

Open hardware y software, herramientas para el desarrollo de competencias educativas.

Open hardware and software tools for the development of educational skills.

Javier Díaz Sánchez

BENEMÉRITA UNIVERSIDAD AUTÓNOMA DE PUEBLA

jdiazsz@hotmail.com

Resumen

El siguiente material presenta una propuesta educativa que se está desarrollando como parte de un Proyecto Académico dentro del Nivel Medio Superior correspondiente a la institución que pertenezco. Es una propuesta que tiene por objetivo acercar las tecnologías de la información y comunicación al alumnado, tocando temas que van desde el medio físico (hardware) hasta el logístico (software de programación), sin necesidad de acentuar en conocimientos demasiado tecnicizados ya que se trata de un acercamiento parcial, lo que no significa escueto, sino significativo para el proceso enseñanza-aprendizaje y que permitirá al estudiante la oportunidad de acercarse al mundo digital, en un hecho que puede ampliar sus expectativas con la ciencia, al estar en contacto e interacción con el mundo tecnológico que lo rodea a través de una “práctica de campo” .

El marco de desarrollo se ubica en una dimensión adecuada a los tiempos y conocimientos previos, teniendo en cuenta el nivel educativo donde se desenvuelve nuestro alumnado, ya que no se tiene un enfoque “técnico” en un bachillerato bivalente, sin embargo, esta situación puede ser solventada con elementos pedagógicos adecuados y materiales didáctico ad hoc, que no implique instrumentación o inmueble especializado y que sean de fácil acceso, es por ello que se propone el uso de la placa electrónica Arduino y el software Inventor, para elaborar una práctica simple que toca al mundo del hardware, el software y la web.

La valoración del desempeño o criterios de evaluación de esta "propuesta" no está a consideración en primera instancia oficial, sin embargo, puede adecuarse como una "práctica transversal" entre las materias de informática y Física, ajustando en los instrumentos de evaluación, la justificación conforme a la dinámica de la educación por competencias, y a las necesidades de actualización pertinentes a la realidad que vive el alumnado.

Abstract

The following materials presents an educational proposal That Is Being developed as part of an academic project in the Middle Level Higher Relating to the institution Which I belong. It is a proposal That AIMS to bring the technologies of information and communication to students, touching on topics ranging from the physical media (hardware) to the logistical (scheduling software) Emphasizing knowledge without too technically advanced as it is a partial approach, Which does not mean brief, but significant for the teaching-learning process and allow students the opportunity to approach the digital world, a fact That can broaden Their expectations with Science, Being in contact and interaction With the technological world around him-through a "practice field" .

The development framework is at a size Adapted to the times and prior knowledge, taking into account the educational level Where our students unfolds as it does not Have to "technical" approach in a dual high school, however, can be solved esta With Appropriate pedagogical elements and ad hoc teaching materials That Does not Involve property or specialized instrumentation and are Easily accessible, Which is why the use of electronic Arduino Inventor software and Intends to Develop simple practice touch the world of hardware, software and web.

The performance appraisal or evaluation criteria of this "proposal" is not a consideration in first official instance, however, it can be Adapted as a "cross-practice" among the subjects of computer science and physics, adapt assessment tools, the justification

ACCORDING TO dynamics of competence education, and Relevant to the needs of the students whos live update.

Palabras Clave / Key words: Open Hardware, Open Software, Arduino, Inventor, Constructivismo, Competencias Digitales. / Open Hardware, Open Software, Arduino, Inventor, Constructivism, Digital Skills.

Introducción

Constructivismo

La teoría constructivista se sustenta en la necesidad de construir el conocimiento a través de hechos desarrollados en contextos ricos de aprendizaje significativo, además de “proponer que el ambiente de aprendizaje debe sostener múltiples perspectivas o interpretaciones de la realidad, construcción de conocimiento, actividades basadas en experiencias ricas en contexto” (Jonasse, 1991). Todo centrado en la construcción a través de tareas auténticas por su relevancia y utilidad en el mundo real y no sólo en su reproducción.

