



*Aseorías y Tutorías para la Investigación Científica en la Educación Puig-Salabarría S.C.
José María Pino Suárez 400-2 esq a Lerdo de Tejada, Toluca, Estado de México. 7223898475*

RFC: ATI120618V12

Revista Dilemas Contemporáneos: Educación, Política y Valores.

<http://www.dilemascontemporaneoseduccionpoliticayvalores.com/>

Año: VI. Número: Edición Especial. Artículo no.: 29. Período: Julio, 2018.

TÍTULO: Educación con Tecnologías Libres para fomentar la Innovación.

AUTOR:

1. Máster. Henry Fernando Vallejo Ballesteros.

RESUMEN: Los paradigmas educativos post-constructivistas han provocado nuevas corrientes metodológicas en el proceso bidireccional de la educación; donde el docente no solo debe dominar la ciencia de su formación académica; sino que debe conocer y dominar las tendencias paradigmáticas del nuevo siglo; que basa como instrumento pedagógico; en las tecnologías de la información, comunicación y automatización (TICAs). El conectivismo, los instrumentos tecnológicos, han facilitado el acceso a grandes volúmenes de información, los que son severamente restringidos; sin embargo; las corrientes de divulgación científicas abiertas, orientados al Software han dado origen a una generación más informada; reto ante el cual el docente tiene una tarea de estar en igualdad de condiciones, respecto a las nuevas generaciones tecnologizadas.

PALABRAS CLAVES: Educación, Tecnología, Software, Descolonización, Metodologías

TITLE: Education with Free Technologies to Encourage Innovation.

AUTHOR:

1. Máster. Henry Fernando Vallejo Ballesteros.

ABSTRACT: The post-constructivist educational paradigms have provoked new methodological currents in the bidirectional process of education, where the teacher must not only master the science of their academic education, but he must know and master the paradigmatic tendencies of the new century; which he bases as a pedagogical instrument in information, communication and automation technologies (ICATs). Connectivism, technological instruments have facilitated access to large volumes of information, which are variously restricted. Nevertheless, the currents of open scientific work, oriented to software, have given rise to a more informed generation; the answer to which the teacher has a task of being in equal conditions, with respect to the new technological generations.

KEY WORDS: education, technology, software, decolonization, methodologies.

INTRODUCCIÓN.

La conducta humana ha sido y es sujeto de estudio desde siempre lo que ha coadyuvado a la Ciencia de la Psicología a grandes debates que parten de lo fisiológico a la psicológico y un devenir entre lo uno y lo otro de manera constante; intentando desde la Filosofía o la Psicología comprender el comportamiento humano; su evolución y cómo ha sido posible ésta con el devenir del tiempo. Así que tratando de explicar la forma en que el hombre ha aprendido y aprende en el ámbito de lo formal; se ha determinado que ha seguido ciertos modelos o paradigmas que de un modo u otro también han modelado su propia conducta.

La Educación sin embargo antecede a todo concepto moderno; pues parte inicialmente de la necesidad en la que un adulto entrenaba a los jóvenes en el conocimiento y habilidades que su sociedad necesitaba. En las sociedades pre-alfabéticas esto se lograba por la vía oral o narración, y claro imitando; de ahí que los estudios antropológicos sugieren que, mediante la narración de historias, cuentos, reseñas; permitieron el paso del conocimiento de generación en generación.

DESARROLLO.

Paradigmas de la Educación.

Mientras esto ocurría en el primer milenio de la humanidad; ya para el siglo XVIII, Wundt (1907) crea el primer laboratorio científico de psicología en el cual se intenta analizar desde un punto de vista científico la conducta humana; en el siglo XX, Watson declara la conducta observable como el objeto de estudio de la psicología; en este punto surgen estudios que intentaron comprender la conducta humana y científicos como Skinner (1990), Pávlov (1928); entre otros; donde mediante el estímulo - respuesta se intenta cambiar o modificar la conducta observable, fruto de los distintos condicionamientos, dando origen al Paradigma Conductista; lamentablemente, la interacción entre el que aprende y enseña no es deseable ya que es netamente mecánica, deshumanizante y reduccionista; la evaluación por otro lado se limita pruebas netamente objetivas.

Por otra parte, (Maslow 2001), psicólogo estadounidense, propone una nueva corriente psicológica centrada en el ser humano donde la persona es cociente de sí mismo y de su propia existencia por lo que debe desarrollarse de manera íntegra; la educación entonces debería ser para todos en donde se debe educar para el trabajo, dando prioridad a las necesidades humanas, con principio de libertad, pero fundamentalmente con experiencias de aprendizajes vivenciales, lo cual nos llevó al nuevo paradigma denominado HUMANISTA en el que, quien enseña es accesible, flexible y claro; ayuda a desarrollar las potencialidades del que aprende, en esta instancia, el que aprende tiene un *aprendizaje significativo* y puede resolver problemas de manera creativa; es más se evalúa a sí mismo.

