ADDIS ABABA UNIVERSITY

SCHOOL OF GRADUATE STUDIES

COLLEGE OF NATURAL SCIENCES

DEPARTMENT OF COMPUTER SCIENCE

MOBILE PHONE BASED PERVASIVE CONTEXT-AWARE EVENT ADVISOR (PCEA)

AYNALEM ABEBE NIGUSSIE

A THESIS SUBMITTED TO
THE SCHOOL OF GRADUATE STUDIES OF THE ADDIS ABABA UNIVERSITY
IN PARTIAL FULFILLMENT FOR THE DEGREE OF MASTERS OF SCIENCE IN
COMPUTER SCIENCE

FEBRUARY, 2015
ADDIS ABABA

ADDIS ABABA UNIVERSITY
SCHOOL OF GRADUATE STUDIES
COLLEGE OF NATURAL SCIENCES
DEPARTMENT OF COMPUTER SCIENCE

MOBILE PHONE BASED PERVASIVE CONTEXT-AWARE EVENT ADVISOR (PCEA)

Aynalem Abebe Nigussie

ADVISOR:
Dejene Ejigu (PhD)

APPROVED BY

EXAMINING BOARD:

1. Dr. Dejene Ejigu, Advisor _____
2. Dr. Dida Midekso, Examiner _____
3. Dr. Yaregal Assabie, Examiner_____

Dedication

To: My Mother

My Father

Acknowledgments

First and for most I would like to thank my advisor Dr. Dejene Ejigu for his continuous guidance and fatherly support throughout this project. And I really want to thank Addisu Yohanes and Letarik Terefe for the help they gave me. Finally, I would like to thank my family and friends for their support and love.

Table of Contents

LIST OF TABLES

LIST OF FIGURES

LIST OF ACRONYMS

AMS: Application Manager System

API: Application Programming Interface

BSC: Base Station Controller

BTS: Base Transceiver Station

GPRS: General Packet Radio Service

GPS: Global positioning system

GSM: Global System for Mobile Communication

HTML: Hypertext Markup Language

HTTP: Hypertext Transfer Protocol

IMDB: Internet Movie Database

ITU: International Telecommunication Union

J2ME: Java 2 Micro Edition

JAD: Java ME application descriptor

JSON: Java Script Object Notation

JSR: Java Specification Request

LAC: Location Area Identity

LBS: Location Based Service

MCC: Mobile Country Code

MNC: Mobile Network Code

MIDP: Mobile Information Device Profile

ORM: Object-Role Modeling

PCEA: Mobile Phone based Pervasive Context-aware Event Advisor

PDA: Personal Digital Assistant

PIM: Personal Information Management

POI: Point Of Interest

RDF: Resource Description Framework

RFID: Radio Frequency Identification

SOAP: Simple Object Access Protocol

SPARQL: Simple Protocol and RDF Query Language

UML: Unified Modeling Language

UMTS: Universal Mobile Telecommunications System

URI: Universal Resource Identifier

Wi-Fi: Wireless Fidelity

WWW: World Wide Web

XML: Extensible Markup Language

Abstract

The main aim of pervasive computing is minimizing distraction on users‟ attention and creating environment that adapt to the users‟ context and needs; thus the day to day development of mobile technologies makes this aim viable. There are a number of researches in pervasive computing that aim at improving our daily life. These researches cover a vast area of different topics. One topic of research is mobile based recommendation. Mobile based recommendations are done in different fields like tourism, medicine and the like. Even if recommendation service is not a new concept the way it is applied and the context it uses differ from one to the other. This project work proposed a client/server architecture for mobile phone based pervasive context aware event advisor (PCEA). The client side of the architecture is responsible for acquiring the user and the event context. Whereas the server side is responsible for processing the request from the mobile phone and come up with a response that match with the client‟s static as well as dynamic context. The proposed architecture utilizes the user‟s preference and dynamic contextual information as well as the dynamic context of the event information. In the context of this project we design our system architecture as three-tiered client-server architecture by combining data-centric and event based architecture style. We use a common context repository to store the user‟s preference, the dynamic contextual information and the event information context as well as a publish/subscribe way to broadcast the event notification message to the subscribed mobile devices. The system architecture has five core components that work together. The acquisition component gathers or collects static and dynamic information of the users as well as the event providers. The context manager component manages the static context i.e. the user profile and the dynamic context information i.e. the schedule of the user, the location of the user and the published and extracted event information. The user context and event information repository component store and manage the context data. The user context analyzer and matcher analyze both the user context and the event information and filter out matched result by considering the context parameter. The Notification manager component is responsible for delivering or pushing event information notification from the server to the users‟ mobile device. As a means of evaluating the proposed architecture we developed a prototype that implements the core components of the proposed architecture for java enabled phones using J2me and we performed a usability test to evaluate the prototype. The already established methodologies for usability testing for desktop environments are not always applicable to mobile application due to the mobility of the user and the restriction of the mobile devices such as the small size, battery and low resolution of the display. We conducted laboratory experiment and field study to

determine the UI of the prototype and the user satisfaction while accomplishing some tasks in the prototype. From the test we performed on the prototype, using the usability scale score, it achieved average value of 78.

Key Words: Pervasive Computing, Event, Context Awareness, Mobile Phone Based Pervasive Context-aware Event Advisor, J2ME

CHAPTER 1: INTRODUCTION

1.1 Background

The trend of computing evolves from the stage of „one computer to many people" into „many computers to one person", that is when the concept of pervasive computing becomes viable. Even if pervasive computing represents a major evolutionary step in a line of work dating back to the mid-1970[4], the concept of pervasive computing was first raised by Mark Weiser as a vision in his seminal paper in 1991. Mark Weiser, described his vision of ubiquitous computing, now also called pervasive computing as „„the most profound technologies are those that disappear. They weave themselves into the fabric of everyday life until they are indistinguishable from it"" [14].The essence of that vision was the creation of environments saturated with computing and communication capability, yet gracefully integrated with human users. When articulated, this was a vision too far ahead of its time, the hardware technology needed to achieve it simply did not exist but after a decade of hardware progress, the evolution in the hardware technology resulted in viable commercial products: hand held and wearable computers; wireless LANs; and devices to sense and control appliances, that led to the realization of Mark Weiser"s vision.

Pervasive computing is characterized by context awareness, invisibility, Ad-hoc network and smart spaces and devices. These characteristics distinguish it from the two earlier distinct computing systems that are distributed computing and mobile computing.The first characteristics of pervasive computing, context-awareness, is the ability of a given system to understand the context of an entity where the entity can be a person or any other thing, and a mobile computing paradigm in which applications can discover and take advantage of contextual information (such as user location, time of day, nearby people and devices, and user activity) is called context-aware computing [5].

But first, what is context? Context is defined differently in different area of disciplines. The Oxford dictionary generally defines context as the circumstances that form the setting for an event, statement, or idea and in terms of which it can be fully understood, but not satisfied by the general definition, many researchers have attempted to define context by enumerating examples of contexts, Schilt tries to define the word context by dividing it into three categories, i.e. computing context, user context and physical context [12]. Guanling Chen and David Kotz [13] also try to define context as the set of environmental states and settings that either determines an application"s behavior or in which an application event occurs and is interesting to the user. One of the most accurate definitions considered by us is given by

Deyand Abowd [10]. These authors refer to context as “any information that can be used to characterize the situation of entities (i.e. whether a person, place or object) that are considered relevant to the interaction between a user and an application, including the user and the application themselves”.

A user‟s context can be quite rich, consisting of attributes such as physical location, physiological state, emotional state, personal history, daily behavioral patterns, and so on.

The other characteristic of pervasive computing is invisibility that is more related to the definition of minimum user distraction. A pervasive computing system that strives to be minimally intrusive has to be context-aware. In other words, it must be cognizant of its user‟s state and surroundings, and must modify its behavior based on this information [4]. The third characteristic is an Ad-hoc network which is a self-configuring (wireless) network of mobile nodes without the presence of static infrastructure and the fourth characteristic is smart space and devices i.e. a space that is created by embedding computing devices and sensors into a building infrastructure in order to detect contextual information.

By incorporating the four characteristics mentioned above, pervasive computing strives to fulfill its main aim i.e. minimizing distraction on users‟ attention and creating environment that adapts to the users context and needs since nowadaysmost precious resource in a computer system is human attention rather than its processor, memory, disk or network.

Applications allowing users to collaborate in real time using wirelessly connected mobile devices building ad-hoc networks have attracted the attention of many authors [2]. Recent researches show that mobile technologies have become part of the everyday life of most people around the world. According to the International Telecommunication Union (ITU), the number of cell phone subscribers has reached five billion during 2010, while mobile broadband subscriptions have exceeded one billion globally [1]. Mobile phones technology is becoming advanced; recently developed mobile phones provide users with a number of features such as Wi-Fi connectivity, Bluetooth and GPS localization, camera and video capture devices and, most interestingly, the capacity for users to program the mobile devices with additional applications. Having all these features made the purpose of the mobile device grow beyond accepting and making calls or exchanging messages, and also unlike the old users, the current mobile device users want to be able to access and manipulate information and service specific to their location, time and environment [7].

Due to the above reasons it is believed that developing service provider application by using mobile devices that most of the time in separable from the user, is one way for realization of

pervasive computing. As a computing platform, mobile phones are both pervasive and personal. They are almost always on and tend to have an intimate relationship with their owners, who store private information on them and often personalize their appearance or ring tones, for instance. On one hand, mobile phones follow the user and have clues about the current situation. On the other hand, their various usage contexts will likely benefit from context awareness [3].This personal nature suggests that mobile phones are well suited for context- aware computing.

Event is defined in Meriam-Webster dictionary as something that happens or occurs, a noteworthy happening or a social occasion. In Oxford dictionary, it is also defined as a thing that happens or takes place, especially one of importance, a planned public or social occasion or each of several particular contests making up a sports competition. In our context events are not only constrained to public or social occasions or big events like sport competition rather it includes information that the user is interested in, specifically the entertainment areas and the point of interests. Here, we try to take advantage on the fact that information is flooding our world. However the world is full of information, the main question here is "is all the information we get is what exactly we require? Did we never miss what"s important for us?" by taking the above question in to consideration, we proposed a mobile phone based pervasive context event advisor holding the main aim of filling this information gap and satisfy the user by providing the needed information anywhere any time by using mobile phones and wireless technology according to the user pre-subscribed information or interest and user agenda, time and current location. The application collaborates the information in hand with the information it perceived from the mobile and the web to notify the user about different events according to his/her interest.

1.2 Motivation

Technologies are now all about creating a suitable and comfortable living environment. Their main goal is helping and improving the life style of a human being. One way of fulfilling this is by providing needed services near and available to people. Providing event information for the user does not seem very important as a real issue like hospital case or tourism or the like but it is very essential in satisfying the people who work in the tourism or hospital area, since if people are satisfied on things they get, everything they do after that is pleasing and pleasant. This made an event a prerequisite for determining the next action of the user. In other word, when people are acting and living in the way they like, their life will be interesting and this will have a great impact on their emotional state which in turn has a huge

impact on their social, economic and other sectors of theirs life in general. The other drive to our motivation is in our country Ethiopia currently there is no well-organized information provider particularly in the entertainment area and even if few providers exist, the information they provide is not current and updated frequently besides the media they use to broadcast the information is not accessed by the mass. Above all these users need to invest substantial amount of time in searching for this kind of information and rarely if they are lucky they get what they are looking for else most of the time end up empty handed. The following scenario will illustrate this concept

Scenario

Mr. y is a big fan of art, who likes to visit galleries and like to participate on whatever events that is held about art, when he is not busy. But due to the fact that this kind of information is not released in a website or other social networks he can‟t get the information in time. Because of this he always misses the event and is always disappointed about it.

The context and event information in this scenario are

Context - schedule, time, user preference or interest, location

Event- opening of Art Gallery, nearby restaurant

In this scenario we observe that there is no well-organized information provider in current status of our country and if there is, the information will not be updated frequently every time when an event held, this also include the user investing substantial time for searching this kind of information. The other main issue is that the recommendation should not only be about recommending Mr. y about art gallery rather it is more of advising the user by recommending the opening of the art gallery according to his preference or pre-subscribed interest and its schedule i.e. if he is free or not in that specific day and time i.e. minimum user distraction. And the other issue is by considering Mr. y participating in the event, if the event is finished at 6:30 AM, the system should consider the time is lunch time and having the user interest of food from the pre-subscribed information and advice the user about suitable restaurants according to the current location of the user as well the menu of the restaurant.

So we engaged full heartedly in this research to create an emotionally satisfied society as well as to move the current contextual information acquisition or timely distribution process in our country one step ahead in a way that minimizes the wastage of users‟ valuable resource that is there attention as well as their time.

1.3 Statement of the Problem

Nowadays, we people are leading a complicated, busy and restless life. Our perception of life is becoming all about work and other related stuff. We forgot that our emotional state matters for our productivity. Attending and participating in different interesting events have a lot of significances for an individual in particular, as well for the society in general. The main issue is how to provide event information for a user without stealing the user‟s attention and the other issue is how a user gets this kind of information without investing his/her substantial time in searching. In order to solve this problem, a technological concept like providing event information for the user by using mobile phones is a best solution. Recommendation domain is applied in different fields like tourism, medicine, entertainment and the so like. In each fields many researchers try to come up with different recommender systems by considering different contextual parameters and the computing platform. In the entertainment areas there are plenty recommender systems that cover the area of movies, music‟s and the so like. As we mentioned above different use of contextual parameters and the source of the information we use specifically by considering the case of Ethiopia differentiate our system from the rest. This study tries to answer the following questions:

- How can the different existing frameworks be made suitable for the case of event advisory system?
- How the different context information like schedule, location, user context/ interest and time can be integrated and come up with better contextual information for event advisory system?
- How architecture of a system that support a context acquisition, management and processing towards event advising can be designed?
- How to avail a context aware event notification without distracting the user while capturing their interest and their day to day activity by being aware of the user personal information and location?
- How resource sharing process between users can be applied for event advisory application?

1.4 Objective

General objective

The general objective of this research work is to design and develop a mobile phone based pervasive context-aware event advisory system.

Specific objectives

The specific objectives of this study include:-

- Study and set context parameters required for event advising.
- Design a method for acquiring and adapting the user context in automatic way
- Design an architecture of a system that supports a context acquisition, management and processing towards event advising.
- Develop prototype of a context aware event advisory mobile application system both client and server side.
- Perform a usability test on the developed prototype

1.5 Scope and limitation

Implementing a context-aware system requires many issues to be addressed. It has its own challenges. A key challenge in pervasive computing environment is obtaining the information needed to function in a context manner [5]. Besides this, the zero availability of pervasive infrastructure and smart spaces in our country has a limitation in realizing the event advisor system in a broader way. The limited network connection availability in mobile phones also affects the output. Having this limitations in mind the scope of these research work are:

- Develop a context aware event advisory mobile application for java enabled mobile phones.
- The event information that will be included is limited by the companies that are willing to share their information or database.
- It does not support automated migration.

1.6 Methodology

In researching for modeling mobile based context aware event advisory, methodologies we will use are

- **Literature Review and Related Works:**
  In the course of this study, a number of published research papers; complete research project works; books and web sites in the area of pervasive computing in general and in the area of context-aware mobile application and event advisory systems in particular will be examined.
- **Questionnaire**: we used questioner to test the usability of the system. We adapt an SUS (System Usability Score) testing questionnaire (Annex C) to analyze the user satisfaction. The participants answer the test post questioner after finishing their tasks.

- **Tools and technologies:** Exploring the required tools and technology to avail the mobile phone based pervasive context-aware event advisor, the tools and technology we have used in the course of this research will be presented in Chapter 6 of this documentation.
- **UML (Unified Modeling Language):** we use a UML methodology to specify and analyze the requirement of the proposed system and to show the system design.
- **Prototyping:** To validate the proposed architecture suggested in this work, we used a prototype as a means of proving the concepts.
- **Validation:** we perform a usability test to identify the application we prototype is user friendly or not and to address the several questions that is needed for designing the application.

1.7Significance of the study

As event advisory is a context aware system, it will be a realisation of the pervasive computing environment that provides services to a user with minimum user distraction. Event advisory will have great contribution for the user by saving substantial amount of time and mostly it will save their precious resource that is their attention.

This research work will provide a number of applications for the community and individuals. Some of the applications are:-

- Provide event information in an organized manner at one place
- Eliminate wastage of user time in search of event information
- Help to get contextually aware service
- Facilitate sharing of information between peer users
- Help to create emotionally satisfied society
- Create easy living situation
- Create a means of providing selective, only required information
- Help to create an opportunity for people to meet or socialize

1.8 Document organization

The remaining part of this documentation is organized as follows. The next Chapter deals with literature review. The literature review part will try to give a summarized report on the state of art in relation to pervasive computing and context awareness in general. Chapter 3 is about related work that is highly relevant and more related to our study. Chapter 4 will discuss the system requirement; Chapter 5will discus the system design and the architecture

in order to build the mobile phone based pervasive context aware event advisor system. Chapter 6 explains the implementation issues of the system. The last Chapter will conclude the paper by indicating future works.

CHAPTER 2: LITREATURE REVIEW

2.1 Pervasive Computing

Computer technologies pass in different evolution in the computing paradigm. The evolution starts from the basic centralized system, to today‟s pervasive computing system. The two distinct earlier steps in the computing evolution, which contribute a lot for the pervasive computing, are distributed systems and mobile computing. Satyanarayanan [4] specified that, the field of distributed systems arose at the intersection of personal computers and local area networks. In distributed computing, the results achieved are remote communication, fault tolerance, high availability, remote information access and security in a distributed system but the appearance of full-function laptop computers and wireless LANs in the early 1990s led researchers to confront the problems that arise in building a distributed system with mobile clients. Although many basic principles of distributed system design continued to apply, four key constraints of mobility forced the development of specialized techniques. These constraints are: unpredictable variation in network quality, lowered trust and robustness of mobile elements, limitations on local resources imposed by weight and size constraints, and concern for battery power consumption [24]. Mobile computing tries to elevate the above problems and it achieves results on the following broad areas, mobile networking, mobile information access, bandwidth-adaptive file access, support for adaptive applications, system-level energy saving techniques, and location sensitivity.

Even if there was a great achievement in these both paradigms, there was more, the concept of anywhere anytime, that‟s when the paradigm of pervasive computing come to the picture. The paradigm of pervasive computing aims to integrate the computing technologies in a graceful and transparent manner, and makes computing solutions available anywhere and at any time. Different aspects of pervasive computing, like smart homes, smart offices, social networks, micromarketing applications, PDAs, etc. are becoming a part of everyday life [56].

The research agenda of pervasive computing subsumes that of distributed and mobile computing, but goes much further. Specifically, pervasive computing incorporates four additional research thrusts into its agenda. These are:

- *Effective Use of Smart Spaces*: that is, by embedding computing infrastructure in building infrastructure to bring together two worlds that have been disjoint until now [23].
- *Invisibility:* The ideal expressed by Weiser is complete disappearance of pervasive computing technology from a user‟s consciousness. In practice, a reasonable

approximation to this ideal is minimal user distraction. If a pervasive computing environment continuously meets user expectations and rarely presents him/her with surprises, it allows him/her to interact almost at a subconscious level [22].

- *Localized Scalability:* The density of interactions has to fall off as one moves away, otherwise both the user and his computing system will be overwhelmed by distant interactions that are of little relevance.
- *Masking Uneven Conditioning* - reducing the amount of variation seen by a user.

