

Software Libre y Responsabilidad Social: Encuentro Inevitable en el Marco de la Educación Superior

Por: Luis Fernando Vargas Cano
lvargas@funlam.edu.co
Fundación Universitaria Luis Amigó
Medellín- Colombia

Resumen:

Las instituciones de educación superior dedicadas a la formación de profesionales cuentan desde hace varios años con el software libre como aliado fundamental para mejorar el objeto de formación propio de las personas dedicadas al uso, aplicación y difusión de las tecnologías de la información y las comunicaciones; sin embargo, y aunque en esencia la filosofía del software libre se encuentra en consonancia con la responsabilidad social, las instituciones educativas no lo han tomado en serio. Algunas de ellas, bajo pretexto de falta de soporte técnico en el medio, continúan involucradas en el modelo de software propietario, lo cual excluye, desde cualquier punto de vista, a los estudiantes que requieren de apertura en el tema tecnológico para lograr la transferencia de conocimiento idóneo, que le permita ser competente e innovador en el contexto de la globalización.

La elección por parte de la institución de educación superior de uno u otro modelo como didáctica para que el estudiante confronte los conocimientos teóricos disciplinares, resulta poco efectiva en momentos donde se requiere de total apertura para que el estudiante escoja la mejor herramienta tecnológica. La opción para que los estudiantes adquieran las competencias tecnológicas necesarias, debe provenir de las estructuras curriculares, y de las opciones que le brinde la dirección del programa al cual está inscrito. Se trata, más que elegir cualquiera de los dos modelos como única propuesta, interactuar con ambos, de tal manera que el estudiante logre introyectar la esencia de la tecnología en su exacta dimensión.

El software libre articula la formación del educando con los aspectos económicos, educativos, culturales, y ambientales, propios de la responsabilidad social, debido al modelo filosófico que ejemplifica lo que deberían ser los procesos colaborativos en torno a una ciencia o una disciplina; sin embargo, el encuentro que se produce en el marco de la educación superior, más que selectivo, debe constituir el punto de reflexión acerca de la posición del cómo se asume el aprendizaje de la tecnología

Entendida la responsabilidad social como el conjunto de prácticas vinculadas al sistema de gestión, ya sea empresarial, gremial y educativo, donde se busca mejorar las relaciones con los diferentes grupos, y, en la cual subyacen expectativas tanto de impacto económico, social y ambiental, emerge el concepto de software libre¹ como elemento transformador y dinamizador de las costumbres socio-económicas de los pueblos.

Un sistema de responsabilidad social en el escenario de las instituciones de educación superior se establece desde su misión, visión, proyecto educativo institucional –PEI-, y en general, la propuesta educativa centrada y formulada bajo criterios de compromiso social y aporte bajo la dinámica de formación para elevar los estándares de cualificación de los futuros profesionales. La responsabilidad social se convierte en una necesidad y en un compromiso que debe ser asumido por todos los actores o grupos institucionales que la ven como indispensable en un mundo globalizado; sin embargo, no puede ser asumido desde una práctica donde únicamente intervenga el fin económico, dejando de lado aspectos culturales, políticos, ambientales y educativos, que en esencia, deben constituir las dinámicas actuales de cambio.

El software, entendido como el conjunto de programas, instrucciones y reglas informáticas para ejecutar ciertas tareas en un computador, donde convergen relaciones inter y multidisciplinarias, permite una interacción compleja que no solamente abarca los aspectos técnicos y tecnológicos que se constituyen a partir de las necesidades particulares de individuos o grupos. El software permite una interacción compleja con los mismos componentes que en esencia involucran a la responsabilidad social: economía, cultura, política, educación, y medio ambiente.

El encuentro software-economía se da desde una perspectiva utilitarista que encuentra en los códigos e instrucciones un camino para aplicar los diversos modelos administrativos y comunicacionales propuestos desde las teorías administrativas y gerenciales, generando una simbiosis entre dos elementos que en la actualidad generan gran expectativa: el software como propulsor de la gestión del conocimiento, y el factor económico como eje transversal de generación de riqueza, o empobrecimiento de pueblos. El software como elemento fundante y tecnológico en la generación de conocimiento se ha desarrollado bajo diversos matices. Uno de ellos tiene que ver con la propuesta de valor generada a partir del software propietario, que corresponde a la elaboración de programas y aplicaciones que tienen un valor comercial legítimamente establecido. Bajo este modelo, el software se convierte en un producto que puede ser distribuido y comercializado bajo diversos parámetros comerciales, amparados por ley en muchos casos, por patentes y registros que buscan salvaguardar el ingenio de las personas naturales o jurídicas que lo crean. En este modelo, el comprador paga un valor por el derecho de utilización de dicha creación; en pocas palabras, adquiere el resultado final de una creación que puede considerarse intangible.

