

UCC Library and UCC researchers have made this item openly available. Please [let us know](#) how this has helped you. Thanks!

Title	UIXSim: A User Interface Experience Analysis Framework
Author(s)	Hashemi, Mohammad; Herbert, John
Publication date	2014-01
Original citation	Hashemi, M. and Herbert, J. (2014) 'UIXSim: A User Interface Experience Analysis Framework', 2014 5th International Conference on Intelligent Systems, Modelling and Simulation, Langkawi, Malaysia, 27-29 January. doi: 10.1109/ISMS.2014.13
Type of publication	Conference item
Link to publisher's version	https://ieeexplore.ieee.org/abstract/document/7280873 http://dx.doi.org/10.1109/ISMS.2014.13 Access to the full text of the published version may require a subscription.
Rights	© 2014, IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.
Item downloaded from	http://hdl.handle.net/10468/8732

Downloaded on 2021-05-12T02:03:48Z

UIXSim: A User Interface Experience Analysis Framework

Mohammad Hashemi, John Herbert
Computer Science Department
University College Cork
Cork, Ireland
m.hashemi@cs.ucc.ie, j.herbert@cs.ucc.ie

Abstract—Quantifying and measuring Quality of Experience (QoE) are important and difficult concerns in Human Computer Interaction (HCI). Quality of Service (QoS) and the actual User Interface (UI) of the application are both important contributors to the QoE of a user. This paper describes a framework that measures accurately the way a user uses the UI in order to model users’ behaviors and profiles. It monitors the use of the mouse and use of UI elements with accurate time measurement. It does this in real-time and does so unobtrusively and efficiently allowing the user to work as normal with the application. This real-time accurate measurement of the user’s interaction provides an important element of the analysis of the user’s QoE.

Keywords—User Modeling; User Interface Experience; Quality of Experience; User Experience; Human and Computer Interaction;

I. INTRODUCTION

User Experience (UX) is a set of behaviors, emotions, and attitudes of using or interacting with a product. Human and Computer Interaction (HCI) can be measured from meaningful, effective, and valuable aspects, which built a part of UX. UX is changing over time for some reasons such as products or environment contexts. UX is subjective, context-dependent, and varies dynamically over time[1], [2]. Studying UX may be possible in a lab experiment over a short time; however the full UX needs to be studied in real environments with real users over longer periods.

QoE is introduced to assess user experience when interacting with a product[3]. QoE represents all users expectation and quality needs. User’s aesthetic and hedonistic needs are important to be considered as a part of product quality assessment.

User behavior is affected by external and internal factors. Internal factors depend on the cognitive, biological, and physiological aspects of the user. External factors are based on context around the user such as social environment[3]. This study concentrates on internal factors along with recorded information of user’s activity.

II. RELATED WORK

”The extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency, and satisfaction in a specified context of use”, is the definition of usability in the Part 11 - ISO 9241 [4].

ISO 9241-11 specifies that user satisfaction is a part of the usability concept. Moreover, [5] believes that assuring effectiveness and efficiency alone guarantee user satisfaction. Instruments that measure user satisfaction tend to measure the efficiency and effectiveness of the interaction too [6]. Monitoring users’ interaction leads to measuring user satisfaction while the efficiency and effectiveness interaction are the metrics.

Since the 90s, there have been many new concepts of QoE metrics and measures like Hedonic-Quality by [7], The Four Pleasure by [8], or Emotional Usability by [9]. With many concepts proposed to evaluate the UX and develop QoE, having a framework to precisely measure the UX metrics, is still needed for UIX development. Because of the complexity of evaluating user interface experiences (UIX), a QoE scheme for UIX has not yet been proposed [10].

Morae is software designed by the TechSmith company to record some useful data from users interaction with a software, analyze data and provide some useful analysis [11]. It is basically designed to test software by collecting some data during an experiment session with some participants. The video or audio recorded during the sessions needs to be analyzed by human. Morae needs some information about a set of functionalities of software that is going to be tested. This information is given by observer or software company during a session. Morae also have no information about the UI elements as our framework provides. All these differences make our work distinguished from Morae. Our framework tends to replace analyzing algorithms instead of having human analyze users activities. Moreover, Morae can be useful for a very small number of experiments, due to have a limit human resources and to be a costly process.

