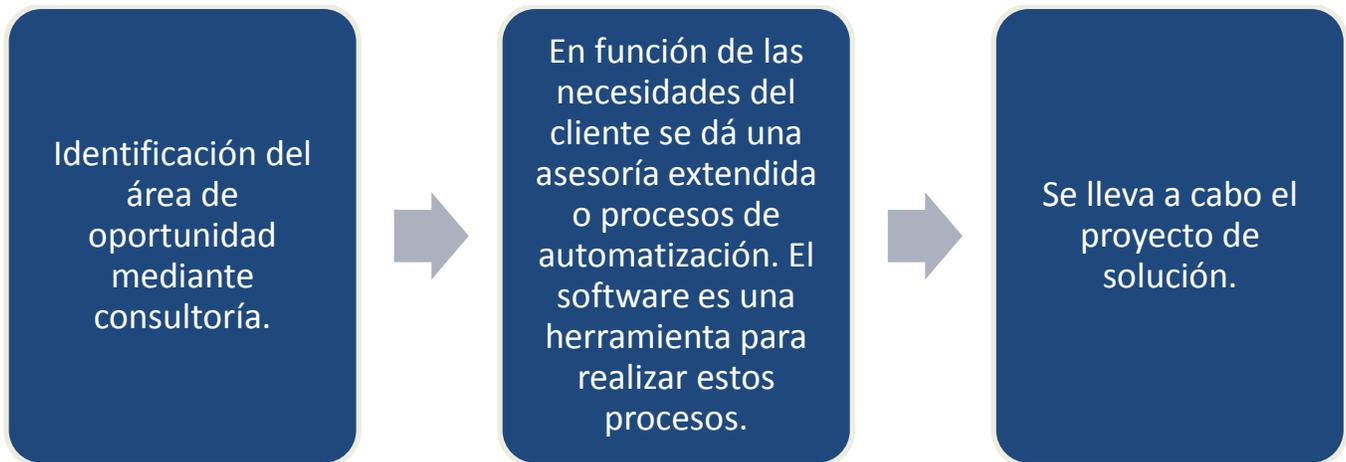
Figura 5. Proceso de respuesta a las oportunidades de negocio para la empresa Verde



Fuente: Elaboración propia con base en las entrevistas (2011).

Para Amarilla el proceso consiste en:

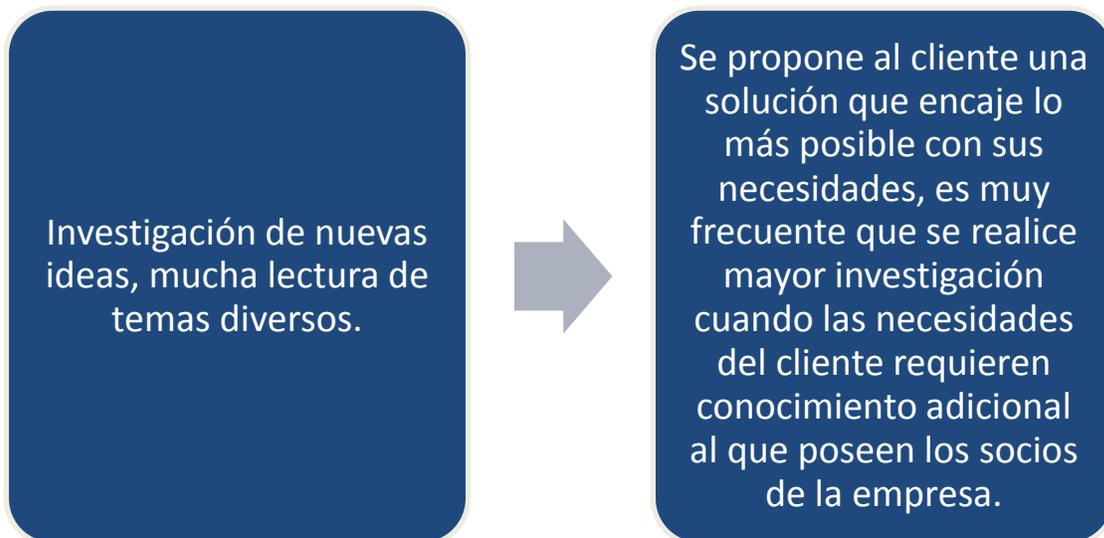
Figura 6. Proceso de respuesta a las oportunidades de negocio para la empresa Amarilla



Fuente: Elaboración propia con base en las entrevistas (2011).

