Baran: An Interaction-centred User Monitoring Framework

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

Keywords: User Modelling, User Profiling, Quality of Experience, User Experience, Human and Computer Interaction.

Abstract: User Quality of Experience (QoE) is a subjective entity and difficult to measure. One important aspect of it, User Experience (UX), corresponds to the sensory and emotional state of a user. For a user interacting through a User Interface (UI), precise information on how they are using the UI can contribute to understanding their UX, and thereby understanding their QoE. As well as a user's use of the UI such as clicking, scrolling, touching, or selecting, other real-time digital information about the user such as from smart phone sensors (e.g. accelerometer, light level) and physiological sensors (e.g. heart rate, ECG, EEG) could contribute to understanding UX. Baran is a framework that is designed to capture, record, manage and analyse the User Digital Imprint (UDI) which, is the data structure containing all user context information. Baran simplifies the process of collecting experimental information in Human and Computer Interaction (HCI) studies, by recording comprehensive real-time data for any UI experiment, and making the data available as a standard UDI data structure. This paper presents an overview of the Baran framework, and provides an example of its use to record user interaction and perform some basic analysis of the interaction.

1 INTRODUCTION

Interaction is very important and as the number of interactive devices grows, providing a good User Experience (UX) is a concern for device and service producers. Understanding UX is difficult, and it is even more difficult when UX is for interactive products and systems (Forlizzi and Battarbee, 2004). There are several approaches to understanding UX. One is the perspective of users that is called user-centred, or focus on products that is called product-centred. Another is to understand UX through the interaction between user and products named interaction-centred (Forlizzi and Battarbee, 2004). Many researchers believe interaction-centred is the most valuable for understanding how a user experiences a product (Forlizzi and Battarbee, 2004). In this study we focus on interaction-centred UX.

User Quality of Experience (QoE) is a subjective entity and difficult to measure. One important aspect of it, User Experience (UX), corresponds to the sensory and emotional state of a user. For a user interacting through a User Interface (UI), precise information on how they are using the UI can contribute to understanding their UX, and thereby understanding their QoE.

As well as a user's use of the UI, other real-time digital information about the user could include the readings from the various sensors on a device such as a smart phone (e.g. accelerometer, gyroscope, magnetometer, light level) and readings from physiological sensors as might be available from a health monitoring wireless BAN (body area network) (e.g. heart rate, GSR, ECG, EEG, EMG). All this real-time digital data together is called the User Digital Imprint (UDI), and it provides all the available external digital information that can reflect the subjective UX when interacting through a UI.

As the popularity of the mobile devices are increasing, users experience different situations while working with their mobile devices. User's context and moods, and environments are dynamic and always are changing. Applications, services, and all computing trends need to know the changes and adapt to them in order to improve user experience. Context is any information characterizing human and computer interaction and the surrounding environment. This work addresses a gap between applications or services and the context. Each service or application needs to implement a component to extract some information in order to know the context. First of all, this is a time consuming process and needs special knowledge about context information. Secondly, processing, analysing, and extracting some useful information out of the raw data about the context is another difficult process. Service producers and application developers should not also have to implement the context component part. Baran is an interaction-centred, user monitoring framework that meets this need. It collects context information such as interaction, sensor, and physiological data. In future work, the collected raw data will be analysed and higher level information will be provided. It currently supports the Windows Operating System (OS) for desktops and the Android OS for smart phones and tablets. Mac OS and iOS versions are under development. The Windows OS version of the framework was presented and published in (Hashemi and Herbert, 2014).

2 BACKGROUND

There are other systems to collect and manage context information for mobile devices include middle-ware, services, and frameworks. CONTextfactORY (Contory), is a middle ware prototype proposed by Riva (Riva, 2006). It collects some internal and external context information and offers a SQL-like interface using a query language to allow third parties to make a query about context with some specifications, specially in an ad-hoc network manner. This work suffers from resource constrained of smart phones and tablets. Storing the context data needs huge storage space and also the CPU, RAM, and battery usage for processing. We address this part by sending the context data to our cloud server in order to store and process. We use a well-designed data structure, UDI, for the data sent to our cloud server. It is explained in section 4.1.

