Discussion of the Usability in Automatically Generated User Interface

Chin Tang Yu
University of Auckland
Auckland CBD, Auckland, New Zealand
cyu045@ec.auckland.ac.nz

1. ABSTRACT

Designing user interfaces is a time-consuming process which takes a lot of effort and resource for a human designer to create a high-quality user interface, and usually the user interface designed by multiple designers on a similar type of device may not be consistent across each other, because of that user interfaces may not guarantee to have the best quality such as usability. Therefore it is impractical to specifically design a user interface for every new system created. Automatic user interface generation eliminates the problem of inconsistency and allow efficient user interface creation at the same time by using a pre-designed model to generate user interfaces with additions of parameter profiles.

This report discusses the usability of automatically generated user interfaces in general terms. Since automatically generated user interfaces are generated through a model, the usability is generally consistent across the generated user interfaces, and the designer has to design the model system itself so the generated user interface have a higher probability to be in high-quality. However it is possible for this automated process to lower the usability of the user interfaces because of its unpredictable nature and the variety of device controllers are increasing. Some suggestions to overcome the weakness of automatic user interface generation process are discussed here. One of the improvements that the generation process may aim for is the ability to create user interfaces with customisable features.

Author Keywords

Automatic User Interface Generation, Ubiquitous Computing, Heuristic Evaluation, Usability Test, Usability Attributes, User Customisation.

2. INTRODUCTION

In the world we are living today, computers have been taken a major role in our daily lives. For example, Introduction of Internet and computer networks, embedded programs and processors are installed into electrical appliances to enhance the quality of the work, the development of everyday devices such as personal computers, mobile phones, PDAs and etc. Through the tremendous development in computing technology, the quality of human and computer interaction are also increased significantly.

In order to allow users to interact with systems, an adequate user interface (UI) for any particular system must be designed in a way that the user can be able to use the system with less difficulty as possible. It also has to be efficient when interacting with both users and systems, i.e. when a user sends a command to the UI, the UI has to process the command and transfer it to the system using the least time possible.

Most of the problems in designing UI often come from UI generation. Some of the notable problems include [1]:

- The number of variety of system and devices are so many that it is not feasible to have a single UI design.
- It is necessary to avoid high-quality UI but the UI itself required to have high-level functionalities.
- It is impractical to design UI for individual users because users have different preferences.
- Designing UI is expensive, error pone and time-consuming process. Therefore it is not ideal to design UI every time a device is built, especially with a human specialist designer. [2]

Therefore to generate UI efficiently it is necessary to generate UI automatically using a pre-designed model rather than designing it every time a system is built. Automatic UI generation help to convey ideas to developers about the quality of the software in terms of how easy to manipulate devices with the UI, how efficient and fast does the UI functions on the device and etc. This paper mainly focus on how automatically generated UI have influences on the system's usability.

3. AUTOMATIC USER INTERFACE GENERATION

Automatic user interface generation is the process of creating a user interface in a device or system using an automated method. This idea was first motivated by the advantages of

separating the UI from the input and output level, and also the idea to allow developers to produce a high-quality UI without having much knowledge in UI design. [1] Automatic user interface generation has been in research for over decades and is still an ongoing research topic today.

The basic idea of how automatic user interface generation works is that it is based on model designs. The UI generator creates a UI using a meta-model as a base template. Other information such as the type of device, graphical resolution, user preferences, required input/output fields for that particular device etc. becomes the parameters of the generator, which allows to generator to create the UI in a particular form given by the constraints. [2, 3]

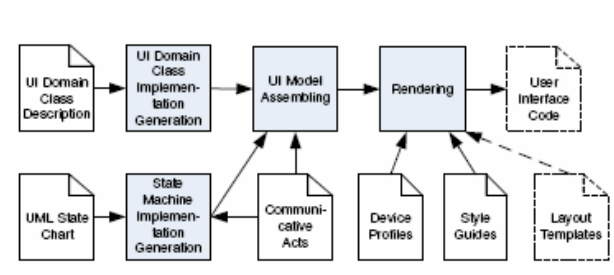

fig 1. An architecture of a fully-automatic UI generation model [2]

Automatic user interface generation is useful in computing because it creates a general UI for particular type of devices, allows the user to manipulate the device with ease. The structure of the UI is defined in the template model; if the template model is well-designed it is most likely that the UI generated using the template is in a high-quality. It also provides consistency because the UI s generated will have the similar structure and layout, i.e. the interface of similar devices should be expected to look and feel the similar way.

Automatic user interface generation reduce human effort to specifically design a UI for each devices. There is a range of variety of devices which the UI design must consider about the specification and requirements of different devices, e.g. the display or graphical resolution of the device, input and outputs required, running performance etc. Because it is time consuming and requires a lot of effort for human designers to produce such product, and moreover the designs are generally inconsistent across each other, it is more favorable and less expensive to generate UI automatically.

