



**DISEÑO Y EVALUACIÓN DE UN CURSO E-LEARNING PARA PROMOVER
COMPETENCIAS LABORALES PARA LA PROMOCIÓN DE LA
COOPERACIÓN EN EL DESARROLLO DE PROYECTOS DE SOFTWARE
LIBRE**

Jairo Alonso García Arias

**Universidad Autónoma de Bucaramanga
Facultad de Ingeniería de Sistemas
Magister en Software Libre
Bogotá, Colombia
2013**

**DISEÑO Y EVALUACIÓN DE UN CURSO E-LEARNING PARA PROMOVER
COMPETENCIAS LABORALES PARA LA PROMOCIÓN DE LA
COOPERACIÓN EN EL DESARROLLO DE PROYECTOS DE SOFTWARE
LIBRE**

Jairo Alonso García Arias

**Tesis presentada como requisito parcial para optar el título de:
Magister en Software Libre**

**Director:
Ph.D., Jorge Andrick Parra Valencia**

**Línea de Investigación:
Pensamiento Sistémico**

**Facultad de Ingeniería de Sistemas
Magister de Software Libre
Bogotá, Colombia
2013**

Página de aceptación

Aprobado por el Comité de Grado en cumplimiento de los requisitos exigidos por la Universidad Autónoma de Bucaramanga para optar al título de Magíster en Software Libre.

Ph.D., Jorge Andrick Parra Valencia

Director

Bucaramanga, junio de 2013

Dedicatoria

A todas las personas que se vieron involucradas en el desarrollo de esta etapa de mi vida y al mundo del software libre.

Agradecimientos

Agradezco a todas las personas que se han visto de una u otra forma involucradas en el desarrollo de esta etapa de mi vida durante los últimos 3 años con sus consejos, colaboraciones, ayuda y voces de ánimo para llevar a feliz término los estudios del master.

Agradezco de manera especial a Jorge Andrick el tiempo dedicado en la guía del presente trabajo para de una u otra forma darle sentido al mismo.

Resumen

En este proyecto se desarrolla un proceso de recolección de información y diagnóstico de dos proyectos de software libre para determinar las competencias laborales enfocadas a promover la cooperación en proyectos de software libre.

Adicional con los resultados obtenidos se propuso un diseño de curso e-learning apoyado en el esquema de la Universidad del Quindío de la ciudad de Armenia.

CONTENIDO

	pág.
AGRADECIMIENTOS	6
RESUMEN	8
CONTENIDO	9
LISTA DE TABLAS	11
LISTA DE FIGURAS	12
1. INTRODUCCIÓN	13
2. MOTIVACIÓN Y JUSTIFICACIÓN	15
3. MARCO TEÓRICO	16
3.1 EL CONCEPTO DE COMPETENCIAS LABORALES	18
3.2 EL CONCEPTO DE DESARROLLOS RÁPIDOS	18
3.3 DILEMAS SOCIALES	19
3.3.1 Dilemas sociales en el desarrollo de software libre	20
3.4 COOPERACIÓN	23
3.5. SOFTWARE LIBRE COMO BIEN PUBLICO	24
4. MÉTODO DE INVESTIGACIÓN	27
4.1 FORMULACIÓN DECLARACIÓN DEL PROBLEMA	27
4.2 OBJETIVO GENERAL	28
4.3 OBJETIVOS ESPECÍFICOS	28
4.3 METODOLOGÍA Y DESARROLLO DEL TRABAJO	29
4.3.1 Secuencia descriptiva de pasos	30
4.3.2 Diseño de instrumentos que integrarán la metodología (Entrevistas, encuestas, observación, etcétera)	30
4.3.3 Técnica usada para la recopilación de datos	30
4.3.4 Cómo se efectuó el análisis	32
5. RESULTADOS	34
5.1 RESULTADOS ENCUESTA	34
5.2 DIAGNÓSTICO: PROYECTOS DE SOFTWARE LIBRE	42
5.3 DISCUSIÓN DE RESULTADOS	50

5.3.1	<i>Encuesta</i>	50
5.3.2	<i>Diagnóstico de Proyectos de Proyectos de Software Libre</i>	51
5.4	CONCLUSIÓN DE LOS RESULTADOS Y COMPETENCIAS LABORALES DEFINIDAS	53
5.4.1	<i>Competencias Laborales</i>	53
5.4.2	<i>Cooperación Activa</i>	54
5.5	DISEÑO DEL CURSO DE E-LEARNING	56
5.5.1	<i>Información General. Identificación del Espacio Académico</i>	56
5.5.2	<i>Justificación</i>	57
5.5.3	<i>Propósito u Objetivo General</i>	57
5.5.4	<i>Objetivos Específicos</i>	58
5.5.5	<i>Contenidos del Curso (Saber)</i>	58
5.5.6	<i>Contenidos Procedimentales (Saber Hacer)</i>	59
5.5.7	<i>Contenidos Actitudinales (Ser)</i>	59
5.5.8	<i>Estrategias Metodológicas</i>	59
5.5.9	<i>Sistema de Evaluación</i>	60
6.	DISCUSIÓN, TRABAJO FUTURO Y CONCLUSIONES	61
6.1	CUMPLIMIENTO DE OBJETIVOS	61
6.2	DISCUSIÓN SOBRE LA FORMULACIÓN DEL PROBLEMA	62
6.3	APORTES	62
6.4	TRABAJO FUTURO	64
6.5	CONCLUSIONES	65
7.	BIBLIOGRAFÍA	68
8.	ANEXOS	69
8.1	ENCUESTA DILEMAS PROYECTOS DE SOFTWARE LIBRE	69

LISTA DE TABLAS

	pág.
Tabla 1. Dilemas sociales enfocados al desarrollo de software	23
Tabla 2. Cooperación Dilema del Prisionero A	23
Tabla 3. Cooperación Dilema del Prisionero B	24
Tabla 4. Datos Empresas para Encuesta	31
Tabla 5. Diagnóstico proyecto OpenOffice	46
Tabla 6. Diagnóstico proyecto GNU Telephony.....	49
Tabla 7. Competencias laborales definidas	54
Tabla 8. Diseño del curso. Información académica	56
Tabla 9. Diseño del curso. Estrategias metodológicas	60
Tabla 10. Cumplimiento de objetivos	62

LISTA DE FIGURAS

	pág.
Figura 1. Desarrollo de Proyectos de Software Libre	16
Figura 2. Razones para formar grupos de trabajo	17
Figura 3. Limitantes de los grupos de trabajo	17
Figura 4. New Commons	25
Figura 5. Encuesta. Genero	34
Figura 6. Encuesta. Rango de Edades	34
Figura 7. Encuesta. Etapas clásicas de un proyecto	35
Figura 8. Encuesta. Consideración de la comunicación.....	35
Figura 9. Encuesta. Dilemas en el desarrollo de proyectos.....	36
Figura 10. Encuesta. Manejo del idioma.....	37
Figura 11. Encuesta. Dilemas en el análisis y diseño	37
Figura 12. Encuesta. Dilemas en el desarrollo o codificación	38
Figura 13. Encuesta. Dilemas en la implementación y/o pruebas.....	39
Figura 14. Encuesta. Dilemas en el control de cambios	40
Figura 15. Encuesta. Estrategias para afrontar los dilemas.....	40

1. Introducción

Es realmente importante que se logre identificar apropiadamente el problema que incentiva el desarrollo de este proyecto, pues el problema en sí es el que le da sentido al mismo. Como se menciona más adelante, el proceso de desarrollo no garantiza la calidad de la información contenida o la fiabilidad en los objetos de aprendizaje, pero seguramente generará espacios de debate acerca de los temas tratados y las conclusiones generadas a partir del desarrollo del mismo.

Es deseable que una vez se logre determinar el objeto de este proyecto se continúe con una mirada rápida a la literatura propuesta en la bibliografía para poder clarificar los conceptos teóricos adoptados en el marco teórico. Con la realización del proceso anterior se puede llevar a cabo el proceso de verificación de los resultados de desarrollo del proyecto y lo ejecutado para lograr deducir los comportamientos contrarios que impactan el proceso de cooperación en los proyectos de software libre.

Una vez se tiene el fundamento necesario, se invita al lector a visualizar los resultados obtenidos en la ejecución de la encuestas y en el diagnóstico de los proyectos de software libre OpenOffice y GNU Telephony. Con ello y después de observar las conclusiones preliminares de dichos procesos, se puede observar las competencias laborales que permiten la promoción de la cooperación en los proyectos de software libre.

Finalmente este proyecto termina con la propuesta de un diseño de curso de e-learning aplicando las competencias laborales, la discusión del proceso desarrollado, los aportes, las conclusiones y un trabajo futuro que le permiten visualizar los logros alcanzados y la futura investigación a plantear.

Clasificación de los documentos bibliográficos

Documento	Texto de conocimiento avanzado	Texto a base de experiencias	Texto de referencias y conocimiento general
SALCEDO MALDONADO, Jorge 2009. Dilemas de Acción Colectiva: Instituciones y Cooperación en Organizaciones Tecnológicas Redes para Innovación.	X		
MCCONNE, David 2006. E-Learning Groups and Communities. David Mcconne.		X	
AXELROD, R. & HAMILTON, W. 1981. The evolution of cooperation			X
BUCK, S. (1998). The global commons: an introduction. Island Press			X
KOLLOCK, Peter 1998. Social Dilemmas: The Anatomy of Cooperation	X		
HARDIN, Garrett 1968. The Tragedy of Commons. "Science"		X	
FUNDACION EOI, 2009. La oportunidad del software libre: capacidades, derechos e innovación.			X

2. Motivación y Justificación

Es importante la investigación, ya que al poder identificar los diferentes dilemas sociales que enfrentan los desarrolladores en los proyectos de software libre, se pueden identificar las competencias laborales necesarias para alcanzar el éxito y todo ello puede ser dado a conocer usando las herramientas disponibles en Internet. Todo esto es esencial a la hora de buscar una mejor colaboración por parte de los integrantes del proyecto buscando la armonía necesaria que ayude al desarrollo y culminación de forma exitosa.

De forma adicional permitió la generación de un diseño de curso e-learning como una de esas herramientas para divulgar el conocimiento a disposición de todo el mundo sin restricción alguna.