As pervasive computing is usually characterized by invisible components that can be embedded around different environments, it is sometimes referred to as "Anywhere Anytime Computing" or "Ubiquitous Computing". The ubiquitous computing is characterized by the omnipresent and mobile availability of services themselves regardless of the target platform. Services will be tailored to the physical capacity of a specific device, whether a mobile telephone, PDA or other value-added communication devices. Ubiquitous computing paradigm implemented in various areas, smart homes (see Figure 2.1, pictures from smart apartment from Technical University of Berlin), smart offices and other ambient intelligence solutions, wearable computing devices, personal digital assistants, social networks, mobile devices as gateways to the Internet [56].

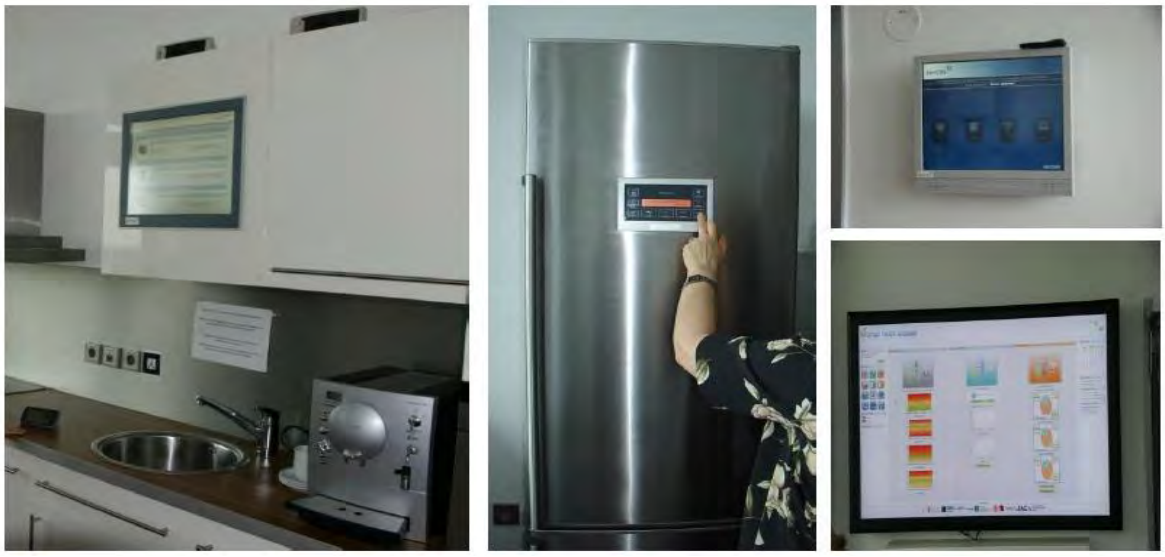


Figure 2.1 Smart home environments Kitchen, Fridge, control panel. Picture taken at the smart apartment at DAI-Labor, Technical University of Berlin [adapted from 56]

Although often used as synonyms, the term "pervasive computing" is often preferred when discussing the integration of computing devices and weaving them into the everyday life and

business environment, while the term “ubiquitous computing” is usually preferred when addressing the interfaces and graceful interaction with the user [56].

The paradigm of pervasive computing pursues two main goals:

1. Graceful integration of computing technologies into everyday life and
2. High availability – the computing services should be available everywhere and at any time.

In pervasive computing, context-awareness, ad-hoc networks and smart sensor networks perform together to serve people on their surrounding environment. The design of pervasive computing system should try to attain the goal of simplicity, versatility and pleasurable while retaining cost-effectiveness, usage-safety and transparency. In order to achieve these pervasive goals, the appliances and the networks comprising of these appliance nodes must be able to automatically discover other devices, services and parameters. In addition, they should be able to carry out unattended negotiation amongst themselves if needed.

2.2 Context and Context-Awareness

One field in the wide range of pervasive computing is the so-called context-aware (or sentient) systems. The word „context“ is a challenging task to define and many researchers tried to find their own definition for what context actually include. Among the many definitions one of the best typical definitions is given by Dey and Abowd [10]. They refer context as “any information that can be used to characterize the situation of entities (i.e. whether a person, place or object) that are considered relevant to the interaction between a user and an application, including the user and the application themselves, and by extension, the environment the user and applications are embedded in. Context is the set of the associated situation and actions characterizing the physical surrounding of a device and captured by a sensor of a device or the infrastructure. In human-computer interaction context is also a powerful, and longstanding, concept. It can be used to interpret explicit acts, making communication much more efficient [26]. By carefully embedding computing into the context of our lived activities, it can serve us with minimal effort on our part. Communication can be not only effortless, but also naturally fit in with our ongoing activities. Pushing this further, the actions we take are not even felt to be communication acts at all; rather, we just engaged in normal activities; and the computation becomes invisible [27].

A system is context-aware if it uses context to provide relevant information and/or services to the user, where relevancy depends on the user‟s task [10]. More and Dourish [26] also mentioned the one goal of context-aware computing is to acquire and utilize information

about the context of a device to provide services that are appropriate to the particular people, place, time, events, etc. Context-aware systems are able to adapt their operations to the current context without explicit user intervention and thus aim at increasing usability and effectiveness by taking environmental context into account.

When dealing with context, three entities can be distinguished: *places* (rooms, buildings etc.), *people* (individuals, groups) and *things* (physical objects, computer components etc.). Each of these entities may be described by various attributes summarized to three main categories: *identity* (each entity has a unique identifier), *location* (an entity‟s position, co-location, proximity etc.), *status* (or activity, meaning the intrinsic properties of an entity, e.g. temperature and lightning for a room, processes running currently on a device etc.) [30]. One popular way to classify context instances is the distinction of different context dimensions. These dimensions are called „external‟ and „internal‟, refers to „physical‟ and „logical‟ context respectively. The „external‟ or physical dimension means context that can be measured by hardware sensors/dynamic sensors, i.e. location, light, sound, movement touch, temperature, air pressure etc. Whereas the „internal‟ or logical dimension is mostly specified by the user or captured by monitoring the user‟s interaction, i.e. the user‟s goals, tasks, work context, business processes, the user‟s emotional state etc. [30]. By sensing context information, context enabled applications can present context information for users, or modify their behavior according to changes in the environment.

2.3 Mobile phone based context- aware application

In the history of computation, the development of hardware devices and wireless technology is a major foundation for realizing pervasive computing vision. Computation is now packaged in a variety of devices. Smaller and lighter laptop/notebooks, as powerful as conventional personal computers, free us from the confines of the single desk. Specialized devices such as handheld personal organizers are portable enough to be with us all the time. Wireless technology allows devices to be fully interconnected with the electronic world. Cell phones are really networked computers. Over the past decade, there has been a widespread adoption of mobile phones and personal digital assistants (PDAs) all over the world. Economies of scale both for the devices and the supporting infrastructure have enabled billions of mobile devices to become affordable and accessible to large groups of users [36].

Since its introduction in the mid-1980s, the sophistication of mobile devices in terms of the numbers and types of services they can provide has increased many times over. The

improvements to mobile devices and back-end infrastructure has allowed for additional information to be used as input to mobile devices and services. In particular, context, or information about the user, the user‟s environment and the device‟s context of use, can be leveraged to expand the level of input to mobile devices and support more efficient interaction with a mobile device.

The use of context in mobile devices is receiving considerable attention in various fields of research including mobile computing, wearable computing, augmented reality, ubiquitous computing and human-computer interaction. A general motivation is that context-awareness can serve to compensate for the abstraction that is required in the first place to make systems accessible in changing environments and situations. Context is particularly relevant in mobile computing. When users are mobile, their context of use changes much more rapidly than when they are stationary and tied to a desktop computing platform. For example, as people move, their location changes, the devices and people they interact with changes more frequently, and their goals and needs change. The actual utility of context-awareness in mobile systems has been demonstrated in a wide range of application examples, in obvious domains such as fieldwork [37, 39] and tourism [40, 41, 38], as well as in emerging areas like affective computing based on bio-sensing [42, 43]. Also, it has been shown, that context is useful at different levels within a mobile device. At systems level, it can be exploited for example for context-sensitive resource and power management. At application level, context-awareness enables both adaptive applications and explicitly context based services. And at the user interface level, the use of context facilitates a shift from explicit to implicit human-computer interaction, toward less visible if not invisible user interfaces [44, 45].

Generally mobile context-aware applications are those that run on wireless devices e.g., mobile phones, PDAs, and have an awareness of the physical and social situation in which they are deployed.

2.4 Context-Awareness, its challenge and suggested ground rules

Context-awareness is fine in theory. One of the huge problems in building a new context aware application is start building from the scratch, no reusable support for building. Dey and Häkkilä in their paper specifically mentioned particular problems that the developers faced. Those are [29]:

- Context often comes from non-traditional devices that developers have little experience with, unlike the mouse and keyboard.

- Raw sensor data is often not directly useful to an application, so the data must be abstracted to turn it into useful context.
- Context comes from multiple distributed and heterogeneous sources, and this context often needs to be combined (or fused) to be useful. This process often results in uncertainty that needs to be handled by the application.
- Context is, by its very nature, dynamic, and changes to it must be detected in real time and applications must adjust to these constant changes in order to provide a positive user experience to users.
- Usability risks for mobile context aware application.

Dey and Häkkilä also mentioned the uncertainties in context recognition, information overflow, lack of user control, application complexity, privacy violation, subjective understanding of context attributes, lack of commonly agreed ontologies, imbalance between automated and user initiated actions and poor interoperability as source of the usability risk. By pointing out the above problems they point out 8 design guidelines in order to avoid these negative design consequences and minimize usability risk for mobile context aware application and provide values that context aware application offer to end users. Those ground lines are:

GL1: *Select appropriate level of automation* – in practice, features such as automation level or level of proactivity may be designed differently if the confidence level of context recognition can be estimated correctly. The greater the uncertainty is in context recognition, the more important it is not to automate the action.

$$uncertainity \sim \frac{1}{automation \;\; level} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (1)$$

The automation level has large impact on the number of expected interruption the system creates for the user.

$$automation\ level \sim user\ control \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (2)$$

GL2: *Ensure user control* – the user must be able to take control of the device and context-aware application at any time, especially when the device is performing erroneous action and the user wants to take a corresponding action and when the user just wishes to feel in control. User control can be implemented for e.g. with confirmation dialogues, however, this must be balanced with the need to minimize unnecessary interruption.

GL3: *Avoid unnecessary interruption* – if the system thinks that the interruption will provide high value or benefit to the user, allowing the interruption is often seen as positive for e.g.

reminders, alarm clocks. But these interruptions depend on the user context and user threshold for putting up with intrusion.

GL4: *Avoid information overflow* – in order to address the usability risk of *information overflow* where several different tasks or events compete for this channel, a priority ordering needs to be defined.

Systems should not present too much information at once, and should implement filtering techniques for to avoid messages that may appear to be spam to users. Also, information should be arranged in a meaningful manner to maintain and maximize the understandability of the system.

GL5: *Appropriate visibility level of system status* – The visibility level of what the system is doing has to be sufficient for the user to be aware of the application‟s actions.

When uncertainty in context awareness is involved, there must be greater visibility of system state in order to allow the user to recognize the risk level and possible malfunction.

System status need not be overwhelming and interrupting to the user but can be provided in an ambient or peripheral fashion, information is dynamically made more visible as the importance value grows and may eventually lead to an interruption event to the user if its value is high enough.

GL6: *personalization for individual needs*– Context-awareness should allow a device or application to respond better to the individual user‟s personal needs e.g. filtering of interruptions according to the user‟s personal preference. It may also be used to improve the subjective understanding of context attribute.

Allowing the user to name or change context attributes, such as location name or temperature limits, contribute to user satisfaction and ease of use.

GL7: *Secure users privacy* – Privacy is a central theme with personal devices, especially with devices focused on supporting personal communication, and impacts, for example trust, frequency of use, and application acceptability. Special care should be taken with applications that employ context sharing.

If necessary users should have the ability to easily specify that they wish to remain anonymous with no context shared with no other entities.

GL8: *Take into account the impact of social context*- The social impact of a context-aware application taking an action must be part of the consideration in deciding whether to take the action or not. In some social contexts, certain device or user behavior may be considered

awkward or even unacceptable. In such situations, there must be an appropriate balance of user-initiated and system-initiated actions.

Having the above problem and ground lines in mind many researchers try to come up with some reusable architectures, middleware‟s and frameworks for designing and developing context aware application.

2.5 Architecture, Middleware and Frameworks

2.5.1 Architecture

Pervasive computing aims to provide proactively adapted services to both user and applications according to the global context. As mentioned above, the main characteristic of devices in such system is their context awareness. Since its apparition, pervasive computing has required tools (architectures, frameworks and middleware), methods and concepts to support the development of a context-aware system and ease their design and implementation [57].

Different approaches have been taken to provide a common architecture for context-aware application and the structuring of functionality of context –reasoning mechanism.

Winograd [30], based on considerations of special requirements and conditions like the location of sensors (local or remote), the amount of possible users (one user or many), the available resources of the used devices (high-end-PCs or small mobile devices) or the facility of a further extension of the system, three different approaches of context aware system architectures can be distinguished. Those are

- *Direct sensor access*: used in devices with sensors locally built in. The client software gathers the desired information directly from these sensors, i.e. there is no additional layer for gaining and processing sensor data. Drivers for the sensors are hardwired into the application, so this tight coupled method is usable only in rare cases as it complicates extensibility. Also it is not suited for distributed systems due to its direct access nature without any component capable of managing multiple concurrent sensor accesses.
- *Middleware based:* introduces a layered architecture to context-aware systems with the intention of hiding low-level sensing details. Compared to direct sensor access this technique eases extensibility since the client code has not be modified anymore and it simplifies the reusability of hardware dependent sensing code due to the strict encapsulation.

- *Context server:* this distributed approach extends the middleware based architecture by introducing an access managing remote component. Gathering sensor data is moved to this so called context server to facilitate concurrent multiple access. Beside the reuse of sensors the usage of a context server has the advantage of relieving clients of resource intensive operations. As probably the majority of end devices used in context-aware systems are mobile gadgets with limitations in computation power, disk space etc. this is an important aspect.

Winograd [30] in his paper also describes three different context management models for coordinating multiple processes and components.

- *Widgets:* Derived from the homonymous GUI elements a widget is a software component that provides a public interface for a hardware sensor [29]. Widget hide low level details of sensing and ease application development due to their reusability. Because of the encapsulation in widgets it is possible to exchange widgets which provide the same kind of context data (e.g. exchange a radio frequency widget by a camera widget to collect location data). Widgets are usually controlled by some kind of a widget manager. The tight coupled widget approach increases efficiency but is not robust to component failures.
- *Networked services:* This more flexible approach resembles the context server architecture. Instead of a global widget, manager discovery techniques are used to find networked services. This service based approach is not as efficient as widget architecture due to complex network based components but provides robustness.
- *Blackboard model:* In contrast to the process-centric view of the widget and the service-oriented model, the blackboard model represents a data-centric view. In this asymmetric approach processes post messages to a shared media, the so called blackboard, and subscribe to it to be notified when some specified event occurs. Advantages of this model are the simplicity of adding new context sources and the easy configuration. Unfavorable is the need of a centralized server to host the blackboard and the lacks.

But according to Moran and Dourish, current research focuses on either different version of a black board –based approach, or on a widget –based approaches. Usually these approaches are implemented in the form of middleware, or in the form of application framework [26].

2.5.2 Middleware

The composition of context-aware systems is characterized by a high degree of heterogeneity. However, all approaches agree in decoupling context capturing and context processing from application composition by encapsulating the context management logic into middleware‟s [46].

Middleware is software that supports mediation or communication between other components across heterogeneous platform and varying resources. Middleware addresses two broad characteristics of pervasive computing: the trade-off between awareness and transparency and cooperation between development support and runtime services. However, it must address specific characteristics as well. Generally, the difference between just good and widely successful middleware is how easily it lets the application software developers exploit its various capabilities. In light of these broad characteristics, middleware for pervasive computing should also demonstrate specific characteristics to facilitate pervasive computing applications [34]. These are uniform development support, application-specific context acquisition, analysis, and detection, context-triggered action and transparent support for ad hoc communication.

Some example of middleware approaches include:

A. Gaia

Gaia is a middleware infrastructure that extends typical operating system concepts to include context-awareness. It aims at supporting the development and execution of portable applications for active spaces. Gaia exports services to query and utilize existing resources, to access and use current context, and provides a framework to develop user-centric, resource-aware, multi-device, context sensitive and mobile applications [30].

Gaia is a distributed context middleware that coordinates software entities and network devices in a physical space. By defining active space, Gaia supplies services to manage context of entities inside the space. However, most context entities involved in active spaces are static. Though services related to a moving person have been developed, no clear and detailed description of data management for mobile entities has been given [6].

B. SOCAM (Service-Oriented Context-Aware Middleware)

SOCAM uses a central server called context interpreter, which gains context data through distributed context providers and offers it in mostly processed form to the clients. The context-aware mobile services are located on top of the architecture: they make use of the different levels of context and adapt their behavior according to the current context [30].

The SOCAM architecture aims to provide an efficient infrastructure support for building context aware services in pervasive computing environments. It is a distributed middleware that converts various physical spaces from which contexts are acquired into a semantic space where contexts can be shared and accessed by context aware services [35]. It consists of the following components, which act as independent service components, see Figure 2.2:

- *Context providers:* They abstract useful contexts from heterogeneous sources, External or Internal; and convert them to OWL representations so that contexts can be shared and reused by other service components.
- *Context interpreter:* It provides logic reasoning services to process context information.

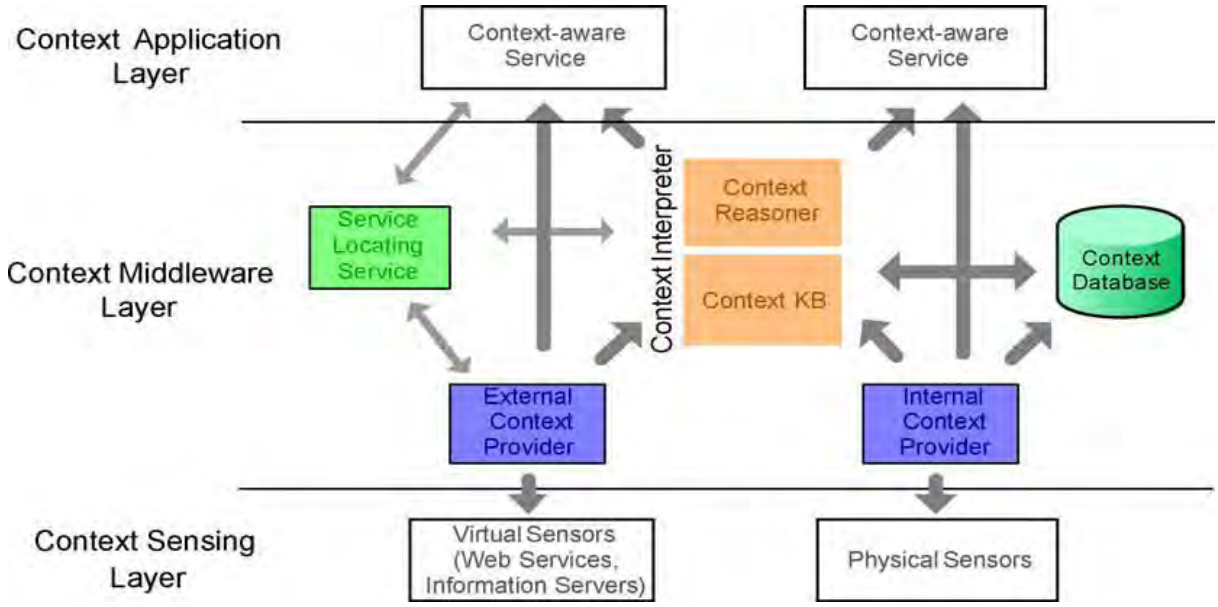


Figure 0-1.2: Overview of the SOCAM architecture [adapted from 35]

- *Context database:* It stores context ontology and past contexts for a sub-domain. There is one logic context database in each domain, i.e. home domain.
- *Context aware services:* They make use of different level of contexts and adapt the way they behave according to the current context.
- *Service-locating service:* It provides a mechanism where context providers and the context interpreter can advertise their presence; it also enables users or applications to locate these services.

