

ASPECTOS PEDAGÓGICOS Y FILOSÓFICOS EN LA ALFABETIZACIÓN CON SOFTWARE LIBRE

Pedagogical and Philosophical Aspects in Literacy with Free Software

- Manuel de Jesús **Madé Zabala**
Universidad de Salamanca, España
Correo-e: manuelmade@usal.es

Resumen

Este trabajo se enfoca en algunos aspectos pedagógicos y filosóficos a considerar en la alfabetización con software libre, analizando la alfabetización tecnológica digital, que, en efecto, depende de la implementación o integración de algún tipo de software. El mismo, se aborda desde la pedagogía, la sostenibilidad y la colaboración en el software libre para la construcción de valores en los alfabetizados y la interrelación del ecosistema tecnológico educativo. A la vez, hace una aproximación filosófica de la libertad y la ética en el software libre, proponiendo su uso en la alfabetización tecnológica digital para la construcción de una cultura libre y sostenible. Por último, en su desarrollo, mantiene una dialéctica “natural” con el software privativo.

Palabras clave: software de código abierto; sostenibilidad; tecnología; educación; libertad.

Abstract

This work focuses on some pedagogical and philosophical aspects to consider in literacy with free software. Analyzes digital technology literacy, which, in effect, depends on the implementation or integration of some type of software. It approaches from the pedagogy the sustainability and the colaborativity in the free software for the construction of values in the alphabetized ones and the interrelation of the educational technological ecosystem. It makes a philosophical approach to freedom and ethics in free software. Consequently, it proposes to use it in digital technological literacy for the construction of a free and sustainable culture. Finally, in its development, it keep a “natural” dialectic with proprietary software.

Keywords: open source software; sustainability; technology; education; freedom.

COMO CITAR:

Madé Zabala, M. de J. (2018). Aspectos pedagógicos y filosóficos en la alfabetización con software libre. *Revista Caribeña de Investigación Educativa (RECIE)*, 2(1), 23-42. <https://doi.org/10.32541/recie.2018.v2i1.pp23-42>

1. Introducción

En todo modelo educativo están implícitos o explícitos los aspectos pedagógicos y filosóficos; aunque no hayan sido contemplados por sus creadores, son inherentes a la educación. Por tal motivo, en este trabajo se analizan dichos aspectos por su relevancia en la alfabetización con software libre, se examinan cómo están representados, cómo afectan los procesos de enseñanza aprendizaje, así como los valores que aporta.

Los computadores están constituidos por dos partes fundamentales: el hardware (parte dura o física) y el software (la parte lógica y abstracta). Pero cada vez más, el hardware deja de estar circunscrito al ordenador convencional, ya que puede estar en cualquier artefacto digital del entorno, por lo que el código que da las instrucciones para su funcionamiento, se ha convertido en uno de los activos más valiosos de las industrias desarrolladoras de software. Cualquier software puede superar por mucho el coste del hardware que lo contiene. Por ende, urge prestar atención a las implicaciones del software en la alfabetización. Philipson (2004, p. 2) manifestó al respecto: “El hardware es importante, pero en un sentido muy real, la historia de la tecnología de la información es la historia del software”.

La discusión referente al software “privativo” no está limitada solo a aspectos económicos, sino también a su propia naturaleza, “privar”. Por tal razón, se analiza como la alfabetización con software privativo conduce al individuo al utilitarismo, alienándolo con licencias que caducan con el tiempo y eliminan la libertad a los usuarios de inferir en su modificación, ya sea como sujeto colaborativo o activo.

Este trabajo asume que las mejoras de los procesos cognitivos que persiguen la pedagogía y la filosofía son esenciales para considerar “seriamente” la selección de herramientas tecnológicas para la alfabetización. En esta, el desarrollo de la cognición es primordial por el carácter mismo de alfabetizar. Integrar el software libre en la alfabetización da la oportunidad a los individuos de desarrollarse continuamente sin límites, libre y éticamente.

2. Acercamiento conceptual a la alfabetización

Desde las primeras acciones alfabetizadoras, el concepto alfabetización ha ido progresando con las revoluciones educativas y los cambios de paradigmas que surgen en cada sociedad. Pensar la alfabetización en la actualidad entreteje una serie de variables que nos hacen reflexionar sobre qué es, y, ¿cuál es el fin de la alfabetización actualmente? El interés por alfabetizar a las personas para que participen del desarrollo que se ha experimentado en cada contexto ha hecho que se replanteen los puntos de partidas y sus propósitos (Kuhn, 2006).

Para la Real Academia Española (RAE, 2017) alfabetizar es enseñar a alguien a leer y a escribir. Esta definición aparece en la segunda acepción. Mientras que para el *Oxford Dictionaries*, alfabetizar es la enseñanza de la lectura y la escritura de una lengua a una persona, en especial a un adulto. Sin embargo, la palabra inglesa para alfabetizar es *literacy*, tanto en *Oxford English Dictionary* (OED, 2017) como en el *Cambridge Dictionary* (CD, 2017), viene acompañada de una segunda acepción que abre un abanico de posibilidades y permiten sacar conclusiones amplias sobre su significado. Por ejemplo, para el *Cambridge Dictionary*: *literacy* significa “conocimiento de una materia en particular, o un tipo particular de conocimiento”, y pone el ejemplo: “La alfabetización informática se

está volviendo tan esencial como la capacidad de conducir un automóvil”. Y para el OED *literacy* es, “La capacidad de ‘leer’ un tema o medio específico; competencia o conocimiento en un área particular”.

De estas definiciones podemos hacer varias observaciones: en primer lugar, la definición más conservadora y tradicional que se puede entender por alfabetización, “enseñar a leer y escribir”, como un dominio básico de la lectura y la escritura para comunicarse a través de medios escritos con el entorno inmediato o comunidad de una misma lengua; en segundo lugar, la aparición de los adultos en la definición del *Oxford Dictionaries*.

En esta parte se hace un intento de agrupar a los destinatarios de la alfabetización por su edad, a los que se podría reflexionar con la pregunta ¿por qué no se habla de alfabetización del niño? La respuesta a esta pregunta, de modo simple y apelando a la naturaleza del desarrollo del niño, en entornos con acceso a escolaridad desde la infancia, sería que la alfabetización puede integrarse en el proceso de formación inicial al igual que se le enseña a identificar los colores. No obstante, este es un tema que se podría profundizar en una futura investigación; y, en tercer lugar, queda la siguiente interrogante: ¿por qué en español el significado de la alfabetización se queda rezagado en lo tradicional, especialmente usando un mismo diccionario (por ejemplo, Oxford)? Sin embargo, se puede apreciar que la traducción en inglés para *literacy* responde de forma más sensata a lo que se puede entender en la actualidad por alfabetización.

Siendo pragmático, hay que entender que “la manera en que se defina la alfabetización se reflejará en las metas y estrategias adoptadas, en los métodos de enseñanza y aprendizaje, en la formación de los educadores, en los materiales y los programas de estudio, y en la forma de evaluar y dar seguimiento a las acciones” (Infante & Letelier, 2013, p. 16). Pese a las ambigüedades en las definiciones que se puedan encontrar algunos prefieren hablar de “alfabetizaciones” (Braslavsky, 2003, p. 3).

2.1. El software en la alfabetización

Anteriormente se abordaron las dimensiones que ha alcanzado la alfabetización desde lo tradicional hasta la actualidad, para llegar a la integración del software en los procesos de enseñanzas alfabetizadoras. En la actualidad, puede alfabetizarse en cualquier área usando software, pues la versatilidad de las tecnologías informáticas ha servido para optimizar las mayorías de herramientas y artefactos que apoyan las actividades humanas. Estamos en lo que se puede llamar la era de los códigos informáticos. La integración de microprocesadores, la codificación y la conectividad de múltiples dispositivos en todo el mundo, el Internet de las cosas¹ ha germinado una sociedad que vive sumergida en el código informático (software).

Pese a todos estos avances, actualmente se registran importantes “brechas digitales” que han afectado el desarrollo de muchos países, llevando a los gobiernos a tomar medidas de contingencias. Para ello, han adoptado diferentes alfabetizaciones tecnológicas (alfabetización digital, alfabetización en las TIC o la Alfabetización mediática, etc.), que más allá de los llamados “nativos digitales”, intentan alcanzar a la población adulta rezagada.