Hermes, is another context-aware application development framework (Buthpitiya et al., 2012). It has local and cloud service in order to provide context. It defines widgets that are responsible for sensors. Widgets can communicate to each other in order to transmit context data. While they address the weaknesses of some other work, they did support other developers to use their framework. They also did not provide samples of their work. Their framework also lacks the functionality of connecting to BAN sensors.

Some other context systems exist, such as the Context Toolkit (Dey et al., 2001) that is a library, RCSM (Yau and Karim, 2004) that is middle ware, and the TEA framework (Schmidt et al., 1999). They provide application developers with uniform context abstractions but mostly without the analysing or processing data.

In this study we propose the Baran framework. We implement it in Java for Android and in C#.Net for Windows OS. Baran is able to collect the context information and combine it with BAN sensor data in order to provide the opportunity to estimate UX. In future we will work on processing and analysing data to provide the higher level context information.

The term, User Experience (UX), was introduced in the 1990s (Forlizzi and Battarbee, 2004) and refers to a user's perceived experience of a service or a product. UX includes all aspects of behaviour, emotions, and attitudes. Cognitive, emotional, aesthetic, physical, and sensual experiences contribute to a user's experience. UX is defined as : *a person's perceptions and responses that result from the use and/or anticipated use of a product, system or service* (ISO 9241-110:2010) (9241-210:2010., 2010). UX is subjective, dynamic, difficult to measure, and also depends on an individual's perceived experience and context. The context could be either external, about the environment of a user, or his/her internal states, such as motivation, needs, or mood.

There are varies methods for UX evaluation and measurement. Questionnaires, interviews, and surveys are used in HCI studies (Vermeeren et al., 2010).

A questionnaire contains a set of questions about the product, its usability, UX metrics, and the user's internal states. One of the most popular and reliable proposed methods in this area is NASA-TLX (Cao et al., 2009). Mental, physical, and temporal demand, performance, effort, and frustration level are all measured in this method. It relies on the workload of a task and the other measurements mentioned before. It has been used in variety of studies (Hart, 2006). However, there is a need for entry of information by users.

AttrakDiff is another popular questionnaire designed to measure hedonic stimulation, identity and pragmatic qualities of a product (Hassenzahl, 2005; Hassenzahl and Tractinsky, 2006). AttrakDiff contains 28 questions. The AttrakDiff questionnaire was used in a study of exploring the effects of hedonic and pragmatic aspects of goodness and beauty of different MP3 player skins (Hassenzahl, 2008). AttrakDiff is also used in another study (Schrepp et al., 2006) on the influence of hedonic quality on attractiveness. There are three different interfaces evaluated by 90 people, who received an invitation email in order to participate.

These methods require users to fill up questionnaires, attend to interview sessions, etc. Complicated, difficult, and confusing questions in an interview or a questionnaire can make it unpleasant for users. It is also not a good user's internal state indicator as determining emotions and moods are difficult.

In Human and Computer Interaction (HCI) studies, understanding a user is the main challenge. It

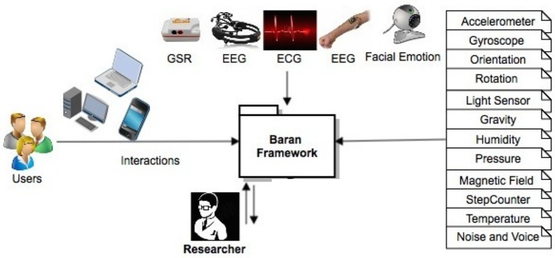

Figure 1: Baran Framework Overview.

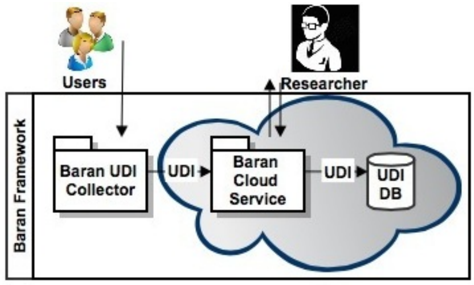

Figure 2: Baran Framework High Level Architecture.

requires knowing what has been done by a user before and after an interaction. So, there is a need to provide a more complete set of information than that from a set of general questions. On the other hand, the methods used to collect the information need to be unobtrusive and transparent to the user.

