

FRAMEWORK PARA LA EVALUACIÓN DE LA CALIDAD EN USO DE SERVICIOS MÓVILES

Iván Pretel García, Ana Belén Lago Vilariño
ipretel@tecnologico.deusto.es, ablago@tecnologico.deusto.es
MoreLab – Mobility Research Lab, Fundación Deusto, España

ABSTRACT: En un mundo cada vez más competitivo, en el que el grado de exigencia y formación del usuario crece a medida que aumenta la oferta, las empresas de telecomunicaciones afrontan grandes desafíos en el proceso de puesta en el mercado de sus productos. Para lograrlo, las empresas son cada vez más conscientes de que no sólo hay que evaluar la calidad interna y externa del producto sino que también es necesario evaluar la experiencia del usuario o calidad en uso frente al producto. En este artículo se propone un framework para la medida y evaluación de la calidad en uso basada en el estándar ISO/IEC 9126-4. La evaluación se basa en cuatro características: efectividad, productividad, seguridad y satisfacción. Se pretende demostrar que es posible agilizar la evaluación de la calidad en uso de servicios móviles mediante la automatización de la captura y análisis de la calidad en uso.

Keywords: calidad en uso, experiencia del usuario, evaluación, servicios móviles, ISO/IEC 9126

1. INTRODUCCIÓN

Las tecnologías móviles están jugando un papel fundamental en la nueva Sociedad de la Información y del Conocimiento. Su auténtico valor radica en la posibilidad que ofrecen de acceso inmediato e instantáneo a servicios, comunicación y contenidos, de una manera cómoda y segura, desde cualquier lugar, sea fijo o móvil. Este valor se traduce finalmente en una mejora de la productividad para empresas y organizaciones de todo tipo y en la mejora de la calidad de vida de las personas en general.

Fruto de ello, la demanda de los consumidores de nuevos servicios móviles ha crecido de manera extraordinaria haciendo que operadoras de telecomunicaciones y los proveedores de servicios móviles deban dar una respuesta inmediata a dicha demanda para poder hacer frente a una competencia cada vez más dura.

A pesar de que las empresas desarrolladoras de servicios son conscientes de la necesidad de evaluar la calidad del servicio antes de su lanzamiento al mercado, debido a los plazos y a su coste, la evaluación de la calidad se ve muy reducida. En este artículo se plantea un framework que permitirá agilizar la evaluación de la calidad del servicio móvil reduciendo los tiempos y costes.

Concretamente, el framework evaluará la calidad desde la perspectiva de la experiencia del usuario, también denominada, calidad en uso. Ésta es un área de investigación importante en el estudio de Interacción Hombre-PC (HCI). Además, los recientes estándares de la ISO reflejan la importancia de tener en cuenta la experiencia del usuario. En concreto, el estándar ISO 9126-1 [1] representa la calidad de un software de producto desde tres perspectivas complementarias: la calidad interna, la calidad externa y la calidad en uso, siendo este último el aspecto más reciente.

El artículo está organizado como sigue. En la próxima sección se definirá la calidad en uso y los diferentes

elementos a evaluar dentro de la misma. En la Sección 3 se describe el framework. En la Sección 4 a su vez se presenta el capturador de la experiencia de usuario. Por último, se extraerán conclusiones sobre el uso del framework y se plantearán una serie de trabajos futuros.

2. CALIDAD EN USO

El estándar ISO 9126-1[2] define un modelo de calidad mediante tres aspectos: calidad interna, calidad externa y calidad en uso.

La calidad interna es medida y evaluada en base a atributos internos del software que pueden ser mejorados durante la implementación del código, la revisión y las pruebas.

La calidad externa es la calidad del software expresada a través de su comportamiento cuando éste se encuentra en ejecución, que normalmente es medida y evaluada durante las pruebas del software en un entorno simulado.

La calidad en uso constituye un concepto nuevo y de suma importancia que ha sido introducido en este nuevo marco de calidad. Representa la calidad percibida por el usuario en el software cuando éste es usado en un entorno y en un contexto específico. De esta forma, mide el grado con el que el producto software es capaz de cumplir con las expectativas y objetivos de los usuarios en un entorno particular de uso (en base a propiedades como la productividad), en lugar de medir las propiedades del software en sí mismo.