Por su parte Azul establece relaciones muy estrechas con los clientes, para solucionar los problemas que tengan de cualquier índole. Su proceso puede representarse de esta manera.

Figura 7. Proceso de respuesta a las oportunidades de negocio para la empresa Azul



Fuente: Elaboración propia con base en las entrevistas (2011).

3.5. Relaciones con los usuarios

Una de las cosas que se afirman del software libre es la estrecha relación con sus usuarios. Al respecto, los empresarios son también usuarios del código fuente, los cuales tienen a su vez clientes finales a los que venden sus servicios. En este aspecto todas las empresas entrevistadas manifestaron tener contacto cercano con sus clientes, así como un sistema para procesar la información que los clientes les proporcionan respecto a sus productos o servicios. Cuando se proporciona el servicio de consultoría o cualquier otro, los clientes conservan una memoria técnica que permite comprender el desarrollo que se les diseñó, esto permite retomar un caso, o que los clientes busquen a otro proveedor que les brinde el servicio. Todas indican tener colaboración con otras entidades, como son los competidores, universidades, centros de investigación y con los miembros de las asociaciones a las que pertenecen.

Los cuatro empresarios reconocen el servicio personalizado como uno de los valores de sus empresas, eso nos da una idea de la importancia que asignan al conocimiento de las necesidades específicas de los clientes. Uno de ellos agrega que además existe confianza, lealtad mutua y eficiencia.

La empresa verde a su vez ha identificado dos tipos de clientes

Tabla 6. Tipos de cliente para la empresa Verde

Tipo de clientes	Resultados	
La mitad buscan únicamente una solución y no les importa la forma en que la empresa llega a ella.	Se obtienen subproductos que pueden ser utilizados para propósitos generales.	La empresa se ha enfocado más en tratar con este tipo de clientes que muchas empresas empiezan a rechazar.
La otra mitad busca el conocimiento dentro de la compañía.	Proveen información sobre problemas, cuya solución se convierte en una oportunidad.	

Fuente: Elaboración propia con base en las entrevistas (2011).

Indica también que su involucramiento en los proyectos no depende del sector económico en el que estén estos clientes, sino que está en función de la personalidad del cliente.

3.6. Protección del conocimiento

Como se observó en el marco teórico, la construcción del software abierto se caracteriza por la participación de distintos actores que no necesariamente pertenecen a una empresa determinada. Las empresas entrevistadas además de usar código externo son generadoras, por lo tanto se les preguntó de qué manera protegían el conocimiento que generaban, en forma de código, manuales, y demás.

Tabla 7. Formas de protección del conocimiento

Azul	Verde	Roja	Amarilla
1. El conocimiento es libre.	1. Compartiendo el conocimiento	1. Se libera código	1. Compartiendo el conocimiento, todo el código se libera, cursos, resultado de las investigaciones
2. Los manuales de tienen derechos de autor, son libres de uso pero no de lucro		2. Registro de marca	
		3. En el caso del Hardware con patentes	

Fuente: Elaboración propia con base en las entrevistas (2011).

Todos manifestaron que el conocimiento al liberarse se compartía con socios, clientes, competidores y otros usuarios. Esto en concordancia con la filosofía del software libre que considera que debe estar a la disposición de las personas que lo quieran utilizar. Sin embargo, hay que recordar que una de las características del software es su composición mediante módulos, aun cuando la totalidad del

código generado para un cliente específico pertenece a este, ello no impide que algunos módulos puedan ser utilizados en proyectos distintos.

En cuanto a la protección del código como se mencionó en el marco teórico el software libre puede liberarse mediante distintas modalidades de licencia que regulan la forma en que puede ser utilizado. Las cuatro empresas entrevistadas se decantan por el uso de las licencias GPL, BSD y MIT.

