Teaching Human-Computer Interaction from Real World Examples – Furnishing Creativity?

Lars Oestreicher
Computer Science Unit, Department of Information Science
Uppsala University, Uppsala, Sweden
larsoe@dis.uu.se
WWW home page: http://www.anst.uu.se/larsoest/

Abstract. This paper argues that it can give good results to use bad and good design examples when teaching Human-Computer Interaction provided that the examples are elaborated by the teacher in a manner that enables a changed (shaken) mind-set in the students. In the paper examples from lectures, exercises and larger assignments using concrete material as a base, are discussed from the perspective of general theories of education. The key issue in the teaching process in order to reach this level is the proper elaboration of the used examples and exercises, leaving the reflection on the problem to the students to a large extent. One conclusion is that it is possible to use elaboration to support the students' learning so that he or she is enabled to understand the general problem and create new solutions to given problems. Elaboration of concrete examples can in this way support even a creative level of learning.

1   Introduction

It is often said that creative thinking should be encouraged in students of Human-Computer Interaction. But how should this be done in practice? This paper takes a standpoint in this issue of teaching Human-Computer Interaction (HCI) to Computer Science Students. One way to stimulate creative thinking is in my experience to support the students' thought processes to break away from traditional ways of thinking about design and problem solving.

I have been an academic teacher since 1991, when I received a lecturing position in Computing Science, with an orientation towards Human-Computer Interaction.

The courses I have taught cover various topics in the area, ranging from Introduction to Human-Computer Interaction, Usability engineering, User-oriented software design and Task and Work analysis. In my teaching I use many examples of bad (and occasionally good) design cases to try to create a new mindset in the students mind. In this paper I will argue that the proper use of practical examples or illustrations (good or bad) can act as a catalyst for creative thinking in HCI design.

Over the years, I have collected many pictures, error dialogues, cartoons and other educational material, which to some extent have been published on the Internet as a free, public resource for teachers, e.g., in Human-Computer Interaction. The material is available on my personal home page [1] and the figures in this article are all displayed on the site. One example of the material that I use is shown in Fig. 1.

Fig. 1. *An example image that is used to illustrate the problem of complex technology in a different way than in an ordinary interface. (photo by the author)*

In this figure I have heaped all the remote controls that are used to (mis-)handle my home video system. These are the controls that I have to manage before I can turn on the TV-combo to see the popular Christmas show (hence the four surrounding candles) if I manage to do this in time. The example appears to be a much more telling way to talk about technological complexity than just showing a complex interface, e.g., to a power plant, or in a cockpit. The students remember this picture, because turning on a family show or the children's channel should not require an engineering degree, whereas most people do not imagine running a power plant or fly an aeroplane. The ridiculous situation has even provoked discussions on power distribution in family life, starting from the question about who has the

knowledge needed to turn on the TV. I will get back to how this type of pictures is used in later parts of this paper. (By the way, I actually happened to get the remote for the car GPS added to heap in Fig. 1 by mistake, which one is that?)

From teaching HCI at the Åland Polytechnic (Åland is a small Swedish speaking island, belonging to Finland) I have also had some experience from teaching in areas with (at least on the surface) minor cultural variations. This has provided an opportunity to see how minor cultural differences may influence the design process. Åland is a society with strong marine connections, inheriting a fishing and seafaring culture, which can also be seen in buildings and smaller design items. The use of cultural issues in teaching is mentioned further in section .

Most of the experiences, and the informal studies described in this paper are from students of Computer Science at Uppsala University, and at the Åland Polytechnic. None of the examples in this article are results from a controlled study, but they are examples that have been selected to reflect observations that have been collected over a long period of teaching in real classroom settings.

In section 2 a theoretical background is given to the approaches I use for teaching. In sections 3, 4 and 5, I will discuss how the theory affects lectures, exercises and assignments through a series of examples. I will also report on the experiences from the exercises when these have been used with groups of students with different backgrounds. Finally in section 6 I will draw some conclusions from the examples, and try to see how teaching can be supported through the ideas in this article.

2   Teaching Human-Computer Interaction

During my employment as a teacher, the focus has often been on making the students grasp the *problems* with software design. I have also tried to extend the area of interest during the courses to the problem of designing machines in a more general sense, but still with computers as the main area of interest. By widening the scope of the topic, the students are more or less forced to see human-machine interaction as an applied science. It also makes them aware of the fact that computers are intruding everywhere in society, and not only as visibly computer-shaped. However, I still use a fair amount of computer-based examples in my teaching, to attach to the students' final profession (for some samples see my home page [1]).

