J Multimodal User Interfaces (2018) 12:1–16
https://doi.org/10.1007/s12193-018-0258-2

ORIGINAL PAPER

Model-based adaptive user interface based on context and user experience evaluation

Jamil Hussain[1] · Anees Ul Hassan[1] · Hafiz Syed Muhammad Bilal[1] · Rahman Ali[2] · Muhammad Afzal[3] · Shujaat Hussain[1] · Jaehun Bang[1] · Oresti Banos[4] · Sungyoung Lee[1]

Received: 11 December 2016 / Accepted: 15 January 2018 / Published online: 1 February 2018

Abstract Personalized services have greater impact on user experience to effect the level of user satisfaction. Many approaches provide personalized services in the form of an adaptive user interface. The focus of these approaches is limited to specific domains rather than a generalized approach applicable to every domain. In this paper, we proposed a domain and device-independent model-based adaptive user interfacing methodology. Unlike state-of-the-art approaches, the proposed methodology is dependent on the evaluation of user context and user experience (UX). The proposed methodology is implemented as an adaptive UI/UX authoring (A-UI/UX-A) tool; a system capable of adapting user interface based on the utilization of contextual factors, such as user disabilities, environmental factors (e.g. light level, noise level, and location) and device use, at runtime using the adaptation rules devised for rendering the adapted interface. To validate effectiveness of the proposed A-UI/UX-A tool and methodology, user-centric and statistical evaluation methods are used. The results show that the proposed methodology outperforms the existing approaches in adapting user interfaces by utilizing the users context and experience.

Keywords Human computer interaction · Personalized user interface · Adaptive user interface · User experience · Context-aware user interfaces · Model-based user interface

✉ Sungyoung Lee
sylee@oslab.khu.ac.kr

Jamil Hussain
jamil@oslab.khu.ac.kr

[1] Department of Computer Science & Engineering, Kyung Hee University (Global Campus), 1732 Deokyoungdae-ro, Giheung-gu, Yongin-si, Gyeonggi-do 446-701, Republic of Korea

[2] Quaid-e-Azam College of Commerce, University of Peshawar, Peshawar, Pakistan

[3] College of Electronics and Information Engineering, Sejong University, Seoul, South Korea

[4] Telemedicine Group, Faculty of Electrical Engineering, Mathematics and Computer Science, University of Twente, Postbox 217, 7500 AE Enschede, The Netherlands

1 Introduction

The user interface (UI) is a dominant part of interactive systems that directly connected to end user to access the functionalities of a system. In most of the well-engineered applications, users use a small portion of the offered functionality and major part goes underutilized due to poor UI [2]. Furthermore, the UI element usage are differ among different users. UI designers face a number of challenges while designing a UI for interactive systems [7] due to the heterogeneity issue [33]. The heterogeneity can broadly be defined as a multiplicity of end users, computing platforms, input/output capabilities, interaction modalities, markup languages, toolkits, user working environments, and contextual variability. The multiplicity of end users is based on their diverse nature of bio-psycho-social characteristics. Similarly, end-users use different computing platforms (i.e., mobile, tablet, computer etc.), which have different input/output capabilities (i.e., mouse, keyboard, HUD, HMD, touch, sensory input, eye-gauze, etc.) using their different interaction modalities (i.e., graphics, speech, haptic, gesture, EEG, ECG etc.) [33].

One way to overcome these differences is adaptive UI called model-based user interface (MBUID) [2,33] as compare to the one-size-fits-all design such as universal design, inclusive design, and design for all [2]. The one-size-fits-all approach cannot handle the context variability that leads to bad user experience (UX). Additionally, building multiple UI for same functionality for handling the context variability is difficult which incur high cost and also not know all context at design time. A main goal of adaptive UIs is *plasticity* [17], the ability of UIs to preserve usability across various of context of use [14].

The context-of-use triplet consists of user, platform, and environment aspects that could support adaptive UI behavior [7,14]. The user aspects include user profiles, demographics, cognition, physical characteristics, sensory abilities, user activities and task. User cognition is all about the user attention, learning ability, concentration, and user perceptions. Physical characteristics are user mobility and abilities or disabilities that effect the user interaction with the system, such as hand or finger precision. User sensory information are user sight, hearing, and touch sensitivity that also have direct impact on user interaction with the system.

The platform aspects include both physical devices (e.g., desktop, laptop, tablet, phone etc.), software (e.g. operating systems, application platforms, etc.) [2] are essential for the efficient adaptation of a UI. For example, smart phone and tablet require different adaptations at the user interface. Additionally, the user preferred input modality is also required for the UI adaptation. The environment aspects include spatio-temporal attributes, tasks, and situation where the interaction take place such as light and noise level, and user location and timing (e.g. where the user is right now or where user was at a particular time).

Although context-of-use is mainly defined based on information about users, platforms, and environments, other dimensions such as application domain, adaptation type, multimodal data source and user feedback that could be related to describe context and to appropriately adapt an interactive system.

In state-of-the-art MBUID adaptive user interface design research [3,15,19,20,27,41], researchers have focused on the development of adaptation rules.These rules are either created with the help of UX experts or system designers that use their own knowledge in the assistive authoring tools [21,41] or by the automatic deduction process of mining relevant rules from the users interaction data with the system. The automatic deduction process is performed using various machine learning techniques and algorithms [28]. These methods considered different adaptation dimensions such as culture, user characteristics, user disabilities like sight, hearing, physical, and user cognition for design the adaptation rules [18,36]. For example, UI adaptation can auto change the color according to culture by considering the cultural meanings of color and color symbolism, increase the font-size for vision impairment users, simplify the UI for novice users, hide/show widgets, and swap the widgets according to the user usage behaviors. However, adaptive UI requires more concrete and practical framework, which cover different adaptation dimensions such as user capabilities, preferences, needs, and user context. The adaptation that covers a diverse set of aspect requires a huge amount of knowledge along with complex adaptation algorithms.

In this paper, we propose a model based adaptive UI methodology and implement an A-UI/UX-A tool that caters the adaptive UI based on the evaluation of user context and user experience. The main objective of the proposed solution is to deal with the personalized approach for building and managing the user interfaces by considering different adaptation dimensions such as context-of-use, multimodal data source, different adaptation aspect, and user in loop. We mainly deal with a user capability, preferences, needs, context-of-use, user interaction deep log, and user feedback for the generating an adaptive UI using adaptation rules created through A-UI/UX-A tool. This eventually leads to the evolution of information in the models and incorporation of personalized aspects in the user interface.

The rest of the paper is structured as follow. In Sect. 2, adaptive user interface related work is described. In Sect. 3, a brief overview of Mining Minds platform is briefly described. In Sect. 4, the proposed adaptive user Interface framework is abstractly described. In Sect. 5, overall proposed framework is presented from architectural perspective, knowledge creation for adaptive UI perspective and runtime UI rendering based on user experience and context perspective. In Sect. 6, implementation of the A-UI-UX-A tool, experiments, and user-based evaluation is presented. Section 7 discusses the significance, challenges and limitations of the A-UI/UX-A tool and Sect. 8 concludes the work.

2 Related work

Adaptive user interface design is a hot area of research since long. Numerous tools and reference architectures has been developed and proposed for creating the adaptive UI. This section briefly explores the proposed reference architectures, adaptation techniques and available tools along with their limitations.

For adaptive smart environment, a 3-layer architecture [31] was proposed that is based on the executable models for generation of an adaptive UI. However, the resulting model of the 3-layer architecture is unable to produce a granular level of adaptation due to generative runtime nature of the model. Furthermore, the proposed model ignores user feedback for improving quality of the UI in an incremental way.

CAMELEON-RT [7] is another reference architecture model for generating the migratable and plastic user interface. It provides the feature of adding adaptive behavior at runtime due to excellent conceptual depict of extensibility of adaptive behavior. However, they suggested the primary heuristics for the practical deployment of run time UI rendering.

TRIPLET: a computational framework for context-aware adaptation [38] consists of meta model, reference framework and adaptation aspects for adaptive UI. Based on the extensive systematic review of existing work, they proposed context-aware adaptation (CAA) framework that covered different aspects such as continuous update (e.g. adaptation technique), platform heterogeneity, and different scenarios consideration. However, it hard to apply in broad perspective.

Malai [12] provides the UI development environment based on model-driven approach. They considered actions, interactions, instruments, presentations, and user interfaces as first-class objects that helps to decompose the interactive system for improving the object reusability. However, runtime adaptation is not supported when the context changes.

Egoki system [20] provides adaptive UI services in ubiquitous environments to users that have physical, sensory, and cognitive disabilities. It use the model-driven approach for the generation of adaptive UI. However, they have some issue in models creation and presentation with final UI.

CEDAR [3] propose a model-driven approach for the adaptive user interface that can be easily integrated with a legacy system. They used a role-based UI simplification (RBUIS) method having a minimal feature-set and an optimal layout functionality to end users. The adaptive behavior of CEDAR system increased the usability. For the evaluation of CEDAR studio, they integration with open-source ERP system so called OFBiz.