Para los años cincuenta surge una nueva corriente psicológica denominada Cognitivismismo como una reacción opuesta al conductismo, donde se considera al sujeto como un ser activo que posee conocimientos y que además tiene la capacidad de aprender independientemente de su contexto; sus principales defensores son Dewey (1964), Piaget (1952), Bruner (1997), Ausubel (1978), entre otros, donde sus principales aportes son que mediante el descubrimiento se pueden obtener

aprendizajes significativos que contribuyen a la modificación de esquemas por medio de la construcción de ambientes de aprendizaje.

Por otra parte, el docente enseña de manera afectiva mediante la elaboración de actividades que potencia las habilidades del alumno; por lo tanto, el alumno es un ser activo que desarrolla sus propias potencialidades según su edad. Se evalúan las actitudes, procedimientos, etc. que es capaz de realizar.

Posteriormente, Vygotski (1962) y Bruner (1997) plantean un nuevo paradigma denominado Sociocultural, como una alternativa en las que el Sujeto y Objeto del conocimiento mantienen una relación recíproca que da como resultado la reconstrucción del mundo socio cultural y de la construcción de funciones psicológicas y de conciencia. Los aportes de este paradigma son las actividades durante y después de la asistencia al aula, la aparición de foros culturales y reconstrucción del contenido curricular en función de los cambios que se presenten.

El maestro en este caso estimula a que el alumno se apropie del conocimiento; por otra parte, el alumno puede reconstruir los saberes en conjunto con otros. La evaluación se vuelve dinámica para diagnosticar el nivel de aprendizaje.

Vygotski (1962) plantea, poco después, una nueva corriente a la que denominaron Constructivismo con dos vertientes: la Psicológica y la Social de tal forma que el alumno construye su propio conocimiento partiendo de sus experiencias dentro de su propio contexto; se plantea el uso de nuevos instrumentos, la relación en base al trabajo, la palabra como forma de generar conocimiento, lo que da origen a una nueva escuela.

El docente entonces propicia la enseñanza a partir de la experiencia y el alumno por su parte es el sujeto activo que construye su propio conocimiento; de tal manera que la evaluación se basa en las interpretaciones que el alumno hace respecto de lo que aprende.

En circunstancias, las TICs se vuelven en una herramienta potenciadora del modelo en los actuales momentos en los que la posibilidad de usar la web, la virtualización, las bibliotecas virtuales permiten experimentar, debatir y proponer nuevas ideas que generan conocimientos nuevos en el

alumno; descubriendo en el proceso nuevos conceptos que a su vez le permiten construir su propio conocimiento; sin que ello tenga un costo que limite el apropiarse de ese conocimiento.

Ante las limitaciones del constructivismo, en pleno desarrollo de la era digital tiene origen un nuevo paradigma denominado Conectivismo, planteado por George Siemens y por Stephen Downes, donde se analiza el efecto que la tecnología ha tenido sobre la forma en cómo nos comunicamos y como permanecemos conectados; por lo tanto la forma en cómo aprendemos en la era digital; tanto es así que el conectivismo es la combinación del constructivismo y el cognitivismo a lo que se ha dado en llamar la “teoría del aprendizaje digital”

Paradigmas de la Programación.

Así como la Ciencias Sociales estudió el comportamiento humano ante los procesos educativos las Ciencias de la Computación también plantearon formas distintas de pensar y por ende de analizar y programar sistemas computacionales o informáticos a fin de dar solución a problemas humanos reales.

Determinemos que es la Información: La información es un conjunto de datos que organizados, ordenados y relacionados entre sí tienen algún sentido ante la toma de decisiones; estos datos, sin embargo, por si solos carecen de sentido a menos de que se organicen adecuadamente para dar origen a la información.

Ahora bien, la Informática surge como una ciencia que estudia los métodos, técnicas y procesos con el fin de almacenar, procesar y transmitir información en forma digital; por lo tanto, involucra a ciertos tipos de dispositivos computacionales denominados computadoras (ordenadores); es decir cualquier dispositivo electrónica que sea capaz de procesar, almacenar y transmitir datos y con cierta capacidad de computo.

Para poder conseguir estos resultados las computadoras necesitan de un conjunto de programas capaces de apoyar al dispositivo (hardware) para poder realizar estos cálculos. Estos programas para

nuestro estudio los podemos dividir en tres tipos: Sistemas Operativos, Lenguajes de Programación y Aplicaciones.

Los Sistemas Operativos son la base fundamental de todo sistema computacional ya que sobre el subyacen todo el software que usamos con propósitos distintos, el Sistema Operativo es ese conjunto de aplicaciones que nos permiten aprovechar los recursos del hardware.

Por otra parte, los lenguajes de programación son un conjunto de instrucciones basados en cierta sintaxis con las cuales podemos construir aplicaciones basados en una lógica determinada, cuyo resultado final serán nuestras aplicaciones.

Las Aplicaciones, por consiguiente, son el resultado de la programación basado en las necesidades del usuario a fin de resolver algún tipo de problema que deba ser automatizado para conseguir eficiencia, eficacia, velocidad y que además pueda ser almacenada de alguna manera.