2.1   Theoretical Foundation

In my teaching I have gradually included ideas that are based on more firm didactic principles. Even if the theoretical approaches to teaching are difficult to apply in the practical teaching situation, the theoretic framework is a good starting point to understand the process of creating exercises and examples for the students to use. The basic ideas in my teaching are still founded on associative learning, such as described by Anderson [2]. In this simple model association is supported by elaboration of concepts into associative networks. Elaboration can be simple, such as just adding related ideas to the concept to be memorized, or it can be extensive by

creating more elaborate 'stories' around the concepts. However, only employing associativity is a too simple teaching strategy, since it mostly addresses simple memorizing, but does not promote such phenomena as understanding. Also more complicated things such as creativity are difficult to address in associative theories of learning.

A second model or framework for learning that has been used extensively is Bloom's taxonomy [3]. The taxonomy does not in itself provide direct models for teaching but can be used as a reference when teaching material is designed. In the revised model, there are several differences from the original [3]. The revised taxonomy is especially interesting in that it considers learning processes rather than the goals of the learning. The following learning processes are mentioned:

1. REMEMBERING: Can the student RECALL information?
2. UNDERSTANDING: Can the student EXPLAIN ideas or concepts?
3. APPLYING: Can the student USE the new knowledge in another familiar situation?
4. ANALYZING: Can the student DIFFERENTIATE between constituent parts?
5. EVALUATING: Can the student JUSTIFY a decision or course of action?
6. CREATING: Can the student GENERATE new products, ideas or ways of viewing things?

Thus the taxonomy in some respect defines knowledge on different levels as how the student uses the knowledge, from the simplest level, where the student just recalls the knowledge (e.g., as when learning the decimals of the irrational number π), to the most complex level where new knowledge can be synthesized from the old, which is closest to what we would like to achieve when we teach how to design. This taxonomy can also be compared to the idea about competences, separating different kinds of knowledge, dependent on what a person can do with it. This is also referred to as the pyramid of competence model [4]:

- Declarative knowledge – What?
- Procedural knowledge – How?
- Judgement knowledge – When?
- Wisdom knowledge – Why?

In order to be skilled designer the 'why'-aspect needs to be a more or less constant awareness. Much of textbook information is giving declarative and procedural knowledge, whereas judgment knowledge and wisdom knowledge is often left as something that grows as a skill. One question is whether the upper levels in Bloom's taxonomy can be achieved through the teaching process, and thus also provide the knowledge displayed in the top of the pyramid of competence.

A fourth model that can be applicable in this context is the scaffolded instruction [4] that stems back to the work by Vygotsky [6]. In this model the tutor prepares the learning process by a scaffolding instruction incorporating the following features [4]:

1. *Intentionality*: The task has a clear overall purpose driving any separate activity that may contribute to the whole.
2. *Appropriateness*: Instructional tasks pose problems that can be solved with help but which students could not successfully complete on their own.

3. *Structure*: Modelling and questioning activities are structured around a model of appropriate approaches to the task and lead to a natural sequence of thought and language.
4. *Collaboration*: The teacher's response to student work recasts and expands upon the students' efforts without rejecting what they have accomplished on their own. The teacher's primary role is collaborative rather than evaluative.
5. *Internalization*: External scaffolding for the activity is gradually withdrawn as the patterns are internalized by the students.

There are many other perspectives on the scaffolding instruction process, but from this perspective it can be used as laying a foundation for learning on the various levels. This means that a teaching process can use the basic features of this model in different ways, depending on the knowledge level that is aimed at. By laying the proper foundation, the students have an easier task to incorporate the new knowledge.

In most cases it is better the closer to the evaluation/creation and judgement/wisdom levels of knowledge we can take students through our teaching. The question is whether it is possible to teach Human-Computer Interaction in such a way that these levels are stimulated?

2.2   Mind-Shaking Teaching

A large part of the education has to be based on a mixture of practical and theoretical materials for the student as was discussed at the INTERACT'99 workshop on Teaching HCI (cf. [7]). During the workshop the need for mind-shaking experiences for the student was explicitly stated [8–9]. The conclusion was the introduction of an approximate general procedure for teaching Human-Computer Interaction based on six 'golden rules' for shaking the students mind when teaching Human-Computer Interaction [8] that were listed as one result from the workshop:

1. Provide thought-provoking literature that illustrates the problem.
2. Let the students observe real users, using real artefacts, anchoring the observations in the literature.
3. Let the students analyze the results of the observations, and stimulate the understanding by making them write reports!
4. Mix the results from the analysis with theory, e.g., cognitive psychology, perception psychology, computer science.
5. Apply the experiences in a redesign of the artefact.
6. Iterate the observation phase with the new artefact.

The foundation of this approach is to encourage an experiential knowledge acquisition in the student. The basic assumption is that it is only by actual practical experience that the student can understand the nature of the knowledge and also understand how the knowledge can be applied. This is in my experience especially true when we teach the topic to students who are not specifically Human-Computer Interaction specialists, but software engineers, or web-programmers etc. These students may be less motivated in learning good design, since they are more intent

on the technical problems in software design. In this context good practical examples are crucial to awakening the necessary understanding of the problem.