El software propietario surge a partir de la manipulación de códigos estructurados, concebidos y operados por personas que adquieren el conocimiento adecuado para realizar dichas creaciones.

¹ Definición de Software Libre por Richard Stallman en su libro *“Software Libre para una Sociedad Libre”*: El software libre es una cuestión de libertad, no de precio. Para comprender este concepto, debemos pensar en la acepción de libre como “libertad de expresión”. Con software libre nos referimos a la libertad de los usuarios para ejecutar, copiar, distribuir, estudiar, modificar y mejorar el software.

Si un usuario del producto determina que requiere de dichos códigos para efectuar modificaciones por sí mismo o por terceros, el creador del producto podrá negociar los códigos fuentes², de tal suerte que el comprador tendrá que pagar un valor muy elevado por dichos códigos, que tendrán como característica fundamental, el estar abiertos y dispuestos para su manipulación.

La educación que vienen recibiendo los estudiantes de las carreras tecnológicas enfocadas a la Ingeniería de Sistemas, Ingeniería Informática, Tecnología en Sistemas, y Técnica en sistemas, por mencionar algunas que se encuentran activas en instituciones educativas latinoamericanas, forman tradicionalmente al estudiante bajo el modelo de software propietario; el cual impone desde un comienzo restricciones que van desde los sistemas operativos³, los cuales se encuentran encriptados⁴ y protegidos por códigos que limitan el acceso al conocimiento, pasando por los lenguajes de programación⁵, que en esencia son objeto de las mismas restricciones, con el agravante, que el estudiante diseña el programa o aplicativo a partir de un conocimiento mínimo, en el cual, no puede intervenir desde el conocimiento básico que permita una mayor interacción con las herramientas que “aprende a usar y aplicar”.

El resultado de la formación que adquiere el estudiante bajo el modelo de software propietario limita al profesional; concentrando en él, un cumulo de conocimientos restringidos que le servirán para desarrollar su quehacer profesional; sin embargo, el hecho de omitir la opción de software libre, que en esencia corresponde al mismo conjunto de programas, instrucciones y reglas informáticas, pero de manera abierta y dispuesta para que cualquier persona pueda tener la libertad de hacer con ellas lo que le perezca, produce lo que he llamado el efecto “Cíclope”⁶, que corresponde a tener una sola visión en cuanto a elementos tecnológicos se refiere, y ponerla a disposición de una comunidad, organización, o grupos de personas. El efecto que produce la limitante de utilizar una sola propuesta (software propietario), generalmente excluye a los estudiantes en cuanto a conocimiento se refiere, y a los profesionales, en opciones que respondan a las necesidades tecnológicas de las pequeñas y medianas empresas, que en general, corresponden a la masa crítica de organizaciones en los estados latinoamericanos.

² Corresponde al conjunto de instrucciones escritas en algún lenguaje de programación de computadores, hechas para ser leídas por seres humanos, y para ser transformadas por alguna herramienta de software (compilador, intérprete, ensamblador) en lenguaje de máquina o instrucciones ejecutables en la máquina.

³ El Sistema Operativo se define como el software responsable de controlar y administrar las operaciones del hardware y del sistema básico. Además, provee una plataforma bajo la cual se ejecutan aplicaciones.

⁴ Consiste en la traducción de los datos a un código secreto. La encriptación es la manera más efectiva de lograr la seguridad de los datos. Para leer un archivo encriptado debe tener acceso a una clave o contraseña secreta que le permita decodificarlo.

⁵ Un lenguaje de programación es un lenguaje artificial que puede ser usado para controlar el comportamiento de una máquina, especialmente un computador. Estos se componen de un conjunto de reglas sintácticas y semánticas que permiten expresar instrucciones que luego serán interpretadas.