¹ El concepto “Internet de las cosas” (*Internet of things*) que ha tenido gran acogida por la comunidad académica, que acuñó Ashton en el 1999, se fundamenta a partir de la integración de chip que se conectan a redes de Internet en aquellos dispositivos no convencionales para los usuarios comunes que navegan en la Internet, con el fin de recolectar datos que pueden utilizarse en múltiples propósitos.

Los computadores dependen de dos partes fundamentales: el hardware y el software, pero cada vez el hardware no está circunscrito al ordenador convencional, puede estar en cualquier dispositivo, por lo que el código que da las instrucciones a los artefactos y objetos, para su funcionamiento, se ha convertido en uno de los activos más valiosos de las industrias desarrolladoras de software. Cualquier software puede superar con mucho el coste del hardware que lo contiene. Así, conviene prestar atención al software y sus implicaciones en la educación.

El software, como lo conocemos hoy día, apareció junto con la tecnología computacional, pero su integración en la educación estuvo determinada a círculos académicos especializados. Su evolución y desarrollo lo popularizó haciéndolo más asequible a todas las personas, pero dejando un rastro de “brecha”, esos huecos que han quedado y que aún existen en los países en vía de desarrollo, regiones como África, Asia, Latinoamérica y el Caribe.

Existen tres términos de alfabetización de gran dependencia de software, que son: alfabetización digital, alfabetización en las TIC y la alfabetización mediática. Estas tres forman parte de las alfabetizaciones sociales que se solapan con las tecnológicas, teniendo mucha importancia para algunos gobiernos

En cuanto a la educación escolar, en las últimas dos décadas, los países han ido integrado las TIC en el currículum educativo. Sin embargo, la falta de profesores calificados ha llevado a los gobiernos a crear programas de alfabetización en las TIC, con el fin de brindar la formación necesaria para enfrentar los desafíos de los nuevos paradigmas educativos que se gestaron con la aparición de las tecnologías informáticas y computacionales.

3. Alfabetización digital ciudadana

La construcción de una definición de la alfabetización digital hay que replantearla continuamente por la emergencia de desafíos tecnológicos y sociales. Las discusiones que se han desarrollado a lo largo de su aparición, en la identificación de los fines, han sido en torno a los sujetos o las tecnologías. Los sujetos, cuando el énfasis en la formación se enfoca en desarrollar individuos competentes que usan las tecnologías como herramienta y con cierto nivel crítico de lo que esta significa, o la tecnología, cuando se basa en el supuesto de que proveer el acceso a los ciudadanos lo convierte en ciudadanos digitales, aunque la formación sea elemental. Esta última es una acción típica de algunos gobiernos.

Gilster (1997) define la alfabetización digital como el conjunto de habilidades sociocognitivas mediante las cuales se puede seleccionar, procesar, analizar e informar del proceso de transformación de información a conocimiento. El acceso a las tecnologías digitales no define el conocimiento o el aprovechamiento que puedan hacer los individuos, el acceso representa el primer acercamiento a un universo de informaciones que necesitan ser filtradas para obtener los conocimientos puntuales sobre necesidades esenciales; es algo más que ratones y teclas (Gutiérrez, 2003). El sujeto y la tecnología forman un conjunto que merecen igual atención en la alfabetización digital, en este trabajo metodológicamente se partirá siempre del sujeto, el individuo con acceso a las tecnologías.

Es considerable la estrecha relación que existe entre la alfabetización digital con la mayoría de las alfabetizaciones actuales. Referente al uso de recursos y objetivos propio de la alfabetización digital, es la alfabetización que más se solapa con las demás. Por ello Gutiérrez (2003) prefiere cambiar el nombre de alfabetización digital por “alfabetización múltiple”. Su propuesta parte de la diversidad de temas y contenidos que se abordan en la alfabetización digital, agregando a estos la heterogeneidad del público a la que se destina.

El concepto alfabetización digital es el más usado en el marco de las políticas sociales de alfabetización tecnológicas de los gobiernos. Para estos proyectos alfabetizadores del “país”, se ha creado el concepto de “ciudadano digital o ciudadanía digital”, que consiste en la capacitación de los habitantes de un país en un conjunto de áreas que se consideran necesarias, para que un ciudadano tenga las competencias para interactuar y desarrollarse en la era digital. Esta tendencia se ha convertido en una meta en Occidente, especialmente en Latinoamérica, estableciendo una oportunidad de mercado para las empresas desarrolladoras de software, especialmente las más representativas a nivel mundial, que han creado estándares de certificaciones en alfabetización digital como son: Microsoft, Cisco Academy e International Computer Driving Licence o European Computer Driving Licence (ICDL/ECDL), esta última no desarrolla software, pero tiene una certificación en alfabetización digital que goza de un gran reconocimiento y aceptación en varios países a nivel mundial.

Hay otros programas que han sido desarrollados a nivel local, por algunos países y otras instituciones especializadas en la tecnología educativa, que tienen algunos modelos propios, pero en su mayoría dependen de las certificaciones de las tres antes mencionadas sirviendo como centros certificadores. Es evidente el monopolio en estas estrategias alfabetizadoras donde bailan los políticos y las grandes corporaciones, con mucha singularidad, Microsoft Academy y sus sistemas Windows y MS Office.

4. Alfabetización en las TIC y el software libre

El acceso a las TIC crece exponencialmente en casi todo el mundo gracias a la adopción de políticas educativas enfocadas en el cierre de la brecha digital y el aprovechamiento de las mismas en los procesos de enseñanza aprendizaje. Estas medidas presentan la disyuntiva de que los usuarios de las TIC son sometidos a un exceso de información que complica la concentración y el enfoque para los cuales se están introduciendo. Algunos hablan de infoxicación (Cornella, 2010)², para referirse a la abrumadora carga de información a las que están expuestos los usuarios, esencialmente en Internet. La preocupación de la educación es cómo orientar las TIC para que sirvan de provecho al proceso de enseñanza aprendizaje. Esta preocupación ha creado tanto interés, que se ha convertido en materia de estudios académicos en universidades y centros de investigación.

La alfabetización en las TIC se fundamenta básicamente en enseñar a los profesores a usar las tecnologías como “medio” (recursos), en un proceso de enseñanza aprendizaje en el que los alumnos participan de manera interactiva. Son recursos educativos no tradicionales pizarras, tizas, rotafolios, papelógrafo, libreta de notas, etc., en ocasiones son complementarios y en algunos casos los sustituyen por completo. La formación mediada por las TIC depende ampliamente de los editores de textos de la Internet, de múltiples recursos educativos digitales y las webs.

La alfabetización digital, en la ejecución, es una aliada de las TIC, por lo que ambos conceptos suelen abordarse en conjunto en las investigaciones y proyectos de formación. La diferencia más marcada entre ambas se encuentra en que la alfabetización en las TIC se orienta a los profesores que no solo usan herramientas tecnológicas para la educación, sino que crean un entorno que responde a una metodología de enseñanza previa. Sin embargo, la alfabetización digital está dirigida a toda la población y usa las TIC como parte de su metodología de enseñanza. La alfabetización en las TIC

² El concepto “infoxicación” es una interpretación del término inglés “information overload” usado por Alvin Toffler en su libro *Future Shock* publicado en el 1970.

es un medio y la alfabetización digital es un fin; lo dicho marca la diferencia en la forma de abordar el uso de algunos softwares que comparten, porque ambas disponen, en algunos casos, del uso de la misma tecnología.

El conocimiento, el uso y la implementación de software libre en las TIC es esencial para la construcción posterior de una cultura que se apropie de los beneficios de vivir con la tecnología y ser libre. Las herramientas que se usan en los entornos educativos tienen gran efecto sobre los estudiantes, quienes pueden ser fácilmente seducidos por ellas sin ver un horizonte más allá. Esto existe gran responsabilidad de los docentes para elegir las mejores herramientas que contribuyan en el desarrollo crítico de los alumnos, evitando alienación o sumisión a ninguna compañía que quiera forzar su voluntad.