In most of the above-mentioned proposed architectures have no support of user feedback. In addition, the integration with a legacy system is very difficult except CEDAR [3], its evaluation required to build new prototype.

Different adaptation techniques have been used, related to UI features, such as layout optimization, content, navigation, and modality [3]. Still there are gaps and limitations in existing adaptation techniques, such as they focus on design-time features minimization according to role rather than at runtime [3]. Most of them are theoretically based on UI features set selection. For example, different versions of UI are designed for different contexts. Several free and commercial software have used fixed role-based tailored UI, such as ERP and Moodle. Most of them used pre-identified UI feature set based on context at design time. However, they lack runtime feature selection methodology, which is essential for contextual changes. Similarly, the existing literature focus on the layout optimization, for example, SUPPLE [19], which automatically generate UI on the basis of user profile, preferences, tasks, and ability. It considers the user motor, vision ability, along with device use and task performed by the user for adapting the UI at runtime. It is very difficult to apply this method to large-scale application due to the human involvement at different levels at the design-stage. Additionally, it supports adaptation at a different aspect i.e. user with disabilities, which cannot be extended due to the specialized nature of adaptation algorithms.

MyUI [42] is another study that presents infrastructure for increasing the accessibility of information by providing adaptive UI. MyUI used multimodal design patterns for generating the adaptive UI according to the user preferences. Due to the multimodal design patterns, it provides transparency for both designers and developers with the share-ability feature. However, the adaptation rules are designed at the development time. Whenever a new rule is to be added, the system need to be redeployed, which is an expensive task.

Roam framework [16] provides environment for developers to create adaptive UI having responsive design feature. This toolkit has two main approaches to generate adaptive UI for the target device. In the first approach, it has used multiple device-depended UI, which is created at the design time. The selection of UI is made at runtime on the basis of the target device. In the second approach, a single UI design (i.e., universal design) is set according to the target device, at runtime, which is device independent. Unlike model-driven approaches, it uses the toolkit for UI creation at design time rather than runtime.

Like Roam framework, [44] XMobile has proposed an environment for the creation of adaptive UI, which uses multiple device-dependent UI variation based on the device characteristics. They have used a model-driven approach, however the code generated from the model is produced at design time rather than runtime.

In the literature, several commercial and academic open source tools are presented for the development of model-driven UI. These tools and software have used different user interface description language (UIDL) [23]. These UIDLs describes different aspects of a UI focusing on multi-platform, multi-context, device independence, and content. Usually based on XML, because XML is easily extensible, very expressive, declarative and can be used by normal users and naive developers. UIDLs can be differentiated on the basis of models, methodology, tools, supported languages, platforms, and concepts. TeresaXML [40] is a UIDL based on ConcurTaskTreeEnvironment (CTTE) [37] tool for modeling and analyzing the task modeling, which is based on the ConcurTaskTree (CTT) notations. Where Model-based lAnguage foR Interactive Applications (MARIA) [41] is an extension of TeresaXML that provides the authoring environment based on MariaXML which is compatible with the Cameleon Reference Framework [7]. It supports non-static behaviors, events, interactive web applications, and multi-target UIs. GrafiXML exploited UsiXML which is another UIDL for automatic generation of UI of different devices according to the contexts [32,35]. It comprises different abstraction levels models, such as task model, abstract UI

Table 1 Comparison of our proposed AUI-UXA with the existing work

Legend: ● Completely; ◑ Partially; ○ Does not; ◌ Not Specified

	Multimodal data source	Directed and indirect adaptation	Modeling approach	Context			Supporting Tool	Adaptation Aspects			User Feedback on Adaptation
				User	Platform	Environment		Presentation	Navigation	Content	
3-Layer Architecture	◑	◑	◑	●	●	●		◌	◌	◌	◑
CAMELEON-RT	●	●	◌	◑	◑	◑	●	◌	◌	◌	○
CEDAR	●	●	●	●	●	●	●	◑	●	●	●
Malai	◑	●	◑	○	●	○	●	●	●	◑	○
TRIPLET	◑	◑	●	●	●	●	◌	●	●	●	◑
Egoki system	◑	◑	●	●	●	○	○	●	●	●	◑
SUPPLE	◑	●	●	●	●	○	●	●	◑	○	○
MyUI	●	◑	●	●	●	●	●	●	●	◑	◑
Roam framework	◌	◑	●	○	●	○	◌	●	○	○	○
XMobile	◌	◑	●	○	●	○	◑	●	◑	○	◌
AUI-UXA	●	●	●	●	●	●	●	●	●	●	●

model, concrete UI model, and transformation model. Some other software such as WiSel focused on a framework for supporting Intelligent Agile Runtime Adaptation by integrating adaptive and adaptable approach [34]. However, the user interface markup language (UIML) [25] is best suited for our proposed A-UI/UX-A tool due to mapping of different resources with the UI elements. It is pioneer in the user interface markup languages and its implementation is dependent on vendor. UIML is an XML-based language which supports device, modality independent method for a UI specification. It interconnects UI appearance and interaction with the application logic.Most of adaptive UI system used the ontological models for storing the information for tailoring the UI [3,15,20]. The Table 1 shows the comparison of our proposed AUI-UXA with the existing work.

Our proposed model based system is designed by taking these limitations into account i.e. our system generates the UI at runtime, does not need to redeploy the system, and with the help of authoring tools new rules are added without effecting the running system. Additionally, the adaptation on UI is made when the context is change, which is observed by implicit and explicit (user feedback) ways and then evaluate the context and user experience.

3 Mining minds platform: an overview

Mining Minds (MM) [8–10] is our labs ongoing project, which is a novel platform that provides a collection of services, by monitoring the users daily routines and providing personalized wellness support services. The MM platform is built on a five layers architecture that uses the concept of curation at different levels in different layers. The curation concept is applied at data level in the data curation layer (DCL) [6], information level in the information curation layer (ICL) [11,45], knowledge level in the knowledge curation layer (KCL), and service level in the service curation layer (SCL) [4].The services are delivered through the support of supporting layer (SL) and personalized at the interface level by using the proposed concept of adaptive UI. Figure 1 shows how these layers are interconnected in the MM platform. MM platform acquires data from heterogeneous data source (various sensors, SNS, survey) via DCL [6]. The

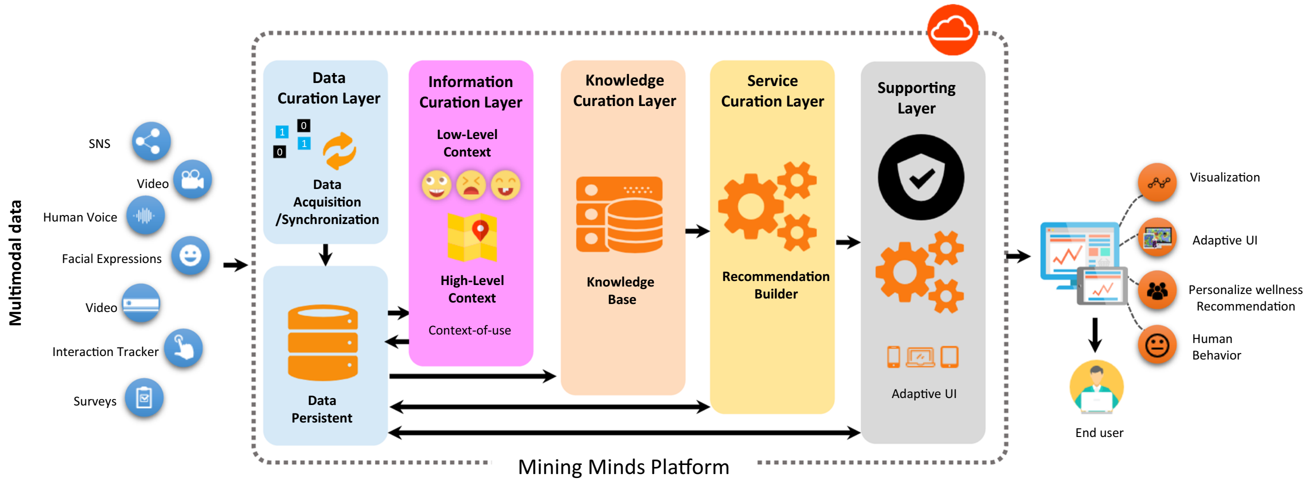

Fig. 1 Mining minds platform

acquired multimodal data is used in ICL to find the low-level and high-level contextual information. The contextual information describes the user context, user behavior, and user mental and social states. ICL sends the inferred information to DCL for storage in user life-log, which is a relational data model. The KCL uses two approaches for knowledge creation: data driven and expert driven. The domain expert uses the knowledge-authoring tool [5] for the creation of wellbeing rules utilizing the insights of inferred information recognized by the ICL. The SCL layer uses the created rules and users current context information in the multimodal hybrid reasoner [4] for the generation of personalized recommendation. The SL is responsible for the adaptive UI generation, service content presentation, information visualization and privacy and security related issues.