⁶ Cíclope, del latín cyclops, -ōpis, y del griego κύκλωψ, corresponde a un gigante de la mitología griega con un solo ojo. Los Cíclopes de la primera generación eran hijos de Urano y Gea, y conocidos artesanos y constructores. Eran seres con un temperamento horrible. Según Hesíodo eran fuertes, testarudos, y de bruscas emociones. Eventualmente sus nombres llegaron a ser sinónimo de fuerza y poder, y se usaban para referirse a armas especialmente bien manufacturadas.

Como lo menciona Richard Stallman⁷ en su libro: “Software Libre para una Sociedad Libre” al referirse a las libertades, es preciso que el estudiante tenga la posibilidad de estudiar el funcionamiento del programa y adaptarlo a sus necesidades específicas (libertad 1), con ello se le está dando la posibilidad de proveer soluciones completas a comunidades y organizaciones que requieren de software de calidad para el desarrollo de las actividades propias, y para incorporar la tecnología como un elemento que permita la evolución y el crecimiento sostenido. La responsabilidad social se aplica bajo el concepto de utilizar y adaptar lo que ya existe, y es libre, para actividades que requieren del elemento tecnológico como eje central para la proyección, no únicamente económica, sino social, cultural y ambiental.

La proyección de la responsabilidad social bajo el concepto de software libre se aplica, teniendo en cuenta la posibilidad de utilizar y adaptar el código fuente ya desarrollado, y en la perspectiva económica organizacional, en la mejor utilización de las inversiones, y en distribución de los recursos destinados a adquisición de licencias, programas, y aplicativos de software, para lograr en las organizaciones una capacitación constante que permita una cualificación de todo el personal. En síntesis, una empresa podrá utilizar los recursos económicos destinados a la adquisición de software, para actividades de carácter formativo y de desarrollo humano, entre otras, que busquen el equilibrio social.

Desde una perspectiva ambiental, la responsabilidad social se da a través de la libertad No.1, cuando el software que ya ha sido desarrollado es puesto a disposición de toda la humanidad sin mayor restricción que la que impone el conocer como implementarla en otros escenarios diferentes a los de su desarrollo inicial. La implementación de soluciones ya desarrolladas reduce los efectos contaminantes que son necesarios en la actualidad para desarrollar software, reduce la inversión en recursos naturales, y en toda la cadena de producción necesaria, y que alguna manera afecta nuestro ecosistema. La reflexión que surge a partir de la responsabilidad ambiental que todos tenemos, se articula con el hecho de no volver a invertir en recursos contaminantes para realizar actividades que han sido previamente desarrolladas.

Desde una perspectiva cultural el software libre a través de la utilización de los programas y aplicaciones ya desarrollados nos acerca a una dinámica colaborativa con personas de diferentes culturas, que tienen como eje motivador, el ánimo de compartir conocimiento sin mayor retribución que la satisfacción propia por la labor desarrollada. El Ingeniero, el Tecnólogo y el técnico de sistemas que se dedican al desarrollo de software, podrán encontrar en el esquema propuesto por el software libre un campo de desarrollo transcultural que indiferente a su posición política, religiosa y filosófica, le traerá amplitud de conocimiento para una sociedad que requiere, ahora más que nunca, elementos colaborativos en función de mejores posibilidades para nuestros estudiantes.

⁷ Richard Matthew Stallman, - Manhattan, Nueva York, 16 de marzo de 1953. Programador y la figura más relevante del movimiento por el software libre en el mundo. Sus mayores logros como programador incluyen el editor de texto Emacs, el compilador GCC, y el depurador GDB. Su influencia se debe al establecimiento de un marco de referencia moral, político y legal para el movimiento del software libre, como una alternativa al desarrollo y distribución del software privativo. Es también inventor del concepto de Copyleft (aunque no del término), un método para licenciar software de tal forma que éste permanezca siempre libre y su uso y modificación siempre se reviertan en la comunidad.

