

# FEETINGS: Un Marco para la Sostenibilidad del Software

Javier Mancebo<sup>1</sup>, Félix García<sup>1</sup>, Coral Calero<sup>1</sup>,  
José A. García-Berná<sup>2</sup>, José L. Fernández-Alemán<sup>2</sup>, Ambrosio Toval<sup>2</sup>

<sup>1</sup> Instituto de Tecnologías y Sistemas de Información, Universidad de Castilla-La Mancha,  
Ciudad Real, España

<sup>2</sup> Departamento de Informática y Sistemas, Universidad de Murcia, Murcia, España

{Javier.Mancebo | Felix.Garcia | Coral.Calero}@uclm.es  
{JoseAlberto.Garcial | Aleman | AToval}@um.es

**Resumen.** El desarrollo del software no debe permanecer indiferente a la necesidad de construir productos que sean sostenibles y respetuosos con el medioambiente a lo largo de su ciclo de vida. Sin embargo, uno de los principales problemas actuales, es la falta de herramientas que permitan medir el consumo de energía cuando un software es ejecutado, y detectar, por ejemplo, cuáles son las partes del software que tienen un consumo de energía excesivo. Por esta razón, se ha propuesto FEETINGS (*Framework for Energy Efficiency Testing to Improve Environmental Goals of the Software*), un marco que permite medir la eficiencia energética del software y así mejorar la sostenibilidad del mismo. En este trabajo, nos hemos centrado en el núcleo del marco, EET (*Energy Efficiency Tester*). EET es un dispositivo hardware de medición dedicado a recopilar los datos de consumo específicos del software que se está evaluando. A lo largo del documento se presenta las principales funciones de EET, y un caso de estudio usando el dispositivo de medición EET, donde se pretende observar si existe una correlación entre los requisitos de usabilidad de un software determinado con el consumo de energía que conlleva al ser ejecutado.

**Palabras clave:** Green software, Software sostenible, Medición del consumo del software, Dispositivo de medición, PHR

## 1 Introducción

La sociedad es cada vez más consciente de la importancia de preservar el medioambiente. Por ello, la sostenibilidad está ganando relevancia en casi todos los aspectos de la vida cotidiana. Entre los aspectos en los que la sostenibilidad ha cobrado gran importancia está la tecnología. Hasta hace relativamente poco, la gran mayoría de los avances se han centrado en el ahorro de energía del hardware. Sin embargo, en los últimos años esta tendencia ha ido cambiando y ya han surgido investigaciones relacionadas con el área del *green* software, enfocadas en mejorar la eficiencia del software, dando lugar a un software más sostenible y respetuoso con el medioambiente. Dick y Naumann [1] definen el software sostenible como aquel software, que, en su desarrollo,

despliegue y/o uso, tiene un impacto negativo mínimo o incluso un efecto positivo en la economía, la sociedad y el medioambiente.

Calero y Piattini [2] identifican tres dimensiones en la sostenibilidad software, según los recursos utilizados:

- Sostenibilidad humana: cómo el software afecta los aspectos sociológicos y psicológicos de las personas.
- Sostenibilidad económica: cómo el ciclo de vida del software protege las inversiones de los interesados.
- Sostenibilidad ambiental: cómo el desarrollo, mantenimiento y uso de los productos software afecta al consumo de energía y otros recursos.

La dimensión de sostenibilidad ambiental, también conocida como Green Software y que es en la que se centra este trabajo, promueve una mejora de la eficiencia energética del software, minimizando su impacto medioambiental y pudiendo provocar un impacto positivo en las otras dos dimensiones [3].

Además, el Green Software se puede dividir en *Green BY* software y *Green IN* software dependiendo del papel que el software juegue. Así, en [4] se define el *Green BY software* como el software que es desarrollado para dominios que trabajan en preservar la sostenibilidad del medioambiente, es decir, el software sirve como herramienta para respaldar los objetivos de sostenibilidad. Por otro lado, los mismos autores relacionan el *Green IN software* con cómo hacer que el software sea más sostenible, dando además como resultado un producto software más respetuoso con el medioambiente.