En sus inicios, los lenguajes de programación estaban orientados fundamentalmente en la estrategia de dar instrucciones al computador; de una a la vez, considerando que las primera generación de computadoras tenían poca capacidad de almacenamiento, memoria y procesamiento; surge de este modo la programación denominada “secuencial” o imperativa; como una forma de pensar y razonar consecutiva de instrucciones que harían que la máquina ejecute una a una las instrucciones; por lo tanto, la forma de abordar los problemas se basaba en una secuencia de pasos.

En su siguiente etapa, con la aparición de la tercera y cuarta generación la técnica de escribir los programas cambio poco a poco desencadenando en estrategias más acorde a velocidad y capacidad de procesamiento, organizando de mejor manera la forma de escribir el código, lo que nos llevó al concepto de estructura, donde pedazos de código se escriben para que cumplan una función específica y son ejecutados cada vez que se lo necesita, esta técnica dio origen al paradigma de programación “estructurada” .

Con la cuarta generación de computadoras las capacidades de cómputos cada vez son más elevadas por lo tanto el paradigma de la programación cambiaría y se encaminaría a una forma más humana de razonamiento en el momento de construir las aplicaciones, creando para ello el concepto de

objetos los cuales incluyen variables, funciones; encapsulados en ellos, con nuevas propiedades como la herencia, polimorfismos, etc. ; por lo tanto se hace más sencillo representar a los elementos de la vida real como objetos representados por un cumulo de instrucciones que se ejecutan cada vez que se los invoca.

A partir de estos tres paradigmas de programación se han ido creando nuevos, que lo que buscan es que la tarea de construir programas sea más sencilla, más acorde a la forma en cómo pensamos y razonamos¹, como, por ejemplo:

- Programación dinámica; donde se procura romper en partes más pequeñas al problema para resolverlo.
- Programación dirigida por eventos. - donde la estructura como la programación están orientadas a la ejecución de los programas van dados por los sucesos que ocurren en el sistema.
- Programación declarativa. - donde se describe al problema declarando propiedades y reglas que deben cumplirse.
- Programación funcional. - basado en definiciones de predicados usado mucho en Python
- Programación Lógica. - basado en definiciones de relación lógica
- Programación con restricciones. - muy similar a la lógica, pero usando ecuaciones
- Programación multiparadigma. - en el cual se utilizan más de un paradigma
- Programación específicos del dominio. - se denomina así al paradigma desarrollado para resolver problemas específicos como por ejemplo en los lenguajes de bases de datos.

Abordando el problema.

La innovación entendida como la capacidad de poder mejorar o reinventar algo, inclusive crear algo a partir de algo, nos lleva a plantearnos como lograr esos objetivos; si la innovación se ve limitada por los derechos de autoría o patentes que limitan la inventiva, considerando que el respeto a ellas

1 (<https://docs.python.org/3.5/howto/functional.html?highlight=paradigm> 2018).

retrasa de alguna manera el desarrollo de nuevas tecnologías en países con poca o ninguna investigación.

Por otro lado, el escaso aporte del estado; en algunos casos también se ha convertido en un obstáculo, ya que no es suficiente ante la carencia de laboratorios especializados y o concentrados en las principales ciudades; no así en las pequeñas unidades de desarrollo e investigación universitarias del resto del país; de ahí que es importante buscar alternativas que nos permitan llegar con el conocimiento suficiente para aprovechar esos recursos.

Plantearnos desde aquí el trabajo colaborativo, evitando el capitalismo cognitivo y lo que ello significa para la innovación, deteniéndola de acuerdo a las conveniencias del capital y por lo contrario plantear una economía social del conocimiento; donde el mismo pueda ser divulgado y reutilizado por quienes lo necesiten en favor de buscar soluciones a problemas concretos, en una sociedad que demanda de tecnologías libres y que vayan en fusión del bien común.

En el campo de las Tecnologías y principalmente en el de la computación, durante años hemos sido alienados; por decirlo de alguna manera, y hasta obligados a usar software propietario y de una forma casi perversa se nos ha “permitido” copiar, distribuir software, protegido por patentes; lo que ha provocado que en las escuelas, colegios y universidades hayamos formado usuarios de software privativo; de igual forma ya en el campo de profesionalización no hemos dedicado a enseñar con herramientas de software privativo, lo que ha coadyuvado a la utilización de conocimientos creados por la industria del software privado, nos hemos dedicado a la formación de usuarios y profesionales que de manera gratuita los hemos formado para usar esas herramientas, provocando a la final que la industria local se vea también obligada a adquirir aplicaciones bajo estos mismos conceptos de privacidad, secretismo y altos costos.

Liberar el Pensamiento (descolonización).

Hablar de liberar el pensamiento como una actitud de rebeldía ante los paradigmas pedagógicos, así como de los paradigmas tradicionales de la programación, implican tomar decisiones de carácter

político, pero sobre todo una actitud de libre pensamiento, que modelen quizá nuevas tendencias que provoquen la innovación tecnológica, no solo en el aula; sino también como resultado de ello, tecnologías nuevas e innovadoras fruto de nuestras propias necesidades.