The Baran framework is designed to address the mentioned challenges and improve the UX measurement methods by providing a more complete version of the information about a user. It also is building a general solution to collect, analyse, and process context information, in order to be used by other researchers, developers, or service producers.

3 BARAN FRAMEWORK

The proposed framework, Baran, transparently, and unobtrusively collects environmental, behavioural, and physio-psychological context information about a user. Figure 1 is an overview of the Baran framework showing the connectivity of internal and external sensors with its engine. The framework can be used as a library in a software product, or installed separately on a computer or a smart phone. The framework provides the ability of connecting to sensors such as EEG, GSR, EMG, or ECG. Thanks to commercial versions of these sensors, there are plenty of them available. They are small in size and can transfer data over Wifi and Bluetooth. When a sensor is configured, the framework synchronizes the data from it with the context information, explained in the next section, and stores them in a well-designed data structure called the User Digital Imprint (UDI). A collection of UDI information makes a profile for a single user. In Baran, a cloud service receives all UDI from the experimental devices. The Baran cloud service stores all UDIs, and later provides them for analysis. The analysis methods can be chosen by the researcher. Baran will also provide analysis methods in the future. As the UDI contains low level information, there is a need to process it to extract some higher level information such as user emotional or movement state. Having all information in one place will provide an opportunity to share experimental information anonymously with other researcher. As the data is in the cloud, third parties do not need to have any connection to the user devices in order to receive context information. They can communicate directly with the Baran cloud server and after analysing the context, can accordingly change their applications or services.

Figure 1 shows interactions through devices, computers, smart phones, or tablets, with services or products, specially through user interfaces (UI). Baran gathers available data and put them together to build a UDI. The UDIs are sent to the Baran cloud service or stored locally.

4 BARAN ARCHITECTURE

The Baran framework contains two main components as it is shown in Figure 2. One is a service working transparently as a background process on the device regardless of being a smart phone, a tablet, or a computer machine. It collects context information and builds the UDI, then periodically sends it to the Baran cloud service. The data structure needs to be light enough due to have a limited network resources. It dynamically changes in size, depends on what device it was built on and what information was available at the time of collecting data. It encourages us to use Java Script Object Notation (JSON) format, because it is a well-designed data structure for dynamic data. Baran framework builds a multi purpose data set of this useful information. Baran provides a general solution giving an opportunity to compare a result of a study to another.

4.1 User Digital Imprint (UDI)

A User Digital Imprint (UDI) is the digital imprint of a user that contains a combination of the context data, UI activities, and the sensor data. UDI, is a data structure used by the Baran framework, can contains UID

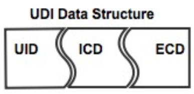

Figure 3: UDI Data Structure.

UID: User Interface Interaction Data
CD: Internal Context Data
ECD: External Context Data

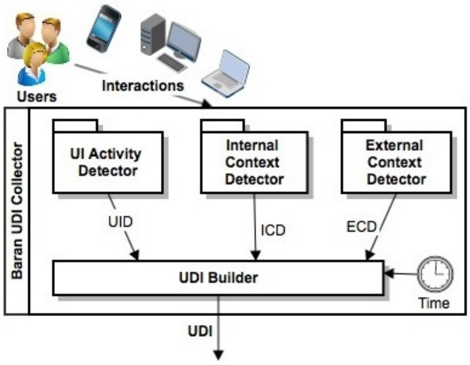

Figure 4: Baran UDI Collector Engine.

(User Interface Interaction Data), ICD (Internal Context Data), and ECD (External Context Data). UDI varies in size depending on the device and the configuration of the framework used in an experiment. Along with UID, ICD, and ECD data, the information about the device such as the OS, the Model, and the ID will be added to the UDI. Figure 3 is an overview of a sample UDI.