In all these models there is also a need for a separation between implicit and explicit learning, where the former is learning that 'just occurs', without the learner consciously trying to learn, and the second is based on an intention to learn something specific. In some cases implicit learning gives a more immediate acceptance of the knowledge, whereas explicit learning requires more learning efforts. Used in the proper way the scaffolding instruction process can be applied to the teaching in a way that seems to make implicit learning much more immediate and effective.

In the following sections I will describe a series of activities used in lectures, exercises and assignments, and which I suggest do address not only the lower levels of the models referred in this section, but also have the potential of addressing the upper levels, thus giving the student a more applied knowledge that can boost creativity in the HCI area.

3   Mind Shakers in Lectures

Many of the courses I have taught have been short courses that can be classified as crash introductory courses, where a course goal has been 'to make the students aware of the problems with human-machine interaction'. One problem with the short courses is that it has been difficult to include the whole design process from scratch in the teaching. One way to work around this has been to work from concrete design examples. However, in order to explain essential concepts and to urge the students to take the ideas further it is not sufficient to provide illustrative examples of good or bad designs. It is also necessary to provide examples that make the students reflect, and even make the students reflect on the further consequences emanating from the more general problems illustrated through the examples. To do this, it is not sufficient just to present the examples, but it is also necessary to complement the examples by prodded real-time reflection and the addition of follow-up material, such as exercises or questions to ponder.

Teaching good design is in itself a big challenge. Teaching good design without delimiting the creativity in the students mind is an even greater challenge. One of the best appreciations I have had as a teacher in HCI was a former student who came back, five or six years after his graduation, telling me that 'You have destroyed my life! I can't open a door or turn on a water faucet, without thinking about you!'. By the concrete examples used in the lectures, the student had been forced to start reflecting on the things around him. His statement left me with a feeling of being successful in the first part of my mission as a teacher, in shaking the students' minds. This reflects my ideas on the effects of successful teaching of Human-Computer Interaction — the making aware and inspiring of a constructive curiosity.

The problem is not to teach the students to make the best design, but to teach the students in daring to use alternative designs, while still being aware of the advantages and disadvantages of their own designs. During both lectures and practical exercises I try to encourage the students to try on new designs that go

outside of the normal thoughts and furthermore to have them challenged in critical examinations. In this process I also encourage the students to observe cultural phenomena, such as shapes and colour schemes that they can observe in their surroundings. By picking up and also stressing cultural design features in the teaching process, it is easier to make the students reflect on the designs that they produce in the end. Often the cultural specialties in the environments are not considered by the students until they are pointed out specifically, as for example, during the lectures.

3.1   Hot or Cold Showers

The first example of such material for use in lecturing is displayed in Fig. 2 below. This particular example has been used to discuss the concept of accessibility, but it contains several other possibilities (see the full example with comments on the web page [10]).
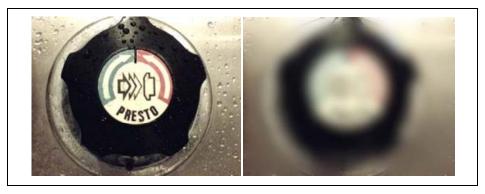

Fig. 2. *Two different pictures of the same faucet displaying a typical problem of accessibility (photo and post processing by the author)*

During a typical lecture, at first the left picture is shown, displaying a faucet that is used to change the temperature of water in a shower at a public bathhouse. The students are asked about what they can see in the picture. Then the picture is changed to the right picture in Fig. 2, showing the situation when someone with bad eyesight removes their glasses. The question is, which way to turn the tap in order to get colder water when you get scalded? The students quickly realize that the normal condition of usage does indeed mean an accessibility problem for some parts of the bathhouse population. Remarkably enough, even the students ordinarily wearing glasses often find themselves a little surprised by this example.

Now, this example by itself does not contribute much unless we use it by adding some complementary exercises, where the students are asked to hand in their own examples of something that might be difficult to use for someone with special needs. In the assignment they also have to explain why the problem occurs, and a possible way around it. The exercise is constrained by using specified disabilities.

The extra elaboration around this example serves several purposes:

- It gives the student a better retention of the example itself.
- It forces the student to adapt the problem to other, similar problems, thus making the problem a more general observation.
- By suggesting a solution it is necessary for the student to engage in a more creative process than the simple viewing of the example.

In the exercises that follow the example it often turns out that the students find a motivating challenge in finding alternative solutions, and they seem to get a better understanding and appreciation of the problems with accessibility, which in many cases does not make any impression on the students during the short courses (between four or five weeks) that we have at our University.

