

AmatiQ: adaptándose al contexto dinámicamente

Diego Ordóñez Camacho

Universidad Tecnológica Equinoccial, Quito, Ecuador, dordonez@ute.edu.ec

Andrea Loayza

Universidad Tecnológica Equinoccial, Quito, Ecuador, lvaf37504@ute.edu.ec

RESUMEN

Este trabajo presenta AmatiQ, una aplicación sensible al contexto, desarrollada para la plataforma Android, que es capaz de adaptar automáticamente su comportamiento dependiendo de las condiciones del entorno y de las preferencias anotadas por el usuario. AmatiQ utiliza los sensores del dispositivo para, mediante un set básico de desencadenadores, ejecutar acciones que por su naturaleza repetitiva o en algunos casos inclusive riesgosa, como al utilizarse mientras se conduce, el usuario desea dejar en manos del dispositivo. AmatiQ cumple con la automatización requerida, y muestra las posibilidades de integrar nueva funcionalidad que permita aumentar el nivel de autonomía del sistema.

Palabras clave: aplicaciones sensibles al contexto, dispositivos móviles, diseño y desarrollo de software

ABSTRACT

AmatiQ is a context-aware application for the Android operating system. AmatiQ adapts its behaviour automatically depending on two conditions: the environment conditions and a set of preferences defined by the user. The application uses the device sensors as triggers to automate a group of predefined actions that, because of being repetitive in nature or risky, like when driving, the user wishes to leave in charge of the phone. AmatiQ fulfills the stated automation requirements and shows the possibilities to integrate new functionalities to increase the system's autonomy.

Keywords: context-aware applications, mobile devices, software design and development

1. INTRODUCCIÓN

Las aplicaciones sensibles al contexto cuentan con la capacidad de adaptar su comportamiento, dependiendo de las circunstancias en las que operan, sean estas circunstancias físicas o situacionales. Esta adaptación, como principio, debe conllevar un mínimo de intervención por parte del usuario. Idealmente las aplicaciones sensibles al contexto deberían "desvanecerse en el background, para que el foco central sea el usuario y sus tareas, más que los dispositivos o las consideraciones técnicas" (Baldauf, Dustdar, & Rosenberg, 2007, p. 1).

AmatiQ se ha concebido como una aplicación para el sistema operativo Android. Está orientada a aquellos usuarios de smartphones que, dependiendo del lugar donde se encuentren, realizan con sus dispositivos determinadas acciones o procedimientos de manera rutinaria. Algunos ejemplos de estas acciones son ponerlo en silencio, reducir el volumen multimedia o desactivar el Wi-Fi. Estos procedimientos pueden convertirse en algo muy repetitivo y tedioso, pero sin embargo casi obligatorio. No realizarlos puede ocasionar efectos indeseados como por ejemplo un timbre en ocasiones inapropiadas, el agotamiento imprevisto de la batería, o una interferencia con equipo médico. El proyecto de investigación del cual nace AmatiQ desea proporcionar a los usuarios de smartphones mejores herramientas para gestionar el estado de sus teléfonos; herramientas que requieran menos de su atención y que puedan hacer más cosas de forma automática. El objetivo principal es entonces el de crear aplicaciones y servicios que monitoreen el estado del entorno y adapten su funcionamiento con respecto a él, siguiendo unas guías y preferencias definidas por cada usuario.

Este tipo de aplicaciones requiere especificidad en la definición de contexto, ya que al ser un término ambiguo que es utilizado en diversas áreas y circunstancias, puede tener interpretaciones muy variadas. Para este estudio consideramos apropiado definirlo como “el conjunto de objetos y características del entorno que al ser percibidas por elementos sensoriales, y puestas en relación con la situación particular de una entidad, dan lugar a una identidad claramente definible y reproducible” (Loayza, Proaño, & Ordóñez Camacho, 2013, p. 96). Esta definición puntualiza que nuestro contexto es un patrón extraído a partir de las particularidades del entorno, los sensores y el usuario. Este patrón debe ser susceptible de generalización, clasificación e incluso predicción.

Históricamente, tanto Uchiyamada (2007) como Baldauf et al., (2007) coinciden en que las aplicaciones dependientes del contexto han venido desarrollándose desde inicios de los años 90. Una de las primeras aplicaciones fue la aplicación llamada Active-Badge (Want, Hopper, Falcão, & Gibbons, 1992) que era un sistema de localización en entorno de oficina. A partir de ese momento se ha visto un incremento tanto en la investigación sobre el tema, como en el desarrollo de aplicaciones. Para una revisión del estado del arte puede ser conveniente revisar a Loayza et al., (2013). A continuación se presentan detalles específicos de la concepción y desarrollo de AmatiQ.