A. User Profiling

User profiling is a user model that represents a user’s preferences, habits, and interests. It can be used to predict how a user might use a UI. Building a user profile can be done explicitly or implicitly.

Explicit information comes from the data a user inputs via forms or collected from some provided UI to choose options as interests. Normally, users do not like to fill up a form or choose options in terms of creating a profile model even when they know the benefits[12]. Basically, this information

contains user’s gender, birth info, hobbies, and any other demographic personal information.

Previously, UI developers used feedbacks to improve a user’s satisfaction. They might randomly choose a user and ask to fill up a survey or take an interview. This is annoying for a user. On the other hand, limit questions and limit time in an interview would not cover everything about user experience. Knowing the real user’s satisfaction is a difficult process.

Moreover, the user’s answer might be inaccurate due to have a misunderstanding of a question. It also has a crucial process to interpret an answer by a reviewer. This step can be done implicitly by simply asking a user to input a social network account and retrieve required information from user’s account, data mining, or even obtain the information from an e-commerce system out of customers data as [13] proposed it.

Another version of building a user profile is to collect some data implicitly that is called implicit information. It is built by observing user’s activities and extract user’s interests, hobbies, and preferences unobtrusively. Gathering implicit information is another main aim of this study.

Having a model of user behavior will be helpful to design and improve an interactive system. Understanding a user helps a system to anticipate a user’s behavior in different situations and accordingly adjust itself to fulfill user’s needs. To achieve the best overall performance, a system needs to determine a user’s situation and predict what quality he/she needs and then try to configure itself with the best match of configurations that the user mostly are happy with[14].

III. UIXSIM FRAMEWORK

This study introduces a framework to transparently and unobtrusively capture a user and implicitly gather information about his/her activities, UI actions, and emotions during UI interactions. ”It is possible for a multimedia service provider to provides contextualized QoE by taking care of user’s contextual information. However, it raises some privacy and security issues that also need to be considered[3].” A user’s activity can be an interactive action with an interactive UI or an interaction between a user and the operating systems (OS). The provided information can be used to build a user model or a profile model for each user. The model can contain interactive information about each interactive software, time and duration of doing a task, and the emotion of a person during doing a single action or a task.

One of the most important differences of this framework to the existing works is that this framework is also able to retrieve some information about the UI elements of the other software, which a user interact with. For instance, once a user open a music player and click the play button of the music player UI, the framework detects the information about the play button such as the size, location, the title of

the button and the default operation the button is responsible for; here the default operation for the play button is to play the selected song. The previous works only record the mouse movements and location of X and Y.

The result and data provided by this framework is useful for UIX developers to find out which part of the UI is used more or to track the activities of all users and compare them to find out the similarities and differences. The data provided here can also be used for measuring QoE; number of clicks, the distance mouse moved, scrolling the mouse wheel, the time an element of a UI clicked, the duration between two actions, and the overall duration of an action, are examples of the output of the proposed framework. Having a comparison of two or more users will provide a UIX developer the idea of where they have weaknesses, where the users have more similar behavior, and where the users have difficulties to interact with a UI.

A. Framework Architecture

UIXSIm consist of two major modules, which are Activity Detector Engine (ADE) and Activity Analysis Engine (AAE).

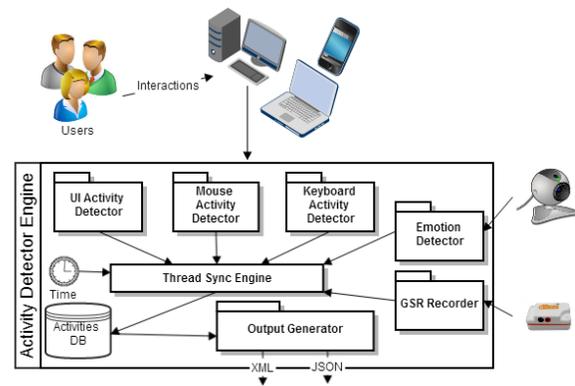


Figure 1: UIXSIm: Activity Detector Engine (ADE)