Automatic user interface generation also enhances ubiquitous computing by generating UIs into controllers to allow user to control systems which are embedded into the environment. [3] Processors embedded into other systems, such as electrical appliances, needs to be able to be controlled by users in order to do the tasks that the user desires. The only place for the UI to be generated is either the system the processor is embedded into, or any handheld devices or controllers which the user possesses. Sometimes the UI provided by the system may not be user-friendly, therefore a better UI may be needed to be generated into another device, and in order to do that it has to be automatically generated.

There are software development packages, such as SUPPLE [3] and UNIFORM [4], available to automate UI generation in a higher standard of quality. These software development packages can either be licensed or open source to the other developers.

4.1. USABILITY

Usability is the measures of how well does the user to be able to interact with the system in terms of the quality of the system. Since the system highly depends on the UI to provide interaction between the user and the system, therefore the usability of the UI often determines how well does the user to be able to use the entire system.

To determine the usability of a system, we can use five quality attributes to determine the usability in the basic level [5]: Learnability is the quality that whether the user can easily recognise and learn how to use the system. This is often measured by how long does a user takes to complete a task after being trained; Efficiency describes how fast can a user complete a task in the system, or how many tasks can a user complete in a set time frame; User retention of the system describes how well does the user memorise the functions of the system and be able to use it after a long period of time; Error rate measures the number of mistakes the user makes when doing a task in the system. This is to indicate that whether the system has a tendency to mislead the user to perform a particular function; and finally user's satisfaction is determined by how the user feels about the system. These quality attributes can be broken down into a much more detailed standard to obtain a better usability analysis. And from the attributes rules can be developed, such as the Nielsen's 10 heuristics rule. [6]

Most of the people thought that the usability of the user interface lies only in the visual and the appearance level; however this is a big mistake. The usability of the user interface is determined by both the visual appearance and the ability to transfer instructions or messages between the user and the system. Therefore it is important to consider the interaction and the appearance to the user at the same time, and a well-defined plan should be prepared before starting to design the UI.

In automatically generated UI, it is worth discussing that whether automatically generated UI improves the quality of usability or worsen it. Some of the factors that may affect the usability of these UI are, 1) since UI are generated from a pre-designed model for multiple devices, it is uncertain that the usability can stay consistent across the range of devices if using the same model; 2) different users may be using the system with the same generated UI, and they may be coming from different backgrounds and knowledge, therefore it is important for UI to cope with inconsistency between users; 3) some systems may have a built-in UI that may not be user-

friendly to the user, a solution of this problem is to generate a UI into another control device, the generated UI for the system may have a positive or negative effect in usability of the system.

4.2. CASE STUDY: USABILITY TEST ON PUC

To examine the usability of a user interface, a usability study is the best way to evaluate whether a system is user-friendly to any types of users within the target range. A usability study is the study of a system which involves real users to perform tasks with the system, and observe the behaviour of the users to evaluate the usability of the system. Usability study pinpoints most of the usability problems of a device because using real users to use the system is the only way to discover usability problems accurately.

fig 2. Example of a user interface generated in a personal universal controller using UNIFORM [Nichols]

One of the user studies found tests the usability of UI of printers which is automatically generated in a personal universal controller (PUC). [1] The UI generator generates the UI for two brands of printers into the PUC, which allows the user to control the printer without using the built-in UI of the printer. Then the users who were selected to conduct the study will attempt to use the printers with the built-in UI, the UI generated in the PUC, and another UI in the PUC with the design of the other brand of the printer; two UI each for each brand of the printer.

The user study conducted shows positive results on using the UI in the PUC, which is significantly better comparing the results given by using the built-in interface. Hence it shows that automatic UI generation is a viable process because it is able to improve usability. [1] The reasons are,

- Firstly, it is able to move a UI from the device to a control device with more interactive capabilities. For example, a printer uses an led display may have limited usability because of the controls and functions are not obvious, but with the technology of automatic UI generation its UI can be generated into another devices such as PDAs or mobile phones which have better display compare to the printer's built-in UI. Hence the usability of the system is improved.
- Secondly, UI can be generated in a way that users with experience using the system can recognize the functions in the generated UI, as UIs are automatically generated, it must be consistent throughout all similar devices. Therefore it is easy to create a general UI for any particular system. This is assuming that the model used in UI generation is well-designed in the learnability of the system.

It also suggests that there exist some external factors such as individual user preference may affect the usability of such UI, and therefore user customisation of automatically generated UI can be considered as future work of this topic.

Apart form usability tests, heuristic evaluation of a system is another way of analysing usability. The purpose of this evaluation is to judge a particular system with expertise and identify any possible issues that may affect the usability of the system. However developers also prove that from 20% to 51% of the usability problems found were actual real problems of the system. [7] The main advantage of conducting heuristic evaluation is that it is the most intuitive way and act as a starter to analyse usability issues.