4.2 Baran UDI Collector

Figure 4 is the internal architecture of the Baran UDI collector. It works as a background process, transparent to the user. It builds a UDI per second as well as building a UDI for every single UI interaction such as touching, scrolling, and sliding in the touch screen devices, and clicking, and typing in the other devices. The Baran UDI Collector contains four modules, UI activity, the Internal, the external Context modules, and a module to build the synchronized version of the UDI. The UDI contains different information depends on the device and the framework configuration. For example, desktop computers normally have no accelerometer or gyroscope which could be added to the Baran Framework as external sensors. The UI activity detector module is responsible for detecting the UI interactions such as clicking and touching. It extracts important information about the detected interaction such as the name of application, the name and the type of the user interface element that was interacted, also the event identifiers such as clicking, scrolling, touching, double-touching. Figure 5 shows an example of a part of the UDI describing the UI interaction data (UID).


```
"uiActivity":{  "className":"Button",
                "content":"",
                "eventType":"VIEW_CLICKED",
                "time":"Jun 14, 2014 12:09:31 AM",
                "text":"[No thanks]",
                "packageName":"chrome",
                "isFullScreen":false,
                "isEnabled":true,
                "isChecked":false
            }
```


Figure 5: User Interaction Data (UID).

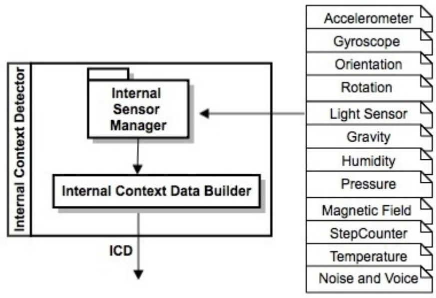

Figure 6: Internal Context Detector.

4.2.1 Internal Context Detector

The Internal Context Detector (Figure 6) is a module of the Baran UDI collector. It is responsible to manage the internal sensors of the device. Normally, it is useful for smart devices such as smart phones and tablets, as they have internal sensors such as accelerometer, gyroscope. Figure 6 lists some available sensors in some of the smart devices. This module extracts the raw data from the mentioned sensors and builds a data structure, called the Internal Context Data (ICD), to be passed to the UDI Builder. The Baran framework provides a set of processing and analysis algorithms to extract high level information out of the low level information. A user movement state (eg. Walking, Running, Still) or emotional state (eg. Happy, Disgusted, Tired) are examples of high level information (Hashemi and Herbert, 2014).

4.2.2 External Context Detector

The External Context Detector is a module similar to the Internal Context Detector. Figure 7 shows the components and the sensors this module is connected to. This part will able the Baran framework to connect to external sensors that are not available in the smart devices. They could be sensors to monitor a user's physiological and psychological states. Combining this data and the information from other components of the Baran UDI Collector provides us the opportunity of better understanding users and their activities.

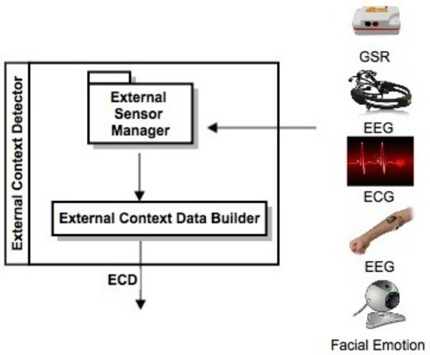

Figure 7: External Context Detector.

Some of the important sensors that our framework are capable with are listed and explained.

- Galvanic Skin Response (GSR): It is also recognized as Electro Dermal Response (EDR), Psycho-Galvanic Reflex (PGR), Skin Conductance Response (SCR) or Skin Conductance Level (SCL), and is as a measurement of the electrical conductance of the skin and shows the level of the stress and the emotional state of a user.
- ElectroCardioGraphy (ECG): It measures electrical impulses from the heart and converts them to a wave form.
- ElectroEncephaloGraphy (EEG): It is a measurement method of brain electrical activity.
- ElectroMyoGraphy (EMG): It detects the electrical potential generated by muscles.
- Facial Expression It reflects the emotional state of a user. Capturing images of the face of a user helps to detect their emotional state and facial expression.

This module builds a data structure that is called the External Context Data (ECD) after collecting all required data from the external sensors and devices. The ECD will be passed to the UDI Builder module.

