

## METODOLOGÍA PARA EVALUACIÓN DE USABILIDAD DEL ENTORNO DE DESARROLLO INTEGRADO DE ARDUINO

### METHODOLOGY FOR USABILITY EVALUATION OF THE ARDUINO INTEGRATED DEVELOPMENT ENVIRONMENT

(Recibido 31/03/2017) – (Aceptado 28/06/2018)  
<https://doi.org/10.32645/13906925.514>

#### **CARLOS XAVIER ROSERO CHANDI**

Doctorando en el programa de Automática, Robótica y Visión, y Máster en Automática y Robótica, ambas por la Universitat Politècnica de Catalunya; Ingeniero Automotriz por la Escuela Politécnica del Ejército. Actualmente se desempeña como profesor de sistemas embebidos y de tiempo real, sistemas de control moderno, y robótica e inteligencia artificial, siendo jefe del área de automática en la Carrera de Ingeniería Mecatrónica de la Facultad de Ingeniería en Ciencias Aplicadas de la Universidad Técnica del Norte, y como director del Grupo de Investigación en Sistemas Inteligentes de la misma. Es miembro del Grupo de Investigación en Robótica Inteligente y Sistemas, del Departamento de Control Automático de la Universitat Politècnica de Catalunya.

#### **CRISTINA FERNANDA VACA ORELLANA**

Magíster en Tecnologías para la Gestión y Práctica Docente por la Pontificia Universidad Católica e Ingeniera en Sistemas Computacionales por la Universidad Técnica del Norte. Se desempeña como profesora de informática aplicada en la Carrera de Nutrición y Salud Comunitaria, y la Carrera de Medicina de la Facultad de Ciencias de la Salud de la Universidad Técnica del Norte. Es autora de varias publicaciones referentes a tecnologías de la información y comunicación aplicadas en la educación superior.

Universidad Técnica del Norte, Ibarra, Ecuador

[cxrosero@utn.edu.ec](mailto:cxrosero@utn.edu.ec)

[cvaca@utn.edu.ec](mailto:cvaca@utn.edu.ec)

### *Resumen*

*Los sistemas embebidos de hardware libre (OSHW, Open Source Hardware) han sido ampliamente aplicados en la enseñanza de sistemas microprocesados; la edición, compilación y programación de aplicaciones para OSHW se realizan a través de entornos de desarrollo integrado (IDEs, Integrated Development Environments). Este artículo presenta una metodología para evaluar la usabilidad de*

*IDEs de OSHW y se centra en la plataforma de Arduino por ser una de las más utilizadas hoy en día en la academia; conocer qué tan fácil y confortable es aprender a programar microcontroladores en Arduino es un objetivo colateral pero de igual importancia. La principal contribución de este trabajo consiste en utilizar enfoques de efectividad, eficiencia y satisfacción del usuario para la evaluación de usabilidad de IDEs de programación de OSHW. La evaluación se centra en la realización de programas con complejidad dentro de una escala ascendente. La efectividad se determina en relación al cumplimiento satisfactorio de tareas; la eficiencia se evalúa en base al tiempo empleado para el desarrollo de programas. El grado de satisfacción del usuario se diagnostica usando encuestas de escala de usabilidad del sistema y de facilidad de uso. Los resultados asociados muestran que la efectividad y eficiencia del usuario utilizando el IDE dependen de la complejidad de las tareas y que el manejo de la interfaz es fácil y cómodo.*

**Palabras Claves:** Hardware abierto; Entorno de Desarrollo Integrado; Efectividad; Eficiencia; Satisfacción del usuario.

### **Abstract**

*Embedded systems based on open source hardware (OSHW) have been widely applied for teaching of microprocessed systems; the edition, compilation and programming of applications for OSHW are carried out through integrated development environments (IDEs). This paper presents a methodology to evaluate the usability of OSHW IDEs and focuses on the Arduino platform as one of the most used currently in academy; knowing how easy and comfortable it is to learn how to program microcontrollers in Arduino is a collateral objective but of equal importance. The main contribution of this work is to use a focus on effectiveness, efficiency and user satisfaction for the usability evaluation of OSHW IDEs. The evaluation focuses on carrying out programs with complexity within an ascending scale. Effectiveness is determined in relation to the satisfactory fulfillment of tasks; efficiency is evaluated based on the time used to develop programs. The degree of user satisfaction is diagnosed using surveys of both usability scale of the system and ease of use. The associated results show that the effectiveness and efficiency of the volunteer using the IDE depend on the tasks complexity and that the interface handling is easy and comfortable.*

**Palabras Claves:** OSHW; IDE; Effectiveness; Efficiency; User Satisfaction.

## **1. Introducción**

El hardware de fuentes abiertas (OSHW, Open Source Hardware) se ha difundido en el desarrollo de sistemas embebidos en la actualidad. OSHW comprende hardware (HW) y software (SW); el primero incluye toda la información referente a su diseño y fabricación, y el segundo se usa para configurar y/o programar al primero (Gibb, 2015). El software puede presentarse como un conjunto de herramientas separadas, por ejemplo Atmel Corporation (2016a), o como un entorno de desarrollo integrado (IDE, Integrated Development Environment) que integra un editor-depurador y herramientas de construcción automática de la aplicación final, agregando detalles como confort,

eficiencia y soporte. Dos ejemplos se encuentran en HP InfoTech (2016) y Arduino Company (2016).