El software existente, así como el que pudiera crearse bajo principios liberadores del pensamiento deberían ser instrumentos que desaten una dinámica de creación propia, de inventos y porque no de soluciones concretas a problemas concretos sin restricciones; sino en busca del bien común.

Es posible que dejar de usar paradigmas de desarrollo que ha generado el capitalismo cognitivo, y usar software que promueva una economía social de conocimiento como un modelo de “negocio”; sea la alternativa para mejorar la economía del profesional (tecnólogo, licenciado, Ingeniero, programador de sistemas, etc.); e impulsar también la economía local y regional.

Es entonces fundamental aprender a desaprender, ya que ello implicaría una descolonización de tecnologías y empezar a crear las propias.

No podemos pensar que todo este hecho, siempre habrá nuevos requerimientos, la dinámica del mundo genera nuevas necesidades y estar atentos a ellas es una oportunidad para desarrollar e innovar tecnología.

De igual forma surge la necesidad imperiosa de que el docente universitario se sintonice con estas tendencias y utilizando metodologías, paradigmas y nuevas estrategias e innove también desde su cátedra, el entorno de aprendizaje (aula o laboratorio).

Pensamiento Lógico computacional.

Jeannette Wing (2006), publicó el artículo *Computational thinking* en el cual se abordaba la necesidad de que nuevas competencias debían ser incluidas en la formación de los niños; puesto que representa una herramienta vital en el aprendizaje de las ciencias, las matemática, la ingeniería y la tecnología.

Definamos que es el pensamiento computacional; tomando las palabras de la propia autora: “El pensamiento computacional implica resolver problemas, diseñar sistemas y comprender el comportamiento humano, haciendo uso de los conceptos fundamentales de la informática”².

Es decir que el pensamiento computacional es: pensar tal como lo haría un científico informático cuando se enfrenta a un problema y como abordarlo para llegar a su solución.

Desde entonces, se han creado otros conceptos que se encuentran disponibles en la literatura científica, por ejemplo:

✚ El pensamiento computacional es el proceso que permite formular problemas de forma que sus soluciones pueden ser representadas como secuencias de instrucciones y algoritmos.

✚ El pensamiento computacional es el proceso de reconocimiento de aspectos de la informática en el mundo que nos rodea, y aplicar herramientas y técnicas de la informática para comprender y razonar sobre los sistemas y procesos tanto naturales como artificiales³.

Con el fin de operativizar la resolución de problemas mediante el Pensamiento Computacional, la Sociedad Internacional de la Tecnología en la Educación (ISTE) y la Asociación de Profesores de Informática (CSTA) ha contribuido con las siguientes características a fin de poder entender mejor como abordar un problema desde esta óptica:

- Formular el problema de forma que se permita el uso de un computador y otras herramientas para ayudar a resolverlos.
- Organizar y analizar lógicamente la información.
- Representar la información a través de abstracciones como los modelos y las simulaciones.
- Automatizar soluciones haciendo uso del pensamiento algorítmico (como una serie de pasos ordenados para llegar a la solución).
- Identificar, analizar e implementar posibles soluciones con el objetivo de lograr la combinación más efectiva y eficiente de pasos y recursos.

² Wing, J. (2006). Computational thinking. *Communications of the ACM*, 49(3), p.33.

³ Alfred V. Aho, (2012) Computation and Computational Thinking. *he Computer Journal*, Volume 55, Issue 7, 1 July 2012, Pages 832–835.

- Generalizar y transferir este proceso de resolución de problemas para ser capaz de resolver una gran variedad de problemas similares.

La idea entonces es: promover el desarrollo del pensamiento computacional usando la gamificación mediante el uso de computadoras de escritorio, laptops y celulares a edad temprana hasta su formación profesional; siendo también posible aplicar esta metodología a cualquier edad y en cualquier asignatura o ciencia (disciplina) haciendo uso de cualquier recurso y no solo desde la programación.

Tratando de explicar de mejor manera y un poco en detalle, buscando la manera en como el pensamiento computacional puede interconectar con otras ciencias por ejemplo la Psicología, tenemos la figura 1, que puede explicar esa correlación.



Figura 1. Software Libre y Privativo.

En un sistema informático moderno el software se complementa directamente con el hardware desde el punto de vista del procesamiento y de la cantidad de bits con los que es capaz de procesar la información, puesto que va a depender directamente del tipo de microprocesador, del bus interno y externo; pues dependiendo de la capacidad de direccionamiento, el procesador será capaz de transferir mayor cantidad de datos entre los dispositivos internos y externos.

El Software cumple un rol fundamental ya que, si partimos el análisis; desde el punto de vista del Sistema Operativo, este debe ir en concordancia con estas capacidades.