En cuanto a lo que concierne a la libertad No.3 descrita por Stallman en su libro, y que promulga el mejoramiento de los códigos de un programa para luego publicarlo para el bien de toda la comunidad (el acceso al código fuente es condición indispensable para ello), recrea la posibilidad de compartir códigos fuentes de programas, métodos y estilos de programación como elemento que rompe con el egoísmo que trae implícito el software propietario. La filosofía que propone el software libre abre una posibilidad que va mas allá de la obtención de un código sin pagar nada por ello. Establece un nuevo diálogo tecnológico que rememora aquellos días de la década de los años cincuenta y sesenta del siglo pasado, en los cuales, Ingenieros compartían abiertamente sus creaciones con otras personas para hacer de la computación un elemento del desarrollo tecno-científico de la humanidad. El código se compartía abiertamente sin el egoísmo que representa hoy en día el secreto empresarial. Los Ingenieros, Tecnólogos y Técnicos de la época, generaron una propuesta que resurge en la actualidad con la posibilidad que nos ofrece el software libre como dinámica propia de la responsabilidad social.

De acuerdo a la posibilidad de ejecutar un programa, sea cual sea el propósito, y descrito bajo la libertad 0 que propone Richard Stallman, es importante mencionar el grado de compromiso que se logra con ello, ya que tener la posibilidad de adquirir un programa y ejecutarlo con total libertad, y sin el dilema moral que representa utilizar software adquirido ilegalmente, genera un mayor grado de responsabilidad que surge por el respeto que se debe tener al utilizar una creación que no es propia, y que de hecho no se paga por ella. El grado de compromiso que un estudiante adquiere al utilizar y proponer software libre, es equiparable con la responsabilidad y la ética que debería adquirir a través de su proceso formativo. El hecho de proponer la utilización de software libre para una organización supone una inmensa carga de índole profesional, moral y ético, debido a que el conocimiento sobre la herramienta debe ser profundo y enfocado a las necesidades específicas requeridas; en tal sentido, emerge nuevamente la responsabilidad social como presupuesto planificado en la implantación de un sistema que debe constituir el desarrollo tecnológico para dicha organización; de no ser así, el software no pasaría de ser una creación artística que no estimularía el desarrollo a nivel mundial.

Desarrollar software libre o propietario no tiene sentido bajo el esquema actual de la globalización, en el cual subyacen elementos socio-económicos, si no se logra definir claramente el motivo por el cual se diseñara y construirá un programa o aplicación completa. La orientación del desarrollo, ya sea bajo un modelo propietario, que en ciernes estipula el cobro como fin último comercial bajo una transacción en la cual interviene la oferta y la demanda, y que orienta el proceso bajo la filosofía de estimar un valor en concreto por cada paso que demarca dicho desarrollo, o el modelo de software libre, que establece una relación distinta en la cual se antepone el conocimiento y la colaboración para desarrollar un proyecto, en el cual está presente un modelo comercial diferente al del software propietario, debido a que el software, es libre, pero no gratis, generará una matriz de decisiones en la cual el usuario definirá el modelo que más le convenga. Si se orienta por el software propietario, es seguro que tendrá un cobro por cada elemento constituyente del proyecto. Si la decisión se orienta por el software libre, es justo que el modelo lo lleve a pagar por la asesoría en la implementación, más no por el desarrollo del software requerido, si ya el producto ha sido desarrollado.

En cuanto a la propuesta de Stallman, respecto a la libertad de distribuir copias del software libre y ayudar así a otras personas, propuesto bajo lo que denomina la libertad No.2, estimula el aspecto colaborativo y solidario que abarca además una posición política, en la cual resalta el beneficio común sobre el beneficio individual. La libertad de distribuir copias para que las personas las utilicen con toda libertad, formula una dinámica diferente a la expuesta por el modelo propietario, ya que propone que cualquier persona podrá distribuir copias de software, que inclusive no son de su creación, sin la problemática actual acerca de los derechos de autor y las patentes. Con la libertad No.2, se propone la creación de un modelo diferente de negocios en torno al software libre, ya que incita a las personas inmersas en el estudio del software y los códigos, a conocer perfectamente dichas creaciones para poderlas ofrecer a las organizaciones y a las personas interesadas. La responsabilidad social encuentra a partir de la distribución de copias con entera libertad, un campo fecundo para la construcción de un modelo de negocios basados en la asesoría y consultoría.

