

1) Prerequisites

To compile and run Lampiro with Eclipse you need to install and configure few plugins.

1. A **WTK** compatible with Eclipse as the Sun Java Wireless Toolkit (<http://java.sun.com/products/sjwtoolkit/>).
2. **Eclipseme** (<http://www.eclipseme.org/>). A nice tutorial that can be helpful to configure Eclipseme and the whole Eclipse environment can be found at <http://eclipseme.org/docs/installation.html>
3. **Proguard** (<http://proguard.sourceforge.net/>)
4. **Antenna** together with its **Eclipse plugin** (<http://antenna.sourceforge.net/> and <http://antenna.sourceforge.net/wtkpreprocess.php> at the end of the page).

It is not mandatory but can be useful to use the **Subclipse** plugin in order to connect to the SVN repository from within Eclipse (<http://subclipse.tigris.org/install.html>)

2) Install Lampiro

The easiest way to download and import Lampiro in your Eclipse environment is this procedure:

- Open Eclipse and right click in the project explorer
- Choose *"Import"* from the menu that is opened
- Click the *"Import"* command
- In the menu that appears choose *"Checkout projects from SVN"*
- Select *"Create new repository location"* and click Next
- Specify this location *"http://lampiro.googlecode.com/svn/trunk/"* and click Next
- In the *"Select folder"* pane select the *"lampiro"* folder
- Select *"Checkout as a project configured using the New Project Wizard"* and click Finish
- Select *"J2ME Midlet Suite"* and click Next
- In the *"Project name"* field specify *"Lampiro"*; make sure that *"Enable Preprocessing"* is selected and click Next
- Specify your favorite *"Device"* for emulation and click Finish

After this procedure you have to add the external libraries to your build path (bouncycastle and xmlpull) used by Lampiro: these libraries are included in the package and can be found in the *"lib"* subfolder of the project.

To add them to your build path you can use the following procedure:

- right click on *"lib/bouncycastle.jar"* in the project explorer and choose *"Add to build path"* from the *"Configure build path"* menu
- do the same procedure for *"lib/xmlpull_1_1_3_1.jar"*

Finally two defines must be added in your project: right click your project in the project explorer, select *"Antenna preprocessor"* and add *"MUC=1,UI=1"* in the defines textbox; meanwhile in the same pane the debug level should be reset to *"None"*.

3) Run Lampiro

To run or debug the Midlet in your Eclipse environment you can right click the LampiroMidlet.java file located in the "*src/lampiro*" folder in your project explorer window and click "Run as Emulated J2ME midlet" or "Debug as Emulated J2ME midlet".

4) Export lampiro.jar

The project comes with build.xml an Ant script file that can be run to compile and deploy Lampiro as .jar file.

To run it you can right click it and choose "*Run as Ant build*": before running it be sure that the options in the file "user-build.properties" fit your system configuration and needs (especially the ones pertaining the WTK tools location). Alternatively you can run the script by hand in a console window with the following command: "ant deployedSuite".

It is possible to use different switches in this procedure in order to enable compression and TLS. To use these features specify the following parameter in "Arguments:" when launching the external tool dialog or in the command line:

- "-propertyfile **eclipseme-build-compression**>" to have compression enabled
- "-propertyfile **eclipseme-build-TLS**>" to have compression and TLS enabled

The exported jar together with its .jad file can be found in the deployed subfolder of your project.

-->