Según los resultados mostrados en [2, 5] la mayoría de trabajos realizados hasta ahora se han centrado en el *Green BY*, aunque recientemente los investigadores han comenzado a centrarse en el *Green IN*. Por otra parte, centrándonos de manera específica en el *Green IN Software*, las áreas del SWEBOK donde hay más investigaciones relacionadas con la sostenibilidad son: calidad, diseño y construcción de los productos software, seguida de los requisitos. En el lado opuesto, las áreas en las que menos esfuerzos de investigación se ha dedicado son: gestión, pruebas, métodos y proceso.

Una de las formas de seguir avanzando en el estudio del *Green IN* y poder evaluar si un producto software es sostenible o, al menos, es más sostenible que otro, es realizando mediciones de dicho software [6]. Las mediciones son un buen método para comprender, controlar, predecir y probar el desarrollo y el mantenimiento del software [7]. Dichas mediciones pueden ser utilizadas, por ejemplo, para identificar aquellas partes del software en ejecución que requieren más necesidades energéticas y esta información puede servirnos para elaborar un plan que permita reducir el consumo energético y mejorar la sostenibilidad de dicho software [8].

Así pues, nuestro trabajo se centra en el *Green IN Software* y pretende usar la medición del consumo de un software con el objetivo final de mejorar su sostenibilidad. Teniendo todo esto en cuenta, hemos definido FEETINGS (*Framework for Energy Efficiency Testing to Improve eNviromental Goals of the Software*) cuyo objetivo es medir la energía consumida por un producto software cuando es ejecutado en un PC, recopilar

esos datos de consumo y proporcionar una visualización adecuada de dicha información.

El resto del documento se estructura de la siguiente manera: en la sección 2 se presentará FEETINGS en detalle. La sección 3 presenta un caso de estudio llevado a cabo, donde se realizarán mediciones de un software PHR (*Personal Health Record*, carpeta personal de salud), para determinar si existe algún tipo de correlación entre la sostenibilidad y los requisitos de usabilidad con los que haya sido diseñado el software; finalmente, algunas conclusiones y líneas de trabajo futuro se exponen en la sección 4.

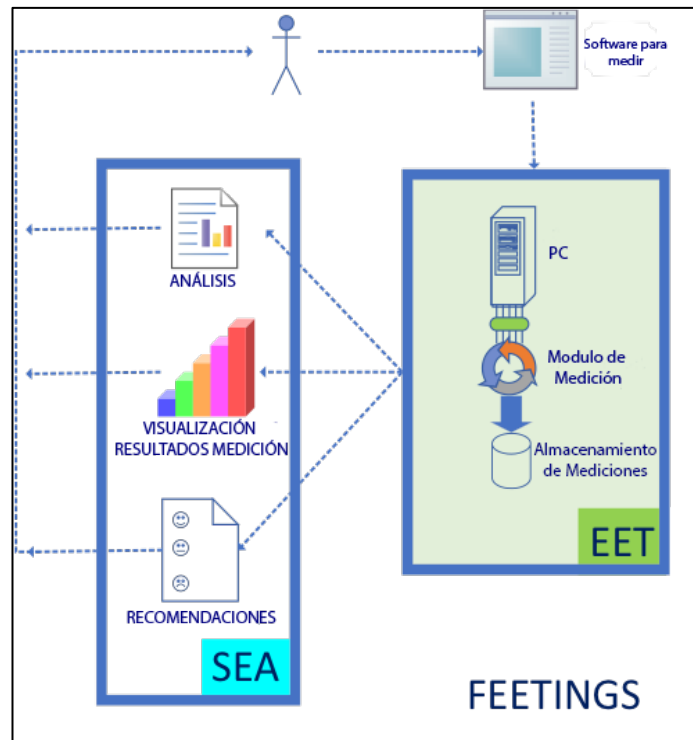
## 2 FEETINGS

Teniendo en cuenta lo mencionado en la anterior sección, se ha definido FEETINGS, cuyo objetivo es medir la energía consumida por un software y utilizar los resultados de esas mediciones para:

- Analizar el consumo del producto software.
- Conocer el comportamiento del software y sus diferentes versiones, para saber si estas versiones empeoran el consumo de energía del software o no.
- Identificar diferentes patrones de consumo que puedan servir de guía para la mejora de la eficiencia energética de los productos software
- Recomendar cambios en el software en un esfuerzo de mejorar su eficiencia energética.