2. METODOLOGÍA

Se ha buscado la manera de que una aplicación pueda determinar las circunstancias del usuario o del contexto, para que esta información pueda ser utilizada con miras a cambiar el comportamiento del teléfono en formas más útiles. Un teléfono consciente del contexto podría cambiar automáticamente el perfil cuando el usuario entra en un restaurante o se sienta en el asiento del conductor de un auto, por ejemplo. En el caso de la gestión del teléfono, casi toda la complejidad radica en la detección y determinación del contexto. Una vez que se conoce el contexto, unas reglas muy simples pueden producir un comportamiento que parece inteligente (DeVaul et al., 2003).

El proceso de desarrollo de AmatiQ consideró una serie de principios y técnicas encadenados e interrelacionados jerárquicamente. En primer lugar, como principios guía, se tomó aquellos del diseño centrado en el usuario u operador humano (International Organization for Standardization, 2010), adoptados de acuerdo a la particularidad de la aplicación y considerando tres directrices de base.

Primera, comprender y especificar el contexto de uso, el cual es múltiple y depende en gran medida del estilo de vida del usuario. El entorno de uso en cuanto a equipo y sistema operativo, se restringe en este estudio a smartphones Android. En cuanto a conectividad, sensores y entorno físico y social puede variar sustancialmente.

Para enfrentarse a las dificultades inherentes al problema mencionado, las aplicaciones sensibles al contexto deben considerar cuatro aspectos fundamentales: qué es el contexto, cómo percibirlo, cómo representarlo y cómo adaptarse al mismo (Wei & Chan, 2007). En la Tabla 1 se puede observar el detalle que considera el tipo de entorno, principalmente interno (bajo techo) y externo; la periodicidad de ocurrencia del entorno, que puede ser constante (darse muy a menudo) y eventual; presentando finalmente el entorno específico o ubicación.

Tabla 1: Detalle de los principales entornos de uso a considerar (Fuente: elaboración propia)

Tipo	Periodicidad	Ubicación
Interno	Constante	Hogar
Interno	Constante	Oficina
Interno	Eventual	Reuniones (de trabajo)
Interno	Eventual	Diversiones (cine, teatro, ...)
Externo	Constante	Automovil (conducción)
Externo	Eventual	Paseos, viajes

La segunda directriz comprende definir las tareas a realizar por el usuario, y presenta un enfoque sustancialmente opuesto al tradicional, ya que lo que nos interesa no es tanto las tareas para las que el usuario va a utilizar la aplicación, sino aquellas tareas que el usuario desea dejar de realizar, y que espera que la aplicación administre y ejecute automáticamente. En esta línea es de particular interés el tratado de Milette y Stroud (2012) sobre cómo

utilizar, en el sistema Android, los sensores del dispositivo para generar aplicaciones que interactúen con el usuario y el entorno de manera efectiva.

La Tabla 2 presenta la definición de las tareas definidas a ser automatizadas en la primera etapa, indicando el tipo de acción a efectuarse y la condición de base o desencadenante del cual dependería la automatización.

Tabla 2: Definición de las tareas a automatizarse (Fuente: elaboración propia)

Categoría	Acción	Desencadenante
Volumen	Subir o bajar	Sonido
Luminosidad de la pantalla	Subir o bajar	Luz ambiente
Wifi	Conectar o desconectar	Ubicación
Bluetooth	Conectar o desconectar	Ubicación
GPS	Conectar o desconectar	Ubicación
Launcher	Cambiar	Circunstancia
Batería	Preservación o rendimiento	Uso

Finalmente, la tercera directiva adoptada habla de establecer las soluciones de diseño y el proceso de desarrollo e implementación, los cuales deben ser iterativos para permitir una retroalimentación temprana proporcionada por el usuario, la cual pueda incorporarse lo antes posible en la siguiente iteración. De Sá y Carriço (2011) han estudiado este problema, enfocándose sobre todo en cómo las aplicaciones móviles sensibles al contexto interactúan con el entorno y las interfaces de usuario. Una interesante conclusión que destacan es que tanto los prototipos como las técnicas de evaluación deben ser diseñados para utilizarse y probarse fuera del laboratorio, en condiciones reales de uso.

Con este fin se incorpora un proceso de prototipado en código o programado, adaptado del recomendado por Arnowitz, Arent y Berger (2010). La Tabla 3 presenta un resumen de los pasos principales que se consideraron en el bucle de prototipado utilizado.

Mediante el uso de los elementos propuestos se persigue que AmatiQ responda sobre todo a tres de los requerimientos planteados por Uchiyamada (2007): aumentar la satisfacción del usuario mediante la provisión de servicios más adaptados a sus características; automatizar funciones rutinarias mediante la definición de comportamientos basados en la información de contexto, la cual debe ser precisa y seleccionada en el momento y lugares apropiados; y finalmente, software menos molesto para el usuario al no interrumpirlo de manera innecesaria y repetitiva, mediante una mejor descripción de la situación del usuario, y una mayor personalización.