En sus inicios el software tenía un modelo de desarrollo cooperativo (1945 – 1955) de manera similar a otras ciencias como la física, la química, etc.; desafortunadamente esto empezó a cambiar en las décadas de 1960 a 1970 momento en el cual nacieron las primeras empresas que decidieron “privatizar” su código.

Aunque los principios de código abierto o fuente abierta existen desde mucho antes de la existencia de las computadoras, tal es así que en los primeros años del desarrollo de la industria automotriz la “Asociación de Fabricantes de Automóviles Autorizados” eran dueños de la patente de un motor de dos tiempos; así que quienes decidían fabricar autos tenían que pagar forzosamente por dichas patentes; lo que provocó que en 1905 que Henry Ford y un grupo, formaran una nueva organización que les permitiera defender la libre fabricación de vehículos a motor; con una lucha que iba en contra de las primeras patentes ya que los motores que se fabricaban en 1911 usaban otra tecnología, lo que finalmente permitió la creación de una nueva licencia múltiple para los fabricantes de automotores.

En la década de los 70, en el MIT, la mayoría de estudiantes de Ciencias de la Computación ya compartían su código, inclusive mucho antes de que el movimiento de Software Libre existiera; por lo tanto, el Software nació libre y así debía permanecer. A decir de Richard Stallman, su principal defensor; el Software no puede estar atado a patentes o licencias que limiten su uso y/o peor aún su desarrollo.

Ya en los años 60, el software no era visto como un “producto” sino más bien como el trabajo colaborativo entre académicos e investigadores que buscaban el bien común.

Richard Stallman (1983) lanza el proyecto GNU (*GNU is Not Unix*) con la idea de escribir un Sistema Operativo completo y libre de restricciones; para ello plantea cuatro libertades que claramente definen al Software Libre⁴:

- Libertad 0 - Ejecutar el programa con cualquier propósito (privado, educativo, público, comercial, militar, etc.).

4 Stallman, R. n.d. (1983), A Quick Guide to GPLv3. Free Software Foundation. Retrieved from: <http://www.gnu.org/licenses/quick-guide-gplv3>

- Libertad 1 - Estudiar y modificar el programa (para lo cual es necesario poder acceder al código fuente).
- Libertad 2 - Distribuir el programa de manera que se pueda ayudar al prójimo.
- Libertad 3 - Distribuir las versiones modificadas propias (para lo cual es necesario poder acceder al código fuente).

Con estas premisas, es fundamental tener acceso al código manteniendo la libertad de poder modificar y redistribuir el código bajo algún tipo de licencia que respete estas libertades y que además vaya en beneficio de la comunidad.

Esto lo que facilita es la posibilidad de desarrollar nuestra propia tecnología y como evidentemente es posible modificar el código; reconstruirlo, etc. Por lo tanto, evoca a la Innovación de nuestras aplicaciones informáticas desde un punto de vista cooperativo y colaborativo.

Ya en este contexto entonces; vale la pena proponer actividades en concreto, para fomentar una educación que provoque cambios significativos basado en los valores, la ética, la ciencia y la tecnología; de cómo la creamos y la utilizamos en nuestro beneficio.

La Innovación desde el aula.

El aula no es solo el entorno físico en el cual se realizan las reuniones académicas; sino que además es el espacio en el cual se debaten ideas, se proponen teorías; se despejan inquietudes. Es allí donde el docente enfrenta un reto ante sus alumnos pues debe innovarse día a día tanto en la metodología y el buen uso de los instrumentos pedagógicos disponibles. Sin embargo, en este siglo lleno de avances tecnológicos, que a irrumpido de manera abrupta en los métodos tradicionales de una pedagogía incipiente; es fundamental que el mediador del conocimiento se actualice y vaya acorde con las necesidades académicas actuales, aplicando los paradigmas del Conectivismo, mediante el uso de Entornos Virtuales de Aprendizaje (EVA) y aprovechando al máximo todos los recursos disponibles (Redes Sociales, Chats, Videoconferencias, Blogs, etc.).

El docente en general debe saber que la sola existencia de un proyector de datos, una laptop o un conjunto de computadoras no es suficiente para poder aplicar estos nuevos paradigmas, va más allá del simple uso de estos instrumentos; es necesario involucrar al alumno en estrategias de Pensamiento Computacional, usando la lógica para resolver problemas.

De la misma forma, el docente de ciencias de la computación, informática o programación; al ser el facilitador de un conocimiento especializado, debe estar consciente que su tarea va más allá del uso de los instrumentos antes indicados; tiene que ver con provocar en el alumno un grado de abstracción que le permita imaginar posibles salidas a un problema y llevarlos a la solución mediante análisis complejos, para poder ser resueltos en el campo de la programación.

Entonces, si buscamos que el alumno comprenda, cree, desarrolle aplicaciones con un alto grado de innovación y que además sea capaz de construir su propio conocimiento (constructivismo), discutirlo, mejorarlos y compartirlo con los demás (conectivismo); la única manera de lograrlo es teniendo acceso al código y observando cómo se solucionaron problemas similares.

