

2D User Interfaces - Exercises

Tutorial 1

Installation of Eclipse and J2ME

Marcus Tönnis, Patrick Maier

Fachgebiet Augmented Reality (FAR)
Technische Universität München

Lecture + Exercises -> Exam

- *Designing the User Interface (digest)*
 - Human Factors
 - Evaluation of User Interface Designs
 - Interaction Styles :Direct Manipulation; Menu Selection Form Filling, ...
 - Interaction Devices
- Exercises
 - Experience issues arising during system development
 - Learn by implementing alternatives and modifications
- Exam
 - Exercises constitute half of the credits and thus make up half of the grade
 - Exam questions based on experiences gained by attending the exercises
 - It will NOT be enough to just read and understand the homework

Matching Exercises - Lecture

- Lecture
 - Theory & Principles
 - Interfaces not yet found in common applications
 - Focus on desktop computers
- Exercises
 - Handmade implementation of often used structures in renewed design shapes
 - BUT: What works on desktop computers can also run on mobile phones
 - The integrated development environment simulates them
 - But you can also deploy them to your mobile phone!!

Overview of Exercises

Iteration between tutorials and 'lab days'

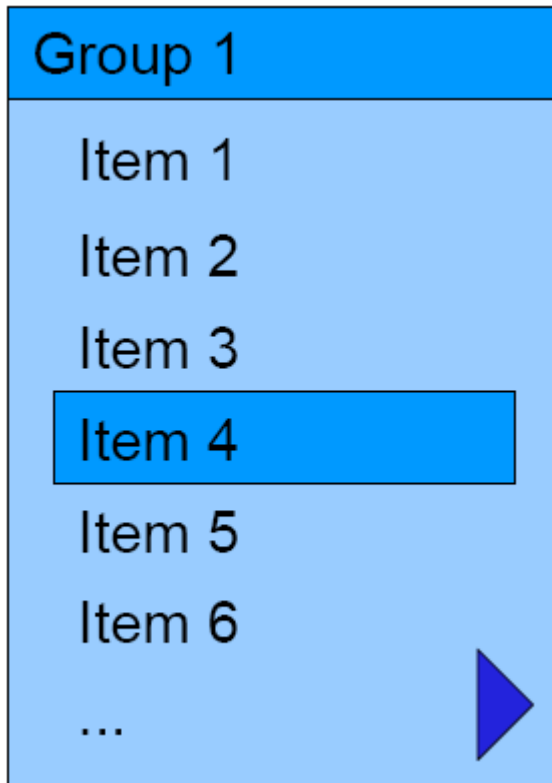
- 16.10.08 Installation of Eclipse and J2ME: Implement HelloWorld Midlet
- 23.10.08 Canvas, Layer, Sprite and LayerManager: Implement SimpleList
- 30.10.08 lab day
- 06.11.08 Loading and storing real-world data: Implement List and Data Storage
- 13.11.08 lab day
- 20.11.08 Mobile Media API: Implement Fisheye List
- 27.11.08 lab day
- 04.12.08 Creating a treeview: Implement Tree
- 11.12.08 lab day
- 08.01.08 Evaluation of input concepts: Conduct Evaluation
- 15.01.08 lab day
- 29.01.08 Final Presentation

Goal of Exercises

- Have implemented some variations of UIs to browse through hierarchical data sets and have them evaluated
 - Hierarchical data sets
 - e.g. Music
 - Genre: Rock
 - Interpret: AC/DC
 - » Album: High Voltage
 - » Track: 5 – T.N.T.
- Evaluation
 - Run your solutions with at enough people to perform a descriptive analysis
 - Compute which one performs best and illustrate significant statements

