Computer Standards & Interfaces 44 (2016) 177–184

Contents lists available at ScienceDirect

Computer Standards & Interfaces

journal homepage: www.elsevier.com/locate/csi

A cloud-based platform to develop context-aware mobile applications by domain experts

David Martín *, Carlos Lamsfus, Aurkene Alzua-Sorzabal

CICtourGUNE, Paseo Mikeletegi 71, Donostia-San Sebastián 20009, Spain

ARTICLE INFO

Article history:
Received 15 December 2014
Received in revised form 8 June 2015
Accepted 19 August 2015
Available online 29 August 2015

Keywords:
End-user development
Domain expert
Context-awareness
Mobile
Middleware

ABSTRACT

Context-awareness enables the personalization of computer systems according to the users' needs and their particular situation at a given time. The personalization capabilities are usually implemented by programmers due to the complex processes that are involved. However, an important trend in software development is that more and more software systems are being implemented not only by programmers but also by people with expertise in other domains. Since most of the existing context-aware development toolkits are designed for programmers, non-technical users cannot develop these kinds of systems. The design of tools to create context-aware systems by users that do not have programming skills but are experts in the domain where the system is going to be deployed, will contribute to speed up the adoption of these kinds of services by the society. This paper presents a cloud-based platform to ease the development of context-aware mobile applications by people without programming skills. The platform has been designed to be used in a tourism domain. This way, tourism experts can send tourist information to mobile users according to their context data (indoor/outdoor location, language, and date and time range). An energy-efficient mobile app has been developed in order to obtain context data from the user's device and to receive personalized information in real time based on these data. The architecture and implementation details of the system are presented and the evaluation of the platform by tourism domain experts is discussed.

1. Introduction

The recent advances in mobile technologies and their increasing popularity are speeding up the achievement of the ubiquitous computing vision [1]. For the first time ever there are more gadgets in the world than there are people [2]. Smartphone sales surpassed one billion units in 2014 [3] and it is expected that over half of mobile phone users globally will have smartphones in 2018 [4]. In addition to this, global mobile data traffic grew 69% in 2014 [5]. It is a fact that these devices are replacing cameras, music players and Global Positioning System (GPS) tools. Definitely, they have become an open window to information and services at any time and place. This computing scenario requires the development of new technologies in order to provide mobile users with the correct information and services on the move. This way, a personalization layer is needed in order to filter and adapt all the information that is consumed by mobile users.

In this sense, context data is essential in order to carry out this personalization process. In the last decade, location has been widely used as one of the key parameters in order to be able to personalize mobile applications. For instance, in the tourism domain [6] Location Based Services (LBS) [7] have been widely used in order to filter the results of mobile searches. In this manner, the information about the nearest points of interest can be automatically adapted, providing tourists with a better search experience. But there are other relevant context data, like the time and date, the user's device characteristics or the environment [8].

The development of context-aware mobile applications and services is usually carried out by experimented programmers that implement the personalization layer of the business logic according to the end-user requirements. But computer programming is becoming a widespread practice. Indeed, one important trend in software technology is that more and more applications and services are being written not only by professional software developers but also by people with expertise in other domains. End-User Development (EUD) [9] emerges as a key ingredient for ensuring the acceptability of context-aware systems by end-users and has accordingly been identified as a key future challenge for research in technological visions of ubiquitous and pervasive computing [10]. EUD can contribute to empower users as active citizens of the information society, improving the user experience and the effectiveness of context-aware systems.

Several middleware and programming frameworks have been proposed in order to simplify the development of context-aware

* Corresponding author. Tel.: +34 943 010885; fax: +34 943 010846.
E-mail address: davidmartin@tourgune.org (D. Martín).

http://dx.doi.org/10.1016/j.csi.2015.08.009
0920-5489/

applications and services [11]. However, most of them offer high level application programming interfaces (APIs) for skilled programmers. This makes the involvement of non-technical domain experts almost impossible. The design and configuration of context-aware applications by domain experts can speed up and improve the quality of these kinds of services, promoting the widespread adoption of context-awareness.

This work presents a cloud-based development platform in order to configure the behavior of context-aware mobile applications by people that do not have technical skills. More precisely, the design of the platform has been focused in the tourism domain, in order to provide tourism domain experts with the tools to create personalized tourism mobile services. The personalization process is based on the configuration of logical rules that trigger the delivery of tourist information based on the mobile users' context data (indoor/outdoor location, language, and date and time range). The platform provides a web front-end where all the features involved in the configuration of context-aware systems can be easily managed without coding any programming line.