La educación superior en América Latina marca un camino que debe construirse bajo estándares basados en la innovación y el desarrollo, de tal manera que las propuestas curriculares basadas en un solo modelo no podrán ser admisibles. El efecto cíclope como se mencionó anteriormente, ha venido marcando el quehacer de las instituciones educativas en centro y sur América, excepto Brasil, México y en parte Chile. Las demás naciones estamos siendo consumidoras y no productoras de tecnología informática. Las propuestas curriculares de programas como Ingeniería de Sistemas, Ingeniería Informática, Ingeniería de la Computación, Tecnología en Sistemas, Tecnología en Informática, y las Técnicas en Sistemas, entre otras, han sido construidas con criterios que se preocupan más por el hacer, que, por el saber hacer. Los currículos se ven influenciados por el modelo de software propietario que estimula la adopción de modelos de construcción de software bajo estándares que no aplican a las realidades latinoamericanas, por ello, es normal encontrar que un ingeniero, que en esencia debería estar abierto a todo lo que le propone la tecnología, se convierta en una persona que aplica un conjunto de reglas pre-establecidas, y que no es capaz de inventar o ingeniar nuevas técnicas, metodologías y lenguajes que le permitan en verdad considerarse un Ingeniero en todo el sentido de la palabra.

Desde luego, la responsabilidad de la formación de las personas que están inmersas en el mundo de las tecnologías de la información y las comunicaciones, corresponde a las instituciones educativas. Son ellas las encargadas de realizar las consultas tanto a nivel local como internacional acerca de las competencias, tendencias y perfiles que nuestros profesionales deben adquirir para afrontar los retos que implica las disciplinas propias de la información y las comunicaciones, y necesariamente, la globalización como escenario altamente competitivo. Organizaciones como ACM⁸, IEEE⁹, y ABET¹⁰, entre otras, construyen las competencias propias

⁸ ACM - Association for Computing Machinery. Fundada en 1947, es la primera sociedad científica y educativa acerca de la computación. Se encarga de publicar revistas y periódicos relacionados a esta área. Su número de afiliados son unos 78.000, y su sede está en la ciudad de Nueva York (EE.UU.). Su principal competencia es la IEEE Computer Society, aunque la ACM se encarga más de la teoría informática y las aplicaciones de software, en tanto la IEEE en el hardware y la estandarización.

⁹ IEEE - The Institute of Electrical and Electronics Engineers. Instituto de Ingenieros Eléctricos y Electrónicos, una asociación técnico-profesional mundial dedicada a la estandarización, entre otras cosas. Es la mayor asociación

que deben tener los egresados de las carreras relacionadas con los sistemas; sin embargo, parece ser que en la construcción curricular de los programas, las recomendaciones de estas organizaciones son asimiladas como simples tendencias que no pasan de ser “ideales de formación para países desarrollados”, y mal interpretadas, cuando al interior de los programas se ven como propuestas metodológicas que se articulan a esquemas pasados de moda, y que en nada se compadecen de los últimos adelantos tecnológicos.

La institución de educación superior que desee articular el ingenio y la innovación a través de herramientas tecnológicas como es el software, independiente si es propietario o libre, debe estructurar un plan que va mucho más allá de enseñar cursos y capacitar en una herramienta determinada. El Ingeniero debe recibir, más que una materia, un proyecto que venga estructurado con bases sólidas que permitan una mayor interacción con los saberes propios de la disciplina. El conocimiento debe venir estructurado desde las ciencias básicas afines con el programa, en el cual, la investigación y la aplicación se den de manera transversal durante todo el proceso formativo. Durante la etapa formativa es importante que el estudiante tenga acceso a todas las herramientas que se deben articular al proceso enseñanza-aprendizaje. El punto central es que el mismo estudiante elija conscientemente acerca de cuál de ellas le permitirá realizar el mejor acercamiento al conocimiento, y si ellas le permitirán incorporar elementos innovadores a las creaciones que propone.

El papel que juega la institución de educación superior en la elección de las herramientas que tanto docentes como estudiantes involucrarán en el proceso formativo de este último, marca la diferencia entre países desarrollados y países en vías de desarrollo. Los primeros elegirán las herramientas más apropiadas de acuerdo a las intencionalidades de formación en conjunto, y de acuerdo a las interacciones que permita la herramienta con los objetivos propuestos, conforme a las necesidades de innovación y desarrollo que se requiera; los segundos, realizarán la elección de acuerdo a las tendencias del momento, y conforme a la orientación que posiblemente marque tendencia una institución educativa de renombre dentro de su propio contexto.

