

GOOGLE ANDROID y MOODLE

Aris Castillo de Valencia
Fac. Ingeniería de Sistemas Computacionales
Universidad Tecnológica de Panamá
Panamá, Panamá
acastillo@country507.com

Abstract— Integrando Moodle y Android se logra una herramienta de M-learning, lo cual es la última tendencia en educación a distancia. Esto se puede realizar a través de agentes y web services. De los distintos protocolos posibles, REST parece ser el más apropiado.

Keywords - M-learning; Android; Moodle, dispositivos móviles.

I. INTRODUCCIÓN

Hoy la tendencia es movilidad. Todos queremos tener a mano en un dispositivo conectado a Internet todo lo que sea posible, desde una agenda con nuestros contactos hasta el acceso a nuestras actividades diarias, y entre ellas, por supuesto, las académicas. Con certeza podemos decir que los dispositivos móviles inteligentes están cambiando la forma como trabajamos y actuamos [1].

M-learning o “mobile learning”, se refiere a la utilización de dispositivos móviles y creación de aplicaciones que permitan el uso de los mismos en entornos educativos a distancia o bien para facilitar el acceso de los estudiantes a recursos educativos sin necesidad de estar físicamente en mismo sitio. M-learning involucra la integración o comunicación entre diversas tecnologías, por un lado están los protocolos asociados con la plataforma de educación a distancia y por otro los de los dispositivos móviles correspondientes.

El objetivo de este artículo es presentar trabajo que se están realizando en la Universidad Tecnológica de Panamá para desarrollar una aplicación que integre dispositivos móviles basados en Google Android con un servidor Moodle para educación a distancia y soporte a la educación presencial [2]. El objetivo de la investigación es ofrecer una solución gratuita a los usuarios de dispositivos de Google Android dado que a la fecha no existe una aplicación personalizada totalmente funcional que les permita acceder a Moodle de una forma transparente y eficiente. Si bien el usuario podría acceder al servidor, lo haría con formatos que no están adaptados para el tipo de dispositivo, por lo que la experiencia podría ser perturbadora. No es éste el caso para otros dispositivos móviles.

II. PROGRESO DE LA EDUCACIÓN A DISTANCIA

La educación a distancia no es nueva. Se remonta a los años setenta cuando los principales medios de comunicación eran el papel a través del correo postal y donde la comunicación era prácticamente unidireccional. Más tarde, los avances

tecnológicos permiten mejor divulgación del conocimiento y mayor pedagogía y andragogía a través de la incorporación de tecnologías tales como la radio y la televisión, aunque la interactividad seguía siendo poca. Luego, con la apertura del Internet y del WWW al mundo, se logra la buscada interactividad a través de ambientes integrados basados en la web. Sin embargo, el acceso era inminentemente estacionario; es decir, el usuario debía estar conectado desde una computadora en algún sitio con conexión a Internet. Hoy, además de eso, el usuario demanda la posibilidad de acceso a estos recursos a través de dispositivos móviles. Esto ha dado paso a lo que se conoce como m-learning.

Las soluciones de m-learning pueden implicar sistemas complejos que involucren distribución de contenido, retroalimentación y discusión y la evaluación misma, o bien puede tratarse de sistemas más simples, que sirvan de apoyo a las actividades presenciales [1].

En nuestro proyecto buscamos proveer sólo una interfaz que facilite a nuestros usuarios de Google Android acceder a los principales recursos del servidor Moodle. Conectarse a través de un computador para realizar sus tareas todavía sería necesario. Las funcionalidades que se incluirán son bajar archivos, participar en foros y chats, ver evaluaciones, realizar ejercicios cortos y ver el calendario [2]. También se pretende desarrollar agentes de alertas, que sincronizando funciones del hardware con otras de software, le brinden un valor agregado al usuario [2]. Tal es el caso de avisos por medio de un sonido o la vibración de su dispositivo. Esto constituye una innovación del proyecto ya que actualmente no tenemos este tipo de funciones.

Por ejemplo, supongamos que hay una tarea próxima a cumplirse. Si el profesor ha configurado un recordatorio en Moodle, el usuario en su dispositivo móvil, recibirá una alerta de acuerdo con dicha configuración. En este caso, el recordatorio es parte de una función configurada en Moodle, pero ésta se sincronizará con la agenda del dispositivo móvil, de manera que el usuario la reciba como algo más que un mensaje de correo.