3.7. Formas de comunicación e intercambio de información

Finalmente, una de las características más importantes del software libre es la cantidad de información que los participantes intercambian a través de la red. Ha sido el desarrollo de internet y las herramientas de comunicación las que han permitido la colaboración de grandes grupos de personas que no comparten la misma ubicación geográfica. Todos los entrevistados mencionaron las herramientas de internet como medios de mantenerse actualizados, además de los congresos, asociaciones y demás, la red les permite estar en comunicación con otros desarrolladores, se mencionó que en ocasiones es posible contratar colaboradores que se encuentran en otro país y todo el intercambio se hace a través de la red. También los boletines de noticias, las redes sociales, sitios especializados, y libros constituyen formas de estar actualizados y de adquirir conocimiento. Los entrevistados manifestaron que era muy importante estar a la vanguardia de lo que sucede con respecto a la tecnología, y a distintos sectores que de alguna forma están vinculados con su negocio, uno de ellos menciona que sus intereses son tan amplios que se vincula con temas que a primera vista no parecen tener relación con su negocio pero que le sirven para diseñar soluciones a la medida de sus clientes.

4. Conclusiones

El propósito de este trabajo ha sido comprender de qué manera se lleva a cabo la innovación en las empresas que operan basadas en software libre. Lo que motivó inicialmente esta búsqueda fue la afirmación de que en el caso del software el aprendizaje se centraba más en los clientes y competidores. Se presentaron las bases teóricas que explican esta singularidad, así como las entrevistas realizadas para comprender cómo se está llevando la innovación en la práctica.

La industria del software es altamente dinámica y su desarrollo se encuentra estrechamente vinculado al crecimiento de otras áreas del sector electrónico informático, como son el hardware o el desarrollo de las telecomunicaciones y el internet, los cuáles han posibilitado por un lado contar con grandes capacidades de almacenamiento y procesamiento de información y por el otro rápido intercambio de información en tiempo real que permite reducir costos de envío así como tiempos de espera. Este crecimiento conjunto ha permitido generar una nueva forma de producción de software que se basa en el libre intercambio de información generada por usuarios que disponen de información y habilidades heterogéneas pero que se organizan en torno a un objetivo común. Esta forma de producción que Benkler (2002) denominó 'producción por pares' caracteriza lo que se ha denominado el software libre pero no le es exclusiva. El software libre se ha beneficiado de esta nueva forma de producción descentralizada y a simple vista anárquica, expandiéndose en todo el mundo y permitiendo utilizar las habilidades de programación de desarrolladores que no comparten ubicación geográfica, lo cual provoca que los desarrollos realizados en cualquier parte del mundo tengan aplicación y también competencia global.

En primer lugar la innovación comprendida como un atributo, proceso y resultado, se encuentra presente en el caso del software libre, cuando se genera un nuevo programa en su totalidad o cuando se genera código, el cual puede ser usado en un programa más grande, que resuelve un problema en una forma distinta. También resulta innovadora la forma en que las empresas basadas en software libre obtienen beneficios económicos, la lista presentada de modelos de negocios

continúa creciendo debido a las distintas respuestas ideadas por los empresarios para volver rentable un proyecto de software libre. Es importante recordar que existen distintos niveles de acción en esta industria, las empresas generadoras de código fuente no funcionan de la misma manera que las empresas desarrolladoras que prestan servicios relacionados y contribuyen a enriquecer el código fuente, también los colaboradores individuales juegan un papel distinto, ya que su participación puede darse una sola vez, pueden involucrarse frecuentemente sin buscar retribución monetaria, motivados únicamente por la 'gratificación del ego' o pueden buscar una retribución monetaria a su participación trabajando de forma temporal para proyectos específicos.

La producción por pares permite que los participantes se auto-adjudiquen los proyectos para los cuales están mejor capacitados, y para los cuáles también tienen la motivación suficiente. El hecho de que las contribuciones puedan ser de distintas magnitudes permite que los proyectos se mantengan vigentes. La publicación de las contribuciones de manera inmediata permite evitar la duplicidad de esfuerzos y la selección de las mejores soluciones. Esta participación abierta provoca que empresas competidoras puedan tener acceso a desarrollos generados por la compañía rival y viceversa. La constitución modular del software permite la generación de programas usando código proveniente de diversas fuentes. La colaboración entre empresas rivales se da de manera no precisamente organizada, sino por el hecho de que concurren en los mismos proyectos, y se retroalimentan mutuamente del conocimiento generado.