El software libre no trata de excluir al software propietario, se trata de encontrar las mejores herramientas que permitan una mejor interacción entre los objetos de formación propios del programa, con los elementos de desarrollo y práctica en los cuales los estudiantes logren adquirir

internacional sin fines de lucro formada por profesionales de las nuevas tecnologías, como ingenieros eléctricos, ingenieros en electrónica, científicos de la computación, ingenieros en informática e ingenieros en telecomunicación. Su creación se remonta al año 1884, contando entre sus fundadores a personalidades de la talla de Thomas Alva Edison, Alexander Graham Bell y Franklin Leonard Pope. En 1963 adoptó el nombre de IEEE al fusionarse asociaciones como el AIEE (American Institute of Electrical Engineers) y el IRE (Institute of Radio Engineers).

¹⁰ ABET - Accreditation Board for Engineering and Technology, Inc., acredita en los Estados Unidos programas de estudio de universidades y colleges en: ciencia aplicada, computación, ingeniería y tecnología. ABET es una federación de 33 sociedades profesionales, incluyendo: AIChE (American Institute of Chemical Engineers), ASCE (American Society of Civil Engineers), ASME (American Society of Mechanical Engineers), IEEE (Institute of Electrical and Electronics Engineers), IIE (Institute of Industrial Engineers), ISA (The Instrumentation, Systems, and Automation Society) y SME-AIME (Society for Mining, Metallurgy and Exploration, Inc.), con más de 70 años de experiencia en aseguramiento de la calidad de la educación superior. Ha otorgado la acreditación a unos 2500 programas de estudio en más de 550 universidades y colleges.

los conocimientos, competencias y perfiles, teniendo en cuenta que la innovación y el ingenio deben estar presentes en todo el escenario formativo. La elección de la herramienta generalmente es definida por la dirección del programa, es ella la que dice el grado de interacción con el objeto de formación propio de la carrera; sin embargo, se encuentra que la elección de la herramienta está ligada al perfil de docente que orientará el curso; de tal suerte, que si el docente desconoce la herramienta, o no ha interactuado lo suficiente con ella, no podrá exigir el máximo provecho que el estudiante debe obtener de ésta. El resultado de la formación será incompleto y deficiente.

En cuanto al software propietario, y volviendo a los planteamientos iniciales, habría que pensar si el hecho de estar cerrado bajo códigos encriptados que le aseguran la confidencialidad, servirán para formar al estudiante de la disciplina tecnológica que requiere estudiar, interactuar, modificar y proponer alternativas a partir de un código ya desarrollado. Es seguro que en el software propietario la interacción que logre el estudiante se vea limitada al conjunto de reglas y opciones que le permite un producto totalmente cerrado. La argumentación de las multinacionales que agrupan el software propietario, se enfocará a que se está entregando un producto que no requiere de la intervención del diseñador de software, debido a que el producto es de alta calidad y en el cual confluyen diseños de última generación bajo herramientas que marcan la pauta del mercado en los actuales momentos. Esta argumentación, aunque lógica bajo los esquemas comerciales y a los derechos de autor, deja por fuera de la innovación y el ingenio a los estudiantes que se están formando y que requieren de apertura y colaboración para realmente ser competitivos en un sector de la economía que requiere permanente evolución.

Un producto compuesto por códigos encriptados, y que de acuerdo a las leyes de protección no se le podrá realizar ingeniería inversa¹¹ para conocer la arquitectura y el modelo con el cual fue construido, desestimula al estudiante de las disciplinas tecnológicas que buscan elementos y modelos de desarrollo de software como ejemplo para diseñar mejores códigos que articulen mayores probabilidades de éxito, en donde la innovación sea un factor de triunfo permanente. El software libre presenta un modelo donde el código es presentado al estudiante de manera abierta, sin restricciones, sin seguridades; de tal manera que inclusive un sistema operativo como linux¹², complejo y bien diseñado, se le ofrezca al estudiante con el código fuente, incluso, con todo el apoyo de la comunidad que lo desarrollo, para entenderlo, estudiarlo y hasta modificarlo. El modelo utilizado por el software libre permite adicionalmente a las libertades expresadas a través de su filosofía, un encuentro coherente y bien fundamentado en las instituciones de educación superior. Dicho encuentro se establece teniendo en cuenta tres elementos fundamentales que debería hacer parte de la responsabilidad social. El primero de ellos, calidad y apertura; con relación a esto, el proceso de formación de los estudiantes se debe acompañar con productos de