Sustenta que el ser humano, tanto en los aspectos cognoscitivos y sociales del comportamiento como en los afectivos, no es un simple producto del ambiente ni un escueto resultado de sus inherentes destrezas, sino una construcción propia que se va cimentando día a día como resultado de la interacción entre esos dos factores. Afirma que el conocimiento no es una copia de la realidad, sino una construcción propia del ser y del ambiente que lo rodea como una relación biunívoca entre estos entes.

En general el concepto constructivismo se ha asociado a múltiples matices, sin embargo, se han considerado dos categorías explícitas:

- a) La conceptualización de Piaget, que está preocupada por entender los procesos cognitivos en sí mismos.
- b) La importancia de lo social en el aprendizaje, relacionadas sobre todo con la teoría sociocultural de Vygotski, la cual da un origen social al lenguaje y al pensamiento (Vygotski, 1988).

Savery y Duffy (1996) combinan ambas categorías al resumir la posición constructivista en tres puntos:

- a) El aprendizaje sucede siempre como resultado de nuestras interacciones con el contexto.

- b) El estímulo para aprender viene de un conflicto cognitivo interno y personal.
- c) El conocimiento se genera socialmente, a través de poner a prueba nuestras propias representaciones con las de los demás.

Finalmente, cabe definir que aunque el constructivismo ha sido uno de los pilares más importantes en la formación de nuevos paradigmas educativos, debe entenderse como un marco epistemológico de referencia, no como un modelo de aprendizaje o de enseñanza (Savery y Duffy, 1996; Gómez-Granell, C., & Coll, 1994; Millán, 1995; etc.).

Con el Constructivismo.

- Se aprende construyendo esquemas mentales.
- Se desarrollan aprendizajes dirigidos a la comprensión del entorno.
- Se tiene conciencia, pues requieren actividad consciente del sujeto.
- Se obtienen aprendizajes generalmente más duraderos.
- Los aprendizajes son significativos, son fruto de la reflexión.

Aprendizaje significativo

Es el conocimiento que integra el alumno a sí mismo y se ubica en la memoria permanente, éste aprendizaje puede partir de un origen o producto de la información, conductas, actitudes o habilidades propias o del medio. La psicología perceptual considera que una persona aprende mejor aquello que percibe como estrechamente relacionado con su supervivencia o desarrollo, de aquí la importancia del entorno.

Tres factores intervienen en la integración de lo que se aprende:

- Los contenidos, conductas, habilidades y actitudes por aprender;
- Las necesidades actuales y los problemas que enfrenta el alumno y que vive como importantes para él;
- El medio en el que se da el aprendizaje.

Los modelos educativos centrados en el alumno plantean que el profesor debe propiciar el encuentro entre los problemas y preguntas significativas para los alumnos asociados a los contenidos, esto favorece la aptitud del alumno para aprender a interrogar e interrogarse dentro de las condiciones donde se desarrolla el proceso educativo, esto explicita que además de tenerse en cuenta que el aprendizaje involucra aspectos cognoscitivos, también los hechos afectivos del medio afectan.

Open Hardware

La expresión open hardware, u open source hardware, se refiere al hardware cuyo diseño se hace público para que cualquiera pueda estudiarlo, modificarlo y distribuirlo, así como el poder producir y vender hardware basado en ese diseño.

El movimiento de hardware abierto o libre, busca crear una gran librería accesible para todo el mundo, lo que permitiría a las compañías una importante reducción de millones de dólares en trabajos de diseño redundantes. Dado que es más fácil tener una lluvia de ideas propuesta por miles o millones de personas, que solo una compañía propietaria del hardware y sus empleados, de esta manera los usuarios interesados en cómo funciona un dispositivo electrónico, puedan fabricarlo, programarlo y poner en práctica esas ideas en alianza con las empresas fabricantes, algo viable en ambos sentidos

Al hablar de open hardware se debe especificar de qué tipo de hardware se está hablando, esto evita confusión en la interpretación del concepto:

- Hardware estático. Se refiere al conjunto de los elementos materiales que constituyen al ensamblaje electrónico (tarjetas de circuito impreso, resistencias, capacitores, LEDs, sensores, etcétera).
- Hardware reconfigurable. Es aquél que es descrito mediante un HDL (Hardware Description Language). Se desarrolla de manera similar al software. Los diseños son archivos de texto que contienen el código fuente y son cargados en el módulo electrónico.