El software libre dispone de suficientes herramientas para llevar a cabo una educación de calidad prescindiendo de las herramientas privativas. No pueden medirse con exactitud la cantidad de herramientas libres elaboradas para la educación, pero sí se puede decir que hay muchas que no tienen competencias en el sector del software privativo, por ejemplo, los softwares de la robótica educativa. En este renglón, el software libre controla la demanda en combinación con el “hardware libre”, en que se vende para diferentes aplicaciones.

5. Alfabetización mediática informacional y digital

A diferencia de la alfabetización en las TIC, la alfabetización mediática estará interesada en los medios por donde circula la información y el potencial que representan para conectar con los individuos. No solo está suscrita a los medios digitales o tecnológicos convencionales, sino que se interesa por medios tradicionales de información. Para entender la Alfabetización Mediática e Informacional (AMI), en la UNESCO (2017) se plantea del siguiente modo:

La alfabetización mediática y la alfabetización informacional se consideran tradicionalmente como campos separados y distintos. La estrategia de la UNESCO reúne estos dos ámbitos como un conjunto combinado de las competencias (conocimientos, habilidades y actitudes) necesarias para la vida y el trabajo de hoy. La AMI abarca todos los tipos de medios de comunicación y otros proveedores de información como bibliotecas, archivos, museos e Internet, independientemente de las tecnologías utilizadas.

Pese a la inclusión de medios no digitales la necesidad de sistemas informáticos (software), en esos medios, es importante para la automatización y eficiencia en la gestión de los datos, a esto se le agrega que el cúmulo de información que transita por las redes de Internet necesita especial atención. De todas formas, el aporte de la alfabetización mediática a la educación es inconmensurable, las formaciones en alfabetización mediática vienen a cubrir necesidades de la sociedad de la información y el conocimiento, que son inherentes al desarrollo intrínseco de la tecnología informática interconectada globalmente.

Es aceptable que la UNESCO combine la alfabetización mediática con la alfabetización informacional para su plan estratégico, partiendo del valor que tienen los contenidos de esos medios que son los que afectan de forma directa o indirecta el desarrollo social de los individuos. Es importante desde el punto de vista pedagógico entender el efecto de los medios en la construcción del conocimiento de las personas, en especial las que dependen de software para responder a los objetivos que planteamos aquí.

La triada compuesta por la alfabetización digital, la alfabetización en las TIC y la alfabetización mediática tiene estrecha conexión en relación a los destinatarios, métodos y herramientas, aunque las tres representan al grupo de las alfabetizaciones sociales que enseñan y usan las tecnologías. La excepción se puede apreciar solo en la alfabetización mediática que puede recurrir a medios no digitales.

Se le suele decir “alfabetizaciones tecnológicas”, sin embargo, es anacrónico y pretencioso hablar de alfabetización tecnológica cuando solo se centran en un área de la tecnología (la tecnología informática o computacional). Hay alfabetizaciones que usan o enseñan un área de la tecnología, por eso es preferible hablar de “alfabetización tecnológica más el área específica del uso o conocimiento que se procura enseñar”. Por ejemplo, “alfabetización científica tecnológica”, porque puede darse el caso de alfabetización tecnológica que prescinden de la computación, como “la alfabetización eco-tecnológica”.

Es evidente en las mayorías de alfabetizaciones actuales la presencia de software, lo cual sugiere importancia al tipo y origen de los softwares que se implementen. Forman parte de los recursos y metodología de los sistemas educativos. En pocos lugares se prescinde de algún recurso digital para la elaboración, distribución o ejecución de los proyectos educativos, no importa sus dimensiones. Se hace necesario revisar su procedencia y viabilidad.

6. La industria del software de cara a la alfabetización

En cualquier seminario, curso o taller, traer a colación el control que tienen las grandes compañías como Google, Microsoft, Facebook, Apple, etc.; genera tanto pánico como los rumores sobre una catástrofe. A esto, se le agrega que los dueños de estas corporaciones están entre los más ricos del mundo, algo que podría hacer que los expertos economistas de inicio del siglo pasado entren en “shock”. En las últimas décadas ha habido más emprendimientos exitosos a gran escala en la industria del software que en otras áreas. El software es el “backend” del siglo XXI. No prestarle atención a este “poder” sería ingenuo.

Paradójicamente, el software nació libre. Los fundamentos teóricos y prácticos de la ciencia computacional que se utilizan en el desarrollo del software, y que se siguen utilizando en la actualidad, nacieron libres. Los algoritmos matemáticos y artefactos creados por Pascal, Leibniz, Babbage, Turing, Lovelace, etc. fueron los aportes que dieron paso al surgimiento de las innovaciones que han hecho posible la tecnología computacional actual. Aunque, esta libertad duró hasta que el software pasó a convertirse en un “producto” más del mercado y no en un conocimiento abierto para la humanidad.

Una pregunta para este tiempo sería ¿qué se entiende por software? La definición más extendida del software en los programas de alfabetización consiste en establecer su diferenciación con el hardware, para así definir como está compuesto el ordenador o computadora. Es el intento de establecer la diferencia entre aquella parte tangible que los usuarios usan para acceder a las funcionalidades intangibles, lógicas y virtuales del ordenador, o sea, al software. Por ende, el software es aquella parte inmaterial y abstracta fundamentada en la lógica computacional que los diferentes componentes del hardware, sobre el cual se encuentran instalado, comprenden y pueden establecer comunicación a través de lenguajes informáticos.

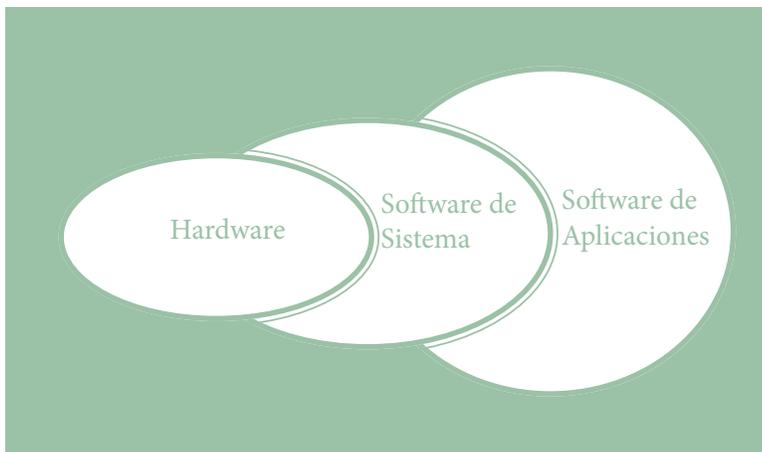
Algunas definiciones aluden al software como aquella parte que “no se puede ver” del ordenador, generando confusión para los usuarios que se sienten más vinculados con el software que con el

hardware, al que definen como “parte visible”. John Tukey (en Leonhardt, 2000) fue el primero en usar el término “software” en el año 1958, para referirse al programa que ejecutaba una calculadora electrónica.

Según la RAE el software es: “Voz inglesa que se usa, en informática, con el sentido de ‘conjunto de programas, instrucciones y reglas para ejecutar ciertas tareas en una computadora u ordenador’. Puede sustituirse por expresiones españolas como programas (informáticos) o aplicaciones (informáticas), o bien, en contextos muy especializados, por soporte lógico (en oposición al soporte físico; hardware)”. Esta definición constata con la concepción del software determinado a ordenadores o computadoras, no se habla de computadora como la unidad central de procesos, CPU (*central processing unit*), sino al conjunto de dispositivos que caracterizan la máquina de computación convencional, que se encuentra en cualquier lugar. Sin embargo, las heterogeneidades de los hardwares sugieren una definición más amplia, como la que brinda ESPASA (2017), “El software es el nivel lógico de cualquier aparato electrónico capaz de ejecutar procesos complejos...”.