4.3 POTENTIAL USABILITY ISSUES

Although automatic UI generation supports usability, it is also possible that automatically generated UI can reduce the quality of usability if the base of the UI generator is not well-designed. One of the reasons is, similar to the problem that adaptive user interfaces are facing, because the autonomous adaptation is unpredictable and is able to reduce the UI's usability. [8] For example, a UI of a particular system is required to generate into a mobile phone and a palm organizer. Between a mobile phone and a palm organizer, there are differences in terms of specification since they are two different types of devices. On the other hand difference in specification can occur between two different systems where the UI is required to be generated. Any of these unpredictable factors can affect the outcome of the generated UI and therefore affect the quality of usability. This is usually a cause of the base of the UI generator with limited design on flexibility coping with the device and system diversity. This issue can become apparent if a new device controller or system is introduced to the computing world.

Another issue involves automatically generated UI is that the design of the UI may limit particular users to use the system in full extent, especially with users with disabilities. [9] For example, users who are motor impaired could not use scroll bars and adjustment tabs, and users who are visual impaired may not be able to read the interface. Since UI generation is

automated it does not have any sense to put considerations about user's ability. There are developments progressing in automatic UI generation to accustom user preferences into UI generation, so the users can control UI into the way they desired, such as the look and feel of the graphical presentation and different modes for controlling the system. However it is both complex and expensive to develop such functionality at this stage. Therefore the functionality is pretty general and limited to the visual/graphical level.

5.1 USABILITY IMPROVEMENT

To improve the quality of the usability in UI often lands on the design of the UI. Because UI is automatically generated from a base or a meta-model, so the design of the meta-model which is used as the base in generation determines the quality of the generated UI. To design a system or a UI it is a best practice to identify and confirm the usability requirements, and communication between designers and system professionals are important. [10] This is somewhat better than designing separate UIs because of inconsistency and expensive usage of resources. A research group proposed the use of high-level models and communicative acts to generate a high quality UI. [2] They also informally assessed the usability of one of the generated UI and prove that with a well-designed generation method it is able to generate UIs that is good enough in terms of usability.

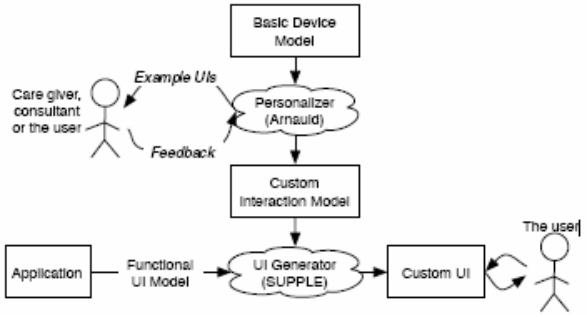

fig 3. Architecture of the UI generation system [9]

To improve the usability of automatic UI generation, redesigning the UI design model into a model which supports user customisation may help users to recognize and use the system with ease. [9] This is especially helpful to users with impaired abilities since considerations are required to broaden their limitations on usability. To generate a UI with user preference, a basic interface model can be integrated with a customized model which contains user's information. Then it is used with the application and generates a UI with customised preferences (see Fig.1). To customise the UI the user of the caregiver needs to provide information to the personalizer system and create a UI model with custom interaction. This helps the generator to generate usable UI because the generated UI can cope with the user's preferences and allow them to be able to recognise and use the system easily.

5.2 UI EXTENTION

One of the proposals to the usability problem is to have a user support system installed in the device as an extension of the UI. This proposal generally comes from the idea that users may not have the suitable thinking process or mental models when using such system, and the extended system would bring the mindset of the user to the correct thinking.

A usability test is conveyed by another research group to aim at improving the usability of a media controller using a user support system to identify the status of the device. [8] A prototype of the user support was implemented in a mobile system. Two groups of participants, one with support and the other without support, attempted to do the tasks assigned with the mobile device. From the test the difference in results from the pre-test and the post-test determines the improvements of learnability, and the results of the supported users determine the ease-of-use of the system.

The outcome of the usability test suggested that the UI with the user support improves the ease of use of the system but reduces in learnability. This is because the user support provides information which makes the device easier to use, but it does not support a user to learn how to use the device quickly. Moreover if the user support system is not learnable enough, the user may have to spend time to learn how to use the additional system, thus take longer to get a task done.

6. FUTURE WORK

One of the difficulties encountered in researching the usability in automatic UI generation is there are not enough research on discussing the usability of these sorts of UI specifically. Therefore this research mainly discusses on the usability of UI in general terms, and the usability issues are potential or possible problems that may encounter when generating UI automatically base on the knowledge on how automatic UI generation works. Although there are some ideas on usability of such UI generation but the information provided are brief, and some of the ideas worth noticing are included in this report as a brief statement. However proof on these statements are necessary and further research and experiment are helpful to develop the automatic UI generation technique.