Por otro lado se pretendió basados en la metodología de desarrollo de software rápido, analizar el desarrollo de proyectos e involucrar los hallazgos en el curso de e-learning.

3. Marco Teórico

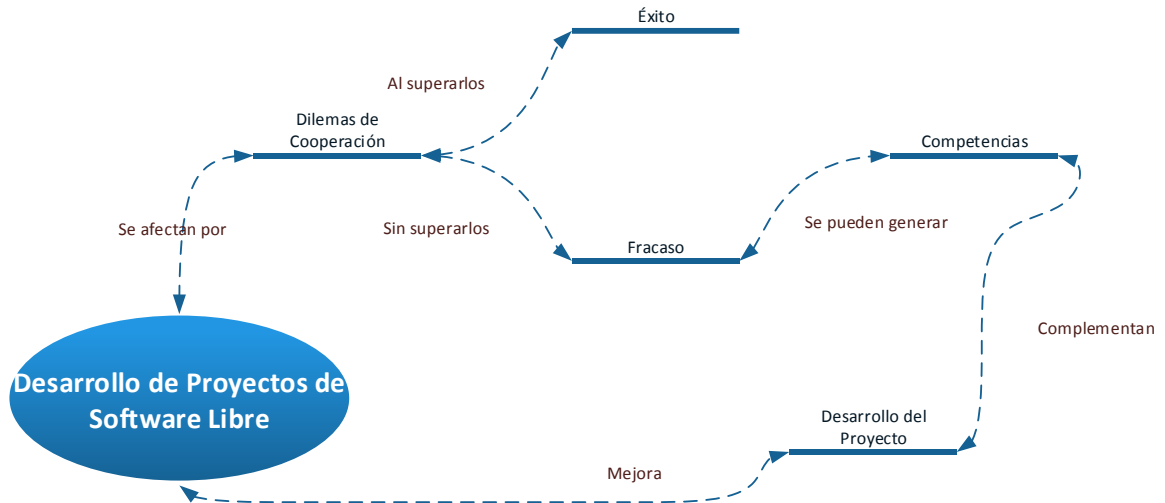


Figura 1. Desarrollo de Proyectos de Software Libre

El diagrama anterior presenta de forma general lo que sucede en el desarrollo de proyectos de software libre cuando son impactados por los dilemas de desarrollo y en donde si no se logran superar como regla general conllevan al fracaso. Sin embargo es desde éste mismo fracaso que se espera poder llevar a cabo la investigación que permita arrojar los elementos faltantes para poder deducir las competencias laborales que de la mano a una buena articulación con desarrollos rápidos o ágiles permitan la mejora del desarrollo de los proyectos y en consecuencia un valioso conocimiento que pueda llegar a ser plasmado en un curso de e-learning diseñado y evaluado para que pueda llegar a cumplir dichos objetivos. Todo como se pretende sea desde la óptica de la cooperación.

- Razones para formar grupos de trabajo basados en cooperación.

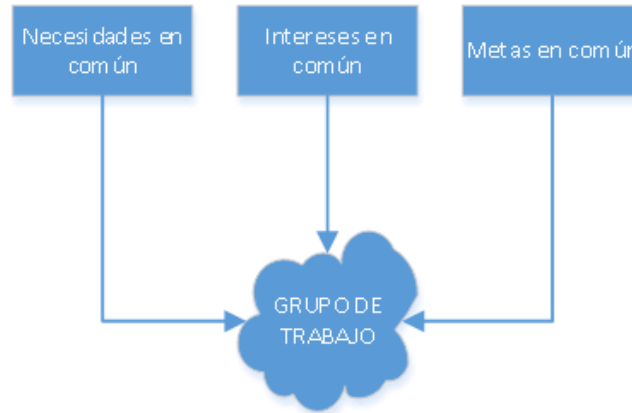


Figura 2. Razones para formar grupos de trabajo

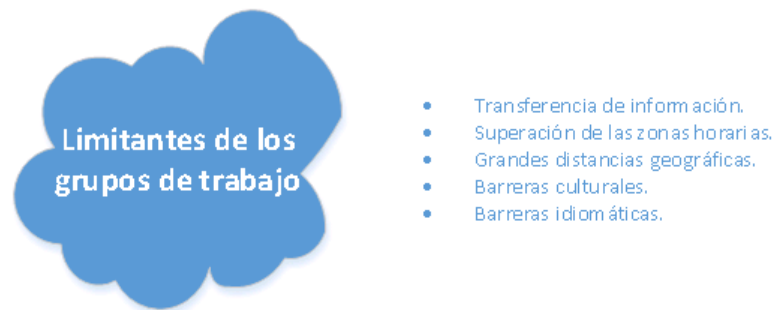


Figura 3. Limitantes de los grupos de trabajo

En consecuencia y según lo visto hasta este momento se podría preguntar:

¿Qué tan impactados son los proyectos de software libre por los dilemas sociales?

¿Qué motiva a las personas a cooperar de proyectos de software libre?

¿Se puede tomar los proyectos fracasados de software libre para determinar en qué punto fallaron y poder a partir de ello generar las competencias necesarias y que estas puedan ser entregadas a la comunidad empleando tecnología de e-learning?

¿Se pueden tener proyectos exitosos en cuanto a la cooperación?

¿Puede la teoría de la cooperación explicar el éxito o el fracaso de la cooperación en proyectos de software libre?

3.1 El concepto de competencias laborales

Las competencias laborales son los conocimientos técnicos, las actitudes personales y las aptitudes profesionales útiles para el desempeño de una persona en ambientes productivos y que se obtienen en el transcurso de su desarrollo educativo como la cimentación más importante para que las mismas se desarrollen en el ámbito productivo.

Esto es entonces una compleja red de atributos que ayudan a la persona a desarrollarse en las diferentes situaciones y que le permiten resolver problemas de tipo profesional de forma autónoma y flexible.

3.2 El concepto de desarrollos rápidos

Las metodologías de desarrollo rápidos consisten en procedimientos y buenas prácticas que permiten reducir el ciclo de vida del software y por lo tanto permite la entrega de diversos componentes de forma iterativa por etapas en un modelo evolutivo que facilita obtener versiones

previas a lo que será el resultado final y que de forma adicional permite correcciones de forma ágil.

Estos modelos en la actualidad son empleados no solo en el desarrollo de software sino también adaptados a otra clase de proyectos dada su flexibilidad y a que por los tiempos en los cuales se llevan a cabo se amoldan a las situaciones que se viven hoy en día por la premura de muchos de los proyectos.

3.3 Dilemas sociales

Los dilemas sociales son situaciones donde para cumplir un objetivo personal es necesario contar con la intervención de otros elementos públicos por llamarlos de alguna forma. Otras definiciones formales dadas a lo largo de la historia son donde generalmente una racionalidad lleva una irracionalidad colectiva (Schlager). De igual manera existen teorías generados a partir de los dilemas como el problema de los bienes públicos y colectivos (Samuelson; Olson), el dilema del compromiso creíble (Williamson), el problema del desertor (Grossman & Hart), entre otros.

En todos ellos se describen los posibles resultados dependiendo de la decisión de los individuos de decidir cooperar o no.

3.3.1 Dilemas sociales en el desarrollo de software libre

En este apartado se trataron los dilemas sociales definidos en el entorno de desarrollo de software libre a través de la encuesta realizada.

Dilema social	Conflicto de Racionalidad	Estrategias de Solución
Falta de liderazgo en el grupo	La falta del liderazgo en el proyecto repercute en la deserción de los integrantes y por ende la pérdida de interés en el mismo.	Incentivar entre los integrantes la toma de iniciativa y el liderazgo. Identificar las personas que más aportan al proyecto y promoverlas a líderes.
Falta de participación	La falta de participación genera que el desarrollo de un proyecto de software se torne lento y por ende poco atractivo a sus integrantes que a la final tienden a desertar del mismo.	Fomentar el sentido de pertenencia de las personas que integran el proyecto. Incentivar la participación de las personas del proyecto a través de reconocimientos dentro del mismo grupo y en los diferentes medios de divulgación del proyecto.
Falta de divulgación del proyecto	La no divulgación o escasa divulgación de un proyecto de	Publicitar el proyecto en todos los espacios

	software libre por más interesante y excitante que resulte para un desarrollador es poco si el mismo no se entera de su existencia. Esto genera que no tenga apoyo por parte del desarrollador.	posibles gratuitos en internet. Generar divulgación del mismo en redes sociales a través de los integrantes del proyecto.
Carencia de herramientas de participación o comunicación grupal	Un proyecto que carezca de herramientas de comunicación va en contra de la voluntad de los desarrolladores a la hora de querer dar a conocer sus necesidades, dificultades y logros.	Búsqueda de herramientas de comunicación modernas de uso libre que de forma adicional den a conocer el proyecto. Un ejemplo de este tipo de herramientas podría ser la suite de comunicación de Google.
Carencia de información	Un desarrollador o grupo de desarrolladores sin la información adecuada, pueden llegar a generar diversas versiones de un mismo elemento que va en contra de la uniformidad del proyecto.	Utilización de sistemas de versionamiento, gestión documental o administración de proyecto. Documentación de los procesos, reuniones y
Interpretación errónea de los individuos	Junto con la carencia de información, la interpretación	

	errónea se presenta por dos factores (1. Falta de liderazgo, 2. Falta de organización). En ambos casos se puede llegar a generar segmentación de los desarrolladores y ello conlleva a la fragmentación del proyecto.	todo lo relacionado con el proyecto.
Deficiencia en el conocimiento de los diferentes individuos	La deficiencia del conocimiento en el desarrollador genera que a nivel individual no pueda cumplir con sus deberes y ello necesariamente repercute a nivel colectivo.	Usar herramientas y definir procesos de formación con el conocimiento necesario para el desarrollo del proyecto.
Malos entendidos a nivel grupal	Los malos entendidos a nivel grupal en el desarrollo de software libre conllevan a la deserción de los integrantes.	Propiciar el buen ambiente dentro del proyecto. Entregar herramientas de comunicación adecuadas para que los participantes tengan una buena interacción y lleven a cabo procesos de integración.
Reglas de participación	La falta de reglas de participación y las reglas bien definidas que incentiven la participación, ocasionan el	Generar un listado o manual con las normas de participación del proyecto que deben de

	abandono continuo de participantes.	ser aceptadas antes de ingresar al mismo.
Perdida de interés en el proyecto por parte de cada individuo	Este dilema se presenta por la presencia de varios de los anteriores dilemas expresados y en su gran mayoría es difícil de contrarrestar dada la pérdida de interés en el proyecto, esto en muchas de las ocasiones genera un conflicto más allá del ámbito personal, afectando el colectivo del proyecto.	Generación de nuevos retos para los participantes. Proyectos de formación grupales de transferencia de conocimiento. Aceptación de aportes de cada individuo.