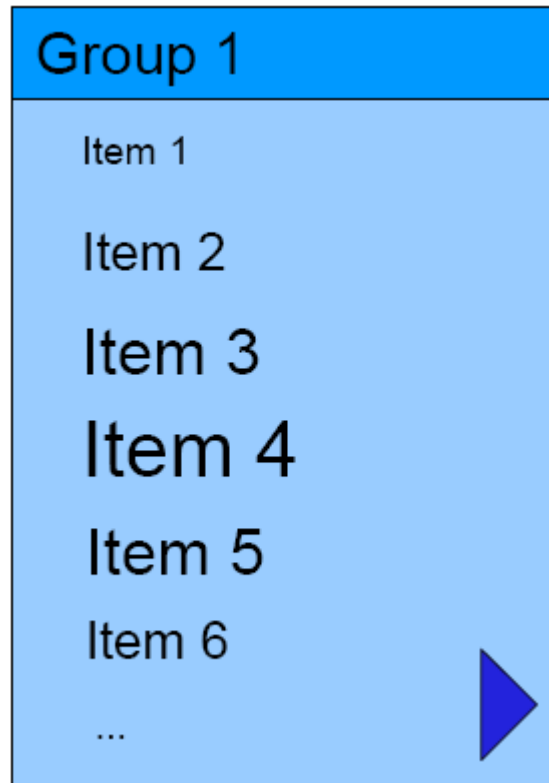
Types of UIs

Simple List



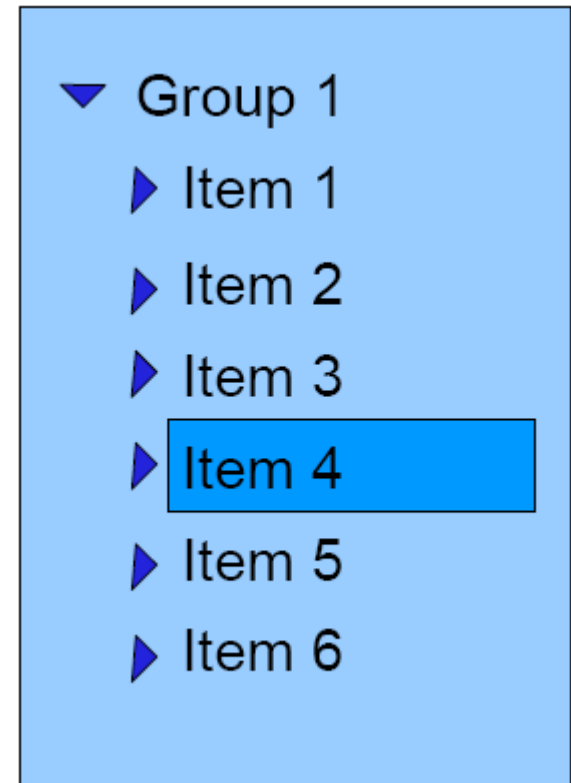
A vertical list of items under a blue header 'Group 1'. The items are 'Item 1', 'Item 2', 'Item 3', 'Item 4', 'Item 5', 'Item 6', and '...'. 'Item 4' is highlighted with a blue background. A blue right-pointing arrow is at the bottom right.

FishEye List



A vertical list of items under a blue header 'Group 1'. The items are 'Item 1', 'Item 2', 'Item 3', 'Item 4', 'Item 5', 'Item 6', and '...'. 'Item 4' is significantly larger than the others. A blue right-pointing arrow is at the bottom right.

TreeList



A tree view under a light blue header. A blue downward-pointing triangle is to the left of 'Group 1'. Below it are 'Item 1', 'Item 2', 'Item 3', 'Item 4', 'Item 5', and 'Item 6', each preceded by a blue right-pointing triangle. 'Item 4' is highlighted with a blue background.

Development Environment

- We do Java
 - With Eclipse
 - And with the Sun Java Wireless Toolkit (formerly JavaME, J2ME)
- Eclipse: <http://www.eclipse.org/downloads/>
- J2ME
 - Sun Developer Network requires registration... (can do - or)
 - Go to <http://ar.in.tum.de/static/files/teaching/08ss/2dui/j2me/>
 - Most likely, you'll want one of the topmost two files (Version 2.5.2)
 - In case they don't work with your Java installation: try 2.2 + Update

EclipseME

- Integration of J2ME with Eclipse
- Can be installed through Eclipse
- Instructions:
<http://eclipseme.org/docs/installEclipseME.html>
- Configuration: <http://eclipseme.org/docs/configuring.html>
- If you are using MacOS: see
 - <http://developers.sun.com/mobility/midp/articles/osx/>
 - <http://www.urbanhill.net/midp/>

Our First Midlet

- Start a new project:
<http://eclipseme.org/docs/createProject.html>
- Create a Midlet:
<http://eclipseme.org/docs/createMidlet.html>

Our First Midlet

```
public class HelloMidlet extends MIDlet implements CommandListener {
    private Command exitCommand;
    private TextBox tbox;

    public hellomidlet() {
        exitCommand = new Command("Exit", Command.EXIT, 1);
        tbox = new TextBox("Hello MIDlet!", "Hello World!", 25, 0);
        tbox.addCommand(exitCommand);
        tbox.setCommandListener(this);
    }

    protected void startApp() {
        Display.getDisplay(this).setCurrent(tbox);
    }

    protected void pauseApp() {}

    protected void destroyApp(boolean bool) {}

    public void commandAction(Command cmd, Displayable disp) {
        if (cmd == exitCommand) {
            destroyApp(false);
            notifyDestroyed();
        }
    }
}
```

Running a Midlet

- Running the Midlet in the emulator:
<http://eclipseme.org/docs/run.html>
- Running the Midlet on your phone:
<http://eclipseme.org/docs/packaging.html>
 - Copy *.jad and *.jar to your mobile device
 - Most phones will complain about an unsigned application – accept nevertheless
- Profit!

Exercise Assignment

- Experiment with your J2ME setup
- Refactor the initial Midlet so that you got clearly structured code
- Write a midlet which interacts with a second button, e.g. by changing the display message

Exercise Assignment

- When you finished your work
 - Come by in our office (with appointment) and present your work before the next tutorial is held (for today's exercise before the tutorial session on 23.10.08 at 13.00h)
 - If you are fast, you can present your solution in the lab session
 - Room: 03.13.035
- Q/A: maierp@in.tum.de / toennis@in.tum.de
- See you back here on 23.10.08 at 13.00h