Para acceder a este hardware reconfigurable, se debe usar un lenguaje de programación con licencia GPL (General Public License). La licencia GPL, al ser un documento que cede ciertos derechos al usuario, asume la forma de un contrato, por lo que usualmente se la denomina contrato de licencia o acuerdo de licencia, que permitirá su manipulación como open hardware, este es el caso de Arduino.

DESARROLLO

Arduino

Es una plataforma de electrónica abierta, basada en software y hardware flexibles, creada para flexibilizar el diseño de entornos u objetos interactivos y facilitar la enseñanza en áreas como la programación y electrónica.

Su historia surge en 2005, cuando un grupo de estudiantes y profesores del Instituto de Diseño Interactivo Ivrea en Italia empezó a desarrollar una plataforma de hardware en open source porque las placas del mercado eran demasiado caras para trabajar. Seis años después, Arduino supone una revolución en la forma en que se crea y usa la tecnología.

Tal y como narra el documental Arduino, al equipo inicial se fueron sumando más personas, y el primer prototipo evolucionó hacia modelos más accesibles y fáciles de utilizar gracias a la colaboración abierta a través de la Red, siguiendo la filosofía del software en código abierto. Las primeras unidades se vendieron a un euro por placa y hoy en día, la comunidad Arduino tiene más de 120 mil usuarios.

La mayor revolución del hardware en open source está en la posibilidad que brinda a la gente de aprender cómo funciona la tecnología, y ser capaz de manipular y crear objetos físicos. Según David Cuartielles, también del equipo Arduino, "por culpa de la _blankización y las patentes, se había cerrado la oportunidad a la gente para aprender cómo funcionan las cosas y solo los expertos y hackers podían abrir y ver lo que hay dentro. El hardware en open source significa poder volver a abrir las cosas y ver cómo funcionan.

Open Software

Software libre (open source) es el software que respeta la libertad de los usuarios y la comunidad; significa que los usuarios tienen la libertad de ejecutar, copiar, distribuir, estudiar, modificar y mejorar el software. Es decir, el concepto es una cuestión de libertad, no de precio. Para entender el término, piense en libre como en libre expresión, no como en libertinaje, existen reglas comunales que deberán acatarse para que no exista el peligro de adueñamiento y todos sigan compartiendo y aportando.

Un programa es software libre si el usuario tiene cuatro libertades esenciales sobre el:

- La libertad de ejecutar el programa como se desea y con “cualquier propósito”.
- La libertad de comprender cómo funciona el programa, y cambiarlo para que haga lo que usted quiera. El acceso al código fuente es una condición insoslayable para ello.
- La libertad de redistribuir copias para ayudar a los demás.
- La libertad de distribuir copias de sus versiones modificadas a terceros. Esto permite ofrecer a toda la comunidad la oportunidad de beneficiarse de las modificaciones. El acceso al código fuente es una condición necesaria para ello.

Un programa es software libre si otorga a los usuarios todas estas libertades mencionadas de manera apropiada. De lo contrario no se encuadra en el concepto libre.

Para qué aprender programación de computadoras

Aprender a programar computadoras no es algo nuevo dentro del sector educativo, desde 1967 se hicieron grandes aportaciones a través del uso del lenguaje de programación LOGO, y que a pesar de ser considerado como una especie de dialecto del lenguaje Lisp; tiene un objetivo educacional. Logo es un lenguaje para aprender, es una herramienta útil para enseñar el proceso de aprendizaje y de pensamiento.

Los responsables por la creación del lenguaje Logo son Seymour Papert y Wallace Feurzeig. Además Papert sustentó su teoría denominada Contruccionismo basada en la teoría del Constructivismo de Jean Piaget, por lo que los fundamentos básicos para la elaboración del lenguaje están sostenidos los principios del aprendizaje, ya que se afirma que el aprendizaje es mejor cuando los niños se comprometen en la construcción de un producto significativo, como un castillo de arena, un poema, un programa, una canción; lo que involucra dos tipos de construcción: cuando los niños construyen en mundo externo y los que construyen interiorizando el conocimiento en sus mentes.