Data acquisition and synchronization (DAS) component in DCL is a REST base service that collects real-time data from multimodal data sources e.g. smart watch, mobile phone, camera, Kinect, and SNS. After acquiring the data, the synchronization is done based on the time stamp of the device, and queued based on event for mining of low level context (LLC) and high level context (HLC) [11,45] that consumed by SCL and SL for personalization of services in the form of adaptive UI. LLC is responsible for converting the multimodal data obtaining from user interaction into the classified data such as physical activities (e.g. running, walking, standing, and busing, etc.), user emotions, location, and weather information while HLC is responsible for the identification of user context by combining semantically the recognized LLC. Both LLC and HLC play important role in forming the adaptive UI from recognized context. For example, based on user recognized context (e.g. walking, running), UI adapt to simplified version such as bigger font-size and Icons etc.

The key role of SL is empowering the overall MM functionality via human behavior quantification, personalized user interface based on implicit and explicit feedback analysis for improving the positive experience via A-UI/UX-A tool [26], and privacy and security [1]. The analyzed feedback data use to enhanced the adaptation aspects such as presentation, navigation, and content. All these types of feedback are devised to help measuring user interest level and devotion of users to the services delivered through Mining Minds. Considering user capability, mood, way of interaction, A-UI/UX-A tool allows the end-user app UI adopted accordingly. This adaptation aligned the UI based on context and user experience with respect to presentation, navigation, and content. Initially, the user interaction data collects from the interaction between the user and the application to evaluate the users ability to understand and use the system, e.g. estimating the magnitude of a specific usability issue, of knowing how well users are actually using an application, Then, measures the satisfaction level based on the analysis of the collected data.

The key focus of this paper is on the design and development of A-UI/UX-A tool for MM platform, which can be easily adapted for any interactive system to provide adaptive user interfaces.

4 Framework for adaptive user interface

Motivations for adaptive user interface is to increase positive user experience in the term of accessibility and user satisfaction. To achieve the stated goal, [26] has proposed an initial adaptive UI/UX authoring tool that dynamically adapts UI based on the user context and experience, which is evaluated automatically. The proposed A-UI/UX-A tool has used a model-based approach, tailored with the UI, which

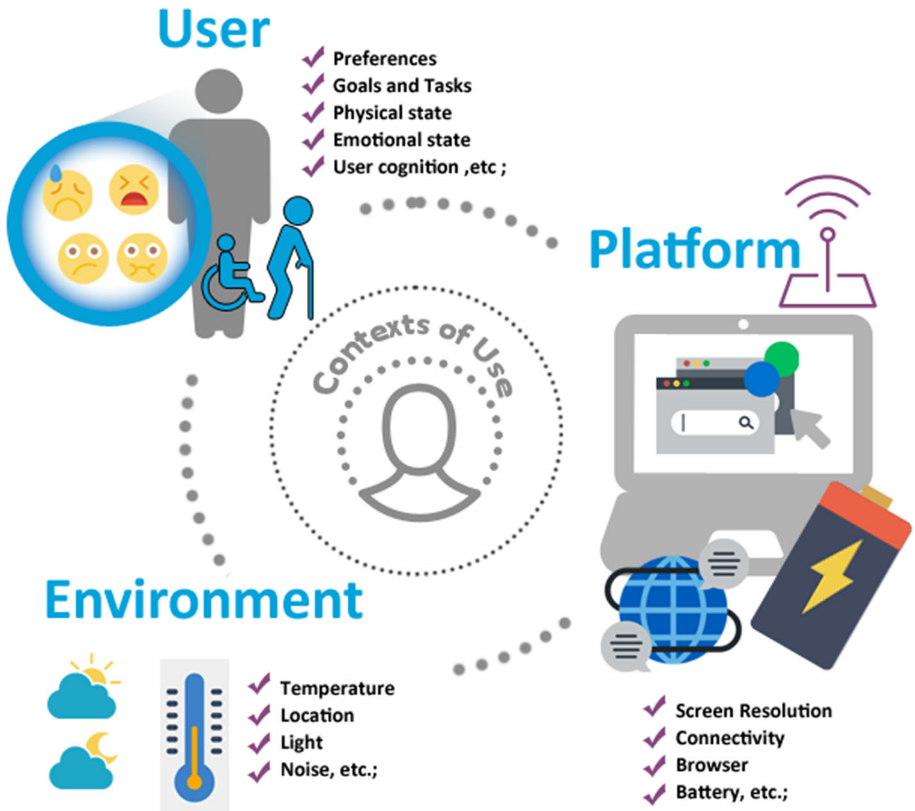

Fig. 2 The context-of-use for adaptive UI

is based on the context-of-use. This paper is an extension of the same work [26] that extends reference architecture, adaptation techniques with detailed empirical and statistical evaluation. Generally, the context-of-use consists of user, platform, and environment [7,14], as shown in Fig. 2.

The proposed A-UI/UX-A tool evaluates context-of-use, and user experience via context monitoring and feedback. The user feedback is collected through various ways, ranging from implicit feedback to explicit feedback. The implicit feedback is acquired from the user behavioral responses, which are collected automatically when user start interaction with the system, while the explicit feedback is acquired through questionnaires. From the evaluation of user response along with the context-of-use, the adaptation aspects are inferred in the term of functionality,navigation, content, and presentation of UI for provisioning personalized services to the end user. All these types of feedback are considered to evaluate the level of interest and devotion of users to the services.

The detailed methodology of the proposed idea of adaptive user interfaces in the context of MM in specific and every other adaptive UI design in general is explained in the next section.

5 Methods for adaptation of user interface

This section introduces the proposed system methodology in the form of an A-UI/UX-A tool, which is based on the evaluation of context and user experience. The construction of proposed system is divided into two processes: (i) offline process for models creation and adaptation rules generation and (ii) online process for adaptive UI generation.

5.1 Models creation and adaptation rules generation

To build A-UI/UX-A tool for adapting UI, the methodology comprises the development of different models and the creation of adaptation rules in the offline phase. These models and rules are the baseline requirements for the adaptive UI generation. The A-UI/UX-A tool has been used for modeling these models. The models main classes shown in the Fig. 3. The detail description of these models are given below.

5.1.1 User model

The user model stores information related to user cognition, physical characteristics, sensory, and user experience (UX). The general user model ontology (GUMO) [24] model is used with additional classes and subclasses required for the adaptive UI creations. User cognition is all about the user attention, learning ability, concentration, and user percep-

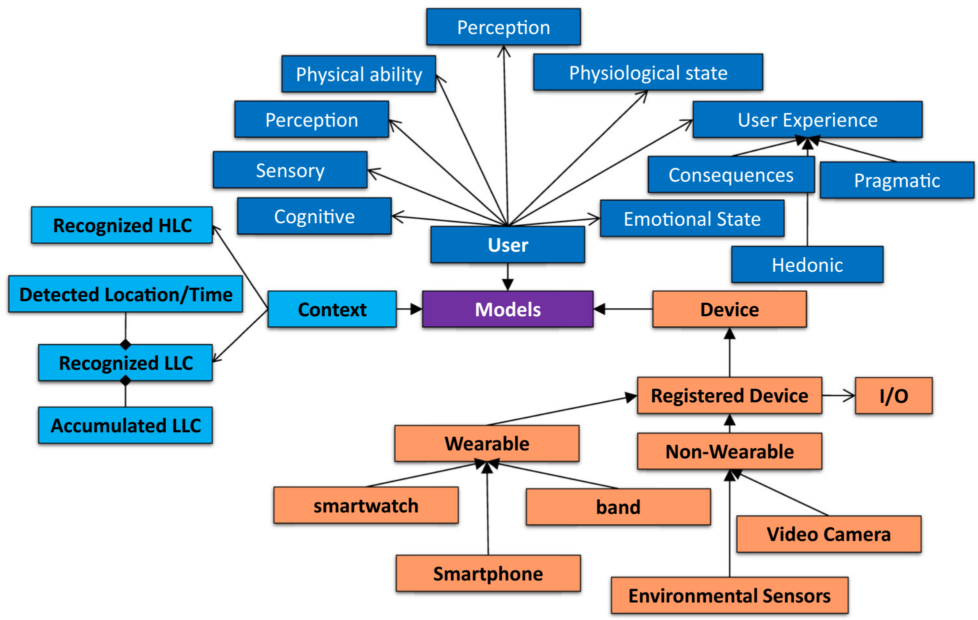

Fig. 3 Models for user, context, and device

tions. Physical characteristics are modeled as user mobility and abilities or disabilities that effect the user interaction with the system, such as hand or finger precision. User sensory information are modeled as user sight, hearing, and touch sensitivity that also have direct impact on user interaction with the system. The user positive and negative emotions are modeled as user experience information. The UX is all about how the user feels about any artifact before and after the usage [30]. The UX constructs was mainly divide into products perceived hedonic quality, pragmatic quality, goodness and beauty [30]. We added new construct, such as emotional state because the current constructs are not enough to model the UX. UX is used to check the level of satisfaction of user interface adaptation after changing the UI according to user context.

