HCI Evaluation

Submitted to University of Wolverhampton – 15th Dec 2006 Systems Development and HCI module.

By Stacey Greenaway

Contents

1. Methods of HCI Evaluation

1.1 Planning the Evaluations.

Evaluations will be conducted on the ATM prototype, produced for the group work element (1) for this assessment and also on Barclays Online Banking system. I chose an online banking system to evaluate as it shares some of the functionality of the Additional Services section of the ATM prototype.

Before creating a usability testing strategy, I needed to undertake more research to find appropriate ways to carry out the evaluations, but also to have a greater understanding of what conditions I was evaluating for. I gathered this information from lecture notes, but also from reading more widely about usability evaluation. I have been researching usability evaluation for digital libraries for another module running consecutively to this one, this research enabled me to gain considerable understanding of the purpose and value of usability evaluation and HCI principles. I have not referenced these papers as they are not strictly relevant to this assignment.

Based on research and a realistic appraisal of my skills and time frame for the evaluations, I decided to conduct user testing with questionnaires and observation and also informal Heuristic Evaluation, following Neilson's guidelines for both tests. I decided against a Cognitive Walkthrough evaluation, despite the fact that it is suited to prototype evaluations for walk up and use systems and ideal when a good range of users are not available Hertzum and Jacobsen (2001); I felt it may be more time consuming to learn and to establish an adequate set of tasks, to complete a thorough usability evaluation due to my limited experience in the area. Also, I wanted to chose an evaluation technique that could be used for both the ATM and website evaluations, in order for the two evaluations to be compared, it is my opinion that Cognitive Walkthrough would not be a suitable evaluation tool for a website that had been 'live' for many years, where I do not have access to all areas of the system, or an involvement in the design and development process.

This being said, I adapted an element of the Cognitive Walkthrough technique into my evaluation strategy in order to give the process more structure. I established a sequence of tasks for the user to perform; I decided to leave out any detail and just give a title for the task, I split the tasks into the appropriate section, which refers the user to the correct menu. I did this to test navigation, i.e. to see if users can find how to get a balance without any instruction of how to do it. These tasks will also form the basis for the Heuristic Evaluation that I will carry out on the ATM prototype and also be adapted to correspond with the online banking system. I formed the tasks based on my knowledge of the prototype, as one of the developers I knew what sections of the prototype were fully functional and could provide the best usability results. Although Pay Bill and Pay Credit card are not fully functional in terms of database interaction they are in terms of a walkthrough of what would happen on a fully working system and so were worthy of being evaluated as any usability problems in feedback and navigation could be highlighted. See Appendix A for the full set of User tasks.

1.1.1 User Testing

With the tasks established I needed to ascertain how to record the users' interactions with the system. I decided upon a questionnaire for the user to record their own experience and also planned to observe the users and make notes on their behaviour and satisfaction with the system. The questionnaire was split into two sections the first capturing how successfully the user could interact with the system, measuring success based on Neilson's heuristics. The second section measured how satisfied or dissatisfied the user found both the appearance and functionality of the system, again inline with Neilson's heuristics. This was a far more difficult task than I would

have at first imagined, further discussion of these difficulties can be found in section 2.1.1. The questionnaire given to users can be found in Appendix B.

A group of users were chosen who, whilst all familiar with ATM's, have differing ability with computers ranging from novice to expert. The users had an age range of 25 – 40. I recognised that some users may find it hard to understand the concept of a prototype and have difficulty interacting with the system as if they were at a "hole in the wall" and not sat at a PC. By testing with users of differing computer literacy it would be interesting to observe whether ability with computers affects how well the user can ignore the PC controls and place themselves in a "hole in the wall" situation. For this reason, I decided to not use users of an older generation (over 50) because the users in this age group I had at my disposal would have trouble understanding the idea of a prototype of an ATM and this would prove too much of a barrier to give reliable usability results. Ideally though and with more time to evaluate, it would be good to use a more varied range of age group, it would be interesting to see how an older generation would interact with the additional services section, especially with a variety of users ranging from no computer experience to advanced knowledge of computers.