La usabilidad describe cómo cierto usuario utiliza un software en particular (Mascheroni, 2013a). Los análisis de usabilidad de IDEs han sido frecuentemente empleados por investigadores para examinar el comportamiento de desarrolladores que los usan en campos de acción determinados, por ejemplo en Bacíková (2015). Estos análisis incluyen destacar qué y cómo son usadas ciertas herramientas para poder rehacerlas a fin de mejorar su efectividad y eficiencia. Considerando que la evaluación de usabilidad de una interfaz se aplica generalmente en la etapa de diseño, existe poca información sobre metodologías de evaluación de software ya difundido (Mascheroni, 2013b). Adicionalmente, ninguno de los trabajos citados presenta metodologías para el análisis de usabilidad de una plataforma OSHW en aplicaciones de enseñanza-aprendizaje. En efecto, sólo en Yago (2015) existe un estudio de usabilidad en la etapa de diseño de un aditamento anexo a la plataforma Arduino, que además aporta con modificaciones significativas en el IDE.

La simplicidad de uso, bajo precio y soporte de una extensa comunidad online, han hecho de Arduino (Arduino Company, 2016) una de las plataformas con mayor impacto en los años recientes (Oxer, 2009). Tal es su trascendencia en la academia que ha sido adoptada en ámbitos que comprenden escuelas, universidades y centros de investigación (Soriano, 2014). Con el uso de Arduino los desarrolladores se centran en la resolución del problema en lugar de interesarse en asuntos complejos de hardware (Plattner, 2016). En efecto, la inevitable parte electrónica es abordada a través de artilugios prefabricados llamados gadgets (Karvinen, 2011), dejando de ser un obstáculo en el desarrollo de cualquier aplicación. En la educación en ingeniería Arduino es una plataforma usada en metodologías de aprendizaje basadas en proyectos que tienen como parte fundamental a la electrónica, los sistemas embebidos, y los sistemas ciber-físicos, véase por ejemplo Soriano (2014) y Grover (2014).

El aporte de este artículo es doble. En primer lugar presenta una metodología para evaluar la usabilidad de IDEs de OSHW enfocada en los atributos de efectividad, eficiencia y satisfacción del usuario, siguiendo los planteamientos propuestos por Mazumder (2015). En segundo lugar, busca comprobar que la aplicación de la metodología permite evaluar el desempeño y la satisfacción del usuario en el proceso de aprendizaje de programación de microcontroladores, para lo cual usa a la plataforma Arduino por ser una de las más utilizadas hoy en día en la academia. Para lo anterior se considera un grupo objetivo compuesto por usuarios novatos. Esta población corresponde a los estudiantes de sistemas microprocesados de la Carrera de Ingeniería en Mecatrónica de la Universidad Técnica del Norte, en el período académico octubre 2016 – febrero 2017.

Se prevé que la metodología propuesta permita evaluar los atributos de usabilidad mencionados de IDEs de OSHW. Como resultado esperado, los estudiantes novatos en el período académico citado pueden aprender en forma más fácil y con una mejor experiencia con Arduino que con otras plataformas, ya que Arduino presenta superior usabilidad (en base a los atributos evaluados). Más allá, la metodología propuesta podría adaptarse para evaluar cualquier IDE utilizado en la programación de OSHW.

Este artículo está organizado tal como se detalla a continuación. En el apartado 2 se propone la metodología para evaluación de usabilidad. En el apartado 3 se describe el montaje experimental y se muestra un análisis comparativo y ciertas reflexiones acerca de los resultados obtenidos. En el apartado 4 se concluye el trabajo, y por último en el apartado 5 se especifican ciertas recomendaciones.

## **2. Materiales y métodos**

Esta sección describe los atributos valorados y pormenoriza la metodología para evaluación de usabilidad propuesta, la misma comprende pruebas y métricas.