Así mismos, valores como la solidaridad, la ética; se verán altamente reforzados ante una estrategia de trabajo en equipo, que denota un alto grado de cooperación intelectual entre sus pares.

Crear nueva tecnología a partir de la ya existente sin restricción alguna, es una manera muy eficiente de motivar en el alumno el desarrollo de habilidades en el campo de la investigación científica, como una forma de sustentar nuevas teorías y paradigmas de la programación.

La programación desde un enfoque innovador.

La innovación definida como el acto de introducir novedades, modificando elementos ya existentes con el propósito de mejorarlos o renovarlos tiene una profunda connotación en las Nuevas Tecnologías de la Información y Comunicación Automatizadas; ya que a diario se crean nuevas y mejores herramientas computacionales que nos ayudan a resolver problemas de la vida diaria, con un enorme impacto en la sociedad moderna.

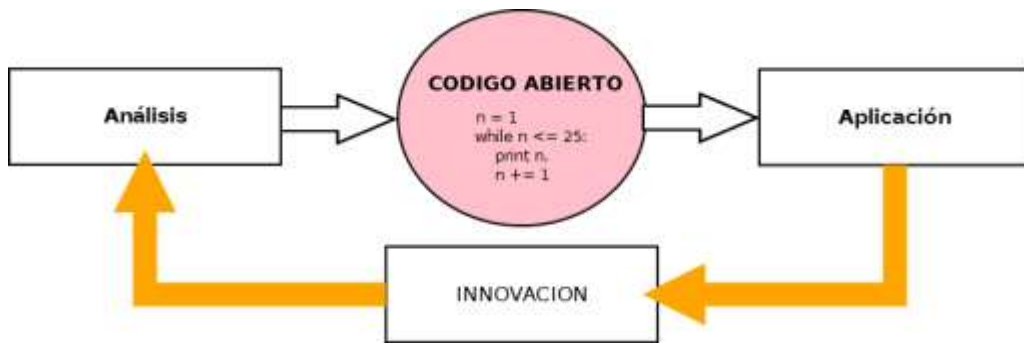
Las ciencias de la computación siendo un pilar fundamental de tales innovaciones, requiere también renovarse y plantear nuevos paradigmas en cuanto al modelo de negocios; tomando en cuenta lo que se ha expuesto con respecto de la independencia tecnológica, la descolonización, el capitalismo cognitivo, que de una u otra forma frenan la propuesta innovadora de una sociedad más justa desde el punto de vista de las ideas y del consumo de información científica que beneficie el desarrollo.

La programación de Software como mecanismo para llegar a la creación de sistemas computacionales no puede estar ligada a simples conceptos de “cajas negras” o arquitecturas de desarrollo cerradas. Si bien la Ingeniería de Software es “el establecimiento y uso de principios fundamentales de la ingeniería con objeto de desarrollar en forma económica software que sea confiable y que trabaje con eficiencia en máquinas reales” *Fritz Bauer [Nau69]*; no dice mucho sobre el hecho de que la ingeniería del software nos entrega los lineamientos metodológicos para crear aplicaciones informáticas dentro de los contextos de calidad, mediante herramientas, métodos y procesos⁵ que buscan un fin.

La programación entonces debe basarse en los principios fundamentales de la ingeniería, más sin embargo los lenguajes utilizados, los compiladores, editores, y el sistema operativo sobre el cual se ejecutan; no deberían restringir su ejecución y uso, así como tampoco deberían tener un costo tecnológico para el usuario; más aún si este último es capaz de responder dentro de los parámetros de confiabilidad y eficiencia.

Un modelo que aporte a desarrollar las capacidades intelectuales, aplicando niveles de abstracción para resolver los distintos problemas, basado en el conocimiento de los lenguajes de programación, con sus respectivos símbolos, reglas, sintaxis y semántica, nos permitirá estudiar el código, para mejorarlo; es decir para innovarlo.

⁵ Pressman, R. (2010). Ingeniería del software. México: McGraw-Hill. 7ma ed. pp 12.



El bien Común.

Las tecnologías Libres tanto hardware como software; pretenden abordar las necesidades humanas de estar comunicados mediante el uso de las TICAs, con este propósito existen herramientas hardware siendo la más destacada las placas Arduino y Raspberry Pi con una amplia información disponible en la red para aprovecharlas en cualquier desarrollo imaginable. Por ejemplo, la Raspberry Pi al ser un computador pequeño de bajo costo ha permitido llegar a comunidades muy pobres donde la tecnología parecía imposible, permitiendo además reducir la brecha tecnológica y digital.

De igual forma, el Software; especialmente aquel que tiene que ver con la operación del hardware, con una variedad de distribuciones en el campo de los Sistemas Operativos, Linux ha sido sin duda quien merece liderar el sector tanto a nivel de escritorio como de servidores; en ese campo Apache es uno de los más utilizados, como puede verse en la figura 3. En la que el 31% de los servicios web funcionan con esta herramienta libre.