Tabla 1. Dilemas sociales enfocados al desarrollo de software

3.4 Cooperación

La cooperación consiste en la relación de confianza que puedan tener básicamente 2 personas al encontrarse en una situación donde dependiendo de las decisiones tomadas por cada individuo, se pueden obtener resultados favorables o desfavorables según cada caso. Esto es precisamente lo que explica Merrill Flood y Melvin Dresher en su dilema del prisionero y en donde los resultados están dados por:

	Tú confiesas	Tú lo niegas
Él confiesa	Ambos son condenados a 6 años.	Él sale libre y tú eres condenado a 10 años.
Él lo niega	Él es condenado a 10 años y tú sales libre.	Ambos son condenados a 6 meses.

Tabla 2. Cooperación Dilema del Prisionero A

	Tú encubres	Tú traicionas
Él encubre	Máximo beneficio común	Tú ganas, él pierde
Él traiciona	Él gana, tú pierdes	Máximo perjuicio común

Tabla 3. Cooperación Dilema del Prisionero B

En el este sentido cabe resaltar que la misma teoría tiene variantes que pueden aplicar o no dependiendo de la situación y son muy utilizadas por la teoría de juegos.

3.5. Software libre como bien publico

El sector de la economía encargado de la provisión de los bienes generales es el sector público; esto es entendido en las actuales economías modernas donde en la actualidad se entiende ello como esencial para la preservación de los bienes actuales para las generaciones futuras.

Es por ello que la disponibilidad de los mismos es una prioridad en cualquiera de las economías, el aprovisionar dichos bienes es en casi todas las circunstancias un problema de tipo económico en donde en la mayoría de ocasiones de destina un porcentaje elevado del erario público a ello. Un ejemplo de los bienes comunes y sus sectores puede verse en la siguiente figura.

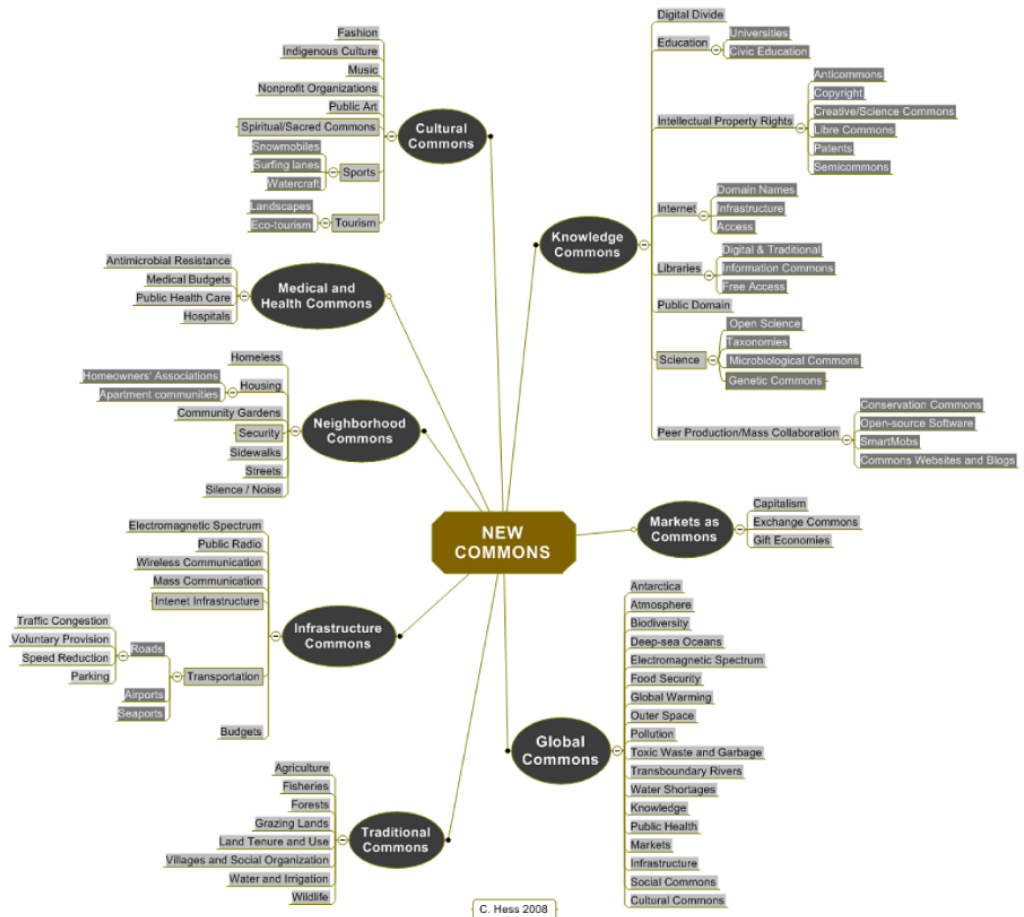


Figura 4. New Commons

Pero, ¿qué es un bien público? En palabras de Elinor Ostrom “Un bien que está disponible a todos y del cual su uso por una persona no subtraher de su uso por otros”. La explicación de la frase en términos tradicionales quiere decir que un bien público puede ser consumido por todos sin importar quien sea, por ende que nadie puede impedirlo y el uso por parte de un individuo no se ve perjudicado por el uso que le den otros.

En este orden de ideas el desarrollo de proyectos de software libre puede ajustarse bajo la misma premisa dado que:

- Una vez producido todo mundo se puede beneficiar de él.
- Una vez producido es imposible evitar que las personas accedan a él.
- El producto no se gasta con el uso y por lo tanto otros pueden volver a usarlo.

4. Método de Investigación

4.1 Formulación Declaración del Problema

¿Qué competencias laborales son necesarias para la promoción de la cooperación en el desarrollo de proyectos de software libre?

Es importante la investigación para poder identificar los elementos que enfrentan a los participantes del desarrollo de proyectos de software libre en el desarrollo de la cooperación de los mismos o lo que se puede deducir como los dilemas que los enfrentan.

Establecer de igual forma los elementos necesarios a modo de competencias laborales necesarias para afrontar dichos dilemas que deriven en el diseño y evaluación de un curso de e-learning.

4.2 Objetivo general

Diseñar y evaluar un curso de e-learning que promueva la cooperación en proyectos de software libre.

4.3 Objetivos específicos

- Identificar los comportamientos individuales que generan catástrofes en los esquemas cooperativos que sirvan como base para identificar las competencias que se deben rescatar a partir de los mismos.
- Diagnosticar 2 proyectos de software libre con fracasos que evidencien la falta de cooperación.
- Estudiar los motivos por los cuales los integrantes de un proyecto de software libre pueden llegar a cooperar activa y continuamente.
- Diseñar un curso de e-learning basado en los hallazgos realizados.

4.3 Metodología y desarrollo del trabajo

El presente proyecto tiene componentes académicos, investigativos, culturales, tecnológicos y sociales. Se llevó a cabo el análisis de las perspectivas de los desarrolladores de software y otros integrantes de proyectos de software libre con dilemas e inconvenientes para llevar a cabo el desarrollo de los mismos.

Dentro de la estrategia llevada a cabo se describe:

- Primero se involucró una fase en la cual se definieron los dilemas sociales enfocados al desarrollo desde la perspectiva de los desarrolladores de dos empresas dedicadas a esta actividad mediante el desarrollo de una encuesta.
- Como segundo paso se valoraron 2 proyectos de software libre que hayan fracasado por la falta de cooperación dentro de sus participantes ya sea por efectos propios del proyecto o agentes externos para afianzar las competencias laborales requeridas para el desarrollo de dichos proyectos.
- En tercer lugar se llevó a cabo un diagnóstico de metodología de desarrollo ágil para complementar la definición de cómo mejorar el desarrollo de un proyecto de software libre integrando las competencias descubiertas.
- Con esta información y el análisis se llevó a cabo las observaciones de los resultados obtenidos y por medio de ello se plantearon las competencias requeridas iniciales que ayuden a la colaboración en un proyecto de desarrollo de software libre para el diseño del curso de e-learning.

- Para finalizar y como quinta etapa, se construyó un documento en el cual se diseñó un curso de e-learning con la definición de competencias laborales y la aplicabilidad del modelo de desarrollo ágil.

4.3.1 Secuencia descriptiva de pasos

La secuencia llevada a cabo durante el proceso del desarrollo del trabajo investigativo fue la siguiente:

- a) Recolección de información mediante experiencias y encuestas necesarias.
- b) Se aplicó la metodología.
- c) Se revisó la información relacionada con 2 proyectos de software libre.
- d) Se analizaron los resultados del proceso de recolección de información y la relacionada con los 2 proyectos de software libre.
- e) Se construyó el documento con el diseño del curso de e-learning.

4.3.2 Diseño de instrumentos que integrarán la metodología (Entrevistas, encuestas, observación, etcétera)

Se escogió una metodología de desarrollo genérica. Se empleó software libre, a través del cual permitió realizar la investigación. En todo el proceso cabe destacar que los procesos de observación, análisis, encuestas y toma de vivencias fueron de vital importancia para la investigación.

4.3.3 Técnica usada para la recopilación de datos

Se empleó las técnicas de observación y encuestas para la recolección de los datos.

En este contexto se llevó a cabo la encuesta a dos grupos de desarrolladores que participan en proyectos de desarrollo de software libre.

Las empresas seleccionadas fueron:

Datos / Empresa	Heinsohn Business Technology	Swat IT
Ciudad	Bogotá	Armenia
Empleados	700	40
Desarrolladores	490	34
Desarrolladores en proyectos con software libre	70	28
Tecnologías empleadas para el desarrollo	Java, JBoss, GlassFish, Mysql, PostgreSql.	Java, Apache, Tomcat, JBoss, GlassFish, Mysql, PostgreSql, Php

Tabla 4. Datos Empresas para Encuesta

De lo anterior se dedujo que la población para la aplicabilidad de la encuesta es de 98 Desarrolladores.