### **2.1 Atributos de usabilidad a evaluar**

La usabilidad es una cualidad cada vez más importante en la calidad de los productos de software; es un atributo intangible del programa computacional y se considera difícil de visualizar, medir y reconocer como un factor determinante de su calidad (Mascheroni, 2013b). En breves palabras la usabilidad define cómo el usuario utiliza un software. La Organización Internacional de Normalización (ISO, International Organization for Standardization) establece que la usabilidad de un producto es la medida en que el usuario puede utilizarlo para lograr sus objetivos con eficacia, eficiencia y satisfacción (Mazumder, 2015).

Para la evaluación de IDEs de OSHW se propone usar la norma ISO 9241 orientada a la calidad en usabilidad y ergonomía en HW y SW, tal como se menciona en Sanz (1997). Así mismo en Preece (2002) se establece que la evaluación de usabilidad permite asegurar que el software sea interactivo, fácil de aprender, eficaz de usar, y agradable desde la perspectiva de la experiencia de usuario. Así, los atributos de la usabilidad a evaluar son:

1. Efectividad que se refiere al desempeño de las tareas; qué tan precisamente y completamente el usuario alcanzó los objetivos.
2. Eficiencia que considera la cantidad de esfuerzo que se requirió para lograr el nivel de efectividad cuando el usuario alcanzó los objetivos. Es la relación entre el nivel de efectividad y el consumo de recursos.
3. Satisfacción que establece qué tan confortable se siente el usuario mientras usa la interfaz.

### **2.2 Metodología propuesta para la evaluación de usabilidad**

Se propone un método de investigación cuantitativo-experimental para recolectar información concerniente a tres tipos de evaluaciones que corresponden a los atributos de usabilidad mencionados en la sección 2.1. La efectividad y eficiencia se valoran mediante la técnica de observación directa; para determinar la satisfacción se aplica la técnica de la encuesta.

#### **2.2.1 Tareas propuestas**

La parte experimental del método está conformada por varias tareas (con grado de dificultad ascendente) que deben ser programadas por los voluntarios sobre el OSHW, usando el IDE sujeto a evaluación en un período limitado de tiempo. Posteriormente, el evaluador registra, analiza e interpreta los datos considerando los tres atributos de usabilidad.

Para seleccionar las tareas se consideran ciertos periféricos del microcontrolador cuyo funcionamiento debe ser explicado previamente de manera teórica en el curso de sistemas microprocesados. Las tareas propuestas son:

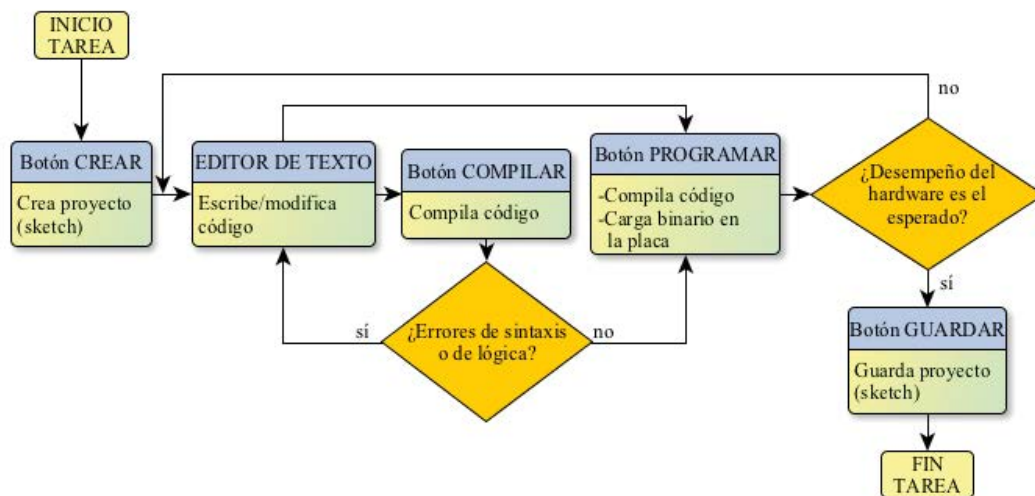
- Tarea 1 (T1), puesta a punto del IDE y del OSHW. Se evalúa el proceso de conexión del sistema embebido a ser programado y el funcionamiento correcto del IDE (véase la Figura 1 para mayor comprensión).
- Tarea 2 (T2), intermitencia de un puerto digital. Consiste en utilizar un puerto digital, configurarlo como salida y generar sobre éste una intermitencia haciendo uso de un retardo pasivo (véase la Figura 2 para comprender el algoritmo recomendado para las tareas T2 a T4).
- Tarea 3 (T3), generación de una señal analógica. Se basa en el uso de un temporizador para modular el ancho de pulso de cierta salida digital cambiando el ciclo útil en forma cíclica entre 0 y 100%.
- Tarea 4 (T4), lectura de una señal analógica y envío de su valor hacia el computador mediante comunicación serial. Se fundamenta en la obtención de una señal analógica (proporcionada por un potenciómetro) a través de un canal del conversor análogo/digital. Los valores obtenidos se envían al ordenador por medio del módulo de comunicación asincrónica.