1.1.2 Heuristic Evaluation – ATM Prototype

The Heuristic evaluation of the ATM prototype will be carried out by myself, the user tasks outlined for the user testing will be followed and the interactions evaluated in accordance with Neilson's heuristics. A record sheet has been created based on an example given in lecture notes (week 8 – User Interface Evaluation), this can be seen in Appendix C. It allows for usability problems for each screen of the system to be recorded accounting for their severity, frequency, persistence and impact on the users interaction with the system. The severity rating can be used to assess costs to fix the problems, Hartson (2004) order usability problems based on their severity rating and form a costs analysis of the evaluated system. This allows for budgets to be assigned to the most severe problems such as navigation dead ends, over consistent font or colour change. This aspect of Heuristic Evaluation makes it especially important that the process is done during the software lifecycle when problems can be more cost effectively corrected, in reality a large proportion of evaluations take place after deployment.

1.1.3 Heuristic Evaluation – Barclays Online Banking System

The user tasks created for the ATM prototype were replicated in an heuristic evaluation of an online banking system, Barclays Bank. I did this because a full evaluation of the system is unrealistic, as I was not involved in development, so am unaware of all the functionality available and may not have access to all the areas of the website. Also as I will be evaluating using my own accounts and my own money and did not want sensitive information to be made available, also it is difficult for example, to evaluate paying a bill when I have no bills to pay. Appendix D shows the tasks I created for the evaluation. Usability problems will be reordered on the same record sheet used for the ATM evaluation, see Appendix C. I expect there to be differences in the type of problems highlighted, mainly due to the web based platform, but also, the online banking system has been 'live' for a number of years and I would expect it has undergone previous usability evaluations that would have highlighted most problems.

1.2 Results of Evaluations

1.2.1 User Testing

Unforeseen circumstances lead to my main user group falling through at the last minute, leaving no time to organise any more users. I therefore only managed to do one user testing which was disappointing as the results were very interesting and valuable feedback was gained.   The user was a novice computer user so I was unable to analyse whether computer literacy affected interaction with the prototype. The users completed questionnaire can be seen in Appendix E, my observations of the user testing can be found in Appendix F.

To summarise the results, the user had difficulty ignoring the mouse and keyboard controls, he found some of the instructions confusing both in the system and in the questionnaire, but on the whole managed to navigate well around the system and complete the majority of tasks.   The user highlighted some common problems, namely the cancel button not working and also the clear button did not put the focus back into the input box which forced the user to use the mouse, which we wanted to avoid with the prototype as it would not correctly model the real world system.

Some major usability problems were highlighted that had been over looked during testing, they violated the heuristic of giving a user choice and feeling in control of the system.   The two major examples of this were in the Pay Credit Card and Transfer Money sections. Once the user had successfully entered their Credit Card customer reference number, they are only offered the options of paying a minimum payment or other amount, there is no option to cancel without paying.   When transferring money the user is not offered an exit route if they change their mind about the transfer, the cancel button does not work, and although the back button works, the account number variable is not passed back, so the system 'forgets' who the user is, but without logging out, so in the real world the card would be in the machine and there would be no way of getting it out.   It is not completely catastrophic, the user is forced to press back until he returns to a screen where cancel can be pressed – but obviously, this is not acceptable.   This user testing showed navigation and menu labelling on the whole to be intuitive and consistent, however, a problem was highlighted in the placement of account Balance.   It is not clear where you get a balance from, the user went to the wrong section to find a balance and therefore couldn't complete the task.   I purposely left the find balance task vague as I wanted to test whether or not it should have been modelled as a separate section.   It would have been beneficial to have been able to conduct more user tests to see if this problem occurred amongst all users, and what ratio could find balance to who could not and which sections they went to to find it.

1.2.2 Heuristic Evaluation – ATM Prototype

As I conducted this evaluation myself after observing the user testing, my results are biased to things I had already learned from the user testing. Also as the interface designer and one of the programmers, it is unlikely I can be impartial in my evaluations and highlight all problems when I am so close to the project.   The evaluation becomes more of an extension to system testing due to my role in the development team.

