

Mobile Processing Workshop I Versión 0.1

Mary Jane Soft
<http://mjs.darkgreenmedia.com>

Copyright © 2005 Mary Jane Soft - Marlon J. Manrique
<http://mjs.darkgreenmedia.com>
<http://marlonj.darkgreenmedia.com>

Productos y compañías mencionados aquí pueden ser marcas registradas.

No se asume responsabilidad por cualquier error u omisión presente en este documento, ni tampoco se asume responsabilidad por daños ocurridos al utilizar la información contenida en este documento.

Este documento puede cambiar sin previo aviso.

Cambios :

Fecha	Version	Descripción
Noviembre 9 de 2005	0.1	Version Inicial

Contenido

1 - Introducción.....	1
2 - Mobile Processing.....	2
3 - Preparativos.....	3
4 - Ambiente de Desarrollo.....	5
5 - Primeros Sketches en Mobile Processing.....	7
5.1 - Cuadro Estatico.....	7
5.2 - Cuadro Movil	8
6 - Dibujando en la Pantalla.....	9
6.1 - Ejemplo Dibujo :	9
6.2 - Ejercicio Dibujo :	9
7 - Movimiento	10
7.1 - Ejemplo Movimiento :	10
7.2 - Ejercicios Movimiento :	10
7.2.1 - Formas en Movimiento.....	10
7.2.2 - Balón.....	10
8 - Interaccion.....	11
8.1 - Ejemplo Cuadro Móvil	11
8.2 - Ejercicio Cuadro Móvil II.....	11
9 - Libreria de Sonido.....	12
9.1 - Ejemplo Reproducción de Sonido :	12
9.2 - Ejercicio Balón Sonoro :	13
10 - Autor.....	14
11 - Enlaces.....	14

1 - Introducción

Este documento ha sido creado como guía para el taller I de programación en Mobile Processing, dirigido especialmente a desarrolladores de software y cubre desde las etapas de instalación, pasando por la creación de sketches básicos, hasta la utilización de multimedia.

Este taller está creado pensando en los usuarios de la plataforma Windows, pero también puede ser utilizado en la plataforma Mac OS X o Linux con pequeños cambios.

Las versiones más actuales de este documento las puede encontrar en el sitio oficial de Mary Jane Soft (<http://mjs.darkgreenmedia.com>).

Si desea enviar correcciones gramaticales, sintácticas, ortográficas o de contenido puede enviarlas por correo electrónico a mjs@darkgreenmedia.com

2 - Mobile Processing

Mobile Processing es un ambiente de desarrollo para dispositivos móviles que permite desarrollar de forma rápida aplicaciones y prototipos. Es un hijo del proyecto de fuente abierta Processing originalmente diseñado para crear aplicaciones de escritorio o web.

El ambiente de desarrollo permite crear fácilmente aplicaciones con un solo click e implementa un subconjunto de las librerías de Processing optimizadas para su utilización en dispositivos móviles además provee un api de alto nivel para el manejo de gráficos, sonidos, y características específicas de los dispositivos como conexiones inalámbricas

El proyecto Mobile Processing, al igual que su padre conserva su característica de fuente abierta y crece cada vez mas gracias a las contribuciones de la comunidad.

3 - Preparativos

Antes de comenzar a utilizar Mobile Processing, debemos conocer los requerimientos básicos para su instalación y funcionamiento.

Primero se debe instalar el ambiente de desarrollo el cual puede bajar del sitio oficial del proyecto :

<http://mobile.processing.org/download>

Antes de comenzar a utilizar la herramienta se debe instalar el conjunto de herramientas para el desarrollo de aplicaciones para móviles el cual se utiliza para ejecutar las aplicaciones en emuladores de los dispositivos móviles.

Los usuarios de la plataforma Windows pueden utilizar el Sun Wireless Toolkit distribuido gratuitamente por Sun Microsystems y el cual pueden obtener en la siguiente dirección :

<http://java.sun.com/products/j2mewtoolkit/>

Los usuario de Mac OS X pueden utilizar el MPowerPlayer SDK distribuido gratuitamente y el cual pueden obtener en la siguiente dirección :

<http://www.mpowerplayer.com/products-sdk.php>

Después de instalar el software se procede a obtener la distribución de Mobile Processing que llene sus necesidades (existen tres diferentes distribuciones).