Ahora con dicho dato se procedió a establecer el tamaño de la muestra empleando la fórmula para poblaciones finitas.

$$n = \frac{N * Z_{\alpha}^2 * p * q}{d^2 * (N - 1) + Z_{\alpha}^2 * p * q}$$

Donde:

n= tamaño de la muestra deseada.

N= tamaño conocido de la población. En este caso 98.

z= valor correspondiente al nivel de confianza. Se utilizó el 95%. 1.96

p= proporción esperada. Como no teníamos el valor definido, se utilizó el valor recomendado por defecto de estos casos 5% 0.05

q= varianza de la población. $1 - p = 1 - 0.05 = 0.95$

d= precisión (3%)

$$n = \frac{98 * 1.96^2 * 0.05 * 0.95}{0.03^2 * (98 - 1) + 1.96^2 * 0.05 * 0.95}$$

$$n = \frac{98 * 3.8416 * 0.05 * 0.95}{0.0009 * 97 + 3.84.16 * 0.05 * 0.95}$$

$$n = \frac{17.8826}{0,0873 + 0,1825}$$

$$n = \frac{17,8826}{0,2698}$$

$$n = 66.2870$$

En este caso el tamaño de la muestra definido a la que se le aplicó la encuesta fue de 66 desarrolladores.

Para ver la encuesta dirigirse al Anexo 1.

4.3.4 *Cómo se efectuó el análisis*

Los resultados de las encuestas fueron graficadas para poder hallar comportamientos.

Las observaciones, revisiones y análisis llevados a cabo, se documentaron para extraer conclusiones.

Para llevar a cabo el proceso de evaluación de los dos proyectos de software libre que fracasaron se utilizó como base de estudio el enfoque para las métricas de software GQM “Goal, Question, Metric” (1994) Victor Basili.

El enfoque de la evaluación no fue determinar su valor monetario, sino identificar objetos de evaluación que permitieran medir su grado de madurez y por ende los puntos que llevaron al fracaso o el momento en que un proyecto se puede definir como fracasado.

Cabe destacar que en el mundo del software libre, por su condición de software libre y cumplir las 4 libertades, es perfectamente reciclable en algún momento posterior y es una práctica común entre los proyectos que en algún momento tuvieron gran acogida por parte de la comunidad.

Los puntos que fueron tenidos en cuenta para la evaluación:

- Desde el punto de vista comunitario. Bugtrackers, listas de correo, foros, repositorios de código.

- Desde el punto de vista de la calidad del proyecto. Organización del proyecto, herramientas de seguimiento de errores, repositorios de código usados.
- Desde el punto de vista del usuario final. Funcionalidad, usabilidad, madurez, diseño, estabilidad.

5. Resultados

El presente capítulo presenta los resultados de las actividades y labores llevados cabo para obtener el material de análisis y evaluación del trabajo desarrollado.

5.1 Resultados Encuesta

El propósito de esta encuesta fue identificar los dilemas que aquejas a los desarrolladores de proyectos de software libre y que fueron la base para determinar las competencias laborales.

a) Genero de las personas encuestadas.

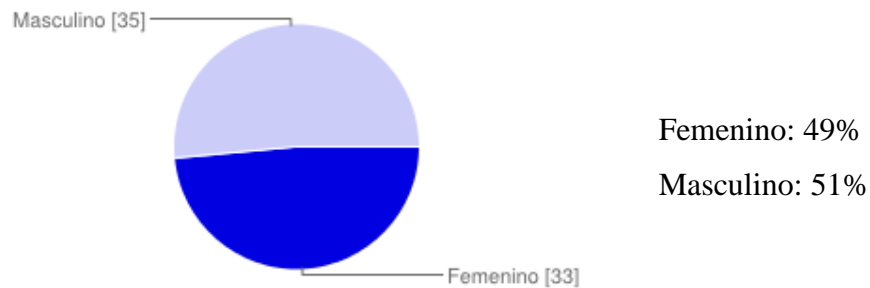


Figura 5. Encuesta. Genero

b) Rango de edades de las personas encuestadas.

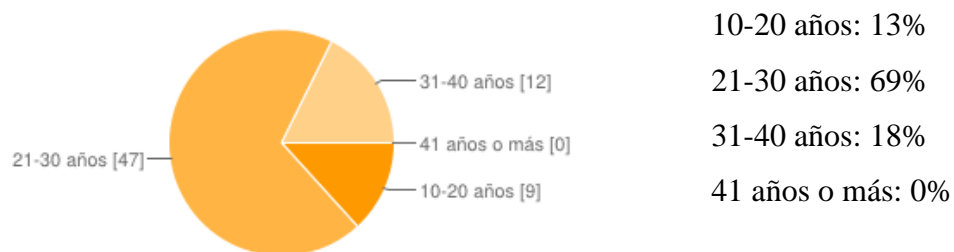


Figura 6. Encuesta. Rango de Edades

c) ¿En cuál de las etapas clásicas de un proyecto de software libre considera hay mayor riesgo de fracaso?

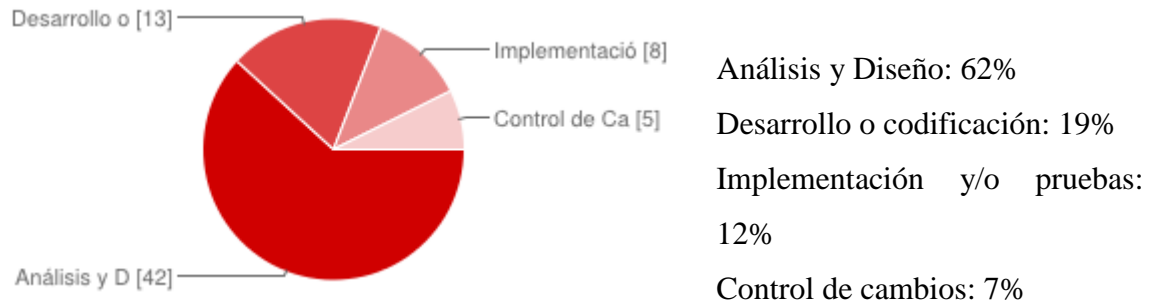


Figura 7. Encuesta. Etapas clásicas de un proyecto

d) ¿Considera que la comunicación es importante en el desarrollo de un proyecto de software libre?

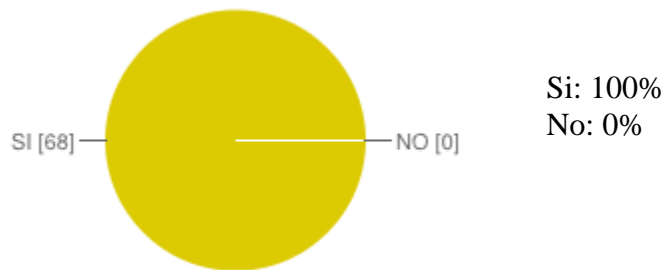


Figura 8. Encuesta. Consideración de la comunicación

e) ¿Cuáles considera son los dilemas más comunes en el desarrollo de un proyecto de software libre?



Figura 9. Encuesta. Dilemas en el desarrollo de proyectos

Falta de liderazgo: 23%

Falta de participación de los integrantes: 14%

Falta de divulgación del proyecto para atraer nuevos integrantes: 10%

Carencia de herramientas de participación: 12%

Trabas al acceder al proyecto: 10%

Falta de comunicación de los participantes: 18%

Escases de los recursos económicos: 11%

Otro: 2%

Otros: Alcance poco claro o no definido, Falta de recursos de capacitación.

f) ¿Considera que el manejo del idioma en el que esté definido un proyecto de desarrollo de software libre genera inconveniente a la hora de participar en él?

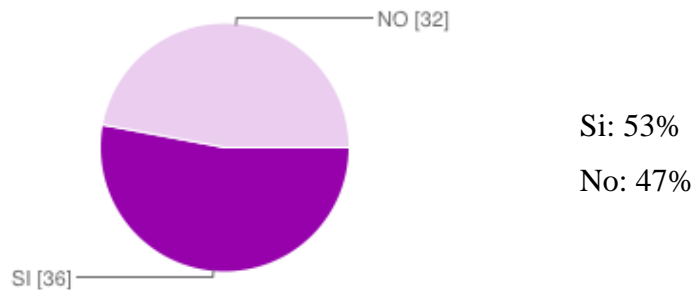


Figura 10. Encuesta. Manejo del idioma

g) ¿Cuáles considera según su experiencia son los dilemas que se deben de afrontar en las etapas de análisis y diseño de un proyecto de software libre?

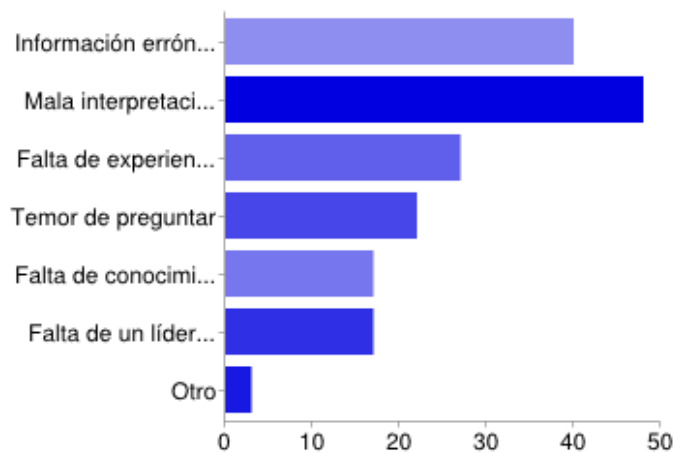


Figura 11. Encuesta. Dilemas en el análisis y diseño

Información errónea por parte del cliente: 23%

Mala interpretación de los requerimientos del cliente: 28%

Falta de experiencia por parte de las personas encargadas del levantamiento: 16%

Temor de preguntar: 13%

Falta de conocimientos: 10%

Falta de un líder en la etapa: 10%

Otro: 2%

Otro: Falta de contacto con cliente o usuarios finales, Falta de Recursos.

h) ¿Cuáles considera según su experiencia son los dilemas que se deben de afrontar en las etapas de desarrollo o codificación de un proyecto de software libre?