3. RESULTADOS

AmatiQ se encuentra, en esta etapa inicial de su desarrollo, organizada por desencadenante para facilitar el proceso de pruebas. Los desencadenantes considerados son aquellos definidos previamente en la metodología. A partir de los desencadenantes se definen los requisitos funcionales, que dan lugar no solo a la estructuración de las acciones sino también de las interfaces de configuración. En la Figura 1 se presenta la definición del requisito de base y su resultante interfaz gráfica. Esta sección se encuentra dividida por desencadenante, para poder presentar la aplicación en su conjunto.

Tabla 3: Proceso de prototipado (Fuente: adaptación de Arnowitz, et al., elaboración propia)

Etapa	Pasos	Detalles
Seleccionar escenarios de aplicación	<ul style="list-style-type: none"> • Verificación de supuestos y determinación de requerimientos • Elaboración de casos de uso • Definición de navegación y flujos de ejecución de las actividades 	A partir de los supuestos establecidos sobre el producto, se levantan y recopilan los requerimientos que permitirán elaborar los escenarios de uso relacionados con el entorno. Se determinan los escenarios relevantes a ser prototipados y se define su flujo.
Inventariar elementos de interfaz, datos y métodos	<ul style="list-style-type: none"> • Determinar las interfaces gráficas y sus elementos, que permitan soportar los escenarios de ejecución • Elaborar el diseño esquemático del prototipo • Determinar los métodos requeridos para proporcionar la funcionalidad esperada por las interfaces • Elaborar listas de control 	Diseño del prototipo a ser elaborado, creando el inventario de funcionalidad tanto a nivel gráfico como de datos y funcional.
Desarrollo e implementación del prototipo	<ul style="list-style-type: none"> • Implementación de un prototipo de alta fidelidad que permita su uso, simulado o real • Recolección e inclusión de los datos de test 	Implementación del prototipo y establecimiento del entorno de test con los datos requeridos para permitir el correcto llenado de las listas de control
Ejecución simulada	<ul style="list-style-type: none"> • Ejecución interna al equipo de desarrollo • Ejecución de tests • Identificación y mitigación de problemas 	Pruebas internas para pulir el prototipo y evitar problemas triviales, previo a someterlo a la consideración de los usuarios seleccionados
Revisión y prueba	<ul style="list-style-type: none"> • Revisión por parte de usuarios tipo • Recopilación de criterios • Validación de listas de control • Evaluación de siguiente etapa: iteración o versionado 	Revisión del prototipo por parte de los usuarios seleccionados. Se llenan las listas de control, convalidan los datos y se decide, basado en el análisis si es recomendable efectuar una nueva iteración o si se puede pasar a la etapa de generación de una nueva versión del sistema

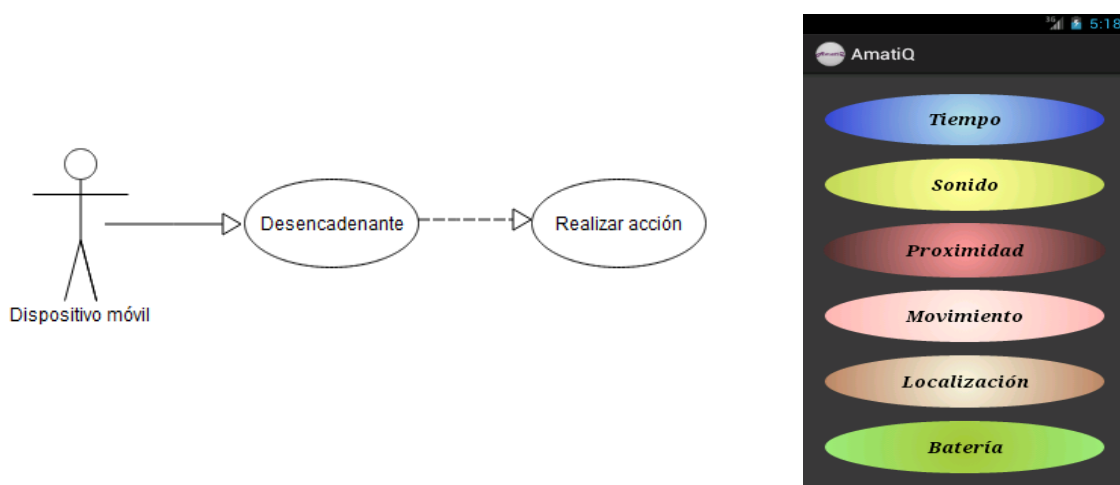


Figura 1: Opciones principales basadas en el desencadenante (Fuente: elaboración propia)

