

PROGRAMACIÓN DE JUEGOS PARA MÓVILES

Keny Vidangos Orellana
Catedrática Universidad del Valle - Cochabamba



RESUMEN

El presente artículo comienza realizando una remem-branza sobre el avance que ha tenido la tecnología respecto a los teléfonos celulares y las capacidades que han ido adquiriendo con el transcurso de los años, desde comunicación por voz hasta el servicio de mensajería con multimedia.

Presenta una breve descripción de los sistemas operativos que corren en nuestros celulares y las diferentes herramientas que permiten el desarrollo de aplicaciones para pequeños dispositivos, tales como J2ME que basa en los conceptos de configuración y perfil todo el desarrollo de aplicaciones (juegos).

Finalmente, se exponen los pasos básicos que deben realizarse para correr con éxito una aplicación creada en el entorno de desarrollo que provee J2ME Wireless Toolkit, denominado KToolBar.

Palabras Clave: tecnología celular, tecnología WAP, CLDC, MIDP

Marco Teórico conceptual

Desde sus inicios a finales de los 70, la tecnología celular ha revolucionado enormemente. Los teléfonos celulares se han convertido en una herramienta primordial para la gente común y de negocios.

Aunque la comunicación telefónica por voz es el principal uso de estas terminales, se han desarrollado nuevas formas de comunicación y otras capacidades para estos pequeños dispositivos. El primero, y quizás el

más lucrativo hasta la fecha, es el uso de la mensajería SMS (Short Message Service). Nuestras pequeñas terminales nos permiten enviar mensajes cortos de texto, que son enviados desde la terminal, al centro servidor de mensajes cortos o SMSC (Short Message Service Centre), que a su vez se encarga de hacer llegar el mensaje al móvil destinatario.

Hoy en día, ya existen celulares capaces de navegar por Internet, pero las limitaciones de pantalla y teclado hacen inviable su uso con páginas web normales. Así que nace la tecnología WAP, que nos permite navegar por páginas especiales creadas en WML en lugar de HTML.

Para paliar las bajas velocidades -sin contar con la baja fiabilidad- de la tecnología GSM para la transmisión de datos, apareció la tecnología GPRS (General Packet Radio Service). GPRS ofrece una red de transferencia de datos sin hilos a una velocidad aceptable; se puede enviar grandes paquetes de información, como fotografías, música e incluso video. Nace una nueva generación de móviles capaz de reproducir canciones más complejas y mostrar gráficos y fotografías a color. De modo paralelo, a la aparición de estos móviles en el mercado, aparece el nuevo servicio de mensajes cortos llamado MMS (Multimedia Message Service). Gracias a MMS, además de texto, podemos enviar fotografías, sonidos, imágenes y otros.

Hoy en día, nuestros teléfonos móviles corren auténticos sistemas operativos. El más conocido es Symbian,

1 Ingeniería de Sistemas - UCB
Maestría en Gestión Empresarial - UMSS
Diplomado en Docencia Universitaria - Universidad del Valle
Docente Universidad del Valle

que es el corazón de gran cantidad de móviles como los Nokia, Sony-Ericsson, Motorola y otros, aunque recientemente Microsoft lanzó su nuevo sistema operativo para celulares Windows Phone 7.

Y no sólo se ha investigado y evolucionado en cuanto a sistemas operativos: Sun Microsystems da un paso adelante dentro de su tecnología Java y nos presenta J2ME (Java 2 Micro Edition), un entorno de producción para pequeños dispositivos que permite la ejecución de programas creados en Java. Una de las principales capacidades que añade esta tecnología a nuestras terminales es la posibilidad de descargar y ejecutar juegos con una calidad razonable (1).

A continuación, presentaremos un resumen de aspectos generales en cuanto a la programación de juegos para dispositivos móviles, utilizando J2ME. J2ME se basa en los conceptos de configuración y perfil, estando formado por la configuración CLDC y por el perfil MID (conocido por MIDP o MID Profile).

Connected Limited Device Configuration (CLDC)

CLDC es una configuración general que describe las características mínimas en cuanto a hardware y software para un amplio abanico de dispositivos, que van desde PDAs a teléfonos móviles y otros. Un perfil define las características del dispositivo de forma más específica. MIDP (Mobile Information Device Profile) define las APIs y características de hardware y software necesarias para el caso concreto de los teléfonos móviles. CLDC cubre las necesidades de pequeños aparatos con limitadas posibilidades en cuanto a interfaz de usuario, poder de proceso, etc. Esta configuración posee la K Virtual Machine, un intérprete de Java preparado para microprocesadores de 16 y 32 bits RISC/CISC, con tan sólo unos pocos cientos de Kb de memoria. Debido a esto, CLDC no incluye ciertos tipos de clases, por ejemplo en la versión 1.0 no se podían utilizar números decimales.

Concretamente CLDC define:

- Cuáles son las características del lenguaje Java incluidas.
- Qué funcionalidad será incluida en la máquina virtual Java.
- Las APIs necesarias para el desarrollo de aplicaciones en móviles.
- Los requerimientos hardware de los dispositivos.