The article is organized as follows: in Section 2, the related work is presented. In Section 3, the implemented context-aware platform is described. Section 4 analyzes the performance of the platform. Section 5 discusses the platform user evaluation. Finally, Section 6 presents the conclusions of this research work.

2. Related work

Context-aware computing has been widely studied in the last years. Several authors have stated different definitions for the concept of context. Some of these definitions consider context as the surroundings of the interaction between the user and the application [12]. Other authors consider the activity or the task of the user as the main context information for the system [13]. A third group of authors consider that context is the needed information to characterize the situation of an entity [14]. In this research work, context will be considered from a computing perspective, having into account the third group of definitions mentioned before.

Apart from context definitions, several authors have proposed development methodologies for the implementation of context-aware systems. Henricksen and Indulska [15] proposed a methodology and a modeling language called Context Modeling Language (CML). This language was developed for conceptual modeling of databases that store context data. It has different constructs for capturing the needed classes and context sources, the quality of context data and the dependencies and constraints between context fact types. This way, system designers can specify the requirements of context data needed by the system. The methodology is divided in five different stages: analysis, design, implementation, configuration of the system and validation. Context-oriented Programming (COP) [16] is a programming model that offers mechanisms to adapt the system to be implemented according to the gathered context data. These approaches are focused on system designers and programmers and do not involve users with no programming skills. In that way, domain experts cannot take part in the development process.

Regarding context-aware toolkits and frameworks, the Context Toolkit [14] is one of the first programming environments designed to ease the development of context-aware systems. This framework presents an architecture composed of different functional modules in order to acquire, aggregate and interpret context information. With this framework, programmers are able to acquire, aggregate and interpret context information. It uses key/value pairs to model context data. Other approaches like CASS [17] propose a layered middleware architecture that uses a relational data model to represent context data. JCAF [18] is a framework and a runtime environment to develop and deploy contextual computing applications. It uses an object oriented model to represent context data. These three approaches use interpreters to convert acquired raw data into higher level context data, but these transformations cannot be very complex because there is no inference mechanism. The CoBra [19] middleware proposes a different approach where software agents are used in order to acquire and process context data in a smart meeting room environment. SOCAM [20] and Semantic Spaces [21] are also frameworks based on three different layers, namely a sensing layer, a middleware layer and an application layer. Several similar approaches have been developed in recent years [22], but all of them are designed for programmers.

The first EUD environments to create applications for mobile devices have mainly targeted desktop and web environments. Examples of desktop EUD environments have been tourism [23] and home applications [24]. Other authors have proposed visual approaches where non-technical users can be involved in the development of context-aware systems. For instance, iCAP [25] allows end-users to design application prototypes by defining rules that are graphically edited through basic operations like dragging different elements onto rule sheets. eBlocks [26] facilitates the creation of customized sensor-based systems and the configuration of condition tables. The VS-CaSP system [27] provides end-users with a 3D environment to configure and design context-aware systems in smart buildings.

Apart from that, there are context-aware web platforms that can be configured using any web browser, like the OPEN [28] framework or Context Cloud [29]. The main drawback of these platforms is that the user still needs the involvement of programmers in the management of context data because some coding skills are required for the creation and configuration of context rules.

Other approaches are supported by mobile applications that can be personalized according to the user's context. For instance, Tasker [30] is an Android app that allows users to perform context-sensitive actions based on simple event-trigger rules. Locale [31] is another Android app that allows users to create situations specifying conditions under which the user's phone settings should change. However, these approaches are ad-hoc applications, so there is still a need for more general solutions, ready for wide use for developing context-aware mobile applications by end-users.

3. Description of the context-aware platform

The implemented prototype is a context-aware platform that enables people with no programming skills to configure the behavior of mobile applications based on context data. More precisely, the platform has been designed for tourism domain experts. This way, they can provide visitors on the move with relevant information according to their situation. This information is delivered automatically to the visitor's mobile device, based on some context parameters that can be configured with the web-based interface of the platform.

The context parameters that can be used to customize the delivery of information are the location and language of the mobile user, a date range and a time range. The location context parameter can be configured creating circular areas over a Google Maps layer, using the web interface of the platform. This way, the platform user can configure the condition of being inside or outside the created area. These areas can be indoors or outdoors. Also, the platform allows tourism domain experts to specify the information that will be sent to the mobile device of the visitor once the context conditions are fulfilled thanks to a WYSIWYG text editor. All these configurations can be carried out through the web interface, without any programming line.

The platform has been designed with three main technological and scientific trends in mind, such as the Web of Things, End-User Development and Cloud Computing.

The Web of Things (WoT) vision [32] is a web based approach for the Internet of Things. WoT is about re-using Web standards to