De igual manera, los casos observados permiten mostrar que la forma en que las empresas procesan la información generada en el exterior e interior de las mismas, difiere en función del tamaño de la empresa, el modelo de negocio bajo el cual opera, así como el servicio o producto que estén ofreciendo. Si hay algo que caracteriza al software libre es la disponibilidad de la información, esta característica ha hecho posible que se dé la llamada *democratizing innovation*. Los participantes de la industria, empresarios o no, pueden acceder a lo que necesiten sin grandes esfuerzos, así como contribuir a la construcción de software y hacer

saber al resto de los participantes en que ha consistido dicha contribución. Esta adjudicación está estrechamente vinculada con el incentivo señalizador mencionado en el capítulo dos.

La participación en los diferentes eslabones de la red de producción del software libre se encuentra motivada por razones de recompensa psicológica además de la monetaria. Lo cual deriva a su vez en formas diferentes de estimulación a la innovación y a la participación en dicha red, tomando un papel central el liderazgo, los beneficios postergados, el reconocimiento y el placer de resolver un problema. En esta industria es muy importante la atribución correcta de los proyectos para motivar mayores participaciones. De otra manera los contribuyentes individuales se verían desincentivados a compartir sus desarrollos, y les bastaría con obtener los beneficios de solucionar problemas individuales.

En el caso del software las posibles soluciones a problemas presentados, o las nuevas aplicaciones son infinitas, por lo cual resulta imposible que una sola organización satisfaga las necesidades totales de sus usuarios. Esto ocurre en todas las industrias, pero en la mayoría de ellas es difícil que un usuario modifique por sí mismo un producto para adaptarlo a sus necesidades específicas. Este problema queda superado en el software libre, debido a que los usuarios pueden modificar el código para adaptarlo a sus necesidades y después compartir su solución con otros usuarios que presenten las mismas necesidades. Es aquí donde pueden ser clasificados como usuarios líderes, debido a que para poder resolver un problema modificando código se requiere un nivel de conocimiento técnico, superior al de un usuario promedio. Además dado que otros usuarios probablemente presenten el mismo problema, primero habrá que hacer una búsqueda para saber si dicho problema ha sido resuelto, lo que requiere que estén informados acerca de lo que está sucediendo en la industria así como tener la motivación necesaria para solucionar el problema y hacer pública la solución.

En el caso de las empresas entrevistadas, a pesar de que tienen productos y modelos de negocio distintos, todas coinciden en ser una especie de intérprete de las necesidades de sus clientes, primero identifican lo que ellos requieren y

posteriormente ofrecen soluciones, utilizando el software abierto como herramienta. Es la forma en que ellos solucionan los problemas relacionados con la denominada '*sticky information*', dado que se requiere comprender las necesidades de los usuarios, cuando en ocasiones estos mismos las desconocen o no son capaces de hacérsela saber a las empresas. Las empresas Roja y Amarilla se involucran mucho al inicio de un proyecto en la identificación de las necesidades de los clientes, realizando diagnósticos y proponiendo respuestas, por lo declarado por ambas empresas se observa que existe una intensa comunicación con los clientes para proporcionarles lo que requieren. En el caso de azul, se da una interacción parecida caracterizada por el hecho de que esta empresa no está especializada en algún tipo de problema. Para verde, la interacción es fuerte también, pero como el empresario indicó los distintos tipos de clientes influyen en la relación que se establece, los clientes que sólo quieren la solución no generan una interacción tan cercana como aquéllos a los que les interesa la forma de encontrarla. Al observar la relación que existe entre los empresarios y sus clientes finales se puede notar que el involucramiento de estos últimos no es tan profundo como para caracterizarlos como usuarios líderes, sin embargo, en los casos en que se involucran profundamente contribuyen a generar soluciones que pueden tener una aplicación más amplia y no únicamente para el proyecto para el que fueron creadas. Es importante tener en cuenta que el producto final que se le otorga a estos clientes no es el software en sí, es más bien un conjunto de servicios en los cuales el software constituye solamente una parte de la solución global, ello también contribuye a explicar que esta relación sea más bien de índole tradicional, en la que si bien existe un fuerte intercambio de información la participación de los clientes en lo que concierne al software es más bien pequeña.