C. Reconfigurable Context- Sensitive Middleware for Pervasive Computing (RCSM)

RCSM is a middleware designed to facilitate applications that require context awareness or spontaneous and ad hoc communication. It considers all three types of application i.e. context sensitivity, ad hoc communication and context – sensitive ad hoc communication to be equally important for pervasive computing environments.

RCSM provides an object-based framework for supporting context-sensitive applications. Taking an object-based approach in RCSM presents additional leverage beyond the benefits that simple object orientation provides. RCSM models context-sensitive application software as context-sensitive objects, which consist of two parts: a context- sensitive interface and a context independent implementation. The interface encapsulates the description of the application‟s context awareness, whereas the implementation remains contexts free [34].

D. Coalition

Coalition is a platform for context aware mobile application development that is capable of locating and extracting relevant context data from large number heterogeneous data sources distributed over many different operating environments through its context data management layer [6].

The middleware architecture consists of four logical layers [32]. These are

- *Physical space layer:* represents the various context data source. Physical space is an operating environment that provides context data from its attached entities such as sensors, actuators and computing devices.
- *Context data management layer:* to efficiently manage physical space and support context data look up.
- *Service management layer:* discovery of system services or third party services.
- *Application layer:* interact with the middleware services to retrieve contextual information, use third party services or orchestrate services to fulfil tasks requiring the collaboration of multiple services.

2.5.3 Frameworks

To really simplify the developing of context-aware applications, an abstract framework is needed. Such a generic infrastructure not only provides client access to retrieve context data, it also permits the simple registration of new distributed heterogeneous data sources [12].

Some applications frameworks that have been proposed are:

A. Context toolkit

The context toolkit was proposed as a tool to help the developers of context-aware systems. It has a layered architecture that permits the separation of context acquisition, representation and adaptation process [57]. Dey and Häkkilä [29] in their work context toolkit they used a number of abstractions to ease the building of applications. The Context Toolkit aims at facilitating the development and deployment of context aware applications. The context information reflects an application's operating environment that can be sensed by the application. The Context Toolkit consists of context widgets and a distributed infrastructure that hosts the widgets [31]. Context Toolkit framework separates acquisition and presentation of context information from the application that requires it, by using the widgets [58].

Context widget is one of the abstractions that is similar to GUI widget but it abstracts the source of an input and only deals with the information the source produces [29]. In other word context widgets are software components that provide applications with access to context information while hiding the details of context sensing [31]. Figure 2.3 shows the context toolkit architecture

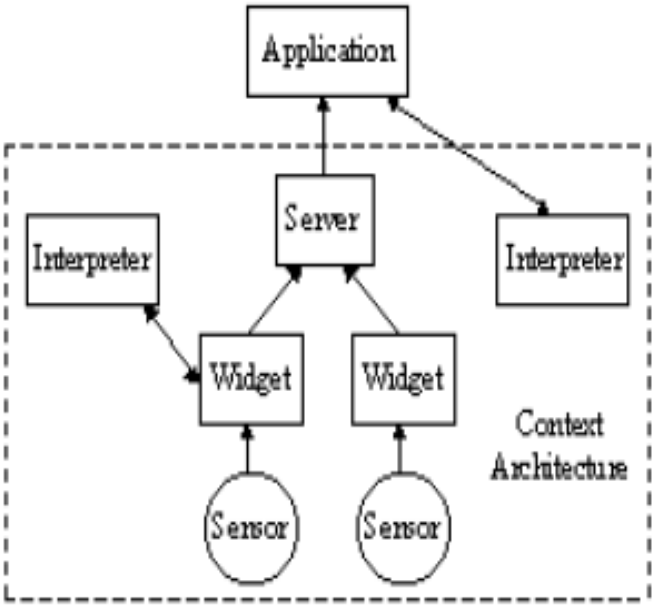


Figure 2.3 Context Toolkit Architecture (CMU) [adapted from 32]

The services of the Context Toolkit are:

- Encapsulation of sensors
- Access to context data through a network API
- Abstraction of context data through interpreters

- Sharing of context data through a distributed infrastructure
- Storage of context data, including history
- Basic access control for privacy protection

B. Context Management Framework (CMF)

The CMF (context management framework) allows semantic reasoning on context in real time and even in the presence of noise, incertitude and rapid variation of context. It delivers contextual information to applications by using a communication model based on events. The framework proposes client/server architecture composed of the following basic components [57]:

- Context manager: responsible for the storage of contextual information on server and the delivery of context to clients using different kinds of mechanisms (request/response, subscription/notification, etc.)
- Resource server: responsible for the acquisition of contextual information from physical sensors and their interpretation according to a specific format before sending them to the context manager
- Context recognition service: responsible for the conversion of the data stream to a presentation defined in the context ontology
- Change detection service: responsible for the detection of service change and therefore the context change
- Security: responsible for the verification and control of contextual information

Figure 2.4 depicts the Context Management Framework.

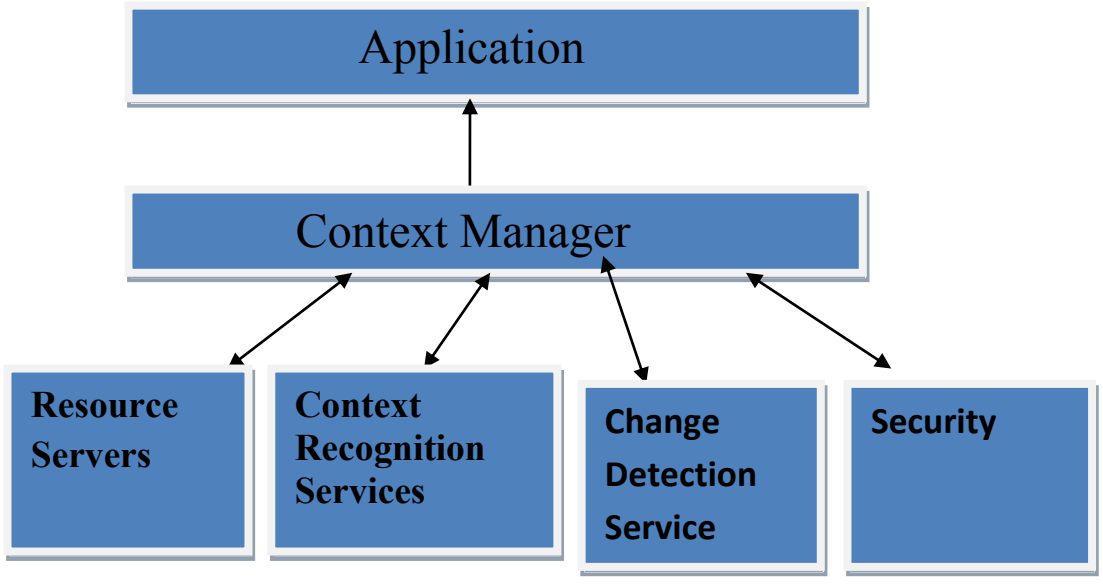


Figure 2.4: Architecture of the Context Managing Framework [adapted from 57]

In mobilife project Flor´een et al. [58] they also design an architecture, which they call the Context Management Framework (CMF) as shown in Figure 2.5.

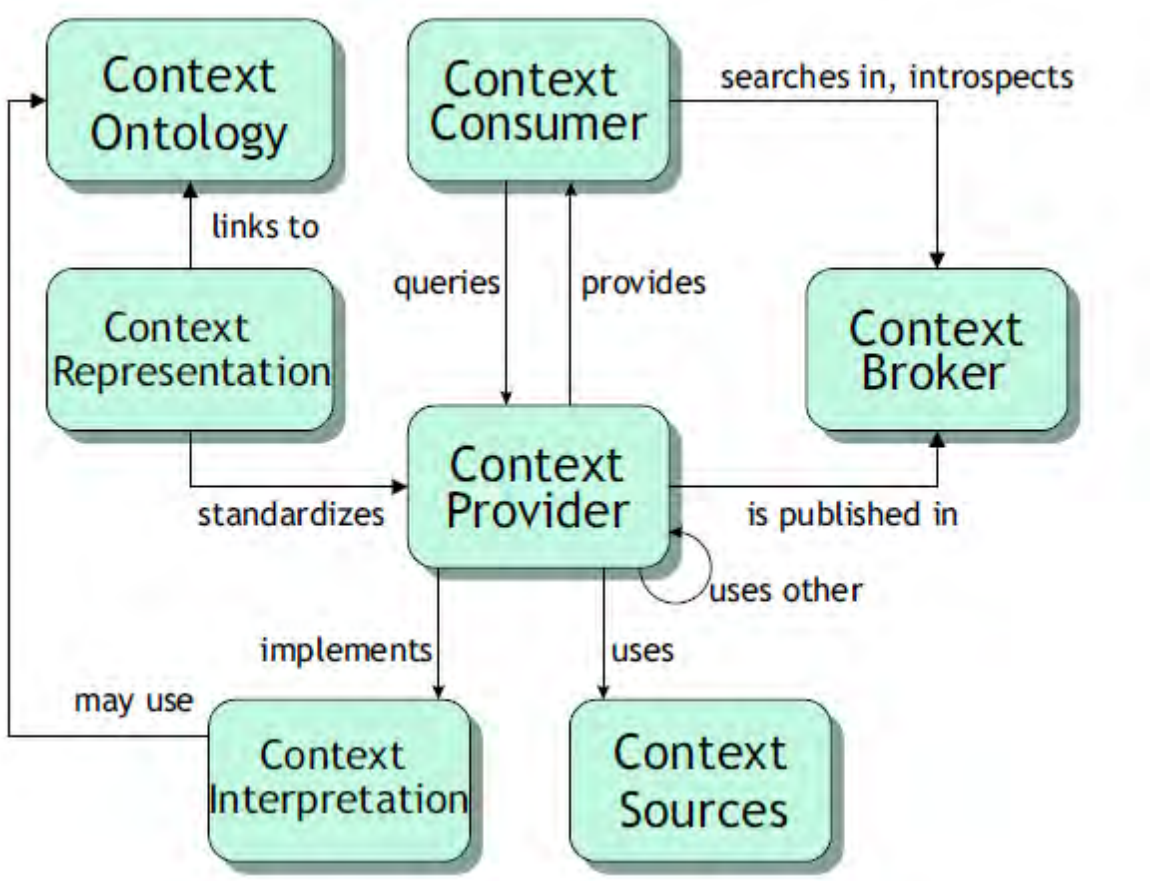


Figure 2.5: Key function of the CMF [adapted from 58]

C. Hydrogen

Hydrogen [69] is both architecture and a framework for context-aware systems. It is a three layered architecture that responds to particular requirements of mobile devices. The architecture (Figure 2.6) has the following layers: adaptation, management and application. The context server (management layer) contains all the sensed information perceived by the sensors of the adaptor layer and provides context to the application layer of the attached device or other devices using a peer-to- peer communication model. The Hydrogen approach considers context as any pertinent information on an application environment and describes it using an object oriented model.

The architecture can be implemented easily, is simple and takes into account the limited resources of mobile devices (battery, memory, processing, etc.) and uses a peer-to-peer communication model (distributed). The adaptor layer does both the sensing and the

interpretation task of context which does not offer a good abstraction of context and limits the reusability of such component. Also, it makes it very dependent to sensors. The architecture does not contain a reasoning module on context to ease the adaptation task.

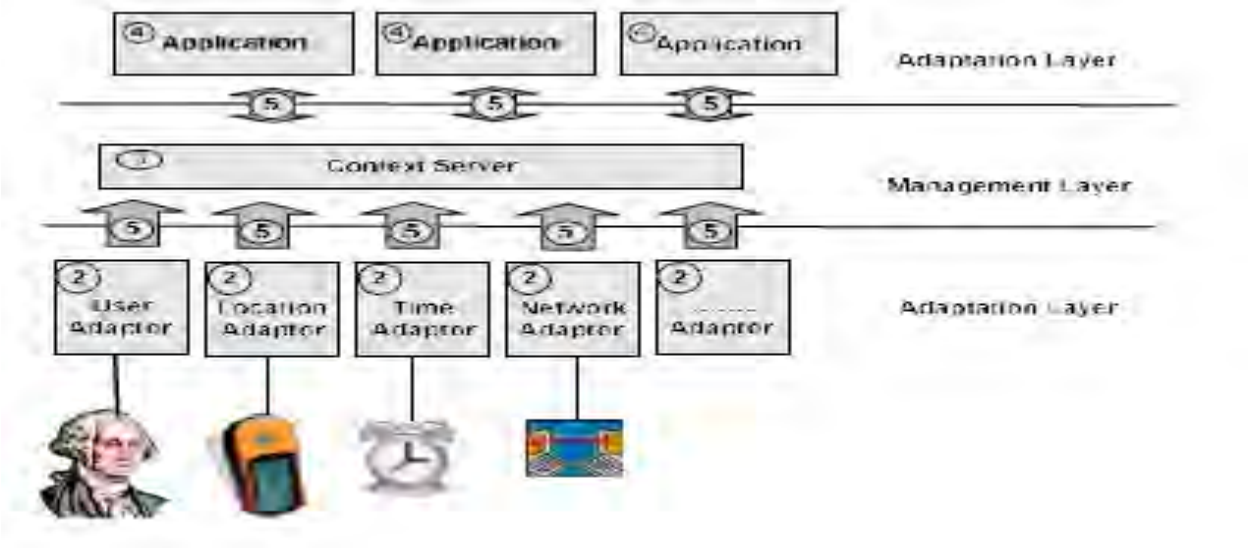


Figure 2.6: The Hydrogen architecture [adapted from 69]

2.6 Context Modeling and Reasoning

As we describe in Section 2.4, developments of context-aware applications is inherently complex. These applications adapt to changing context information: physical context, computational context, and user context/tasks. Context information is gathered from a variety of sources that differ in the quality of information they produce and that are often failure prone. The pervasive computing community increasingly understands that developing context-aware applications should be supported by adequate context information modeling and reasoning techniques. These techniques reduce the complexity of context-aware applications and improve their maintainability and evolvability [54].

2.6.1 Context Modeling

Pervasive environments are characterized by different kinds of elements, e.g. dynamicity, devices, sensors and so on. Two important tools to deal with these smart environments are context modeling and context awareness. Context awareness is important for pervasive computing environments to adapt computational entities to changing situations such as the

user‟s needs and technical capabilities whereas context modeling is an important instrument to deal with contexts and how they are collected, organized and represented.

Schmohl and Baumgarten [46] point out that context model is a model that represents formally contexts in a way computers can interpret it and context modeling is the process of abstracting and representing contextual information for further processing.

Schmohl and Baumgarten [46] and Strang and LinnhoffPopien [50], point out the most relevant context modeling approaches or sets of context models that are classified by the scheme of data structures which are used to exchange contextual information in the respective system. These are:

- *Key-value models*: These are the most simple data structures associating context attributes with specific values of contextual information.
- *Markup scheme models*: These models consist of hierarchical data structures based on markup tags including attributes and comments.
- *Graphical models*: A quite intuitive approach to model context is to represent contextual entities and their relationships graphically. The most prominent examples are Unified Modeling Language (UML), which is suitable due to its generic structure paired with a strong graphical component; and Object-Role Modeling (ORM), which can be nicely utilized to represent context graphically by identifying facts and enriching those with types and roles.
- *Object-oriented-models:* consist of encapsulating contextual information into objects. The information can only be accessed through well defined interfaces and is therefore hidden from other objects. Due to the nature of object-oriented modeling this approach emphasizes reusability and controlled access to contextual information.
- *Logic-based model:* represents a highly formal modeling approach. It is based on logics, which define conditions on which concluding expressions or facts may be derived from sets of other expressions or facts. Those conditions are described by rules in a formal system, so that the facts, expression and rules put all together define the context.
- *Ontology-based models:* use ontologies, which are used to represent concepts and relations between concepts. They represent a uniform way for specifying the model‟s core concepts as well as sub concepts and facts, thus enabling contextual knowledge sharing and reuse.

2.6.2 Context Reasoning

Some context information of interest to particular context-aware application cannot be measured or obtained directly, but can only be derived. This inference of new higher-level context information, as well as predicting future values of context attributes to support proactive applications, is the domain of context reasoning [54,55].

Context reasoning means to automatically deduce further, previously implicit facts from explicitly given context information [51].

M.Perttunen et al. [52] categorize and summarize context representation and reasoning (KR&R) as logic-programming, case-based reasoning and ontology based reasoning. They also mention Bayesian networks, fuzzy and probabilistic logic to deal with representing and reasoning under uncertainty.

2.7 Location based service (LBS)

A location-based service (LBS) is a mobile application that is dependent on the location of a mobile device, like mobile phone. A Location Based Service (LBS) is an information and entertainment service, accessible with mobile devices through the mobile network and utilizing the ability to make use of geographical position of the mobile device [33]. LBS help us locate position of any object on our planet. Common uses of LBS are navigation, mapping, troop deployment in military, vehicle tracking etc. The most popular positioning methods used by LBS systems are:

Satellite Based Positioning

GPS Constellation consists of 24 satellites that orbit the earth every 12 hours. To use this GPS system all we need is GPS receiver. This receiver is tracked by 3 or more satellites and the calculations are done to get a correct position of the GPS receiver.

Cellular Network Based Positioning

In this method, the service provider network towers calculate the position of the mobile phone by calculating the position based on the signal strength from various towers. However GPS approach gives more accurate results.

Short Range Positioning

In this method, the Bluetooth talks with a local server to get position.

The above approaches are taken care by the Location API implementation and finally we get latitude, longitude, altitude and precision of service. LBS development to track a mobile user

can be done in two ways - Application on mobile approach & Service Provider Web service approach

Application on mobile approach

A J2ME or any other platform application sits on the mobile phone and calls API's to find the position. This approach has the following features:

- A GUI based application on your mobile phone.
- No need of any tie-up with the service provider. However, service providers may charge you for access to this service.
- Target devices can only be J2ME phones or any others phones that support Symbian, Pocket PC etc.

Service Provider Web service Approach

Here, your server application directly queries the service provider web service which gives the mobile user position. This approach has following features:

- Major advantage of this approach is that any mobile phone can be tracked immaterial of platform, OS etc.
- This approach is not suitable for mobile maps
- Tie-up is needed with the service provider and the service providers will charge you for this service.

In order to make LBS services possible, some infrastructure elements are necessary, including mobile devices, applications, communication network, positioning component, and service servers [45].

Figure 2.7 shows the interactions among these components, and the process of a LBS service. First, user sends a service request using the application running on mobile device (Step 1). The service request, with user's current location information obtained from the positioning component (in this example, GPS data), is sent to service server via the mobile communication network (Step 2). The service server requests geographic database and other related database to get required information (Steps 3, 4). At last, the requested information is sent back to user's mobile phone via mobile communication network (Steps 5, 6).