Statistics for websites using Web Server technologies

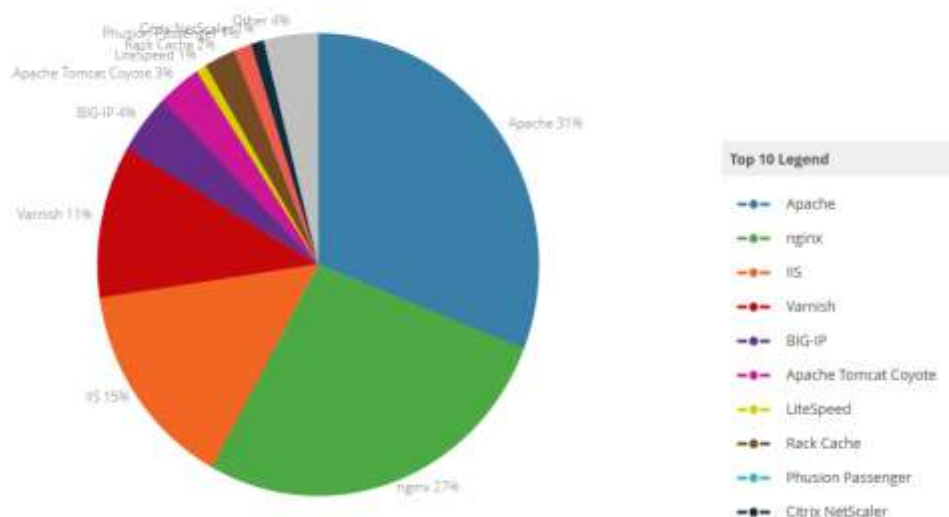


Figura 3. Fuente: <https://trends.builtwith.com/web-server>

No obstante, el software libre también debe hacer frente a numerosos retos, más allá de las limitaciones de partida que impone el capitalismo cognitivo, como: mejorar la accesibilidad para los usuarios medios, o la distancia entre actualizaciones y, sobre todo; ofrecer soluciones en cantidad y calidad, ajustadas a las necesidades sociales para hacer efectiva esa transición hacia la economía social del conocimiento común y abierto⁶.

Lo deseable es que al hacer uso de tecnologías libres (software y hardware) decante en el bien común, para disminuir la brecha tecnológica, pero sobre todo apropiarse de una tecnología que nace de la investigación y culmina con la innovación.

El uso del Software Libre en la Educación Superior.

En el Ecuador desde hace mucho tiempo un conglomerado de activistas que defienden el uso del software libre y otras tecnologías similares han venido impulsando el uso de nuevas tecnologías con software libre; desde una perspectiva más humana; en la que la independencia tecnológica o descolonización nos permitan en primera instancia; disminuir la enorme brecha tecnológica, así como la democratización de las mismas. En tal virtud se han desplegado un sinnúmero de eventos masivos de divulgación de dichas tecnologías en escuelas, colegios, universidades, entidades

⁶ Software Programas libres y de código abierto en la Administración Pública Buen Conocer - FLOK Society, IAEN/CIESPAL, pp 658.

públicas y privadas, con el propósito de dar a conocer las ventajas y desventajas de usar software y hardware libres y abiertos.

Desde el año 2008, las Asociación de Software Libre del Ecuador (ASLE); entidad sin fines de lucro, ha impulsado mediante diálogos; la posibilidad de que sea el estado ecuatoriano quien asuma la responsabilidad de llevar a todas las entidades públicas a usar software libre, con el firme propósito de ahorrarle recursos al estado y de mejorar la mayoría de los servicios que el mismo brinda; Ahorrándole al estado cerca 2.7 millones de dólares en licencias de software por año.

En abril del 2008 se emitió el decreto 1014 que en cumplimiento de la Carta Iberoamericana de Gobierno Electrónico aprobada por la “IX Conferencia Iberoamericana de Ministros de Administración Pública y Reforma del Estado “, que recomienda el uso de estándares abiertos y software libre como herramientas informáticas. Con los mismos principios de las cuatro libertades del Software libre.

De igual forma la Ley de Educación Superior del 12 de octubre del 2010; aún vigente, en su artículo 32 segundo párrafo dice:

“Las instituciones de educación superior obligatoriamente incorporarán el uso de programas informáticos con software libre”⁷

Por consiguiente, en su respectivo reglamento del 2 de septiembre del 2011, en la sección cuarta dice: “Las instituciones de educación superior obligatoriamente incorporarán el uso de programas informáticos de software libre en los casos que las funcionalidades de estos programas sean similares o superiores al software propietario”⁸.

En definitiva, el espíritu de la ley aspira que el software libre se convierta en una forma de alcanzar la soberanía tecnológica, así como un ahorro significativo de los recursos públicos; los cuales pueden y deben ser dirigidos a esfuerzos que coadyuven a la investigación, desarrollo de ciencia y tecnologías propias.

7 Registro Oficial. LOES. 12 de octubre del 2010 No. 298.

8 Registro Oficial. Reglamento LOES. 2 de septiembre del 2011.

CONCLUSIONES.