7. SUMMARY/ CONCLUSION

In the previous researches, automatic UI generation proves to be useful in the computer technology. Basically it provides automatic UI generation without having human designers to specifically design the UI for a particular system, thus saves a lot of time and resources in human-computer interaction design.

In the general level, automatic UI generation can improve the quality of usability and viability of the UI given that the UI model is well-designed. However it can be a weakness to usability at the same time because of its autonomous mechanism is unpredictable and lack of user's consideration to support a wider range of users, notably users with disabilities may be greatly disadvantaged. Therefore future

work on user customisation and support can be a desirable development to improve the usability of automatic UI generation. Several researchers have attempted to design a UI generation system with user customisations and were quite successful. This can be a starting point of improving the automatic UI generation system to be able to create a much more user-friendly interface.

REFERENCES

1. Nichols, J, Chau, D H and Myers, B.A (2007) Demonstrating the viability of automatically generated user interfaces. In M B Rossons (Ed.) *Conference on Human Factors in Computing Systems: Proceedings of the SIGCHI conference on Human factors in computing systems* (pp. 1283 - 1292) San Jose, California, USA: SIGCHI. http://doi.acm.org/10.1145/1240624.1240819
2. Falb, J.; Popp, R.; Rock, T.; Jelinek, H.; Arnautovic, E.; Kaindl, H.; (2007) Fully-automatic generation of user interfaces for multiple devices from a high-level model based on communicative acts. In *System Sciences, 2007. HICSS 2007. 40th Annual Hawaii International Conference on* (pp. 26-36) Waikoloa, HI. http://ieeexplore.ieee.org/search/freesrchabstract.jsp?arnumber=4076422&isnumber=4076362&punumber=4076361&k2dockey=4076422@ieeecnfs&query=%28fully-automatic+generation+of+user+interfaces%29%3Cin%3Emetadata&pos=0
3. Gajos K Z, Weld D S (2004) SUPPLE: automatically generating user interfaces. In J. Vanderdonckt (Ed.) International Conference on Intelligent User Interfaces: *Proceedings of the 9th international conference on Intelligent user interfaces* (pp. 93 - 100) Funchal, Madeira, Portugal. http://doi.acm.org/10.1145/964442.964461
4. Nichols J, Myers B A, Rothrock B (2006) UNIFORM: automatically generating consistent remote control user interfaces. In R Grinter (Ed.) *Conference on Human Factors in Computing Systems: Proceedings of the SIGCHI conference on Human Factors in computing systems* (pp.611-620) Montréal, Québec, Canada. http://doi.acm.org/10.1145/1124772.1124865
5. Ferre, X. Juristo, N. Windl, H. Constantine, L (2001) Usability basics for software developers. In H Erdogmus (Ed.) *Software, IEEE Volume: 18, Issue: 1* (pp. 22-29) Univ. Politecnica de Madrid, Spain. http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?isnumber=19528&arnumber=903160&count=19&index=3
6. Nielsen J (2005), Heuristics for User Interface Design, Retrieved 26th April 2008, from http://www.useit.com/papers/heuristic/heuristic_list.html
7. Nielsen J, Molich R (1990) Heuristic evaluation of user interfaces. In J C Chew (Ed.) *Conference on Human Factors in Computing Systems: Proceedings of the SIGCHI conference on Human factors in computing systems: Empowering people* (pp. 249 - 256) Seattle, Washington, United States. http://doi.acm.org/10.1145/97243.97281
8. Paymans, T F, Lindenberg, J and Neerincx, M (2004). Usability trade-offs for adaptive user interfaces: ease of use and learnability. In J Vanderdonckt (Ed.) *International Conference on Intelligent User Interfaces: Proceedings of the 9th international conference on Intelligent user interfaces* (pp. 301-303). Funchal, Madeira, Portugal. http://doi.acm.org/10.1145/964442.964512
9. Gajos K. Z, Long, J J and Weld, D S (2006). Automatically generating custom user interfaces for users with physical disabilities. In S. Keates (Ed.) *ACM SIGACCESS Conference on Assistive Technologies: Proceedings of the 8th international ACM SIGACCESS conference on Computers and accessibility* (pp. 243-244). Portland, Oregon, USA: ACM SIGACCESS. http://doi.acm.org/10.1145/1168987.1169036
10. Sousa K, Furtado E (2005) From usability tasks to usable user interfaces. In M. Sikorski (Ed.) *ACM International Conference Proceeding Series; Vol. 127: Proceedings of the 4th international workshop on Task models and diagrams* (pp. 103 - 110), Gdansk, Poland. http://doi.acm.org/10.1145/1122935.1122956