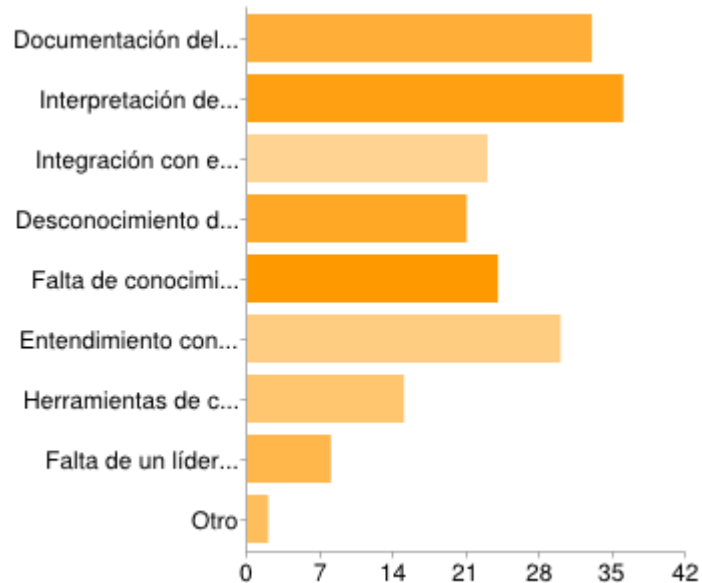


Figura 12. Encuesta. Dilemas en el desarrollo o codificación

Documentación del código: 17%

Interpretación de los requerimientos levantados previamente: 19%

Integración con el grupo de desarrollo: 12%

Desconocimiento de los entornos o herramientas de desarrollo: 11%

Falta de conocimiento en el seguimiento de buenas prácticas: 13%

Entendimiento con las personas que llevaron a cabo el análisis y diseño: 16%

Herramientas de comunicación: 8%

Falta de un líder en la etapa: 4%

Otro: 2%

Otro: Utilizar un modelo de desarrollo distribuido.

- i) ¿Cuáles considera según su experiencia son los dilemas que se deben de afrontar en las etapas de implementación y/o pruebas de un proyecto de software libre?

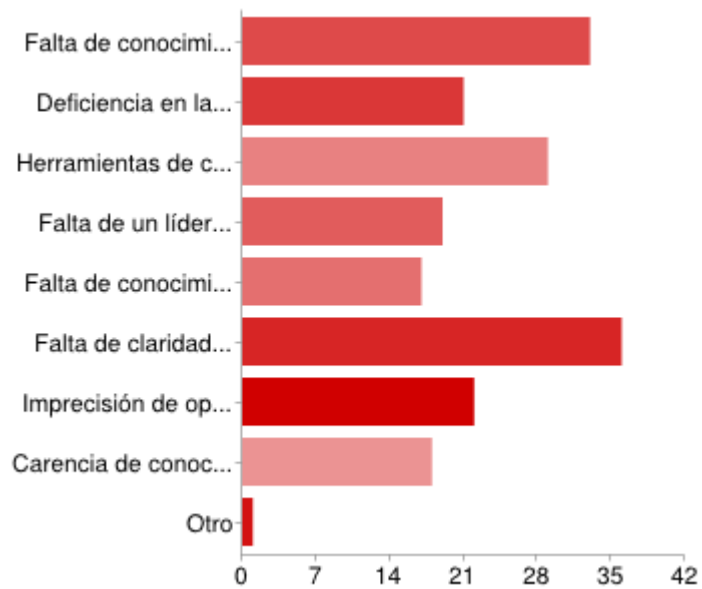


Figura 13. Encuesta. Dilemas en la implementación y/o pruebas

Falta de conocimiento de infraestructura: 17%

Deficiencia en la iteración con los compañeros de las etapas anteriores: 11%

Herramientas de comunicación: 15%

Falta de un líder en la etapa: 10%

Falta de conocimientos en el seguimiento de buenas prácticas: 9%

Falta de claridad en las pruebas a ser ejecutadas: 18%

Imprecisión de operación en las entregas recibidas por parte de los desarrolladores: 11%

Carencia de conocimiento a nivel funcional de los procesos del cliente: 9%

Otro: 1%

Otro: Falta de conocimiento en infraestructura.

j) ¿Cuáles considera según su experiencia son los dilemas que se deben de afrontar en la etapa de control de cambios de un proyecto de software libre?

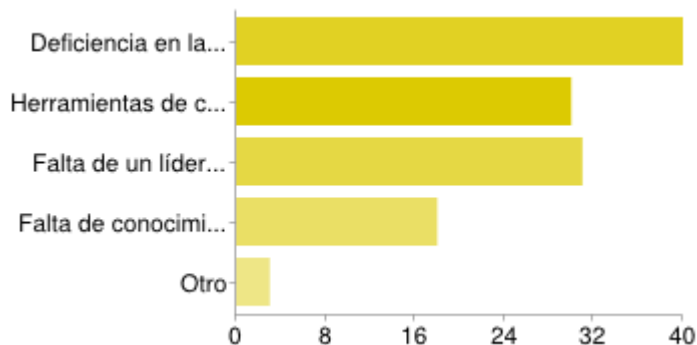


Figura 14. Encuesta. Dilemas en el control de cambios

Deficiencia en la iteración con los compañeros de las etapas anteriores: 33%

Herramientas de comunicación para los cambios: 25%

Falta de un líder en la etapa que coordine los cambios: 25%

Falta de conocimientos en el seguimiento de buenas prácticas: 15%

Otro: 2%

Otro: Código mal documentado, falta de recursos humanos, aceptación de la comunidad.

k) De las siguientes estrategias ¿Cuáles utiliza para enfrentar los dilemas sociales?

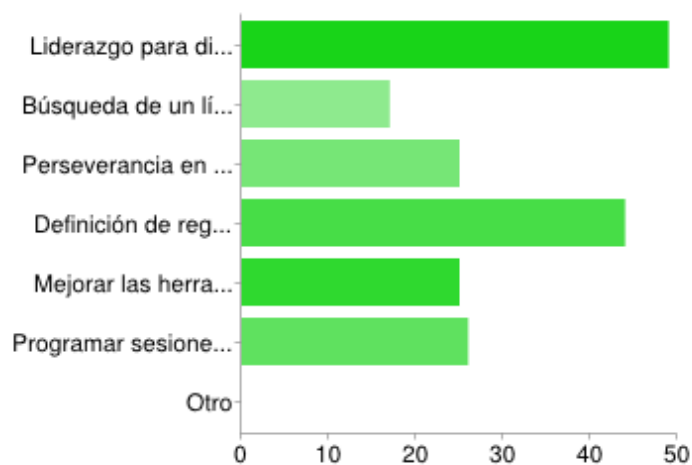


Figura 15. Encuesta. Estrategias para afrontar los dilemas

Liderazgo para dirigir o corregir el rumbo en momentos de estancamiento: 26%

Búsqueda de un líder por méritos: 9%

Perseverancia en el desarrollo del proyecto: 13%

Definición de reglas claras: 24%

Mejorar las herramientas de colaboración para difundir el conocimiento de forma grupal:
13%

Programar sesiones de capacitación para reforzar los conocimientos de las personas: 14%

Otro: 0%

5.2 Diagnóstico: Proyectos de Software Libre

Proyecto: Open Office (OOo). Antes de ser donado a la fundación Apache por Oracle (Apache OpenOffice “AOO”)

Punto de vista: Comunitario	
Bugtrackers	<p>El proyecto tenía estructurado a nivel comunitario el reporte de errores apoyado en herramientas personalizadas por la comunidad basadas en Bugzilla.</p> <p>El esquema incluía reglas de participación para la notificación de errores y herramientas para que los usuarios del proyecto pudiesen de una forma fácil reportar los mismos sin ser ingenieros o desarrolladores.</p> <p>http://www.openoffice.org/qa/issue_handling/basic_rules.html http://www.openoffice.org/qa/issue_handling/pre_submission.html</p>
Listas de correo	<p>Las listas de correo fueron estructuradas siguiendo el esquema de gobierno del proyecto y por grupos de desarrollo dentro del producto.</p> <p>De esa forma se mantenía la comunicación en los avances. Como ejemplo se tenía las listas de desarrolladores en cada uno de sus componentes como el write, la traducción de idiomas, el marketing, la generación de instaladores, etc.</p> <p>La función de estas listas de correo, mantener a todos los involucrados notificados de los avances del proyecto en su respectiva área.</p>

	<p>http://openoffice.apache.org/mailling-lists.html</p>
Foros	<p>Cuenta con diversos foros estructurados por idiomas y por actividades tales como inicio rápido, traducción, desarrollo, reporte de fallos, generación de instaladores.</p> <p>Cabe destacar que en dicho proceso se definieron de igual forma las reglas de participación.</p> <p>La participación en los foros del proyecto por parte de la comunidad era activa y ello lo demostraba la cantidad de ingresos y registros dentro del mismo.</p> <p>En el caso de la comunidad de foros en español llego a tener 7180 usuarios con más de 30.000 mensajes y 7000 temas.</p> <p>http://forum.openoffice.org/es/forum/index.php</p>
Repositorios de código	<p>Este proyecto fue soportado por el sistema de control de versiones CVS en sus principios y luego fue migrado a SVN, la cantidad de desarrolladores inscritos en el proyecto se calcula fue de 600 en promedio.</p> <p>Adicional gracias al buen esquema de gobierno del proyecto y a la financiación por parte de terceros, el avance del mismo era de forma continua y planeada, lo cual generaba en la comunidad un aliciente por seguir desarrollando el proyecto y a los usuarios.</p>