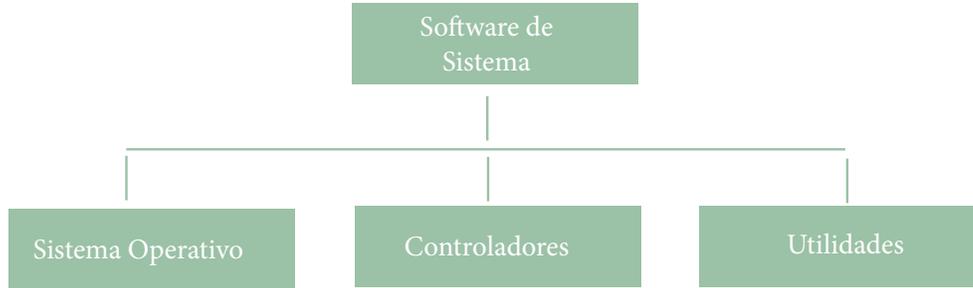
Debido a la función que realizan o la integración en los diversos dispositivos pueden llamarse programas, aplicaciones (comúnmente denominadas Apps, por la abreviatura de la palabra inglesa application), sistemas operativos, controladores, juegos, etc. El software a partir de su interacción con el hardware se puede dividir en software del sistema, y software de aplicación, estos dos a su vez integran otras subcategorías. En la figura 1 se muestra la estructura fundamental de la mayor parte de artefactos computacionales:

Figura 1: Interacción entre el hardware y el software (adaptación propia del aporte de Mzahma et al., 2015)



El ordenador o cualquier aparato que disponga de funcionalidades automatizadas por procesadores computacionales siguen esta estructura. Hay dispositivos que pueden disponer solo de un software de sistema que se encargue de controlar todas las funciones del hardware, como son la mayoría de electrodomésticos y algunas máquinas industriales que siguen procesos básicos, pero que están sustentados en lógicas computacionales, ejecutándose en los chipsets de una tarjeta electrónica. Veámoslos en el esquema de la figura 2:

Figura 2. Subcategoría de los software de sistemas (Mzahm et al., 2015)

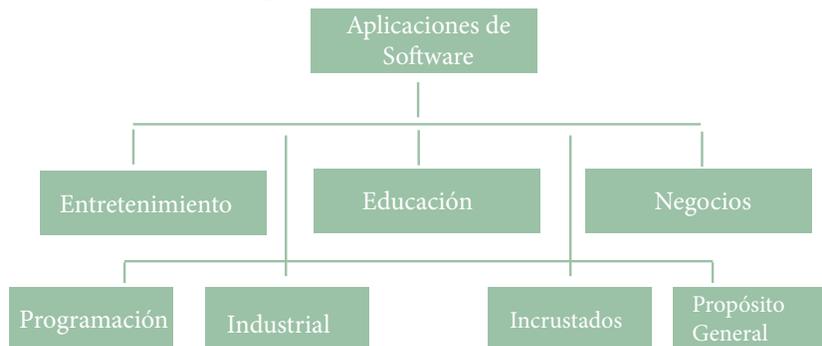


Esta categoría es la que accede a los recursos del hardware para intercambiar con los softwares de aplicaciones las peticiones específicas que pueden solicitar los usuarios. El desarrollo en este campo es uno de los más complejos y se necesitan, mayormente, conocimientos técnicos de programación de bajo nivel. El activo económico más potente de las industrias de software en esta área es el monopolio, convirtiéndola en la industria de software menos representativa y la menos divulgada entre los usuarios finales que adquieren los equipos con ellos preinstalados.

Sin embargo, en la categoría de aplicaciones se puede apreciar la diversidad que pueden adoptar los softwares para responder a actividades específicas de la vida humana en general. Son muchas las razones que explican la evolutiva heterogeneidad del software, pero la explicación fundamental se encuentra en el momento que los softwares pasan de ser líneas para ejecutar comandos, a una interfaz gráfica enfocada en los usuarios (también se le llama en el lenguaje informático GUI (Graphical User Interface)³. Esto hace posible que usuarios con poca o sin ninguna experiencia tomen un ordenador, y realicen tareas, siendo esto un gran aporte didáctico aprovechable para los diversos tipos de aprendizaje, incluyendo el auto-aprendizaje.

A continuación, la figura 3 nos muestra un esquema de las aplicaciones de software con algunas clasificaciones.

Figura 3. Subcategorías de los softwares de aplicaciones (Mzahm et al., 2015)



³ Son aquellos lenguajes de programación que se usan para el desarrollo de software, que pueden comunicarse directamente con el hardware y que traducen las órdenes para ponerlas a disposición de las aplicaciones, las cuales puedan hacer peticiones al dispositivo que lo contiene. Ejemplo: los controladores (en inglés *driver*) de impresoras proveen a los programas las informaciones necesarias para que puedan configurar sus características, con el fin de obtener una impresión de acuerdo a cada necesidad. El desarrollo de este controlador se hace a bajo nivel.

Cabe destacar dos conceptos importantes en el desarrollo de aplicaciones de software que son: el *backend* y el *frontend*. El *backend* se define como la parte de administración y de gestión, mientras que el *frontend* como aquella parte a la que los visitantes tienen acceso visual e interactivo. El *backend* se puede dividir en dos partes, una de cara al administrador como usuario y otra al desarrollador, en el software libre esta parte está totalmente abierta [open] y disponible a los adquirientes de un software para que se le hagan mejoras y cambios al código fuente. En los softwares privativos, esto no es legalmente posible. El *frontend* está limitada a la interacción con el usuario final o huésped que usa la aplicación. Estos términos tienen mucha importancia en el desarrollo de páginas web y aplicaciones móviles, pero pueden aplicarse a muchos softwares que disponen de interfaces gráficas.

7. El software y la cotidianidad

Definir el software puede ser simple, pero para la mayoría de usuarios identificar donde hay un software funcionando puede generar confusiones, ya que muchos usuarios tienden a fusionar el software y el hardware como uno solo. Por ejemplo, “el ordenador es Windows” o para referirse al uso de OS X de Mac, pueden responder “¡Lo siento! Soy usuario Mac”. Ambos hardwares pueden incorporar otros softwares que no sean ni Windows ni OS X. Por lo general, los usuarios de software libre son los más conscientes, capaz de distinguir el software del hardware.

En principio queda claro que el software depende del hardware para su ejecución, es su núcleo vital, la cuestión es saber ¿qué cosas están siendo asistidas por software? Es fácil identificar un software en un ordenador, smart phone o tablet, gracias a la interfaz gráfica e interactiva que provee a los usuarios. No obstante, si se mira en el entorno, con el conocimiento explicado más arriba, saltará a la vista un mundo de artefactos que tienen algún software detrás para distintas funcionalidades. La posibilidad de conectar todas las cosas amplía el espectro de la innovación en el desarrollo de software.

Estos avances representan otro desafío para la alfabetización y un gran aporte para la educación. La labor titánica de la educación es identificar cuáles son los aportes que benefician al desarrollo cognitivo y mejora de los procesos enseñanza aprendizaje, yendo más allá de la adopción tecnológica por tendencia o moda.

Existen ambigüedades entre algunas licencias que suelen usar las industrias del software y, por algunas razones, es común que los usuarios obvian las clausuras de licenciamiento cuando lo adquieren e instalan, confundiendo que parte del producto se ha comprado y hasta donde llega su libertad.

Hay dos categorías de software que podríamos identificar: “privativo” y “libre”. Ambos conceptos, aunque han sido muy difundidos, aún siguen discutiéndose en algunos escenarios.

7.1 El software privativo

Los términos “privativo” y “propietario” para referirse a algunas licencias de software, comenzaron a aparecer al intentar establecer diferencias entre las formas de distribuciones de software y la sutil estrategia de mercado que las compañías habían iniciado en la década de los 70 y parte de los 80 que restringían, privaban y cobraban por ellos. “Privativo” es el término que acuña Richard Stallman (2004) para designar aquellos softwares “no-libres”. Las empresas de software privativos

restringen aquellas acciones que pueden realizarse con, o en el software, a las que les pueden sacar algún tipo de beneficio, especialmente económico. Desde entonces, los usuarios solo compran una parte muy condicionada por las licencias del software.

7.2 El software libre

Trazar lineamientos sobre la liberación de software ha sido un camino intrincado. La propuesta más aceptada para categorizar el software libre ha sido la de Richard Stallman (2004, pp. 59-60), la cual establece que para que un software pueda ser libre, debe ofrecer a los usuarios las siguientes libertades:

- 1- Libertad para ejecutar el programa sea cual sea nuestro propósito.
- 2- Libertad para estudiar el funcionamiento del programa y adaptarlo a tus necesidades
- 3- Libertad para redistribuir copias y ayudar así a tu vecino.
- 4- Libertad para mejorar el programa y luego publicarlo para el bien de toda la comunidad.

