Bas Bakx s123612

DG-308 Discrete Interaction Design

Matthias Rauterberg

# Objective

The objective of Discrete Interaction Design was learning to communicate the feel of a system with designers creating a prototype. This prototype can be used to do usertests, improving the system before it’s reaching its final development stages.

The focus of the assignment really was on the user, how to make sure that a system is appropriate for a user? This question was an integral part of the design process; how can one communicate a product feel with a designer and how can a designer give shape to this concept in a meaningful, user friendly way?

# Process

## Step 1: Specification

At the beginning of the assignment I was teamed up with Max Bogers, Pieter Bron and Rik Burger. We were asked to propose a new product and communicate it using a Petri Net.

Because of a misinterpretation within our group we recreated an existing concept and made a Petri Net specifying it. Even though technically wasn’t the correct choice, it was a good exercise in formally specifying systems. We decided on specifying a Microsoft Kinect interaction, the movement tracking options appealed to us because of the wide range of interaction it involves, we could explore the borders of what users find awkward and to what extend they are prepared to follow Kinect instructions.

We chose a game in which the user was standing in front of a glass wall, on the other side of the wall was water. At some points 1, 2 or 3cracks would appear in the glass and users had to gesture their hands and feet towards the cracks within a few seconds, or else they lost the game. If they succeeded in closing the cracks a new set of cracks appeared.

We decided to model the interaction of three sets of cracks. We made a Petri Net using the program CPN tools

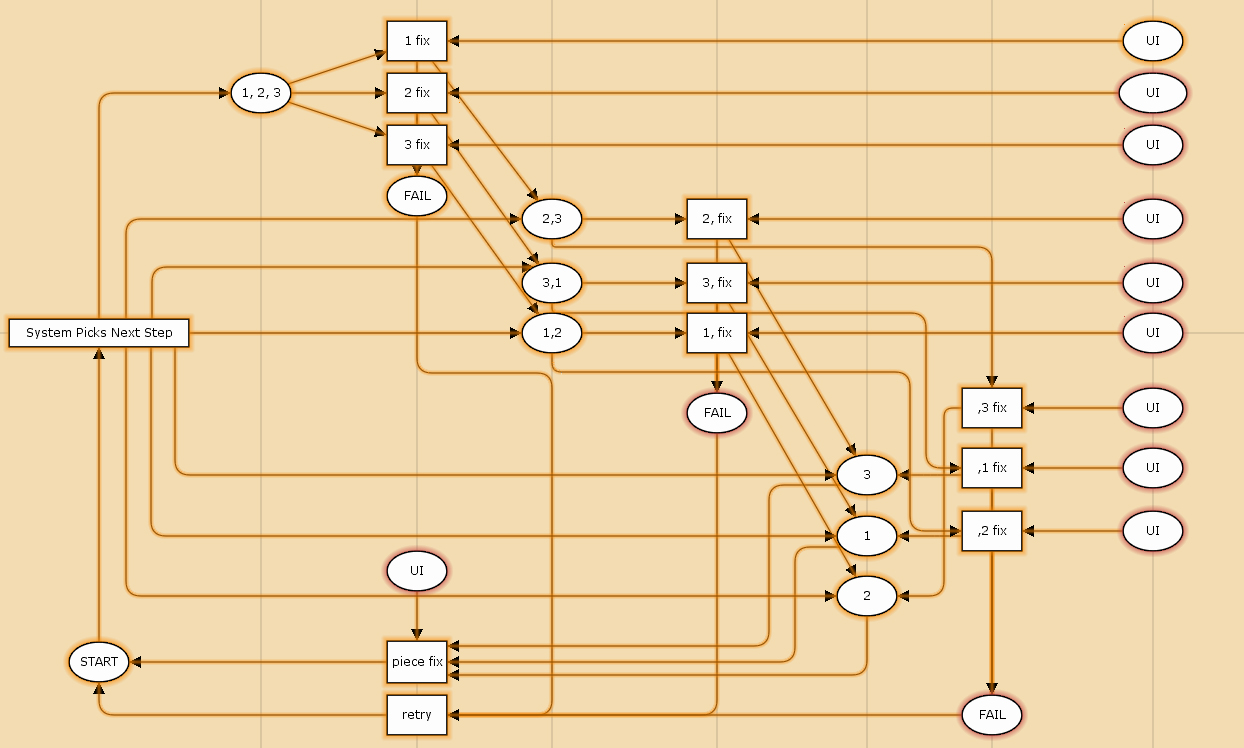


Figure : First Petri Net

When we were attended to the fact that a model of a not yet existing system was required we got together and did so. Still staying with the Kinects we decided to model a tool for the Kinect. The Tool would have to test to what movement users are prepared and able. It was specified as follows:

Figure : Kinect Calibration Specification

**Calibration tool Kinect**

Imagine a Kinect in a public setting (for example in a supermarket or in a waiting line for a club). Depending on how comfortable you feel in the environment you are prepared to do “crazier” motions. For example: you might feel more comfortable jumping when you’re with friends instead of jumping when you’re around strangers.