- La distribución Standard que deben utilizar aquellos usuarios que no son expertos en el desarrollo de software, la cual contiene la plataforma Java necesaria para la ejecución del ambiente de desarrollo.
- La distribución sin Java (para expertos), que pueden utilizar los usuarios que ya tengan instalada una versión del kit de desarrollo de aplicaciones en Java o JDK.
- La distribución para Mac OS X si usted posee esta plataforma.

Después de bajar la distribución mas apropiada para sus necesidades, simplemente descomprima el contenido en un nuevo directorio llamado mobile-processing dentro de cualquier directorio dentro de su maquina, por ejemplo la raíz de su directorio de

usuario o de alguna partición o disco donde usualmente coloque las aplicaciones.

A continuación y con la ayuda del explorador de archivos ejecute la aplicación mobile simplemente haciendo doble click al ejecutable.

En ese momento se abrirá la ventana principal del ambiente de desarrollo, para terminar de instalar la aplicación vaya al menú File y a la opción Preferences (o simplemente pulse CTRL-,) y en la pestaña Mobile suministre la ubicación del Wireless Toolkit antes instalado seleccionando la opción Browser o simplemente copiando la ruta de este en el cuadro de texto, al terminar presione el botón OK para continuar.

Ya esta listo el ambiente de desarrollo para comenzar a escribir sus primeros sketches.

4 - Ambiente de Desarrollo

Mobile Processing posee su propio ambiente de desarrollo o IDE el cual simplemente es un editor texto con sintaxis resaltada que permite escribir y ejecutar sketches. Este ambiente de desarrollo posee las típicas opciones para almacenar y recuperar sketches, exportarlos, ejecutarlos, visualizar errores de sintaxis.



Illustration 1: IDE Mobile Processing

Este IDE posee una barra de herramientas que es muy fácil de manipular, con seis diferentes comandos: Ejecutar, Detener, Nuevo, Abrir, Guardar, Exportar.







 Ejecutar	Ejecuta el sketch en edición, generando el código java correspondiente, lo compila, abre la ventana de visualización y finalmente lo ejecuta.
 Detener	Detiene la ejecución del sketch, pero no cierra la ventana de visualización.
 Nuevo	Crea un nuevo sketch (proyecto)
 Abrir	Permite seleccionar y cargar sketch previamente almacenados. Un menú se despliega permitiendo seleccionar sketches de su directorio de usuario, ejemplos o desde cualquier parte de su computador e incluso desde la red.
 Guardar	Permite actualizar el código del sketch en edición, es muy importante siempre actualizar el código antes de ejecutarlo. Para almacenar el sketch con otro nombre puede utilizar la opción del menú de archivo, Guardar Como
 Exportar	Exporta el sketch como un MIDlet listo para ser instalado en un dispositivo móvil

Tabla 1: Barra de Herramientas

Comandos adicionales pueden ser encontrados en los menús: Archivo, Edición, Sketch, Tools, Help, que permiten realizar diferentes tareas.

5 - Primeros Sketches en Mobile Processing

Con la creación de estos primeros sketches se espera que el usuario identifique los pasos para la edición y ejecución básica de pequeñas aplicaciones.

5.1 - Cuadro Estatico

Para conocer mas como desarrollar aplicaciones en Mobile Processing procederemos a crear un programa sencillo para identificar las formas básicas de codificación del sketch.

Para cumplir esta tarea se debe haber instalado el IDE de Mobile Processing disponible en la pagina oficial del proyecto (<http://mobile.processing.org>)

Se ejecuta el IDE y simplemente se copia el siguiente programa :



```
rect (20,20,width-40,height-40);
```

Presione el botón ejecutar del ambiente de desarrollo, esto hará que se compile el sketch utilizando la herramienta de compilación para crear el MIDlet y ejecutando el código en el emulador.

En este caso las dimensiones de la pantalla están dadas por el dispositivo y no se pueden cambiar, entonces para pintar un cuadro en la mitad de la pantalla se debe recurrir a las constantes `width` y `height` que almacenan el ancho y alto de la pantalla del dispositivo.

Illustration 2:
Cuadro Estatico en Mobile Processing

Después de exportar el sketch este puede ser probado en diferentes emuladores de dispositivos para probar la correcta ejecución de este y su comportamiento en diferentes plataformas de hardware.

A este modo de codificación se le llama modo básico.