However, I tried to be as impartial as I could be to carry out an objective evaluation. The usability record sheet I filled in can be seen in Appendix G. The results show I found numerous high priority usability problems, mainly concentrating on navigation, broken buttons or lack of suitable navigation options; as discussed above, section 1.2.1. The heuristic evaluation highlighted similar problems to the usability testing, however, I also evaluated usability positives found in the system. Based on Neilson's

heuristics, the results of my evaluation found that the interface had a minimalist clean design and that it was primarily consistent throughout (apart from one or 2 easily fixed exceptions). The font was clean and easy to read, instructions were clear and concise, there was sufficient error messaging and further options available. Users are given the opportunity to correct errors in the most part, some improvements need to be made to this. Menus are clearly labelled and sections clearly identified, with the one exception of account balance. There were no catastrophic usability dead ends, but there is room for improvement in the Pay Credit Card and Transfer Money sections.

1.2.3 Heuristic Evaluation – Barclays Online Banking System

The tasks outlined in Appendix C were evaluated as to whether the system meets Neilson's guidelines. A usability evaluation record sheet was completed which can be seen in Appendix H. Below is a list of the main findings of the Heuristic Evaluation on Barclays Online Banking system:

- User given plenty of choices, both services and options within a service.
- Clear error messaging throughout although some times the message is too general not highlighting the specific error.
- Effectively reduced the amount of recall needed by the user and relies on recognition, also allows for customisation of main features which allows for recognition to be easier.
- Allows saving of bills etc. time saving, shortcuts for experienced users.
- Erratic behaviour of browser back button - logs out, generates error, crashes browser or works.
- Provision of back button on some screens – not consistent
- Does not allow for ease of correction, especially due to the erratic nature of the back button mentioned above, if the browser back button is not to be used then it should be disabled and offer an onscreen back button that the system can control.
- Plenty of help and documentation including an online demo. Clear help buttons next to sections as well as good descriptions of the task.
- Clear step by step indication of progress through a process. Each process labelled.
- Sometimes the system is too slow taking a long time to process simple actions in forms.
- Plenty of confirmation form processes and alternative options such as back, cancel, confirm. Good descriptions of exactly what will happen if you choose an action.
- Navigation confusing, two separate navigations one text links and one in select box – which I assume is a shortcut menu, but it is not labelled. It is hard to distinguish banking navigation form Barclays services navigation. Should be more distinction.
- Too many adverts to distract the user from the actions they are undertaking. The lack of distinction in navigation is victim to this as the bank wants the user to make a mistake and click on an advert link instead of a banking service link.
- Design responds well to increasing the font size, content still readable at a large font and navigation still clear and usable.
- User is made to feel secure, padlock visible in bottom right of browser and description of secure certificate provided. This is beneficial to combating phishing attacks.
- It is clear to the user they are on Barclays site as the logo and navigation is consistent throughout.
- No usability traps – navigation dead ends, user can always complete a task or cancel without completing.

In general the site has few usability problems and would be accessible by a wide range of users. However I would argue that some of the screens are a little too cluttered with too much information that could be removed and put into help files for in experienced users. There are too many adverts which are distracting, as the eye is taken to the adverts rather than the information you are trying to retrieve. The back button was a major issue, as I found it very frustrating that I could not go back with the browser and yet no other back option was available. An alarming issue I noted in evaluating the login procedure was that I can log in using my maiden name, indicating that there are not sufficient checks in place on the surname entry field.

As a user of Barclays Online Banking service I feel that I have a reasonable amount of knowledge of the system, having used it for 6 years, to have carried out a relatively thorough evaluation, however, my results could show bias as I will have pre defined ideas about functionality of the system.

2. Critical Evaluation of HCI Evaluation Methods.

2.1 Limitations of Evaluation

A barrier to evaluation and in creation of the prototype was the limitations enforced by the ATM hardware. From as early in the process as conceptualising the system, it proved difficult to overcome perceptions of a system using a mouse and keyboard and controls common in most systems, such as select boxes. Whilst as a designer and developer of the prototype I had to quickly overcome this barrier and think very carefully about what hardware is available at a hole in the wall machine. This had greatest impact on the interface design as it was difficult to offer minimal design and allow for recall of information without having the option of using select boxes. The greatest problem was found in designing the additional services sections; it seems that the functionality for this section was adapted from services available in an online banking environment, that are not suited to a hole in the wall situation, i.e. viewing last 20 transactions, there is not enough screen size to do this adequately. It is for this reason primarily that I chose to evaluate an online banking service so that the functionality could be compared through evaluation.