	Sumado a esto la estrategia de emplear sistemas de control de versiones tradicionales y masivos como el subversión que es ampliamente conocido por los desarrolladores.
Punto de vista: Calidad del proyecto	
Organización del proyecto	<p>El proyecto se encontraba bien estructurado de la siguiente forma:</p> <ul style="list-style-type: none"> • Consejo de la comunidad: Integrado por miembros de la comunidad, era quien regia los designios del proyecto de forma democrática. • Un nivel por debajo se presentaban los diferentes grupos de desarrollo, mantenimiento, traducción y empaquetamiento. <p>Toda esta organización era apoyada principalmente a nivel económico por Sun Microsystem y de igual forma recibía donaciones de otras empresas que se beneficiaban de los adelantos como IBM.</p>
Herramientas de seguimiento de errores	Para el seguimiento de errores el proyecto empleaba una versión personalizada de BugZilla, el cual es una herramienta para el reporte de defectos ampliamente utilizada en proyectos de software libre desde 1998.
Repositorios de código usado	<p>Los repositorios de código empleados en el proyecto son soportados por el proyecto Subversión (SVN).</p> <p>Este sistema de control de versiones reemplazo en la línea tecnológica al CVS (Concurrent Versions System), es uno de los</p>

	más usados y difundidos a nivel de comunidad de software libre desde el año 2000.
Signos de actividad	<p>Durante el desarrollo del proyecto (1999 - 2009) se llevaron a cabo un lanzamiento aproximado de 13 versiones mayores y un número mayor de actualizaciones.</p> <p>El promedio de una versión mayor por año y sus respectivas actualizaciones menores, generaron un gran interés en la comunidad y que de forma adicional llamaba a otros desarrolladores a participar del proyecto.</p>
Punto de vista: Usuario Final	
Funcionalidad	<p>A nivel de funcionalidad, OpenOffice se comparaba favorablemente con Microsoft Office 2003, 2007 y 2010. Existen múltiples sitios que llevan a cabo evaluaciones donde hicieron dichos estudios no formales basados en documentación técnica y experiencias.</p> <p>http://blog.open-office.es/index.php/inicio/2011/03/04/microsoft-office-2010-vs-openoffice-org-3-3</p> <p>http://wiki.openoffice.org/wiki/Documentation/OOoAuthors_User_Manual/Migration_Guide/General_Differences</p> <p>http://download.cnet.com/OpenOffice-org/3000-18483_4-10263109.html</p> <p>http://www.pcworld.com/article/2010005/5-free-open-source-alternatives-to-microsoft-office.html</p>
Usabilidad	Fue tal vez el mayor punto en contra del proyecto, pues el aspecto visual del mismo frente a Microsoft Office era muy plano para un

	<p>usuario acostumbrado al cambio radical que le dio la empresa Microsoft a la suite de ofimática a partir de la versión 2007.</p> <p>Por otro lado para una persona que estuviese acostumbrada al uso de la suite de Microsoft, se le complicaba el proceso de cambio hacia el proyecto Open Office.</p> <p>Lo anterior no sucedió con las personas acostumbradas al mundo del software libre, quienes por el contrario vieron un gran cambio en la versión 3 del proyecto con su cambio visual.</p>
Estabilidad	<p>Al proyecto siempre se le critico la estabilidad de uso por el gran consumo de memoria a raíz del uso desmedido que el motor de JAVA hacía del mismo especialmente en sistemas operativos Windows. Esto género que en la versión 3 se llevara a cabo un proceso de optimización del proyecto y con ello la evolución o integración de soluciones que llevaran en gran medida a frenar el consumo desmedido y por ende a estabilizar la aplicación en cuanto a consumo de recursos.</p> <p>De esto también se encontraba información por parte de los usuarios en los foros del proyecto.</p> <p>http://forum.openoffice.org/es/forum/viewtopic.php?f=39&t=3441</p>

Tabla 5. Diagnóstico proyecto OpenOffice

Proyecto: GNU Telephony.

Punto de vista: Comunitario	
Bugtrackers	<p>El proyecto para el reporte y administración de defectos empleaba Mantis BT, en donde los desarrolladores y participantes podían dejar el registro de los mismos y donde adicional se podía ver el avance.</p> <p>A diferencia del proyecto de OpenOffice, un usuario que no perteneciera al proyecto para registrar un defecto primero debía darse de alta en el sistema y ser aceptado antes de poder reportar defectos.</p>
Listas de correo	<p>A diferencia de otros proyectos, las listas de correo son pocas y se encuentran definidas en 5 grupos:</p> <ul style="list-style-type: none">▪ gnucomm-privacy@gnu.org▪ sipwitch-devel@gnu.org▪ ccrtp-devel@gnu.org▪ bug-commoncpp@gnu.org▪ bayonne-devel@gnu.org <p>Esto podría demostrar dos posible situaciones:</p> <ul style="list-style-type: none">- Una donde el proyecto no requería tener más área de desarrollo.- La otra donde se evidenciaba lo poco a tractivo para los desarrolladores y por ende la poca comunidad inscrita para el desarrollo del mismo.
Foros	<p>El proyecto tenía poca información y solo cuenta con un sitio principal bajo un diseño de wikki.</p>

		Adicional se encontraron sitios enlazados que en ningún caso fue foro.
Repositorios de código	de	El proyecto utilizo el sistema de control de versiones GIT y como el proyecto no tenía infraestructura propia definida ni el patrocinio contante de alguna empresa, opto por usar el servicio público GitHub. https://github.com/gnutelephony
Punto de vista: Calidad del proyecto		
Organización del proyecto	del	El proyecto se encontraba organizado en dos equipos. El primero encargado de la administración del proyecto y el segundo encargado del desarrollo de los diversos componentes del mismo.
Herramientas de seguimiento de errores	de	Para el seguimiento de errores el proyecto empleaba la herramienta Mantis BT.
Repositorios de código usado	de	El proyecto utilizo el sistema de control de versiones GIT y como el proyecto no tenía infraestructura propia definida ni el patrocinio contante de alguna empresa, opto por usar el servicio público GitHub. https://github.com/gnutelephony
Signos de actividad		Por haber sido un proyecto de investigación basado en un protocolo estándar y no un producto en sí. El proyecto no tuvo muchas liberaciones de un producto final.

	El aporte del proyecto fue evidenciar la potencia de uso del protocolo SIP para la telefonía y a partir de ello dieron inicio otras series de proyectos.
Punto de vista: Usuario Final	
Funcionalidad	A nivel de funcionalidad para el usuario final fue casi nula dado que no existió un producto directamente que se pudiese usar.
Usabilidad	La usabilidad de lo desarrollado era nula para el usuario final dado que no existió un producto final que se pudiese usar.
Estabilidad	A pesar de que no existió un producto final, los logros del proyecto empleados en otros proyectos fueron o son estables.

Tabla 6. Diagnóstico proyecto GNU Telephony

5.3 Discusión de Resultados

5.3.1 Encuesta

Como conclusiones en el desarrollo de la encuesta se tuvieron:

- La población encuestada fue muy similar en cuanto a género se refiere.
- La mayoría de desarrolladores encuestados se encontraban en el rango de edad de 21 a 30 años.
- El 62% de los desarrolladores consideran que hay mayor riesgo de fracaso de un proyecto de software libre en la fase del análisis y diseño.
- El 100% de los desarrolladores considero que la comunicación es importante para el desarrollo de un proyecto de software libre.
- Se consideraron que los dilemas en el desarrollo de un proyecto de software libre que más lo afecta son la falta de liderazgo y la falta de comunicación de los participantes.
- Solo un 53% de los desarrolladores encuestados consideraron que el idioma de un proyecto genera inconvenientes para participar en él.
- Referente a las etapas clásicas del desarrollo, esto fue lo que se encontró en cuanto a dilemas según los desarrolladores:
 - Análisis y diseño: interpretación errónea de los requerimientos e información errónea por parte del cliente.
 - Desarrollo o codificación: interpretación de los requerimientos previos, documentación del código e integración del grupo de desarrollo.
 - Implementación y/o pruebas: claridad en las pruebas a ser ejecutadas, falta de conocimientos en infraestructura y las herramientas de comunicación.
 - Control de cambios: deficiencia en la iteración con los compañeros en las etapas anteriores, herramientas de comunicación y falta de un líder.

- En cuanto a las estrategias para enfrentar dichos dilemas enfocados al desarrollo de software libre, los desarrolladores respondieron que consideran clave un buen liderazgo donde además se definan reglas claras de participación en el proyecto.

5.3.2 *Diagnóstico de Proyectos de Software Libre*

En los casos del diagnóstico de los proyectos de software libre se presentaron 2 proyectos.

El primero ampliamente reconocido, con tradición y que ha despertado interés de la comunidad de desarrolladores en general, con una estructura organizativa, comunitaria e infraestructura definida y a lo largo de su desarrollo con patrocinios de grandes empresas del sector tecnológico apoyando económicamente su proceso de evolución respetando su independencia.

Por otro lado un proyecto con espíritu de innovación en el tema de comunicaciones, sin definición de los elementos básicos estructurales para su funcionamiento como proyecto y con una pequeña comunidad de desarrolladores tratando de sacarlo adelante.

En ambos casos a pesar de sus diferencias estructurales, económicas y de objetivos de desarrollo, se tiene como resultado que a causa de factores como la pérdida de independencia del proyecto por la compra de Oracle de OpenOffice y en el segundo caso a raíz del pobre avance del mismo en temas de resultados a nivel de desarrollo en GNU Telephony. Las comunidades de desarrolladores por grandes o pequeñas que fueron, dejaron de cooperar activamente y buscaron nuevas alternativas de proyectos como la Fundación LibreOffice y el proyecto Open Source Asterisk.

¿Por qué se definió que el proyecto OpenOffice (oOO) fracasó?

Se pudo definir que el proyecto OpenOffice fracasó a raíz de las siguientes situaciones:

El actual proyecto se llama Apache OpenOffice y es considerado como descendiente del proyecto que originalmente se llamó OpenOffice.

El código del proyecto se mantuvo bajo la licencia pública general limitada de GNU (LGPL) hasta que SUN Microsystems fue adquirida por Oracle y está mantuvo congelado el proyecto sin un norte definido durante un año, lo que generó que muchos de los participantes de la comunidad se retiraran y consolidaran la Fundación LibreOffice.

A raíz del cambio del licenciamiento, el proceso de transición y el tiempo de congelamiento; muchas comunidades de desarrolladores no lo apoyan porque los desarrollos que se hagan bajo la nueva licencia de la fundación Apache pueden terminar en proyectos comerciales sin retribución al proyecto.

En consecuencia el proyecto OpenOffice (OOo) era diferente al actual proyecto Apache OpenOffice (AOo).