En la Figura 1 se propone un algoritmo general para la puesta en marcha del IDE cuando el HW se conecta por primera vez. Se puede notar que el proceso termina cuando el usuario presiona el botón Compilar.



**Figura 1:** Algoritmo general de puesta a punto del IDE

Un algoritmo general para el empleo del IDE en el desarrollo de los programas encomendados (véase el diagrama de flujo de la Figura 2) permite estandarizar su utilización. Esto mitiga el desconocimiento total sobre su uso y evita las variadas primeras exploraciones que podrían realizar los voluntarios. Como consecuencia, se homogeneiza en la muestra el efecto del desconocimiento del IDE sobre el tiempo empleado para realizar las tareas.



**Figura 2:** Algoritmo general de ejecución de tareas

### Consideraciones generales para la evaluación

El algoritmo de cada tarea es entregado a los voluntarios (a través de un diagrama de flujo), además de ciertas indicaciones referentes a la configuración del hardware. Esto permite asegurar uniformidad en el desarrollo de los programas por parte de los voluntarios. No se consiente el acceso a internet ni el uso de programas de ejemplo proporcionados por el IDE.

Se establece el tiempo límite para cada tarea como varias veces el tiempo empleado por el profesor de la asignatura en el desarrollo de la misma. Para considerar que un voluntario ha terminado satisfactoriamente una tarea ésta debe cumplir rigurosamente con el desempeño solicitado y ser entregada dentro del tiempo límite. Si una tarea no cumple lo requerido o el usuario no la termina dentro del tiempo estipulado, es declarada como no satisfactoria con un tiempo de desarrollo igual al tiempo límite.

### Prueba de efectividad

La prueba de efectividad considera el cumplimiento (sí o no) de las tareas  $T_x$  donde  $\chi \in \{1,2,3,4\}$ . La efectividad  $e_x$  en cada tarea se acota dentro del rango  $[0,1]$  como sigue: baja,  $0 \leq e_x \leq 0.33$ , media,  $0.33 < e_x \leq 0.67$ , y alta,  $0.67 < e_x \leq 1$ . La misma es calculada usando la relación

$$e_x = \frac{m_x}{n}, \quad (1)$$

donde  $m_x$  es la cantidad de voluntarios que terminan la tarea  $x$  satisfactoriamente, y  $n$  el total de evaluados (población).

### Prueba de eficiencia

Esta prueba consiste en comparar los tiempos de desarrollo de las tareas  $T_x$  obtenidos tanto con el IDE a evaluar como con otro IDE usado para el mismo objetivo. El tiempo promedio de cada tarea

$t_x^-$ , es calculado usando la ecuación.

$$\bar{t}_x = \frac{1}{n} \sum_{i=1}^n t_{x_i}, \quad (2)$$

donde  $t_{x_i} \in \mathbb{R}$  es el tiempo que el voluntario  $i$  usa para completar la tarea  $x$ ;  $n$  es el total de voluntarios. El radio  $rt_x$  se calcula como la relación entre los  $\bar{t}_x$  de los dos IDE.

### Prueba de satisfacción

Esta prueba se fundamenta en dos evaluaciones: satisfacción y comodidad/facilidad. Ambas poseen una escala de calificación de cinco niveles discretos comprendidos dentro del rango [1,5]. En las dos se consideran estadísticas referentes al valor medio y a la desviación estándar. Para calcular el valor medio  $y_q$  se usa

$$\bar{y}_q = \frac{1}{n} \sum_{i=1}^n y_{q_i}, \quad (3)$$

donde  $q$  representa cada pregunta definida como  $\{q \in \mathbb{N} | 1 \leq q \leq 10\}$  para la evaluación de satisfacción, y  $\{q \in \mathbb{N} | 1 \leq q \leq 4\}$  para la evaluación de comodidad/facilidad;  $y_{q_i}$  es la calificación que el usuario  $i$  coloca en cada pregunta  $q$ ,  $n$  es el tamaño de la población. La desviación estándar  $\bar{S}_q$  se calcula como

$$\bar{S}_q = \sqrt{\frac{\sum_{i=1}^n (y_{q_i} - \bar{y}_q)^2}{n}}, \quad (4)$$

donde todos los parámetros han sido definidos en (3).

Para evaluar la satisfacción de los voluntarios al usar el IDE se propone la encuesta de escala de usabilidad del sistema (SUS, System Usability Scale), donde 1 indica total desacuerdo y 5 total acuerdo con las preguntas realizadas, como se observa en la Tabla 1.