The concept is to have a screen precede the game that asks the player to perform a certain action, like standing on one leg stretching both hands into the air. When a player doesn’t fully stretch his arms and leaves his feet on the ground, the system will know that this player can’t or won’t use his legs and doesn’t want to move his arms too far in the game following.

The tool would enable the game to adapt the necessary motions to the user preferences. So if a user wants to play tennis and is really lazy, the Kinect transforms tiny wrist motions in real life into grand swinging motions on screen.

**The calibration tool works as follows:**Willingness of expression has to be measured per limb. So the Kinect asks you to strike a pose or perform an action which requires you to use all your limbs. While doing so, the Kinect scans your limbs. First, the expressiveness of the left arm is scanned and the scale of expressiveness (0,1,2,3) is sent to the Kinect (0 means there is no arm (Or it’s not used at all), 3 means the user is using its arm as much as possible, i.e. completely stretched). Then the same happens for the other limbs. After which the user can choose to continue the game, which then retrieves the data from the Kinect and uses it to determine the gameplay. If a user has the variables set low, the game will change accordingly.

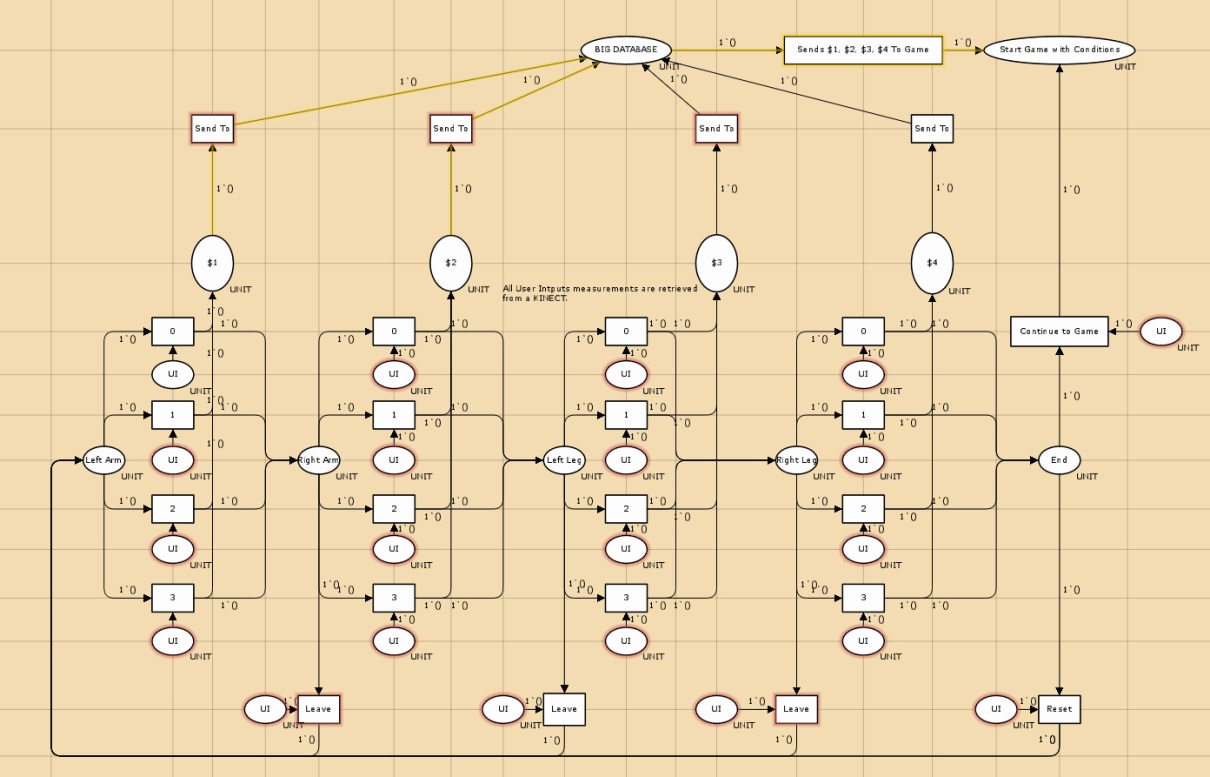


Figure : Second Petri Net

## Step Two: Designing

After finishing the specification part of the assignment the group was split, leaving me with Max to elaborate on a concept we got assigned.