3.2   Less Extra Sugar?

The second example shown in Fig. 3 is also a picture that is used during a lecture to illuminate a problem, namely the problem of achieving clarity in interface messages. The lecture example is in this case immediately followed by a larger design exercise (performed in the same way as he exercises described in section 0), where the students are encouraged to focus their design ideas around clarity of the concepts. Unfortunately the example is from a Swedish coffee vending machine, but the idea should be easy to get from the labels anyway (the figure caption gives the proper translation).
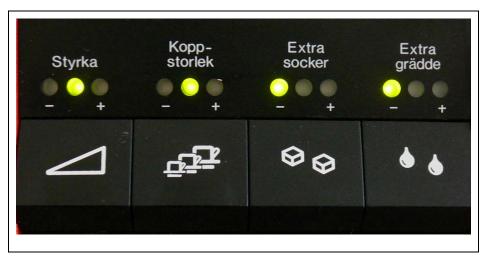

Fig. 3. *The adjustment buttons on a common coffee vending machine in Sweden. The labels on top read 'Strength', 'Cup Size', 'Extra Sugar' and 'Extra Cream' from the left to the right. The question is what the minus signs signify in combination with the expression 'Extra…' (photo by the author)*

The example in Fig. 3 is from a very capable coffee vending machine, which provides the customers with an assortment of hot drinks including chocolate and mixtures of coffee and chocolate. The machine also offers the user to adjust some

parameters in the ordered drink. You can, for example, have a stronger or weaker drink than normal (first button from the left) and you can have a larger or a smaller cup than normal (second button from the left). It also allows you to adjust the amount of cream and sugar added to the coffee (this will not work for chocolate drinks, although this is not visible in the interface). However, here is also where the problem starts. The two remaining buttons are labelled 'Extra Sugar' and 'Extra Cream'. But the labels beneath marked '+' and '–', indicate that you can also have 'Less Extra Sugar' and 'Less Extra Cream' added to your cup. If we would compare the function of these two buttons to those for cup size and for strength, 'Less' should denote, less than normal, but now it is modified by the word 'Extra'. So we end up with having 'Less Extra Sugar than Normal' in our cup. The problem is therefore to explain how much sugar is actually added to a normal sized cup.

Often the students at first don't even discover that this is a problem, since they are already used to the machine. When they start seeing the problem they often remark, 'Well, it is no really big thing'. But then of course, if it is not a big thing, then why is the button there at all? Once here, the students sometimes find themselves engaged in a discussion on what the expression 'Less Extra Sugar' means in practice, which normally also means it is time for a real-time coffee break in the lecture (possibly for some exploratory experimentation on the coffee machine).

As a follow-up to the lecture, the students are presented with an exercise where they are challenged to first pinpoint the more general problem with this interface, i.e., the lack of visibility of status and function availability, and then to suggest an alternative design that avoids the problems that they have found.

3.3   Changing Mindsets – Stimulating Creativity

Creativity is in this context not something that can be taught, but rather an ability in the student that has to be awakened and stimulated. Understanding human-computer interaction is in my opinion very much an issue about managing to put oneself in the situation of the computer users [8]. This is a difficult thing to teach, and sometimes the theoretical explanations do not support this change of perspective in the student. One typical example is how to explain *affordance*, since the notion is difficult to explain in simple terms. When provided with definitions, students are often able to repeat and even in some cases explain the idea behind it. However, when, instead of explanations, they are provided with a few, to them well-known, cases of bad affordance, and the following discussion of the problem, there is a remarkable change in understanding in the students. When asked to provide a definition, however, that is much more difficult from the examples only. This suggests that the examples have to be accompanied by the explanation during the afterwork after the examples. The examples then serve as a platform on which to build the more general framework through elaboration and discussion.

By providing good design examples and by discussing bad design examples, it is quite possible to induce a change in the mindset of the forthcoming software engineers. In this context I believe more in the bad examples, as long as they are not just shown, but also elaborated in an appropriate way. The examples shown in this section are typical for the type of examples that I have collected over the years. A

general property of these examples is that the problem with the design is easy to understand in each case, but the solution more difficult to find. The students also recall these examples and use them as starting points when discussing the more theoretical issues around the design exercises, and during the exams. Therefore, the examples are published on the website with ample explanations and elaboration suggestions for the teachers.

One of my first sources of inspiration for this kind is the first book by Donald Norman: The Psychology of Everyday Things [11]. In this book, everyday users' experiences of the world are brought up from the subconscious to the conscious level of thinking. However, in some way the process stops at the point of observation the phenomenon. In my teaching I often try to go further with the observations, and try to make the students reflect further on the problem and its consequences. Challenging them to redesign things that don't work with clear constraints on the new design often causes them to start thinking in a more lateral way.