5.2 - Cuadro Movil

El siguiente sketch muestra un ejemplo del cuadro móvil en el emulador.

```
int x,y;
void setup() {
  x = width/2 - 10;
  y = height/2 - 10;
}
void draw() {
  background(200);
  rect(x,y,20,20);
}
void keyPressed() {
  if(key == '8') y += 10;
  if(key == '2') y -= 10;
  if(key == '4') x -= 10;
  if(key == '6') x += 10;
}
```

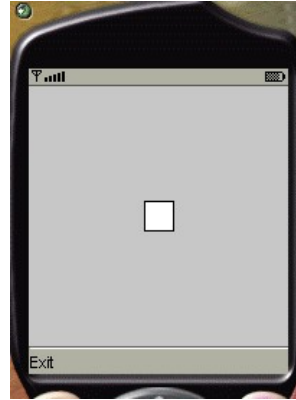


Illustration 3: Cuadro Movil en Mobile Processing

En este ejemplo se pueden observar tres diferentes construcciones setup, draw y keyPressed que son funciones que son ejecutadas por la aplicación móvil de la siguiente manera :

- La función setup es ejecutada una vez cuando el programa comienza a ejecutarse. Es utilizada para definir propiedades iniciales del ambiente como el color de fondo, cargar imágenes, inicializar variables. Solo puede existir una función setup por sketch.
- La función draw es ejecutada continuamente y es utilizada para dibujar elementos en la pantalla. El numero de veces que la función draw es ejecutada puede ser controlada por con la función delay o suministrando el numero de marcos por segundo con la función framerate.
- La función keyPressed es llamada cada vez que el usuario presiona una tecla, la variable del sistema key siempre contiene el valor de la tecla presionada mas recientemente, si se desea detectar las teclas de juego se debe utilizar la variable keyCode.

A este modo de codificación se le llama Standard.

6 - Dibujando en la Pantalla

Existen diferentes funciones que permiten dibujar en la pantalla entre ellas están :

- `background(r,g,b);` o `background(color);`
Causa que el color de fondo de la aplicación cambie, se puede utilizar con los valores de los componentes rojo, verde y azul.
Ejemplo :
`background(0);` // hace que la pantalla tenga un color negro de fondo
`background(255,0,0);` // hace que la pantalla tenga un color negro de fondo
- `stroke(r,g,b);` o `stroke(color);`
Causa que el color de las líneas y de los bordes cambie.
- `point(x,y);`
Dibuja un punto en la pantalla con el color del stroke
- `line(x1,y1,x2,y2);`
Dibuja una línea entre el punto de coordenadas (x1,y1) y el punto (x2,y2)
- `rect(x1,y1,width,height)`
Dibuja un rectángulo en las coordenadas suministradas con el ancho y alto suministrado.
- `fill(r,g,b);` o `fill(color);`
Cambia el color de relleno.

6.1 - Ejemplo Dibujo :

```
background(0,0,0);  
stroke(255,255,255);  
line(0,0,60,40);  
stroke(255,255,0);  
line(30,50,100,100);
```

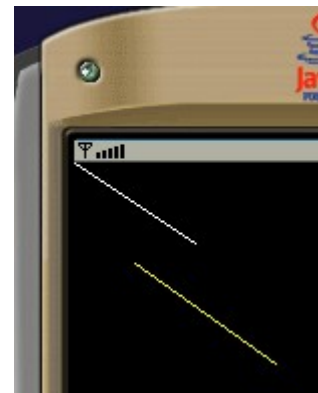


Illustration 4: Ejemplo Formas

6.2 - Ejercicio Dibujo :

Dibujar diferentes formas en la pantalla utilizando las funciones descritas anteriormente, modificar el color del fondo de la pantalla y del color de cada figura utilizando fill y su contorno utilizando stroke.

7 - Movimiento

Para darle movimiento a las formas en la pantalla es necesario utilizar el modo Standard de codificación haciendo uso de la función draw para dibujar las formas en lugares diferentes y dependiendo de la creatividad del usuario crear efectos visuales sencillos pero impactantes.

7.1 - Ejemplo Movimiento :

```
int x = 0;
void setup() {
    noStroke();
}
void draw() {
    background(190);
    rect(x, 0, 5, height);
    x++;
}
```

Este ejemplo visualiza un rectángulo blanco que se mueve de izquierda a derecha.

7.2 - Ejercicios Movimiento :

Para realizar este ejercicio el usuario puede obtener ayuda de la pagina de referencia de Mobile Processing que contiene documentación acerca de cada unas de las funciones disponibles (<http://mobile.processing.org/reference>)

7.2.1 - Formas en Movimiento

- Dibujar varias formas que se mueven por la pantalla.
- No permitir que las formas se salgan de la pantalla (utilice la construcción del lenguaje llamada if)

7.2.2 - Balón

- Dibujar un balón utilizando la función ellipse que rebote en los límites de la pantalla.