5 DATA COLLECTION AND ANALYSIS

We use the Baran framework to collect some UDI data to demonstrate some basic analysis and data variation. After the data collection process, it was transferred to the database of the framework, analysed, and processed for the demonstration. Five user devices are chosen for this experiment. Baran was installed on the device as a service and users had worked normally with their devices with no distraction. Baran captured the data, built UDIs, and stored them locally on the memory of the device as well as sending UDIs to the Baran Cloud Service. It is vital to mention that the experiment is to show a sample of interaction data and experiments with more sensors, EEG, GSR, etc., are under way.

Table 1: Devices Information and Experiment Duration.

Device	Android OS Version	Brand	Model	Type
Dev1	4.4.2	Asus	Nexus 7	Tablet
Dev2	4.3.0	Samsung	GT-I9300	S-Phone
Dev3	2.3.5	HTC	Explorer	S-Phone
Dev4	4.1.2	Samsung	GT-I8190N	S-Phone
Dev5	4.2.2	Sony	CS2305	S-Phone

Table 2: User Interaction Usage.

Devices	Day : Hour : Minute : Second
Dev1	01:03:04:29
Dev2	01:02:05:15
Dev3	00:00:06:19
Dev4	00:02:08:51
Dev5	02:01:13:30

Table 3: User Interface Interactions (Every Second Data not included in Total Per Device.

Devices	Touch/Click	Scroll	View Changed	View Selected	Every Second	Total Per Device	Average Interaction Per Day
Dev1	437	890	18	16	76355	1389	138
Dev2	197	574	30	50	-	893	223
Dev3	45	-	3	18	-	404	202
Dev4	194	170	413	16	8249	850	170
Dev5	667	7367	1957	124	120337	10452	1045

Table 1 provides information about the devices used in this experiment. They are the Android smart phones or tablets. Users are researchers, age ranged 30 to 35 years old. Table 2 shows the total time users interact with the devices and Baran collects UDIs. Table 3 shows a summary of users interaction such as touching and scrolling that are the basic interactive gestures. Other events are defined as View Changed, that is the event when the user interface elements changed, View Focused, that is trigged when a user interface is activated and comes to the front, and Ev-

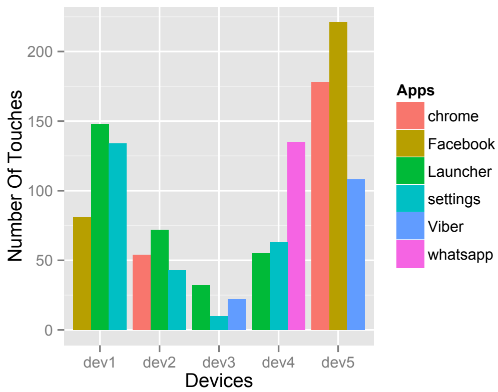

Figure 8: Touches/Clicks Summary for the Top Three Applications Per User.

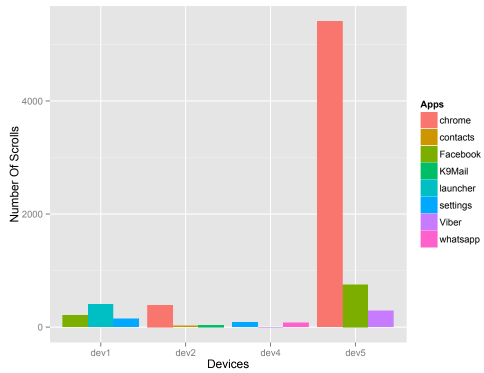

Figure 9: Scrolling Summary for the Top Three Applications Per User.

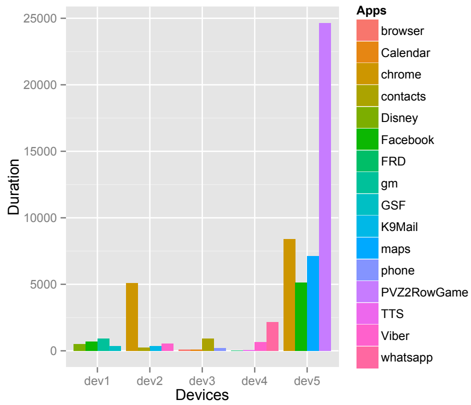

Figure 10: Time Usage Summary for the Top Four Applications Per User.

ery Second, that is an event designed by the Baran framework to capture the sensor data every second in order to have a better overview of the contextual information. User five's data shows that he/she uses his/her smart phone more than others. Although the duration of the data collection process for him/her was more than others, however the average interaction per day metric also shows a considerable difference.