Toda esta información podrá ser utilizada por los diferentes *stakeholders* del software bajo evaluación para conocer y/o mejorar su eficiencia energética. La **Fig. 1** muestra una visión general de FEETINGS. Como se puede observar FEETINGS se compone de dos elementos principales:

- El núcleo es EET (*Energy Efficiency Tester*), que es el dispositivo hardware que se ha construido para recolectar los datos del consumo de un producto software. EET ya está finalizado y está siendo utilizado para mediciones reales.
- La parte software, SEA (*Software Energy Assessment*), que se encarga de procesar los datos recopilados por EET, analizarlos y generar una visualización adecuada de los resultados (según su naturaleza), de modo que: (1) se puedan identificar los patrones de consumo; y (2) se genere una lista de recomendaciones que ayuden a los profesionales a mejorar la eficiencia energética del software medido. Actualmente, SEA está en pleno desarrollo.



**Fig. 1:** Visión general de FEETINGS

## 2.1 EET

Como ya hemos introducido, EET es el dispositivo hardware que hemos construido [9] y que permite medir el consumo de energía de una serie de componentes hardware utilizados por un producto software durante su ejecución.

EET admite la medición de tres componentes hardware diferentes: procesador, disco duro y tarjeta gráfica, de cualquier PC. También incluye dos sensores que proporcionan tanto el consumo total de energía del PC como el gasto energético del monitor conectado al equipo en el que se está ejecutando el software que se está evaluando. Una vez realizadas las mediciones, las mismas se almacenan en una memoria extraíble, para que puedan ser utilizadas para su análisis.

EET es portátil y fácil de conectar al PC donde se ejecuta el software que deseamos medir; únicamente es necesario conectar su fuente de alimentación al PC donde se ejecutará el software, reemplazando su fuente de alimentación.

La **Fig. 2** muestra EET, el dispositivo de medición. En la misma figura puede verse un ejemplo de los resultados de medición de una aplicación software.

El dispositivo de medición EET ha sido validado con el fin de demostrar la fiabilidad y estabilidad de sus mediciones. Para llevar a cabo la validación, se realizó un caso de prueba en el cual se ejecutaron seis algoritmos de ordenación implementados en tres lenguajes de programación diferentes (Java, C y Python) y sobre dos sistemas operativos diferentes (Microsoft Windows 7 y Linux Xubuntu). Tras un conjunto de ajustes, EET ha quedado validado y es un dispositivo que realiza mediciones estables y fiables.

A falta de concluir el desarrollo de SEA, los datos obtenidos de EET están siendo procesados de manera semiautomática. Los resultados de la información procesada están siendo utilizados para realizar análisis de consumo del software, y así, identificar qué tareas del software son las que consumen más energía y descubrir qué estructuras, algoritmos, patrones o arquitecturas particulares requieren un mayor gasto de energía. Esta información está siendo utilizada por los desarrolladores para la mejora de la eficiencia energética de su software.

A modo de ejemplo, en la siguiente sección se expone un caso de estudio que incluye la medición del consumo energético de un software PHR con el dispositivo EET, para realizar el análisis de los resultados obtenidos, y poder ver si existe relación entre el consumo energético y la usabilidad.