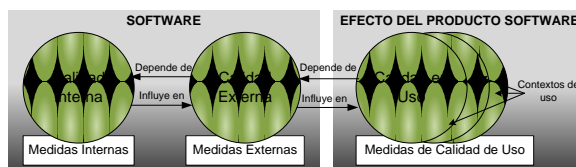


Fig. 1. Relaciones entre los distintos tipos de calidad

En la opinión de algunos expertos, como Bevan [3], el cambio fundamental que supone la calidad en uso es la atención cada vez mayor que merecen tanto los usuarios como el contexto considerado, cuando se evalúa el desempeño de un producto software en uso.

El estándar ISO/IEC 9126-1 define calidad en uso como “la capacidad de un producto de software de facilitar a usuarios específicos alcanzar metas específicas con eficacia, productividad, seguridad y satisfacción en un contexto específico de uso”.

Añade que “calidad en uso es la visión de calidad de los usuarios de un ambiente conteniendo software, y es medida sobre los resultados de usar el software en el ambiente, antes que sobre las propiedades del software en sí mismo”.

Las características de calidad en uso [4] son agrupadas en cuatro categorías que se muestran en la Figura 2.

CARACTERÍSTICA	DEFINICION
Efectividad	La capacidad del producto de software para facilitar a los usuarios alcanzar metas específicas con exactitud y completitud en un contexto específico de uso.
Productividad	La capacidad del producto de software para invertir la cantidad apropiada de recursos en relación a la eficacia alcanzada en un contexto específico de uso.
Seguridad	La capacidad del producto de software para alcanzar niveles aceptables de riesgo de dañar a las personas, el negocio, el software, la propiedad o el ambiente en un contexto específico de uso.
Satisfacción	La capacidad del producto de software para satisfacer a los usuarios en un contexto específico de uso. En el estándar ISO también definen satisfacción como la respuesta del usuario a la interacción con el producto, e incluye la actitud hacia el uso del producto.

Fig. 2. Definición de las características de calidad en uso

El estándar 9126-4 [4] define las métricas que dan soporte a la medición de la efectividad, productividad, seguridad y satisfacción.

2.1 Herramientas de medición de la calidad en uso

Durante el desarrollo del framework se analizaron diversas herramientas que pueden clasificarse en dos tipos: herramientas comerciales y herramientas desarrolladas en el ámbito de la investigación.

Existen diversas herramientas en el mercado (entre ellas, Morae [5], The Observer [6],...) que facilitan a los analizadores su trabajo de evaluar la calidad de un producto al ser manipulado por un usuario. La mayoría proporciona la posibilidad de grabar en vídeo las tareas

realizadas por el usuario y permite contestar a encuestas de satisfacción. Con respecto a la evaluación de servicios móviles, estas herramientas incorporan una cámara externa al móvil, permitiendo grabar las interacciones del usuario con el móvil. El problema radica en que no automatiza la evaluación, requiere que un experto analice posteriormente los vídeos, realice el cálculo de las métricas y emita un informe.

Con respecto a las herramientas desarrolladas en el ámbito de la investigación, destacan dos de ellas [7] [8] que se han basado en el estándar ISO/IEC 9126-4 y que evalúan la calidad en uso pero sólo de sitios web y no de servicios móviles.

El framework propuesto en este artículo pretende automatizar la captura, el cálculo de las métricas en la medida de lo posible y la emisión del informe final para servicios móviles. El evaluar servicios móviles presenta una problemática especial: la captura debe realizarse en el dispositivo móvil. Éste posee unos recursos de memoria y procesamiento limitados por lo que la información recogida debe ser mínima y el consumo reducido. Por otro lado, los dispositivos móviles poseen sistemas operativos muy variados siendo necesario que la herramienta de captura les dé soporte a todos ellos.