En la actualidad existe diversos entornos que siguen invitando a “construir” productos derivados de lenguajes o entornos un tanto simplificados pero no menos productivos que son capaces de producir importantes aportaciones, no sólo personales; sino el inicio de futuros desarrollos, algunos de estos lenguajes son: Scratch, Alice, Kodu, RoboMind, RoboLab y AppInventor. De este último se hace uso para el desarrollo de la propuesta académica explicitada en esta investigación.

La reseña anterior permite justificar la necesidad de utilizar los medios tecnológicos actuales en pro del desarrollo académico en el sentido propio y comunal, Díaz Barriga y Hernández (1998) escriben: “La concepción constructivista del aprendizaje escolar se sustenta en la idea de que la finalidad de la educación que se imparte en las instituciones educativas es promover los procesos de crecimiento personal del alumno en el marco de la cultura del grupo a que pertenece.”

Además es evidente que el quehacer institucional debe considerar las adecuaciones necesarias para incluir actividades capaces de colocar a la vanguardia al alumnado en el manejo y comprensión de las nuevas herramientas tecnológicas, mismas que formarán un importante precedente en su formación académica e incluso se puede ahondar más en esta temática y sustentar el hecho de una formación transversal por competencias que estará implícita en el perfil de egreso.

App Inventor

Es un entorno basado en la interpretación de un lenguaje de programación estructurado en bloques, es también una herramienta de diseño y un entorno de desarrollo de aplicaciones para móviles que funcionen con el sistema operativo Android. App Inventor permite también ejecutar las aplicaciones en un emulador descargable o en el intérprete de una App propia de MIT, por lo que no es indispensable disponer del teléfono para probar los programas elaborados.

Es una aplicación Web 2.0 que permite desarrollar aplicaciones para dispositivos móviles que utilicen el sistema operativo Android a través de su “codificación” utilizando un navegador de manera muy rápida y fácil.

Para la enseñanza de la programación esta app dispone de elementos gráficos que “interaccionan” y “compilan” en tiempo real.

Para crear las aplicaciones se utilizan dos componentes:

1. El App Inventor Designer, donde se eligen los componentes para la aplicación.
2. El App Inventor Blocks Editor, donde se ensamblan los bloques de programas que especifican como los componentes deben funcionar de manera visual.
3. La ejecución se realiza utilizando un emulador, físicamente y vía la ejecución en el móvil mediante una máquina virtual utilizando una url o código QR a través de una conexión web.

Ejecución

Lo más destacado es que se puede testar la aplicación en el móvil para valorar su respuesta “real”, sin necesidad de arriesgar el dispositivo, y una vez terminado el trabajo de “programación”, se tiene la capacidad de descargar el archivo para instalar (apk), lo que garantiza una migración adecuada entre los diversos smartphones, tablets o phablets con sistema Operativo Android.



Ventajas

- App Inventor puede ejecutarse en sistemas operativos como MAC OS, Linux o Windows
- Es simple, visual, intuitivo e interactivo, ya que la “compilación” se ejecuta en tiempo real y libera de la necesidad de saber programar exclusivamente para Android.

Además de los elementos pedagógicos, App Inventor tiene algunas características que lo hacen especialmente adecuado:

1. Es software libre, no es necesario la adquisición de licencias por parte del profesor y los alumnos

2. Es multiplataforma: requiere sólo de un navegador y la máquina virtual de Java instalada, con “java web start”.

Es programación para dispositivos móviles; los teléfonos y las tabletas están marcando hoy en día buena parte del futuro de la informática y es atractivo para los alumnos poder enseñar en sus propios teléfonos el fruto de su trabajo.

Objetivos de la Investigación

Educativos

- Establecer una línea de investigación pedagógica que permita elaborar una práctica de tipo transversal en áreas tecnológicas y teóricas de materias afines.
- Tangibilidad de lo abstracto a lo pragmático
- Este trabajo va dirigido a estudiantes del VI semestre de las áreas tecnológicas principalmente, pero la proyección considera a las demás ofertas educativas.

Prácticos

- Elaborar un mecanismo electrónico manejado a través de una interface móvil vía comunicación bluetooth.
- Documentar el desarrollo mecánico, electrónico y digital para su uso en una práctica de laboratorio.
- Disponibilidad económica de los dispositivos electrónicos y lógicos.