Figure 8 reports the number of interaction in terms of touching or clicking the user interfaces of different applications. Launcher is the name of the operating system UI manager also called "Home". It is obvious the it is used mainly by users one to four. Device setting also is used many times bye some users that shows it is an important part od a smart phone. Facebook is used by user one and five as it is in top three most touched applications. User five mainly used Viber, Facebook and Chrome. Chrome, which is a web browser application, is also used by user two and five as the second most touchable application. Viber is an internet messaging and calling application, that is used bye some of the users. It is used by user three and five in most touchable applications. It shows this application is used mainly for messaging because the touching occurs when a user open an application or typing.

Figure 9 shows the number of scrolling for the top three applications used by users. The device three does not support scrolling events and is not reported in this figure. This figure shows that user five mainly used chrome with scrolling gesture. Facebook is the second most scrollable application that user five used. Chrome and Facebook are also used by other users as the top scrollable applications which shows scrolling within the user interfaces of these two applications is important and needs to be considered as a key point in UI designing. Comparing Figure 8 and 9 would help designers to distinguish between touchability and scrollability of their applications in order to improve UX and UI functionality.

Figure 10 shows how long users spend on using the applications. It reports top four applications for each user. There is a game played by user five that is the main application he/she used. Moreover, there are some applications that are used in common between users such as Chrome, Facebook, Contacts, Map application.

Figure 11 shows the difference between number of touching and scrolling events for top ten applications used by users in total. It is clear that scrolling is the most popular gesture in the Chrome, the Facebook, and also some other applications. On the other hand, touching is mostly used in the Maps and the whatsApp application. We also compare these top ten applications in terms of time duration users spend in an application.

Figure 12 reports that Chrome, Contacts, and Home are the top three applications that are used by users in total. There is an interesting observation that although Facebook application is not used for a considerable amount of time, however the number of in-

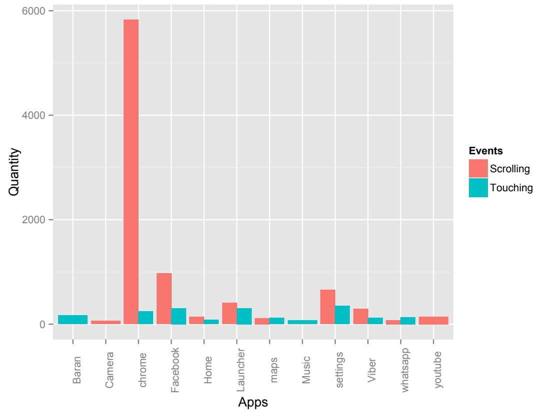

Figure 11: UI Interactions Summary for the Top Ten Applications for All User.

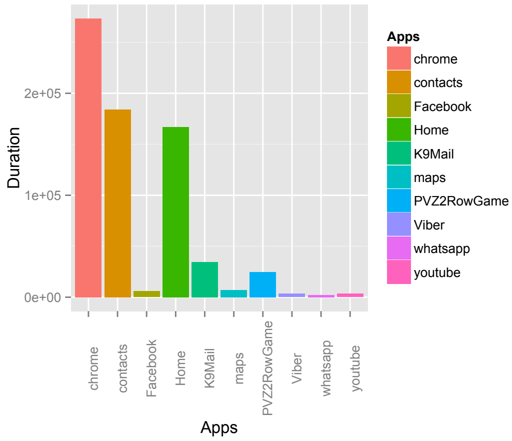

Figure 12: Time Usage Summary for the Top Ten Applications for All User.

teractions in it, is always high. This data can be used in order to model users and track their activities to study abnormal behaviour.