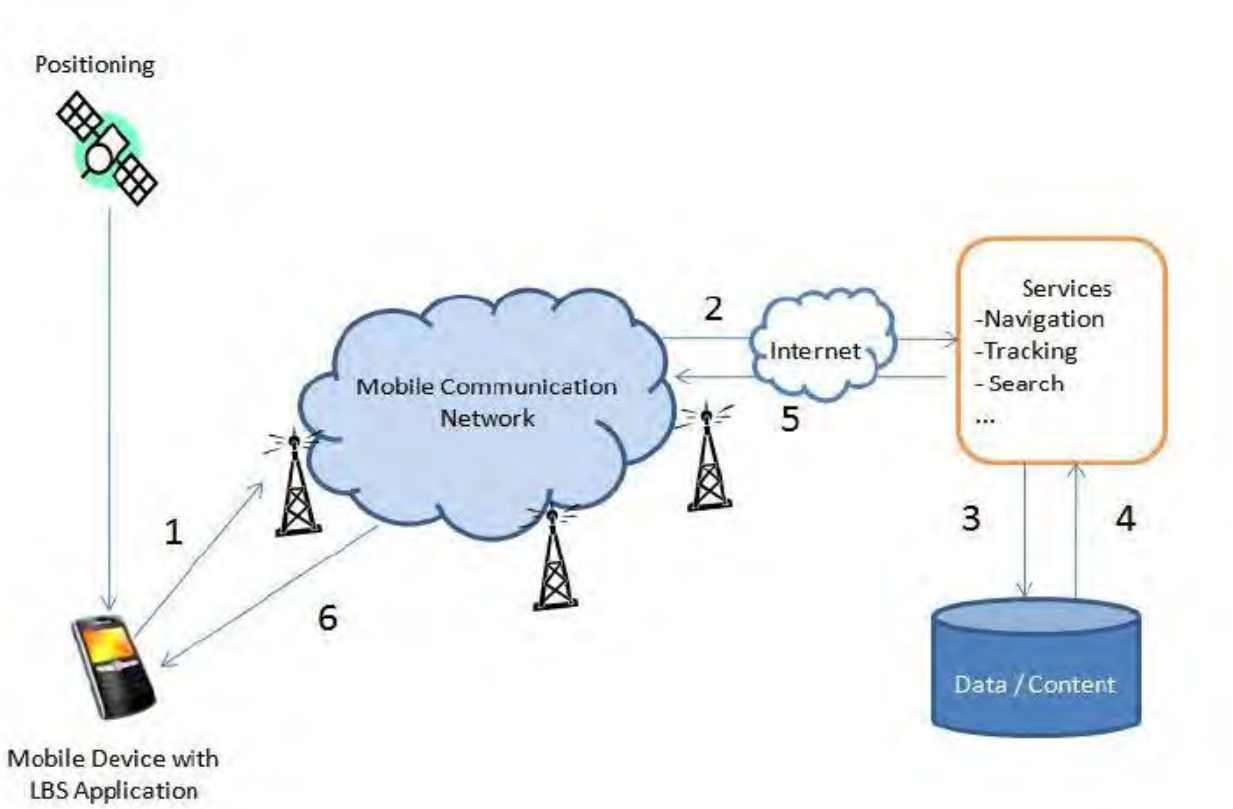


Figure 2.7: LBS components and Service Process [adapted from71]

2.8 Information Extraction from the Web

2.8.1 The World Wide Web

The World Wide Web (W3) was developed to be a pool of human knowledge, which would allow collaborators in remote site to share their ideas and all aspects of a common project [48].It view is of documents referring to each other by links. This simple view is known as the hypertext paradigm [47]. W3 has come to stand for a number of things, which should be distinguished. These include [48]:

- The idea of a boundless information world in which all items have a reference by which they can be retrieved.
- The address system (URI/Universal Resource Identifier) which the project implemented to make this world possible, despite many protocols. URI the string used as addresses of objects (e.g. menus, documents, images) on the web.
- A network protocol (HTTP)-Hypertext transfer protocol (HTTP) is the software conventions used by client and server programs for WWW to request and transfer hypermedia documents.

- A markup language (HTML) which every W3 client is required to understand, and used for the transmission of basic things such as text, menu and simple on line help information across the net. Hypertext mark-up language (HTML) is the system of codes used by authors to build the hypertext-pages/files in WWW, for instance to create a title or an anchor.
- The body of data available on the Internet.

The web operates without regard to where information is, how it is stored, or what system is used to manage it [47]. As shown in Figure 2.8, the architecture of W3 is one of browsers (clients) which knows how to present data but not what its origin is, and servers which know how to *extract* data but are ignorant of how they will be presented.

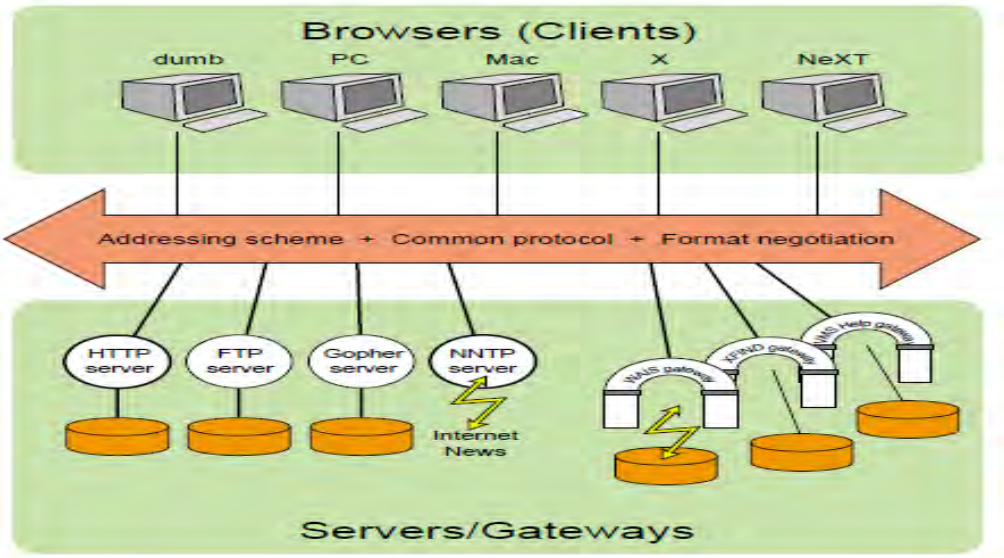


Figure 2.8: Architecture of W3 [adapted from 47]

With its rise in the late nineties, the web was intended as a medium to distribute content among an audience. Alike newspapers and magazines, the communication was merely one way. The content published on the web was presented in an often attractive format and lay-out, using a natural language [49].

Nowadays, the web is a place where people can easily contribute, share and reuse thoughts, stories or other expressions of creativity. The popularity of social web sites enriches the information available on the web. This mechanism turned the web into a place where people can form nuanced opinions about virtually any imaginable subject [49]. The information on the web is currently not only presented in a human-friendly fashion, but also in formats that

allow interpretation of information by machines. The so-called Social Web, or Web2.0, enables people to easily create and publish content. Moreover, content can be easily reused and combined. A movement next to the social web is the semantic web. The semantic web community has created a dedicated formal language to express concepts, predicates and relations between concepts. Using this mathematical language for general information, knowledge can be expressed on every imaginable topic. The semantic web can be seen as a distributed knowledge base. Instead of browsing through web pages, the semantic web enables direct access to information [49].

The web is a way of accessing information over the medium of Internet. The Internet is a huge collection of data that is mostly semi-structured or unstructured. It is very important for websites to be light and responsive. They are designed in such a way that data retrieval is very quick and efficient. Hence, the websites do not use the structured data model that is normally associated with databases but instead use XML or JSON to enable huge amounts of data to be stored and retrieved efficiently. The technologies used in the Internet are designed in such a way that each website can store and present the data differently. With the Internet expanding across the world and the data on the Internet is accessible by everyone, makes the Internet the most efficient and effective platform. Due to the flexibility and scope of the Internet, a lot of development is taking place to invent new technologies to constantly improve the visibility of its content. Many efficient methods have been employed for the storage and retrieval of data. The same content and its interface can be developed in multiple ways using various technologies due to the lack of a fixed structure or hierarchy of data across the Internet. Due to the constant changes involved, designing and developing a standard algorithm or technique to extract data from all the web pages has become very difficult.

2.8.2 Web Mining

Web mining is the process of identifying data patterns on the Internet by using various data mining methods. Internet content is generated from different data sources. The mining of content on the Internet can be divided into three sections.

Web Usage Mining

Many companies and educational institutions are investing time and effort on research to find new methodologies to improve the process of mining data from the web. Web usage mining is especially useful for e-commerce websites, where the company is concerned about the

user‟s interest and would like to present the data in their website according to the user‟s interest. The data from server logs and mouse events of the user are captured and the data is used to identify the user‟s requirement. Some users might look for textual data and some might look for media. This information is later used to analyze and predict the user‟s interest. Then the user is provided with information they are looking for, thereby making the website more user-friendly. Typically, a web application would store the IP address of the user, the sections of the web page accessed by the user, the searches made by the user, and the user‟s mouse events.

Web Structure Mining

Web structure mining is a method to understand the relationship between various hyperlinks available in a website. This would help the website designers to understand the hierarchy of the website and to connect relevant information through links. These links will help the user to access the content of the website.

Web Content Mining

Web content mining is an efficient method to extract useful data from the web page contents. Since the data is semi-structured, the extraction of this data is challenging and also the lack of a data model makes it very difficult to develop a standard technique to identify the information from the websites.

Web content can be divided into two points of view, the IR view and the database view. The IR view or information retrieval view, deals with extracting information from semi-structured data source. The database view tries to determine if the data is stored in the database. In this project, we focus on web content mining and use various techniques to extract the data from the websites.

CHAPTER 3: RELATED WORK

Research topics related to our system include recommender system, location-based service, context-aware mobile recommendation and social recommendation for context aware services.

3.1 Recommender Systems

Adomavicius and Tuzhilin [11] point out that recommender systems emerged as an independent research area in the mid-1990s, when researchers and practitioners started focusing on recommendation problems that explicitly rely on the notion of ratings as a way to capture user preferences for different items. Since the publication of the first papers on collaborative filtering [Hill et al. 1995; Resnick et al. 1994; Shardanand and Maes 1995], recommender systems have become a burgeoning research field [1].

In Adomavicius and Tuzhilin [11], present a comprehensive overview about the traditional or the 2D recommender system i.e. a paradigm that deal with applications having only two types of entities, users and item. The approaches to these recommender systems focus on recommending the most relevant items to individual users and do not take into consideration any contextual information, such as time, place and the company of other people. The rating function R is given by:

$$R: User \times Item \rightarrow Rating \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (1)$$

Rating is a rating domain representing the ordered set of all possible rating values. In recommender systems the goal is to estimate the unknown rating function R total. Once the function R is estimated for the whole User × Item space, a recommender system can recommend the highest-rated item (or k highest-rated items) for each user. Figure 3.1 shows

Figure 3.1: General component of the traditional recommendation process (adapted from [1])

However, in many applications, such as recommending a vacation package, personalized content on a web site, or a movie, it may not be sufficient to consider only users and items – it is also important to incorporate the contextual information into the recommendation process in order to recommend items to users in certain circumstances. By considering this, Adomavicius et al. propose and give a brief description about the context aware recommender system (CARS) especially that is based on contextual user preference

elicitation and estimation approach. Here they advance some possible extensions that could improve the user experience. Among others, these extensions include an enhanced understanding of users and items, and the inclusion of information about user context into the recommendation process:

R: User×Item×context →Rating……. (2)

They typically employ all collected data to determine appropriate recommendations and they also mention how the contextual information or information about the current (or desired) context „**c'** can be applied at various stages of the recommendation process. They point out the different approaches to using contextual information in the recommendation process, i.e.

(1) Recommendation via context driven querying and search. Systems using this approach typically use contextual information (obtained either directly from the user, e.g., by specifying current mood or interest, or from the environment, e.g., obtaining local time, weather, or current location) to query or search a certain repository of resources (e.g., restaurants) and present the best matching resources (e.g., nearby restaurants that are currently open) to the user.

One of the early examples of this approach is the Cyberguide project, which developed several tour guide prototypes for different hand-held platforms. Some Other examples of context-aware tourist guide Systems are GUIDE, INTRIGUE, COMPAS and MyMap systems.

(2) Recommendation via contextual preference elicitation and estimation. It attempt to model and learn user preferences, e.g., by observing the interactions of this and other users with the systems or by obtaining preference feedback from the user on various previously recommended items. To model users‟ context-sensitive preferences and generate recommendations, these techniques typically either adopt existing collaborative filtering, content-based, or hybrid recommendation methods to context-aware recommendation settings or apply various intelligent data analysis techniques from data mining or machine learning (such as Bayesian classifiers or support vector machines).

They also mention that there are applications like UbiquiTO system which implements a mobile tourist guide, provides intelligent adaptation not only based on the specific context information, but also uses various rule-based and fuzzy set techniques to adapt the application content based on the user preferences and interests, that combine both general approaches (i.e., context-driven querying and search as well as contextual preference elicitation and

estimation) into a single system. Similarly, the News@hand system also uses semantic technologies to provide personalized news recommendations that are retrieved using user‟s concept-based queries or calculated according to a specific user‟s (or a user group‟s) profile.

Even if the traditional or the 2D recommender system is not sufficient for the purpose of personalized context aware recommendation, making the recommendation multi dimensional has its own problem. The system proposed by Adomavicius et al. [64] relies on a reduction-based approach, which reduces the problem of multidimensional contextual recommendation to the standard 2D User×Item recommendation space. The approach takes into account only the ratings related to the context of the user-specified criteria in which a recommendation is suggested. However, the exact context sometimes can be too narrow either specific context may not be significant or may not have enough data for accurate ratings prediction which is known as the "sparsity" problem in recommender system.

The reduction-based approach, on the one hand, focuses recommendations on a particular segment and builds a local prediction model for this segment, but, on the other hand, computes these recommendations based on a smaller number of points limited to the considered segment. This tradeoff between having more relevant data for calculating an unknown rating based only on the ratings with the same or similar context and having fewer data points used in this calculation belonging to a particular segment (i.e., the sparsity effect) explains why the reduction-based recommendation method can outperform traditional 2D recommendation techniques on some segments and underperform on others. Which of these two trends dominates on a particular segment may depend on the application domain and on the specifics of the available data.

Based on this observation, Adomavicius et al. propose to combine a number of contextual pre-filters with the traditional 2D technique (i.e., as a default filter, where no filtering is done). Adomavicius and Tuzhilin [4] also point out the contextual post filtering approach that ignores context information in the input data when generating recommendation and contextual modeling approach that uses information directly in the recommendation function as an explicit predictor of a user‟s rating for an item. Both approaches use Heuristic and model based techniques for recommendation. The authors describe a movie recommendation application that includes multidimensional contextual information, such as when, with whom, and where the movie was seen. The general recommendation system differs with our system in the recommendation approach it follows and the considered contextual information and above all our application is specially designed for mobile users.

3.2 Location-Based Services

Among the most popular application there are location-based service (LBSs), in which knowledge of the end user‟s location is used to deliver relevant, timely and engaging content and information [62].

At the present time, information about the user‟s location is the most analyzed contextual element and it turns to be the only one used in popular location-based services (LBSs) such as Google maps, Yahoo! Maps and Bing maps. Location-based services are value added ones in which position information is used to present diverse and interesting services to user including emergency services, vehicle navigation systems, tourist broadcasting services, searching in country-wide or urban sites data banks using map. Location based services (LBSs) can be provided by using GPS, BSC & BTS, RFID and Google Map [59].

Guide is one of location based service applications which provides tourists with up-to-date and context aware information about a city via PDA. Based on the closest access point, the client determines the approximate location of the user and provides tourists with information about sights, a map and the possibility of creating a tour. But to provide this information, the application use, not only location rather personal context (e.g. preferences, current location and a history of already visited attraction) and environmental context (e.g. links between nearby attractions, opening and closing time, relevance to user interest) [66].

Park et al. [65] propose a map-based recommendation system is able to take user preferences into account through a model based on Bayesian networks. The system collects user request and information about the context of use (e.g., location, time, and weather) from the mobile device. Then, it leverages the user profile to display the most relevant POIs on the map. But this system does not extract information from web sources.

As recommender system is an evolutionary concept, H. Costa et al. [61] also by considering an exponential growth of the information available in the last decades, point out that Recommender Systems (RS) are a promising technique to be used in location-based systems. By having this idea in mind they develop a system that used an agent based approach for making context and intention-aware recommendations of points of interest (POI). Mainly, they analyze the advantage of using a multi agent system (MAS) capable of filtering irrelevant information while taking into account the user‟s context. Here, two parted agents architecture was used, with an agent responsible for gathering POIs from a location-based mobile application i.e. the Gowalla‟s Application, and a set of Personal Assistant Agents (PAA) that will collect information about the context and intentions of its respective user.

The PAA have a probabilistic classifier that was trained with the Naïve Bayes Updatable algorithm, for making recommendation. The system architecture has a master agent that is responsible for both agents. It is capable of aggregating the POIs returned from the web agents into a well defined knowledge representation. Figure 3.2 depicts the system architecture.

Figure 3.2: System's Architecture [Adapted from 61]

The system uses standard POI attributes and also it integrates dynamic context data like user‟s context and goals in order to process the request. They produce a scenario by combining the user context i.e. defined by, proximity related to specific POI, current time of day, current day of the week and user‟s goal and POIs context i.e. defined by id, category, price, timetable and day off. POI that Gowalla application provides are food, shopping and night life. Since the main idea behind is, a different recommendation for different users even if their context is the same, the system is able to understand the difference between each user unique preference, intention and behavior. In order to evaluate this, they choose only one profile that can be seen as a stereotype of a user who prefers POIs that are near, cheaper and not closed. They use metrics like correlation coefficient, to correlate two different types of data, precision to evaluate the quality of recommendation and recall to evaluate the quantity of POIs extracted. When an agent recommends POIs to its user, the agent expects the user to rate each recommendation, and saves this information into its memory, which allows it to learn from the experience.

3.3 Context-Aware Mobile Recommendation

In 1994, Schilit et al. [60] pioneered the term context-aware pervasive systems. Their work detailed a model of computing in which several diverse mobile and stationary systems interact with the user in order to determine, according to the user‟s location, POIs and people that are near, as well as changes in those objects over time.

Bohmer et al. [67] pointed out a reason that the current recommender systems for mobile application under achieve since they neglect the very mobility of users. Recommender system should adapt to user‟s context to improve the ease of use. Contextual information required or that is useful for providing better recommendation can be different depending on the application domain and the available data.

Appazaar [67] is one of context aware recommender system that is implemented as a prototype for Google android platform by taking advantage of what the platform provide, like applications to capture context information (e.g. the user‟s calendar, contacts, messaging and the call logs). The authors propose an approach for determining the context of use for mobile applications by observing the user‟s application usage. The application communicates with the server via HTTP request mainly to upload the recorded data and receive recommendation. The server makes all incoming data persistent. The user that has never been in a specific context before receives recommendations on what other users have used in a similar context before and the server maintains Meta information on the application (e.g. type, icon, title, and so on) and on the device (e.g. version of the operating system).

In Y. Lin [68] does not exist present a smart phone application “motivate” that provides users with personalized and contextualized advice on possible physical activities to do. They present the design, implementation and evaluation of a context- aware recommendation system that promotes the adaptation of a healthy and active life style. The application is developed using the Android software development kit. It consists (1) Motivate web application, which is developed for users to edit user personal information and for system admin and (2) Mobile application, that sends the phone location detected by GPS or GSM localization to motivate service which generates advices. Motivate services include advising, location, agenda, weather, profile time and event and if there is a message found, a notification is sent to the user.

Al Tair et al. [63] propose a system which uses contextual information in order to provide recommendations that are more suitable to the particular individual user. The authors propose alternative designs architecture for a traveler recommender system that will use a proactive

multi-agent to achieve its objective. This system integrates the multi-dimensional approach where contextual information is used, with multi-agent system and applies it for tourism. It provides services to the tourist based on their profiles and the contextual information. The system uses profile, username and password, trip details and selection of recommendation as input, and displays or recommends hotels, schedule, map reminder, and notification. In the system, there are three types of specialized agents that are hotels, events/places to visit and restaurants. This information is collected from different websites that provide the information. They implement multi-dimensional approach, conditional probability to make the feedback optional (to minimize user interaction), multi-attribute theory by using reduction-based approach and knowledge based recommendation. The system normally will bring recommendations mostly according to the weight of the ratings, yet when a user rejects a recommendation in favor of another choice, the system will reduce the recommended record and double the ratings of the newly selected choice.