5.1.2 Context model

Context model is used to adapt the system based on the current situations. It store information about the contextual factors such as light, noise level, and event occurrences in the environment. The context information is classified as follows.

- *Physical context* The environmental variables, such as light and noise level, temperature and weather information are included as physical context, which are collected through the environmental sensors.
- *Time and location context* The temporal and location information are the essential elements of any context model and we model them together to enable the system for answering questions, such as where the user is right now or where he/she was at a particular time.

5.1.3 Device model

Device model stores information about different characteristics of the devices, such as screen resolution and their abilities of displaying content. These characteristics are essential for the efficient adaptation of a UI. For example, smart phone and tablet require different adaptations at the user interface. Additionally, the user preferred input modality is also required for the UI adaptation. The device characteristics are mainly divided into two types:

- *Hardware* All hardware related features are modeled as input/output capabilities (e.g., mouse, keyboard, HUD, HMD, touch, sensory input, eye-gauze), interaction modalities (e.g., graphics, speech, haptics, gesture, EEG, ECG), memory, battery, connectivity and so on.
- *Software* Software related information, installed on the device are modeled as operating system platform, web browser and supporting markup languages and so on.

5.1.4 Adaptation rules generation using rule authoring tool

The A-UI/UX-A tool is web-based that provide a way to create the adaptation rules in intuitive way. In rule authoring tool, the concepts are selected from model hierarchy, that associated with the contextual dimension (user, platform and environment). The user can create rules in the form of Conditions-Actions [22] starting either from trigger or from

Table 2 A partial list of the adaptation rules used in the generation of adaptive UI on the basis of context

RuleID	Rule name	Descriptions	Event	Condition	Action
R1	Noisy environment	For noisy environment, the UI should be in only-graphical mode	The environment becomes noisy	The graphical and vocal modality used by application for user interaction	The application changes to the only-graphical modality
R2	Light Level	The environment light intensity is high or low, then the application switch to night or day mood accordingly for the greater information accessibility	Based on light sensor lux values	The light level is too low	The user interface changed to a night mood
R3	Color Blind	If the user is colorblind then change the application color to black and white	onRender	The user is a colorblind	Change the foreground color to black and background color to white
R4	Low Vision	If the user has low vision by checking then increases the UI size accordingly	The size of the text of the UI is smaller than 16px	The user has low vision	Increase the size of the text of the UI to 16px
R5	Cognitive	If the user has cognitive problem then simplify the UI	The application contains too many different interaction elements for performing different tasks	the user has a cognitive disability	Split the UI into simplified UI having multi-steps to achieve the desire goal
⋮	⋮		⋮	⋮	⋮
Rn	Mobility	If the user health condition is Parkinson and current context is user is motion, then the UI mode change to multimodal mode	The user begins to move	The user has the Parkinson AND the UI is not of the type of multimodal mode	The UI changed to multimodal mode

actions. Each rule has two parts: condition part, and action part as follows:

IF ⟨Condition Part⟩ Do ⟨Action Part⟩

The ⟨Condition Part ⟩ has event(s) that describes the occurrence of actions of the rule. It can be either one condition or more, concatenated using boolean operators for the executions of action(s). In the action part, there might be one or more actions associated to the same condition part when the rule triggered. Below, there is a partial list of adaptation rules supported by the MM Platform shown in Table 2. For each rule, we use a rule name with a brief explanation and the three key parts, i.e., event, condition, and action.

5.2 Adaptive UI generation

The whole adaptation process is pictorially represented in Fig. 4. In the offline phase of adaptive UI design, all the relevant models are built and the adaptation rules are generated using rule authoring tool. The Created rules subscribed as event in context evaluator.

This process is termed as real-time monitoring of the users context and reasoning. In the monitoring process, the information required for the reasoner to adapt the UI behavior, is obtained using implicit and explicit strategies. The real adaptive behavior data preparation process start from user interaction with the system. The monitoring module is responsible for data collection while user is interacting with the system through different sensors and trackers (e.g., facial, vocal, eye, and analytics). We also consider the user feedback as a self-reported data. The evaluator component evaluates the acquired information and decides whether adaptation is required on UI or not. If any adaptation is needed, UI is adapted accordingly, otherwise ignored. The adaptation on user interface is made when the context is changed, which is monitored by the context monitor, and sent the context information to the context evaluator. The context evaluator makes decision about the adaptation on UI by checking the current states of the system according to the context-of-use. Based on the decision made by context evaluator the adaptation engine invoked. All the data and model that are required based on current situation by the adaptation engine are loaded along with the adaptation rules. Adaptation engine preforms

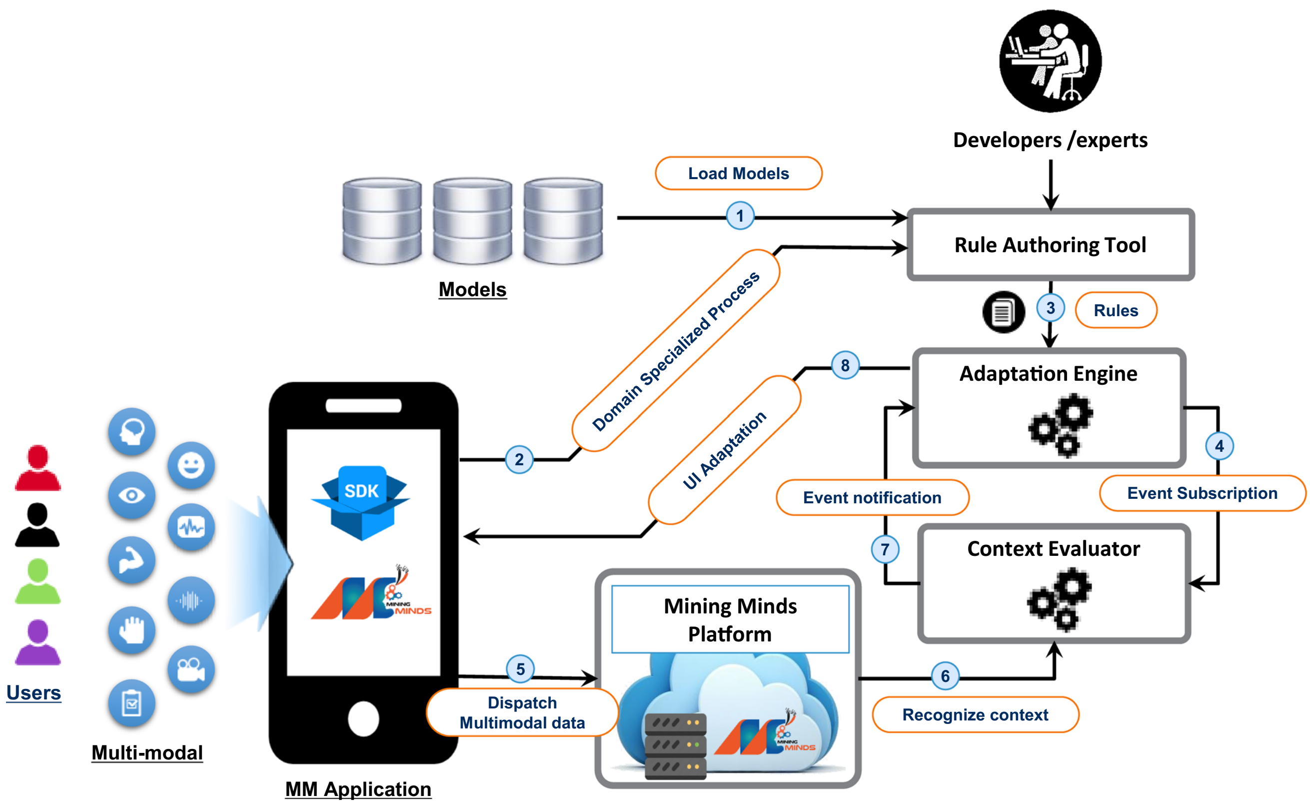

Fig. 4 Adaptive behavior data flow

reasoning using reasoning module called reasoner. Reasoner possess a pattern matcher that uses a forward chaining mechanism, by checking the conditions of selected rule loaded in the reasoner. The rule added in the resultant result of the pattern matcher, if all conditions of rule are satisfied. The resultant rules list is passed to the conflict resolver. The conflict resolver acts as a trade-off between multiple adaptations aspects in the given situation because the resultant list of pattern matcher might be many rules that might have conflict. After that, the result generator fires the final rules and sends to the adaptation engine to generate the UI by the generator engine module in the form of content, presentation and navigation adaptation aspects at the target device.

