Building Adaptive User Interface using Cloud Computing

Nikos Dimokas, Kostas Kalogirou, Pavlos Spanidis, Evangelos Bekiaris
Hellenic Institute of Transport
Centre for Research and Technology Hellas
Thessaloniki, Greece
{dimokas, kalogir, spanidis, abek}@certh.gr

Abstract—Recent advances in cloud computing and ubiquitous computing have changed the computing paradigm rapidly. Cloud computing is the computing that relies on a large pool of systems interconnected in private or public networks to deliver computing resources like services and data. Nowadays, there is no need of having local servers or personal devices to handle applications and data. Access to these resources for everyone and anywhere has become a necessity. Moreover, the evolution of ubiquitous computing combined with cloud computing implies the development of personalized services and applications considering the user preferences in order to provide an appropriate user interface anywhere anytime. However, the user interfaces so far are not conforming to the needs of people and especially those with disabilities or elderly. The paper introduces an accessible tool for declaring and managing user preferences that participate in personalization implementation in cloud computing environment. The proposed tool focuses mostly on elderly and people with disabilities and helps the users to build preference sets in order to improve access efficiency.

Keywords— cloud computing; ubiquitous computing; user preferences; user profile; personalization;

I. INTRODUCTION

Cloud computing [1], [2] is a recent trend in Information Technology (IT) that moves computing and data away from desktop and portable PCs into large data centers. It refers to applications delivered as services by using Internet technology. Especially, the cloud computing environment distributes IT resources – namely the hardware and software and allocates according to user’s request. Thus, computing is becoming more and more mobile and ubiquitous [3], [4], [5]. In order to fulfill the vision of ubiquitous computing [6], [7], [8] we have to support personalized services and applications. Personalized services and applications aim to assist the user in his/her daily life. In order to increase user acceptance of those services and applications, the personalization technologies have to consider the user needs and preferences.

The Global Public Inclusive Infrastructure (GPII) [9] architecture intends to enable the personalization on different device platforms (desktop, mobile, web, info kiosk, digital TV) and content based on the user's needs and preferences. Additionally, GPII provides a cloud computing infrastructure that implements the anywhere computing vision. The user starts the process by describing their needs and preferences to the system using the proposed tool. The tool is called Profile Management Tool (PMT) and it is a web application available on personal devices like PCs and smart phones where the user can set his/her needs and preference. Then, the system is able to automatically configure any device platform such as mobile device, desktop, or web with the user's preferred applications (e.g., a screen reader) and corresponding settings (e.g., their preferred voice speed). The content will be automatically adapted to suit the user's preferences as well. This process is taken place via the transformation mechanisms. Each setting (e.g. the font size) can vary between different type of devices and it has to be adapted according to each device platform characteristics.

The proposed tool implements the personalization according to user needs and preferences functionality. The user is able to define preference sets, share preferences with other users, store them in the cloud and finally activate user adaptations. Additionally, the user interface provided by the tool is designed and implemented in order to be accessible and meet the personal preferences and requirements of the users [11], [12], [13]. This interface adaptation ensures not only that the user interface is accessible to the user but also easy and comfortable to use. User interface adaptation relies on personally defined preferences, instead of pre-defined accessibility schemes for certain impairments, medical conditions, etc. The proposed tool (PMT) focuses mainly on elderly and people with disabilities (like visual impairments). However, it can be used by everyone.

The rest of the article is organized as follows: In Section II we describe the relevant work while in Section III we present the system architecture. Section IV describes the proposed tool and Section V exposes the experiments and obtained results according to well defined metrics. Finally, Section VI concludes the article.

II. RELEVANT WORK

Nowadays, users rely on much less sophisticated systems that consist of options that are changed and saved by the user. The interfaces for making adjustments are difficult to understand, hard to find, and not easily modified. Changes are

---

Research supported by CloudHall project [10] which is funded by the European Commission under the Grant Agreement 289016.
stored to a local profile and cannot be applied to other devices or contexts easily.

There are different approaches for the personalization of adaptive user interfaces and profile management. The European project GUIDE (Gentle User Interfaces for Elderly People) employs a guided tour with questions, tasks, and interactive games, in order to determine user skills and preferences. A similar approach is taken by the European project MyUI (Mainstreaming Accessibility through Synergistic User Modeling and Adaptability). Both projects aim to provide accessible user interfaces by system-initiated adaptations. GUIDE relies mainly on augmentations of an existing user interface by additional elements to improve the accessibility [14]. In contrast, MyUI generates individualized user interfaces at run time from an abstract description of the interactive application [15]. A main difference to the proposed approach is that both projects rely on a user profile, which describes user characteristics, i.e. capabilities and disabilities. GUIDE especially takes an approach to user profiling which has to be carried out within an explicit step of system initialization [16] and is very close to medical diagnosis. This categorization of users according to their limitations is not desirable, since it often generalizes about those with similar disabilities despite their unique differences, individual challenges, and personal preferences. Another big difference is that the proposed approach is not dynamically creating adaptive interfaces. However, it focuses on managing preference sets and launching assistive technologies that may produce an interface change.