3.4 Social Recommendation for Context-Aware services

Context encompasses more than just the user‟s location because further elements in the current situation relevant to an application are also mobile and changing [60]. Biancalana et al. [1] point out that rating or posting comments associated with items of interest and friends with people are important sources of data that can be analyzed and exploited to improve recommendation techniques and develop new recommendation strategies.

Mobile phones have restricted text input capabilities and small size displays so they recommended that we need a system like polar, that addresses all issues by proposing the modeling of user preferences in order to adapt recommendation to meet his specific need, the definition of richer representation of the context with a view to giving the user only results actually consistent with his current needs and the enhancement of potentialities to exploit the vast amount of information from social networking, user reviews and local search web sites.

The application has different web data sources such as Google Maps, Yelp, Zagat, forums, blogs and face book pages and groups and context augmented real time (e.g. weather or traffic report services). In this system, personalized recommendation is performed in two steps: (1) the POIs that match the current context are retrieved by the local database and (2) POIs ranked according to the user preferences highlighting the most relevant ones on the UI.

The social recommender system includes a context-aware recommendation engine based on artificial neural networks and has explicit representation of the user preferences. They use an associated profile with each user in order to represent his interests (adapted tag based profile). Social networks are also used with location to provide recommendation [59]. Due to many current generation of place based service that do not provide users with personalized suggestions, rather that offer suggestions close to interests based on users distance from the place where they are, they propose a social recommender system that contains capability of identifying users interest and preference and based on them and user‟s current place that offer suggestion.

The personalization process is fulfilled using the information obtained from the learner. It helps to suggest a collection of comprehensive activities taking into consideration factors such as location, user preference and interest and so on. Most personalized systems use the nearest neighborhood algorithm to find the inter relation between user and user. The recommender system uses the inter relation in order to measure the rate of similarities between the two users to identify the users who are expected to be categorized together. The main idea of the social recommender system is similar users receive similar service i.e. if two users have similar interests they are placed in one group.

Social recommender systems are a combination of social data on web like; user‟s social networks and spatial information. Because user‟s information includes personal information and interests in social network sites, considering user‟s current location and the information existing in social network database, it is possible to provide user with a suitable suggestion. Through this method users‟ interaction decreases and they can acquire their favorite information and services. They sugested a model as showen in Figure 3.3

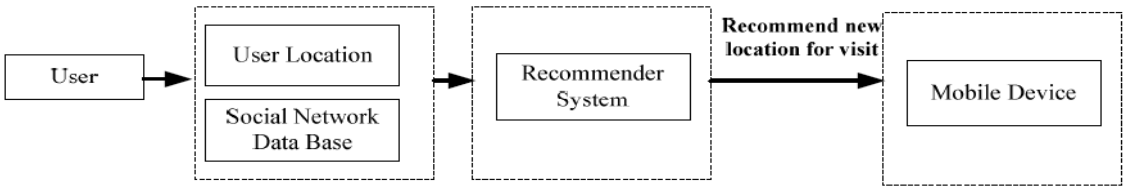


Figure 3.3: The process of suggesting new place to visit [Adapted from 59]

They also show how the recommender system recommends the traffic spam as shown in Figure 3.4.

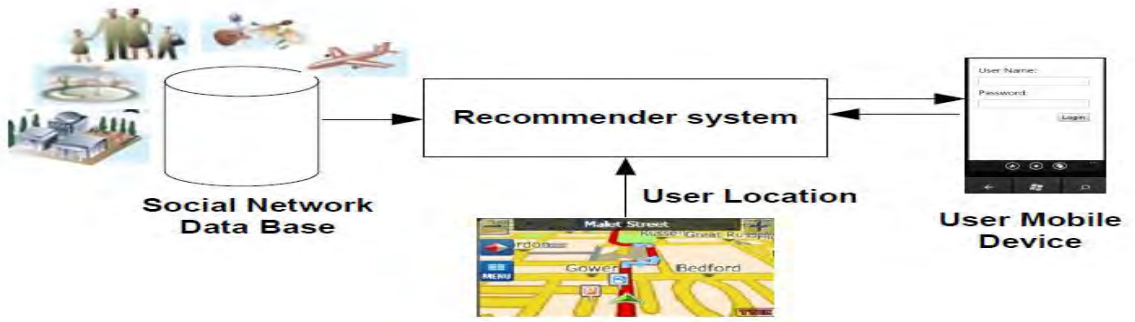


Figure 3.4: Spam traffic [adapted from 59]

3.5 Summary

With the flourishing of pervasive computing there are a lot of researches done and continued to be done in order to make viable its concept and vision. One of those areas that the researches employed is recommendation service or application. There are different recommender systems or applications developed for e.g. tourist guide, book, music, movie etc. For each recommender system, the contextual information that is used is different according to the usability of information and the services.

Adomavicius and Tuzhilin [11] mentioned two approaches i.e. 1) recommendation via context driven querying and search and (2) recommendation via contextual preference elicitation and estimation. In its most common formulation, the recommendation problem is reduced to the problem of estimating ratings for the items that have not been seen by a user. This estimation is usually based on three methods i.e. content based (e.g. if user „A" and user „B" have similar music preferences, then songs liked by „A" but not yet considered by "B" will be recommended to „B"), collaborative filtering (e.g. if user „A" likes song „S", then songs having content i.e. musical feature similar to "S" will be recommended to „A") and hybrid (the combination of content based and collaborative filtering) methods. The general recommender system significantly differs from our system by the recommendation approach it follows and the fact that it is not designed for mobile applications.

Costa et al. [61] location is used as main contextual information. Our system will not use a location based mobile application like Gowalla rather it ties to get the location information from the user or the mobile phone and surf the needed information from the web according to the user"s personal preference. The contextual information needed for the recommendation process in these two cases is also different. Our system also do not expect the user to rate each recommendation, for the purpose of minimizing the interaction of the user with the system.

Under Context aware mobile recommendation also there are different applications [67, 68, and 69] that consider the mobility of the user. There are also systems like [59] and [1] that use social networks for gathering the contextual information of user preference. In this Chapter, we reviewed recommender systems that utilize different advanced methods and techniques to recommend services to users. In our system, we will try to make an entertainment advisory system by studying the case of the services in our country. Our system will create development architecture by combining the contexts relevant to the application like location, schedule, interest or user preference and time for providing event advisory service and it tries to incorporate external contents via web service as well as peer users in order to make the service more efficient. Context information"s used by those applications can also be used here, according to their relevance for the advisory system.

CHAPTER 4: SYSTEM REQUIREMENT

4.1 Existing System

Entertainment is a form of activity that holds the attention and interest of an audience, or gives pleasure and delight and can be considered as one of the common needs for most of human beings. It can be an idea or a task, but is more likely to be one of the activities or events that have developed over thousands of years specifically for the purpose of keeping an audience's attention. Although people's attention is held by different things, because individuals have different preferences in entertainment, most forms are recognizable and familiar. History taught us storytelling, music, drama, dance, and different kinds of performance exist in all cultures, and over time they developed into sophisticated forms and became available to all citizens. This process has been accelerated in modern times by an entertainment industry which records and sells entertainment products. Organizations that mediate the deliverance of the entertainment products and host different events play a major role in delivering the entertainment products to whole citizens.

The experience of being entertained has come to be strongly associated with amusement, so that one common understanding of the idea is fun and laughter, although many entertainments have a serious purpose. This may be the case in the various forms of ceremony, celebration or religious festival. Hence, there is the possibility that what appears as entertainment may also be a means of achieving insight or intellectual growth.

As we mentioned above, the development of entertainment is accelerated by the different entertainment industries. Nowadays in Ethiopia, particularly in the capital city Addis Ababa, there are multiple public and private sectors that provide these services. When we see entertainment areas or when we consider by what activities the people entertain we can generally categorize the entertainment areas as movies, theatre, music, art, sports, festival, restaurants and malls.

The movie industries in Ethiopia are now flourishing more than ever. Locally produced movies as well as foreign movies are watched widely and cinema halls are becoming one of the most favorite places for people to entertain. Among the 9 cinemas available in Addis Ababa, Edna mall cinema and Alem cinema are known for having website to maintain the schedule of the cinema and some movie related information even though the Alem cinema website movie schedule page is not updated periodically. The rest cinemas use paper based information to tell what is on their screen. Therefore, anyone who wants to watch a movie in these cinemas needs either directly go where the cinema is located to see their schedule or

have access to different newspapers and pamphlet on which they use to distribute their schedule and movie related information. On the other hand, some commercial web sites try to maintain partial information about some of the movies and schedules of some cinemas. But, they are not still in a position to relay on for proper information because of the fact that, the information they maintain is dependent on the objective of their organization, as a result it is incomplete and also not interactive too. Musical event information is also advertised using radio, television and posters that hang on some tall buildings of the city. Art event information is even did not get much coverage like movie or music. Mainly, information can be obtained if you have direct contact with the artist, the art event host or if you know a person that works in that area. Sports, festivals, restaurants and mall information are mainly disseminated or advertised using radio or television media.

Looking at the media the current entertainment provider‟s use, we come up with two basic questions that are (1) how many of people get the event information advertised and transmitted in the right time? And (2) how much amount of time did the users invest in searching of the event information and how those searched information turned out to be as the user required? By having these questions in mind and with having high opportunity access to technologies, like phones and computers, which provide access to the Internet, a proper, easily accessible, a user and event context aware delivering event information becoming the very need in the world of entertainment.

4.2 System Requirement

In order to alleviate the pre described problems, in Section 4.1, available in the existing system and to achieve high degree of access and delivery of information related to the entertainment industry, the proposed system focuses on developing a mobile phone based application that will provide personalized event information suggestion to the user by considering the contextual information of its client. In this section we describe the general and specific requirements of the proposed system i.e. a feature that the system must have or a constraint that must be satisfied to be accepted by the client.

4.2.1 General Requirement

The mobile phone based pervasive context-aware event advisory system needs two basic parties to be alive i.e. the user and the event information provider. The system requests the user to register their preference information in order to make a personalized suggestion and it

extracts known stored URLs or the event information providers to come up with the event information that matches with the user preference and other user contexts. Besides, the user and the event information provider, the system also need the basic telecommunication network infrastructure or the wireless network provided by the Ethio Telecom in order to deliver the acquired and processed event information to the user.

4.2.2 Specific Requirement

This section will list down the functional requirement, the system features and the non-functional requirements of the mobile phone based pervasive context-aware event advisor system.

4.2.2.1 Functional Requirement

Functional requirements describe functions the software to execute. The mobile phone based pervasive context-aware event advisor system shall:

- Register the user.
- Register the event information providers.
- Allow the user to manage preference.
- Allow the system administrator to manage the users.
- Manage the extracting of event information from the web.
- Manage user published event information.
- Manage extracted event information.
- Manage user location.
- Manage user schedule.
- Manage the event notify.
- Manage the event and user context matching.

4.2.2.2 Non-functional requirement

The non-functional requirement describes constrains for implementing these research project. Some of them are the user interface, hardware consideration, software consideration, performance requirement, security requirement and some other related factors.

User interface

The user of mobile phone based pervasive context aware event advisor will have an interface to interact with the system. Generally, the user interface shall:

- Have consistent and uniform gray backgrounds.
- Be easy to use and have consistent graphical representation.

Hardware consideration

The proposed system has the client side or the mobile application and the service center has a dedicated server for the system and the database. The mobile side of the application runs on any mobile phone that supports j2me platform and the service center of the application run on any server that supports windows operating system.

Software Consideration

Through the progress of the research project work, the following software components will be used:

- Sun Java(TM) Wireless toolkit 2.5.2 _01 for CLDC
- MYSQL Server 5.0
- MYSQL Workbench 5.2
- Microsoft Visio
- Office application

Performance consideration

- The system shall have acceptable performance with regard to response time. At the same time, the system must provide accurate, consistent and reliable data to the system users.
- User interface screens shall respond with speed so as to ensure fast data entry to the system.

Availability

- The system shall be available 100% for the user and is used 24 hours a day and 365 days a year unless it encountered problem and the system is down.

Security requirement

- The system must verify each user in order to secure unauthorized access of update of preference information and to identify which user is the log in user

- The system must allow users to access it at any time from any location.

Error handling

Possible runtime exceptions shall be addressed thoroughly using exception handling routines. The error handling shall produce appropriate, meaningful and unambiguous error messages.

4.3 System Model

The system model section documents the different models and diagrams that are used to model the proposed system, which are use case model, class diagrams and sequence diagrams.

4.3.1 Use case Model

Actor description

We have three actors that the system interacts with.

- **User**: any person that has access to the mobile side of the application and is in need of having a suggestion of event related information.
- **Event Information Provider**: any person or an organization that host and organize an event.
- **System Administrator**: a person that manages and controls the system, the users and the event information providers.
- **Ethio Tele:** is an external system that used as a service provider for the system by providing the current location information of a user when the system accesses its cell tower information.

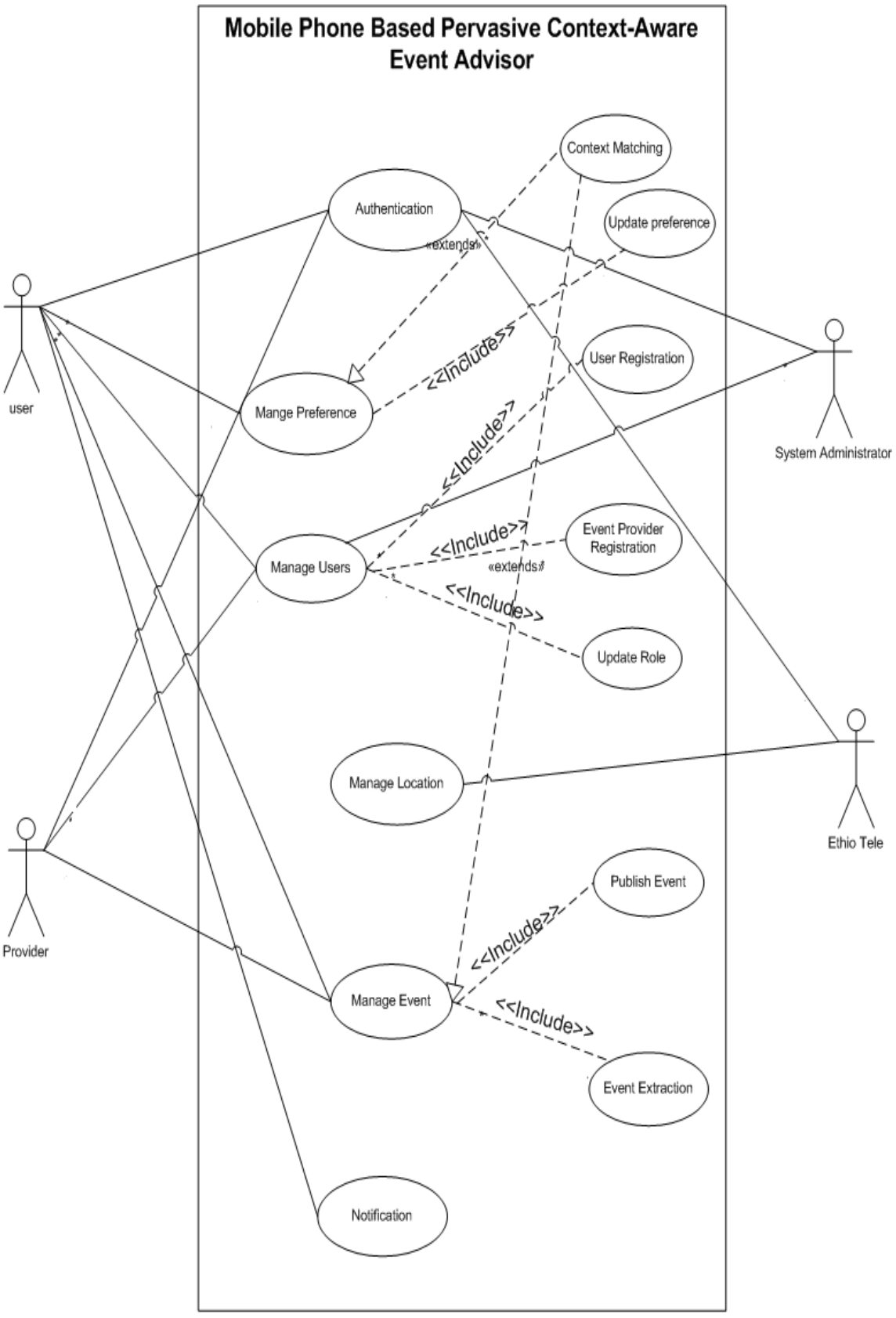


Figure 4.1 System Use Case Diagram

Use case Description

Table 4.1: Use case Authentication

Use Case ID:	**UC-1**
Use Case Name:	Authentication
Actors:	User, System Administrator, Provider, Ethio Tele
Description:	The system needs to check if the user, the system administrator, the event provider and the Ethio Tele is genuine or truly what they said to be.
Pre conditions:	Users must have an account.
Post Condition:	The users will be authenticated.
Normal flow:	1. User clicks log in button. 2. System displays the log in page. 3. Users select their role. 4. Users fills user name and password and press log in button. 5. System process and confirms.[Alt 5] 6. System displays option page.
Alternative flow:	Alt 5: User name or password mismatch 5.1 System send try again message. 5.2 User fills again and submits.

Table 4.2: Use case Manage Preferences

Use Case ID:	**UC-2**
Use Case Name:	Manage Preferences **extends** (Update Preference and Context Matching)
Actors:	User
Description:	The users manage their event information preference as well the application access of dynamic information from or by using the mobile device.
Pre conditions:	user must sign up to the application
Post Condition:	The user preference will be stored in the database.
Normal flow:	1. user signup to the system 2. System sends confirmation on the response form.

	3. User select next button. 4. Application displays the setting preference form. 5. User chooses their preference. 6. User click next button. 7. User selects their preference for other event type categories. 8. User finishes and submits. 9. System displays a confirmation message.
Alternative flow:	None

Table 4.3 Use case Manage Users

Use Case ID:	**UC-3**
Use Case Name:	Manage Users **includes** (User Registration, Provider Registration and Update Role)
Actors:	User, Provider and System Administrator
Description:	The users of the system specifically user and provider must be managed by the system. The two users must be registered to access and use the system and the system administrator must manage the two users‟ role that includes auditing spam users and block them or kick them out from the system.
Pre conditions:	Users must have the application on their mobile device.
Post Condition:	The users will be registered to get the application services and the system administrator control the two users.
Normal flow:	1. Users select Sign up from the menu. 2. System displays the sign up form. 3. Users select their role. 4. System displays sign up page according to their role. 5. Users fill the required information and click sign up button. 6. System process and confirms. [Alt 6] 7. System displays the next page according to their role.
Alternative flow:	Alt 6: Users inserted already existing user name or password. 6.1 System lets user to try again 6.2 Users fill again.

Table 4.4 Use case Manage Event

Use Case ID:	**UC-4**
Use Case Name:	Manage Event **includes** (Event Extraction and Publish Event) and **Extend** (Context Matching)
Actors:	User, Provider
Description:	The user and the provider manage the event information that is used as current event information to be suggested to the user by publishing event information and by posting event information on their website respectively.
Pre conditions:	Users must been logged in to the application.
Post Condition:	The users can provide event information and the information in the system will be updated.
Normal flow:	**User** 1. User log in to the application. 2. System displays the option form. 3. User select publish button. 4. System displays the information publishing page. 5. User selects and fills the required fields. 6. User selects the publish command. 7. System save the given event information. (What about the Provider flow) **Provider** 1. Provider post event information in their website. 2. System extracts the information. [Alt 2] 3. Extracted event information will be saved in the data base.
Alternative flow:	Alt 2: If event extraction fail 2.1 The system triggers event extraction manager with in the specified time. 2.2 System extracts the information. 2.3 System save the information in the data base.