6 Implementation, experiments and evaluation: realization of the adaptive UI methodology

6.1 Use-case scenario

To validate the proposed adaptive UI methodology, we considered a wellness application scenario from a real world health and wellness platform, so called Mining Minds (MM). The MM platform is to provide wellness recommendation to different age users having different characteristics, using different devices under different context-of-use. An application on the top MM platform is previously developed and we designed the application UI to validate the proposed methodology from the operability and accessibility perspectives. The initial UI design of the MM application is shown in Fig. 5.

The main sections of the MM application UI are: list view of generated recommendation based on user activities, social sharing, archive, user activities graphs, user feedback invoking by users on recommendations and overall application features, and prompt feedback invoking by UX evaluator based on user app usage behavior. These are the defaults controls and elements of the MM application to validate the proposed model-based adaptive UI methods, consider the following real world scenario shown in Fig. 6.

6.1.1 Scenario

John is 31-year-old, overweight person with a visual impairment. He installed the MM application and use it for getting physical activity recommendation to control his body weight. As John has special conditions, therefore the UI of MM application is adapted according to his special characteristics. The adaptation process for this scenario is described below.

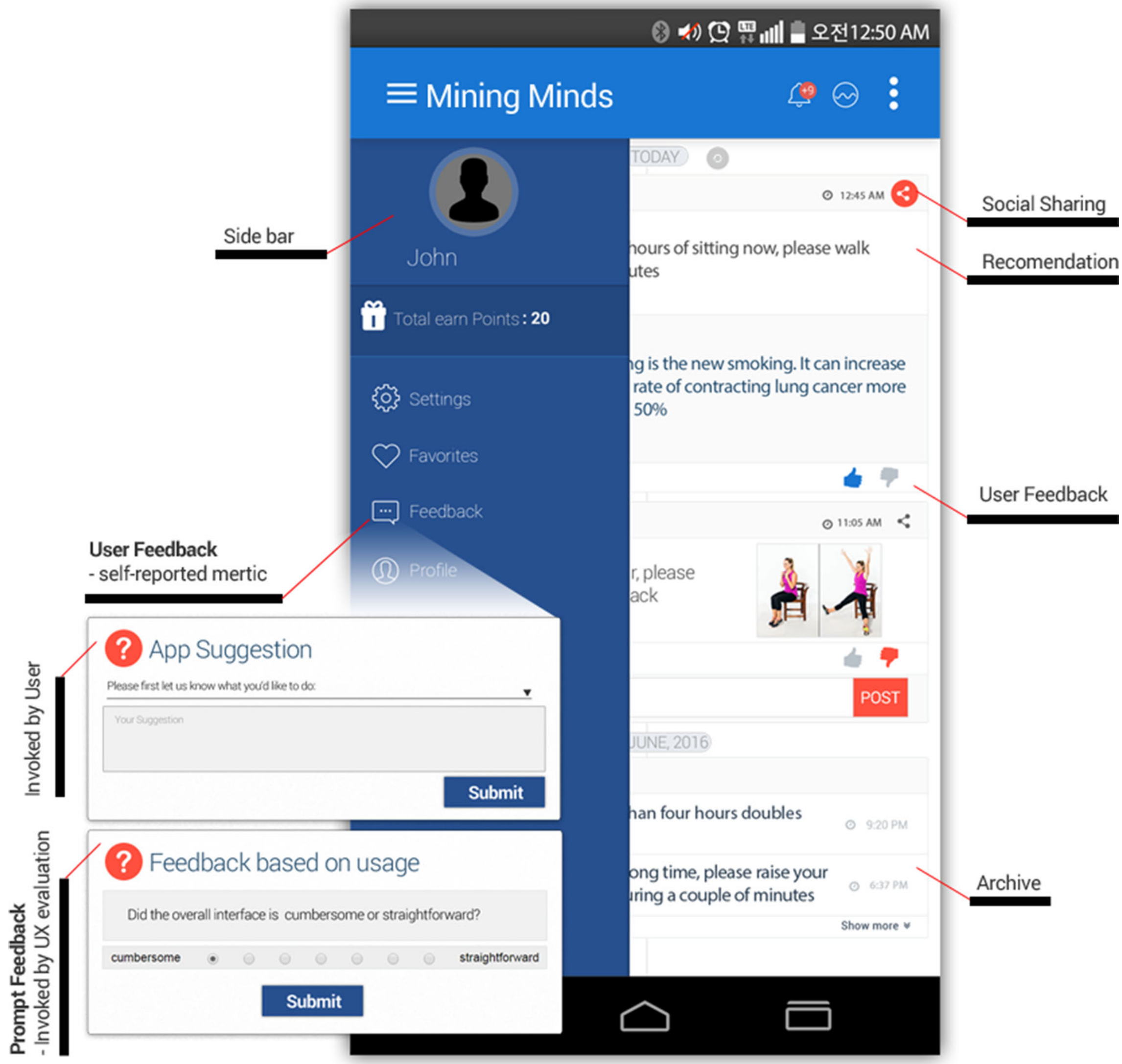

Fig. 5 Mining minds platform application dashboard

1. John's characteristics such as his preferences, visual impairment, and cognition information are collected during his registration in Mining Minds application and stored in the user model.
2. After registrations, the user gets login to use MM application for the accessing wellness services.
3. As user has a low vision, the context evaluator infers the UI needs to be changed. It provides a flag to the reasoner to start reasoning for the corresponding adaptation.
4. In the adaptation engine, the reasoner is invoked to fire the appropriate rule (Rule 4) for the required adaptation according to the current situation.
5. The action of adaptation engine is get effective and the adaptation takes place (i.e., bigger fonts, icons size, and simple UI) for the generation of adaptive UI.

6.2 Implementation

To execute the proposed methodology of adaptive UI, we developed the adaptive UI engine so a so-called A-UI/UX-A tool. The tool is developed in the laravel PHP Framework [43] as a web application along with other additional libraries as follow.

1. Protégé editor is used for models creation.
2. Pallet reasoner and OWL API are used for accessing the model ontologies and do inferencing using Semantic Web Rule Language (SWRL) rules.
3. Easyrdf, a PHP library, is used for data accessing and storage from/to Resource Description Framework (RDF).
4. For the xml documents creation, parsing, and manipulation, a laravel-parser is used.