3.1 TIEMPO

Las acciones de interés se definen dependientes del día y la hora, actuando en cierto sentido como una alarma ampliada. Como se presenta en la Figura 2, estas acciones son:

- Permitir ingresar un recordatorio de un evento, con un mensaje personalizado, el cual se despliega en el momento indicado
- Indicar cuándo queremos que el teléfono entre en modo silencioso
- Registrar un mensaje de texto que sea enviado, en el momento indicado, al contacto seleccionado

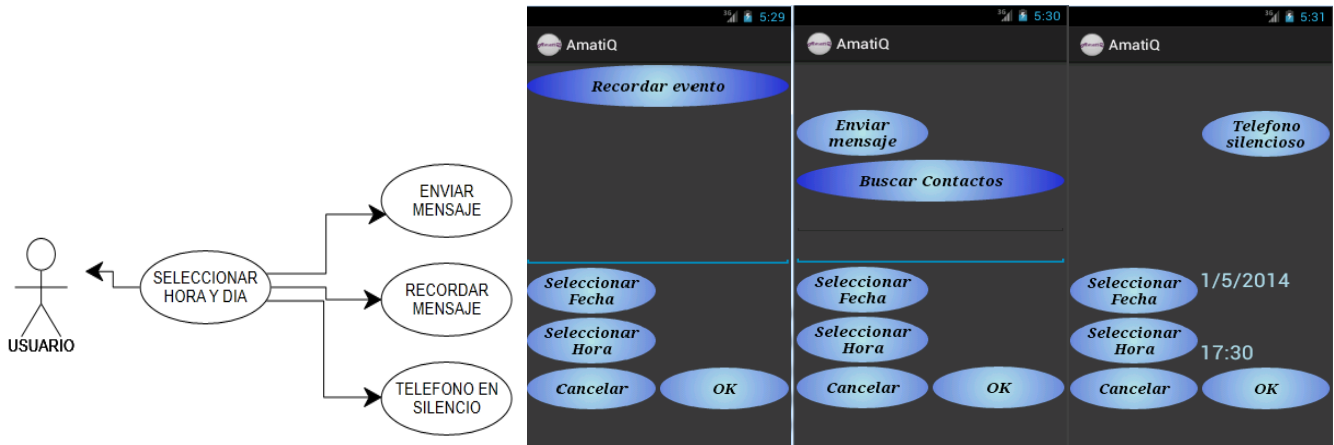


Figura 2: Configuración del tiempo como desencadenante (Fuente: elaboración propia)

3.2 SONIDO

El volumen del dispositivo puede ocasionar problemas cuando no está en sintonía con el ruido ambiental. Un volumen muy bajo del dispositivo cuando el ruido ambiental es muy alto, por ejemplo al caminar en una calle congestionada, puede ocasionar la pérdida de llamadas urgentes. Por otro lado, el caso inverso, volumen del dispositivo muy alto en ambientes sin ruido puede llamar innecesariamente la atención hacia el usuario. Este desencadenante, graficado en la Figura 3, nos permite configurar:

- Subir o bajar el volumen del timbre o advertencias del teléfono dependiendo del ruido ambiente, basados en un controlador PID (Patikirikorala, Colman, Han, & Wang, 2012), para mantenerlo dentro de un rango diferencial cómodo para el usuario.
- Enviar automáticamente un mensaje de respuesta cuando en circunstancias de alto ruido ambiente no se haya contestado una llamada telefónica en proveniencia de contactos dentro de algún grupo predeterminado de la agenda.

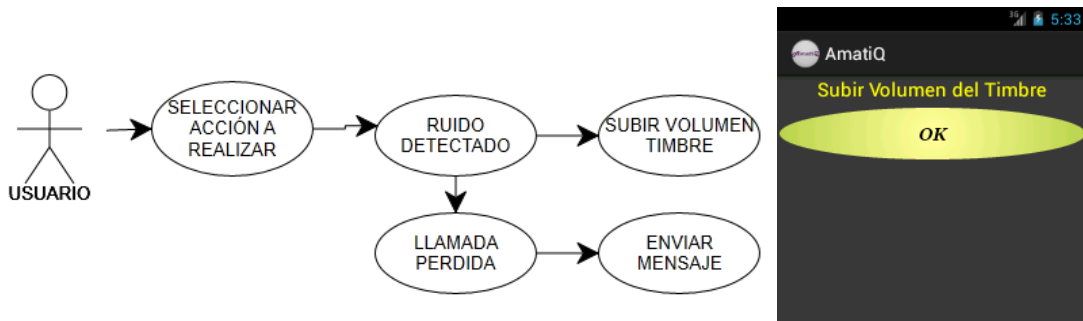


Figura 3: Configuración del sonido como desencadenante (Fuente: elaboración propia)