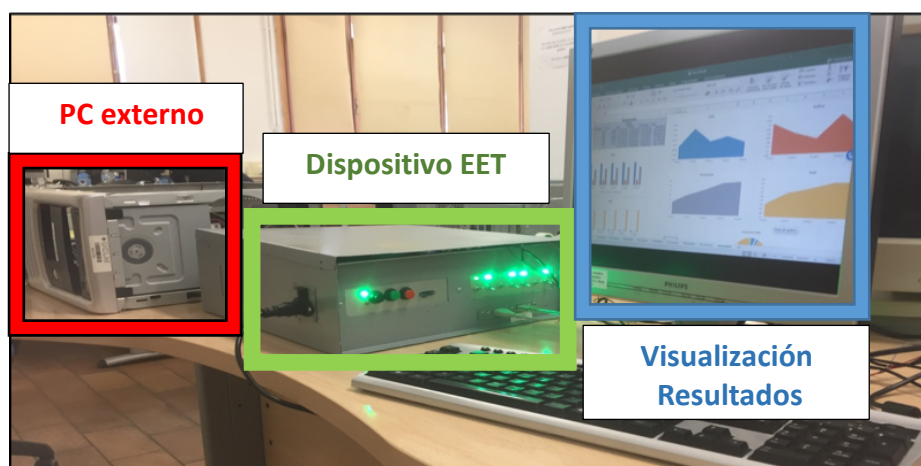


Fig. 2: Dispositivo de medición EET

### 3 Caso de estudio: análisis de PHR

#### 3.1 Presentación

Cada vez más la web y los servicios relacionados favorecen el desarrollo de las sociedades [10]. Poder consultar el correo electrónico, leer noticias o visitar las redes sociales son algunos ejemplos de las posibilidades que ofrece la tecnología actual para potenciar el crecimiento de las sociedades de la información. En general dentro del

ámbito de las TIC, y en el área médica en particular, están surgiendo aplicaciones informáticas para fomentar e involucrar a los pacientes en el cuidado de su salud. Es el caso de la Carpeta Personal de Salud, o en inglés Personal Health Record (PHR), que son herramientas electrónicas gestionadas por los propios pacientes, que permiten almacenar y consultar datos sanitarios en cualquier lugar y en cualquier momento con un dispositivo electrónico [11].

Con el paso de los años estas aplicaciones son más conocidas. De hecho se ha demostrado mediante estudios empíricos una creciente aceptación por parte de los ciudadanos en el uso de estas aplicaciones y actualmente se están integrando en los servicios de salud de las Comunidades Autónomas [12].

En función de sus características podemos encontrar diferentes clasificaciones de los PHR. Algunos son gratuitos, otros son de pago. Los hay disponibles en la web, en el móvil o a modo de aplicaciones de escritorio. También podemos encontrarlos especializados en un ámbito médico como por ejemplo el PHR Gestavida enfocado a embarazadas [13]. La actual tendencia es integrar el PHR con la historia clínica electrónica. Se distingue entre PHR versión *standalone*, que funciona de forma independiente o la versión *tethered* en los cuales existe interconexión con otros sistemas hospitalarios para la transferencia de la información automática. Un caso de éxito de una versión *tethered* es Kaiser Permanente. Este software de pago puede considerarse uno de los más reconocidos y utilizados con más de 11 millones de usuarios registrados a finales de 2017 [14].

El objetivo de este apartado es estudiar el gasto eficiente de la energía en estos sistemas y relacionar su consumo energético con la usabilidad de los mismos desde un enfoque de la sostenibilidad. Para ilustrar la investigación, nos centramos en un PHR con un elevado número de usuarios, y en el dispositivo EET con el cual se realizaron las mediciones del consumo energético de dicho PHR.

## 3.2 Metodología

### 3.2.1 Selección de PHRs

En esta sección se presenta el proceso que se ha llevado a cabo para seleccionar el PHR que será analizado en este caso de estudio (**Fig. 3**). En primer lugar se buscaron PHRs en bases de datos y en el portal myPHR.com. Esta búsqueda se realizó limitando los resultados a PHRs gratuitos, con acceso en la web y en la versión *standalone*. Los PHRs que habían dejado de estar operativos fueron descartados. A continuación se enumeran los PHRs seleccionados tras aplicar los anteriores criterios de inclusión: Doclopedia, Health Companion, Health Vault, Healthspek, Mi Via, My Chart Link, My Health Folders, My Health Vet, My Med Wall, No More Clipboard, Patient Power, Patients Like Me, Telemedical, WebMD, Zweena Health.

El siguiente criterio de inclusión que se examinó fue la popularidad de los portales web, de cara a seleccionar aquellos PHRs que suscitaran un mayor interés para el presente estudio. Para ello se utilizó la herramienta Alexa de Amazon [15]. Los PHRs que no tenían ninguna puntuación de popularidad fueron descartados.