One (informal) observation I have made over the years is that software engineers tend to be caught in traditional ways of thinking around software. In their opinion, graphic user interfaces cannot be too different from what they look like today. In this perspective I see it as an important task to encourage non-traditional thinking about the interface. In these contexts, therefore, examples like the ones shown in this section are the more important, since they challenge the traditional ways of thinking on interfaces. In the next section it will also be shown how restraining can force the students into lateral thinking, when the traditional solutions are constrained away.

4   Creativity Stimulating Exercises

Creative exercises do not exist in themselves; rather it is the context and the setup that does make an exercise inspire creativity. I have over the years used a number of exercises that were designed to talk to the creative sides in the students. There are several aspects that are important for triggering free thinking in the student:

- *Inspiring curiosity*: Curiosity is one basic incitement for creativity. By inspiring the students to use curiosity, it is easier to open their minds to understanding problems and solutions. The 'what if'-type of questions are very important catalysts for new designs.
- *Group dynamics*: I value the use of group dynamics both for design and for evaluation. Using parallel design (or snow-ball group dynamics) the students are forced to rethink their design suggestions several times over time. By constructive evaluation they are also forced to not only criticize the design but also suggest improvements, i.e. rethink the problem situation.
- *Grounded exercises*: The exercises have to be based on real problems. They should be relevant to the student and based on their reality if possible. By considering well-known areas, the 'why haven't I thought about this before' part of the brain is often activated, possibly inspiring further curiosity.
- *A constrained design space*: By added constraints in the design space (not the actual design or the creative process) the students are forced to move away from the traditional solutions. This removes the uncertainty from the students about

what is and is not allowed in the exercises. Furthermore, it allows the teacher to remove conventional solutions from the design space.

I will start by presenting the exercise process and then the exercises in more detail. I will get back to the importance of the process also in the evaluation of my design examples and the assignment.

4.1   Process

In this section two design exercises are presented. They are each performed in four separate stages:
1. Individual design.
2. Small group redesign (2 – 3 students per group).
3. Large group redesign (15 – 20 students per group).
4. Final examination/evaluation of exercise.

Stage two and three also involve a critical assessment of the previous design, if possible with written comments and motivations. If time permits, the critique is also discussed in the groups, in order to train the critical thinking. After the large group redesign, I have also tried a method of 'The critical eye', where all students have been asked to take down three essential advantages and three essential disadvantages with the 'final' design. Each exercise normally takes about four hours to complete, with a longer mid-exercise break in the middle.

During the first two stages of the design exercises the students are left more or less to themselves, with the teacher/session leader moderating the discussion in order to avoid inactivity. During the presentation of the final stage the teacher enters a different role as a constructive critic. In this role the students are challenged to defend their choices and also defend their non-choices, i.e. the solutions that they have discarded. Informally they describe a partial design rationale (cf. [12]). All the design work is made only on paper, preferably large display sheets. Computer tools are not allowed for the design. This makes the students focus on the design, rather than the technology.

One point of caution is worth mentioning here. In the group design stages (2 and 3) there is a large risk that there is someone (or a couple of persons) who drives the design process, ruling out other suggestions. This is especially difficult in interdisciplinary groups, if one of the participants has a deeper knowledge in the application area. Avoiding this could be a matter of the teacher monitoring the group work. In general, it is better the more active all the participants are in the larger sessions.

4.2   Design Exercise Examples

The two design exercises I present in this paper were also included in the position paper for the CONVIVIO 2006 workshop (see [13]) but the descriptions have been extended in order to make the examples more clear. The two examples are to some extent similar to each other. Normally I let the students alternate between computer-oriented design examples and design examples that are more anchored in the 'real'

world. The sample results presented in section 4.3 come from courses held with Swedish speaking students at the Åland Polytechnic.

4.2.1 Example Exercise 1

The first example exercise is to redesign a printing dialog that allows a user to print a document on a printer and really (!) get the *desired* printout. The functionality is specified in advance, and corresponds to a standard ink jet printer dialog, including the 'Page Setup'-function. The dialog should allow the user to achieve the result in a minimum number of steps with respect to the intended task. There are a few general restrictions placed on the final design:

- It should be task-driven and not consider technical difficulties (such as that round windows may be more difficult and inefficient to draw).
- It should use a minimum number of buttons and traditional controls such as dropdown menus, tabs, etc.
- It should adhere to minimalist design
- It should provide immediate feedback on actions

Prior to the exercise, the students are also told that one important aim is to achieve a design that has to be as non-standard as possible, while still offering a good and usable interface to the printer.