3.3 LOCALIZACIÓN

La automatización de acciones basada en la localización del dispositivo es probablemente uno de los campos de investigación más activos en la actualidad, en lo que se refiere a la sensibilidad al contexto (Bicocchi, Castelli, Mamei, & Zambonelli, 2011; Leung, Lee, & Lee, 2011). En este sentido AmatiQ implementa en esta etapa, como se grafica en la Figura 4, dos acciones que utilizan el GPS para determinar la cercanía con un lugar determinado por el usuario:

- Enviar un mensaje a un usuario determinado al detectar que la cercanía con las coordenadas preestablecidas es menor al valor configurado. Esto es de particular utilidad para dar alertas tempranas en casos cuando por ejemplo, se necesita que alguien permita el acceso, sin necesidad de distraer la conducción para efectuar una llamada.
- Encender la antena WiFi, al detectar que se está suficientemente cerca de un lugar establecido que ofrezca cobertura, permitiendo un cierto ahorro de energía al evitar la constante búsqueda de hotspots en la que incurre una antena inalámbrica activa. Esto se puede fácilmente extender a bluetooth, y combinarse con una desconexión inteligente del GPS para incrementar aún más la reacción automática a la ubicación y el ahorro de la batería.

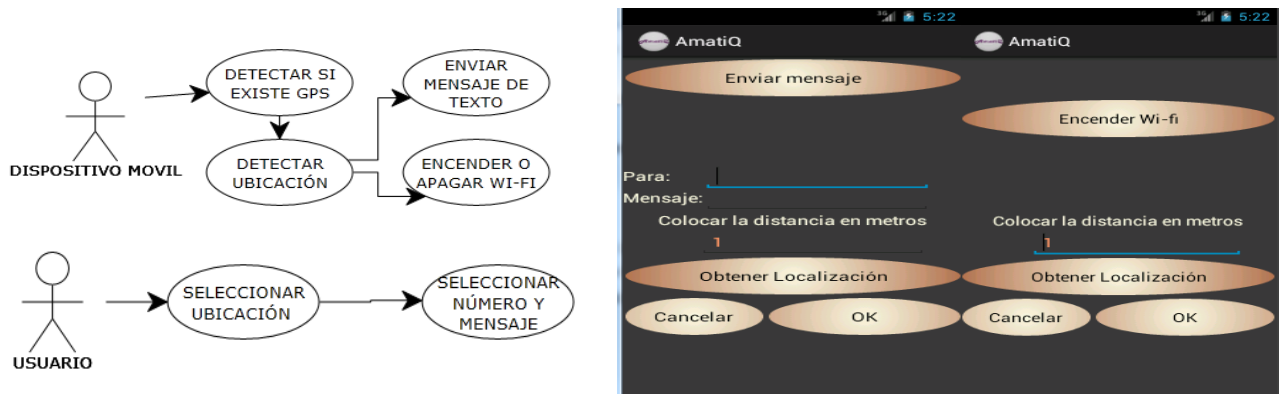


Figura 4: Configuración de la localización como desencadenante (Fuente: elaboración propia)

3.4 MOVIMIENTO

La utilización del acelerómetro como técnica para detección de actividades cobra interés actualmente como demuestra sobre todo el trabajo realizado en Japón por el Human Activity Sensing Consortium (Kawaguchi et al., 2011), entre otras investigaciones e iniciativas que se detallan en la revisión hecha por Loayza et al. (2013). Un ejemplo podría ser detectar que el usuario está en reposo y poner el celular en silencio para permitirle descansar, o marcar un número predeterminado con solo sacudir el teléfono de cierta manera.

AmatiQ, como se presenta en la Figura 5, utiliza actualmente el acelerómetro para detectar el tipo de movimiento que ha sufrido el teléfono (de entre un set predeterminado de alternativas) y ejecutar una aplicación previamente definida, poniéndola en el foreground.

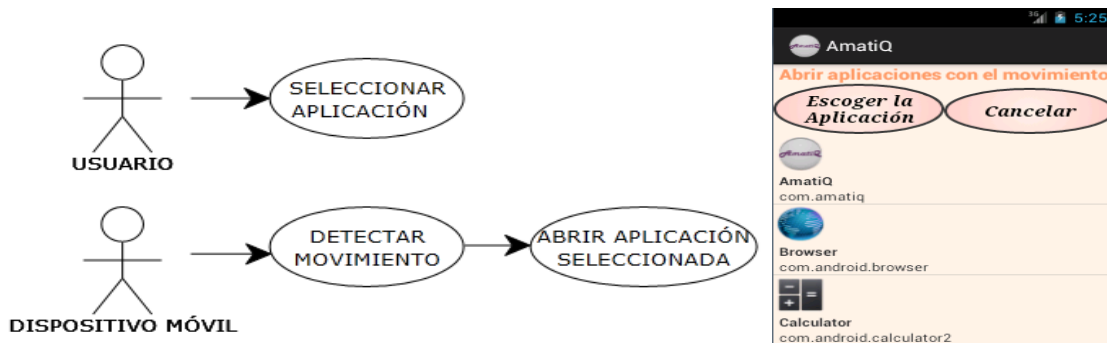


Figura 5: Configuración del movimiento como desencadenante (Fuente: elaboración propia)