Competencias Educativas que son aplicables en este Proyecto

Las siguientes competencias pertenecen al SNB que permite establecer un Marco Común Curricular entre los subsistemas y entidades que componen al sector educativo a nivel nacional conforme a los acuerdos SEP.

Competencias Genéricas

Se expresa y comunica

4. Escucha, interpreta y emite mensajes pertinentes en distintos contextos mediante la utilización de medios, códigos y herramientas apropiados.

Atributos:

1. Expresa ideas y conceptos mediante representaciones lingüísticas, matemáticas o gráficas.
5. Maneja las tecnologías de la información y la comunicación para obtener información y expresar ideas.

Piensa crítica y reflexivamente

5. Desarrolla innovaciones y propone soluciones a problemas a partir de métodos establecidos.

Atributos:

1. Sigue instrucciones y procedimientos de manera reflexiva, comprendiendo como cada uno de sus pasos contribuye al alcance de un objetivo.

Competencias disciplinares

Ciencias Experimentales

4. Obtiene, registra y sistematiza la información para responder a preguntas de carácter científico, consultando fuentes relevantes y realizando experimentos pertinentes.
5. Contrasta los resultados obtenidos en una investigación o experimento con hipótesis previas y comunica sus conclusiones.
9. Diseña modelos o prototipos para resolver problemas, satisfacer necesidades o demostrar principios científicos.

Competencias			Competencias Disciplinarias		
Atributo	Genérica 4	Genérica 5	Experimental 4	Experimental 5	Experimental 9
1	<ul style="list-style-type: none"> Interpreta conceptos de electrónica básica. Utiliza código y manuales en el manejo de circuitos y componentes (resistencias, diodos, leds). 	A partir de los conocimientos adquiridos en el manejo de los componentes y circuitos, elabora soluciones que responden a necesidades de interfaz entre el mundo analógico y el digital	<ul style="list-style-type: none"> Consulta manuales de IC, Arduino, Componentes y Programación. Para llevar a cabo pruebas y ensamblaje. 	Experimenta y Comprueba a través de pruebas de laboratorio en forma responsable	Elabora el ensamblaje del proyecto, manufactura o propone mejoras al prototipo, ya sea de manera mecánica, digital o informática.
5	Utiliza medios digitales e internet para buscar información sobre hojas de datos aplicados a integrados diversos y componentes				

Tabla de Competencias desarrolladas por el alumnado en el Proyecto

El Proyecto Desarrollado

Elementos Electrónicos que la componen

Sensor Ultrasónico HC-SR04

El desarrollo de un móvil requiere del uso de sensores capaces de proporcionar respuesta a condiciones externas que permitan tomar una decisión a situaciones que comprometan el estado de un sistema, estas decisiones son programadas en el interior del algoritmo desarrollado expofeso, por lo que es necesario adaptar que sensor o grupo de sensores es viable. Para este caso y por las restricciones y necesidades básicas de desplazamiento se hace uso del sensor ultrasónico HC-SR04.

El módulo de comunicación BlueTooth

La comunicación inalámbrica es imprescindible, ya que las aplicaciones actuales exigen la implementación de elementos mínimos de intercambio de datos en medios externos no fijo, que permiten además un manejo más libre entre los dispositivos y su entorno, y en este trabajo no podía ser la excepción, para ello se hará uso de módulo bluetooth que permite ser incorporado a Arduino a través de una pastilla básica.

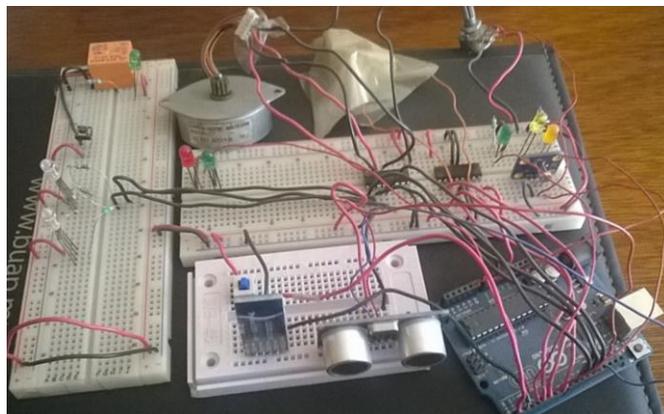
Motor CD como elementos para el desplazamiento

