Interface Design and Development Tool Limitations

Stella Mills
CGCHE

Introduction

Human-Computer Interaction (HCI) is a well established subject within computing and deals with the design elements of the system which effect the human user. The obvious candidate here is the design of the interface ('screen') itself but HCI is also involved with other more indirect users' aspects such as relevant functionality for the user's tasks. Within HCI, there are accepted guidelines for design which attempt to optimise the users' position and these should be followed when designing applications. However, in some cases, the guidelines have to be compromised because of the limitations of the software. This paper discusses some of these problems caused by the limitations of two different Multimedia Development Tools (MDTs). The first is a Virtual Reality (VR) development shell and was used to produce a small application for a joint project with a university in Portugal. The second MDT is a well-known commercial multimedia development tool and was used to produce a commercial application available in the public domain.

Multimedia Development Tool 1 (MDT1)

Software Description

The area of application for the software development tool was to aid teaching about the problem of assigning resources subject to constraints - a problem very common in operational research. The particular example was taken from a case-study (Heimann and Lusk, 1979) and only the first part of the study was implemented. The scenario consisted of planning for the expansion of nursing-home provision subject to building constraints in any one of four geographical areas. It was decided to implement the case-study using a Virtual Reality (VR) development tool even though a more interactive case-study, requiring more movement in a three-dimensional world, may have been better as this would have exhibited more obviously the characteristics peculiar to a virtual environment. However, the drawback to this is the time taken to build the VR world in the software which could be two years or more for a fairly simple world (Mason, 1996) and the advantage of the chosen scenario was that the software could be developed in a few months and so within the timescale of the project.

Limitations

The software design followed the usual method of storyboarding (Bunzel and Morris, 1994) and prototyping. The initial paper prototypes (Figure 1) of the screen had to be modified to be accommodated by the software thus illustrating the development still

needed in VR development shells in order to allow the easiest learning environment for the user. The biggest problem here was ensuring that the user could see the method of calculation used to arrive at the result as this was the learning kernel, thus avoiding covert action by the system. This can be expressed as the system must clearly and directly give feedback about users' actions (MoD, 1996). Eventually it was decided to give the mathematical explanation needed in a dialogue box at the end of each short section of the implementation (Figure 2). However, reading and understanding these could easily be ignored by weaker students, particularly those who completed the sections correctly. Consistency was maintained between screens by using similar designs for each screen and each overall section.

The software was found only to support languages without accents and as the software was destined for use in Portugal this proved to be disconcerting for the users since the only distinguishing mark between two frequently used Portuguese words is an accent. Consequently, the users had to guess from the context which word was required, thus increasing the cognitive load on the user (Nielsen and Mack, 1994), (MoD, 1996).

Another problem was that the software was at times unpredictable and a considerable amount of the development had to be completed at least twice because of lost work. Full advantage was taken of the clinics available to subscribers and provided by the manufacturer of MDT1. This included several visits to the manufacturer and numerous telephone calls. However, it is felt that this was well worth the trouble as it ensured that the software was developed to the highest quality possible.

This example has shown very clearly how the user has had to accommodate the short-comings of the software often by increasing cognitive load. In specific aspects, the user could not have the information needed to hand immediately but had to wait until the whole sequence of events had been completed. Inevitably, this led to the user's understanding being diminished.

Multimedia Development Tool 2 (MDT2)

Software Description

This tool was used to develop a multimedia CD of walks in the UK which has been both produced and sold commercially through retailers and one walking magazine. In all, there are 100 walks spanning seven regions of the UK with each region's section being sponsored by a commercial company. This is emphasised by the use of speech at the entry point of each section and the user is given the chance to replay this as many times as he/she wishes! Needless-to-say, this becomes rather tedious after a few entries into each section. The walks in each region are graded into four categories plus a 'long walk' section. Each walk can be printed out using *Acrobat Reader* which comes on the CD with the software. This is very good and easy; a map is included along with full directions and this prints easily too. Each region also gives information on visitors' attractions, camping and caravanning sites and tourist information centres as well as important items such as weather information and mountain rescue details.

Limitations

First of all, one questions the use of the 'replay audio' button as all this does is replay the spoken details about the sponsorship of the production for each region. While the sponsors requested this, the lack of a facility for turning off the speech about the sponsorship contradicts the HCI principle of flexibility to the extent that a user can become irritated by the repetition as they cannot tailor frequent actions to their own needs (Nielsen and Mack, 1994).

However, perhaps the most important disregard for HCI guidelines is that of the cursor not changing shape while the system is 'busy'; the time to load some sections is understandably long and the user needs to know that the system has not 'hung'. The system state should always be known to the user (Nielsen and Mack, 1994). Response times are important too and they need to be sufficiently short so that the user does not lose interest or become frustrated (MoD, 1996). In this software, an audible tap is used to confirm a mouse-click but perhaps here is an opportunity for some music or some speech (other than sponsorship details!) introducing the section being loaded.