Estas leyes, como Stallman les llama, son genéricas para las licencias de software libre, pero para evitar contradicciones con las traducciones del concepto free, en free software (software gratis o libre), desde el inglés al español, se suelen usar también los conceptos FOSS y FLOSS (Free Open Source Software y Free/Libre Open Source Software). En estos conceptos se aprecia la posibilidad de comercialización del software.

7.3. Aspectos pedagógicos de la alfabetización con software libre

El crecimiento exponencial de la tecnología se ha convertido en una oportunidad para la educación y, a la vez, en un desafío, puesto que, los recursos tecnológicos entrañan las mismas condiciones de análisis que los recursos tradicionales para su implementación en la educación. Es necesario que se identifiquen los aspectos pedagógicos que lo integran, para la elección de los que se adecuen con mejores resultados en los procesos de enseñanza aprendizaje. En el caso del software educativo Gallego Arrufat (1994) aclara, que es todo programa que de alguna forma responda a una determinada modalidad de Enseñanza Asistida por Ordenador (EAO) y desarrolle aspectos de tipo “instructivo”.

Barrón (1989, p. 686) dice, “Desde el punto de vista, la razón “pedagógica” no puede ser reducida a una razón “técnico-instrumental”, ya que supone una deliberación moral, crítica creativa, capaz de adecuar el conocimiento al contexto específico de acción, así como de “recrear” la secuencia de intervención en función de las situaciones creadas”. Estas dimensiones son elementos pedagógicos esenciales que deben ser contemplados en la alfabetización, y que cobran valor con la introducción de tecnologías de software libres, porque su “razón” va más allá del tecnicismo, integrando al contexto de acción colaborativamente.

Un aspecto pedagógico a considerar en el software es la sostenibilidad que puede ser sustentada desde el Objetivo de Desarrollo Sostenible (ODS) número 4, que contempla: “Garantizar una educación inclusiva, equitativa y de calidad y promover oportunidades de aprendizaje durante toda la vida para todos”. (Naciones Unidas, 2017). Si tomamos la parte que dice “...durante toda la vida para todos”, hay que considerar, entre otras cosas, las herramientas que se usarán en los procesos formativos. Sin lugar a duda, el software libre es la opción más pertinente, tomando en cuenta que los países con mayor tasa de analfabetismo están en vía de desarrollo y otros son pobres. En este

punto, la ONU no puede ser una promotora del desarrollo apoyando la obsolescencia programada en que incurren muchas compañías tecnológicas, porque entonces su propuesta no sería “inclusiva” ni “equitativa”. Stallman (2004) explica que el motivo que le llevó a comenzar el proyecto de desarrollo del software libre fue que le negaron los códigos fuente del programa controlador de una impresora. Y que no quería repetir lo que le hicieron con el resto del mundo.

La rápida caducidad de los productos tecnológicos genera preocupación en los usuarios. El software libre junto a los nuevos desarrollos de hardware libre, brindan a los usuarios la oportunidad de usar y rehusar las tecnologías para satisfacer sus necesidades y ajustarlas a su poder adquisitivo libremente. En cambio, si se alfabetiza con software privativo, se sume a los individuos en el consumismo y se mantiene a los sistemas monopolistas que sostienen las grandes corporaciones.

8. Reducción de la obsolescencia programada

Un artefacto puede estar electrónicamente y mecánicamente bueno, sin embargo, no ser funcional. Con tan solo unas líneas de códigos se pueden programar la vida útil del mismo. Esta práctica se le suele atribuir a compañías como Microsoft, Apple, Samsung (Moreno, 2017), fabricantes de impresoras, etc.

El software aporta a la caducidad del hardware de varias formas: a) a través de actualizaciones que exigen mayores recursos ralentizando el aparato; y b) con el lanzamiento de nuevas versiones de software que solo soportan un determinado hardware, dejando así de darle soporte.

Metodológicamente el ciclo de vida de la tecnología tiene grandes implicaciones para cualquier proyecto alfabetizador. Es evidente que el software libre es la opción más factible para hacer frente a esta aberración en la sostenibilidad. Cabe aclarar que alfabetizar con software libre no romperá “rotundamente” el ciclo del consumo, pero crea la posibilidad de reducirlo.

9. Comunidades de colaboradores

La colaboración ha sido esencial en la evolución del software libre. Las comunidades de colaboradores están por diferentes partes del mundo, brindando asistencia y soporte técnico en foros, blogs, chat, etc.; en la mayoría de casos, de forma gratuita. De estas colaboraciones han resultado las traducciones a otras lenguas y dialectos, que incluyen aplicaciones accesibles a comunidades aborígenes (Toboso, Feltrero & Maltrás, 2015). La capacidad metalingüística y accesible la convierte por antonomasia en tecnología inclusiva. Esta labor al software privativo le sería muy tediosa y costosa.

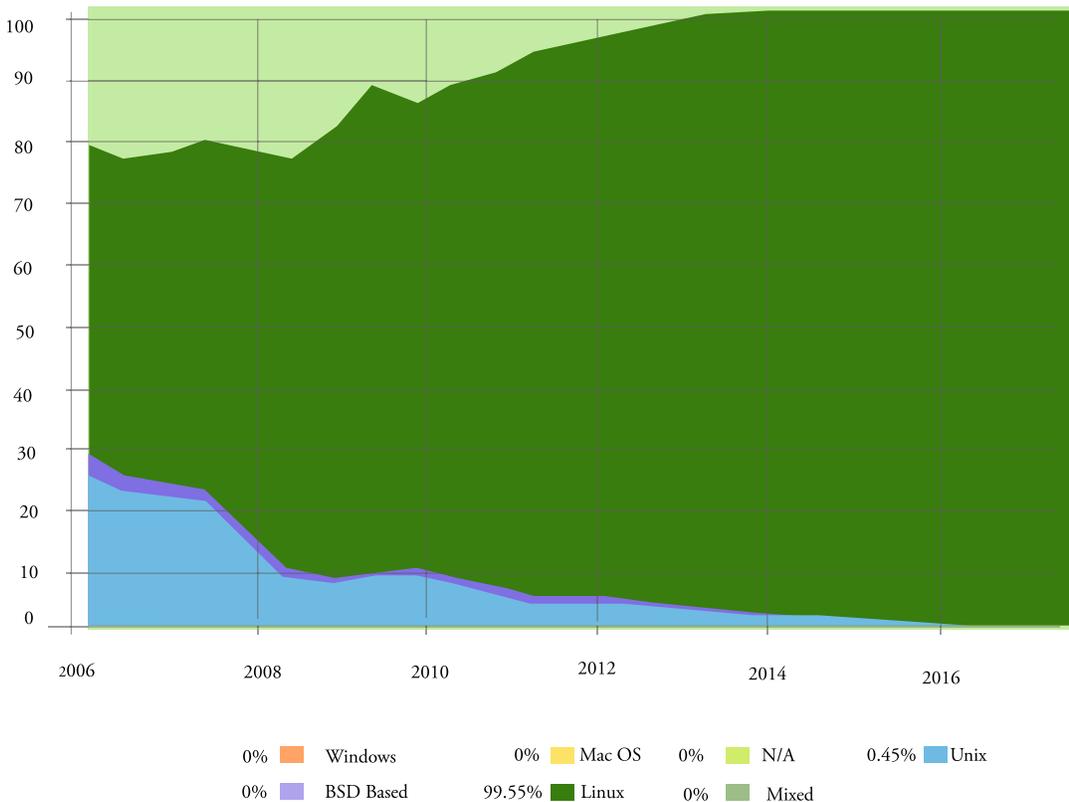
La participación de los usuarios promueve la capacidad de colectivos para generar propuestas tecnológicas viables y orientar la innovación hacia necesidades sociales reales, contando con la presencia de actores que habitualmente no son tenidos en cuenta en el proceso de desarrollo tecnológico (Toboso et al., 2015). La colaboración está siendo adoptada por las industrias del software privativo, pero sin retribuciones a los usuarios, puro utilitarismo. El software libre ha mantenido esta política desde sus inicios, significando un gran aporte para la educación.