3.5 PROXIMIDAD

El sensor de proximidad es generalmente utilizado por el sistema para evitar que se efectúen acciones indeseadas por el contacto de la oreja o de partes del rostro con la pantalla táctil, mientras se efectúa una llamada. En la Figura 6 esquematizamos brevemente cómo AmatiQ extiende esta funcionalidad permitiendo que una llamada se conteste simplemente con acercarse el teléfono al rostro.

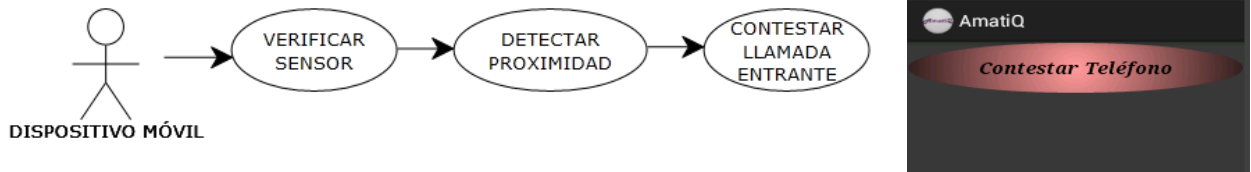


Figura 6: Configuración de la proximidad como desencadenante (Fuente: elaboración propia)

3.6 BATERÍA

El consumo de energía es de primordial interés en un dispositivo móvil. El usuario busca al menos poder llegar al término del día con su teléfono encendido y funcional. En ocasiones el uso intensivo del teléfono en ciertas jornadas hace difícil conseguir este objetivo, por lo que AmatiQ usa como desencadenante el nivel de carga de la batería para reducir funciones cuando el nivel baja de un cierto rango, y reactivarlas cuando dicho nivel se normaliza. Actualmente AmatiQ considera dos acciones que se presentan en la Figura 7:

- Desactivar completamente el WiFi al reducirse el nivel de carga por debajo de un valor indicado por el usuario, y reactivarlo al normalizarse la carga de la batería.
- Reducir el brillo de la pantalla a un nivel establecido, igualmente al reducirse la carga de la batería, y restaurarlo al subir nuevamente la carga sobre el límite establecido.



Figura 7: Configuración del nivel de carga de la batería como desencadenante (Fuente: elaboración propia)

3.7 RESULTADOS GENERALES

Luego de la fase inicial de depuración del prototipo se solicitó a un grupo de usuarios evaluar la funcionalidad y solidez de la versión inicial de la aplicación. Por cada una de las acciones definidas por desencadenante se pidió que consignen su apreciación en tres categorías: estabilidad de la funcionalidad luego de varias ejecuciones, facilidad de configuración y uso, y finalmente la utilidad percibida de la alternativa. Los resultados consolidados se presentan en la Tabla 4, destacándose que el manejo de sonido y proximidad presentan inconsistencias en los resultados, y que en general los usuarios esperarían poder combinar las diferentes reglas y acciones para obtener un mayor nivel de automatización.

Tabla 4: Pruebas de usuarios (Fuente: elaboración propia)

Desencadenante	Estabilidad	Facilidad	Utilidad
Tiempo	Estable	Alta	Alta
Sonido	Inestable	Media	Media
Proximidad	Inestable	Media	Baja
Movimiento	Estable	Baja	Alta
Localización	Estable	Alta	Alta
Batería	Estable	Alta	Alta

3.8 NOTAS SOBRE LA IMPLEMENTACIÓN

Desarrollar aplicaciones sensibles al contexto requiere combinar activamente el conjunto de sensores del dispositivo con las preferencias del usuario respecto a las posibles interacciones entre dicho dispositivo y las actividades que el usuario se encuentra realizando, las cuales muy posiblemente estarán relacionadas a su localización. Hacer trabajar todos estos elementos de manera armoniosa requiere el uso de técnicas y metodologías específicas de software para facilitar el desarrollo de estas aplicaciones.

No está dentro del alcance de este trabajo presentar detalles de la implementación de la aplicación, sin embargo se considera necesario destacar ciertos objetos y clases dentro de la API de Android que pueden ser de interés general de los lectores.