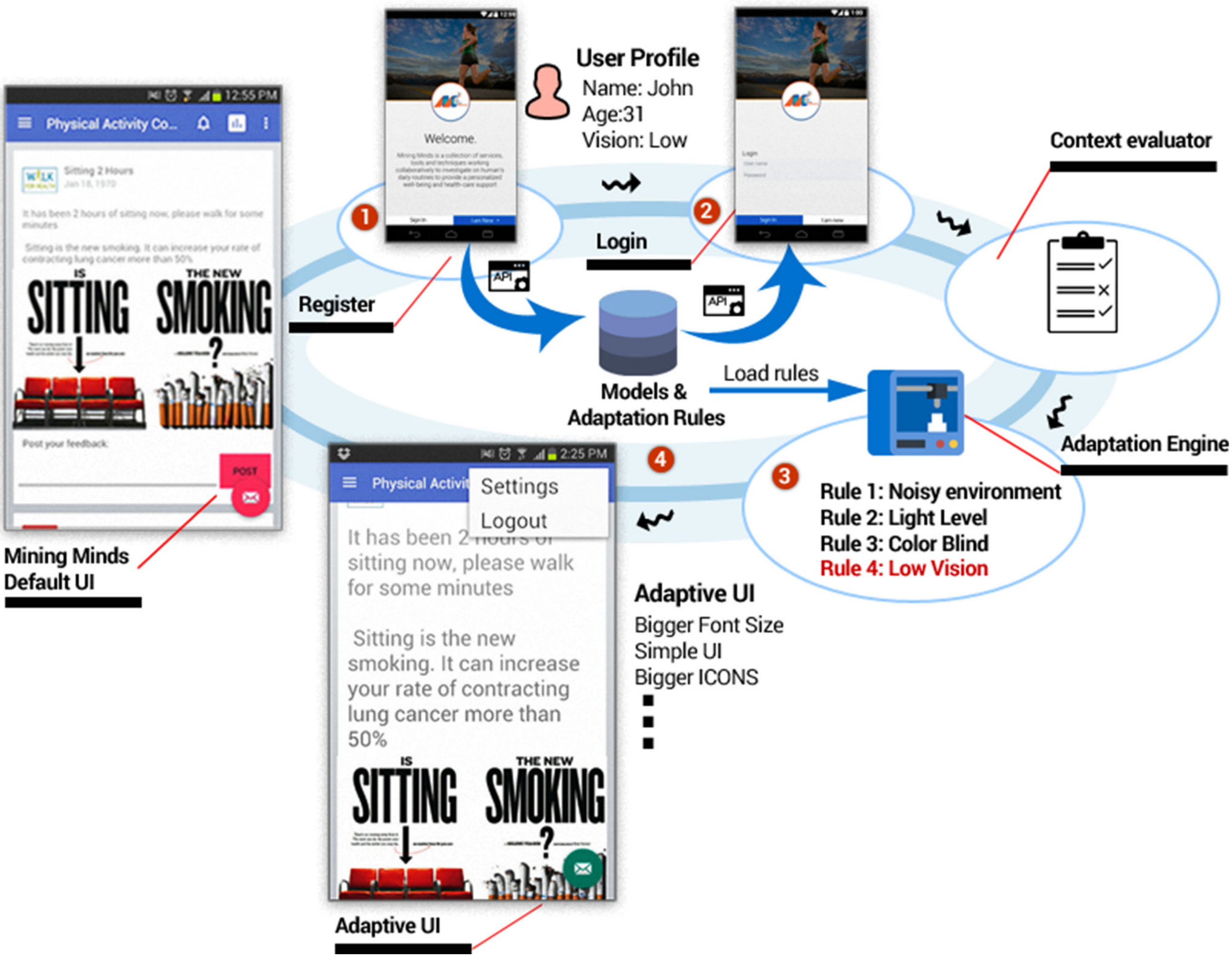

Fig. 6 Low vision scenario

All the ontological models, used in A-UI/UX-A tool, are developed using OWL in Protégé editor and SWRL rules are used for inferencing over the pre-adaptation rules. The final user interfaces are web-based UIs, which are designed using HTML, JavaScript (JQuery, and AngularJs framework). The rationale of using these techniques and technologies is to support interactivity and extraction of users behaviors from the UI.

6.3 Experiments and evaluation

We performed user-based evaluation for adaptive user interfaces that are automatically generated using the proposed model-based adaptive UI methodology using the developed A-UI/UX-A tool. For the evaluation, we address the following research questions:

RQ1 How the adaptive UI behavior improves the efficiency?

RQ2 How the adaptive UI behavior improves the user satisfaction?

RQ3 How adaptive UI improves the positive user experience (UX)?

6.3.1 User recruitment

In the evaluation of adaptive user interface of MM application, 32 participants (MM users) were used for evaluation purpose and their profile information are shown in Table 3. Participants are from different countries and observed different cultures. The participants were from Pakistan, Vietnam, China, Korea, Egypt, Spain, Yemen, Ecuador, Guatemala, Bangladesh, India, Iran, and Australia. Each of the users had different demographics, such as age, gender, vision impairment, education, and wellness applications expertise etc. The participants were provided with initial training of the MM application usage. The participants are briefly addressed regarding the purpose of the research and got their willing-

Table 3 Personal profile information of the volunteers who participated in the evaluation of mining minds platform (n = 32)

	No. of users	% of users	Mean (SD)
Age (years)			29.125 (6.8)
18–24	10	31.25	
25–34	13	40.625	
35–44	9	28.125	
Gender			
Male	25	78.125	
Female	7	21.875	
Health Status			
Normal	12	37.5	
Hypertension	10	31.25	
Obesity	10	31.25	
Activity Level			
Normal	13	40.625	
Active	10	31.25	
Sedentary	9	28.125	
Disabilities			
Vision	17	53.125	
Limb	7	21.875	
Hearing	4	12.5	
No	4	12.5	
Education			
Under graduation	19	59.375	
Graduation	8	25	
Post-graduation	5	15.625	
Computation Expertise			
Expert	27	84.375	
Intermediate	4	12.5	
Novice	1	3.125	
Ethnicity, culture			
East Asia	12	37.5	
South Asia	11	34.375	
Australia	4	12.5	
Middle East	3	9.375	
Europe	2	6.25	
Upper limb Usage			
Right hand	16	50	
Both	11	34.375	
Left hand	5	15.625	

ness. The participants had personal computing devices like smart phone, laptop and desktop, and tablets and had access to internet on these devices 24/7. These participants were already using wellness application and health conscious.

6.3.2 Types of experiments and evaluation criteria

We performed three types of experiments. These includes Perceived Usability, User Satisfaction, and User Experience Assessment. For the perceived usability, we used the System Usability Scale (SUS) [13], which is one of the most commonly used measures in literature. SUS questionnaire performed more accurately than computer system usability questionnaire (CSUQ) and Post Study System Usability Questionnaire (PSSUQ) when sample size greater than 8. The user subjective satisfaction is assessed by using the criteria of Questionnaire for User Interaction Satisfaction [39], which measures the overall system satisfaction in term of nine specific UI factors. The user experience assessment, called User Experience Questionnaire (UEQ) [29] is used. The UEQ allow a rapid assessment of the user experience by getting user express feelings, impressions, and attitudes after using a product. It measures both classical usability aspects as well as user experience aspects. It has been used by different companies for the evaluation their products and is a good measure, therefore we have also adopted it in our study.

6.3.3 Evaluation process

For the user evaluation of the proposed methodology, the real-world application A-UI/UX-A tool, developed as a part of mining minds platform, was given to all the participants to use it for a period of one month. After full use of the application, the participants were asked to fill-out the questionnaires (SUS, QUIS, and UEQ) to find out the A-UI/UX-A tool perceived usability, user satisfaction, and user experience. The results of each of the experiments are given in the sub-sequent sections.

6.3.4 Perceived usability and efficiency results

The average SUS score is 89.7, which is ranked as B+ means that MM application is higher perceived usability.

6.3.5 User satisfaction

In many cases, the efficiency is less important than how satisfied the users are while they are experiencing the product. Therefore, for the user satisfaction measurement, we used the Questionnaire for User Interaction Satisfaction (QUIS), Fig. 7 shows the means values for each scale.

The mean response for the questions was 5.833 with SD = 1.048, which means that the overall user satisfaction of MM application is above the average. The confidence intervals for the scale means are smaller that estimate higher is the precision, more trust the results and shows how consistent the participants judged the A-UI/UX-A tool. The alpha-coefficient values are higher than 0.7 for all the scales

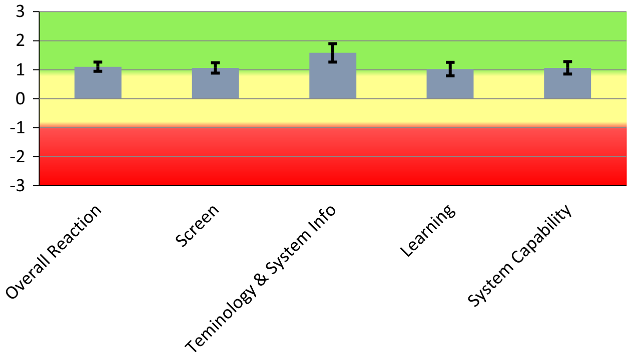

Fig. 7 User interaction satisfaction (QUIS) scores for each factors

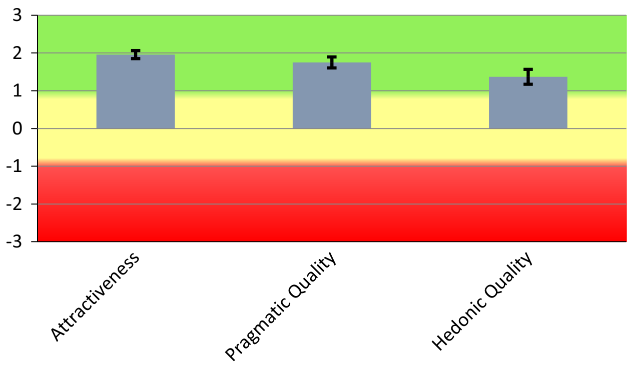

Fig. 8 The UEQ pragmatic and hedonic quality score

except terminology and system information. This may be due to the users misinterpretation of the terminology and system information.

6.3.6 User experience assessment

For the user experience assessment, the participants were asked to fill the UEQ questionnaire. UEQ is the widely used questionnaire for the subjectivity measurement of the user experience of any interactive system. They provide a tool in the form of excel sheet for capturing the user experience of users, while they are interacting with the product. It consists of six dimension scales such as attractiveness, perspicuity, efficiency, dependability, stimulation, and novelty.

The scales of questionnaire are grouped into the pragmatic quality (perspicuity, efficiency, and dependability) and hedonic quality (stimulation, originality). The pragmatic quality is related to the task, while the hedonic quality is represented as non-task related aspects. Figure 8 shows the pragmatic and hedonic quality aspects of MM application along with the application attractiveness.

The results show that all the scales have quite good results including the hedonic and pragmatic aspect of the MM application. In Fig. 9, the smaller confidence interval indicates that the measurements are accurate. The value of Cronbachs alpha-coefficient of attractiveness is higher than 0.7, which shows that users like the adaptive UI generated by the A-UI/UX-A tool. The value of Cronbachs alpha-coefficient for novelty is low, which means that it does not play an important role in adaptive UI.