3. FRAMEWORK DE EVALUACIÓN DE LA CALIDAD EN USO

La metodología que se va a utilizar para la Gestión de la Experiencia de Usuario ha condicionado la arquitectura y la herramienta de captura de datos. La manera más objetiva y exacta posible para evaluar la experiencia de usuario de aplicaciones móviles es el dotar al usuario de un terminal con la aplicación real instalada, dejar que la utilice en un entorno real y lograr la captura de esa información sin influenciar ni condicionar al usuario o al entorno. Por ello, se debe ser capaz de registrar toda la información generada por el uso de la aplicación, evitando la influencia de la herramienta de captura de datos sobre la experiencia de usuario. Una vez generada dicha información se trata con una herramienta de análisis calculando las métricas y finalmente realizando la evaluación final.

El framework consiste en dos herramientas: la herramienta de captura y la herramienta de análisis. La herramienta de captura se instalará en el dispositivo móvil con la aplicación, capturará la información sobre la experiencia de usuario y posteriormente la herramienta de análisis generará los resultados de la evaluación con los datos obtenidos.

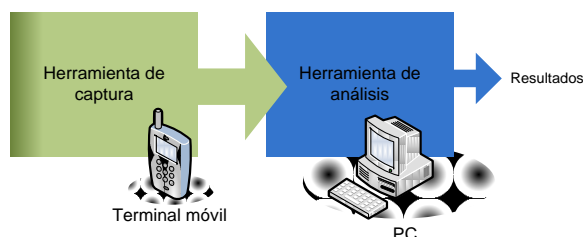


Fig. 3. Herramientas para la Gestión de la Experiencia de Usuario

4. HERRAMIENTA DE CAPTURA

Para realizar una captura y monitorización de las tareas a ejecutar y poder generar la información suficiente para evaluar la experiencia de usuario se ha implementado una aplicación denominada Ne-Monitor. El usuario ejecutará la aplicación en el terminal a la vez que se ejecuta el servicio móvil a evaluar y realizará las tareas encomendadas por la aplicación que actuará de guía. El usuario debe seleccionar una tarea y notificar a la aplicación que comienza a evaluarse. Mientras ésta se efectúa, la aplicación registrará la actividad del usuario y automáticamente le realizará tests al comienzo, en el transcurso, y al final de la tarea si los mismos son relevantes en esos momentos.

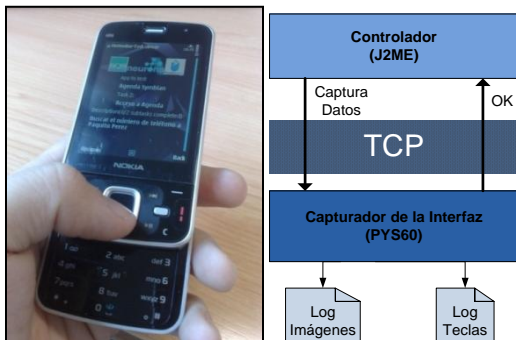


Fig. 4. Terminal ejecutando Ne-Monitor con la división de módulos

Consta de dos principales módulos que se comunican entre ellos con comandos enviados mediante TCP: el Controlador que ofrece la interfaz gráfica y el Capturador de la Interfaz que registra la información de las teclas pulsadas y realiza las capturas de pantalla.

4.1 Controlador

La implementación del controlador de Ne-Monitor se ha realizado mediante la tecnología Java ya que así permite el soporte multiplataforma al tener la mayoría de terminales su versión de la máquina virtual de Java. Es el módulo que dispone de toda la lógica de la aplicación ya que lee los ficheros que almacenan las tareas, los tests disponibles para las tareas, etc. Además, es el que controla cuándo se debe comenzar, pausar o finalizar una captura enviando comandos al Capturador de la Interfaz.

El formato de los ficheros utilizados donde se definen las tareas es XML. Se deben crear Ficheros de Tareas donde se almacena la información de las tareas y subtareas a realizar por el usuario junto con parámetros de cómo ha de ser la prueba (directorio donde se almacenan los datos de captura, tests a realizar en la prueba,...). Un ejemplo de fichero es el que se muestra en la Figura 5.