The concept we got assigned was that of a smart room, the specification was as follows:

Explanation Intelligent Room

**How the interface works**

The user starts at ‘start menu’.

From there, the user can choose from 3 options:

* select preset
* add/delete preset
* adjust manually

**select preset**

when the user selects “select preset”, he has 4 options:

* preset 1
* preset 2
* preset 3
* preset x

When the user selects either one, the choice is stored in the database, from which the system knows how to control the room’s atmosphere. After that, the interface goes back to the start menu.

**Add/delete preset**

When the user selects “add/delete preset”, he has 3 options

* add
* delete
* back to start menu

When the user wants to delete a preset, he selects “delete” and after that selects the preset he wants to delete. This preset will be deleted from the database. After that, the interface will go back to the “add/delete preset” menu.

When the user wants to add a preset, he selects “add” after which the new preset is made and stored in the database. The interface then leads the user through a couple of steps to select the preferences you want. Al those preferences will be stored in your new preset.

**Adjust manually**

The user can also choose to adjust the presets he has already stored manually. He then selects the preset he wants to adjust, and adjusts the values he wants to. These changes are stored in the database of the selected preset.

**Petri Net**

Yellow = a choice the user has: a button

Green= something that happens because the user chose something

Orange= a place for user input, where the user makes a decision between ‘yellow buttons’