Tabla 1:  
Escala de usabilidad

Pregunta ( $q$ )	Enunciado [valorado entre 1 y 5]
1	Me gustaría usarlo con frecuencia
2	Lo encontré innecesariamente complejo
3	Pensaba que era fácil de usar
4	Necesitaría ayuda para poder usarlo
5	Las diversas funciones estaban bien integradas
6	Había demasiada inconsistencia
7	La mayoría de la gente aprendería a utilizarlo muy rápidamente
8	Lo encontré muy engorroso/difícil de usar
9	Me sentí muy confiado usándolo
10	Necesitaba aprender muchas cosas antes de que pudiera empezar a trabajar con este IDE

Para medir la comodidad y facilidad de interacción entre el usuario y el IDE (Tabla 2) se adapta la encuesta de Manresa-Yee (2010); en cada aspecto evaluado el valor de 5 indica total comodidad y/o facilidad.

Tabla 2:  
Comodidad/facilidad de uso

Pregunta (q)	Enunciado [valorado entre 1 y 5]
1	El esfuerzo mental requerido para el desarrollo de las tareas ha sido [1 mucho esfuerzo, 5 poco esfuerzo]
2	La velocidad de funcionamiento es [1 muy lenta, 5 muy rápida]
3	El confort es [1 muy incómodo, 5 muy cómodo]
4	En general, el manejo del IDE es [1 muy difícil, 5 muy fácil]

## Resultados y discusión

A continuación se aplica la metodología propuesta sobre una plataforma ampliamente difundida en la academia. Además, se realiza el análisis de resultados correspondiente.

### Configuración del experimento

Dentro de la familia de placas Arduino, se ha escogido el OSHW Arduino UNO revisión 3 (2016) debido a su bajo costo y a que es la opción más usada y documentada dentro de la comunidad científica.

En lo concerniente al IDE Arduino, se ha instalado la versión 1.6.13 (Arduino Company, 2016) sobre computadores con Windows 10 de 64 bits, con procesador Intel Core i5 a 1.7Ghz y memoria RAM de 8GB. Un segundo IDE usado para la comparación de resultados en las pruebas de eficiencia es el CodeVision AVR 3.29 (HP InfoTech, 2016).

La población a evaluar comprende solamente a usuarios nuevos de la plataforma Arduino. La evaluación sobre usuarios con cierta experiencia previa produciría resultados con tendencia a valores de eficacia y eficiencia altos, ya que el desempeño del usuario aumenta conforme se incrementa su familiaridad con el SW (Manresa-Yee, 2010). La selección de la muestra es aleatoria a fin de garantizar su representatividad dentro de la población. Así, se han seleccionado 10 usuarios/voluntarios, estudiantes de la cátedra de sistemas microprocesados. Su edad fluctúa entre 20 y 21 años y en referencia a su escolaridad, todos se encuentran cursando las mismas clases. Dentro de su récord académico no consta que hubiesen repetido curso alguno. Ninguno de ellos ha utilizado la plataforma Arduino previamente.

A continuación se muestran los resultados de las pruebas de efectividad, eficiencia y satisfacción considerando las tareas (T1 a T4) estipuladas en la sección 2.2.1. Los tiempos máximos de ejecución luego de los cuales estas tareas se consideran fallidas son 150, 180, 300 y 360 segundos, respectivamente.



### Prueba de efectividad

La Tabla 3 muestra los resultados de esta evaluación donde se detalla el cumplimiento o no de las tareas por parte de los voluntarios. En la Figura 3 se observan las efectividades  $e_x$  correspondientes a cada tarea calculadas a través de la Ecuación 1.

Tabla 3:  
Efectividad en el desarrollo de las tareas

Usuario	Tarea 1 (T <sub>1</sub> )	Tarea 2 (T <sub>2</sub> )	Tarea 3 (T <sub>3</sub> )	Tarea 4 (T <sub>4</sub> )
1	Sí	Sí	No	Sí
2	No	No	No	No
3	Sí	Sí	Sí	No
4	Sí	No	Sí	Sí
5	Sí	Sí	Sí	No
6	Sí	Sí	No	Sí
7	Sí	No	Sí	Sí
8	Sí	Sí	No	No
9	Sí	Sí	Sí	Sí
10	Sí	Sí	Sí	No

