

## Exercise 6 – Mensch-Maschine-Interaktion 1

### High-Fidelity Prototype for an Interactive Tabletop Application

As announced last week, you will develop a high-fidelity prototype for an interactive tabletop application in the next couple of weeks.

The interactive tabletop at the basement of Amalienstrasse 17 is camera-based. It returns the coordinates of **two touch points at the most** and one ID for each set of coordinates. The tabletop can not distinguish between people or objects. Also it cannot detect forms.

You will be programming in C#. As there is only one interactive table for all students, you will have to do most of the programming on your notebooks or PCs.

### Proceeding

- Collect the feedback you got from the evaluation of the paper prototype and factor it into an improved UI design and concept.
- Implement your prototype on your desktops / notebooks. See “Instructions and References” for further information.
- Demonstrate the prototype at the tabletop (December 17<sup>th</sup> / 18<sup>th</sup> / 19<sup>th</sup>).

### Prototype Demonstration

Here are the timeslots where the tabletop is available for you to test and demonstrate your prototype:

	Mo, 17.12.	Di, 18.12.	Mi, 19.12.
9-10	free	free	free
10-11	free	free	free
11-12	free	free	free
12-13		free	free
13-14		free	free
14-15		free	free
15-16		free	free
16-17		free	free
17-18		free	free

Please send an email to [Sara.Streng@ifi.lmu.de](mailto:Sara.Streng@ifi.lmu.de) to reserve one of these timeslots until **Friday, December, 7th**. Specify your group number and 4 time slots that would work for your group. Start with the one you prefer most.

If your prototype successfully runs on the interactive table at the end of the first session, your group is finished with the task. If not, you will get a second chance in the first week after the Christmas break. For the second try you will arrange time slots at the end of your first session.

## Instructions and References

- Download and install the Visual C# 2005 Express Edition:  
<http://www.microsoft.com/germany/msdn/vstudio/products/express/download.msp>
- Download the template: <http://www.medien.ifi.lmu.de/lehre/ws0708/mmi1/tabletop-template.zip>. This template allows you to attach multiple mice to one computer and thereby simulate the multi-touch functionality of the tabletop.
- Connect two external mice (touchpads won't work) to your computer.
- Double-click ttBarebone.sln and run the program. Each mouse gets one ID (either 20 or 22). When you press a mouse button, one of the disks (red or blue) should jump to the cursor position. While you keep the button pressed you should be able to move the disk. With the other mouse you should be able to control the other disk.

For each mouse event you should get one line in the Output Window (which you will see after activating: Main Menu – View – Output Window). The line should look like this:

Mouse 1:	Mouse 2:
20: Down Event	22: Down Event
20: Move Event	22: Move Event
20: Up Event	22: Up Event

- Be careful to always remember that the multiple mice are only a simulation. There are some differences compared to the tabletop, that you should always keep in mind:
  - Moving a mouse without clicking the left mouse button is not an event that will be handled by the table. Only if an object or finger touches the table, movement will be recognized.
  - While the mice have fix IDs, the IDs of objects on the tabletop are constantly changing. Therefore, do not use the mice IDs to identify people etc.
- If the template does not run make sure the references are set correctly.  
In the project explorer window (top right window) the references need to include
  - Interop.SBSDKComWrapperLib (Tabletop library)
  - RawInputSharp (Library which allows you to use multiple mice)Both libraries are in the *lib* folder of the template. If necessary remove and add them again.
- Integrate your prototype in the template. Mouse or tabletop event handlers both trigger the following methods in the *Controller* class:
  - `public void Down(int x, int y, int id)`
  - `public void Move(int x, int y, int id)`
  - `public void Up(int x, int y, int id)`You can define the handling of these events in these three methods. However, do not change the regions **Smartboard Event Handlers** or **Mouse Event Handlers**!  
If you don't change these parts of the template, running your working prototype on the tabletop should not be a problem.
- To find information about C# visit
  - <http://msdn2.microsoft.com/de-de/vcsharp/default.aspx>
  - <http://www.codeproject.com/>