¹¹ El objetivo de la ingeniería inversa es obtener información a partir de un producto accesible al público, con el fin de determinar de qué está hecho, qué lo hace funcionar y cómo fue fabricado. Los productos más comunes que son sometidos a la ingeniería inversa son los programas de computadoras y los componentes electrónicos, pero básicamente casi cualquier proceso puede ser sometido a un análisis de Ingeniería Inversa. La ingeniería inversa es un método de resolución. Aplicar ingeniería inversa a algo supone profundizar en el estudio de su funcionamiento, hasta el punto de que podemos llegar a entender, modificar, y mejorar dicho modo de funcionamiento.

¹² Linux es un sistema operativo de distribución libre nacido a partir del UNIX. Trabajado, diseñado y publicado por programadores de todo el mundo. Es multitarea y multiusuario de 64 bits para PC. Fue desarrollado inicialmente por Linus Torvald.

software que permitan la praxis, la interacción completa, producto de los requerimientos específicos y que provea todo el ciclo de desarrollo de software. Para que ello se lleve a la práctica de manera innovadora y creativa es necesario contar con productos que tengan en su filosofía la calidad y la apertura para que los estudiantes puedan interactuar desde lo más sencillo, hasta lo más estructurado y complejo de los códigos. En segundo lugar, pertinencia y proyección. Al respecto, el acceso a productos tecnológicos de última generación acerca al estudiante y lo proyecta bajo una perspectiva innovadora que busca en términos generales, apropiación y prospectiva. Utilizar productos abiertos y de última generación en la formación de los estudiantes, genera una dinámica que estimula el ingenio y la creatividad, elementos esenciales que deben estar presentes en toda creación de software. Finalmente, accesibilidad y costo. Con relación a esto, la posibilidad de acceder libremente y en todo momento a códigos desarrollados bajo altos estándares, y contar con el soporte de toda una comunidad que a través de sitios web, foros, canales especializados, y en general todas las herramientas disponibles en internet, permiten que la investigación y la aplicación se conviertan en eje central del desarrollo a gran escala de estudiantes que en el futuro serán los empresarios del software.

El punto de vista que asume la institución de educación superior, en torno a la decisión de incorporar el modelo de software libre en los procesos de formación de los estudiantes de las disciplinas tecnológicas, enfocadas y orientadas a los sistemas y las comunicaciones, se verá complementado si no es excluyente con el modelo de software propietario. Se trata de no escoger un solo modelo, ya que una institución que elige lo uno o lo otro, corre el riesgo de sesgarse, lo cual excluye de beneficios a los mismos estudiantes. Aunque el modelo de software libre se articula perfectamente con el objeto de formación, es importante que la orientación y el enfoque que proponga la misma institución, vaya en sintonía con las tendencias tecnológicas del momento. El software propietario, producto de los avances y perspectivas del software libre, viene cambiando su filosofía en torno a la academia, ya que encontramos algunas multinacionales dedicadas a la producción de software que están permitiendo que sus productos sean descargados gratuitamente de internet, obviamente con ciertas restricciones, con el objetivo que sean usados en su versión académica para la formación de estudiantes y docentes.

La competencia por captar adeptos en uno u otro modelo aún comienza; sin embargo, estamos viviendo un momento donde el software libre empieza a tener una masa crítica, compuesta por estudiantes, docentes, directivos y empresarios, que están encontrando, además de productos de buena calidad, el escenario propicio para que la actualización tecnológica se convierta en eje de desarrollo de sus diferentes actividades. Por otra parte, las pequeñas y medianas empresas requieren, en momentos donde el proceso de globalización genera una alta competencia, productos de tecnología (software) que les permita competir en igualdad de condiciones. El acceso a la tecnología de estas organizaciones se venía haciendo en la medida que las posibilidades económicas lo permitían. Los costos por adquisición de licencias de software dificultaban el acceso a tecnología de punta. Con el software libre no existe el costo de adquisición por licencias, únicamente, el valor por asesoría y consultoría en el proceso de implementación y capacitación.