4.2.2 Example Exercise 2

The second example exercise is to design an interactive display that allows a car buyer to select from a vast number of cars on a site for used and new cars. The selection dialogue is to be primarily directed towards a non-technical user, while still being interesting to use for more knowledgeable car buyers. Prospective buyers have to be provided with sufficient support in order to make the appropriate choices for a new or used car.

In this exercise more stress is also placed on that they find good solutions for the designs than on that they cover all possible options for selecting a car. The functionality is therefore not completely specified, but the students may instead add their own functionality.

In the task setting there are also a few general restrictions on the design:

- It should not be primarily text based,
- It should be a minimalist design,
- It should allow fuzzy requirements, or multi-valued choices in an easy way

Like in exercise 1, the students are told that non-traditional solutions are preferred to standard interface alternatives. This is stressed during all the three design stages of the exercise. It also turns out to be a major difficulty throughout for the more computer science oriented students.

4.3   Exercise Results

The following presentation of the experiences from the exercises is divided into a set of general observations on the procedure and type of exercises, as well as the actual outcome on the teacher side from the respective design exercises. Since the final designs from the student groups were presented on drawing boards or black boards,

there are unfortunately no figures available showing the resulting design suggestions from the students.

4.3.1 Exercise 1 – Results
The solutions to the exercise with the printing dialogue that were presented evolved gradually over the design stages, and in the evaluation processes, the restrictions were among the first things that were discussed and often the properties that were not conforming to the restrictions were changed or removed in the designs after the first evaluation – redesign cycle.

In this exercise the gradual extension of the design group from individual design to smaller groups to a single large group proved constructive, when the design solutions were presented. In general the students found that it was interesting to try to provide the given functionality in the new design, without complicating it by using standard widgets, such as tabs. Most of the specified functionality was also incorporated in the design solution that was presented after stage 3 or 4, i.e. as soon as they 'got the gist of the design problem'.

*Example of a final design***:** The final solution used a preview type of display with direct manipulation switches that directed choices, such as orientation, size, number of copies, etc. A number of visual switches displayed on-off type of attributes, such as binding edges and similar properties in the printed material.

4.3.2 Exercise 2 – Results
The search form for used cars appears to have been more difficult from the start, and it was clear that the students were less experienced car buyers. There were initial problems specifying the relevant criteria for selecting a used car, and due to time constraints there was no time to arrange for the students to interview real car buyers. Also there were initial problems with the constraints on the design, such as how to avoid using text for selecting various criteria. In one or two cases it was even necessary to intervene and discuss the problem with the group, before they could start. Interestingly enough this also seemed to make them more restricted in the thought to traditional (i.e., the results were more similar to the existing web search sites for used cars).

Also the change between the stages was subtler, changing more in details than conceptual design, compared to in exercise 1. The final solution was also using more features of the individual suggestions, and did not evolve quite so much over the stages as the printing dialogue.

*Example of a final design***:** The design presented was based on a picture of a car body that changed visually as the user was selecting important features. The visual presentation of the body changed when the user selected properties such as seating, loading capacity and similar. In case of multiple selections, the picture was duplicated. Additional features such as engine type, type of gear box, etc. was indicated graphically on the pictures.

4.4   General Observations on the Exercises

The exercises were generally well liked by the students. For the more constrained examples it was considerably easier to get students started on the exercise. The

process used for the exercises seems to gradually allow students to proceed from an applied type of knowledge, to knowledge that uses justification of the design decisions. There are also indications (especially in the first exercise) that the students shift from a traditional design perspective to a more inventive view, when they consider the constraints of the exercise. Thus, they are approaching the upper learning or competence levels in the models presented earlier.

The first exercise seems to induce a more direct connection to the students' imagination, probably (according to some comments from the students during the design discussions) since the students have been using the print dialogue extensively and also seen the problems with the dialogue. The second exercise was more difficult for the students to get started with. Common comments were such as 'How can you display information without using text?' and 'It is difficult not to make something that looks like those sites that already exist!'. Some mental 'prodding' seems necessary in order to get the students into non-conform thinking.

The outcome of these exercises has also been interesting from a methodological standpoint. Many students tended to fall back to designing a conventional type of display. Utilizing direct manipulation was often not considered an option, and a more or less standard 'Apply' or 'OK' button (although often 'disguised' in order to comply with the given restrictions on the interface) was not an uncommon feature in the dialogue.

In a 'Pilot' exercise I gave the students free hands on the first exercise, which actually rendered it much more difficult for the students to begin working on the exercise. This does not mean that it has to be specified in details, but rather that the overall constraints have to be well specified, giving the students a solid framework for the exercise.

5   Example Assignment

Even the major assignments on a Human-Computer Interaction course need to be aimed at the upper levels of Bloom's taxonomy and the pyramid of competence (cf. section 2.1). Most assignments on the courses I teach have been designed to promote a synthetic or at least an analytic aspect of the design problem.