III. QUÉ ES GOOGLE ANDROID?

Android es el nombre con el que se conoce al sistema operativo que opera el hardware de ciertos dispositivos móviles. Este software open source es desarrollado por la alianza “Open Handset Alliance” e incluye tanto un sistema operativo como el middleware y aplicaciones para los

dispositivos móviles [7], [13]. La figura 1 muestra la estructura de Android.



Figura 1. Arquitectura de Android

En la parte más baja está el kernel de Linux donde se realizan las principales funciones para administración de memoria, de procesos y de entrada/salida. A partir de allí, se encuentran las librerías, las cuales están desarrolladas en C y C++, pero llamadas a través de interfaces de Java, corriendo sobre una máquina virtual Dalvik. Incluye el sistema de administración de base de datos relacional SQLite; el API gráfico 3D OpenGL ES 2.0 para visualización efectiva de videos y gráficos; el layout engine WebKit el cual permite a los browsers renderizar páginas web; librerías de gráficos SGL, y protocolos criptográficos para la comunicación en Internet SSL y las librerías GNU C, glibc para las diferentes arquitecturas de hardware. Con la máquina virtual Dalvik, que es una parte del sistema operativo Android, se logra mayor eficiencia en los recursos del hardware ya que convierte cada aplicación a correr en el sistema a un formato compacto (.dex), diseñado para dispositivos con limitaciones de memoria y procesamiento [10]. Las aplicaciones básicas (core) de Android son el cliente de email, SMS, calendario, contactos y mapas.

Los dispositivos basados en este sistema operativo son capaces de recibir, manejar y entregar buena calidad de contenido multimedia por paquetes, incluyendo, voz, audio, creación y recepción de documentos de texto y gráficos. Así mismo permiten la transmisión de paquetes en tiempo real ya que cuentan con los protocolos de servicios para aplicaciones de este tipo. Por lo tanto, pueden perfectamente ser adaptados para servir de medio para acceder a sistemas de educación a distancia basados en CMS (Course Management System).

En sí, lo que hace falta es integrar las funciones del CMS con las del dispositivo de manera que ambos puedan comunicarse efectivamente. El proyecto aquí discutido parcialmente consiste en desarrollar una interfaz de comunicación entre dispositivos basados en Android y la plataforma de administración de contenido Moodle.

IV. QUÉ ES MOODLE?

Moodle es un sistema de administración de contenido (CMS) open source. También se le reconoce como un sistema de administración de aprendizaje (LMS, learning management system) o un ambiente de aprendizaje virtual (VLE) comparable con Blackboard y WebCT. Las estadísticas muestran que es la plataforma más utilizada actualmente con alrededor de 50,000 sitios registrados y 37 millones de usuarios y 3.7 millones de cursos en 210 países y en 75 lenguajes [11].

Moodle es un sistema modular cuyo núcleo está formado por módulos para cursos y actividades, usuarios, grupos y agrupaciones, registro y control de acceso, actividad y construcción de cursos, navegación y configuración, librería de formas, librerías JavaScript, instalación y actualización y logs y estadísticas [11]. En un nivel más alto, están módulos para consultas, foros, diario, cuestionario, recursos, encuesta y wiki.

Es importante aclarar que nuestro proyecto no requiere modificar Moodle, ni siquiera habría necesidad de modificar la base de datos. Por lo tanto, sólo ha sido necesario evaluar cuáles de estas funciones deseamos que estén disponibles para el usuario en el dispositivo móvil. Tampoco queremos que en el dispositivo esté todo lo que se puede realizar desde un computador. Lo que buscamos es que el usuario tenga la facilidad para realizar aquellas tareas más importantes, útiles y de fácil desarrollo, dadas las limitaciones de pantalla y de ergonomía de teclado. Para dar con estas funciones, realizamos un estudio con el conjunto de requerimientos definidos por RUP (Rational Unified Process), a saber, Visión, Stakeholder Requests y Software Requirements. También estudiamos Moodle y evaluamos qué servicios eran funcionales en el dispositivo móvil, tomando en cuenta aspectos de ergonomía. Así, se determinó que habilitaríamos bajar archivos, participar en foros y chats, ver evaluaciones, realizar ejercicios cortos, ver el calendario y ofrecer alertas. Luego se desarrollaron los casos de uso correspondientes.