Las aplicaciones sensibles al contexto son particularmente dependientes de los sensores y el manejo que se haga de los mismos, y AmatiQ no es la excepción. Android permite acceder a los sensores internos del dispositivo a través de las clases `Sensor`, `SensorEvent`, `SensorManager`, y la interfaz `SensorEventListener`, del paquete `android.hardware`.

La clase `BroadcastReceiver` se revela de gran importancia ya que permite registrar eventos del sistema a ser capturados permitiendo recuperar información de los sensores. Esto se realiza mediante el registro en el `AndroidManifest` o mediante `Intent` implícitos, por nombrar ciertas alternativas. Muchas acciones requerirán también la ejecución de tareas en segundo plano, para lo cual `Service` y `AsyncTask` son necesarias.

El paso relativo a la ejecución de acciones puede hacer necesario usar clases como `AlarmManager`, `AudioManager` o `TelephonyManager`, y por supuesto para persistir la información de configuración es conveniente el uso de `SharedPreferences`.

Finalmente, se destaca que AmatiQ fue desarrollada para ejecutarse en versiones de Android a partir de la 3.0, y fue probada en distintos tipos de AVD, con el soporte de telnet y adb para el envío de información simulada, así como en dispositivos físicos Galaxy Tab y SIII mini.

4. CONCLUSIONES Y RECOMENDACIONES

Varios son los objetivos de este trabajo. En primer lugar se presenta una revisión de la conceptualización fundamental de la sensibilidad al contexto en las aplicaciones móviles. También se discute el interés del usuario de smartphones para extraer constantemente más funcionalidad de su dispositivo, intentando delegar en el mismo la responsabilidad de determinar autónomamente la conveniencia de ejecutar actividades, o de activar y desactivar funcionalidad, entre otras características.

En segundo lugar se presenta una metodología de prototipado adaptada al caso de las aplicaciones orientadas al contexto guiadas por desencadenantes. Se muestra cómo, siguiendo esta orientación, se puede conseguir la aplicación deseada, presentándose finalmente nociones guía sobre la implementación y las pruebas, destacando al mismo tiempo las posibilidades de ampliación de la autonomía del sistema.

Los resultados muestran que los usuarios están razonablemente satisfechos con la funcionalidad global de la aplicación, sin embargo algunos problemas se presentaron con los desencadenantes de sonido y proximidad. En ambos casos parece afectar el hecho de que los dispositivos suelen encontrarse generalmente guardados en un bolsillo.

En el caso del sonido, la intensidad del ruido ambiente no se detecta correctamente, y sería recomendable una reconfiguración más fina del controlador PID, así como estudiar la posibilidad de implementar un mecanismo diferente, probablemente de aprendizaje automático.

En lo relativo al desencadenante de proximidad, este tiende a ejecutarse antes de que la acción verdadera se lleve a cabo, dado que la acción de sacar el dispositivo del bolsillo tiene muchas variables fuera de control. Se propone evaluar una combinación con el sensor de luz. Sin embargo la funcionalidad de proximidad es la menos apetecida por los usuarios y se debe evaluar también su relevancia.

Un punto en que los usuarios en general coinciden, es en su interés por combinar las reglas y acciones para generar una conducta automática más compleja. Este punto se considerará en una segunda versión del aplicativo, donde se estudiarán los posibles beneficios, así como las posibles dificultades producto de las combinaciones de alternativas probablemente incompatibles.

La interacción del usuario con los dispositivos móviles sensibles al contexto es un área activa de investigación. AmatiQ es una aplicación concebida y diseñada para evolucionar, y entre las posibilidades que se presentan se considera particularmente interesante orientar el estudio hacia el uso de interfaces multimodales que se intercambien automáticamente dependiendo del contexto (Lemmelä, Vetek, Mäkelä, & Trendafilov, 2008). Más aún, distintos tipos de usuario y distintos tipos de dispositivos pueden requerir distintos tipos de interfaces (Naumann, Wechsung, & Hurtienne, 2010).

Otras áreas activas se enfocan en la localización precisa del usuario para proporcionarle información relevante y en la detección de las actividades con especial preocupación en el control de interrupciones. Finalmente, se muestra una tendencia hacia el desarrollo de aplicaciones inteligentes sensibles al contexto, que hagan uso de técnicas de aprendizaje automático para encontrar relaciones significativas a partir del contexto (Miller, 2013; Sayid, 2013).

RECONOCIMIENTOS

El presente trabajo ha sido realizado en el contexto del proyecto de investigación DADMDV- VI.UIO.ING.23, financiado y apoyado por el Instituto de Transferencia de Tecnologías, ITT, de la Universidad Tecnológica Equinoccial, Quito – Ecuador.