In the smart-phone and tablet OS, each entity interacted with has a title and possibly a content as a description. Table 4 lists the titles of these entities that are interacted with during the experiment. For instance, user one in the device one, touches an entity titled ”Home” more than 100 times. The entity could be an application named ”Home” or any touch that was done in the home screen of the device. Analysing this data needs a more sophisticated methods to extract useful information. Table 5 reports information from the accelerometer, gyroscope, orientation, and rotation sensors. It shows the average and standard deviation of the sensor readings per device. They need to be processed in a more sophisticated way to extract better and useful information. For example, they could be analysed to find the applications that a user was using on the move. It can be understood

Table 4: The Titles of the Entities Interacted by Users.

Device 1		Device 2	
Content	Interactions	Content	Interactions
[Home]	109	[TouchWiz Home]	47
[Apps]	92	[Chrome]	27
[Subway Surf Game]	67	[K9 Email]	22
[Gmail]	33	[Apps]	16

Device 3		Device 4	
Content	Interactions	Content	Interactions
[HTC Sense]	35	[TouchWiz Home]	31
[Phone]	16	[unlock,no missed events.]	20
[All apps]	7	[WhatsApp]	20
[End Call]	5	[Apps]	8
[Calendar]	3	[Flipps]	3

Device 5	
Content	Interactions
[Facebook]	69
[PvZ 2]	53
[Maps]	49
[Chrome]	47
[Viber]	43

from the accelerometer sensor data that the devices two and three, were used more on the move.

6 CONCLUSION

Baran is a framework that simplifies the process of the data collection during an experiment. It is very valuable to have the contextual information combined with the User Interaction data. It provides a better understanding of user behaviour during an experiment. It also provides a better understanding of the cooperation between internal UI elements and can find where a bug occurs. The proposed framework gives the ability of collecting useful information. This paper demonstrates results of experiments that show the complexity of the collected data and the variety of the results obtained from different users. There is a huge space to add sophisticated analysis algorithms to the framework to extract more information. It is also very valuable to be able to combine the contextual information with various sensors, such as EEG or ECG. The Baran framework can be used by researchers without implementation effort. Researchers can simply configure the framework, then the needed data

Table 5: Average and Standard Deviation of Some Sensors.

Axis	X		Y		Z	
DeviceProfileID	Average	Standard Deviation	Average	Standard Deviation	Average	Standard Deviation
Accelerometer Sensor Data						
Dev1	-4.4900	0.5313	0.8105	2.1964	9.6616	1.1105
Dev2	-3.8800	1.3853	5.8777	2.5352	7.2268	1.9845
Dev3	-0.5919	2.0480	4.8383	3.226	6.7858	3.5863
Dev4	-0.2411	1.0037	2.5825	2.8138	9.1920	1.3464
Dev5	5.8751	3.8237	-1.1253	3.1923	-0.8628	5.0985
Gyroscope Sensor Data						
Dev1	-2.50	6.3103	2.80	8.2105	6.099	8.1606
Dev2	4.403	0.3170	-2.05	0.2812	1.32	0.2384
Dev3	0	0	0	0	0	0
Dev4	-9.104	0.184	-1.01	0.1126	-4.509	0.2666
Dev5	0	0	0	0	0	0
Orientation Sensor Data						
Dev1	178	130	-5	14	0	3
Dev2	195	144	-37	18	-2	8
Dev3	0	0	0	0	0	0
Dev4	201	87	-15	22	-1	7
Dev5	211	34	89	94	-40	24
Rotation Sensor Data						
Dev1	2.58	7.8603	-2.07	9.8902	-0.1908	0.4541
Dev2	0.2871	0.1388	-1.2	0.1617	-4.403	0.332
Dev3	0	0	0	0	0	0
Dev4	9.1704	0.1434	2.820	0.1361	0.2388	0.7033
Dev5	0	0	0	0	0	0

will be collected and provided to them for any further analysis. Health care system, habit discovery, or abnormal behaviour studies are some application areas the Baran framework and the UDI can be successfully used. Users also can use it to have a monitoring system and improve their UX by finding good and bad habits. Moreover, the data is valuable for researchers who study context based systems (Moldovan et al., 2013; Sourina et al., 2011). This framework also able researchers to share data for analysis in science community. UDI meets all this needs in order to contribute to the science community and help researchers and developers.