Para comunicar Moodle con Android se utilizan web services que permiten el paso de mensajes entre la base de datos y los dispositivos móviles. En el lado del servidor, solamente es necesario realizar las configuraciones para habilitar los web services. Se puede entrar así: `your_moodle/webservice/wsdoc.php?protocol=soap` [11]. Por su parte, en el lado de los dispositivos finales, sería necesario instalar la aplicación.

V. INTEGRACIÓN

Algunas plataformas para el desarrollo de web services son XML y HTTP. El primero para la representación e interpretación de mensajes, mientras que el segundo permite la comunicación sobre Internet. Los web services, a diferencia de los browsers, crean una página web para solicitar y procesar algo; y luego ésta es consumida por un programa autónomo. Protocolos para web services incluyen a SOAP, REST, XML-RPC. Esto es especialmente útil para dispositivos con recursos limitados en potencia y procesamiento como lo son los dispositivos móviles.

Moodle provee soporte para los protocolos REST, SOAP, XML, RPC y AMF [11]. De estas opciones XML-RPC es la menos indicada dado que es más antiguo, difícil y de más baja interoperabilidad. En el caso de SOAP y REST son más fáciles de utilizar y son de tipo semánticos. La ventaja de SOAP es que, dado que para cualquier transacción se empotra el nombre del método en el paquete junto con las cadenas de solicitud y respuesta, puede existir sobre cualquier protocolo. REST por otro lado, empotra sólo el nombre del método en la solicitud URL, con lo cual se liga con HTTP.

Como se ve en la figuras 2 y 3, REST es más simple y menos verboso que SOAP. REST cuenta con millones de sustantivos (nouns) para representar conceptos, pero sólo verbos universales para describir las acciones sobre los sustantivos, a saber: GET para obtener información, POST para agregar nueva información, PUT para actualizar la información, y DELETE para borrar información. Esto lo hace un protocolo de bajo acoplamiento. El servidor REST de Moodle utiliza parámetros POST y XML para valores de retorno [11].

Otra ventaja de REST es que Android tiene librerías agregadas de éste y no así de SOAP ni de XML-RPC [18]. La política de Android es utilizar aquellos protocolos más eficientes y de menos consumo para el dispositivo.

Ejemplo de una solicitud SOAP

```
GET /course HTTP/1.1
Host: htt://mlea.utp.ac.pa/mle

<?xml version="1.0"?>
<soap:Envelope
xmlns:soap="http://www.w3.org/2001/12/soap-envelope"
xmlns:m=" htt://mlea.utp.ac.pa/mle ">
  <soap:Header>
    <m:DeveloperKey>1234</t>
  </soap:Header>
  <soap:Body>
    <m:GetSelectCourse>
      <m:CourseName>HUMC</m:CourseName>
    </m:GetSelectCourse>
  </soap:Body>
  <soap:Header>
    <m:DeveloperKey>1234</t>
  </soap:Header>
  <soap:Body>
    <m:GetSelectCourse>
      <m:CourseName>HUMC</m:CourseName>
    </m:GetSelectCourse>
  </soap:Body>
</soap:Envelope>
```

Figura 2. Solicitud SOAP

Ejemplo de una solicitud REST

```
GET /SelectCourseName=HUMC HTTP/1.1
Host: htt://mlea.utp.ac.pa
```

Figura 3. Solicitud REST

Los agentes de comunicación entre las plataformas hacen posible la implementación de los web services[10]. Esto no es más que la pieza de software que envía y recibe mensajes. Un ejemplo de API de un cliente para Moodle con REST consta de

varios elementos. Primero es necesario crear una implementación para la interfaz del cliente, para lo cual utilizamos DefaultHttpClient(); luego se crea la solicitud a través de la instancia de implementación de la solicitud con HttpGet(); luego se implementa la ejecución con execute(), tal como se aprecia en la figura 4.