El presente trabajo de investigación demuestra que las tecnologías libres aportan de manera significativa en la innovación tecnológica; tanto como un instrumento pedagógico, así como una forma de aplicar los principios morales y éticos que son parte de la formación integral.

Es misión del docente involucrarse de manera significativa en procesos más dinámicos que involucren al alumno de manera individual; y a sus similares de forma colectiva, para generar su propio conocimiento mediante modelos educativos constructivistas y que además les permita compartir ese conocimiento con los demás, usando paradigmas del conectivismo que de la misma forma; mediante este paradigma, se puede propiciar la investigación científica.

Las tecnologías libres son por principio dinamizadoras; que permiten crear nuevas tecnologías, rediseñarlas, renovarlas e innovarlas; desde el punto de vista de código libre y abierto puesto que el estudio de dichos códigos nos permite comprender el funcionamiento de los elementos que forman un sistema informático.

Si aplicamos la Teoría Genel de los Sistemas de (Bertalanffy 1976), como una ciencia que trata de explicar un sistema, dividiéndolo en sus partes y a su vez analizando cada una de ellas, se puede entender cómo funciona el conjunto.

En conclusión, la Educación con Tecnologías Libres fomenta la Innovación; puesto que permite al docente motivar la investigación en sus alumnos utilizando como instrumentos pedagógicos software de código abierto y libre para comprender su funcionamiento y experimentar con el mismo; para que a partir de ahí se puedan buscar alternativas para mejorarlo o adaptarlo a necesidades específicas; de tal forma que sea posible crear nuevas aplicaciones con características de innovación tecnológica.

REFERENCIAS BIBLIOGRÁFICAS.

1. Ausubel, D. P. (1978). In defense of advance organizers: A reply to the critics. *Review of Educational Research*, 48, 251-257.

2. Bertalanffy, L. (1976). *Teoría general de los sistemas: fundamentos, desarrollo, aplicaciones*. México: Fondo de Cultura Económica, ISBN 968-16-0627-2
3. Bruner, J. (1997) Cultura, mente y educación, p. 19-62 en *La educación, puerta de la cultura*. Madrid: Visor.
4. Dewey, J. (1964).: *La ciencia de la educación*. Buenos Aires, Losada.
5. Maslow, A. & Hoffman, E. (2001). *Visiones del futuro*. Barcelona: Kairós. ISBN 84-7245-416-9.
6. Pavlov, I. P. (1928). *Lectures on conditioned reflexes* (Trans. de W. H. Gantt,). Nueva York: International Publishers.
7. Piaget, J. (1952). *The origins of intelligence in children*. Nueva York: International Universities Press.
8. Pressman, R. (2010). *Ingeniería del software: un enfoque práctico*. México: McGraw-Hill., Ed. 7ma. ISBN: 978-607-15-0314-5 McGraw Hill
9. Siemens George y por Stephen Downes (1976). *Conectivismo*. Obtenido en <https://prezi.com/shqlgfb3ohq/conectivismo-de-george-siemens-y-stephen-downes/>
10. Skinner, B. F. (1990). Can psychology be a science of mind? *American Psychologist*, 45, 1206-1210.
11. Vygostky, L.S. (1962). *Thought and Language*. Cambridge: The MLT Press.
12. Wing, J. (2006). Computational thinking. *Communications of the ACM*, 49(3), p.33.
13. WUNDT. W.(1907). *Outlines of psychology*. Wilhelm Enelemann.

BIBLIOGRAFÍA.

1. Alonso Salas, José (2012). *Historia General de la Educación*. RED TERCER MILENIO. ISBN 978-607-733-032-5.
2. Marrou, H. & Barja, Y. (1985). *Historia de la educación en la Antigüedad*. Madrid: Akal.AKAL.
3. Viñas, D. & Barandiaran, X. (2015). *Buen Conocer/FLOK Society: modelos sostenibles y políticas públicas para una economía social del conocimiento común y abierto en el Ecuador*. Quito, Ecuador: Instituto de Altos Estudios Nacionales (IAEN) Centro Internacional de Estudios de Comunicación para América Latina (CIESPAL. ISBN: 978-9978-55-123-3

DATOS DEL AUTOR.

1. Henry Fernando Vallejo Ballesteros. Máster en Informática Educativa y Multimedia, Doctor En Ciencias de la Educación / mención Informática, Diplomado Superior en diseño de Proyectos, Diplomado Superior en Créditos Acumulables y Transferibles, Especialidad en e-Comerce (Comercio Electrónico), Licenciado En Ciencias de la Educación / mención Informática, Tecnólogo en Informática Programación de Sistemas, Profesor Principal de la carrera de Ingeniería de Sistemas de la Facultad Ciencias Administrativas, Gestión Empresarial e Informática en la Universidad Estatal de Bolívar. Correo electrónico: hvallejo@ueb.edu.ec

RECIBIDO: 2 de junio del 2018.

APROBADO: 19 de junio del 2018.