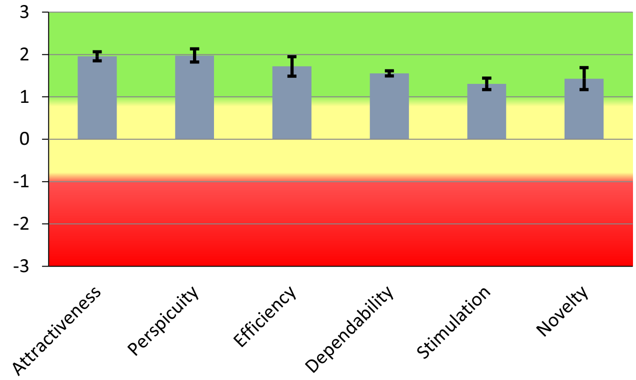

Fig. 9 UEQ resultant scores for six dimensions scales

Table 4 represents the correlation among the UX factors. The evaluation depicts that attractiveness is correlated to perspicuity, stimulation; perspicuity is correlated to efficiency and dependability; dependability is correlated to stimulation and novelty; and Stimulation is correlated to novelty.

The UEQ also provide a benchmark that contains data collected from 4818 participants of 163 products evaluation. The benchmark easily gives insight of a comparative analysis that a product satisfactory user experience to be successful in the market. In Fig. 10, the comparison results for the evaluated MM application are relatively good as compared to benchmark data.

The Kendalls correlation is shown Table 5, which depicts that there is agreement among the participants for all UEQ factors. The value above 0.7 is considered excellent in its agreement, which is the case for 4 factors: **Attractiveness**, **Perspicuity**, **Efficiency**, and **Novelty**. The minimum level of agreement is shown in the **Stimulation** and **Dependability** factors.

7 Discussion

The evaluation results obtained from user based evaluation, out of 32 participants, there were 3 participants which were not able to use the application for maximum of 5 days. On average all the participants use the application more than 27 days. From the results achieved, we concluded that the adaptive UIs generated by A-UI/UX-A tool for all users having impairments have positive user experience because the accessibility of all services functionality are increased.

The user based evaluation results show that performance of the UI improved system functionality. UI is adapted

Table 4 Correlation of UX factors/scales

	Attractiveness	Perspicuity	Efficiency	Dependability	Stimulation	Novelty
Attractiveness	1	0.1598701	0.525075	0.214811376	0.49154268	−0.2987
Perspicuity		1	0.246627	0.091777454	−0.47288	−0.197
Efficiency			1	0.054750579	−0.4114699	−0.5516
Dependability				1	0.20404049	0.65361
Stimulation					1	0.01932
Novelty						1

Significant with $p < 0.05$

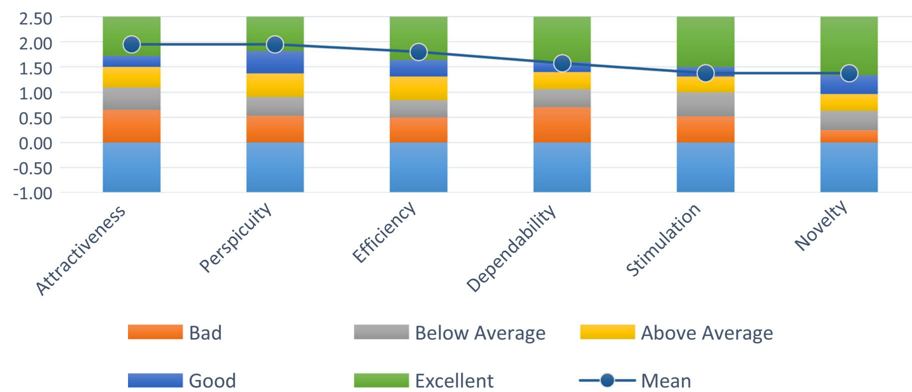

Fig. 10 UEQ resultant scores for six dimensions scales with benchmark data

Table 5 Kendall's W of UEQ factors

Attractiveness	0.771
Perspicuity	0.855
Efficiency	0.836
Dependability	0.556
Stimulation	0.453
Novelty	0.753

according to the user ability and requirements. The SUS evaluation scored greater than 89% which ranked it as B+. It means that the users efficiency increased with the adaptability behavior of the UI. It is noted that the adaptive accuracy of UI has significant impact on user performance.

The hypothesis regarding the user satisfaction is evaluated through QUIS with alpha coefficient score which is more than 0.7. It means that users are more satisfied with the adaptive ability of the MM application. However, frequent adaptation which causes change in UI, annoying some of the hypertensive users. It disturbs their learning ability and cause the negative impact on overall reaction.

The user experience in terms of hedonic and pragmatic quality is evaluated through UEQ. The evaluation represents that hedonic quality is little low than pragmatic quality. It is because the occasional diminish of UI representation due to adaptive UI behavior. However, A-UI/UX-A have some issues to be considered.

– *Issue with the final UI presentation* The analysis of user revealed problem with the final user interface presentation such as UI elements adjustment and alignment, which sometimes break the UI design and functions. Automatically generated user interfaces are generally perceived less aesthetic appeal as compared to create by a designer. User interfaces created by a designer reflects the creativity and are well aligned with application. Furthermore, recurrent adaptations diminish the consistency in the UI, and reduce the learning rate. For example, frequent changes in the UI may frustrate and confuse some users.
– *Issue with model and adaptation rule creation* Indeed, model-driven user interface begin with models creation, which required expertise even the system provide graphical user interface for creating such models. Although, we provide A-UI/UX-A tool, the designer can create models and adaptation rules that can manage the adaptation in user interface based on the user context. However, the creation of complex rules is difficult to manage.

8 Conclusion

The proposed model-based system is designed by taking the limitations of existing system into account. The existing systems are not capable of generating UI at runtime, require the redeployment of the system, and new rules are not added without effecting the running system. In addition, these systems lack in modeling approach, considering multimodal data sources, user feedback and content base adaptation. While our proposed methodology comprehends multimodal data for context identification; support direct and indirect adaptation; converting generalized context model into specialized domain context through authoring tool while considering the environment, platform and user; and focusing the content along with presentation and navigation in adaptation aspects. Last but not least, the adaptation on UI is made when the context is change, which is observed by implicit and explicit (user feedback) ways and then evaluate the context and user experience. It considers the dynamics of the UI associated with the user in the form of context-of-use.

It helps in improving the information accessibility, usability, user experience of system. The efficiency of the proposed methodology with respect to adaptive UI ranked as B+ which is considered as quiet acceptable in term of usability. The QUIS questionnaires are used to evaluate the overall user satisfaction of the proposed methodology. The obtained alpha score is higher than 0.7 for all the scale except terminology and system information due to misinterpretation. The user experience assessment is evaluated through widely used UEQ questionnaire for the subjectivity measurement of the user experience of any interactive system in six dimensions e.g. attractiveness, perspicuity, efficiency, dependability, stimulation, and novelty. The results show that hedonic quality is lower than pragmatic quality due to occasional diminish of UI representation. Adaptive UI representation generation is generally perceived less aesthetic as compared to create by a designer. Designer created User interfaces reflects the creativity and are well aligned with application. Furthermore, recurrent adaptations decrease the consistency in the UI, and reduce the learning ability.

Currently the rule authoring is able to manage basic level adaptation rule. In future, we will improve the rule-authoring tool for management of complex adaptation rules and as well a final UI presentation issue. The authoring tool can be enhanced for application users to add specialized rules, based on personalized context. In addition to user based evaluation, we will enhance evaluation through physiological measurements to remove subjectivity in evaluating user experience.

Acknowledgements This work was supported by the Industrial Core Technology Development Program (10049079, Develop of mining core technology exploiting personal big data) funded by the Ministry of Trade, Industry and Energy (MOTIE, Korea). This work was supported by the MSIT (Ministry of Science and ICT), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2017-0-01629) supervised by the IITP (Institute for Information & communications Technology Promotion). This work was supported by Institute for Information & communications Technology Promotion (IITP) grant funded by the Korea government (MSIT) (No. 2017-0-00655) and NRF-2016K1A3A7A03951968.