The assignment described here is a slightly older assignment, but one that has been very successful over the years. Even though technology has speeded ahead, the design challenge is still the same, although for an updated version I would probably rather use displays of mobile phones, PDAs or why not, remote controls for home video systems as example applications. However, the essential design problem remains the same, i.e. achieving good functionality in a limited space.

5.1   Process

The larger assignments on my courses have often included a separate phase of task analysis including interviews with potential users of the artefact, prior to the following design task. This means that the assignments have been run over longer

periods of time, to give the students ample time for reflection over the problems involved.

Assignments of the type described in this chapter normally run over a period of two to three weeks, with smaller groups (< 20 students) of computer science students. The assignments are initially run in the same manner as the exercises in the previous section (see section 4.1). They are in most cases started as snowball group exercises, i.e., where the work is initially done in small groups (or individually) and the groups are gradually merged into larger groups by a sequence of constructive group critique/redesign cycles, composing larger and larger design groups. In this way the students are forced to rationalize their designs from the beginning. In the assignments, the students are required to hand in a larger part of a design rationale, which is now formal, rather than informal as in the exercises (cf. section 4.1). In contrast to the exercises the assignments are run in smaller end groups, consisting of at most six students (but often less). This means that the overall positive effect of snowballing becomes smaller.

5.2   Designing a Telephone Switch

The assignment that the students have been working with is to design an innovative telephone interface (to a standard telephone) using a colour touch-screen display. The display was intended to allow a user to handle all the necessary switchboard functions on the normal home telephone, such as wakeup calls, moving calls from the home phone to another number, etc. These are the functions in a standard telephone that are (or were) handled via combinations of two-digit numbers and '#' and '*' buttons. These combinations are of course difficult, if not impossible, for a user to remember unless he or she is an expert on handling the telephone switchboard.

The following restrictions were given for the final design:
- The maximum size of the display was set to eight by eight centimetres.
- The display should allow for also users who are blue-collared workers, i.e., the controls should be easy to handle also with big hands.
- The display should be readable by people with normal or slightly reduced eyesight.
- All the functionality (approximately 10 different functions) that was described in the official manual had to be represented in the final system.

There is also an informal requirement stating that the resulting product should not primarily look like a computer display. This requirement was initially slightly difficult to carry through, primarily due to the lack of good prototyping tools. Today, this prerequisite would be much easier to fulfil with the drawing tools that are available. On the other hand, the best prototypes were often the ones that were 'hand made' and where no toolkit was available, since then the students were able to focus on the design, rather than the implementation (cf. the use of paper designs in section 4.1).

The first designs are sketched individually and then disseminated in a first common seminar. Then the students are regrouped into pairs, and a new design

process initiated. This continues throughout the whole assignment, where in the end the whole group sets out to create a final design for presentation.

5.3 Results

The assignment described in this section has been used over a period of totally 6 years, and interestingly enough, there has been a large variation in the suggested design solutions over time. In the beginning the assignment was less specified in the instructions, but it turned out that when the restrictions on the assignment were more specified, the students made much better initial designs.

The students often commented on that they were encouraged to change their ideas when they saw the other group members' solutions, and many also agreed on that the group discussions added to their understanding of both the problem and the requirements on the solutions. Over the 6 years there were only five or six 'similar' design suggestions, from the smaller groups (four students) before the final common design phase.

On the negative side it is possible to note that there is a tendency to use good examples from lectures and literature in a quite literal manner, in that the students ended up using many of the design suggestions given during lectures in their designs. This implicates that in order to increase the amount of creative thinking, the teacher has to be carefully considering *how* solutions to given problems are discussed, e.g., in lectures.

6 Conclusive Summary

Over the doors of the main lecture hall in the old university building at Uppsala University there is an old motto from around 1850, which states: "To think freely is good, to think correctly is better".[2] The original intention of the motto is that it is preferable to follow the logic of thought rather than free elaboration. However, it has often been misinterpreted, taking 'right' to be standing for proper thinking or being politically correct. Now, I still think that in some cases there is a firm belief from the students in a single 'right' thinking, which does prohibit 'free' thinking. A frequent problem is that the students expect to find 'the single right solution'. They more or less expect that there is a 'solutions page' also for such free and open ended exercises, as the ones described in this article. In fact, they sometimes initially get disappointed to find that there are many possible solutions to a problem.

From the experiences of using lecture illustrations, exercises and assignments such as the ones in this paper, it can be seen that it is possible to use good and bad design examples to start the students off towards creative thinking in human-computer interaction design. But it also turns out that it is almost mandatory to give the students a constraining framework to work within. So how do we handle the balance between using free exploration in the exercises (to encourage creativity) and constraining the exercises and assignments (to support the knowledge forming in

---

[2] The quotation is a great source of smiles to most visitors of the university.