Finalmente, se seleccionaron preliminarmente los siguientes PHRs —todos ellos gratuitos, y disponibles en la web. HealthVault (<https://international.healthvault.com>), que pertenece a Microsoft y está traducido a varios idiomas; HealthVet, desarrollado para veteranos de guerra del ejército de los Estados Unidos (<https://www.myhealth.va.gov>); PatientsLikeMe con una experiencia de usuario muy similar a la ofrecida en una red social (<https://www.patientslikeme.com>), NoMoreClipboard (<https://www.nomoreclipboard.com>) homologado para su uso ambulatorio y HealthCompanion (<https://www.healthcompanion.com>), con herramientas para administrar las finanzas de la salud.

El último criterio de inclusión adoptado fue la funcionalidad. Partiendo de las recomendaciones con respecto a la funcionalidad básica publicadas en myPHR.com [16] se optó finalmente por el PHR HealthVault, que es el más completo en lo que se refiere a tareas básicas de un paciente a la hora de registrar sus datos de salud [11]. A continuación se indican las funcionalidades básicas de los PHRs [16].

- Identificación personal
- Personas de contacto en caso de emergencia
- Nombre, dirección y teléfono de contacto de médicos especialistas
- Información del seguro médico
- Testamento de vida
- Autorización de donante de órganos
- Fechas de enfermedades padecidas o procedimientos quirúrgicos
- Medicamentos y dosis
- Vacunas
- Alergias
- Enfermedades hereditarias
- Resultados de exámenes médicos
- Opiniones de especialistas
- Resultados de pruebas importantes
- Conexión entre el usuario y los proveedores
- Materiales educativos relacionados con la salud
- Cualquier información relevante que deseen incluir los pacientes
- Información sobre prácticas dietéticas