REFERENCIAS BIBLIOGRÁFICAS

- Arnowitz, J., Arent, M., & Berger, N. (2010). *Effective Prototyping for Software Makers*. Burlington: Elsevier Science.
- Baldauf, M., Dustdar, S., & Rosenberg, F. (2007). A survey on context-aware systems. *International Journal of Ad Hoc and Ubiquitous Computing*, 2(4), 263. doi:10.1504/IJAHUC.2007.014070
- Bicocchi, N., Castelli, G., Mamei, M., & Zambonelli, F. (2011). Augmenting mobile localization with activities and common sense knowledge. In *Proceedings of the Second international conference on Ambient Intelligence* (pp. 72–81). Berlin, Heidelberg: Springer-Verlag. doi:10.1007/978-3-642-25167-2_8
- De Sá, M., & Carriço, L. (2011). Designing and Evaluating Mobile Interaction: Challenges and Trends. *Found. Trends Hum.-Comput. Interact.*, 4(3), 175–243. doi:10.1561/11000000025
- DeVaul, R., Sung, M., Gips, J., & Pentland, A. “Sandy.” (2003). MIThril 2003: Applications and Architecture. In *Proceedings of the 7th IEEE International Symposium on Wearable Computers* (p. 4–). Washington, DC, USA: IEEE Computer Society. Retrieved from <http://dl.acm.org/citation.cfm?id=946249.946911>

- International Organization for Standardization. (2010). *ISO 9241-210:2010 Ergonomics of human-system interaction – Part 210: Human-centred design for interactive systems*. Retrieved from http://www.iso.org/iso/iso_catalogue/catalogue_ics/catalogue_detail_ics.htm?csnumber=52075
- Kawaguchi, N., Ogawa, N., Iwasaki, Y., Kaji, K., Terada, T., Murao, K., ... Nishio, N. (2011). HASC Challenge: gathering large scale human activity corpus for the real-world activity understandings. In *Proceedings of the 2nd Augmented Human International Conference* (pp. 27:1–27:5). New York, NY, USA: ACM. doi:10.1145/1959826.1959853
- Lemmelä, S., Vetek, A., Mäkelä, K., & Trendafilov, D. (2008). Designing and evaluating multimodal interaction for mobile contexts. In *Proceedings of the 10th international conference on Multimodal interfaces* (pp. 265–272). New York, NY, USA: ACM. doi:10.1145/1452392.1452447
- Leung, K. W.-T., Lee, D. L., & Lee, W.-C. (2011). CLR: a collaborative location recommendation framework based on co-clustering. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval* (pp. 305–314). New York, NY, USA: ACM. doi:10.1145/2009916.2009960
- Loayza, A., Proaño, R., & Ordóñez Camacho, D. (2013). Aplicaciones sensibles al contexto. *Tendencias actuales. Enfoque UTE*, 4(2), pp. 95 – 110.
- Millette, G., & Stroud, A. (2012). *Professional Android Sensor Programming* (1st ed.). Hoboken: Wiley.
- Miller, C. C. (2013, July 29). Apps That Know What You Want, Before You Do. *The New York Times*. Retrieved from <http://www.nytimes.com/2013/07/30/technology/apps-that-know-what-you-want-before-you-do.html>
- Naumann, A. B., Wechsung, I., & Hurtienne, J. (2010). Multimodal interaction: A suitable strategy for including older users? *Interact. Comput.*, 22(6), 465–474. doi:10.1016/j.intcom.2010.08.005
- Patikirikorala, T., Colman, A., Han, J., & Wang, L. (2012). An Evaluation of Multi-model Self-managing Control Schemes for Adaptive Performance Management of Software Systems. *J. Syst. Softw.*, 85(12), 2678–2696. doi:10.1016/j.jss.2012.05.077
- Sayid, R. (2013, May 8). Emotion Sense: App to track your mood could turn your mobile into a pocket therapist. *mirror*. Retrieved September 24, 2013, from <http://www.mirror.co.uk/news/technology-science/technology/emotion-sense-app-track-your-1874867>
- Uchiyamada, K. (2007). Context-Aware. Universidad Católica Nuestra Señora De La Asunción. Retrieved from http://www.jeuazarru.com/docs/context_aware.pdf
- Want, R., Hopper, A., Falcão, V., & Gibbons, J. (1992). The Active Badge Location System. *ACM Trans. Inf. Syst.*, 10(1), 91–102. doi:10.1145/128756.128759
- Wei, E. J. Y., & Chan, A. T. S. (2007). Towards context-awareness in ubiquitous computing. In *Proceedings of the 2007 international conference on Embedded and ubiquitous computing* (pp. 706–717). Berlin, Heidelberg: Springer-Verlag. Retrieved from <http://dl.acm.org/citation.cfm?id=1780745.1780822>

Autorización y descargo de responsabilidad

Los autores autorizan a LACCEI a publicar el artículo en las actas de la conferencia. Ni LACCEI ni los editores son responsables por el contenido o las repercusiones de lo expresado en el artículo.