

Appendix B: Electronic Resources

Data Collected and Recorded from Application Interaction and Experiments

Please visit <http://people.arch.usyd.edu.au/~yohann/mphil.htm> to view the data collected and recorded as a result of the experiments and interaction with the application.

Compiling the Source Code

In order to compile the source code, the Sun Wireless Toolkit must be installed. The current version is 2.5.1 and can be downloaded from <http://java.sun.com/products/sjwtoolkit/>.

1. Launch the KToolbar in Windows by clicking Start > Programs > Sun Java Wireless Toolkit 2.5 for CLDC > Wireless Toolkit 2.5. The KToolbar window will appear as shown in Figure 1.

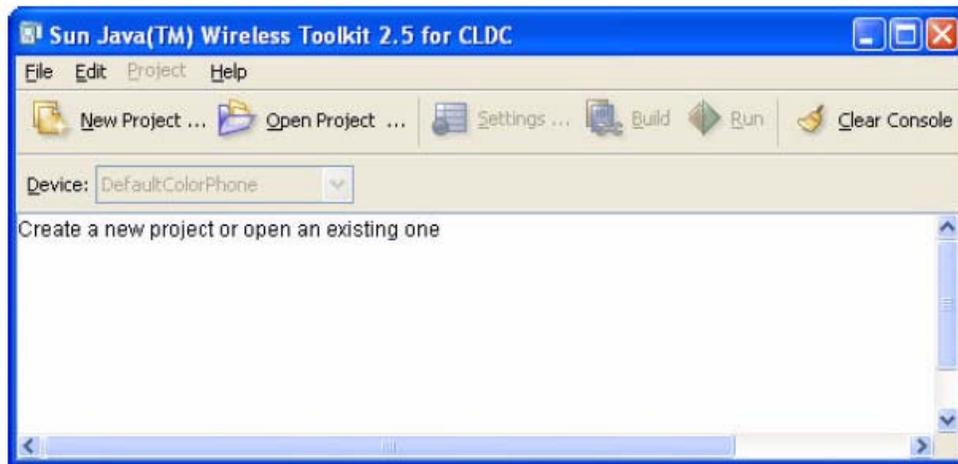


Figure 1. The KToolbar interface.

2. Press the New Project button (or click File > New Project)
3. A window prompt will display asking for the project name, enter `BT_RMS_CM`
4. It will also ask for the MIDlet class name, enter `BluetoothDiscovererMidlet`
5. Press the Create Project button
6. A Settings window will appear as shown in Figure 2. With API Selection tab highlighted in the left column, select Custom from the Target Platform drop down menu.
7. In the Profiles box, select MIDP 2.0. In the Configurations box, select CLDC 1.1. In the Optional box, tick the Bluetooth/OBEX for J2ME (JSR 82). Press the Ok button once all these settings are selected.