2.1.1 User Testing

Reasoning that I had experienced this barrier when interacting with the prototype, it was fair to assume that users would also have an issue with trying to mimic their actions at an ATM (hole in the wall) when using the prototype at a pc with keyboard and mouse. Environment is an important factor in evaluating user interaction with a system, it is impossible to get accurate results about how a user will interact with a hole in the wall ATM when their interactions are not being evaluated in situ. For this reason I tried to place as much emphasis in the user tasks as I could on using the ATM prototype controls instead of the keyboard and mouse. I had hoped that if prompted to not instinctively use the mouse and keyboard they may look for the alternatives. I am disappointed that the full user testing I had planned fell through as I do not have enough results to fully evaluate different users' interactions with the prototype. If I was to do the evaluation again I would plan a contingency user group at a different date so if one fell through I would hopefully have enough quality results from the second user group.

The most difficult aspect of user testing was creating the questionnaire. The challenge lay in trying to generate questions that would be easily understandable by a wide range of users, with differing levels of technical knowledge and cognitive ability, whilst referring the questions to Neilson's heuristics, which do not use terminology that is understandable by the majority of users. A further challenge was in summarising what I wanted to capture in one statement or question, whilst still making sure I received useful feedback to the heuristic.   Another challenge was in trying to tailor the questionnaire to be relevant to the 30 separate user tasks I had created. I did not want to create 30 different questionnaires, as much as the user would not want to fill-in 30 questionnaires! I felt it was important that the questionnaire be limited to 2 sides of A4 so as not to deter the user from participating in the testing, but I also needed to leave enough space to write comments. The user testing I conducted proved to be as much an evaluation of the prototype as an evaluation of my questionnaire writing skills. The user found some of the instructions on the questionnaire confusing and found he didn't have enough space to comment about the individual tasks.   He felt that a tick box was not sufficient enough to record the problems he found.

2.1.2 Heuristic Evaluation – ATM Prototype

As previously stated I carried out this evaluation as impartially as I could, but as the interface designer of the system, I am too close to the project for the results to be free from bias. A benefit of being so close to the project was that I knew the intricacies of the system and could evaluate every aspect of it, however I found I got more concerned with issues related to testing than in really considering the different needs of users of the system. It is extremely difficult to model how different users will perceive and interaction, all users are different and have different abilities. Whilst it maybe easier to consider how visually impaired users may have difficulty using a system and evaluate whether or not the prototype is accessible to them, it is extremely difficult to evaluate how different users may be able to recall information or follow instructions. I would argue that usability evaluation, such as Heuristic Evaluation, that is conducted by experts, is extremely useful at highlighting usability problems that compare to well-established guidelines, that make for a generally well useable site; but unless there is user testing in place that actively tests the system with users of different cognitive ability, visual and physical impairment, it is difficult to fully evaluate any problems they may face in interacting with the system.

The table below models one user interaction with the ATM Prototype, it details what senses are used by the user and what cognitive skills they require to complete the task:

Task	Input	Process	Output
Change PIN	Vision Touch	Long Term memory – old PIN Short Term Memory – Confirm PIN	Enter three pin numbers for one task.

The limitations of the ATM hardware creates many problems in trying to accommodate the needs of all users, the same problems must be faced by designers of systems for small hand held devices. It is difficult to provide constant navigation on every screen as there is not enough screen size to fit information and navigation. The user has to rely on short term memory to remember which service was on which menu and how to get back to the menu to access the desired service. The screen size also limits the amount of information that can be relayed to the user, ATM hardware add a further limitation to this in that users stand at a distance from the screen, so the font needs to be large enough to be readable, which reduces the amount of information that can be displayed. There is little availability to provide shortcuts to experienced users as there is a limit to the amount of form controls that can be used. Through evaluation I feel the ATM prototype meets a considerable amount of the HOI guidelines and makes considerable attempts to be accessible by a range of users, through aesthetics and layout of the interface, system instructions and feedback.