Para disponer de movilidad para este vehículo a desarrollar, será necesario el uso de un motor DC adaptado a los requisitos mecánicos más básicos para el desplazamiento y velocidad, aunque si bien no se considera el control de velocidad esta puede ser controlado con un el manejo de un dispositivo INTEGRADO denominado **L293D** que proporciona el control analógico digital de la interface.

Motor a pasos, control de dirección

Desplazarse también requiere de giros de dirección, para que el vehículo tenga esta capacidad será necesario implementar un control de dirección, que estará sujeto a la disposición de grados de libertad que pueda manejar la estructura del mismo. Para solventar esta necesidad, se hace uso de un motor a pasos que permite un desplazamiento gradual controlado.

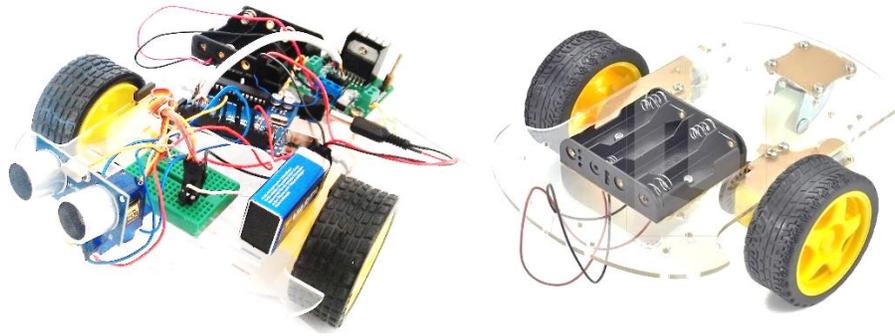
Conclusión



Se presentan en las imágenes la estructura general del trabajo de ensamble electrónico, que se llevará al montaje mecánico adecuado, para dar forma al proyecto final. Este último trabajo implica analizar con detalle elementos que proveerán de autonomía al vehículo a desarrollar, por ejemplo, se considera el tipo de pilas, el peso, la alimentación del Arduino y el control de los motores a través del módulo bluetooth.

Finalmente, elaborar un manual descriptivo sobre como ensamblar la práctica no sólo requiere de terminología específica, sino también de metodología que permita llevar a cabo la secuencia didáctica adecuada, misma que parte desde la conceptualización hasta el ensamblaje electrónico y mecánico, esta estructura debe propiciar un ambiente de aprendizaje significativo que permita al alumnado en primera instancia la elaboración y como consecuencia la “creación” del prototipos correspondientes al mundo digital, que a su vez sean tangibles a su realidad.

Este proyecto pretende sustentar la necesidad de llevar a la educación, la apertura a la generación de nuevos aprendizajes a través de los nuevos lineamiento tecnológicos, que pueden darse en una práctica, que si bien los recursos son limitados, estos materiales pueden ser adquiridos a un bajo costo, no son derogaciones altas, sino financiables, que darán oportunidad a las nuevas generación de acercarse a la ciencia y tecnología.



Bibliografía

EVANS, Brian W. Arduino Programming Notebook: A Beginner's Reference (2008).

LÓPEZ, Miguel Ángel. Aprendizaje, competencias y TIC. Pearson. México. 2013

PERRENOUD, Philippe. Construir competencias desde la escuela. Dolmen Ediciones. Chile. 2002

MARZANO, R. J., & J.Pickering, D. (2005). *Dimensiones del aprendizaje. Manual para el maestro*. Tlaquepaque.Jalisco, México: Instituto Tecnológico y de Estudios Superiores de Occidente (ITESO).

MONK, Simon. Programming Arduino: Getting Started With Sketches (2011). ISBN-13 978-0071784221.

PRESSMAN, Roger. Ingeniería del Software. Un Enfoque Práctico. ISBN 9701054733, 900 Páginas. McGraw-Hill. Sexta Edición.