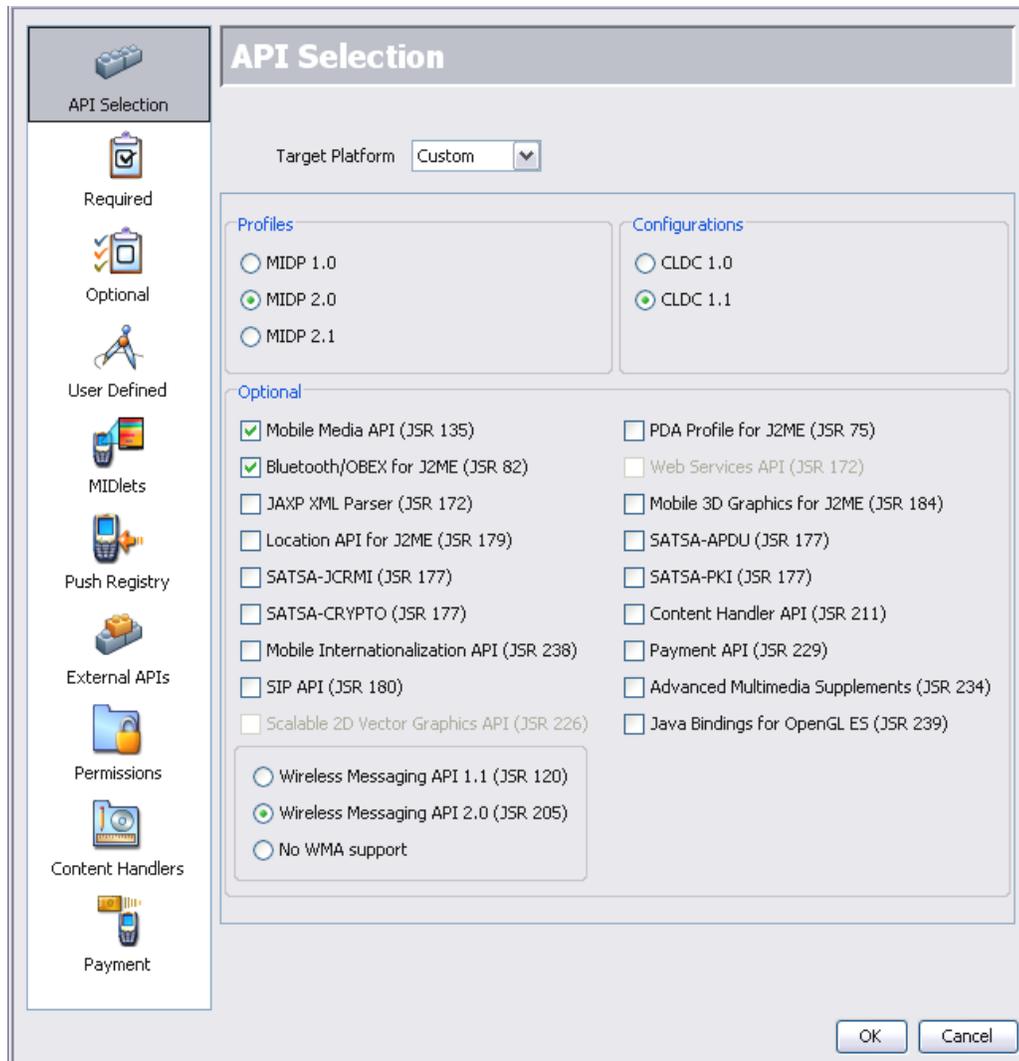


Figure 2. The project settings interface.

8. If you have not done so already, download the source code from <http://people.arch.usyd.edu.au/~yohann/mphil.htm> and unzip the file to a directory.
9. When a new project is created, the Sun Wireless Toolkit will create a new sub-directory named with the Project Name (in this case `BT_RMS_CM`) under the `apps` sub-directory where the Toolkit is installed (usually `C:\WTK25`). Within the sub-directory `BT_RMS_CM`, you will see a set of sub-directories: `bin`, `lib`, `res` and `src`. The `src` directory is where you will put all the Java code. The `bin` directory is the sub-directory where the compiled and packaged Java application will be placed.
10. Locate the directory where you have unzipped the files. Copy and paste the files into the `src` folder (`C:\WTK25\apps\BT_RMS_CM\src`).

11. Return to the KToolbar and press the Build button (or click Project > Build). All the Java code will compile and you will see a message in the KToolbar window that the Build is complete.
12. You have successfully compiled the code and will be able to run the emulator by pressing the Run button (or click Project > Run).

Packaging and Deploying the Application

1. Once the code is compiled successfully, you will need to package the code and deploy it onto a mobile device that supports the J2ME run-time environment and JSR-82.
2. In the KToolbar, click on Project > Package > Create Package. The KToolbar window will display the following message:

```
Building "BT_RMS_CM"  
Wrote C:\WTK25\apps\ BT_RMS_CM \bin\ BT_RMS_CM.jar  
Wrote C:\WTK25\apps\ BT_RMS_CM \bin\ BT_RMS_CM.jad  
Build complete
```

3. The jar file contains the compiled code. The jad file contains the description of the jar file. Both files reside in the in bin directory.
4. You will only need the jar file.
5. You have now successfully packaged the code.
6. To deploy and run the application on the target mobile device, you will need to use the mobile device manufacturer's software to transfer the jar file. This will be different for all mobile device manufacturers and is not explained in these instructions. Please refer to the mobile device software manual to complete this process for deploying and running the application.