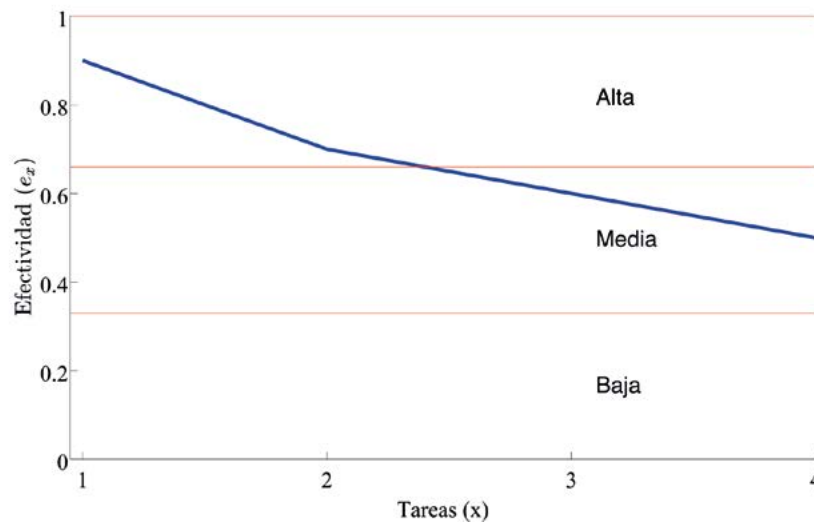


Figura 3: Efectividad en el cumplimiento de las tareas  $T_1$  a  $T_4$

### Prueba de eficiencia

Consiste en comparar los tiempos de desarrollo de las tareas  $T_1$  a  $T_4$  obtenidos tanto con el IDE Arduino como con el IDE CodeVision AVR. En CodeVision los usuarios programan los retos directamente sobre el microcontrolador de la placa. En la parte izquierda de la Figura 4 el tiempo promedio de las tareas  $\bar{t}_x$  es calculado para cada IDE usando la Ecuación 2; el radio de los tiempos de desarrollo  $rt_x$  se muestra en la parte derecha.

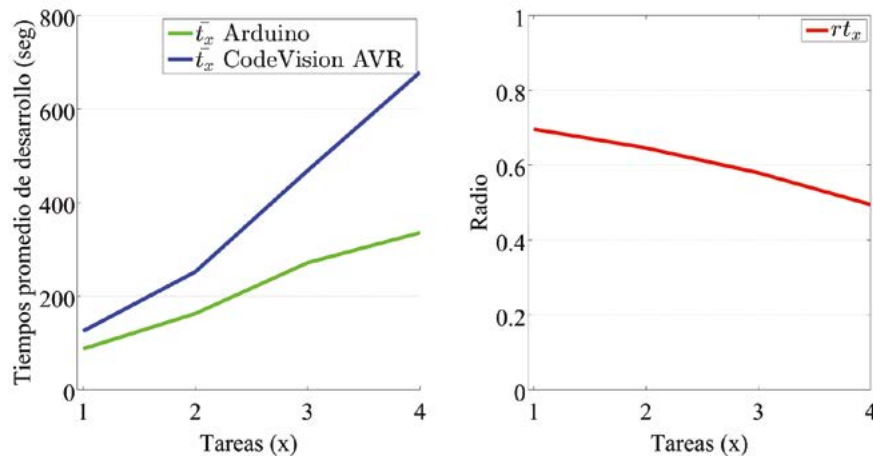


Figura 4: Tiempos promedio de desarrollo (izquierda) y radio de los tiempos de desarrollo (derecha), para las tareas  $T_1$  a  $T_4$

### Prueba de satisfacción

Los resultados de la encuesta de escala de usabilidad se pueden observar en la Tabla 4; se han calculado el valor medio y la desviación estándar a través de las Ecuaciones 3 y 4.

Tabla 4:  
Escala de usabilidad

Pregunta ( $q$ )	Valor medio ( $\bar{y}_q$ )	Desviación estándar ( $\bar{S}_q$ )
1	$\bar{y}_1=4.5$	$\bar{S}_1=0.67$
2	$\bar{y}_2=1.8$	$\bar{S}_2=0.74$
3	$\bar{y}_3=4.6$	$\bar{S}_3=0.4$
4	$\bar{y}_4=1.5$	$\bar{S}_4=0.80$
5	$\bar{y}_5=4.1$	$\bar{S}_5=0.94$
6	$\bar{y}_6=1.5$	$\bar{S}_6=1.20$
7	$\bar{y}_7=4.3$	$\bar{S}_7=0.90$
8	$\bar{y}_8=1.9$	$\bar{S}_8=0.94$
9	$\bar{y}_9=4.5$	$\bar{S}_9=0.92$
10	$\bar{y}_{10}=2.0$	$\bar{S}_{10}=1.61$

Por otro lado, los datos obtenidos de la encuesta de comodidad/facilidad de uso se encuentran en la Tabla 5. También se han calculado valores medios y desviaciones estándar y se realiza una apreciación final para cada pregunta.