10. La seguridad del alfabetizando importa

Las campañas de alfabetización digitales y el movimiento de Internet Seguro, también llamado Internet Sano, se enfocan en la seguridad de los ciudadanos desde su comportamiento en la Internet. Estas campañas intentan sensibilizar al usuario para que adopten medidas sobre una plataforma que pocos conocen. Se debería incluir en el programa de alfabetización temas sobre la seguridad en los softwares. Lo que se suele hacer es alfabetizar con un software privativo –que caduca a los 30 días y que luego el usuario tiene que crackear⁴ para habilitarlo. Disponer de software libre aprobados por comunidades reconocidas puede evitar muchas frustraciones futuras a los alfabetizados.

No existe seguridad absoluta en ninguna parte: es falacia garantizarla, en la informática recurrir a antivirus, *firewall* y *antispyware* son las medidas más convencionales. Sin embargo, las decisiones y comportamiento de los usuarios tienen mayor impacto que todas las herramientas de seguridad que se puedan instalar. La elección de software seguro para la alfabetización es el principal paso para la seguridad de los alfabetizados, aunque no el único.

Figura 4. Familia de sistemas operativos instalados en superordenadores (Top 500, Jun-2017)



Existe un gran consenso en que el sistema operativo GNU/Linux y sus derivaciones ofrecen mayor seguridad que los privativos de Microsoft Windows y OS X de Apple. Esto se puede corroborar

⁴ Este término *cracker* es una adopción del inglés que se usa para designar al individuo que logra quebrar la seguridad de un software o sistema informático. También suele llamársele así a pequeños *scripts* que eliminan las restricciones de tiempo.

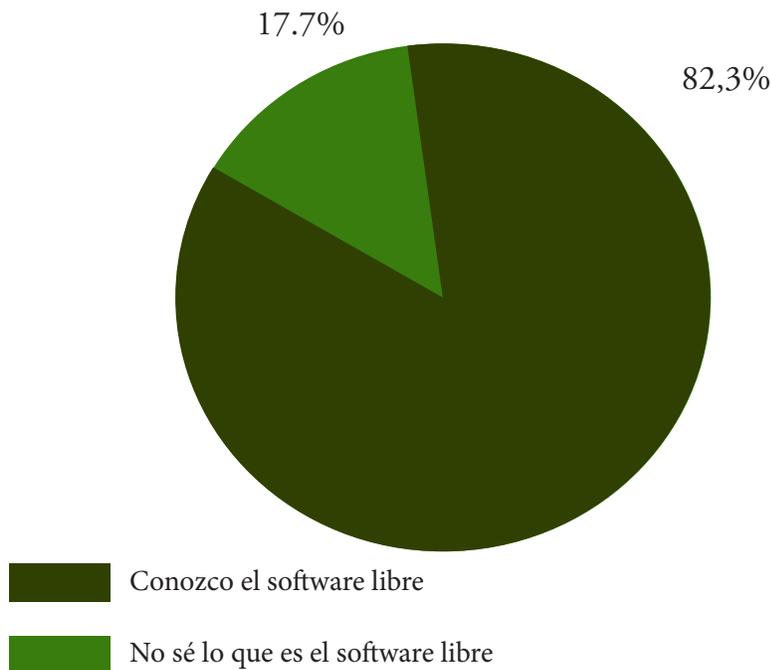
con los servidores que demandan más seguridad del mundo, un 99.55% están usando GNU/Linux y el 0.45% que resta usa UNIX, el cual dispone de versiones open source (figura 4).

Quizás sean inconsistentes estos datos para compararlos con el uso de los servidores locales y los ordenadores personales, pero nos ofrece una panorámica de la importancia que tiene elegir un software libre y abierto a la hora de elegir el sistema que administrará informaciones, y en especial, la de un centro escolar y todos los usuarios que se conectan inocentemente. La seguridad del software libre radica en la capacidad de identificar todos los puertos y administrarlos como se desee, algo que no pasa en los softwares privativos que traen puertas traseras (en inglés *back door*), y que dejan vulnerable a los usuarios aun después de haber pagado por el ellos

11. Alfabetización en y con software libre

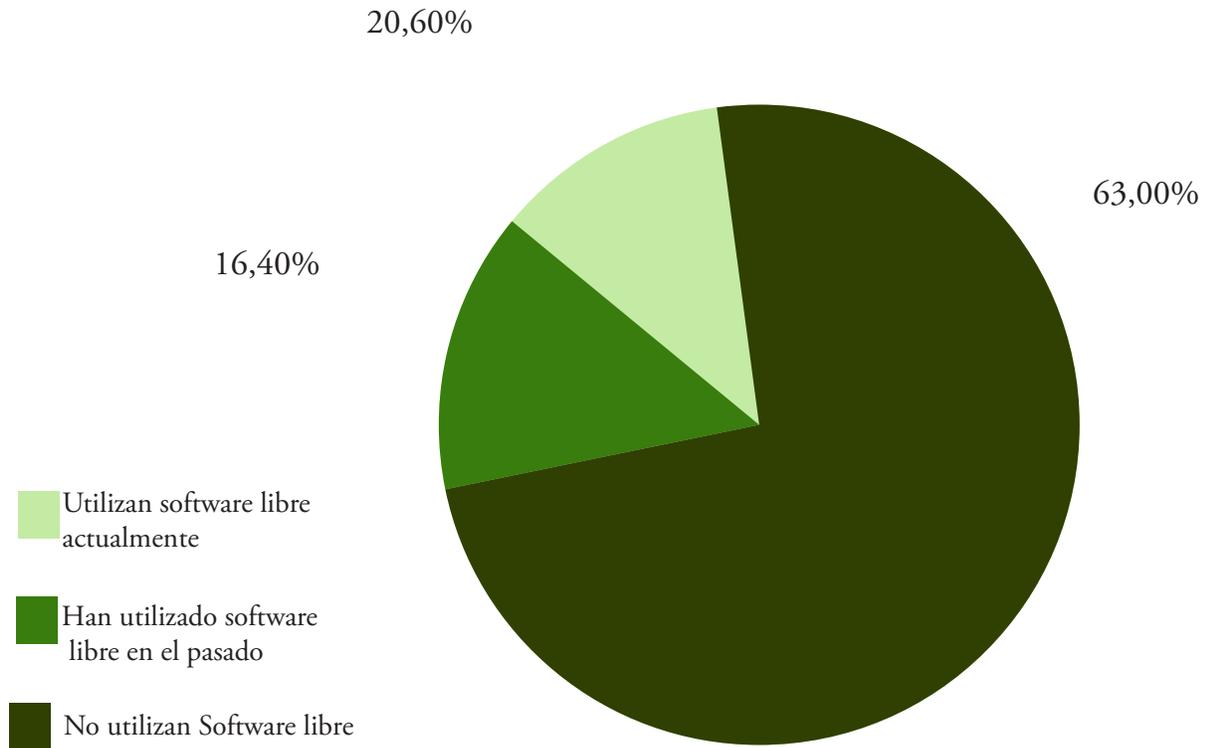
Los datos muestran que hay que alfabetizar en el conocimiento sobre el software libre para desmitificar el software privativo impuesto, en la mayoría de ocasiones, por las estrategias de *marketing* que hacen coaliciones con los fabricantes de hardware, y para que los usuarios creen que no existe nada más allá. El informe hecho por el Observatorio Nacional de Software de fuentes abiertas (ONSFA/CENATIC, 2011) nos revela esta necesidad (ver figura 5):

Figura 5: Conocimiento de los programas de software libre (ONSFA/CENATIC, 2011)