El propio módulo realiza los tests al usuario definidos en el Fichero de Tareas, pero dichos tests han de ser definidos en Ficheros de Test. En ellos se almacenan las preguntas necesarias con el formato que se observa en la Figura 6.

```
<?xml version="1.0" encoding="iso-8859-1" ?>
<tasks>
  <task id="1" name="Búsqueda de cines">
    <application>Nokia Maps</application>
    <desc>Localizar cines Zubiarte</desc>
    <subtask id="1.1" name="Loc 1" status="1">
      <desc>Situarse en Bilbao</desc>
      <timeavg>120</timeavg>
      <capturedata>
        <screenshotsdir>E:\</screenshotsdir>
        <logsdir>E:\...\logs</logsdir>
        <tests>
          <pretest>t1.xml</pretest>
          <middletest>t2.xml</middletest>
          <posttest>t3.xml</posttest>
        </tests>
      </capturedata>
    </subtask>
    <subtask id="1.2" name="Loc 2" status="1">
      <desc>Filtrar cines</desc>
      <timeavg>40</timeavg>
      <capturedata>
        <screenshotsdir>E:\</screenshotsdir>
        <logsdir>E:\...\logs</logsdir>
        <tests>
          <pretest>t1.xml</pretest>
          <middletest>t2.xml</middletest>
          <posttest>t3.xml</posttest>
        </tests>
      </capturedata>
    </subtask>
  </task>
</tasks>
```

Fig. 5. Ejemplo de fichero de tareas

Partiendo de estos datos de configuración el usuario inicia la aplicación Ne-Monitor donde puede visualizar en una lista todas las tareas que debe realizar con una aplicación concreta. Debe seleccionar una tarea a realizar y seleccionar la opción de Nueva Sesión para comenzar. Acto seguido debe hacer la tarea y cuando la termine volver a la aplicación Ne-Monitor y pulsar la acción correspondiente. De este modo las tareas se van realizando hasta tener todas completadas (Figura 7).

```
<?xml version="1.0" encoding="iso-8859-1" ?>
<test id="0932897">
  <question id="1" type="choose">
    <statement>¿Cuántas veces abre la App?</statement>
    <choose>
      <option>ninguna</option>
      <option>de 1 a 5 veces al día</option>
      <option>de 5 a 10 veces al día</option>
      <option>más de 10 veces al día</option>
    </choose>
  </question>
  <question id="2" type="choose">
    <statement>¿es fácil?</statement>
    <choose>
      <option>si</option>
      <option>no</option>
      <option>depende de la situacion</option>
    </choose>
  </question>
</test>
```

Fig. 6. Ejemplo de fichero de test

4.2 Capturador de la interfaz

Cuando el usuario decide comenzar, parar, continuar o terminar una tarea el módulo Controlador se comunica con el módulo Capturador de Interfaz para que éste haga la captura de pantallas y el registro de imágenes.

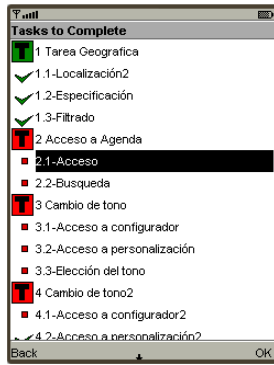


Fig. 7. Selección de la tarea de Ne-Monitor

Dicho capturador consiste en un pequeño módulo demonio en otro lenguaje más característico dependiendo del terminal ya que Java no permite el acceso a los eventos de los botones ni a la captura de pantalla cuando la aplicación no dispone del foco del sistema, lo cual es indispensable a la hora de ejecutar otra aplicación. En el caso de terminales con Symbian O.S. el capturador de la interfaz se ha implementado con PyS60, la versión móvil de Python. Dicho módulo externo se comunica mediante TCP con la aplicación Ne-Monitor y es el encargado de generar un Log donde se registran las teclas pulsadas con su tiempo exacto. Además, si se requiere también registra capturas de pantalla en el momento en el que se pulsa una tecla.

Los datos capturados de las teclas y el estatus del teléfono se registran como Logs en ficheros de texto y las capturas de pantalla como imágenes en formato JPEG. Es importante mencionar que el nombre de los ficheros de imagen está formado por el tiempo exacto en el que la tecla correspondiente a la misma ha sido pulsada junto con la tarea a la que pertenece. Los datos que se almacenan en el fichero de Log son la tecla pulsada, su tiempo exacto, nivel de batería y espacio de memoria RAM ocupada.

