

Mobile Devices in Software Engineering

Lab 1

Objective

The objective of this lab is to:

1. To use and compare two different Integrated Development Environments for mobile applications
2. To compare an application on various mobile device emulators

Experiment 1

In this experiment you will compare two different IDEs that are available and view their different emulators and development environments.

Exercise 1

In this exercise we will download, install, and explore the Sun Microsystems Java Wireless Toolkit for CLDC. This is an IDE which contains documentation, emulators, and sample mobile applications that are compatible for CLDC/MIDP compliant mobile devices.

1. If required, download and install a Java Standard Edition Development Kit available at <http://java.sun.com/javase/downloads/index.jsp>
2. Download and install the Sun Java Wireless Toolkit for CLDC for Windows or Linux at <http://java.sun.com/products/sjwtoolkit/download.html>

Now we will explore the Wireless Toolkit which allows us to build mobile applications for compatible devices.

1. Run the Sun Java Wireless Toolkit for CLDC.
2. Click on **File -> Open Project**
3. Select the sample project called JBricks and click **Open Project**
4. Once the project has been loaded click **Build** and then click **Run**

The application with the emulator will open. Ignore the second warning screen asking you to run the application via OTA, this is not necessary for the purposes of this lab. Continue and play around with the application. After you have completing running the application, close the emulator and complete the following steps.

5. Click on the **Device** drop down menu in the Wireless Toolkit

6. Select the **QwertyDevice**
7. Click **Run** and explore the application again
8. Repeat the above steps with the other devices available

Exercise 2

In this exercise we will download, install, and explore another IDE available which is the Blackberry Java Development Environment (JDE). This IDE is specifically geared towards developing applications for Blackberry devices.

1. Download and install the Blackberry Java Development Environment which is available at <http://na.blackberry.com/eng/developers/downloads/ide.jsp>

NOTE: You will be required to fill out a form so you can download the application.

Once you have installed the software perform the following steps.

1. Run the JDE installed on the machine
2. Click on **File** -> **Open** and open the HelloWorld.java sample program located in ...\\samples\\com\\rim\\samples\\device\\helloworld\\HelloWorld.java
3. Click on **Build** -> **Build All and Run**
4. Navigate to the Applications selection and click **Enter**
5. Find the Hello World Demo and run the application
6. Close the Blackberry emulator and explore the HelloWorld.java code

What sort of differences do you see between the applications and emulators with respect to display, layout, key inputs, etc.? How does this impact design choices for developers?

Experiment 2

In this experiment we will load a java application onto a Blackberry device. We will be required to convert the .jar and .jad files into a .cod file which is suitable for Blackberry devices from a MIDlet that we will create. This can be done using the rapc compiler that comes with RIM's JDE. The "rapc" compiler can be accessed from the "bin" directory of RIM's JDE installation directory.

First we will create a MIDlet application. The following code can be found at <http://developers.sun.com/mobility/midp/articles/getstart/>.

HelloMIDlet.java

```
import javax.microedition.midlet.*;
import javax.microedition.lcdui.*;

public class HelloMIDlet extends MIDlet
    implements CommandListener {
    // The exit command
    private Command exitCommand;
    // The display for this MIDlet
    private Display display;
    // create a ticker
    private Ticker hi = new Ticker("J2ME is cool");

    public HelloMIDlet() {
        display = Display.getDisplay(this);
        exitCommand = new Command("Exit", Command.SCREEN, 2);
    }

    public void startApp() {
        TextBox t = new TextBox("Hello MIDlet",
                                "Wireless Internet", 256, 0);

        t.addCommand(exitCommand);
        t.setCommandListener(this);
        t.setTicker(hi); // set the ticker
        display.setCurrent(t);
    }
    public void pauseApp() { }
    public void destroyApp(boolean unconditional) { }
    public void commandAction(Command c, Displayable s) {
        if (c == exitCommand) {
            destroyApp(false);
            notifyDestroyed();
        }
    }
}
```

This code updates a MIDlet found in the Wireless Toolkit installation by adding a scrolling ticker at the top of the application. Take a moment to read over and understand the code.

Create and compile the application by following these steps:

1. Click on **New Project...**
2. Name the Project Name and MIDlet class name 'HelloMIDlet'
3. Proceed to the directory displayed in the Wireless Toolkit console instructing you the location to place the source files
4. Copy the above code into HelloMIDlet.java in the given folder

5. Save the file and then proceed to click the **Build** button in the Wireless Toolkit console
6. Click on **Run**

Now that we have created a MIDlet we will port this application to a Blackberry device.

1. Proceed to the 'bin' folder of the HelloMIDlet project and copy the .jad or .jar file to the 'bin' directory in the RIM JDE.
2. Follow the instructions in the following article to convert the files to a .cod file.
<http://developers.sun.com/techttopics/mobility/midp/articles/blackberrydev/>
3. We can now load the .cod file onto the Blackberry, follow these steps:
 - a. Get a BlackBerry device and a USB cable from the instructor
 - b. Connect the USB cable to the device and the desktop
 - c. Follow the instructions in the following article to use the 'javaloader' command
http://developers.sun.com/techttopics/mobility/midp/articles/blackberrydev
 - d. Disconnect the device from the cable
 - e. Verify that the application has been successfully loaded onto the device
4. After you have shown your work to the instructor remove the application from the device with the USB cable and the following command (javaloader -usb erase FileName.cod)
5. Return the device