REFERENCES

9241-210:2010., I. D. (2010). Ergonomics of human system interaction - part 210. Technical report, International Standardization Organization (ISO). Switzerland.

Buthpitiya, S., Luqman, F., Griss, M., Xing, B., and Dey, A. K. (2012). Hermes – A Context-Aware Application Development Framework and Toolkit for the Mobile Environment. In *Advanced Information Networking and Applications*.

Cao, A., Chintamani, K., Pandya, A., and Ellis, R. (2009). Nasa tlx: Software for assessing subjective mental workload. *Behavior Research Methods*, 41(1):113–117.

Dey, A., Abowd, G., and Salber, D. (2001). A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-Aware Applications. *Human-Computer Interaction*, 16(2):97–166.

Forlizzi, J. and Battarbee, K. (2004). Understanding experience in interactive systems. In *Proceedings of the 5th Conference on Designing Interactive Systems: Processes, Practices, Methods, and Techniques*, DIS '04, pages 261–268, New York, NY, USA. ACM.

Hart, S. G. (2006). Nasa-task load index (nasa-tlx); 20

years later. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 50(9):904–908.

Hashemi, M. and Herbert, J. (2014). Uixsim: A user interface experience analysis framework. In *2014 Fifth International Conference on Intelligent Systems, Modeling and Simulation*, 2014 Fifth International Conference on Intelligent Systems, Modelling, and Simulation, pages 29–34. IEEE.

Hassenzahl, M. (2005). The thing and i: Understanding the relationship between user and product. In Blythe, M., Overbeeke, K., Monk, A., and Wright, P., editors, *Funology*, volume 3 of *Human-Computer Interaction Series*, pages 31–42. Springer Netherlands.

Hassenzahl, M. (2008). The interplay of beauty, goodness, and usability in interactive products. *Hum.-Comput. Interact.*, 19(4):319–349.

Hassenzahl, M. and Tractinsky, N. (2006). User experience a research agenda. In *March-April 2006*, pages 91–97.

Moldovan, A.-N., Ghergulescu, I., Weibelzahl, S., and Muntean, C. (2013). User-centered eeg-based multimedia quality assessment. In *Broadband Multimedia Systems and Broadcasting (BMSB), 2013 IEEE International Symposium on*, pages 1–8.

Riva, O. (2006). Contory: A middleware for the provisioning of context information on smart phones. In *Proceedings of the ACM/IFIP/USENIX 2006 International Conference on Middleware*, Middleware '06, pages 219–239, New York, NY, USA. Springer-Verlag New York, Inc.

Schmidt, A., Aidoo, K. A., Takaluoma, A., Tuomela, U., Laerhoven, K. V., and de Velde, W. V. (1999). Advanced Interaction in Context. *HUC '99 Proceedings of the 1st international symposium on Handheld and Ubiquitous Computing*, pages 89–101.

Schrepp, M., Held, T., and Laugwitz, B. (2006). The influence of hedonic quality on the attractiveness of user interfaces of business management software. *Interact. Comput.*, 18(5):1055–1069.

Sourina, O., Liu, Y., Wang, Q., and Nguyen, M. K. (2011). Eeg-based personalized digital experience. In *Proceedings of the 6th International Conference on Universal Access in Human-computer Interaction: Users Diversity - Volume Part II*, UAHCI'11, pages 591–599, Berlin, Heidelberg. Springer-Verlag.

Vermeeren, A. P. O. S., Law, E. L.-C., Roto, V., Obrist, M., Hoonhout, J., and Väänänen-Vainio-Mattila, K. (2010). User experience evaluation methods: Current state and development needs. In *Proceedings of the 6th Nordic Conference on Human-Computer Interaction: Extending Boundaries*, NordiCHI '10, pages 521–530, New York, NY, USA. ACM.

Yau, S. S. and Karim, F. (2004). A context-sensitive middleware for dynamic integration of mobile devices with network infrastructures. *Journal of Parallel and Distributed Computing*, 64(2):301–317.