1) *Activity Detector Engine (ADE)*: Activity Detector Engine (ADE) includes number of individual components which is called detectors. UI Activity, Mouse, Keyboard detectors build the heart of the framework (see Figure 1). The latest version of the framework utilized with two other components that are Emotion Detector and GSR Recorder. Emotion detector requires to have a camera, and GSR recorder needs a Galvanic Skin Response (GSR) kit equipped with blue-tooth to send the GSR data over it to the framework. GSR or galvanic skin response is a measurement method to measure electrical conductance level of the skin which is normally used to determine emotional states of a user or user’s stress level. ADE is implicitly monitor UI activities and transparently working in the background and storing the activities into its local database. In case of using Emotion Detector, it basically take a picture every few

second (three seconds by default) and extract seven emotions (Happiness, Neutral, Sadness, Angry, Surprise, Disgust, and Fear) from the detected face of the user. Moreover, in case of using GSR Recorder, the GSR data recorded at the same time of any activities. Having all this information is very useful for analyzing QoE of a user working with a specific software’s UI. Knowing what action, interaction, or task makes a user unhappy or stressful would be useful for adjusting systems to prevent it not to happen again or change it in a different way.

2) *Activity Analysis Engine (AAE)*: Activity Analysis Engine (AAE) is a part of framework with the ability of analyzing the stored information in local database and generates some reports based on the activities detected by ADE (see Figure 2). Input Analyzer checks the format of input files and validates it, then converts and stores it to the local database. Analysis Engine is responsible to do analyzing on the stored data and provide some overview information in terms of activity, time, frequency, and descriptive point of view. Report Generator is exporting a known type of the result such as XML or JSON.

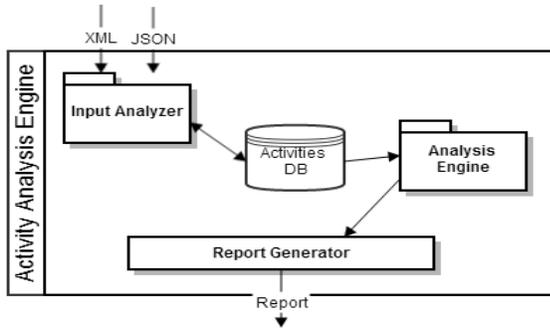


Figure 2: UIXSim: Activity Detector Engine (ADE)

3) *UIXSim Data Structure (UDS)*: The data stored during human (user) and computer interaction. UDS contains some information about UI elements such as element’s Name, Role, Value, Width, Height, Location-X, Location-Y, and Default Action. this information will be attached with some information about the UI owner, service, or software name and time and date, emotions object in case of using emotion detector, GSR object in case of using GSR recorder, and then will be exported to a single XML or JSON file. The exported file later will be used to analyze user’s activity. The desirable results will be carried out that is shown in the following section.

- Elements: Element’s Name, Role, Value, Width, Height, Location-X, and Location-Y, Default Action.
- Mouse: Mouse’s position X, position Y, scrolling, and events.
- Process: Process ID, Process Name, Window Name, and Window Handle ID.

- Emotions: Happiness, Neutral, Sadness, Angry, Surprise, Disgust, and Fear.
- GSR: GRS Raw, GSR Resistance.

B. Result Samples

There are two experiment has been done by five users to show some samples of the framework result. In the first experiment, three users spend 15 to 20 minutes working with Microsoft Excel 2010, doing the same task given to them at the start of experiment. Duration of the task is depend on the ability of the persons doing a task such as understanding the task and finding right element in the UI. It is also depend on how familiar they are with the Microsoft Excel UI. An example of the task given to the participants is to do 'Select Cell C1 and Click on the Bold Button'. In the first experiment, emotion and GSR are not included. In the second experiment, two users are asked to do the same task as the previous experiment; but the emotion and GSR data are included and only seven minutes of the task is analyzed.