References

1. Ahmad M, Amin MB, Hussain S, Kang BH, Cheong T, Lee S (2016) Health fog: a novel framework for health and wellness applications. J Supercomput 72(10):3677–3695
2. Akiki PA, Bandara AK, Yu Y (2014) Adaptive model-driven user interface development systems. ACM Comput Surv CSUR 47(1):9
3. Akiki PA, Bandara AK, Yu Y (2016) Engineering adaptive model-driven user interfaces. IEEE Trans Softw Eng 42(12):1118–1147
4. Ali R, Afzal M, Hussain M, Ali M, Siddiqi MH, Lee S, Kang BH (2016) Multimodal hybrid reasoning methodology for personalized wellbeing services. Comput Biol Med 69:10–28
5. Ali T, Lee S (2016) Wellness concepts model use and effectiveness in intelligent knowledge authoring environment. In: International conference on smart homes and health telematics. Springer, pp 271–282
6. Amin MB, Banos O, Khan WA, Muhammad Bilal HS, Gong J, Bui DM, Cho SH, Hussain S, Ali T, Akhtar U et al (2016) On curating multimodal sensory data for health and wellness platforms. Sensors 16(7):980
7. Balme L, Demeure A, Barralon N, Coutaz J, Calvary G (2004) Cameleon-rt: a software architecture reference model for distributed, migratable, and plastic user interfaces. In: European symposium on ambient intelligence. Springer, pp 291–302
8. Banos O, Bilal Amin M, Ali Khan W, Afzal M, Hussain M, Kang BH, Lee S (2016) The mining minds digital health and wellness framework. Biomed Eng Online 15(1):165–186. https://doi.org/10.1186/s12938-016-0179-9
9. Banos O, Bilal-Amin M, Ali-Khan W, Afzel M, Ahmad M, Ali M, Ali T, Ali R, Bilal M, Han M, Hussain J, Hussain M, Hussain S, Hur TH, Bang JH, Huynh-The T, Idris M, Kang DW, Park SB, Siddiqui M, Vui LB, Fahim M, Khattak AM, Kang BH, Lee S (2015) An innovative platform for person-centric health and wellness support. In: Proceedings of the international work-conference on bioinformatics and biomedical engineering (IWBBIO 2015)
10. Banos O, Bilal-Amin M, Ali-Khan W, Afzel M, Ali T, Kang BH, Lee S (2015) The mining minds platform: a novel person-centered digital health and wellness framework. In: Proceedings of the 9th international conference on pervasive computing technologies for healthcare (PervasiveHealth 2015)
11. Banos O, Villalonga C, Bang JH, Hur TH, Kang D, Park SB, Hyunh-The T, Vui LB, Amin MB, Razzaq MA, Ali Khan W, Hong CS, Lee S (2016) Human behavior analysis by means of multimodal context mining. Sensors 16(8):1–19
12. Blouin A, Beaudoux O (2010) Improving modularity and usability of interactive systems with malai. In: Proceedings of the 2nd ACM SIGCHI symposium on engineering interactive computing systems. ACM, pp 115–124
13. Brooke J et al (1996) Sus: a quick and dirty usability scale. Usability Eval Ind 189(194):4–7
14. Calvary G, Coutaz J, Thevenin D, Limbourg Q, Bouillon L, Vanderdonckt J (2003) A unifying reference framework for multi-target user interfaces. Interact Comput 15(3):289–308
15. Castillejo E, Almeida A, López-de Ipiña D (2014) Ontology-based model for supporting dynamic and adaptive user interfaces. Int J Hum Comput Interact 30(10):771–786
16. Chu H, Song H, Wong C, Kurakake S, Katagiri M (2004) Roam, a seamless application framework. J Syst Softw 69(3):209–226
17. Coutaz J (2010) User interface plasticity: Model driven engineering to the limit! In: Proceedings of the 2nd ACM SIGCHI symposium on engineering interactive computing systems. ACM, pp 1–8
18. Daniel AO, Yinka A, Frank I, Adesina S (2013) Culture-based adaptive web design Int J Sci Eng Res 4(2)

19. Gajos KZ, Weld DS, Wobbrock JO (2010) Automatically generating personalized user interfaces with supple. Artif Intell 174(12):910–950
20. Gamecho B, Minón R, Aizpurua A, Cearreta I, Arrue M, Garay-Vitoria N, Abascal J (2015) Automatic generation of tailored accessible user interfaces for ubiquitous services. IEEE Trans Hum Mach Syst 45(5):612–623
21. Ghiani G, Manca M, Paternò F (2015) Authoring context-dependent cross-device user interfaces based on trigger/action rules. In: Proceedings of the 14th international conference on mobile and ubiquitous multimedia. ACM, pp 313–322
22. Ghiani G, Manca M, Paternò F, Santoro C (2017) Personalization of context-dependent applications through trigger-action rules. ACM Trans Comput Hum Interact TOCHI 24(2):14
23. Guerrero-Garcia J, Gonzalez-Calleros JM, Vanderdonckt J, Munoz-Arteaga J (2009) A theoretical survey of user interface description languages: preliminary results. In: Web congress, 2009. LA-WEB'09. Latin American. IEEE, pp 36–43
24. Heckmann D, Schwartz T, Brandherm B, Schmitz M, von Wilamowitz-Moellendorff M (2005) GUMO—the general user model ontology. In: International conference on user modeling. Springer, pp 428–432
25. Helms J, Schaefer R, Luyten K, Vermeulen J, Abrams M, Coyette A, Vanderdonckt J (2009) Human-centered engineering of interactive systems with the user interface markup language. Hum Cent Softw Eng 139–171. https://doi.org/10.1007/978-1-84800-907-3_7
26. Hussain J, Khan WA, Afzal M, Hussain M, Kang BH, Lee S (2014) Adaptive user interface and user experience based authoring tool for recommendation systems. In: International conference on ubiquitous computing and ambient intelligence. Springer, pp 136–142
27. Jorritsma W, Cnossen F, van Ooijen PM (2015) Adaptive support for user interface customization: a study in radiology. Int J Hum Comput Stud 77:1–9
28. Langley P (1997) Machine learning for adaptive user interfaces. In: Annual conference on artificial intelligence. Springer, pp 53–62
29. Laugwitz B, Held T, Schrepp M (2008) Construction and evaluation of a user experience questionnaire. In: Symposium of the Austrian HCI and usability engineering group. Springer, pp 63–76
30. Law ELC, van Schaik P (2010) Modelling user experience-an agenda for research and practice. Interact Comput 22(5):313–322
31. Lehmann G, Rieger A, Blumendorf M, Albayrak S (2010) A 3-layer architecture for smart environment models. In: 2010 8th IEEE international conference on pervasive computing and communications workshops (PERCOM workshops). IEEE, pp 636–641
32. Limbourg Q, Vanderdonckt J, Michotte B, Bouillon L, López-Jaquero V (2004) USIXML: a language supporting multi-path development of user interfaces. In: International workshop on design, specification, and verification of interactive systems. Springer, pp 200–220
33. Meixner G, Paterno F, Vanderdonckt J (2011) Past, present, and future of model-based user interface development. i-com 10(3):2–11
34. Mezhoudi N, Khaddam I, Vanderdonckt J (2015) Wisel: a mixed initiative approach for widget selection. In: Proceedings of the 2015 conference on research in adaptive and convergent systems. ACM, pp 349–356
35. Michotte B, Vanderdonckt J (2008) GrafiXML, a multi-target user interface builder based on Usixml. In: Fourth international conference on autonomic and autonomous systems (ICAS'08), pp 15–22. https://doi.org/10.1109/ICAS.2008.29
36. Miñón R, Paternò F, Arrue M (2013) An environment for designing and sharing adaptation rules for accessible applications. In: Proceedings of the 5th ACM SIGCHI symposium on engineering interactive computing systems. ACM, pp 43–48
37. Mori G, Paternò F, Santoro C (2002) CTTE: support for developing and analyzing task models for interactive system design. IEEE Trans Softw Eng 28(8):797–813
38. Motti VG, Vanderdonckt J (2013) A computational framework for context-aware adaptation of user interfaces. In: 2013 IEEE seventh international conference on research challenges in information science (RCIS). IEEE, pp 1–12
39. Norman KL, Shneiderman B, Harper B, Slaughter L (1998) Questionnaire for user interaction satisfaction. University of Maryland (Norman, 1989) Disponível em
40. Paternò F, Santoro C (2003) A unified method for designing interactive systems adaptable to mobile and stationary platforms. Interact Comput 15(3):349–366
41. Paterno F, Santoro C, Spano LD (2009) Maria: a universal, declarative, multiple abstraction-level language for service-oriented applications in ubiquitous environments. ACM Trans Comput Hum Interact TOCHI 16(4):19
42. Peissner M, Häbe D, Janssen D, Sellner T (2012) MyUI: generating accessible user interfaces from multimodal design patterns. In: Proceedings of the 4th ACM SIGCHI symposium on engineering interactive computing systems. ACM, pp 81–90
43. Surguy M (2013) History of Laravel PHP framework, Eloquence emerging. Maxoffsky. http://maxoffsky.com/codeblog/history-of-laravel-php-framework-eloquence-emerging
44. Viana W, Andrade RM (2008) XMobile: a MB-UID environment for semi-automatic generation of adaptive applications for mobile devices. J Syst Softw 81(3):382–394
45. Villalonga C, Banos O, Khan WA, Ali T, Razzaq MA, Lee S, Pomares H, Rojas I (2015) High-level context inference for human behavior identification. In: International workshop on ambient assisted living. Springer, pp 164–175