The proposed user-centric approach attempts to empower the user to declare the best fitting options. A project that compliments this approach to preference set management is the Fluid Infusion Project [17]. The Fluid Infusion framework includes a toolkit of flexible, ready-to-use user interface components for setting preferences as well as the framework to build and support new, custom components.

A. Motivation and Contribution

The proposed approaches so far present various limitations. However, the proposed tool (PMT) improves declaring and managing user needs and preferences using cloud-based technologies. Conceptualizing tools that help the users define preferences and create preference sets is a novel approach that improves access efficiency. The whole concept of defining preference sets, storing them in the cloud, and then using them to activate user interface adaptations is an entirely innovative attempt. The proposed tool provides also advanced functions like sharing preferences among peers. Due to the number of possible interface adjustments, PMT offers an effective clustering of these adjustments helping users identify them. To our knowledge, there is not any comparable effort to apply auto-personalization from preferences (APfP) across such a broad range of products today.

III. SYSTEM ARCHITECTURE

The components of GPII architecture are written in JavaScript and hosted by the Node.js runtime environment. The main component of the GPII architecture is the flow manager. The flow manager is the most critical GPII element which controls the overall distribution between components and implements the configuration of the device by taking into account the user preferences, the available applications (e.g. screen reader) and access features, the information about the characteristics of the device and the user’s working environment. The flow manager transmits the user’s preferences to the Preference Management Tool (PMT) through JavaScript Object Notation (JSON) formatted file and PMT informs the local GPII version about the user’s needs and preferences. Initially, PMT was considered to be online tool. However, a stable internet connection is not an aspect that can be assured for every device. Furthermore some users have concerns about using and storing data in the cloud. To meet this need, a locally available solution of GPII has been developed. After receiving the JSON file, the tool parses those preferences and applies them to the application environment through local GPII module. Thus, the Auto-Personalization from Profile (APfP) is taken place to the device.

The communication between the Preference Management Tool and the GPII architecture could be realized in two ways (Fig. 1). According to the first approach, PMT communicates only with locally installed GPII version. This could be happen when the user wants to use a local instance of GPII. The usage of local instance is useful in case the device loses the internet connectivity. Regarding the second approach, the PMT is connected to the GPII remotely and then communicates with the locally GPII instance. The second approach assumes reliable internet connectivity. PMT communicates with server side GPII through internet in order to send and receive the JSON files that describe the user’s needs and preferences. After that, PMT sends the preferences to the local GPII instance in order assistive technology applications to be activated.

Fig. 1. System architecture

The Node.js module is integrated with GPII architecture and its flow manager module is responsible for handling the communication with the PMT. The communication between the PMT and the GPII architecture is taken place via AJAX calls that are being initiated by the PMT. The APfP process takes place transparently via GPII remote flow manager.
To achieve a better standardization across different platforms available for setting user needs and preferences, the preference management tool have been built according to the UI Options approach of the Fluid Infusion framework. Thus, PMT is running on top of Fluid Infusion framework. Infusion is a JavaScript based application framework built on top of JQuery for developing applications. The UI Options component is part of the PMT and allows users to customize the presentation of user interfaces by providing controls for adjusting preferences (adjusters).

IV. PREFERENCE MANAGEMENT TOOL

The proposed Preference Management Tool aims to mainly enable the declaration of preference sets and the initialization of the platform for managing preferences for different devices and contexts in a flexible way. The tool is a browser based application available on personal devices like PCs or smartphones. The PMT provides basic and some advanced functionalities which are the following:

- Create and edit the settings for the base and conditional preference sets
- Search for preference adjusters
- Save changes on preferences and adjusters
- Adjusters preview
- Define and share conditional preference settings (e.g. one specific preference set applied to a particular device or time)

Within the Cloud4All project [10], a requirements analysis and pilot tests with different end-users and stakeholders, as well as experts was conducted to collect the main requirements regarding preference management and automatic personalization of devices and user interfaces. According to the analysis of the results, the preferences have been categorized in five major categories. The categories include contrast, volume (Fig. 3), language (Fig. 3), increase size (Fig. 4) and finally visual alternatives (Fig. 5). Each category contains a set of preferences and controls for adjusting preferences. The preferences are contrast, universal volume, language, appearance, magnifier, speak text and Braille.

Fig. 2. Contrast preference

Fig. 3. Volume and language preferences

Fig. 4. Appearance and magnifier preferences
Each preference contains a set of adjusters in order the user to be able to set the preferred values. For example, Fig. 5 presents the available adjusters for the speak text preference. If the user activates the speak text preference, then there is a variety of adjusters available like voice pitch, volume (the volume of the application, not the system volume), words spoken per minute etc. Additionally, the previous figure depicts the text size and cursor size adjusters. The user is able to preview the changes in text or cursor size.