Tabla 5:  
 Comodidad/facilidad de uso

Pregunta ( $q$ )	Valor medio ( $\bar{y}_q$ )	Desviación estándar ( $\bar{S}_q$ )	Apreciación final del IDE
1	$\bar{y}_1=4.5$	$\bar{S}_1=0.67$	Requiere poco esfuerzo
2	$\bar{y}_2=4.0$	$\bar{S}_2=0.89$	Rápido
3	$\bar{y}_3=4.2$	$\bar{S}_3=0.97$	Cómodo
4	$\bar{y}_4=4.8$	$\bar{S}_4=0.40$	ácil de usar

### Discusión final

La Figura 3 muestra que la efectividad del voluntario al usar el IDE Arduino disminuye desde 0.9 hasta 0.5 conforme la complejidad de las tareas aumenta. Teniendo en cuenta que los usuarios evaluados son inexpertos en el manejo del IDE, es previsible que existan fallos y que no se alcance una efectividad de 1 incluso en la tarea de menor dificultad.

En la Figura 4 se observan los resultados de la evaluación de eficiencia. La curva de radio de tiempo de desarrollo permanece siempre menor que 1, indicando que los tiempos con Arduino son menores que con CodeVision AVR. Con el descenso desde 0.7 hasta 0.5, se evidencia un incremento en la eficiencia conforme la dificultad de las tareas aumenta. Esto indica que el IDE Arduino permite que los programas de mayor complejidad se puedan desarrollar de forma más eficiente.

Si bien el atributo de satisfacción permite identificar percepciones y sentimientos subjetivos de los usuarios en relación al IDE, los valores medios y las desviaciones estándar muestran una apreciación cuantitativa de ellos. En las Tablas 4 y 5 se observan directamente las calificaciones de cada pregunta a través de los valores medios  $\bar{y}_q$ , mientras que las dispersiones en las respuestas se muestran mediante las desviaciones estándar  $\bar{S}_q$ .

La Tabla 4 evidencia una valoración global de satisfacción positiva del usuario al interactuar con el IDE Arduino debido a los favorables valores medios encontrados. Sin embargo las desviaciones estándar relativamente grandes muestran que los usuarios tuvieron respuestas dispersas dentro de la escala de calificación.

La Tabla 5 muestra que los usuarios se sienten cómodos al utilizar el IDE y que además lo consideran fácil de usar. Como las desviaciones estándar muestran valores relativamente bajos, se evidencia concordancia en las apreciaciones de todos los voluntarios evaluados.

En general, las medidas subjetivas de las pruebas de satisfacción soportan a las pruebas de eficiencia y efectividad. A pesar de la existencia de resultados favorables en cuanto a escala de usabilidad y comodidad/facilidad de uso, las desviaciones estándar corroboran el desacuerdo de los usuarios que no cumplieron las actividades satisfactoriamente.

Los resultados no pueden ser contrastados debido a que en la literatura no se han encontrado evaluaciones de usabilidad que consideren criterios de efectividad y eficiencia para IDEs usados en la programación de OSHW.

Queda planteado el problema de considerar evaluaciones de usabilidad para tareas que contemplen el manejo de opciones avanzadas de HW, tales como interrupciones. Además, se podrían realizar evaluaciones complementarias de desempeño en función del tiempo de compilación, tamaño del archivo binario generado, tiempos de ejecución de los programas creados, entre otros.

### Conclusiones

En este artículo se ha presentado una metodología para análisis de usabilidad de IDEs para OSHW que además fue comprobada en la plataforma Arduino; se evaluaron los atributos de efectividad, eficiencia y satisfacción del usuario. Al comparar el IDE Arduino con el IDE CodeVision AVR, se demostró que el primero es más eficiente que el segundo. Con Arduino, conforme la complejidad de los programas aumenta, la efectividad de los usuarios disminuye pero a la vez la eficiencia del IDE se incrementa. Hablando de satisfacción, existe poco esfuerzo mental para el desarrollo de tareas con Arduino, se tiene un IDE rápido, confortable y fácil de usar. Además, trabajos futuros se podrían centrar en aplicar la presente metodología de análisis de usabilidad a otras plataformas de desarrollo de sistemas embebidos, así como también se podrían proponer nuevas métricas.

### Recomendaciones

En base a los resultados obtenidos en este estudio se podrían realizar cambios en el IDE Arduino para mejorar su usabilidad; se considera lo enunciado por Lakhani (2015) donde se argumenta que todo sistema de código abierto se encuentra bajo depuración continua ya que cualquier usuario tiene la información necesaria para modificarlo.