8 - Interaccion

Una característica primordial de las aplicaciones es su posibilidad de interactuar con los usuarios. Una de las formas básicas es a través del teclado el cual en la mayoría de los teléfonos móviles se conoce como el keypad el cual contiene las teclas numéricas

Mobile Processing provee diferentes funciones para capturar los eventos de teclado :

- La función `keyPressed` es llamada cuando se presiona una tecla cualquiera.
- La función `keyReleased` es llamada cuando la tecla antes presionada deja de serlo.
- La variable `key` contiene el carácter que representa la última tecla pulsada
- la variable `keyCode` contiene un código con la representación de la última tecla pulsada.

8.1 - Ejemplo Cuadro Móvil

El ejemplo del cuadro móvil presenta la funcionalidad de interacción descrita aquí, ya que utiliza el método `keyPressed` para identificar que tecla numérica fue pulsada, para realizar el movimiento.

8.2 - Ejercicio Cuadro Móvil II

En este ejercicio debe cambiar la implementación de la función `keyPressed` para que los movimientos sean realizados con las teclas de dirección UP, DOWN, LEFT y RIGHT (para esto debe utilizar `keyCode`)

```
if (keyCode == DOWN) y += 10;
```

Cambie también la implementación para que el cuadro se mueva mientras las teclas de dirección están presionadas.

9 - Librería de Sonido

También es posible adicionar nuevas funcionalidades a Mobile Processing, a través del uso de librerías que dan soporte a características especiales de los dispositivos o incluyen nuevas funcionalidades que no posee el núcleo.

Estas librerías pueden ser suministradas con la distribución oficial, o pueden ser obtenidas de terceros que han contribuido con su creación.

Una de estas librerías es la librería que permite reproducir sonidos y la cual esta incluida en la distribución.

Para poder utilizar esta librería se debe importar esta al espacio de trabajo, lo cual se hace seleccionado el menú Sketch, presionando la opción Import Library y finalmente haciendo click sobre la librería que se desea importar.

Utilice los tipos de datos y funciones que provee la librería para acceder a las nuevas características.

9.1 - Ejemplo Reproducción de Sonido :

En este ejemplo crearemos un sketch que simplemente reproduce un sonido cuando se pulsa una tecla cualquiera.

Utilizaremos la librería sound, la cual contiene un tipo de datos llamado Sound el cual puede alojar un sonido en formato wav o midi dependiendo del dispositivo y el cual se puede reproducir utilizando la función play.

Ejecute los siguientes pasos :

- Cree un nuevo sketch y guarde en el directorio de sketches.
- Consiga un archivo wav que desee reproducir en la aplicación.
- Cree el directorio data dentro del directorio del sketch y copie allí el archivo .wav que desea reproducir, renómbrelo como sonido.wav

- Copie el siguiente código en el sketch :

```
import processing.sound.*;
Sound s;
void setup() {
    s = new Sound("sonido.wav");
}
void keyPressed() {
    s.play()
}
```

- Pruebe la aplicación.

9.2 - Ejercicio Balón Sonoro :

Utilizando el sketch creado en el ejercicio del balón que rebotar, reproducir un sonido cada vez que el balón rebota.

10 - Autor

Marlon J. Manrique :



Docente en la Universidad Autónoma de Manizales, desarrollador de aplicaciones en Mary Jane Soft, entusiasta de la música y el cine. Posee el título de Ingeniero de Sistemas y Telecomunicaciones, programador certificado en Java e instructor certificado en Java.

<http://marlonj.darkgreenmedia.com>
marlonj@darkgreenmedia.com

11 - Enlaces

- **Mobile Processing**
Sitio Oficial
<http://mobile.processing.org>
- **Java 2 Platform, Micro Edition**
Sitio Oficial Plataforma J2ME
<http://java.sun.com/j2me>
- **Mary Jane Soft**
Información adicional y librerías para Mobile Processing.
<http://mjs.darkgreenmedia.com>
- **Learning DarkGreenMedia**
Cursos y talleres en línea sobre Mobile Processing y otros temas.
<http://learning.darkgreenmedia.com>
- **Marlon J. Manrique**
Información adicional sobre el autor de este documento.
<http://marlonj.darkgreenmedia.com>