PMT provides also preview section in a few adjusters in order to show the user an example of what a preference adjusts. The proposed tool provides preview mode in those adjusters that can be visually represented and do not belong to third parties applications. In case of selected speak text for example the PMT activates a screen reader like NVDA or JAWS (the default screen reader for the device).

Each user will have exactly one base preference set, which contains the preferences that the Matchmaker will try to apply to any device anytime the user starts using. Then, on top of this base set, the user can define other time and device specific sets that are going to be applied under special starter conditions (e.g. between 22:00 and 01:00, on laptop). Preferences are always linked to certain conditions. So, the idea of the advanced approach is to provide a menu for condition selection to allow the user to more granularly define when a preference set should be applied. Fig. 6 depicts the dialog for setting a new conditional preference set. The user is able to set the title of the conditional preference set (e.g. morning) and define the device and the period of time. Regarding the devices option the user can select a computing device among a list of devices like desktop, laptop, tablet, phone, kiosk and bank machine.

So let's suppose a user wants to activate the screen reader during morning hours while the user wants to activate the magnifier during late hours in order not to bother neighbors. The user has to create two conditional preference set. The first set contains only the speak text activated with the desired values for the adjusters (e.g. volume 100) while the second set contains only the magnifier activated with the desired magnification level. The user will not have to activate any adjuster for the base set since the preferences are totally different for the two set. The user can set preferences in base set only in case the preferences, that are going to be activated, are independent of the device and the time period (e.g. the user wants always active the screen reader).

The interaction between the PMT and server side GPII is taken place when the user logs in to the tool. Additionally, the interaction between the tool and the local GPII version happens when either the user is disconnected from the tool by pressing the logout button or when the user presses the save button. The
login process of the user is quite simple and easy. When the user saves the preference sets for the first time, it receives a unique user token that can be saved in USB stick or as a QR code. After that, the user is able to login with his/her QR card or USB stick.

The storing and exchanging data is based on JSON format. JSON offers a simple, ubiquitous data interchange format. The proposed schema supports the definition of several preferences and the conditions that define the validity of those preferences. The following figure depicts a JSON example that represents an empty base set and a second preference set called morning that contains user’s preferences. The user wants to enable screen reader between 8 and 11 at the morning. Each preference is presented as an object. The object is an unordered set of name/value pairs. The name is identified by a unique URI (e.g. the following URI defines the speak text adjuster http://registry.gpii.net/common/screenReaderTTSEnabled) while the values can be numbers, strings, boolean values, arrays and objects.

```
"contexts": {
  "gpii-default": {
    "name": "Default preferences",
    "preferences": []
  },
  "morning": {
    "name": "morning",
    "preferences": {
      "http://registry.gpii.net/common/screenReaderTTSEnabled": true
    },
    "conditions": {
      "type": "http://registry.gpii.net/conditions/timeInRange",
      "from": "08:00",
      "to": "11:00",
      "inputPath": "http://registry.gpii.net/common/environment/temporal\_time"
    }
  }
}
```

Fig. 7. JSON format example

PMT has been implemented with HTML5, CSS3 and Infusion framework (especially UI Options library). The graphical components, icons of the UI are font icons in order to be scalable and managing them through CSS. Moreover, the whole tool is developed according to WAI-ARIA standards. Finally, specific acceptance tests [19], [19] have been written and integrated into the GPII core architecture framework. An acceptance test is a test that validates the entire (real-time framework) system end to end.

V. EVALUATION

We evaluated the usability of Preference Management Tool and the adaptations of the UI according to user needs and preferences. We conducted a large number of experiments with the participation of thirty elderly and vision impaired users. The users filled in questionnaires in order to provide valuable feedback about the usability of the tool and the UI adaptations. The evaluation process lasted approximately two hours with participants carrying out specific tasks based on pre-defined scenarios reflecting respective use cases. A user-centered approach was adopted early, therefore it was essential to gather feedback from potential end-user. Subjective evaluation was directed towards main usability elements according to ISO directives which are the following:

- Effectiveness
- Efficiency
- Satisfaction
- 30 (18 male and 12 female) elderly and vision impaired users participated in the trials.

The most representative results regarding effectiveness and usefulness are presented in Fig. 8 and Fig. 9. Fig. 8 depicts the effectiveness, while Fig. 9 depicts the usefulness of the proposed tool. In case of effectiveness around 70% of the users found the application effective. However, the perceived usefulness of PMT approaches almost 80%.

![Fig. 8. Perceived system effectiveness](image)

![Fig. 9. Perceived system usefulness](image)
VI. CONCLUSIONS

Anywhere and anytime computing combined with the cloud computing results in services and applications that require automatic personalization according to user needs and preferences. Additionally, the applications have to be accessible by elderly and people with disabilities. The proposed tool aims to provide an accessible user interface for building adaptive user interface that can be used in any device (smart phone, tablet, desktop etc) for various time periods. The user is able to declare and share preference sets, store them in the cloud and trigger the adaptations. The performance evaluation attests that the Preference Management Tool is accessible by elderly and vision impaired users.

REFERENCES