En cuanto a la forma en que protegen el conocimiento generado todos los empresarios entrevistados respondieron que la mejor forma de proteger lo que producían era hacerlo público, el código generado tiene la firma de su creador lo que permite adjudicarle la autoría. Al hacer público determinado código se garantiza que alguien más no se adjudique este conocimiento, así pasa con otros

productos generados como manuales, cursos y demás. Esto es posible debido a que como vimos los modelos de negocio no dependen de la venta del software.

Los modelos de negocio presentados por las empresas nos muestran que en la mayoría de los casos están basados en consultoría, primero se identifican los problemas de un cliente y en función de estos se les ofrece una solución global que incluye una aplicación de software libre. También están los cursos y manuales, lo más importante no es el software sino qué es lo que las empresas son capaces de hacer con él, cuan bien identifican las necesidades de sus clientes y cuan ideal es la solución generada para satisfacerlas. La existencia de las memorias técnicas permite que los empresarios y los clientes sepan cómo dichas necesidades están siendo satisfechas, e incluso permite que si el cliente no está de acuerdo con la solución pueda acudir a otra empresa para que realice el trabajo. De manera que la innovación permite a las empresas resolver los problemas de sus clientes de manera rápida y con lo último que existe en el mercado, pero no garantiza la fidelidad de su cliente, ya que otras empresas tienen acceso a la misma información. Pero sí constituye un factor fundamental para no ser expulsados del mercado, ya que no pueden ofrecer una solución obsoleta o en desuso, porque los competidores ofrecerán una versión más reciente y seguramente con un desempeño superior. Ello explica la razón de que se le destine una gran cantidad de recursos, ya sea tiempo o dinero, a la búsqueda de innovación. Cuando se genera un producto totalmente nuevo, este tendrá una vigencia de apenas unos cuantos meses y muy pronto será replicada, por lo tanto, el conocimiento y la innovación son fundamentales para la supervivencia de las empresas.

Otra de las observaciones obtenidas de las entrevistas, se da en cuanto a la relación con los colaboradores de las empresas, debido a que en la mayoría de los casos, las relaciones al interior de las empresas siguen la estructura jerárquica de las empresas tradicionales. En cuanto a la motivación de los colaboradores, todos los empresarios reconocen su importancia y coinciden en manifestar que el motivador principal no es el salario, resalta el hecho de que uno de ellos considere

la edad como factor de influencia en las motivaciones. De manera que a temprana edad los incentivos de gratificación del ego tienen mayor importancia que los concernientes a la carrera. Resultaría interesante conocer el promedio de edad de los usuarios y desarrolladores del software libre para saber si existe una correlación entre esta y su participación en el desarrollo del mismo. Del mismo modo destaca el liderazgo al interior de la empresa, la mayoría de las empresas manifestaron que se basaba principalmente en el conocimiento, que este era un factor muy importante para que alguien liderara algún proyecto. Así mismo, indicaron que el liderazgo de la empresa se daba de manera lineal, incluso utilizaron la palabra dictadura. Por lo tanto tenemos un liderazgo autocrático basado no en la fuerza, sino en el conocimiento y la habilidad técnica.

También se observa una fuerte carga ideológica en la participación de la industria del software libre. A pesar de las diferencias de concepción entre el concepto de software libre y abierto, ambos comparten el reconocimiento de la libertad como un valor fundamental en la industria. Estas formas de concepción se encuentran reflejadas en las distintas versiones de licencia que regulan la forma en que se comparten el conocimiento generado por los participantes del software libre. Esto también caracteriza a la industria, ya que no se trata de legislación generada por el Estado, sino de una solución generada por los propios participantes, y los cuáles tienen la libertad de elegir que licencia adoptar en función de sus necesidades así como de su ideología. Respecto a esto último, el movimiento FLOSS es de alcance global e involucra actividades de activismo que escapan el alcance de este trabajo pero que es importante conocer para comprender su desarrollo y la forma en que está creciendo incorporándose a las políticas públicas en lugares como España, que se ha convertido en uno de los principales generadores de software libre a nivel mundial.