Debido a las limitaciones de hardware en el que correrá la máquina virtual, algunas de las características del lenguaje Java han sido recortadas.

Importante: El verificador de J2SE es demasiado grande para ser incluido con el CLDC; de hecho es más grande que el KVM. Por tanto, debemos verificar los archivos antes de mandarlos al equipo donde queremos que se ejecuten. Para esta tarea, los SDKs poseen herramientas que nos ayudarán en su proceso y no será necesario realizar esta actividad manualmente.

Perfil

Un Perfil es una especificación de las APIs de Java, que funciona en conjunción con la capa de configuración. Añaden nuevas clases a las que venían con la configuración.

Existen muchos perfiles, pero los más importantes en referencia a J2ME son Personal Profile, Foundation Profile y MID Profile.

MIDP

Éste perfil está diseñado para funcionar especialmente con CLDC.



Primer MIDlet (programa para celulares)

Existen diferentes herramientas válidas para construir programas bajo el estándar J2ME, como el propio "Sun One Studio" de Sun Microsystems, aunque el más sencillo de utilizar es el "J2ME Wireless Toolkit 2.0" que proporciona Sun. Este entorno no ofrece gran potencia a la hora de desarrollar aplicaciones recompensando este hecho con la facilidad de aprendizaje (2).

Para instalar J2ME Wireless Toolkit, primero hemos de instalar el entorno de programación de J2SE (JDK). Se puede descargar la última versión de JDK desde <http://java.sun.com/j2se/download.html>. Una vez descargado e instalado, estaremos en condiciones de descargar e instalar J2ME desde <http://java.sun.com/j2me/download.html>. El entorno de desarrollo que nos provee el Wireless Toolkit se llama KToolBar.

Se debe construir paso a paso el MIDlet utilizando por ejemplo "J2ME Wireless Toolkit". Una vez creado el proyecto y el archivo, KToolBar crea la estructura de directorios necesaria para albergar el proyecto. Cada una de las carpetas creadas tiene una misión concreta. Por ahora, bastará saber que nuestros archivos fuente irán emplazados en el directorio src, y los recursos necesarios como gráficos, sonidos, etc. se alojarán en el directorio res.

A diferencia de otros entornos de programación, KToolBar no cuenta con un editor integrado para editar los programas; por lo tanto, se debe utilizar uno externo. Se puede utilizar JBuilder, JCreator o su editor favorito, que tenga soporte para Java.

En el editor, se crea el programa y se lo guarda en el directorio src que ha creado KToolBar. Una vez hecho esto, se retorna al entorno KToolBar y se comienza la compilación del archivo. Si todo va bien, aparecerá el texto Build Complete. Ya tenemos nuestro programa compilado y podemos ejecutarlo en el emulador. En el desplegable Device, se puede seleccionar el emulador que desee utilizar. El DefaultColorPhone tiene soporte de color, así que le resultará más atractivo. Pulse el botón Run. Verá aparecer un emulador con forma de teléfono móvil. En la pantalla del móvil aparecerá ya la ejecución del programa.

Una vez realizada la comprobación de que el programa funciona en el emulador, estamos listos para empaquetarlo y descargarlo a un dispositivo real. En KToolBar existen herramientas que permitirán crear el package, es decir archivos .jar y .jad dentro del directorio bin. Estos son los archivos que habremos de transferir al teléfono móvil. Existen varias formas de hacerlo, dependiendo de la marca y modelo del dispo-

sitivo. Si el teléfono tiene soporte de infrarrojos o bluetooth y el ordenador tiene puerto IrDA o bluetooth, se podrá transferir el archivo fácilmente sin necesidad de cable alguno. Si no, tendrá que recurrir a un cable de datos (consulte el manual de su teléfono). Otra posibilidad es poner los archivos JAR y JAD en un servidor wap o un espacio web y descargarlos desde el móvil. Para ello, es necesario que el dispositivo tenga un navegador wap o web y soporte GPRS para una descarga fiable.

CONCLUSIÓN

El presente artículo pretende hacer notar lo cerca que tenemos ahora con el avance de la tecnología y las herramientas TIC, al conocimiento. Sólo basta tener acceso a una computadora conectada a internet y el tiempo necesario para investigar algún tema en particular.

REFERENCIAS BIBLIOGRÁFICAS

- (1) <http://java.sun.com/javame/index.jsp> (15 de mayo de 2010)
- (2) <http://developers.sun.com/mobility/midp/articles/wtoolkit/> (18 de marzo de 2010)

Bibliografía Consultada

- Oracle, Sun Developer Network (SDN), 2010, <http://java.sun.com/javame/index.jsp> (15 de mayo de 2010)
- Oracle, Sun Developer Network (SDN) – Wireless Development Tutorial, 2006, <http://developers.sun.com/mobility/midp/articles/wtoolkit/> (18 de marzo de 2010)