Table I: OVERVIEW SUMMARY

Task Info	User 1	User 2	User 3
Starting Time	15:09:00	16:14:07	15:00:29
Ending Time	15:26:53	16:42:59	15:26:35
Duration (minutes)	17:09.5	28:52.3	26:06.4
Number of Records	24261	46903	33530
Mouse Movement (Pixels)	117553	417559	211279
Mouse Wheel	409	427	32
Mouse Moving	23429	45767	33015
Mouse Up Clicks	214	353	240
Mouse Down Clicks	209	356	243
Mouse Right Clicks	0	6	0

Table II: UI ACTIVITIES

Controls		Participants		
Name	Role	User 1	User 2	User 3
Excel Worksheet Cells	Client	123	168	124
Accounting Number Format	Drop down button	1	1	3
All Borders	Menu item	3	3	3
Paste	Split button	7	4	12
Insert Function	Dialog	2	14	5
Function Arguments	Dialog	3	18	7
Merge & Center	Split button	9	3	3
Middle Align	Push button	2	4	3
Shading	Drop down button	3	5	7

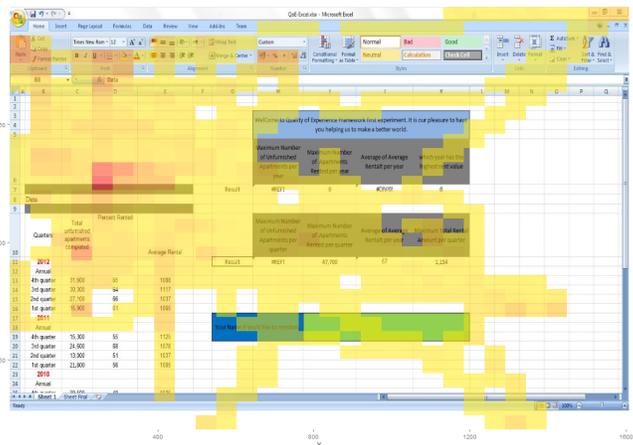


Figure 3: Mouse Activity (Movements and Clicks) User 1

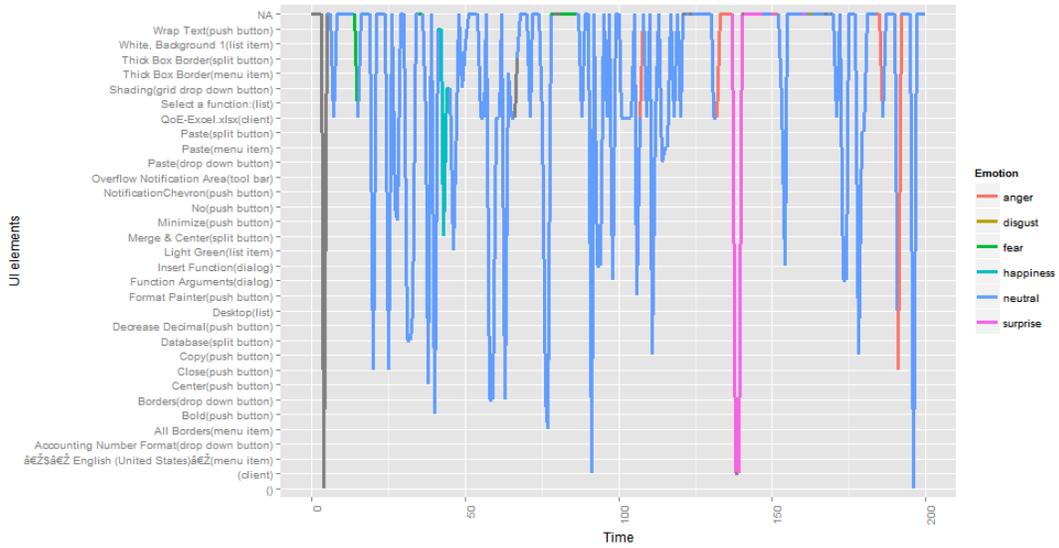


Figure 4: Series of UI Elements clicked by User 1 and his/her Emotions

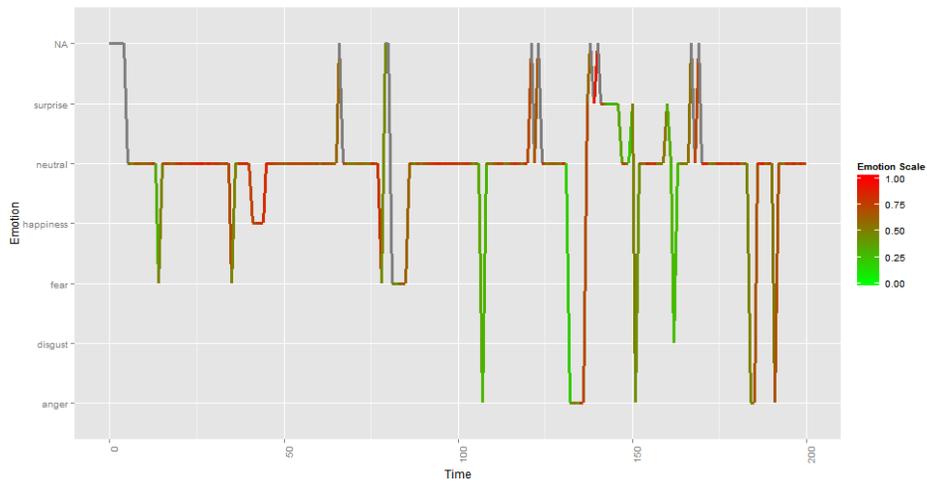


Figure 5: Emotions for User 1 During Experiment Two

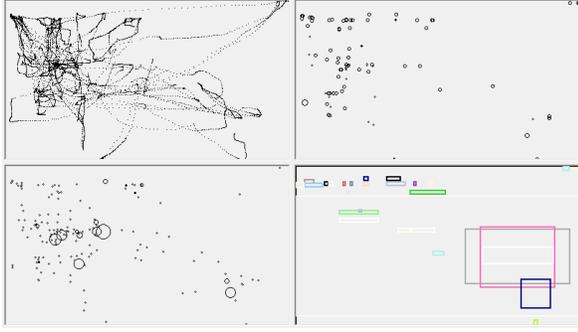


Figure 6: UI Activity (Mouse Movements, Clicks, Waiting times, UI Elements) User 1

1) *Experiment One:* Table I shows an overview information of participants doing the experiment one and Table II shows the frequency usage of UI elements for three participants of the first experiment. The provided information here is very useful for UI quality analysis; how long does a task takes, how much mouse movement and how many clicks needed for it, which part of UI is most viewed, or what UI element, how many times, is clicked. Figure 3 shows a heat-map of the first user's UI activities includes mouse movements and clicks in size of the target software (here is MS Excel 2010). In Figure 6, at the left