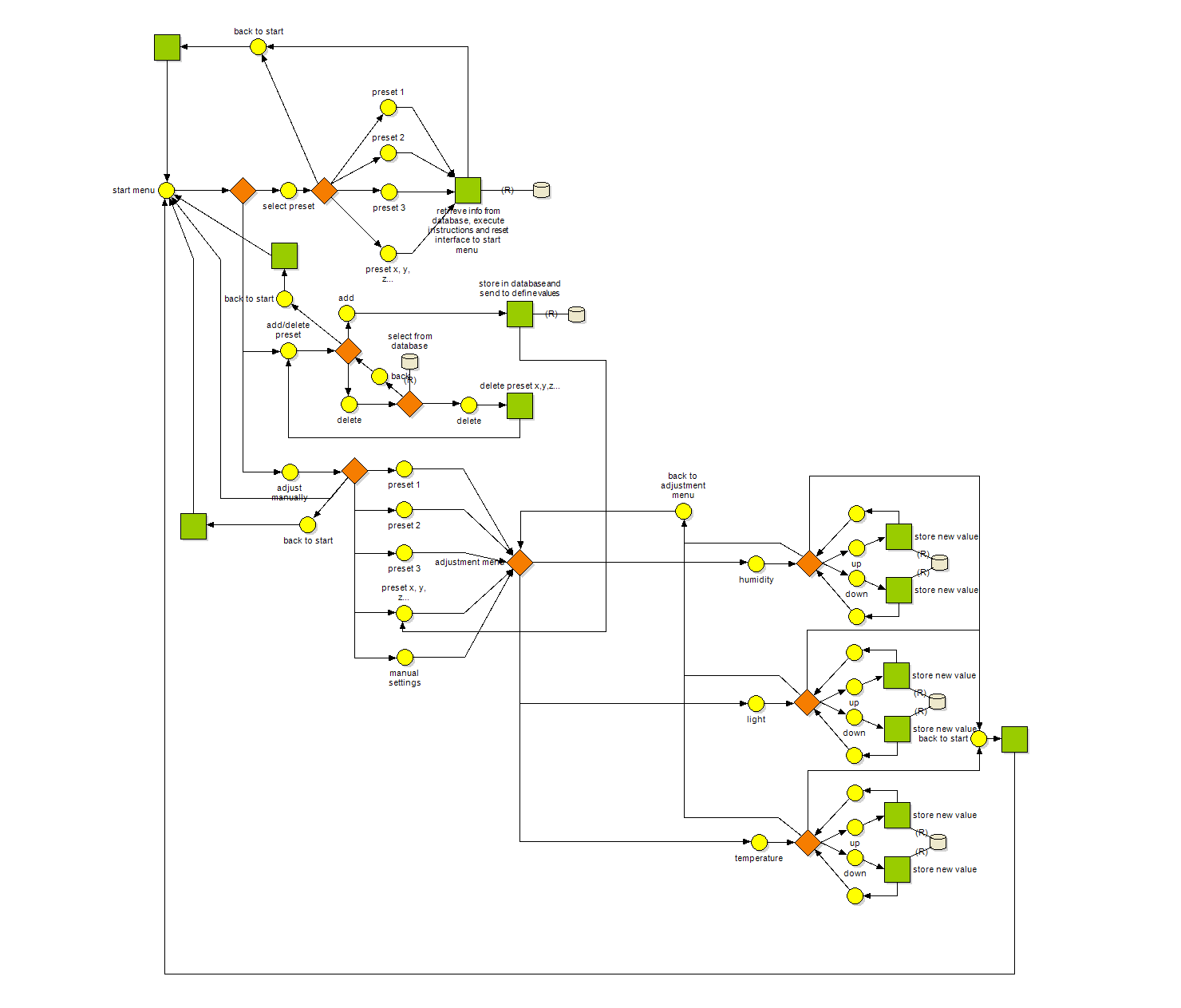


Figure : Smart Room Specification

Max and I went to work with the specification and came up with a prototype to present to our fellow students. It was a simulation of a room with temperature, humidity and light controls. Presets (as mentioned in the specification) were not included yet. Adjusting the settings (left side) made the current values (right side) adjust to match them. An animated ventilator, heater, humidifier/dehumidifier, lamp and window provided visual feedback to users as to what they were doing.



Figure : Prototype 1

This prototype was well received but lacked some obvious things like a clear interface and the preset options.

We tweaked the functions of the prototype a bit and created a clearer user interface in the second prototype. Presets were also added and could be edited and saved.



Figure : Prototype 2

## Step 3: Usertesting

For the last step Max and I split up again and set out to test our prototype on users.

I selected four users. They were between the age of 17 and 20, two of them were male, the other two female, all of them were healthy. They were asked to perform a series of 7 tasks and to think out loud while doing them.

Figure : User Testing Tasks

Tasks

1. Raise the temperature by two degrees.
2. Turn on the lamp.
3. Turn on the humidifier.
4. Select the “Cool” preset.
5. Turn of the ventilator.
6. Change the “neutral” preset to the values 20, 3, 60.
7. Change the “cool” to values 12, 3, 35 and then select the neutral preset again.

For each task I timed how long it took to discover any problems with the interface, the results are mapped below.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | User 1 | User 2 | User 3 | User 4 |
| Temperature | 3.5 | 2.9 | 4.1 | 3.0 |
| Lamp | 5.7 | 7.3 | 3.4 | 5.5 |
| Humidifier | 2.9 | 7.3 | 2.1 | 1.3 |
| “Cool” preset | 6.2 | 14.5 | 22.1 | 10.2 |
| Ventilator | 17.5 | 5.7 | 15.8 | 49.1 |
| Edit “neutral” preset | 21.0 | 19.5 | 10.5 | 9.8 |
| Edit “cool” preset but select “neutral” preset | 14.7 | 15.2 | 32.7 | 11.8 |

Figure : Results Table

The results coupled with the thinking out loud of the test subjects I took a few conclusions on how to the prototype could be improved even further:

* There should be more emphasis on presets; they should be immediately visible to anyone using the interface.
* An option to turn off individual devices could be added to the interface.
* The current weather values could be integrated into the interface, making for a single, unified interface.
* Users should be enabled to change and save presets without having to select them.
* The visual representations could be improved: ventilators merely circulate the air, not cool it. And the lamp could have different light intensities.

# Evaluation and Reflection

In the end learned a lot from Discrete Interaction Design. One of the most powerful things I learned was communicating the feel of a design without actually having to build and prototype the entire design first. In a relatively simple way I can create a representation of an interaction (Petri Net) and communicate it to others. I can also use the petri nets to discover problems and bugs in an interaction quickly and effectively.

Next to making Petri Nets I learned how to interpret them, during a few lectures I learned about the symbols and structures of Petri Nets. When presented with a Petri Net and a simple description and minimal verbal communication I was able to create a prototype to specification.

Another thing I am happy to have learned in DG308 is a user centred approach. The system being easy to use for users was always the goal of developing it. I learned to set up and interpret a user test in a meaningful way, to anticipate user reactions and avoid problems in a user test that are not related to the actual design.

Concluding, I think Discrete Interaction Design is a very important part of my user focussed process and valuable in my overall development as a designer.