'good' directions)? The issue is of course whether we should encourage 'free' thinking and look away from 'right' or 'wrong', or look more decisively at right or wrong designs with the risk of reducing the creativity in the students?

From my experiences with the assignments, there are good reasons to expect that exercises will produce better and for all, more creative results if there are well-defined restrictions on the design space, and on the methods that are allowed to achieve the goal (this was discussed at the second CONVIVIO 2006 workshop [13]). This will give the students a stricter framework to work within, thus reducing their problems to understand the task. Thus by using examples and exercises that stimulate the upper levels of the educational models, while at the same time restraining the framework within which the students can work, we may even increase the students' abilities to use lateral thinking in the design exercises and assignments.

Furthermore, it is also suggested in this paper that neither lecture examples, exercises or assignments (regardless of their quality) will on their own bring the students to the wisdom level of competence (according to the pyramid of competence [4]), or the creative level of learning (according to Bloom's extended taxonomy [3]). The teacher must prepare this development in the students by a proper combination of examples, exercises and assignments, in combination with a good (and preferably guided) discussion on the problem area. The use of real world examples requires a large amount of preparation on the part of the teacher. Also, it is essential that the students are given ample time to consider and reflect on the teaching elements described in this article, something which might be difficult in short courses.

A good example is still only an example, and it does not become knowledge without additional effort, from both the student and the teacher.

Note

Those interested in the material discussed in this article are welcome to check out my home page [1] with some of the material I use in my lectures, with supportive annotations and comments on how to use it. Also, the remote control for the GPS in Fig. 1 is the small black one, closest to the leftmost candlestick.

Acknowledgements

References

1. L. Oestreicher, The author's home page with educational resources http://www.anst.uu.se/larsoest/pmwiki.php?n=Education.EducationalResources (cited June 26, 2007).
2. J.R. Anderson, *Cognitive Psychology and its Implications.* (Freeman, New York, 1990).
3. M. Forehand, Bloom's taxonomy: Original and revised, in: *Emerging Perspectives on learning, teaching and technology*, edited by M. Orey (cited on January 2, 2007): http://www.coe.uga.edu/epltt/bloom.htm (2005).
4. P. Kotzé, K. Renaud, and J. Van Biljon, *Don't do this - Pitfalls in using anti-patterns in teaching human-computer interaction principles*, Computers & Education, DOI: doi:10.1016/j.compedu.2006.10.003, (2006).
5. L. Lipscomb, J. Swanson and A. West, Scaffolding. In: M. Orey (Ed.), *Emerging perspectives on learning, teaching, and technology.* (cited on January 2, 2007): http://www.coe.uga.edu/epltt/scaffolding.htm (2004).
6. L.S. Vygotsky, *Mind in Society.* (Harvard University Press, Cambridge, MA, 1978).
7. M. Cox, L. Oestreicher, M. Quinn, M. Rauterberg and M. Stolze, HCI in Education: Theory or Practice. *Proceedings of INTERACT'99,* (Kluwer, Edinburgh, UK, 1999).
8. L. Oestreicher (ed.), The Six Golden Rules to shake the Students Mind — Report from a workshop on HCI education in Edinburgh. (cited on June 26, 2007): http://www.anst.uu.se/larsoest/uploads/Research/SixRules.pdf (IFIP Working Group 13.1:8, 1999).
9. L. Oestreicher and P. Kotzé, Teaching Human-Computer Interaction: Qualitative Support for an Alternative Approach, In: *Usability: Gaining a Competitive Edge, Proceedings of IFIP World Computer Congress,* Montreal, Canada, p. 267 - 281. (2002).
10. L. Oestreicher, The 'Hot or Cold Shower?' example with colour pictures (cited on June 26, 2007) http://www.anst.uu.se/larsoest/pmwiki.php?n=Education.-HotOrColdShowers (2007).
11. D.A. Norman, *The Psychology of Everyday Things* (Basic Books, New York, 1988).
12. S. Buckingham Shum, Design Argumentation as Design Rationale. In: *The Encyclopedia of Computer Science and Technology* (Marcel Dekker Inc: NY), Vol. 35, Supp. 20, 95-128. (1996).
13. K. Baumann, P. Kotzé, L. Oestreicher, L. Bannon, A. Varey, D. Van Greunen, G. Van der Veer, H. Petrie, I. Jounila, I. Mavrommati, N. Garay-Vitoria, O. Ozcan, P. Purgathofer, and P.A. Silva, EISH - Exercises in Studying HCI, in: *Creativity3: Experiencing to Educate and Design - Proceedings of HCI Educators 2007*, edited by P. Alexandra Silva, A. Dix, and J. Jorge, (DesignEd, Aveiro, 2007), pp. 134 – 137