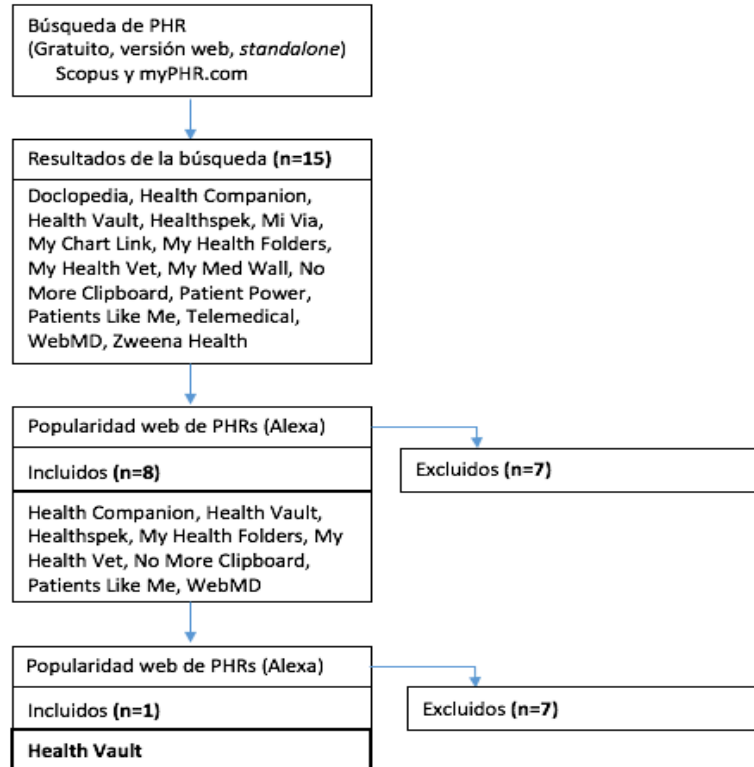


Fig. 3. Proceso de selección de PHR

### 3.2.2 Variables de estudio: consumo energético y usabilidad

A partir de las funcionalidades expuestas en myPHR.com se propusieron una lista de 19 tareas habituales para un paciente con las que analizar el consumo energético y la usabilidad del portal web seleccionado. Las tareas representativas de las funcionalidades básicas de los PHR se realizaron 5 veces, en un PC conectado al dispositivo de medición EET, de manera que se obtuvieron 5 mediciones de consumos de potencia de cada uno de los componentes principales del PC: disco, tarjeta gráfica, procesador, monitor y fuente de alimentación. Por otra parte, posteriormente se obtuvo una medida de la usabilidad por medio de una evaluación basada en modelos, dentro de la metodología del análisis de expertos [17]. Se evaluó la usabilidad del PHR Health Vault [13], utilizando los criterios de usabilidad definidos por el autor Alan Dix valorados en una escala Likert para cada una de las tareas. A continuación se muestran los criterios definidos:

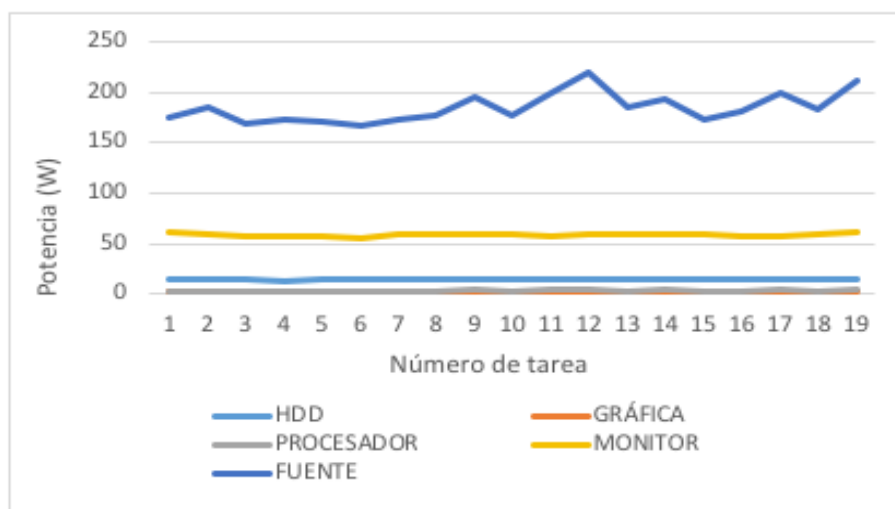
- Facilidad de aprendizaje
- Flexibilidad
- Consistencia
- Robustez
- Recuperabilidad



- Tiempo de respuesta
- Adecuación de las tareas
- Disminución de la carga cognitiva
- Predictibilidad
- Eficiencia
- Placer estético
- Claridad
- Compatibilidad
- Comprensibilidad

### 3.3 Resultados

A continuación se comentan los resultados que se obtuvieron tras la realización de las mediciones de consumo con EET, y la evaluación de la usabilidad. La **Fig. 4** muestra el consumo medio (de 5 mediciones) de los componentes del PC: disco, tarjeta gráfica, procesador, monitor y fuente de alimentación, durante el uso del PHR Health Vault en 19 tareas.



**Fig. 4.** Gráfica de consumos obtenidos

En primer lugar se estudió la correlación entre los consumos de potencia de los componentes del ordenador. Se utilizó el Test Shapiro-Wilk para estudiar la normalidad de la distribución de las muestras de los consumos de los 5 componentes hardware. Se observó que los datos de disco duro ( $W=0,310$ ), procesador ( $W=0,151$ ) y monitor ( $W=0,909$ ) habían sido extraídos de una población normalmente distribuida, mientras que la tarjeta gráfica ( $W=0,015$ ) y la fuente de alimentación ( $W=0,038$ ) no. Por tanto, la correlación de los datos se analizó con el coeficiente de Pearson para el disco duro,

procesador y monitor, mientras que se utilizó el coeficiente de Spearman para la tarjeta gráfica y la fuente de alimentación. Como se muestra en la Tabla 1, se obtuvieron 4 correlaciones significativas: (1) disco y tarjeta gráfica, (2) disco y monitor, (3) fuente de alimentación y procesador, y (4) monitor y procesador.