5.4 Conclusión de los Resultados y Competencias Laborales Definidas

5.4.1 Competencias Laborales

Con la información recolectada, el análisis ejecutado y la discusión desarrollada se lograron extraer las siguientes competencias que se deberían desarrollar en un curso de e-learning que tenga como objeto la promoción de la cooperación en el desarrollo de software libre.

Competencia Definida	Descripción y/o Objetivo
Inspiración de compromiso	Desarrollar una visión y valores sólidos que permitan obtener confianza que de forma adicional pueda ser transmitida generando lazos de cooperación.
Generación de cambios	Ser capaz de liderar cambios a través de habilidades de comunicación e interrelación.
Generación de alianzas	Capacidad de análisis e interpretación del entorno para poder llevar a cabo alianzas dentro y fuera del entorno laboral.
Innovación de soluciones	Buscar la forma de mejorar el cómo hacer las cosas a medida que transcurre el desarrollo de sus labores en el proceso natural de mejora del aprendizaje, experiencia y capacidades.
Capacidad de participación	Fomentar un ambiente de cooperación y trabajo en equipo para lograr el desarrollo de los objetivos laborales.

Orientación a resultados	Capacidad de mantenerse motivado a fin de poder llevar a cabo la continuación de procesos buscando continuos resultados.
Capacidad de negociación	Capacidad de gestión de intereses comunes sobre los particulares en pro del desarrollo de los objetivos laborales en momentos de incertidumbre o dificultades.

Tabla 7. Competencias laborales definidas

5.4.2 Cooperación Activa

La cooperación brinda oportunidades de beneficio que no son posibles conseguir de forma individual por cada individuo, los beneficios se obtienen a la iteración sostenida con el tiempo, es decir, los beneficios no son inmediatos sino por el contrario diferidos a mediano y largo plazo. Hay que tener presente que dichas oportunidades de mutuo beneficio se pueden frustrar por el comportamiento racional-egoísta de cada individuo.

Los dilemas sociales ocurren en situaciones de interdependencia en las que los individuos enfrentan incentivos de corto plazo que los llevan a escoger acciones individualistas, cuando podrían generar acuerdos para cooperar de forma activa y obtener los mayores pagos tanto para los individuos como para el grupo (Kollog, 1998). Todas las estrategias de intervención de dilemas sociales suponen un mecanismo para ganar confianza (Ostrom & Walter, 2005).

De lo anterior, la ejecución de la encuesta y del diagnóstico de los dos proyectos tuvimos las siguientes ideas de cuando un integrante de un proyecto de software libre coopera de forma activa.

- Cuando un proyecto se encuentra definido y proyectado en el tiempo con objetivos ambiciosos.
- Los objetivos del proyecto presentan un reto al integrante.
- El proyecto es independiente de sus fuentes de financiamiento.
- Las herramientas de comunicación y colaboración permiten o generan un ambiente de cooperación adecuado.
- El proyecto es publicitado.
- El alcance de los objetivos propuestos es reconocido a sus integrantes.
- El proyecto genera valor de crecimiento profesional y personal a sus integrantes.

5.5 Diseño del Curso de e-learning

Para el diseño del curso se utilizó como base guía, el proceso de diseño general en esta área empleado por la Universidad del Quindío a fin de dar sustento educativo y formalidad a pesar de que el mismo esté enfocado a la comunidad del desarrollo del software libre.

5.5.1 Información General. Identificación del Espacio Académico

Nombre del espacio académico	Competencias Laborales Para La Promoción De La Cooperación En El Desarrollo De Proyectos De Software
Categoría de la actividad académica ¹	A.A.E.C (Actividad Académica Electiva Complementaria)
Naturaleza o tipo del espacio académico	Teórico – practico
Aspectos administrativos	Definido por cada institución, un ejemplo es definir si es habilitable, validable u homologable.
Porcentaje de virtualización	100%
Número de encuentros presenciales	Ninguno
Área, dimensión o núcleo al cual pertenece el espacio académico	Definido según la estructura de cada ente educativo.
Ubicación en la malla curricular	Todos los semestres o los indicados según lo definido por cada ente educativo.
Créditos académicos	Los créditos académicos definidos por el ente educativo.
Número de horas de encuentros presenciales	4 horas semanales (Virtuales)
Número de horas de asesoría	1 hora semanal (Virtual)
Número de horas de trabajo independiente por semana	4

Tabla 8. Diseño del curso. Información académica

¹ Cada ente Educativo define las categorías de Actividades Académicas; ejemplo de ello pueden ser Básicas, Profesionales, obligatorias Institucionales y lectivas Complementarias y profesionales.

5.5.2 Justificación

La necesidad de valorar el conjunto de conocimientos, habilidades y destrezas desarrolladas por una persona es en esencia el desarrollo personal y educativo que se da en el transcurso del inicio en la etapa laboral. Sin embargo es necesario desarrollar la capacidad para emplearlas en determinadas situaciones en donde por lo general hay que resolver problemas y desenvolverse en un entorno complejo de múltiples tareas.

En ese sentido las competencias laborales son el conjunto de conocimientos, habilidades y actitudes que aplicadas en situaciones del ámbito productivo, se traducen en resultados efectivos que al sumar un proceso de cooperación, contribuyen al logro de objetivos de la organización o negocio.

Teniendo en cuenta este concepto el Gobierno Nacional a través del Ministerio de Educación ha definido una política de “Articulación de la Educación con el Mundo Productivo a través de la Formación de Competencias Laborales”²

El propósito del curso, para el desarrollador de software libre es mostrarle las competencias laborales esenciales que puede utilizar en el desarrollo de sus actividades dentro de un proyecto para mejorar el proceso de cooperación.

5.5.3 Propósito u Objetivo General

Ofrecer al desarrollador de proyectos de software libre un panorama general de las competencias laborales que promuevan la cooperación útiles en la actividad

² <http://www.mineducacion.gov.co/1621/article-85777.html>

profesional del desarrollo de dichos proyectos, a través de una experiencia de formación 100% virtual.

5.5.4 Objetivos Específicos

- Contextualizar al desarrollador sobre la importancia de las competencias laborales y su terminología.
- Aplicar las competencias laborales enfocadas a la promoción de la cooperación para la mejora del desarrollo de proyectos de software libre.
- Propiciar un espacio de aprendizaje, que servirá al desarrollador para crear comunidades cooperativas en el entorno del desarrollo de proyectos de software libre.

5.5.5 Contenidos del Curso (Saber)

Ambientación

- Contextualización con el espacio de formación, la dinámica propia de la formación virtual y el rol de estudiante virtual.
- Manejo de recursos propios del aula virtual.

El Concepto de Competencias

- ¿Qué son las competencias? Historia y Conceptos.
- Competencias básicas, ciudadanas y laborales.
- Terminología

Entendiendo las Competencias Laborales

- Tipos de Competencias Laborales.
- Competencias laborales generales y específicas.
- Competencias laborales que fomentan la cooperación. (Definidas en el presente trabajo).

5.5.6 Contenidos Procedimentales (Saber Hacer)

Que el desarrollador aplique en el contexto de su desarrollo laboral o en el contexto de participación de un proyecto de software libre las competencias laborales que fomentan la cooperación atacando una dificultad presente.

5.5.7 Contenidos Actitudinales (Ser)

Estimular al desarrollador a que se interese por la adopción y valoración de las competencias laborales que fomentan la cooperación en el ámbito laboral.

5.5.8 Estrategias Metodológicas

El desarrollador tendrá acceso a la plataforma, a las actividades, recursos y herramientas colaborativas las 24 horas del día. Las actividades programadas para cada uno de los temas deberá desarrollarlas en una semana, en el día y hora que el considere conveniente y se ajuste a sus necesidades, cumpliendo con las fechas estipuladas para cumplir con los objetivos definidos.

Metodología	En que Consiste	Para qué se usa
Lectura previa de la guía general del curso y las guías de la unidad	Lecturas sobre las temáticas, descripción de las actividades y tiempos propuestos.	Dar a conocer las temáticas a desarrollar para propiciar una mayor interacción entre los estudiantes y el tutor.
Talleres individuales o en equipo (Wikis)	Desarrollo de actividades que lleven a una indagación individual o colectiva.	Reforzar temas vistos en cada unidad y permitir que los estudiantes descubran debilidades de los mismos.
Lecturas de apoyo	Consultas de los desarrolladores en su horario independiente.	Mantener al desarrollador informado sobre los avances o material adicional al que se encuentra en la plataforma.

Foros	Respuestas de interrogantes y generación de discusiones en torno a un tema en particular.	Propiciar espacios de discusión para que los desarrolladores construyan su propio conocimiento.
Sesiones de Interacción (Videos, chats, correo electrónico)	Uso de material multimedia.	Fortalecer el proceso de enseñanza/aprendizaje.

Tabla 9. Diseño del curso. Estrategias metodológicas

5.5.9 Sistema de Evaluación

La evaluación del espacio académico será el resultado del promedio de 5 notas, cada una con un valor de 20%. En las 4 primeras notas, se evaluarán los diferentes temas vistos mediante el producto de las actividades individuales y grupales.

La nota 5, será la evaluación de la aplicación dada por parte del desarrollador de lo visto en el contexto donde se desenvuelve laboralmente o en el proyecto de software libre donde participa.

6. Discusión

Este capítulo sintetiza los aportes alcanzados por el desarrollo del trabajo. Inicialmente se revisará el cumplimiento de los objetivos propuestos. Luego se presentará la discusión sobre la formulación del problema que orientó el trabajo.

6.1 Cumplimiento de Objetivos

Objetivo	Cumplimiento
Identificar los comportamientos individuales que generan catástrofes en los esquemas cooperativos que sirvan como base para identificar las competencias que se deben de rescatar a partir de los mismos.	Cumplido: con base en el ejercicio de la ejecución de la encuesta y el posterior análisis se determinaron las competencias necesarias para afrontar los comportamientos individuales que generan catástrofes cooperativos. (Véase Capítulo 5.1 y capítulo 5.3.1)
Diagnosticar 2 proyectos de software libre con fracasos que evidencien la falta de cooperación.	Cumplido: se llevó a cabo el levantamiento de información de los 2 proyectos de software libre que han fracasado por falta de cooperación, ya sea por agentes externos al proyecto o por elementos internos. (Véase Capítulo 5.2 y capítulo 5.3.2).
Estudiar los motivos por los cuales los integrantes de un proyecto de software libre pueden llegar a cooperar activa y continuamente.	Cumplido: A raíz del análisis de los resultados de la encuesta, las observaciones de fracaso de los 2 proyectos de software libre y la revisión de evolución de los 2 proyectos en internet y blogs especializados, se plantearon los motivos por los cuales los integrantes de un proyecto pueden llegar a cooperar activa y

	continuamente. (Véase Capítulos 5.3.1, 5.3.2 y 5.4.2).
Diseñar un curso de e-learning basado en los hallazgos realizados.	Cumplido: Se propuso el diseño del curso de e-learning siguiendo una estructura curricular del ámbito de un ente de educación en Colombia, con los hallazgos realizados en el capítulo 5.4.1. (Véase Capítulo 5.5).