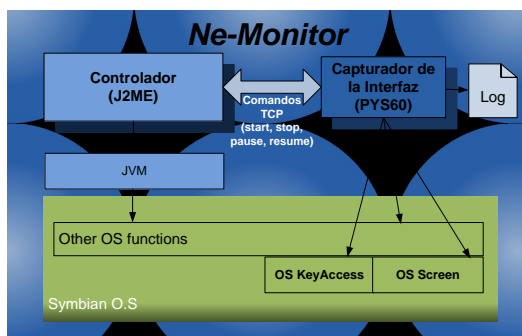


Fig. 8. Arquitectura general de la herramienta Ne-Monitor

5. CONCLUSIONES Y LÍNEAS FUTURAS

La evaluación de la calidad en uso es un aspecto que las empresas desarrolladoras de servicios móviles deben tener en cuenta en un mercado tan competitivo.

Para facilitar la evaluación de la calidad en uso de servicios móviles, se ha definido un framework compuesto de dos herramientas: una herramienta de captura que reside en el dispositivo móvil y una herramienta de análisis que reside en PC.

```

//[[TIEMPO]TAREA 2.1 COMIENZA
[1261128994.42][2.1]TASK STARTED
//[[TIEMPO] [TECLA] [BATERÍA] [MEMORIA]
[1261128996.72][53][0][23826432]
[1261128997.05][53][0][23822336]
[1261128997.4][53][0][23822336]
//[[TIEMPO]TAREA 2.1 PAUSADA
[1261128994.42][2.1]TASK PAUSED
//[[TIEMPO]TAREA 2.1 RETOMADA
[1271428556.16][2.1]TASK RESUMED
//[[TIEMPO] [TECLA] [BATERÍA] [MEMORIA]
[1271428556.66][53][0][23822336]
[1261128567.93][41][0][23822336]
[1261128568.07][33][0][23822336]
[1261128568.53][29][0][23822336]
//[[TIEMPO]TAREA 2.1 FINALIZADA
[1261128968.92][2.1]TASK ENDED

```

Fig. 9. Ejemplo de Log que genera el Capturador de la Interfaz

Los beneficios que proporciona el framework son: la rapidez al realizar una evaluación de la calidad en uso de servicios móviles, la reducción de errores en la captura de información de la evaluación y mayor precisión y exactitud en el cálculo de las métricas de la calidad en uso.

En este artículo se ha presentado el desarrollo de la herramienta de captura mostrando cómo se han logrado solventar las limitaciones impuestas por el dispositivo móvil.

Como trabajo futuro, se pretende desarrollar la herramienta de análisis y evaluar diversos servicios móviles con el framework para demostrar la fiabilidad del mismo.

6. AGRADECIMIENTOS

Este framework ha sido desarrollado dentro del proyecto "NEURONA: Interconexión de infraestructuras abiertas para la validación integral de servicios móviles convergentes de nueva generación" y financiado por el Ministerio de Industria, Turismo y Comercio a través del Subprograma Avanza I+D (TSI-020501-2008-181).

References

- [1] ISO/IEC 9126:2001. (2001), *Information technology - Software product evaluation - Quality characteristics and guidelines for their use*.
- [2] ISO/IEC 9126-1:2001. (2001), *Software Engineering—Software Product Quality—Part 1: Quality Model*.
- [3] Bevan, N. (1995), *Measuring usability as quality of use*. *Software Quality Journal*, 4: p. 115-150.
- [4] ISO/IEC 9126-4. (2004), *Software Engineering –Product Quality - Part 4: Quality In Use Metrics*.
- [5] Morae from Techsmith, <http://www.techsmith.com/morae.asp>.
- [6] The Observer from Noldus Information Technology, <http://www.noldus.com>.
- [7] Echartea, D., Moreno, M., González, G. (2008), *Evaluación de la calidad en uso de sitios web asistida por software: SW - AQUA*. *Revista Avances en Sistemas e Informática*, vol 5 No.1: p. 147-154.
- [8] Covella, G. J., Olsina, L. A. (2006), *Assessing quality in use in a consistent way*. In *Proceedings of the ACM 6th international Conference on Web Engineering* (Palo Alto, California, USA, July 11 - 14, 2006). *ICWE '06*, vol. 263: p. 1-8