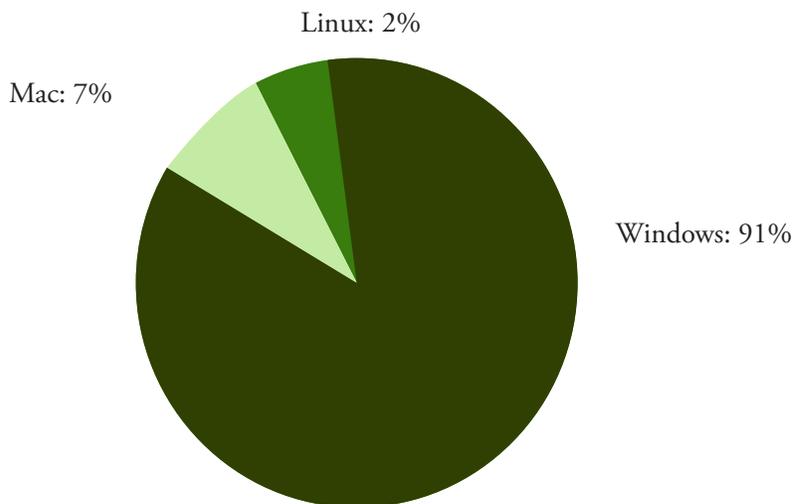


Es engañoso este número de un 82,3% de internautas que respondieron conocer el software libre, cuando se le preguntó si lo utilizan, solo un 20,60% afirmó utilizarlo y un 16,40% de la muestra dijo haberlo utilizado en el pasado. Ver figura 6:

Figura 6: Utilización de aplicaciones de software libre en España (ONSFA/CENATIC, 2011)



Considerando la definición de software libre y los diferentes tipos de licencias, podría cuestionarse ese conocimiento que dicen tener sobre el software libre, tal y como reveló otro estudio realizado en la comunidad de Cataluña, el cual mostró que los maestros y los alumnos no sabían distinguir conceptualmente entre software libre y no-libre a pesar de disponer una distribución de Linux, Llure (Da Costa & Escofet 2013). Otra cuestión es que España es uno de los países hispanoamericanos que más ha incidido en los proyectos de software libre para la educación y es referente de muchos países latinoamericanos. En la figura 7 se muestran los sistemas operativos más usados, donde Linux queda en tercer lugar con un 2%:

Figura 7: Uso de sistemas operativos a nivel mundial (NetMarketShare, 2017).

Entendiendo que Linux es el Sistema Operativo (SO) y las aplicaciones de softwares libres son diversas, incluyendo aquellas que se pueden ejecutar sobre SO privativo como Windows y OS X de Apple, se podría ampliar el margen de personas que conocen y usan software libre, pero aún no es suficiente. La cuestión sobre alfabetizar en software libre y con software libre radica en la construcción de una cultura que use tecnología libre. Un derecho que les corresponde a los ciudadanos es saber el tipo de tecnología que se le ofrece y las implicaciones que estas tienen en su vida presente y futura.

12. Aspectos Filosóficos de la alfabetización con software libre

La filosofía lleva tiempo mostrando gran interés en el desarrollo e impacto de la tecnología en la humanidad, la cuestión preocupante es que los filósofos tienen la oportunidad de reaccionar, en ocasiones, a posteriori, para reflexionar cuando ya las compañías han pactado con las instituciones gubernamentales para que las implementen.

Stallman (2004, p. 27) refutando la reflexión de Raymond (2003, p. 7), quien decía, “...todo buen trabajo de software empieza cuando un desarrollador se plantea un reto personal”, expone que, “Es posible que esté en lo cierto, pero muchos componentes esenciales del software GNU se desarrollaron con el fin de crear un sistema operativo libre y completo. Su origen está en una visión y un plan, no en un impulso individual”. Este punto de vista atina con los aspectos filosóficos que entraña el software libre, apuntando a una reflexión crítica previa a la producción de software que posteriormente afectará a la colectividad. Los aspectos filosóficos en la alfabetización con software libre responden a una “visión” y un “plan” que se preocupe por los alfabetizandos y no por otras entidades.

“Estamos aquí como peces en un acuario. Y sintiendo que hay otra vida a nuestro lado, vamos hasta la tumba sin darnos cuenta.” (Brandão, 2007, p. 27). Desde los antiguos griegos hasta nuestros tiempos la libertad, probablemente, es uno de los temas de los que más se ha escrito, esto revela que existe una necesidad ontológica por esta condición humana. La emancipación del ser humano, así como su esclavitud han fluido en una dialéctica interminable; aunque la esclavitud ha sufrido

cambios silenciosos que han hecho que los esclavizados, a veces, no sean capaces de percibir el estado idílico en el que se encuentran. Esto ha hecho que los abordajes sean distintos.

El software libre está fundado sobre la base de la libertad. A diferencia del software privativo, ofrece la posibilidad de adaptabilidad a las necesidades evitando penalidades.

A veces, el software libre es excluido de la alfabetización aludiendo a que es para expertos, sin embargo, se podría decir que, ¿los usuarios no son expertos o no se le permite serlos? La importancia de ser usuarios de software libre se basa en que a mayor cantidad de usuarios crean mayor posibilidad de mejoras, partiendo de aquellos usuarios que terminan siendo colaboradores (Raymond, 2003). Una pregunta ontológica, y quizás ingenua, es ¿les importa a los usuarios la libertad? Probablemente no tengamos una respuesta, pero la alfabetización debe dejar la jaula abierta, que el pajarillo quiera volar hacia el exterior o no, esa es su voluntad. Ahora bien, si se le cierra la jaula y él descubre que no puede volar al exterior, se le castiga.

El elemento de la libertad en el software libre posibilita el modelo de alfabetización que se requiere en este siglo XXI, superando los residuos de la educación tradicional que se entremezclan ralentizando el desarrollo sostenido en la sociedad. Una cita acertada de Paulo Freire (2005, pp. 60-61) que acentúa el aspecto liberador es la siguiente:

En este sentido, la educación liberadora, problematizadora, ya no puede ser el acto de depositar, de narrar, de transferir o de transmitir “conocimientos” y valores a los educandos, meros pacientes, como lo hace la educación “bancaria”, sino ser un acto cognoscente. Como situación gnoseológica, en la cual el objeto cognoscible, en vez de ser el término del acto cognoscente de un sujeto, es el mediatizador de sujetos cognoscentes –educador, por un lado; educandos, por otro–, la educación problematizadora antepone, desde luego, la exigencia de la superación de la contradicción educador-educandos. Sin esta no es posible la relación dialógica, indispensable a la cognoscibilidad de los sujetos cognoscentes, en torno del mismo objeto cognoscible.

El miedo a la libertad (Fromm, 1977) debe ser una cuestión individual no una coerción del sistema que educa. Proveer de software libre es detonar un mundo de posibilidades para el desarrollo cognitivo, es rescatar el “eureka” en los alumnos que aprenden y descubren nuevas formas de hacer y “recrear el mundo” (Freire, 2005, p. 131).

13. Más allá de lo gratis, se comparte

Stallman (2014, p. 12) dice, “Es una cuestión de libertad, no de precio. Para entender el concepto piense en libre como en “libre expresión”, no como en “barra libre”. Se suele decir, “en este mundo no hay nada gratis”, pero ¿por qué se dice? posiblemente cada persona la use de distinta forma, sin embargo, desde el software libre es claro que no es gratis, es una campaña deshonesto decirlo, es libre, aunque exima de pago alguno a los usuarios. Centrar el discurso del software libre en lo “gratis”, en vez de sumarle, le resta.

El software libre se “comparte libremente”. Este concepto evoca un sentido de comunidad que recibe y “reparte”, que no posee para sí, aquello que generosamente ha recibido. Es un valor alfabetizador que agrega esperanza en la construcción de un futuro tecnológicamente humanizado.

La dirección del discurso debe girar en torno a estos dos criterios:

- El software libre se comparte. Más allá de promover que es gratis hay que hablar de “compartir”, compartir libremente. Este concepto evoca un sentido de comunidad que recibe y reparte, que

no posee para sí, aquello que con gratuidad ha recibido. Rompe con el término “transferencia de conocimiento”, el conocimiento se comparte y se enriquece en la comunidad para retornar a sus orígenes mejorados.

- El software libre es colaborativo. El objetivo del software libre es que los usuarios sean colaboradores, lo cual se puede hacer de múltiples maneras: reportando los errores del software para aumentar su seguridad, realizando mejoras si eres desarrollador, haciendo donaciones a las fundaciones y los desarrolladores por sus aportes, descargando los softwares para uso personal y distribuyéndolos, etc.