### Referencias bibliográficas

- Gibb, A. (2015). *Building Open Source Hardware: DIY Manufacturing for Hackers and Makers*. New Jersey: Pearson Education Inc.
- Atmel Corporation (2016a). *Atmel AVR Toolchain for Linux* [Software]. Recuperado de <http://www.atmel.com/tools/atmelavrtoolchainforlinux.aspx>
- HP InfoTech. (2016). *CodeVisionAVR 3.29 Integrated Development Environment for the 8-bit Atmel AVR and XMEGA Microcontrollers* [Software]. Recuperado de [http://www.hpinfotech.ro/cvavr\\_features.html](http://www.hpinfotech.ro/cvavr_features.html)
- Arduino Company. (2016). *Arduino IDE 1.6.13* [Software]. Recuperado de <https://www.arduino.cc>
- Mascheroni, M., Greiner, C., Dapozo, G., & Estayno, M. (2013a). Ingeniería de Usabilidad: una Propuesta Tecnológica para Contribuir a la Evaluación de la Usabilidad de Software. *Revista Latinoamericana de Ingeniería de Software*, 125-134. Recuperado de <http://sistemas.unla.edu.ar/sistemas/redisla/ReLAIS/relais-v1-n4-p-125-134.pdf>
- Bacíková, M., Maricák, M., & Vancík, M. (2015). Usability of a Domain-Specific Language for a Gesture-Driven IDE. *Federated Conference on Computer Science and Information Systems (FedCSIS)*, 909-914. doi:<https://doi.org/10.15439/2015F274>
- Mascheroni, M., Greiner, C., Dapozo, G., & Estayno, M. (2013b). *Automatización de la Evaluación de la Usabilidad del Software*. XV Workshop de Investigadores en Ciencias de la Computación, 572-575. Recuperado de [http://sedici.unlp.edu.ar/bitstream/handle/10915/27243/Documento\\_completo.pdf?sequence=1](http://sedici.unlp.edu.ar/bitstream/handle/10915/27243/Documento_completo.pdf?sequence=1)
- Yago, T., López, A., Portilla, J., & Riesgo, T. (2015). A Serial Port Based Debugging Tool to Improve

- Learning with Arduino. *Conference on Design of Circuits and Integrated Systems (DCIS)*, 1-4. doi:<https://doi.org/10.1109/DCIS.2015.7388612>
- Oxer, J., & Blemings, H. (2009). *Practical Arduino: Cool Projects for Open Source Hardware*. New York: Springer.
- Soriano, A., Marín, L., Vallés, M., & Albertos, P. (2014). Low Cost Platform for Automatic Control Education Based on Open Hardware. *19th IFAC World Congress*, 47(3), 9044-9050. doi:<http://dx.doi.org/10.3182/20140824-6-ZA-1003.01909>
- Plattner, H., Meinel, C., & Leifer, L. (2016). *Design Thinking Research: Taking Breakthrough Innovation Home*. Springer.
- Karvinen, K., & Karvinen, T. (2011). *Make: Arduino Bots and Gadgets*. O'Reilly Media Inc.
- Grover, R., Krishnan, S., Shoup, T., & Khanbaghi, M. (2014). A Competition-Based Approach for Undergraduate Mechatronics Education Using the Arduino Platform. *4th Interdisciplinary Engineering Design Education Conference (IEDEC)*, 78-83. doi:<https://doi.org/10.1109/IEDEC.2014.6784685>
- Mazumder, F., & Das, U. (2015). Usability Guidelines for Usable User Interface. *IJRET: International Journal of Research in Engineering and Technology*, 3(9), 79-82. Recuperado de <http://esatjournals.net/ijret/2014v03/i09/IJRET20140309011.pdf>
- Sanz, J. (1997). Las Normas Técnicas ISO 9241 y EN 29241 sobre Pantallas de Visualización. *Boletín Factores Humanos*, 44-64. Recuperado de [http://www.insht.es/InshtWeb/Contenidos/Documentacion/TextosOnline/Divulgacion\\_Normativa/Ficheros/FDN\\_1.pdf](http://www.insht.es/InshtWeb/Contenidos/Documentacion/TextosOnline/Divulgacion_Normativa/Ficheros/FDN_1.pdf)
- Preece, J., Rogers, Y., & Sharp, H. (2002). *Interaction Design Beyond Human-Computer Interaction. 4th Edition*. New York: John Wiley & Sons.
- Manresa-Yee, C., Ponsa, P., Varona, J., & Perales, F. (2010). User experience to improve the usability of a vision-based interface. *Interacting with Computers, Special Issue on Inclusion and Interaction: Designing Interaction for Inclusive Populations*, 22(6), 594-605. doi:<https://doi.org/10.1016/j.intcom.2010.06.004>
- Arduino UNO revisión 3 [Hardware]. (2016). Ivrea, Italia: Arduino Company.
- Lakhani, K., & Von Hippel, E. (2003). How Open Source Software Works: Free User-to-User Assistance. *Elsevier Research Policy*, 32(6), 923-943. doi: [http://dx.doi.org/10.1016/S0048-7333\(02\)00095-1](http://dx.doi.org/10.1016/S0048-7333(02)00095-1)