hand side there are two figures; the top one shows the mouse movement and at the bottom it is showing the points where the mouse waits and stop moving and also the circles are showing the duration of waiting; at the right side, there are two more figures; the top one is showing the points user click such as right or left click of the mouse and the circles are showing the number of clicks in that area; at the bottom of that there is the scheme of UI elements that user interacted with.

2) *Experiment Two:* Figure 4 shows how user 1 used UI elements in order to finish the experiment. Knowing how a user is feeling when interactive with a UI is very useful; in case of having enough information about users working with the same UI, it will draw us an overview of its UI evaluating. User's emotion is detected from user's facial emotion; it is scaled [0, 1] for each of seven emotions(see Figure 5). This kind of result can be achieved for any other software's UI elements for a single or multiple tasks. The result of a long period of time and the average of user's emotion while working with a UI would help to distinguish the negative and positive aspects of UIs. If the Quality of Service (QoS) will be recorded at the same time, this result might help to find the technical weaknesses of a product too.

This paper has proposed a framework to recognize users' behaviors whilst interacting with a UI. UI is an important part of software to recognize the users' quality of experiences, which is telling how a user works with the software. The framework provides some interesting data and figures to evaluate a UI based on users' activities. Monitoring users'

activities of the UI is not just related to the quality of UI elements, it can be considered as the quality of the software itself and the performance it provides as well.