Table 4.5 Use case Manage Location

Use Case ID:	**UC-5**
Use Case Name:	Manage Location
Actors:	Ethio Tele
Description:	The Ethio Tele using its cell towers manage the information of a user current location.
Pre conditions:	User must be authenticated.
Post Condition:	Be the location provider and provides the user location by identifying the nearest cell tower to the user.
Normal flow:	1. System authenticate location provider. [Alt 1] 2. System accesses the service provider system. 3. System gets the cell id information of the nearest cell tower to the mobile device.
Alternative flow:	Alt 1: If the system cannot retrieve the location provider. 1.1 The system tries to retrieve the location provider after 2 minute.

Table 4.6 Use case Notification

Use Case ID:	**UC-6**
Use Case Name:	Notification
Actors:	User
Description:	The user can get and view his/her personalized event notification.
Pre conditions:	User must have the application that bundled the main event advisor and the notification midlet on their mobile device and must log in to the application.
Post Condition:	View and read event information notifications.
Normal flow:	1. System sends notification messages. 2. The application displays a pop up message on the mobile device screen. 3. User clicks view and read the message. [Alt 3]
Alternative flow:	Alt 3: If the user cancel viewing the message, he/she can read

	later 3.1 User log in to the application. 3.2 Application displays the option page. 3.3 User select notification button. 3.4 Application displays recently pushed notification messages. 3.5 User read or views the notification messages.

4.3.2 Class Diagram

Class diagram represents the static view of an application. It describes the attributes and operations of a class and also the constraints imposed on the system. Class diagrams help us to visualize the relationships between classes, interfaces, and collaborations.

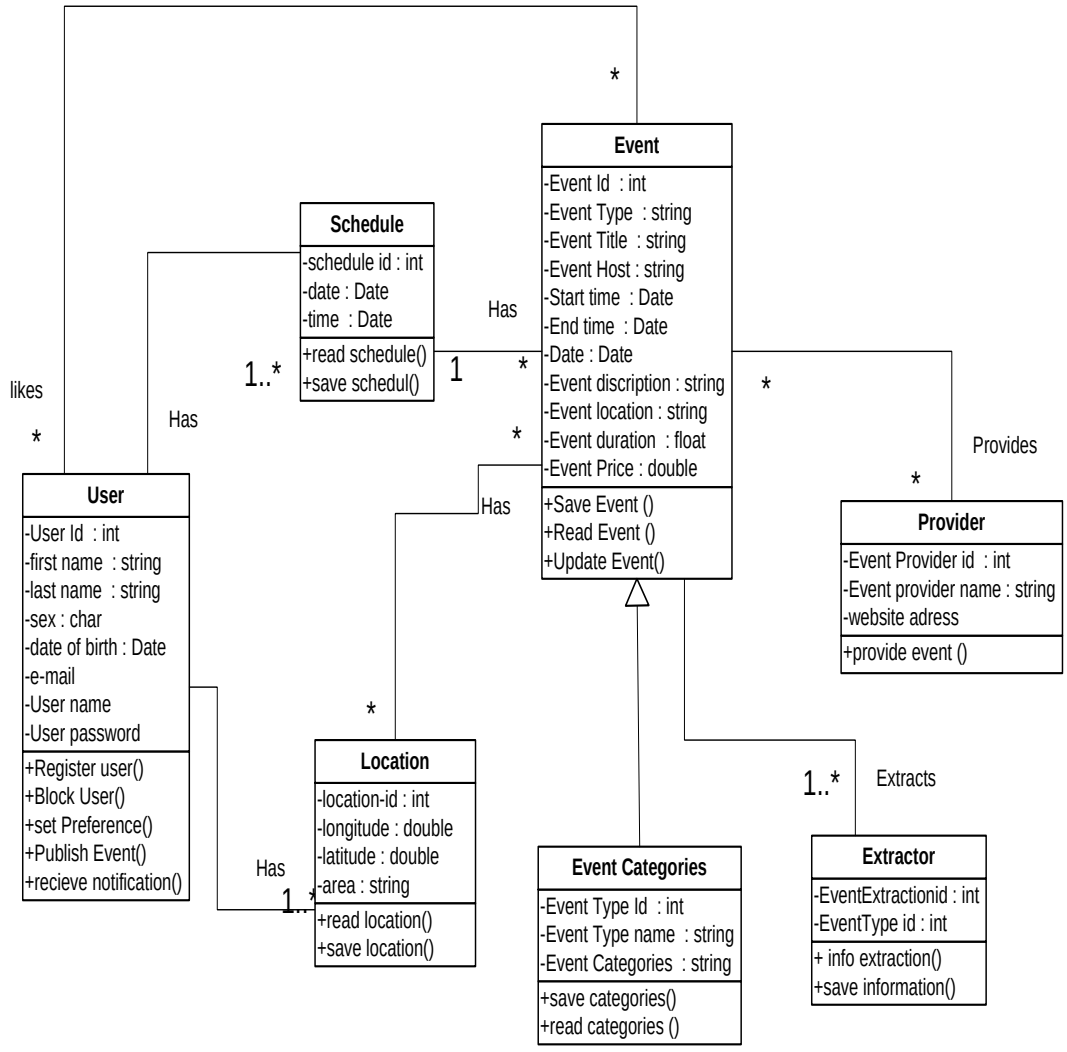


Figure 4.2 Class Diagram

4.3.3 Sequence Diagram

Sample sequence diagram with its description for the proposed system is presented in Figure 4.3. Sequence diagram for some of the identified objects are presented in (Annex A).

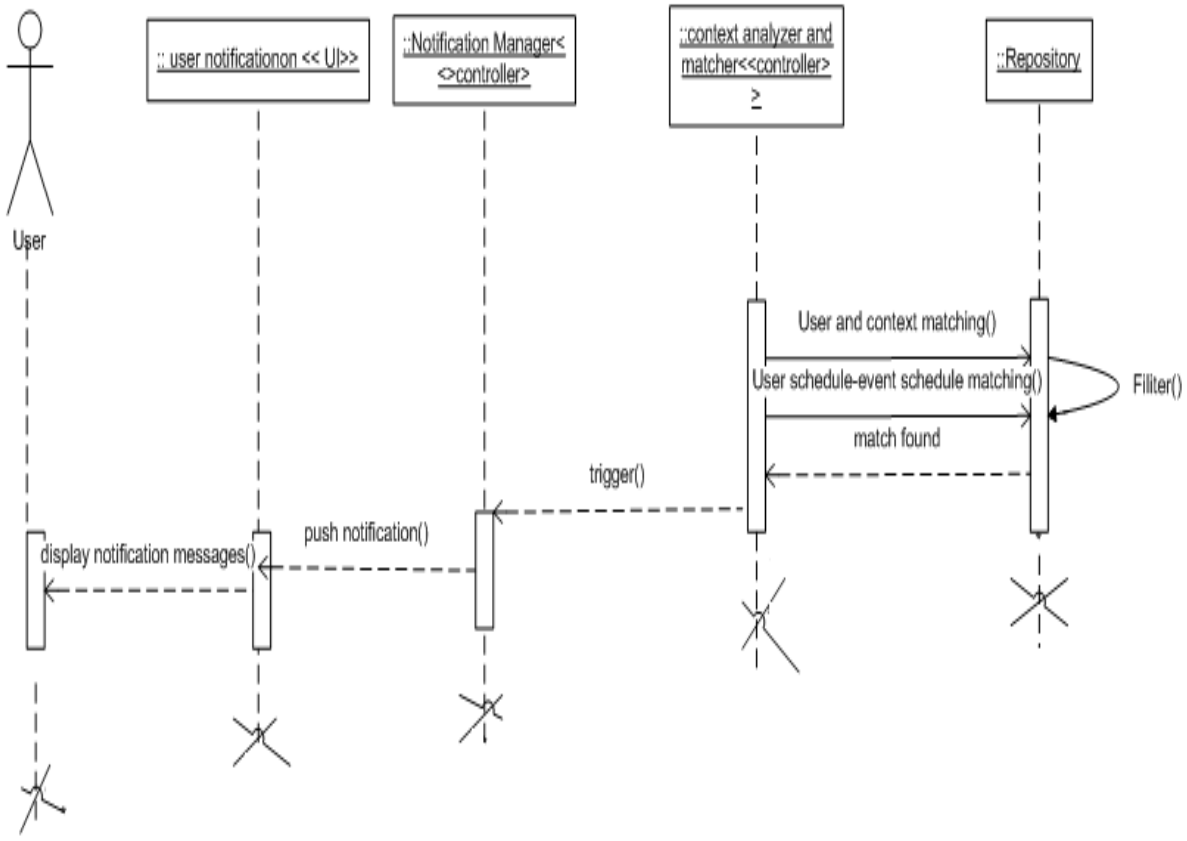


Figure 4.3: Sequence diagram for matching movie event preference

CHAPTER 5: SYSTEM DESIGN

5.1 Design Goal

The design goals specify the qualities of the system that should be achieved and addressed during the design of the system. The design goals for the system are grouped into four categories. These are:

- Performance
- Dependability
- Maintenance
- End user

5.1.1 Performance

In order for the mobile phone based pervasive context-aware event advisor system to give services for multiple concurrent users per each minute the system should meet the following performance criteria‟s.

- **Response time: -** Depending on the network connection that the user machine has the system is going to interact and respond to user‟s request that requires the processing of the data base in a maximum of 2-3 seconds.
- **Memory:-**The J2me mobile application requires 32 KB workspace, 128KB for MIDP implementation and 8KB for MIDlets to run.

5.1.2 Dependability

The system should achieve the following dependability characteristics in order to resist crash and be available and reliable.

- **Robustness: -** even if the system is a mobile application and require user to fill specific information, we try to minimize the entering of error data by making the system a menu and selecting driven.
- **Availability: -** as long as there is an the mobile network or wireless connection on the user mobile phone the system will be available 7 days a week and 24 hours a day.
- **Security: -** the system provides a software security by having a user name and password so that only authenticated users can use the service. Hard ware security is not considered here.

- **Reliability:** the information provided by the system can be considered reliable since it process known stored URLs or official sites for the event information. The pit fall here can be the trust level of the event information provided by the users. To make the system more reliable the information that is sensitive to location change must have a delay of average 3-5 minutes to be notified to the user. This is necessary to prevent delivery of already passed event information.

5.1.3 Maintenance

In time of failure or need modification the system need to be maintained. To be maintainable the system should meet the following maintenance criteria

- **Extensibility: -** if it is needed to add new functionality to the system, this must be achieved by only adding a command in the menu and by making a separate form and integrate this form with the existing system. But in order for the user to access the added functionality they must upgrade their application to the new version.
- **Modifiability:** - if in the system, some functionality requires to be modified, this modification must be done specifically to that function without affecting the overall system organization.
- **Portability: -** the system is developed to be viewed and retrieved from mobile phones that support J2me platform that means the system is portable in any mobile phone that run J2me.

5.1.4 End user

From the user point of view the system should provide the following usability software feature. It should be

- Easy to use
- Easy to learn the application
- Convenient to the end user by considering the user context

5.2 Architecture of the system

In the context of this research project we design our system architecture as three-tiered client-server architecture by combing data-centric and event based architecture style. We use a common context repository to store the user and the event information context as well as a publish/subscribe way to multicast the event notification message to the subscribed mobile devices. The system architecture is shown in Figure 5.1.

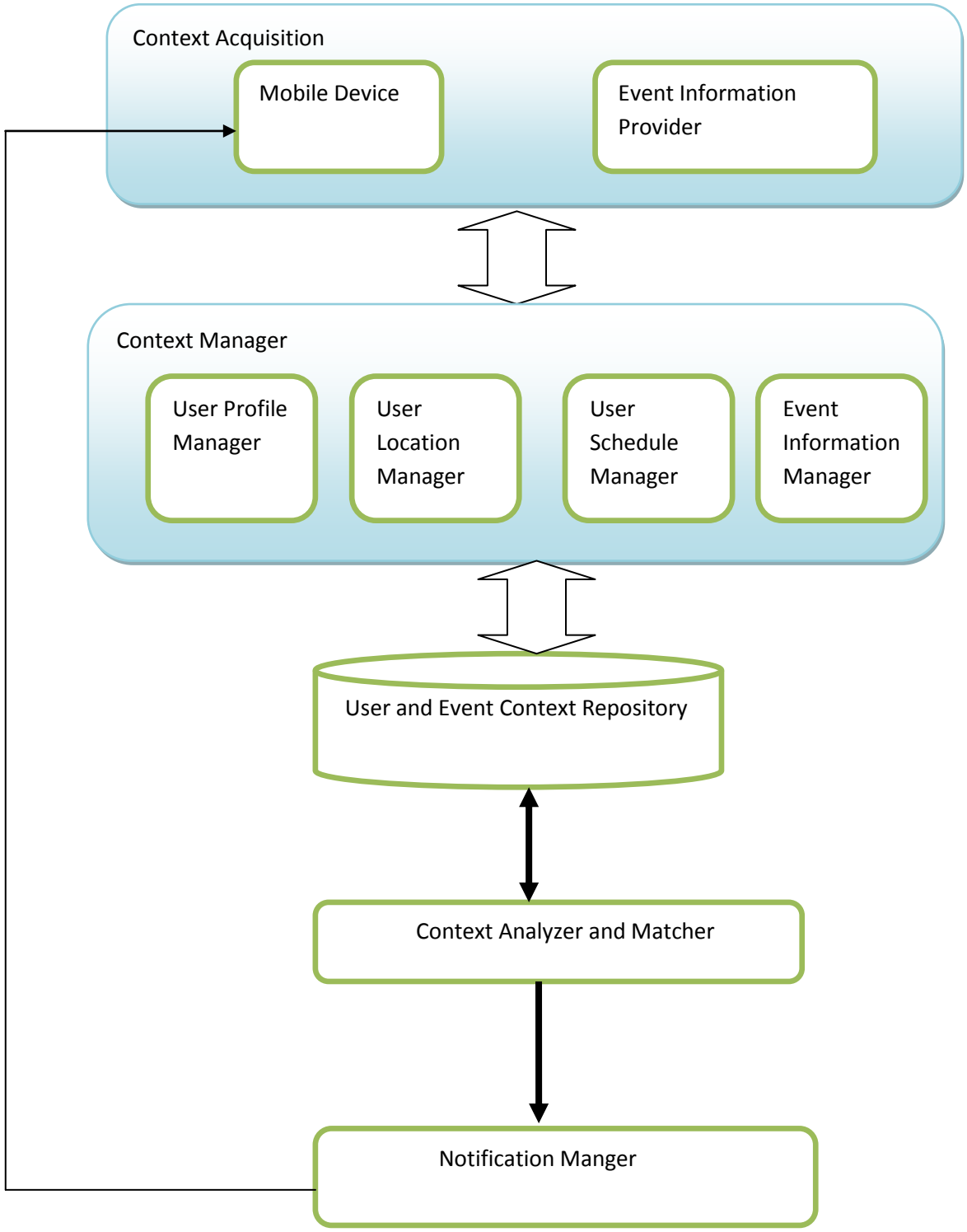


Figure 5.1 Architecture of the Proposed System

5.2 Detail view of the proposed Architecture

In the proposed architecture we have five basic core components. These are Context Acquisition, network manager, context manager, user and event context repository, context analyzes and matcher and notification manager.

5.2.1 Context Acquisition

In this component, we have two modules that are mobile device and event information provider. From the mobile device and event information provider modules we acquire the user and the event contextual data respectively.

- **Mobile Device** - In the client side the mobile phone of the user is used as a means of acquiring the static and dynamic user contextual information.
    - **Static user context**

      Static contexts are those aspects of a system that are invariant or less dynamic. In our context, user profile and user preference are example of less frequently updated static contextual information that can be acquired from the user mobile phone. In the proposed architecture, in order to make advising more of personalized; each user must register and must be subscribed to the service and each authenticated user must provide their personal event preferences.
    - **Dynamic user context**

      Dynamic context is contextual information that frequently changes over time. In dynamic context there is high volatility and variability. In the proposed architecture, the user schedule and user location are considered as dynamic user contextual information.
- **Event Information Provider-** will be the main information provider of the upcoming events as well as the events that are found around the user. It provides the dynamic event context.
    - **Dynamic Event Context**

      In this module, we acquire event information from the web. Here, we use different known URLs that hold event information that are related specific to the user preference. The event advisor server will query the event provider in search of new events with a constraint of time in a daily basis.

5.2.2 Context Manager Components

In this component we have four modules that deal with and manage the static and dynamic data of the user as well as dynamic data of the events. Those are

- **User Profile Manager**

  This module will manage the static information of the users‟ that is the user account information as well the user preference selection. Each user should have a unique user name and password combination as well as must specify his/her personal preference in terms of the services he/she wants to get from the system. This module will deal with accepting information from the interface and pass the information to the server side so that it can be stored in a database for a latter use. It is also responsible in managing the published event information from the users. In event advising system, there are two kinds of event provider. The first and the main event information provider is the web or the event provider that help us extract the event information from other third party servers that hold the required information. But since the system has to be user contextual and dynamic in serving the users, it needs other resource providers besides the websites. Since most websites might not include every useful event and can be down at times and less updated too, we found that using other resources are essential for making the system resourcefully rich. So, using the publishing form users will be able to pass information to the server which analyzed and pushed to other users that prefer the published event information with the consideration of other contextual parameters. This helps users of the system to get more out of the system and not only depend on system based (extracted) information only.

- **Schedule Manager**

  This component manages the schedules of the user that are acquired through the use of the user mobile device. The dates that are retrieved from the mobile phone calendar will be sent to the server side in order to be saved; by doing so a user gets notification of events that are in the time range specified suitable. The scheduling system is categorized in to busy and free based on day time categories. Such classification enables the user to have a high flexibility and be more contextual in getting notification of events that are only in the free and specified range.

- **Location Manager**

  The user‟s current location is the most dynamic data in the system. Every time the user location changes the system is aware. This means when a user gets a preferred service,

the current location will also be considered. This information is mostly usable for suggesting the user, about preferred factual information for example restaurants and or malls that are found near by the user. The location manager is responsible for acquiring the geographic coordinates or the current location of the mobile device.

- **Event Information Manager-** related web resources or known URL‟s are periodically analyzed to be used as a resource input for the system. The web sites which provide event data are specified or registered in advance by the system and the web extraction manager deals with connecting, extracting, converting the data and finally providing these resources to the system to be used as primary resources for users.

5.2.3 User Context and Event information Repository

Context repository component stores the information acquired and managed. The user account, preference, schedule and location information collected from the user mobile phone will be stored in a web server database so that the system can use this information at time of suggesting. The user‟s current location is fetched within constant time interval and will be saved in a server database for immediate use. Moreover, users currently in the same location area will be able to get the same services of the system by considering their specified preferences. Beside the user context data, extracted event information will be stored in the database for the purpose of suggesting or advising the user about events.

5.2.4 User Context Analyzer and Matcher

All of a user‟s specific preferences, schedule and location information are aggregated together to give a contextual definition of a specific user‟s need or expectation from the system. In this component, the user contextual information will be matched and filtered out with the event information acquired.

5.2.5 Notification Manager

This component deals with delivering the event information to the user. It manages and controls the notification process. To start notification service it requires the user to be authenticated and to have new event information in hand that meet the user contextual parameters.

5.2 System Decomposition

System decomposition is decomposing the system into manageable parts or subsystems based on use case and analysis model. Subsystems are collection of classes, associations, operations, events and constraints that are closely interrelated with each other. We select the subsystems based on functional requirements, cohesion and coupling. Figure 5.2 presented the system decomposition of mobile phone based pervasive context-aware event advisor system.