Tabla 10. Cumplimiento de objetivos

6.2 Discusión sobre la formulación del problema

La pregunta formulación del problema que orientó el desarrollo del trabajo fue la siguiente:

¿Qué competencias laborales son necesarias para la promoción de la cooperación en el desarrollo de proyectos de software libre?

En este trabajo se ha presentado un proceso de análisis y evaluación desde el punto real de los desarrolladores activos de software libre de dos empresas Colombianas donde cada uno desde el punto de vista de su experiencia laboral han aportado a la definición de las competencias laborales necesarias para llevar a cabo el proceso de cooperación.

6.3 Aportes

No se ha encontrado en la literatura una definición puntual de las competencias laborales que debería tener un integrante de un proyecto de software libre para promover la cooperación dentro del mismo. Si bien existe teoría acerca de la definición de competencias y la diversificación que existe de las mismas y la importancia que hoy en día se les da por temas de evolución de los mercados laborales, la misma teoría solo aborda temas generales y no se

enfoca en definir las competencias puntuales en las diversas áreas del aparato productivo de una economía.

En esencia el aporte del presente trabajo fue el de definir dichas competencias.

6.4 Limitaciones

Las limitaciones del este trabajo son:

- El alcance dado en la población de la encuesta. El mundo o la población del software libre es universal y la encuesta fue enfocada al pensamiento y experiencia de los desarrolladores colombianos, lo cual no representa necesariamente las opiniones de los desarrolladores de otras regiones u otros continentes donde se tienen otras tradiciones y métodos de trabajo y donde los resultados pueden variar.
- Las características heterogéneas de los integrantes objeto de estudio. Estas características están dadas en función a sus labores y ámbitos de desempeño; una condición de homogeneidad sería más propicia para garantizar un porcentaje más elevado en los resultados.

7. Trabajo Futuro

El desarrollo del trabajo deja abiertos varios caminos para el desarrollo de otros trabajos o la continuación del mismo mediante el desarrollo del curso de e-learning.

- Desarrollo de los contenidos propuestos del curso e-learning.
- Continuar con la investigación del impacto que tienen las competencias laborales seleccionadas para promover la cooperación en un ambiente productivo.
- Buscar un proyecto donde se encuentre en riesgo el proceso de cooperación y aplicar el curso de e-learning para observar los resultados.

8. Conclusiones

Con el desarrollo de este proyecto se pueden definir las siguientes conclusiones:

- Las competencias laborales que se pueden definir para el desarrollador de proyectos de software libre se dan gracias a la experiencia evolutiva de la participación en dichos proyectos.
- El proceso de evolución de experiencia del desarrollador se ve afectado por la presencia de dificultades presentes en el proceso de desarrollo.
- Las competencias laborales del desarrollador se pueden definir en base a sus conocimientos, habilidades, actitudes y valores.
- Los procesos de cooperación activa se ven afectados directa o indirectamente por agentes internos o externos donde se vean involucrados la organización del proyecto, la independencia o el planteamiento de objetivos ambiciosos.

Por otro lado y según lo definido al principio del proyecto se tuvieron las siguientes preguntas:

¿Qué tan impactados son los proyectos de software libre por los dilemas sociales?

Los proyectos de software libre se ven afectados por los dilemas sociales en la medida que estos afecten el proceso de cooperación dentro del proyecto.

¿Qué motiva a las personas a cooperar en proyectos de software libre?

De acuerdo a lo planteado en el desarrollo del trabajo las motivaciones son:

- Cuando un proyecto se encuentra definido y proyectado en el tiempo con objetivos ambiciosos.
- Los objetivos del proyecto presentan un reto al integrante.
- El proyecto es independiente de sus fuentes de financiamiento.
- Las herramientas de comunicación y colaboración permiten o generan un ambiente de cooperación adecuado.
- El proyecto es publicitado.
- El alcance de los objetivos propuestos es reconocido a sus integrantes.
- El proyecto genera valor de crecimiento profesional y personal a sus integrantes.

¿Se puede tomar los proyectos fracasados de software libre para determinar en qué punto fallaron y poder a partir de ello generar las competencias necesarias y que estas puedan ser entregadas a la comunidad empleando tecnología de e-learning?

La evaluación, análisis o diagnóstico que se llevó a cabo de los proyectos de software libre por si solos no permitieron deducir las competencias laborales necesarias, solo un diagnóstico subjetivo a de los elementos de que genero la falla del proyecto a partir de su estructura y de los sucesos en el desarrollo de la historia.

¿Se pueden tener proyectos exitosos en cuanto a la cooperación?

Es posible tener proyectos exitosos si el proceso de cooperación es constante y se logra obtener una interacción básica entre los integrantes teniendo en cuenta que la forma más extrema de la cooperación es la integración y es ese el objetivo a cumplir.

9. Bibliografía

SALCEDO MALDONADO, Jorge 2009. Dilemas de Acción Colectiva: Instituciones y Cooperación en Organizaciones Tecnológicas Redes para Innovación.

MCCONNE, David 2006. E-Learning Groups and Communities. David Mcconne.

AXELROD, R. & HAMILTON, W. 1981. The evolution of cooperation

BUCK, S. (1998). The global commons: an introduction. Island Press

KOLLOCK, Peter 1998. Social Dilemmas: The Anatomy of Cooperation

HARDIN, Garrett 1968. The Tragedy of Commons. "Science"

FUNDACION EOI, 2009. La oportunidad del software libre: capacidades, derechos e innovación

10. Anexos

10.1. Encuesta Dilemas Proyectos de Software Libre

Universidad Autónoma de Bucaramanga
Master en software libre
Facultad de Ingeniería
Encuesta: Dilemas proyectos de software libre
Jairo Alonso García Arias
2013



El propósito de esta encuesta se encuentra enfocado a identificar los dilemas que aquejan a los desarrolladores de proyectos de software libre.

Dilema: se define como situaciones, problemas o inconvenientes que se deben afrontar en el desarrollo de un proyecto de software libre.

Nombre(s) y Apellidos

Genero

- Femenino
- Masculino

Edad

- 10-20 años
- 21-30 años
- 31-40 años
- 41 años o más

¿En cuál de las etapas clásicas de un proyecto de software libre considera hay mayor riesgo de fracaso?

- Análisis y Diseño
- Desarrollo o codificación
- Implementación y/o Pruebas
- Control de Cambios

¿Considera que la comunicación es importante en el desarrollo de un proyecto de software libre?

- SI
- NO

¿Cuáles considera son los dilemas más comunes en el desarrollo de un proyecto de software libre?

Puede seleccionar más de una.

- Falta de liderazgo
- Falta de participación de los integrantes
- Falta de divulgación del proyecto para atraer nuevos integrantes
- Carencia de herramientas de participación
- Trabas al acceder al proyecto
- Falta de comunicación de los participantes
- Escases de los recursos económicos
- Otra... ¿Cuál?

¿Considera que el manejo del idioma en el que esté definido un proyecto de desarrollo de software libre genera inconveniente a la hora de participar en él?

- SI
- NO

¿Cuáles considera según su experiencia son los dilemas que se deben de afrontar en las etapas de análisis y diseño de un proyecto de software libre?

Puede seleccionar más de una.

- Información errónea por parte del cliente
- Mala interpretación de los requerimientos del cliente
- Falta de experiencia por parte de las personas encargadas del levantamiento
- Temor de preguntar
- Falta de conocimientos
- Falta de un líder en la etapa
- Otra... ¿Cuál?

¿Cuáles considera según su experiencia son los dilemas que se deben de afrontar en las etapas de desarrollo o codificación de un proyecto de software libre?

Puede seleccionar más de una.

- Documentación del código
- Interpretación de los requerimientos levantados previamente
- Integración con el grupo de desarrollo
- Desconocimiento de los entornos o herramientas de desarrollo
- Falta de conocimientos en el seguimiento de buenas practicas
- Entendimiento con las personas que llevaron a cabo el análisis y diseño
- Herramientas de comunicación
- Falta de un líder en la etapa
- Otra... ¿Cuál?

¿Cuáles considera según su experiencia son los dilemas que se deben de afrontar en las etapas de implementación y/o pruebas de un proyecto de software libre?

Puede seleccionar más de una.

- Falta de conocimiento de infraestructura
- Deficiencia en la iteración con los compañeros de las etapas anteriores
- Herramientas de comunicación
- Falta de un líder en la etapa
- Falta de conocimientos en el seguimiento de buenas practicas
- Falta de claridad en las pruebas a ser ejecutadas
- Imprecisión de operación en las entregas recibidas por parte de los desarrolladores
- Carencia de conocimiento a nivel funcional de los procesos del cliente
- Otra... ¿Cuál?

¿Cuáles considera según su experiencia son los dilemas que se deben de afrontar en la etapa de control de cambios de un proyecto de software libre?

Puede seleccionar más de una.

- Deficiencia en la iteración con los compañeros de las etapas anteriores
- Herramientas de comunicación para los cambios
- Falta de un líder en la etapa que coordine los cambios
- Falta de conocimientos en el seguimiento de buenas practicas
- Otra... ¿Cuál?

De las siguientes estrategias ¿Cuáles utiliza para enfrentar los dilemas sociales?

Puede seleccionar más de una.

- Liderazgo para dirigir o corregir el rumbo en momentos de estancamiento

- Búsqueda de un líder por méritos
- Perseverancia en el desarrollo del proyecto
- Definición de reglas claras
- Mejorar las herramientas de colaboración para difundir el conocimiento de forma grupal
- Programar sesiones de capacitación para reforzar los conocimientos de las personas
- Otra... ¿Cuál?