El hecho de que importantes empresas de la industria del software y el hardware hayan integrado a sus estrategias alguna forma de software abierto parece indicar que existen beneficios únicos que éste proporciona y los cuáles estas empresas quieren aprovechar, incluso, sin abandonar sus modelos de negocio tradicionales.

Por lo tanto antes de pensar en que el modelo del software abierto sustituirá al modelo propietario tradicional es posible pensar en que ambos modelos coexistirán para satisfacer las necesidades de los usuarios.

Bibliografía

- ARECHAVALA, Ricardo., MONTOYA Luz María, RODRIGUEZ Gerardo y SAMPEDRO José Luis. (2010). *La industria del software en el mundo*. En HUALDE A. (Coord.), *La industria del software en Baja California y Jalisco* (pp 29-62). México: El Colegio de la Frontera Norte, Textual, UAM
- ALEIXANDRE, Guillermo. (2002). *Las estrategias para la innovación tecnológica en castilla y león*. Tesis de doctorado. Universidad de Valladolid. Valladolid. España
- BELL, Martin y PAVITT, Keith. (1995). *The development of Technological Capabilities*. En I. Ul Haque (ed). *Trade, Technology and International Competitiveness*. (pp 69-101). Washington D.C.: Banco Mundial, IDE.
- BENKLER, Yochai (2002). *Coase's Penguin, or, Linux and the Nature of the Firm*. *The Yale Law Journal* 112(3)
- CENATIC (2008) *Guía básica del software de Fuentes abiertas*. España: CENATIC
- DABAT, Alejandro. y ORDOÑEZ, Sergio. (2009). *Revolución informática, nuevo ciclo industrial e industria electrónica en México*. México: UNAM, Casa Juan Pablos.
- DRUCKER, Peter. (1998, November-December). *The Discipline of Innovation*. En *Harvard Business Review*, 3-8
- ESCORSA Pere. y VALLS, Jaume. (2003). *Tecnología e innovación en la empresa*. España: EDICIONS UPC.
- FAGERBERG, Jan. (2003). *Innovation: A guide to the literature*. Oslo. University of Oslo.

- GONZÁLEZ, Rocío y GARCÍA, Fernando (2011) *Innovación abierta: un modelo preliminar desde la generación del conocimiento*. Intangible capital, vol 7, núm 1, p 82-115
- HARISON Elad. y KOSKI, Heli. (2010). *Applying open innovation in business strategies: Evidence form Finnish software firms*. Research Policy, Volume 39, Issue 3, pages 351-359.
- LERNER, Josh y TIROLE, Jean. (2000) *The simple Economics of Open Source*. NBER Working paper No 7600.
- LEE, Matt. (2010) *What is free software and why is it so important for society?* Free Software Foundation, en <http://www.fsf.org/about/what-is-free-software>. Consultado el 3 de Abril de 2011.
- LIPPOLDT, Douglas y STRYSZOWSKY, Piotr. (2009) *Innovation in the Software Sector*. OECD innovation strategy
- MEDELLÍN, Enrique. (2008). *El conocimiento y su administración en las empresas*. En MICHELI, Jordy., MEDELLÍN, Enrique., HIDALGO, ANTONIO, Hidalgo. y JASSO, Javier., (Coord.), *Conocimiento e innovación: retos de la gestión empresarial* (pp 57-89). México: UAM, UNAM y Plaza y VALDÉS.
- MORRIS, Langdon. (2003). *Business Model Warfare*. USA: University of Pennsylvania
- OSTERWALDER, Alexander. y PIGNEUR Yves. (2009). *Business Model Generation*. USA: Self Published.
- PORTER, Michael. (1996, November-December). *What is Strategy?* En Harvard Business Review, 61-78
- RAYMOND, Eric. (2001). *The Cathedral and the bazaar: Musings on Linux and Open Source by an Accidental Revolutionary*. O'Reilly: Sebastopol.
- ROSSI, Andrea., GONZALES, Pascual., IGLESIAS, Carlos y ROMERA, Teo. (2008). *Modelos de negocio para SW libre*. Ministerio de Industria, Turismo y Comercio español.