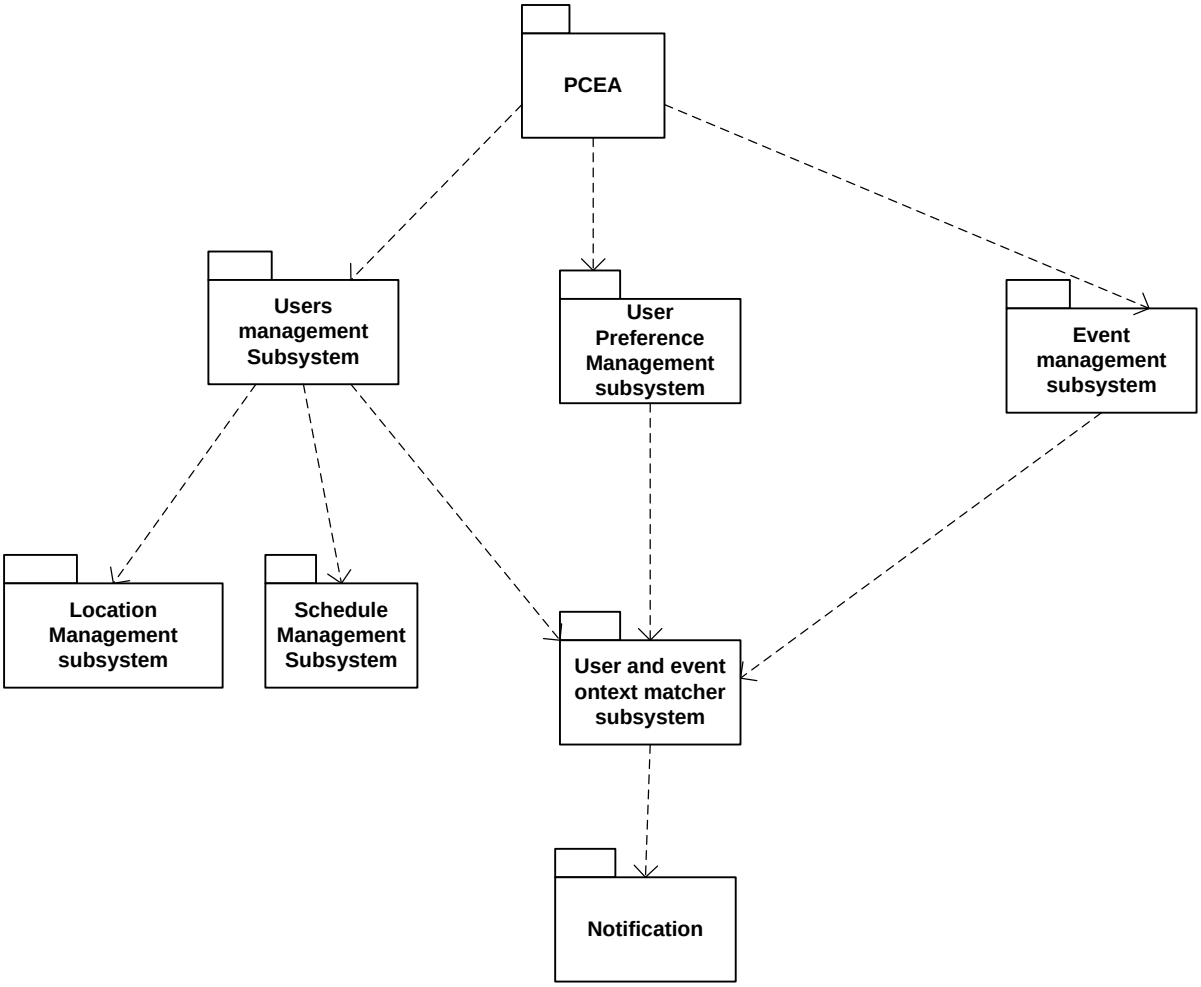


Figure 5.2 Subsystem decomposition of the proposed System

5.4 Hardware/ software mapping

The system will have three processes, deployed in separate machine, that run in parallel, namely, the mobile application process, web server process and the database process. The database process, which runs on MYSQL Server database engine, is responsible for maintaining data manipulation operations and storing the user as well as the event contextual information. Whereas, the web server process is responsible to process and manage the context data that are retrieved from the mobile device and the event information data that are extracted from the third party event adviser web server. It is also responsible to process, analyze and find match between those data and deliver the notification to the mobile device. In case of the client side, the mobile application of the system resides there. It request service from the web server and accept the processed responses.

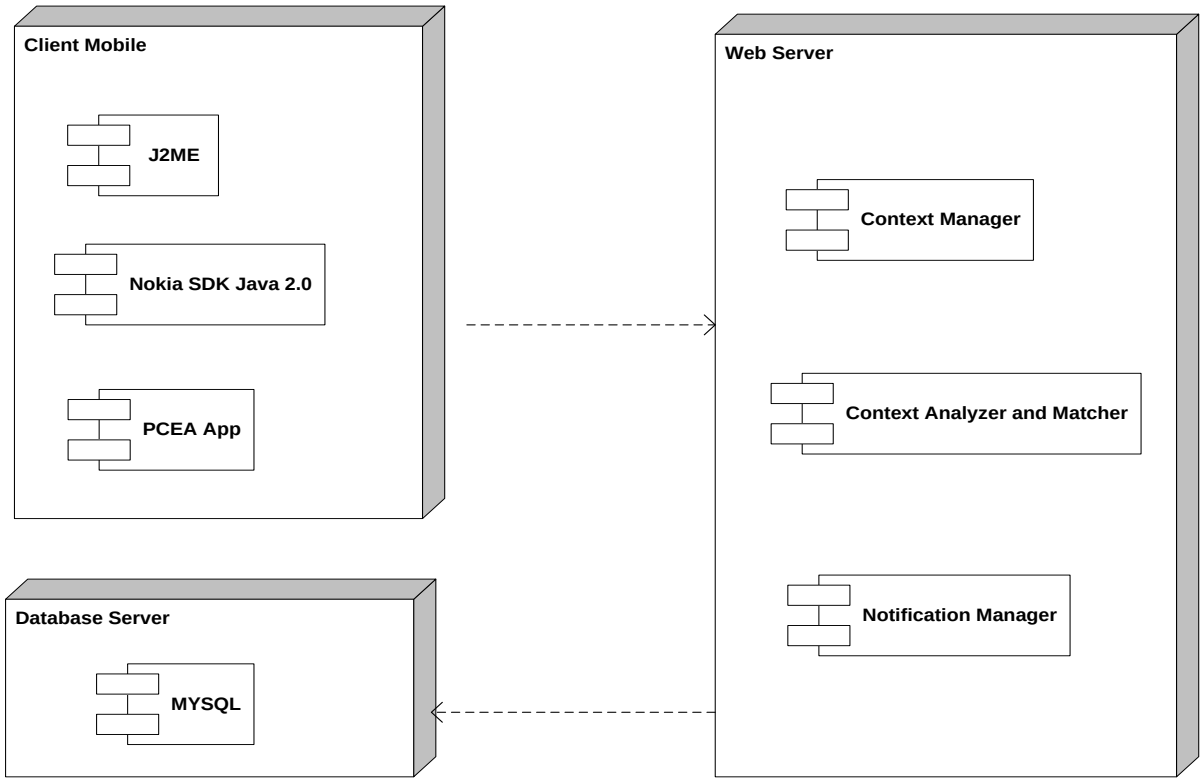


Figure 5.3: Hardware/Software mapping of the proposed architecture

5.5 Database Design

A program will create a large amount of data throughout its execution. Each item of data will have a different life time. Objects have a lifetime. They are explicitly created and can exist for a period of time that, traditionally, has been the duration of the process in which they

were created. A file or a database can provide support for objects having a longer lifeline longer than the duration of the process for which they were created. From a language perspective, this characteristic is called object persistence. An object can persist beyond application session boundaries, during which the object is stored in a file or a database, in some file or database form. The object can be retrieved in another application session and will have the same state and relationship to other objects as at the time it was saved. The lifetime of an object can be explicitly tenanted. After an object is deleted, its state is inaccessible and its persistent storage is reclaimed. Its identity, however, is never reused, not even after the object is deleted.

To map objects to relational databases the place to start is with the data attributes of a class. An attribute will map to zero or more columns in a relational database. Remember, not all attributes are persistent; some are used for temporary calculations. At some point the attribute will be mapped to zero or more columns. Figure 5.4 depicts the mapping of PCEA class object in to database tables.

`

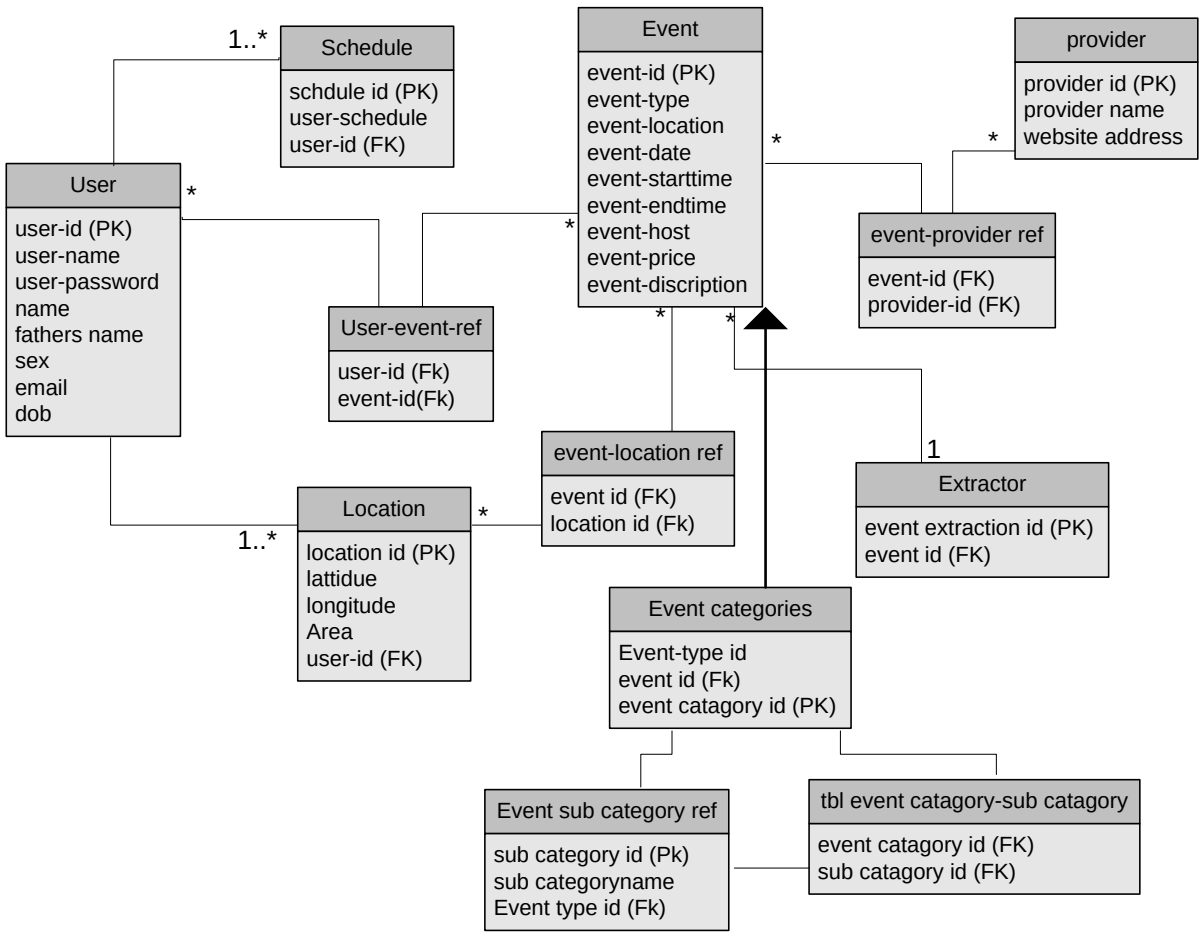


Figure 5.4: class- database tables mapping

CHAPTER 6: IMPLEMENTATION AND VALIDATION

6.1 Tools and Technologies Used for Prototype

The following list of software tools and technologies were used for the full development and implementation of PCEA:

- **Microsoft Windows 7** operating system is used for PCEA server
- **NetBeans IDE** (Integrated Development Environment) is used for writing and running the client as well the server code.
- **Java 2 Micro Edition (J2ME): version 2.5.2 Sun Java Wireless Toolkit** for CLDC (Connected Limited Device Configuration) of the MIDP (Mobile Information Device Profile) for mobile device simulator development.
- **MySQL Server 5.0** -for developing the database.
- **MySQL Workbench 5.2 -**for creating the schemas and we mainly use it for writing database queries.
- **MySQL _connector_java 5.1.23**
- **HTTP** (Hyper Text Transfer Protocol) version HTTP/1.1 - used for communication between the mobile phone and Event Advisor Web Server.
- **Jsoup version 1.7.3-** for extracting the web content
- **Nokia_SDK_2_0:** For accessing the Nokia LocationUtil package
- **Vertigo Server** used for developing our web server.
- **Microsoft Visio 7:** For designing the UML(Unified Modeling Language) diagrams of the system

For implementing the proposed architecture we select Java ME (J2ME) platform since the platform represents the only true open solution for building mobile applications for the industry. Since j2me platform support different devices we try to narrow down the selection of device we use for deployment by taking a random sample test on 20 mobile phone users. The result we get is as shown in table 6.1.

Table 6.1Sample random test for identifying which mobile devices are used by most users

Device Name	Number of user	Number of user in Percent
Samsung	5	25%
Nokia	9	45%
Motorola	0	0%
Blackberry	1	5%
Other	5	25%

As shown in the above table we get 45% user of Nokia mobile phone, 25% Samsung, 25% other mobile devices that are Huawei, Tana and Android phones, 5% Blackberry and 0% Motorola. From the result we deduce that the amount of Nokia mobile phone users are abundance in number so we choose a Nokia mobile phone specifically Nokia Asha 201 and Nokia C-01as a deployment device. The application will be benefited with additional API packages that Nokia provided.

6.2 The Prototype

In developing the prototype of the system we map our application in to client/server architecture. In our application the MIDlet invoke the methods that are found in the server by using the Http request response protocol.

6.2.1 The J2me/mobile side of the application

In developing the prototype of the proposed system, we have to make a simple user interface that help the user to register and select their own personal preferences to get service. This is achieved by using java ME SDK and creating a MIDlet application, which uses the Mobile Information Device Profile (MIDP) of the Connected Limited Device Configuration (CLDC) for the Java ME environment.

The PCEA application have different interfaces that help the user to register and select their preference and also a setting that allow a user to enable or disable the access of their personal data or information like location and schedule by the application. The users can edit their preference or their setting by login in to the application. Generally, on the mobile side of the user there are, four functionalities performed those are account creation, event preference selection,

setting preference selection and publish event information. Below we present each functionalities user interface from the user perspective.

A. Account registration

The application welcome page has welcome note and short knowhow information about what the system do for the user. This form has three menus sign up, sign in and exit as shown in Figure 6.1.

Sign up

User of the system should sign up using the sign up form that is show in Figure 6.2 in their first use of the application in order to have an account of their own as well to feed their preference to the application. While signing up, if the new entered user name does not exist in the data base the web server will send a confirmation message and let the user to access the application.

Figure 6.1: Welcome page and its menu

Figure 6.2 Sign up Interface

Signing In

Users can sign in to the application using an account that they create when they first install and use the application. While signing in, the web server will send a response message to the user, if the user exists it will send a welcome message and it will lead the user to the option page but if the user entered a wrong user name or password it will send a try again message.

Figure 6.3 Sign in interface

Exit: this command help to close and exit the application.

B. Event Preferences selection

User event preferences are event type categories that the user would prefer to be informed about. These preferences are entertainments like movie, music, Art, festivals and sport, list of cuisine type of food and drinks and shops/mall. Figure 6.4 shows the event type movie preference page. Here the user select genre of movies they want to be informed about.

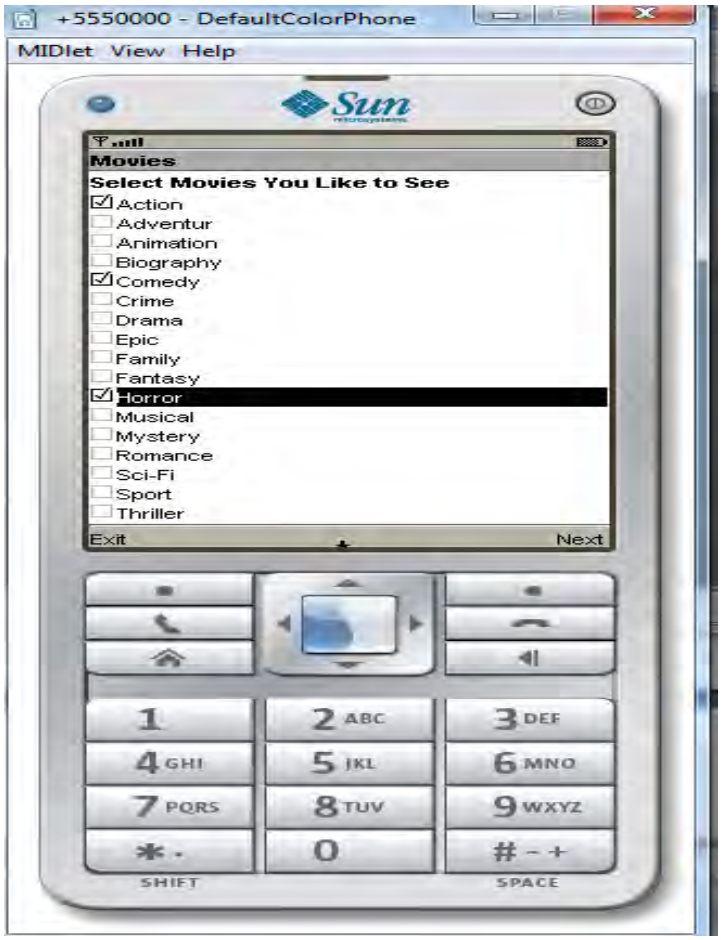


Figure 6.4: Event movie preference selecting page

Other preference categories are shown in Figure 6.5

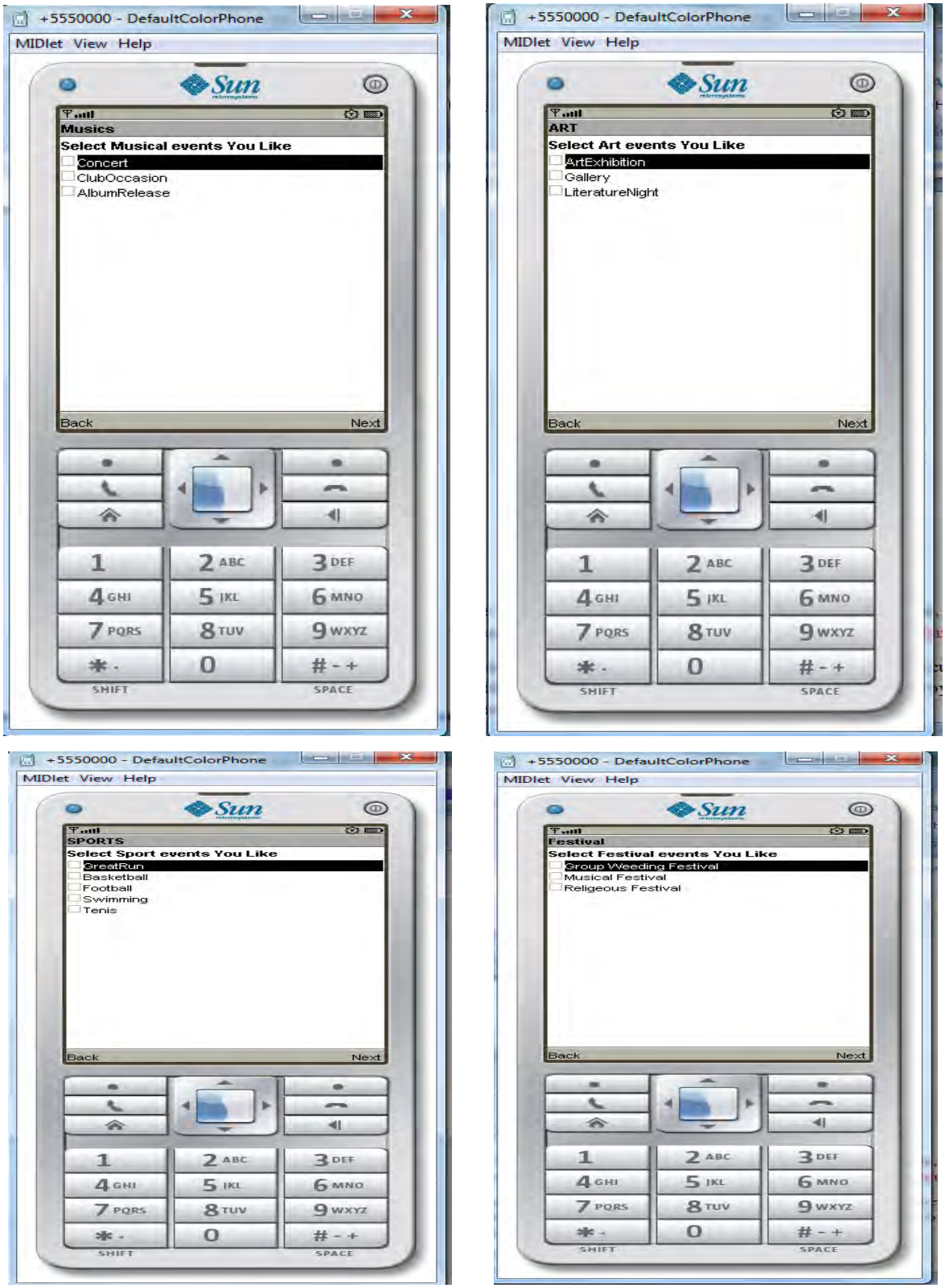


Figure 6.5: Interface of the user preferences

Food: Here, users select cuisine type like Ethiopian, Mediterranean, Armenian, Mexican, Continental, European, Chinese, French and Italian food they like as shown in Figure 6.6. The application will suggest restaurants that serve cuisine or foods that match with the cuisine preference of the user if location of the user is around location of the restaurant as well by considering the time i.e. breakfast time, lunch time as well dinner time.