2.1.3 Heuristic Evaluation – Barclays Online Banking System

Through evaluating both the ATM prototype and the online banking system using a similar set of tasks and evaluating to the same guidelines, it was clear to see how the limitations of the ATM hardware effects what services can be made available. The online banking system can make itself more easily usable by users of different cognitive ability, as it can offer methods to customise the system so that it is tailored to individual users needs i.e. change font size, or label accounts. By being able to change font size the system can be made accessible by visually impaired users, but

not at the expense of able bodied users.   By allowing customisation such as labelling accounts, it allows for users of different memory skills to tailor the system, so that instructions make more sense to them and they can more easily manage their accounts. This functionality cannot be made available with ATM hardware due to the restricted amount of controls and screen space available. Through evaluation of the two systems and issues raised during the development process, maybe a possible development of ATM's in the future will be to remove them from the traditional hole in the wall infrastructure and for them to become more akin to online banking systems, allowing the user more choice and control over their banking activities.

3. Process Evaluation

The main failing of our project was time management and the ordering of tasks. Although a Gantt chart was produced it was not adequately followed, we could have made much better use of this tool. We spent a lot of time on requirements analysis, UML diagramming and Interface design, which were important activities and served our project well, but they were done at the expense of the database. The team member whose role it was to create the database needed more support than was given, as errors were over looked in the modelling of entities, normalisation and creating of the tables. These errors caused us problems later on in implementation and by then it was too late to fix them. The order in which we prioritised tasks also caused us problems in implementation, we gave precedent to the front end design of the system over the back end functionality. In hindsight, it would have been advisable to concentrate on functionality first, giving the system a basic interface, then perfecting the interface after testing was complete. This was more evident as two group members were working consecutively to programme the system. As the interface designer it would have been easier for me to have added the interface to existing controls once functionality was complete. We were aware that this would be a better solution but made the decision that as time was at a premium we would do both tasks together, this was a mistake. This error in judgement was further highlighted during testing and also through the usability evaluations.

Problems encountered in programming concerning the database, delayed the completion of functionality by approximately three weeks. Although the front end was designed by this point it did not fully reflect every stage in a transaction and the layout chosen in some cases would have made programming more difficult than it needed to be. At the late stage that this was discovered, the remaining programming had to be completed in the last week as well as finalising the front end to meet HCI guidelines. Testing also had to be done in this week. In the end, testing was conducted only by myself on the half of the system that I had working. This is clearly evident, the screens that were not tested stand out in the usability evaluations as they are full of dead ends, broken buttons and errors that have not been tested for by the programmer. This is a project management error however, as we should have allowed more time for testing so that the interface could have been fixed inline with usability guidelines. It would have been beneficial with more time to leave up to three weeks after programming is completed, for testing and usability evaluation and then fixing of problems. If there had been a networked system in place, where both myself and the programmer worked on the same files, then this would not have been so much of an issue, but as we worked individually on separate files it became a substantial problem.

In terms of interface design, which was my main role in the project, I created a clean, consistent minimalist design that is easy to use and allows for informative instructions and feedback. However I spent too long trying to perfect instructions and concerning my self with consistent font and control placement when I should have designed a basic wireframe including spaces for system feedback and let the programmer create the message. It could always have been corrected at a later stage through testing and evaluation. This is an example of failings that occurred throughout the project , in that we all had our own roles that we were completing and there was not enough cross over of skills so that there was support for each member to check what had been created. When you have designed something you are too close to it so it is hard to spot spelling mistakes or extra instructions that should have disappeared. The programmer did not pick up and correct my mistakes, same as nobody picked up and corrected the database errors, in sufficient time; equally we struggled with only one programmer, ideally we needed at least two

dedicated programmers, although I helped out with the programming I had other roles to fulfil on the project and could not give as much help as was required.

To summarise what lessons I have earned, from the process of creating the ATM prototype, that I will apply when I am next involved in an implementation project, the main lesson is to allow adequate time for testing and evaluation of the system and subsequently for the errors found to be fixed. It would be beneficial to include user testing in the development lifecycle as valuable information was elicited through the evaluations that would improve the system. The database needs to be more thoroughly designed and tested before programming begins, it is the back bone of any system and it is very difficult to correct database problems once the implementation stage has begun. A basic interface design can be created first, but the finished interface needs to be added after the system functionality is complete and testing has been conducted. Time management is crucial, allow an extra week for external unforeseen circumstances aside of system development e.g. ill health. Create a Gantt chart at the very beginning and look at it regularly, do not be afraid of changing it to better reflect what needs to be done once the project is underway.