STALLMAN, Richard. (2011). *The Free Software Foundation Management*. Free Software Foundation, en <http://www.fsf.org/about/leadership/>. Consultado el 7 de Abril de 2011.

The Open Source Definition. n.d. Open Source Initiative en <http://www.opensource.org/docs/osd>. Consultado el 3 de Abril de 2011.

TEECE, David J. (1986). *Profiting from technological innovation: Implications for integration, collaboration, licensing and public policy*. USA: University of California Berkeley.

VON HIPPEL, Eric. (1988). *The sources of Innovation*. New York: Oxford University Press

VON HIPPEL, Eric. (2005). *Democratizing Innovation*. Massachusetts: The MIT Press

YOGUEL, Gabriel. (2000) "Creación de Competencias en Ambientes Locales y Redes Productivas". *Revista de la CEPAL* No. 71. Agosto de 2000.

Anexos

Con el fin de conocer el comportamiento de las empresas y verificar su correlación con la teoría, se realizaron entrevistas a cuatro empresas que operan utilizando software libre.

El contacto con estas empresas se logró gracias al apoyo de Daniel Ceballos presidente de la Asociación mexicana de software libre (AMESOL). Se buscó que las empresas tuvieran tamaños, productos y modelos de negocio diferentes. También fue muy importante la disposición de los empresarios a responder las preguntas del cuestionario aplicado.

Fueron entrevistadas cuatro empresas a las que se denominó Azul, Verde, Roja y Amarilla por motivos de confidencialidad.

El contacto con la AMESOL se inició en Febrero de 2011 y permitió el acceso a información relacionada con la industria y a la construcción del marco teórico de este trabajo. Así como la construcción de un cuestionario enfocado en los temas más relevantes para la investigación evitando la inclusión de preguntas innecesarias o que resultaban obvias para quién tuviese conocimientos básicos de la industria. Como resultado de esta interacción se llevaron a cabo las entrevistas las cuales fueron aplicadas entre Julio y Septiembre de 2011, de manera directa, utilizando un cuestionario de cuarenta y dos preguntas de modalidad abierta y cerrada.

Con las entrevistas se buscaba obtener la siguiente información:

- Perfil de la empresa y del empresario. Tiempo de vida, modelo de negocio, principal cliente y producto.
- Motivación en la industria del software libre. La motivación para participar en la industria del software libre, así como para innovar.
- Intercambio del conocimiento. De qué manera se protege el conocimiento generado al interior de las empresas, y cómo se comparte con otros usuarios.
- Metodología de innovación. De qué manera llevan a cabo la innovación al interior de las empresas, si poseen una metodología definida o no.

- Relaciones con clientes y competidores. De donde obtienen oportunidades de negocios y cómo se comunican con sus clientes y cómo obtienen retroalimentación de ellos.

Para las empresas Azul, Verde y Amarilla se entrevistó a los dueños de las mismas, únicamente en el caso de Roja se entrevistó a un líder de proyecto. A continuación se presenta el cuestionario aplicado.

Cuestionario

Perfil del empresario

- 1.- Nombre del empresario:
- 2.- Edad:
- 3.- Formación académica:
- 4.- ¿Qué funciones desempeña en la empresa?

Dirección
RH
Contabilidad
Ventas

Perfil de la empresa:

- 5.- Nombre:
- 6.- Año de fundación:
- 7.- ¿Cuál es su modelo de negocio?
- 8.- ¿Puede explicarlo brevemente?
- 9.- ¿Cuál es su producto principal?
- 10.- ¿Cuál es el número y perfil académico de sus empleados:

Técnicos: Soporte:

Técnicos:	Soporte:

Cursos
Licenciatura
Maestría
Doctorado
Otro

- 11.- ¿Quiénes son sus clientes principales?
- 12.- ¿Cuál es el porcentaje de ingresos generado por los clientes principales?
- 13.- ¿Pertenece a alguna asociación?
- 14.- ¿Desde cuándo?

15.- ¿Qué áreas conforman a la empresa?