IV. FUTURE WORKS

The final version of this framework will be able to provide UI activities, emotions, GSR data, and all hardware performances (QoS) in real time. A pattern recognition modules needs to be designed for this work to analyze UI activities and find similarities and the same activities in order to classify them; this will also help to define smaller tasks within a larger task. In case of having information such as what exact task a user is doing and his/her emotion and stress will lead to have QoE for that specific task. Combination of these small tasks will make a larger task and finally QoE of a complete task. UI activity classifier and pattern recognition module should work individually without knowing task information (unknown tasks).

V. CONCLUSION

This paper has proposed a framework to recognize users' behaviors whilst interacting with a UI. UI is an important part of software to recognize the users' quality of experiences, which is telling how a user works with the software. The framework provides some interesting data and figures to evaluate a UI based on users' activities. Monitoring users' activities of the UI is not just related to the quality of UI elements, it can be considered as the quality of the software itself and the performance it provides as well.

The real-time user activity analyzing can also give the opportunity to adjust the service quality based on users' needs by predicting them. Also changing the UI elements based on what a user needs and interacts more; hiding those parts that a user does not tend to use might also help to have a better UI experience.

ACKNOWLEDGMENT

I'd like to thank my supervisor, Dr. John Herbert, my beautiful wife and my lovely family, who have always supported me. I thank you all and wish you the best.

REFERENCES

- [1] S. Kujala, M. Vogel, A. E. Pohlmeyer, and M. Obrist, "Lost in time: The meaning of temporal aspects in user experience," in *CHI '13 Extended Abstracts on Human Factors in Computing Systems*, ser. CHI EA '13. New York, NY, USA: ACM, 2013, pp. 559–564. [Online]. Available: <http://doi.acm.org/10.1145/2468356.2468455>
- [2] ITU-T, "Definition of quality of experience (qoe)," ITU-T, Tech. Rep., Jan 2007.
- [3] K. Laghari and K. Connelly, "Toward total quality of experience: A qoe model in a communication ecosystem," *Communications Magazine, IEEE*, vol. 50, no. 4, pp. 58–65, 2012.

- [4] *ISO 9241: Ergonomic requirements for office work with visual display terminals (VDTs) Part 11: Guidance on usability*, International Organization for Standardization Std., 1998.
- [5] M. Hassenzahl, A. Platz, M. Burmester, and K. Lehner, "Hedonic and ergonomic quality aspects determine a software's appeal," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '00. New York, NY, USA: ACM, 2000, pp. 201–208. [Online]. Available: <http://doi.acm.org/10.1145/332040.332432>
- [6] G. Lindgaard and C. Dudek, "What is this evasive beast we call user satisfaction?" *Interacting with Computers*, vol. 15, no. 3, pp. 429–452, 2003.
- [7] H. M., "The effect of perceived hedonic quality on product appealingness," *International Journal of Human-Computer Interaction*, vol. 13, no. 4, pp. 481–499, 2001.
- [8] P. Jordan, *Designing Pleasurable Products*. taylor and francis, 2002.
- [9] R. J. Logan, "Behavioral and emotional usability: Thomson consumer electronics," *Usability in practice*, Michael E. Wiklund (Ed.), vol. 1, pp. 59–82, 1994.
- [10] Z. Jie and H.-J. Kim, "User feedback oriented quality of experience management framework," *Communications, China*, vol. 10, no. 1, pp. 72–80, 2013.
- [11] TechSmith. (2013, November) Morae. TechSmith Co. [Online]. Available: <http://www.techsmith.com/morae.html>
- [12] S. Schiaffino and A. Amandi, "Artificial intelligence," in *Artificial Intelligence*, M. Bramer, Ed. Berlin, Heidelberg: Springer-Verlag, 2009, ch. Intelligent User Profiling, pp. 193–216. [Online]. Available: <http://dl.acm.org/citation.cfm?id=1793943.1793954>
- [13] G. Adomavicius and A. Tuzhilin, "Using data mining methods to build customer profiles," *Computer*, vol. 34, no. 2, pp. 74–82, 2001.
- [14] A. Pentland and A. Lin, "Modeling and prediction of human behavior," *Neural Computation*, vol. 11, pp. 229–242, 1995.