Despite the problems we encountered and countless aspects of the project we could have done better, I feel that in the short time frame we had to complete the project we have managed to produce a working prototype, which looks professional and adheres to HCI guidelines. As a group we worked well together, each member of the team was committed to the project and worked to their strengths.

4. References

HARTSON, H.R. SHIVAKUMAR, P. PÉREZ-QUIÑONES, M. A. (2004) Usability Inspection of Digital Libraries: A Case Study. *in International Journal on Digital Libraries*, Springer Berlin / Heidelberg, 4(2), pp. 108-123.

HERTZUM, M. JACOBSEN, N. E. (2001) The Evaluator Effect: A Chilling Fact about Usability Evaluation Methods. *in International Journal of Human-Computer Interaction*, **13**(4), pp. 421-443.

JOHN, B. E. PACKER, H. (1995) Learning and using the cognitive walkthrough method: a case study approach. *in Proceedings of the SIGCHI conference on Human factors in computing systems Denver, Colorado, United States* pp. 429 – 436.

FRANZKE, M. (1992) Using the cognitive walkthrough in iterative design. *in Posters and short talks of the 1992 SIGCHI conference on Human factors in computing systems Monterey, California,* pp. 37 - 37

WIEDENBECK, S. WATERS, J. SOBRADO, L. BIRGET, J.C. (2006) Design and evaluation of a shoulder-surfing resistant graphical password scheme. *in Proceedings of the working conference on Advanced visual interfaces Venezia, Italy,* pp. 177 – 184.

Appendix A – User Tasks

Usability Testing – User Tasks

SETUP:
There is no need to install the application, it is installed on the flash drive, but the flash drive needs to be set to U:// for the system to work.

NOTE: - This system is a prototype for a new ATM machine, therefore it is required that the keyboard is only used to enter data when it is requested. Only a number keypad and the on screen buttons (which mimic the ATM hardware) will be available in the finished system.

LOGIN:

1. Mimic insert card – Enter the Account number 15642008

2. Enter incorrect Pin – 5555

3. Enter Correct PIN – 5500

TRADITIONAL SERVICES

CHANGE PIN:

4. Change PIN to 1234

5. Enter incorrect digits and correct (using ATM buttons not keyboard)

6. Press Back Button

7. Press Cancel Button

WITHDRAW CASH:

8. Withdraw £400

9. Withdraw £50

10. Enter wrong amount and correct it (using ATM buttons not keyboard)

11. Press Back Button

12. Press Cancel Button

ADDITIONAL SERVICES

PAY SERVICE BILL:

13. Enter Customer Reference number: 12345678

14. Enter incorrect digits and correct mistake (using ATM buttons to correct not keyboard)

15. Press Back Button

16. Press Cancel Button

PAY CREDIT CARD:

17. Enter Customer Reference number: 987654321

18. Enter incorrect digits and correct (using ATM buttons not keyboard)

19. Pay minimum payment

20. Pay £75

21. Press Back Button

22. Press Cancel Button

GET A BALANCE:

23. Do not Print Receipt

24. Print receipt

25. Press Back Button

26. Press Cancel Button

TRANSFER MONEY:

27. Transfer £50

28. Enter wrong amount and correct it (using ATM buttons not keyboard)

29. Press Back Button

30. Press Cancel Button

Appendix B – User Testing Questionnaire

USER OBSERVATIONS.

Name:_____

Did you manage to complete all the tasks?　　　　Yes / No

If no what stopped you and for which task?

Please enter task number in top row and tick applicable condition:

Task No (1-30)																		
Error message																		
System crash																		
Button not working																		
No available navigation																		
Confusing instruction																		
Other																		

If other please give details (state task no if more than one):

```



```

Was there any aspect of the process you found frustrating?
Please give details:

```



```

On a scale of 1 – 10 how satisfied were you with the following?
1 = very satisfied, 10=very dissatisfied.

Satisfied ⟶ Dissatisfied

Case	1	2	3	4	5	6	7	8	9	10
General appearance										
General ease of use										
Font										
Colour										
Placement of controls										
Amount of Controls										
Ease to navigate										
Placement of Back/Enter buttons										
Useful system feedback										
Easy to understand instructions										

Any other comments?

Thank you for your time.