El modelo de software libre indudablemente está ligado a la responsabilidad social, la cual busca mejorar las relaciones con diferentes grupos a través de acciones que respondan al mejoramiento

de las condiciones económicas, sociales, educativas-culturales, políticas, y ambientales. En tal sentido, el software libre produce un encuentro que se argumenta desde la esencia filosófica como manifiesto de la libertad que promulga. La proyección que se logra con el estudiante, debería articularse al quehacer del egresado en el medio; en otras palabras, la responsabilidad social se ve proyectada a través de acciones que el profesional proponga ante la comunidad. La institución educativa transfiere todos los elementos de la responsabilidad social para que el estudiante los tome como insumo, con el objetivo de lograr desarrollo e innovación en el escenario escogido para tal fin.

La institución educativa que pretenda formar profesionales idóneos, con altas competencias disciplinares, con ética e integralmente formados en la responsabilidad social, no podrá ser excluyente con el modelo de software propietario, y mucho menos con el modelo de software libre. La responsabilidad que asume, debe ir en consonancia con el objeto de formación que requiere una sociedad en la cual subyace la competitividad, el crecimiento acelerado, y la toma de decisiones en tiempo real. Las propuestas curriculares deben acompañar la praxis en los laboratorios y en los sitios de práctica, a partir de las competencias que requiere un mercado que todos los días necesita mayor grado de conocimiento y aplicación. Lo anterior se logrará, solo si se abandona el efecto “cíclope”, el cual limita la capacidad del estudiante al incorporar un único modelo a su quehacer disciplinar.

Por último, es necesario establecer una política de formación y actualización tecnológica que incluya el tema de software libre para los docentes, ya que son ellos los que fundamentan y estimulan la utilización de herramientas tecnológicas. Si el docente universitario desconoce la filosofía del software libre y no ha interactuado directamente a través de esta, seguramente propondrá en la sala de sistemas, en el laboratorio, o en otros ambientes de práctica, la utilización de lo que ya conoce (software propietario), y que de alguna manera se convertirá en algo excluyente para el estudiante. En este caso, el presupuesto de la institución educativa por formar a sus estudiantes bajo la democratización del acceso a la tecnología, quedaría solo en simples propósitos, perjudicando el proyecto educativo del programa –PEP.

Bibliografía:

- Bonnet, Santiago. Problemas detectados en la difusión del Software Libre en las empresas. Revista Software Libre No.9, Noviembre/Diciembre de 2007.
- Stallman, Richard M. Software Libre para una Sociedad Libre. Traficante de Sueños. Madrid, España. ISBN: 84-933555-1-8.
- Arboleda Montoya, Bernardo H, Vargas Cano, Luis F, Serna Montoya, Edgar de J. Estado de la implementación, desarrollo y soporte al año 2004 de Plataformas GNU/GPL en el sector solidario y Pymes del área metropolitana de Medellín. Fundación Universitaria Luis Amigó – 2006. ISBN 9589768555.
- García, Juantomás, Romeo, Alfredo. La Pastilla Roja. Open Service Formación S.L. Madrid, España – 2003. ISBN: 84-932888-5-3.
- A.Abella, J. Sánchez, R.Santos y M.A.Segovia. El Libro Blanco del Software Libre. Versión 0.997-2003. Madrid, España.
- Matellán Olivera, Vicente, González Barahona, Jesús M, de las Heras Quirós, Pedro, Robles Martínez, Gregorio. Compilación de ensayos sobre software libre. Universidad Rey Juan Carlos. Madrid, España – 2004.
- García Castaño, Alba Nidia. Una Aproximación Conceptual hacia la Responsabilidad Social Institucional. Medellín, Antioquía – 2007.
- COMPROMETERSE. Cámaras de Comercio por la Responsabilidad Social Empresarial en las Pymes colombianas. ¿Qué es RSE? Disponible en Internet:
http://www.comprometerse.org.co/index.php?option=com_content&task=view&id=20&Itemid=35
- Instituciones Educativas y Responsabilidad Social. Lic. Julissa Mendoza - Dpto. Servicios Educativos. Artículo publicado en la Revista Proactivo - Año 2 - Numero 3. Lima, enero 2004.