At all times, the user should be able to achieve the required task as quickly and easily as possible (Nielsen and Mack, 1994) but in this system the user is given no instructions at any time; buttons clearly labelled indicate navigation but some user knowledge is assumed particularly relating to mouse-movements. For example, in the Quick Time Virtual Reality (QTVR), it is assumed that the user will know to keep the right mouse button depressed while moving the mouse in order to move around the video clip. Facilities for zooming are available in the QTVR but the required action by the user is not very obvious.

The limitations of the software above are, in part, due to the limitations of the development tool. Others are due to meeting requests of sponsors but in both cases the user's enjoyment of the software may suffer.

Discussion of Issues

Both examples given here have suffered from the limitations of the software development tool in that the user has had to accommodate the short-comings often through an increase in cognitive load and a lack of transparency of the interface. In both examples, the designers had been trained in HCI skills and had tried to incorporate these into the design. However, full implementation of a design using these principles was hampered by the limitations of the MDT. Of course, one theoretical way around this problem is for the designer of the application to use a more 'suitable' MDT but this is not practical for a number of reasons. First, the assumption is made that there *is* a more 'suitable' MDT available; secondly, that the developer actually has sufficient in-depth knowledge of all available alternative MDTs to make an alternative choice; thirdly, it is assumed that the finance is available to purchase a different MDT for each application if necessary. Again, this is not really practical since, for example, MDT1 above cost around £6,000 for the basic development set.

Consequently, the only solution is to modify the design to suit the MDT and in this the designer has a responsibility to inform the user of the limitations of the MDT and also of the lack of features should this be applicable. Where the user population is known (as in MDT1) this is feasible at the design stage, but where the application software is to be released into the public domain (as in MDT2), then this is only practical through advertising and user help facilities. Yet again, it seems, that the user is the one who has to cope with the system's short-comings.

Conclusion

This paper has shown through two small examples that lack of features leading to an interface designed to HCI standards at times can be ascribed to the limitations of the MDT used. Thus selection of the optimum tool is crucial, together with the designer having a full knowledge of HCI guidelines so that the problems caused can be minimised through an optimum solution.

References

Bunzel, M J and Morris, S K, (1994), *Multimedia Applications Development using Indeo Video and DVI Technology*New York: McGraw-Hill Inc.

Heinmann, S R and Lusk, E J, (1979), *'Health Facility Planning: An Example of a Decision Flexibility Approach'*, Operational Quarterly Review, **27**(2, ii), 449-457.

Mason, C, (1996), *'Desktop Virtual Reality in the RAF'*, Paper given at Multimedia in Business Conference, June, 1996, Cheltenham & Gloucester College of Higher Education, Cheltenham, UK.

MoD, (1996), *Interim Defence Standard 00-25 Human Factors for Designers of Equipment (Part 13 - Human Computer Interaction)*, Glasgow: Ministry of Defence.

Nielsen, J and Mack, R L, (1994), *Usability Inspection Methods, New York:* John Wiley & Sons Inc.

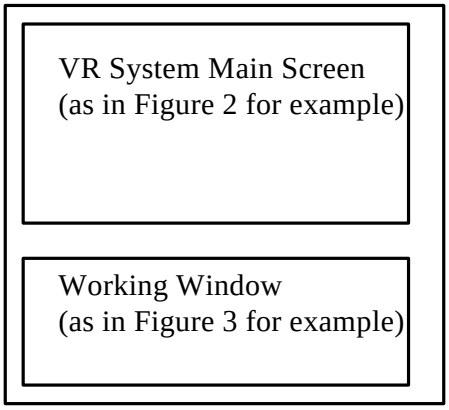

Figure 1 - Initial Prototype (simplified)

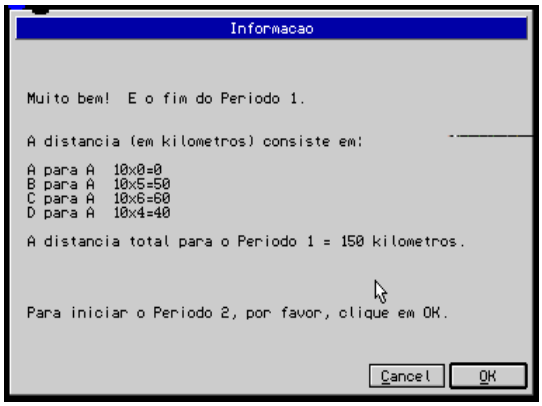

Figure 2 - a Typical Screen showing mathematical Explanation