Exercise 4 – Praktikum Mobile Productivity

Using Bluetooth and GPS

This exercise deals with the basics of programming Java ME applications that use Bluetooth. In particular you will use Bluetooth to access an external GPS-receiver and retrieve its data so that you can use it in your own Java ME applications. The following steps explain the different tasks of this exercise in more detail:

- For this exercise create a MIDlet that searches for Bluetooth-devices first. The Bluetooth-specification differentiates between searching for Bluetooth-devices (also called Inquiry) and searching for the Bluetooth-services that are offered by these devices. For this exercise it is sufficient to implement a Bluetooth-client that executes an Inquiry that finds and lists nearby Bluetooth-devices, among them the GPS-receiver.
- From the list of discovered Bluetooth-devices you should be able to select the GPS-receiver and connect to it.
- After that display an interface that shows you the positioning data that is acquired from the GPS-receiver through the Bluetooth-connection.

You can test your application with the GPS-receiver that is available from Gregor Broll in Amalienstr. 17, room 503.

Each student must check-in his/her own solution to his personal SVN folder by **Monday, November 13th, 12 p.m.** Create a sub-folder called *excercise4* and hand in a zip-file named *exercise4.zip*.
Please remember that this date is also the deadline for submitting your solutions for exercise 3.

Appendix: Hints and Details

- Programming with Java ME's Bluetooth API can sometimes be a little bit tricky and definitely benefits from good preparation. Please see [1] for a collection of useful articles on Bluetooth for Java ME. Especially [2] focuses on the usage of the core Bluetooth APIs while the JSR 82 Bluetooth API [3] provides detailed information about its different classes. As always, the CLDC and MIDP APIs are valuable resources for specific issues. They are available at [4].
- For the first part of the exercise, implement a classic Bluetooth Inquiry that finds all Bluetooth-devices in the vicinity of your device. For this exercise, you only have to implement a Bluetooth client and only handle the search for Bluetooth devices, not for the services they offer.
- Display a list with the names of all discovered devices from which you can choose the Bluetooth GPS-receiver in order to connect to it.
- As for the second part of the exercise, retrieve the Bluetooth Address from the GPS receiver (it should be something like 0002?????C56 ;-) and use it to establish a stream connection at port 1. Connecting to this address is sufficient for getting GPS-data from the receiver; no connections to any services are involved.
- Connecting to the GPS-receiver using its Bluetooth address is quite similar to connecting to the internet (keyword GCF). Use the URL format for RFCOMM stream connections.
- Similar to connections to the internet (see exercise 3), use different threads for searching the Bluetooth-devices and for retrieving data from the GPS-receiver.
- The data that you will get from the GPS-receiver uses the NMEA-0183-format [5]. Please parse the received, comma-separated data according to the specific format before displaying it on the screen.
- Please consider during your implementation that you can't test Bluetooth from your Java ME emulators, but only on real devices. Thus please test your applications at the university and share resources by testing them in groups, since you also have to share only 1 GPS-receiver.

Resources

[1] http://developers.sun.com/techtopics/mobility/allarticles/#bluetooth
[2] http://developers.sun.com/techtopics/mobility/apis/articles/bluetoothcore/index.html
[3] http://java.sun.com/javame/reference/apis/jsr082/
[4] http://java.sun.com/javame/reference/apis.jsp
[5] http://www.kowoma.de/gps/zusatzerklaerungen/NMEA.htm