También se estudió la correlación entre el consumo de cada componente hardware (disco, gráfica, procesador, monitor y fuente de alimentación) y la medida de usabilidad obtenida a partir de los 14 criterios Dix para cada tarea. Los datos obtenidos de la evaluación de la usabilidad se extrajeron de una población normalmente distribuida ( $W=0.4323$ ). En la Tabla 1 se puede observar las correlaciones obtenidas.

**Tabla 1.** Matriz de correlaciones. En cursiva la correlación de Spearman.

Componente	HDD	<i>Gráfica</i>	Procesador	Monitor	<i>Fuente de Alimentación</i>	Crit. Dix
HDD	1					
<i>Tarjeta gráfica</i>	<i>0,604**</i>	<i>1</i>				
Procesador	0,344	<i>-0,011</i>	1			
Monitor	<i>0,714**</i>	<i>0,300</i>	<i>0,507*</i>	1		
<i>Fuente de alimentación</i>	<i>0,055</i>	<i>0,012</i>	<i>0,967**</i>	<i>0,425</i>	<i>1</i>	
Crit. Dix	-0.370	<i>0,119</i>	-0,026	-0,426	<i>-0,062</i>	1

\* La correlación es significativa en el nivel 0,05 (bilateral)

\*\* La correlación es significativa en el nivel 0,01 (bilateral)

En cuanto a la correlación entre la evaluación de usabilidad y los consumos de los componentes hardware del ordenador no se observó ninguna correlación significativa. No obstante, merece la pena destacar que existía una correlación negativa  $r=-0,426$  entre el consumo del monitor y la usabilidad, próxima a ser significativa ( $p=0.069$ ). Esto indica que una mayor usabilidad se asocia a un consumo energético del monitor menor. Sin embargo, estos datos hay que interpretarlos con cierta cautela ya que la evaluación de la usabilidad se realizó por parte de un único evaluador.

## 4 Conclusiones y trabajo futuro

El software cada vez juega un papel más importante en el consumo de energía global del PC. Para poder desarrollar software sostenible es necesario realizar mediciones del impacto energético que tiene cada software según el PC donde se ejecuta, y así identificar las partes del software que consumen más energía para poder mejorarlas, convirtiéndolo en un software más sostenible. Así pues, nuestro trabajo se ha centrado en desarrollar el dispositivo hardware de medición EET, que es el componente central de FEETINGS, un marco cuyo objetivo es evaluar el consumo de energía de un producto

software cuando es ejecutado, proporcionando una visualización adecuada de la información obtenida, de modo que nos permita recomendar cambios en el software analizado para mejorar su eficiencia energética.

EET es un dispositivo, que conectado a varios componentes del ordenador (por ejemplo, CPU, disco duro, etc.) donde se va a ejecutar el software a analizar, puede realizar un seguimiento tanto del consumo de energía como de temperatura del PC. EET tiene un papel muy importante en la mejora del consumo de energía, ofreciendo una medición continua del consumo del software ejecutado. Dicha información es analizada, para detectar partes o componentes del software con un consumo excesivo de energía. Esto permite a los investigadores saber si la ejecución del software implica un consumo excesivo en la CPU, en los componentes gráficos o en las unidades de almacenamiento masivo. Tal información conduce a un análisis muy preciso del software, marcando las partes particulares que causan un consumo tan alto, que puede ayudar a los desarrolladores a comprender los efectos del consumo de energía de sus decisiones y, por lo tanto, promover el desarrollo de software de eficiencia energética.

ETT ha sido probado en el caso de estudio de la sección 3, donde se ha usado para analizar la relación entre la usabilidad y el consumo energético del PHR Health Vault, obteniendo una correlación negativa entre el consumo del monitor y la usabilidad, con  $p$  cercano a 0.05, aunque no se obtuvo significación estadística. Con el objetivo de continuar con el estudio de la relación entre usabilidad y consumo energético de la Carpeta Persona de Salud se propone como línea futura de investigación ampliar el estudio a un mayor número de PHRs, con un mayor número de evaluadores. De esta forma, la muestra sería más representativa y el análisis de los datos permitiría alcanzar unos hallazgos más concluyentes. Además, se pretende trabajar en la creación de un conjunto de patrones de requisitos de usabilidad y asociarles un consumo energético. Estos patrones se extraerían a partir del estudio de la usabilidad de los PHR con las diferentes tareas identificadas.