Android incluye tres “parsers” o analizadores gramaticales de XML, a saber: W3C DOM (org.w3c.dom), SAX (org.xml.sax), y XML. Adicionalmente, incluye JSON (org.json) [18]. Esta parte constituye el último elemento del agente. En nuestro caso, se prefirió usar JSON (Java Script Object Notation) porque es ligero, fácil de leer y escribir, y fácil para las máquinas interpretar y generar [19].

```
//Implementación de la interfaz del cliente
HttpClient httpClient = new DefaultHttpClient();

//Creación de la solicitud
HttpGet httpget = new HttpGet(url);

//Ejecución y lectura con JSON
HttpResponse response;
try {
    response = httpClient.execute(httpget);
    HttpEntity entity = response.getEntity();
    if (entity != null) {
        InputStream instream = entity.getContent();
        result = convertStreamToString(instream);
    }
}
```

Figura 4. Cliente de Android para HTTP

Sería necesario crear tantos agentes como actividades se deseen permitir al usuario, por ejemplo, para comunicación asíncrona y síncrona, para las alertas, para la actualización y participación en foros, etc. Entonces en un momento particular, de acuerdo con la cantidad de usuarios que están conectados al sistema utilizando sus dispositivos móviles, así sería la cantidad de agentes que estarían corriendo en paralelo.

VI. CONCLUSIONES

Son muchas las facilidades que el m-learning puede ofrecer, no sólo a estudiantes sino a profesores en términos de la ventaja que representa poder acceder en cualquier lugar y en cualquier momento a los recursos del sistema. Nuestro proyecto busca la integración de Moodle con Android con el fin de proveerles esta facilidad a nuestros usuarios de Moodle con dispositivos Android, ya que actualmente no existe. Más que permitirles acceder a ciertos recursos de Moodle, la aplicación les ofrecerá alertas, con lo cual se sacará mayor provecho de las funciones nativas del hardware, y que representa un valor agregado para los usuarios de estos dispositivos móviles.

La integración es posible a través de agentes y web services. Existen varios protocolos de web services, de ellos

REST es el más recomendable para dispositivos móviles, como Android, ya que es más flexible y menos pesado. Además, porque sus librerías están agregadas en Android y Moodle lo soporta. Además seleccionamos utilizar JSON como parser teniendo en cuenta que es ligero y fácil de escribir e interpretar.

Actualmente nos encontramos en la fase de desarrollo de los webservices y agentes, para posteriormente, hacer las pruebas. Finalmente, pondríamos la aplicación a disposición de los usuarios.

RECONOCIMIENTOS

Este proyecto es financiado por la Secretaría Nacional de Ciencia y Tecnología e Innovación (SENACYT) de Panamá por el contrato de Colaboración Internacional No. COL-09-045 en la convocatoria I+D COL 2009.

REFERENCIAS

- [1] Mohamed, Ally. Mobile Learning: Transforming the Delivery of Education and Training.
- [2] Proyecto MLEA: <http://mle.utp.ac.pa/mlea/>
- [3] Rogers, Lombardo, Mednieks, Meike. Android Application Development: Programming with the Google SDK.
- [4] Fing, Brian. Mobile Design and Development: Practical Concepts and Techniques for Creating Mobile Sites and Web Applications.
- [5] Mizoguchi, Dicheva, Greer. Future of Learning: Semantic Web Technologies for e-learning.
- [6] Mlearning: <http://en.wikipedia.org/wiki/MLearning>
- [7] Android: [http://en.wikipedia.org/wiki/Android_\(operating_system\)](http://en.wikipedia.org/wiki/Android_(operating_system))
- [8] Market share de smartphones en EUA: <http://www.celularis.com/tag/market-share-smartphones>
- [9] Smartphone Market Share: Android Bypasses Apple: <http://www.clickz.com/clickz/stats/1935425/smartphone-market-share-android-bypasses-apple>
- [10] Dalvik Virtual Machine: <http://www.dalvikvm.com/>
- [11] Moodle: <http://moodle.org/>
- [12] Introduction to web services: http://www.w3schools.com/webservices/ws_intro.asp
- [13] What is Android?: <http://developer.android.com/guide/basics/what-is-android.html>
- [14] Programming Android: <http://programming-android.labs.oreilly.com/>
- [15] Web services architecture: <http://www.w3.org/TR/2003/WD-ws-arch-20030808/#id2608803>
- [16] Development: Web services: http://docs.moodle.org/en/Development:Web_services
- [17] Haseman, Chris. Android Essentials. APRESS.
- [18] Murphy, Mark. Beginning Android 2. APRESS. 2010.
- [19] Introducing JSON: <http://www.json.org/>