Dirección	<input type="checkbox"/>
Contabilidad	<input type="checkbox"/>
RH	<input type="checkbox"/>
Desarrollo	<input type="checkbox"/>
Ventas	<input type="checkbox"/>
Otros	<input type="checkbox"/>

Acerca del negocio

16.- ¿Por qué eligió hacer negocios con software libre?

17.- ¿Qué valores posee su negocio?

Puntualidad	<input type="checkbox"/>
Servicio postventa	<input type="checkbox"/>
Servicio personalizado	<input type="checkbox"/>
Otro	<input type="checkbox"/>

18.- ¿Cómo identifica oportunidades de negocio?

Competidores	<input type="checkbox"/>
Clientes	<input type="checkbox"/>
Redes	<input type="checkbox"/>
Congresos	<input type="checkbox"/>
Otros	<input type="checkbox"/>

19.- ¿Cómo desarrolla estas nuevas oportunidades?

20.- ¿Cuán importante considera la innovación en su negocio?

El negocio depende de ello	<input type="checkbox"/>
Muy importante	<input type="checkbox"/>
Importante	<input type="checkbox"/>
Útil	<input type="checkbox"/>
No es fundamental	<input type="checkbox"/>

21.- ¿De qué manera lleva a cabo la innovación en su negocio?

22.- ¿Existe un responsable de este proceso?

23.- ¿Quién es?

24.- ¿En dónde se dan las mayores innovaciones?

Proceso
Producto
Marketing
Organización

25.- ¿De qué manera protege el conocimiento que genera su empresa?

26.- ¿Con quienes comparte el conocimiento que genera su empresa?

Colaboradores
Socios
Competencia
Clientes
Otros usuarios

27.- ¿Con que modelo de licencia opera sus desarrollos de software?

28.- ¿De qué manera se comunica con sus usuarios/clientes?

29.- ¿Le proporcionan retroalimentación respecto a los productos o servicios que usted brinda?

30.- ¿Tiene algún proceso establecido para procesar esta información?

31.- ¿Puede proporcionar un ejemplo?

32.- ¿En caso afirmativo, qué hace con esta información?

33.- ¿Tiene colaboración con otras entidades?

Competidores
Universidades
Centros de
investigación
Otros

34.- ¿Cuál es la ventaja competitiva de su negocio?

35.- ¿De qué manera se ejerce el liderazgo en su empresa a nivel dirección?

36.- ¿Qué perfil considera que se requiere como líder?

Iniciativa	<input type="checkbox"/>
Conocimiento técnico	<input type="checkbox"/>
Autoridad formal	<input type="checkbox"/>
Empatía	<input type="checkbox"/>
Otros	<input type="checkbox"/>

37.- ¿De qué forma motiva a sus colaboradores?

Acciones	<input type="checkbox"/>
Salario	<input type="checkbox"/>
Reconocimiento	<input type="checkbox"/>
Ascensos	<input type="checkbox"/>
Otros	<input type="checkbox"/>

38.- ¿Cuál considera la principal ventaja del software libre sobre el software propietario?

Costos	<input type="checkbox"/>
Eficiencia	<input type="checkbox"/>
Rapidez	<input type="checkbox"/>
Otros	<input type="checkbox"/>

39.- ¿Considera que existen aplicaciones idóneas del FLOSS o cree que puede ser usado en cualquier aplicación?

40.- ¿De qué manera se actualiza respecto a lo que sucede en la industria?

Congresos	<input type="checkbox"/>
Cursos	<input type="checkbox"/>
Boletines especializados	<input type="checkbox"/>
Redes laborales	<input type="checkbox"/>
Otros	<input type="checkbox"/>

41.- ¿Qué características toma en cuenta cuando contrata a alguien?

Técnicas

Dominio de programas
Escolaridad
Conocimientos
especializados
Idiomas
Otros

De personalidad

Iniciativa
Creatividad

Contactos personales
Trabajo en equipo
Otros

42.- ¿Cuáles consideraría, de mayor a menor importancia, las motivaciones de sus colaboradores para innovar?

Salario
Sentido de logro
Búsqueda de
reconocimiento
Aprendizaje
Solucionar una necesidad
particular
Participación en la
empresa
Otros