También se pretende continuar trabajando en el desarrollo del marco FEETINGS, buscando detectar cuáles son las mejores técnicas para analizar la información, ofreciendo oportunidades para mejorar el consumo de software y mejorar la sostenibilidad del software como resultado.

## **Agradecimientos**

Este trabajo forma parte de los proyectos GINSENG-UCLM (TIN2015-70259-C2-1-R) y GINSENG-UMU (TIN2015-70259-C2-2-R), financiado con el apoyo del Ministerio de Economía, Industria y Competitividad (MINECO) y el Fondo Europeo de Desarrollo Regional (FEDER), y el proyecto SOFTWARE SUSTAINABILITY (SOS-SBPLY/17/180501/000364) financiado por la Consejería de Educación, Cultura y Deportes de la Dirección General de Universidades, Investigación e Innovación de la JCCM.

## Referencias

1. Dick, M. and S. Naumann. *Enhancing software engineering processes towards sustainable software product design*. in *EnviroInfo 2010, Integration of Environmental Information in Europe, Proceedings of the 24th International Conference on Informatics for Environmental Protection*. (2010).
2. Calero, C. and M. Piattini, *Puzzling out Software Sustainability*. *Sustainable Computing: Informatics and Systems*, **16**: p. 117-124 (2017).
3. Penzenstadler, B., et al. *Systematic mapping study on software engineering for sustainability (SE4S)*. in *Proceedings of the 18th International Conference on Evaluation and Assessment in Software Engineering*. ACM (2014).
4. Calero, C. and M. Piattini, *Introduction to green in software engineering*, in *Green in Software Engineering*. Springer. p. 3-27 (2015).
5. Penzenstadler, B. and H. Femmer, *A generic model for sustainability with process- and product-specific instances*, in *Proceedings of the 2013 workshop on Green in/by software engineering*. ACM: Fukuoka, Japan. p. 3-8 (2013).
6. Moraga, M.Á. and M.F. Bertoa, *Green software measurement*, in *Green in Software Engineering*. Springer. p. 261-282 (2015).
7. Briand, L.C., S. Morasca, and V.R. Basili, *Property-based software engineering measurement*. *IEEE transactions on software Engineering*, **22**(1): p. 68-86 (1996).
8. Bozzelli, P., Q. Gu, and P. Lago, *A systematic literature review on green software metrics*. VU University, Amsterdam, (2013).
9. Mancebo, J., et al., *Aparato para medición del consumo eléctrico de equipos informáticos (PC)*, UCLM, Editor.: Spain (2017).
10. Torrente, M.C.S., et al., *Sirius: A heuristic-based framework for measuring web usability adapted to the type of website*. *Journal of Systems and Software*, **86**(3): p. 649-663 (2013).
11. Fernández-Alemán, J.L., et al., *Free web-based personal health records: An analysis of functionality*. *Journal of medical systems*, **37**(6): p. 9990 (2013).
12. Alemán, J.L.F., I. Hernández, and A.B.S. García, *Encuesta de opinión sobre el uso de historias personales de salud en la Región de Murcia*. *Gaceta Sanitaria*, **27**(5): p. 454-458 (2013).
13. Bachiri, M., et al., *Mobile personal health records for pregnancy monitoring functionalities: Analysis and potential*. *Computer methods and programs in biomedicine*, **134**: p. 121-135 (2016).
14. KaiserPermanente, *Kaiser Permanente at a Glance*. (2018).
15. Amazon. *Alexa*. 2018; Available from: <https://www.alexametrics.com/siteinfo>.
16. AHIMA. *myPHR.com*. 2018; Available from: [http://www.myphr.com/StartaPHR/Create\\_a\\_PHR.aspx](http://www.myphr.com/StartaPHR/Create_a_PHR.aspx).
17. Dix, A., et al., *Human-Computer Interaction (3rd Edition)*. Prentice-Hall, Inc. (2003).