Appendix C – User Tasks – Barclays Online Banking Evaluation

LOGIN:

1. Enter incorrect Surname

2. Enter incorrect reference number

3. Enter incorrect passcode

4. Enter incorrect letters for memorable word

5. Enter correct Surname

6. Enter correct reference number

7. Enter correct passcode

8. Enter correct letters for memorable word

SERVICES:

9. View Balance.

10. View Statement for one account.

11. Change statement time period to cover 3 months.

12. Observe options in pay bill section.

13. Transfer £50 from one account to another.

14. Change personal details.

15. Note services offered – how easily find them, clear what is offered? Easy to understand terminology?

16. Test Back Button.

17. How secure do you feel?

18. Does the site instil trust?

Appendix D – Usability Evaluation Record

Usability Problem Record Sheet

Interface Evaluation:	
Evaluator:	
Date:	
Task Description:	

Screen No.	Description of problem	Heuristic violated	Frequency	Impact	Persistence	Severity rating

Appendix E – User Testing – Completed Questionnaires

Appendix F – User Testing – Observations

User – Carl Greenaway

- Pressing graphical buttons not the buttons that mimic the ATM hardware.
- User stated it's tricky when using a mouse and not the ATM machine itself
- Found instructions confusing – both in questionnaire and on system.
- Found it very hard to ignore the mouse and keyboard controls and to use the back/clear/cancel buttons of the prototype to correct mistakes.
- **Error found – Multiples of 10 is not checked for, user could withdraw £55.
- Clear button - not obvious that it is delete
- **Error found – When asked to confirm if amount to withdraw is correct in Withdraw cash screen, if click NO, focus is not put back to the input box to allow the user to re enter the amount. It is if the clear button is pressed.
- User had no problem navigating through menus, seemed to understand labelling. Informative text used.
- User found the cancel button annoying, that it logged out every time, wished it would have took him to a menu screen.
- **Error found – Highlighted a font change in the Customer ref input box for Credit Card payments.
- **Error found – Cancel button not working when entered credit card customer reference number.   User has no choice but to pay the bill, cannot cancel without paying. Usability trap as reduces user choice and navigation dead end.
- **Error found – Confusing instruction - on credit card confirmation screen "please confirm details" header should disappear once the Thank you message appears.
- **Error found – Clear button does not reset the Account Number input box, there is no way of correcting a mistake in this box without using the mouse and keyboard, which does not reflect the real word use of the system.
- Task 18 – Found the instruction to re enter customer ref no confusing as there is too much information on screen, also asked to go back for more options or cancel to return card.
- Task 23 – User couldn't find where to get a balance. He went to mini statement, which is not functional for the prototype. Found view all accounts confusing, in particular the task states "do not print receipt", however there is only an option to print receipt and no other option.   It is not obvious balance is in view all accounts. This is a major design flaw in the prototype, we should have thought more of users' needs and added balance in a separate section, maybe in traditional services and not adhered so stringently to the requirements of the brief (client).
- **Error found – Task 23 – Cancel button doesn't work.
- Transfer Money – Not enough distinction between the Transfer From and Transfer To screens.
- Transfer Money – Cannot clear amount if make mistake
- Transfer Money – Lots of account number to remember, but user did state if the system was using his own accounts, he would recognise the numbers, it is only a problem because he has never seen the number before and so doesn't know if they are correct or not. Our system meets the heuristic of recognition over recall.
- Transfer Money – Major problem – The Account number disappears when the back button is pressed, the variable is not being passed back, then cancel doesn't work so can't get out of the system and the system has 'forgotten' who the user is, but the user hasn't been logged out. In order to exit the user

needs to press back to get to a screen where the cancel button works. Then log back in again.

General Observations aside of tasks
- Change Pin too obvious – it is not the first and most important task to a user of an ATM.
- Withdraw cash should be the first item on the menu.
- View Balance available on the first menu screen (traditional services)
- User would like to see return card option.
- Found the meaning of Back/ Clear /Cancel confusing, not clear what they actually do, user not sure that the functionality of them in the prototype accurately represents their use in real world ATM.

Appendix G – Heuristic Evaluation - ATM Prototype

Completed Usability Evaluation Record

Appendix H – Heuristic Evaluation – Barclays Online Banking

Completed Usability Evaluation Record