Conclusión

Para integrar el software libre en la alfabetización, la mediación debe ser una colaboración, no se contempla mediación que otorgue poder sinuoso a ningún particular o institución en el software libre. Este es otro de los temores de muchos sectores, involucrase en algo que no le agregue algún tipo de poder. Esta actitud puede responder a la pregunta ¿por qué si el software libre es bueno las instituciones y estados no lo usan? Los Estados como mediadores tienen el potencial de promover las iniciativas del software libre, y en la alfabetización, sería el mejor escenario para construir una cultura sostenible. La imparcialidad de los Estados en este aspecto representa una amenaza a las iniciativas locales y el fomento internacional del desarrollo de software libre (González Barahona, 2002).

Hace falta apropiación por parte de los gobiernos, las grandes compañías le están sacando partido al software libre, sin embargo, la apatía de los gobiernos está excluyendo a los ciudadanos de menos recursos, al acceso a tecnología libre y alienándolos a la privativa, estas acciones violan el deber del Estado y perjudica a los ciudadanos encegueciéndolos y derrochando los recursos del erario público, como diría González Barahona (2002) en su artículo “¿Que se hace con mi dinero?”.

Freire (2005) se manifiesta en este sentido:

Ahora mismo ya nadie educa a nadie, así como tampoco nadie se educa a sí, los hombres se educan en comunión, y el mundo es el mediador. Mediadores son los objetos cognoscibles que, en la práctica “bancaria”, pertenecen al educador, quien los describe o los deposita en los pasivos educandos”. El software libre ha venido creciendo gracias a esta capacidad de las personas a actuar en comunidad. “La teoría materialista de que los hombres son producto de las circunstancias y de la educación, y de que, por lo tanto, los hombres modificados son producto de circunstancias distintas y de una educación distinta, olvida que las circunstancias se hacen cambiar, precisamente por los hombres, y que el propio educador necesita ser educado. (Marx & Engels, 1888).

El camino recorrido desde la alfabetización tradicional hasta los enfoques filosóficos y pedagógicos en la alfabetización con software libre han brindado una visión teórica general sobre las implicaciones que conlleva integrar software en la educación, sin tomar en cuenta ciertos parámetros determinantes para el futuro del alfabetizando y la cultura en general.

Algunos temas tratados en este artículo necesitan ser ampliados o desarrollados de manera más profunda a través de investigaciones, por el aporte que pueden brindar al pensamiento pedagógico filosófico contemporáneo.

Finalmente, vale recalcar que el discurso de que, “hay que humanizar la tecnología” carece de sentido lógico, porque no es necesario humanizar algo que producen los humanos. Si hay algo que amerita humanización es el mismo humano que se ha descarriado de su compromiso o deber consigo mismo. La tecnología del software libre es intrínsecamente humana, integrarla en la alfabetización

es aportar a la sostenibilidad. Los valores que la definen son necesarios para la construcción de una sociedad crítica, libre y equitativa. Es abrir las posibilidades y el acceso a aquellos que están del otro lado de la balanza, para que participen colectivamente en la utopía de que “un mundo mejor es posible, y con tecnología”.

Referencias

- Ashton, K. (2009). That «Internet of Things» Thing. *RFID Journal*. Recuperado de <http://www.rfidjournal.com/articles/view?4986>
- Barrón Ruiz, Á. (1989). Presupuestos para un programa racional de formación del enseñante, *Borrón*, 4(4), 681-690.
- Brandão, R. (2007). *Húmus*. Madrid: Biblioteca ERL Ediciones.
- Braslavsky, B. (2003). Qué se entiende por alfabetización. *Lectura y Vida. Revista Latinoamericana de Lectura*, 24(2), 2-17. Recuperado http://www.lecturayvida.fahce.unlp.edu.ar/numeros/a24n2/24_02_Braslavsky.pdf
- Cambridge Dictionary* (2017). Recuperado de <http://dictionary.cambridge.org/es/diccionario/ingles/literacy>
- Da Costa Silva, F. de A., & Escofet, A. (2013). Un estudio de caso sobre el uso del software libre en la enseñanza secundaria en Cataluña. *Campo Abierto. Revista de Educación*, 32(2), 71-95.
- Freire, P. (2005). *Pedagogía del oprimido (3ra ed.)*. D.F.: Siglo Veintiuno Editores.
- Fromm, E. (1977). *El miedo a la libertad*. Buenos Aires: Editorial Abril.
- Gallego Arrufat, M. J. (1994). *El Ordenador, el currículum y la evaluación del software educativo*. Granada: Proyecto Sur.
- Gilster (1997). *Digital literacy*. New York: Wiley Computer Pub.
- González Barahona, J. (2002). ¿Qué se hace con mi dinero? *Revista Todo Linux*, 17(12-13). Recuperado de <https://bit.ly/2LEaaTV>
- Gutiérrez, A. (2003). *Alfabetización digital: algo más que ratones y teclas*. Recuperado de <https://bit.ly/2ISuuPz>
- Infante, M. I., Letelier, M. E. (2013). *Alfabetización y Educación: Lecciones desde la práctica innovadora en América Latina y el Caribe. Red Innovemos de OREALC/UNESCO*. Recuperado de <http://unesdoc.unesco.org/images/0021/002191/219157s.pdf>
- Kuhn, T. S. (2006). *La estructura de las revoluciones científicas (Vol. 213)*. Ciudad: Fondo de Cultura Económica.
- Leonhardt, D. (2000, 28 julio). *John Tukey, 85, Statistician; Coined the Word «Software»*. *The New York Times*. Recuperado de <https://nyti.ms/2MONfc0>
- Marx, K., & Engels, F. (1888). *Versiones de las Tesis sobre Feuerbach de Carlos Marx: Tesis sobre Feuerbach* (1888). Recuperado a partir de <https://bit.ly/1pswbID>
- Moreno, G. (2017). Fabricados con fecha de caducidad. El Portal de las Estadísticas. Recuperado <https://bit.ly/2seNaDb>
- Mzahm, A. M., Ahmad, M. S., Tang, A. Y. C., & Ahmad, A. (2015). Software analysis for Agents of Things (AoT) applications. En 2015 International Symposium on Agents, Multi-Agent Systems and Robotics (ISAMSR) (pp. 12-17). Recuperado de <https://doi.org/10.1109/ISAMSR.2015.7379123>

- Naciones Unidas. (2017). *Objetivos de Desarrollo Sostenible*. Recuperado de <https://bit.ly/2MttOGi>
- NetMarketShare (2017). Operating system market share. Recuperado 30 de junio de 2017, a partir de <https://bit.ly/2BJOS6z>
- ONSFA/CENATIC. (2011). *Estudio sobre el Software Libre en los Hogares Españoles, Resultados de la Encuesta sobre el uso de Software Libre en los Hogares 2011*. España. Recuperado de <https://bit.ly/2xkdhgD>
- Oxford English Dictionary* (2017). Recuperado a partir de <http://www.oed.com/view/Entry/109054>
- Philipson, G. (2004). *A short history of software*. Recuperado 31 de mayo de 2017, a partir de <http://www.thecorememory.com/SHOS.pdf>
- Real Academia de la Lengua (2017). *Alfabetizar*. Recuperado de <https://bit.ly/2NcwXa0>
- Raymond, E. (2003). *La catedral y el bazar, versión en castellano*. Buenos Aires: OPENBIZ. Recuperado de <https://bit.ly/2seNxO5>
- Software (2017). *ESPASA* [Versión electrónica]. Barcelona: Editorial Planeta. Recuperado de <http://espasa.planetasaber.com/encyclopedia/default.asp?idreg=8709&ruta=Buscador>
- Stallman, R. (2014). El software libre es una cuestión de libertad, no de precio. *UC3M, Repositorio institucional e-Archivo*, 42(3). Recuperado de <https://bit.ly/2Ln2vrS>
- Stallman, R. (2004). *Software libre para una sociedad libre*. Madrid: Traficante de sueños. Recuperado de <https://bit.ly/2IQucwS>
- Toboso, M., Feltrero, R., & Maltrás, B. (2015). Proyecto Heliox: Entornos de interacción para la diversidad funcional / Heliox Project: Technological Environments for Functional Diversity. *Revista Internacional de Ciencias Humanas*, 4(2). Recuperado de <https://bit.ly/2LDx1yT>
- UNESCO. (2017). *Formación en capacitación en información y medios de comunicación* | Organización de las Naciones Unidas para la Educación, la Ciencia y la Cultura. Recuperado de <https://bit.ly/2wajQQ9>