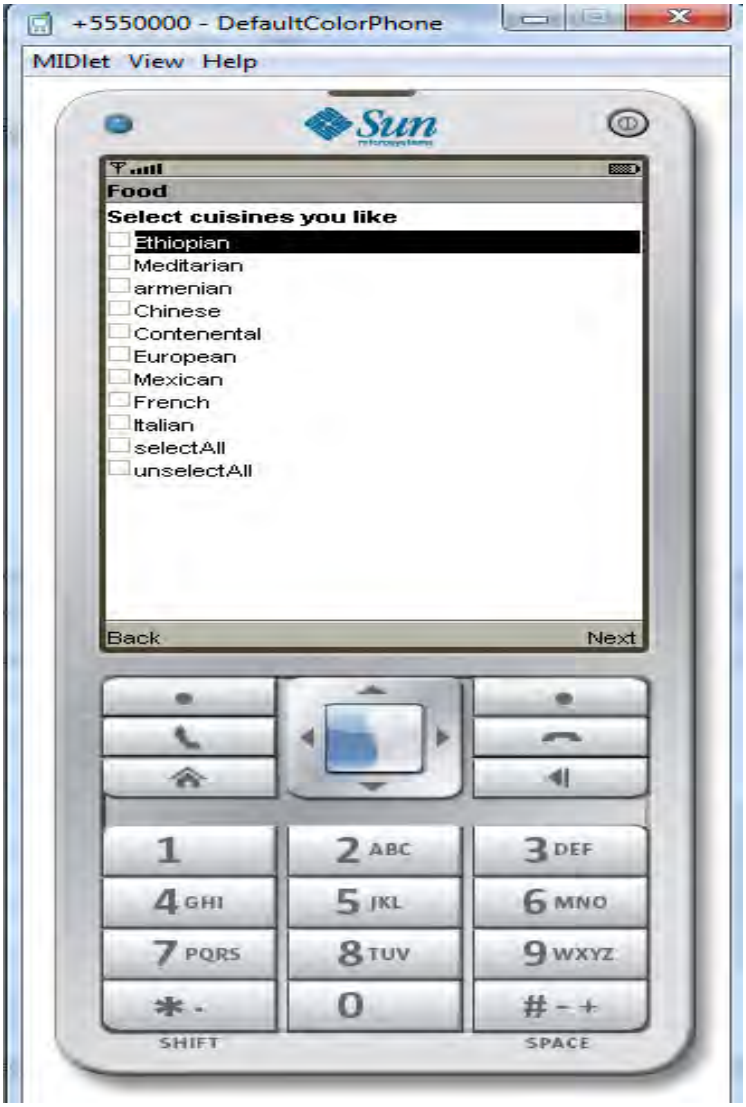


Figure 6.6: Food cuisine type preference selecting form

Shop/Malls: here a service of point of interest performed. If user check or select shop or malls, the application will suggest malls that are found near or around the location of the user. The interface for selecting shop or malls is as shown in Figure 6.7.

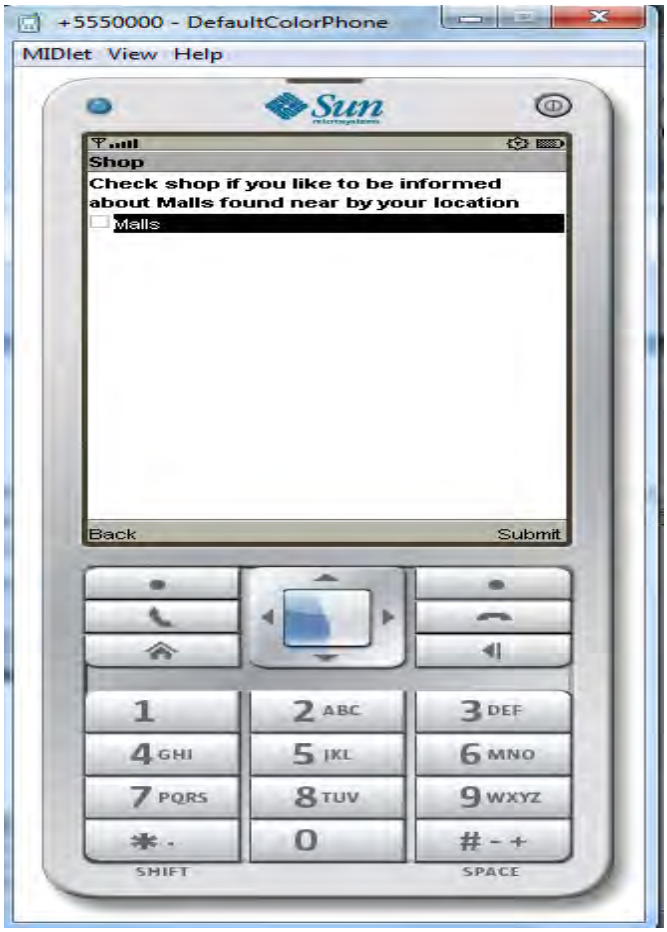


Figure 6.7: Shop/Mall selection form

C. Setting preference selection

After signing up or sign in the user can set whether the application can read their dynamic data from the mobile phone or not using the setting command as shown in Figure 6.8.

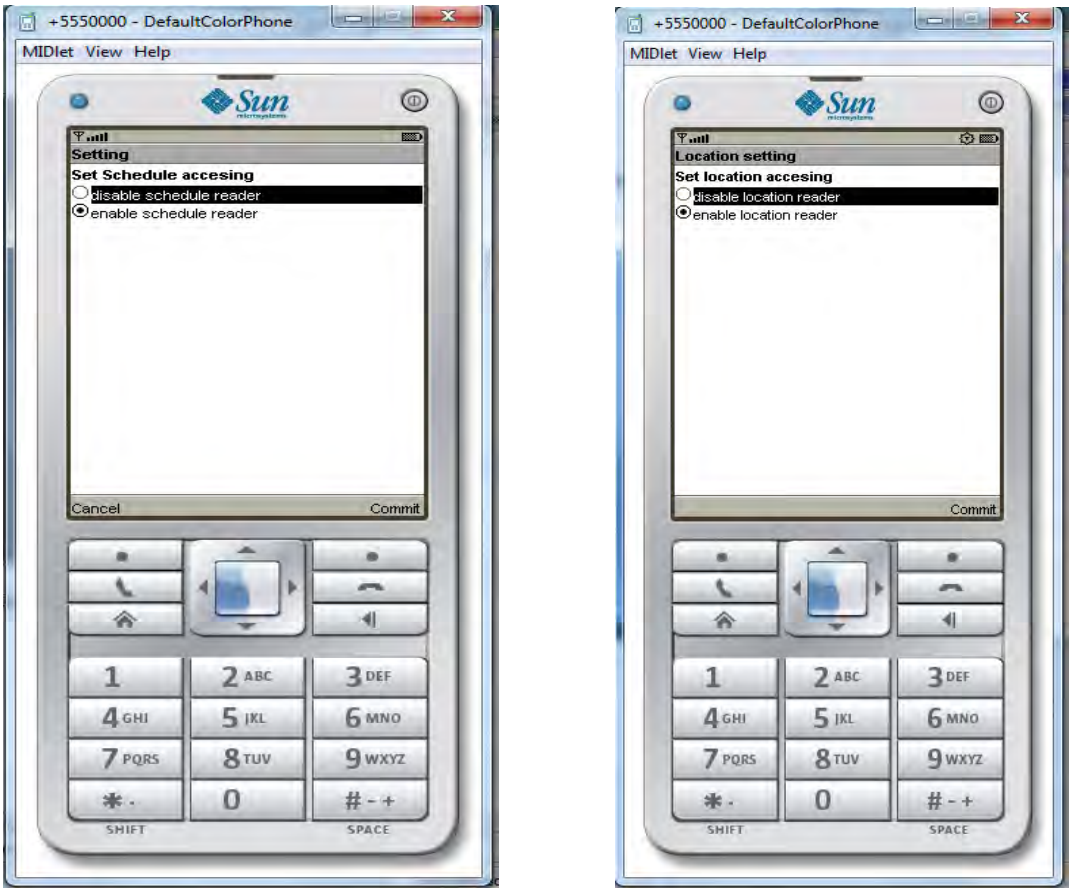


Figure 6.8: Dynamic data reading setting page

D. Publish

Additional to users selecting their own preference to get service from the event provider they can also share event information on hand for other users using the publishing page.

When a user wants to publish information that he/she knows and thought that is useful they can select the publish command and fill in the required event information and then publish it to the server as shown in Figure 6.9. Here we assume each user as a trusted user.

Figure 6.9: Info publication page

In order for the application to be alive the above interfaces i.e. the account registration, the event preference selection, setting preference selection and publish must have some functionality that is performed behind them. Below we present the functionalities

- **Account registration**

  After getting the user basic information that are required to authenticate the user the data will be send to the server in order to be saved and the server come up with the response it get from the database and send it to the mobile phone.

- **Event preference Selection**

  Here the selected checkboxes value will be send to the server in order for them to be saved for latter retrieval.

- **Setting preference Selection**

  Here two basic functions performed.

1. Schedule reading

In order for the application to get the schedule of the user and provide a more context aware application, the user should let the application to access the mobile phone local calendar.

J2me has *PIM API* that is an optional package defined in the Java Specification Request (JSR) 75.The PIM API allows applications to manage PIM data that is stored on the handset. The PIM database hold Contact list, event list and to do list data. In our case by using the PIM API we access the mobile phone event list database and retrieve the stored event date values as shown in Figure 6.10. From this data we get the user busy date.

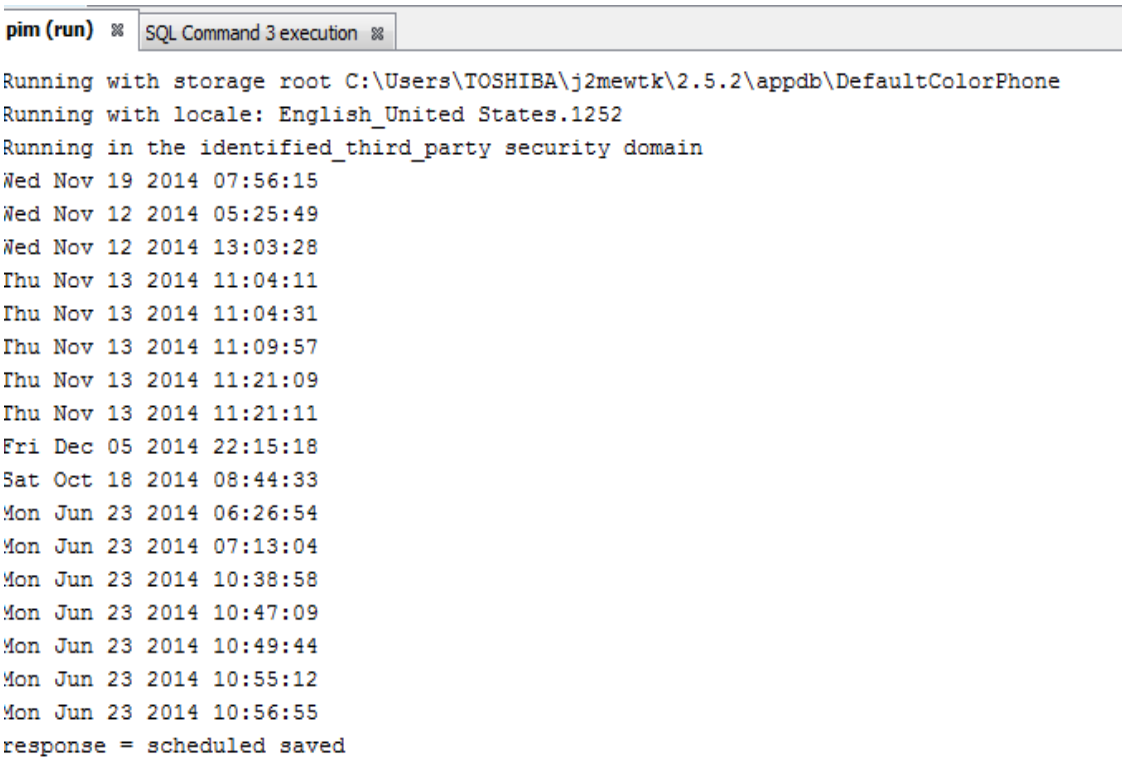


Figure 6.10: list of busy or scheduled date and time of the user extracted from the user mobile phone calendar

2. location reading

Many mobile phones today support the Location API for J2ME under JSR-179. In order to build location based application a Java MIDP2.0 enabled and JSR 179 compliant mobile phone required. In theory, the JSR is able to obtain location data from a number of different technologies, such as cell triangulation, WI-FI triangulation or GPS signals. JSR-179 is

available for CDC and CLDC 1.1 since floating Point required. The main classes of the Location API are:

- LocationProvider
- Location and LocationListener
- Coordinates and PromixityListener
- LandMark and LandMarkStore

Like as shown in Figure 6.11 you can simulate a location provider and your current position using the external event generator by running an xml script.

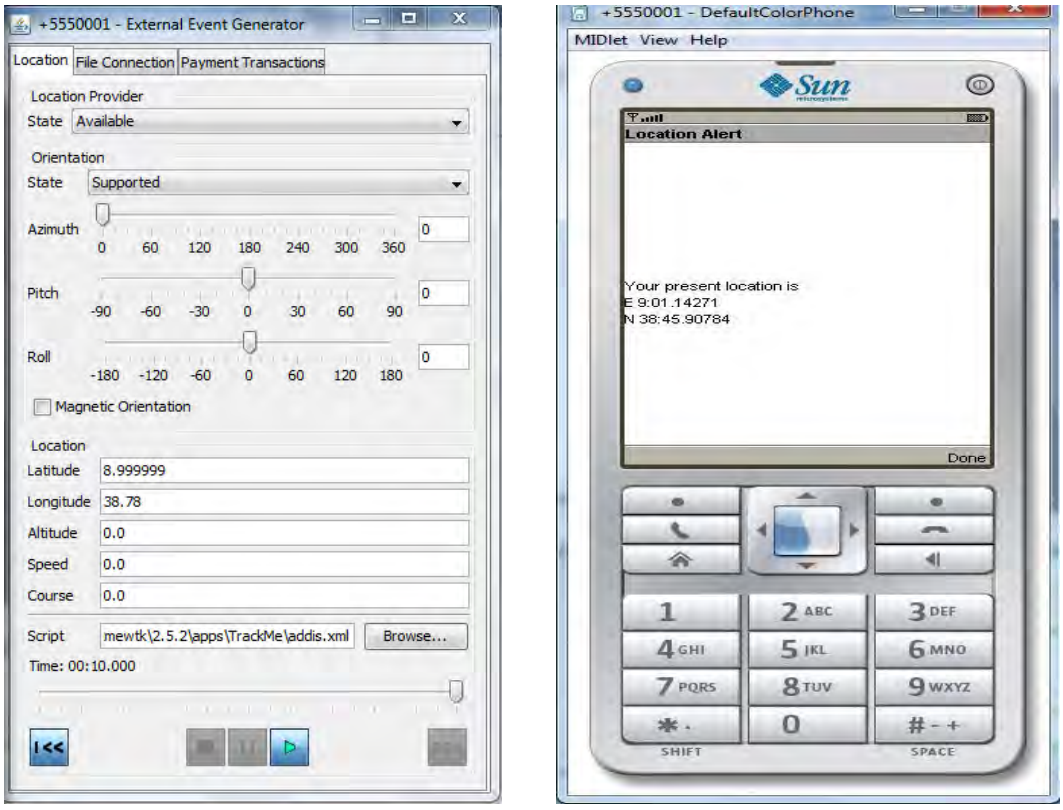


Figure 6.11: External Location Event Generator

In our application we need location information of the user in order to advise events that are found near or around the location of the user. Our location based service answer basically two questions i.e. where our client is? And what‟s around him/her? Here we try to get the location of the user using cell ID. A CellID is a number which is associated with a specific cell (the radio tower to which your handset is connected). In most cases, this is the closest tower to your location. So by knowing the location of this tower, then you can know approximately where the handset is. Even if location CellID accuracy is

lower than GPS accuracy because of a tower can cover a huge area, from a few hundred meters, in high density areas, to several kilometers in lower density areas we select it as our main location coordinate provider since we try to cover mobile phones that don‟t either have a built in GPS or a Bluetooth GPS. In order to access the location of the user we use preferred method of online cell ID and we import the Nokia mobile phone package com.nokia.mid.location.LocationUtill.

We can get online cell id data from OpenCellID.org. OpenCellID is the world‟s largest open-source project that collects GPS positions of cell towers for improved localization without GPS. It provides free worldwide database of Cell IDs, MCC (Mobile Country Code), MNC (Mobile Network Code), latitude, longitude and LAC (Location Area Identity). The unique location area identities of the cell towers can be collected by devices that utilize the wireless network provided by those cell towers. Every mobile phone or devices update everything in every minute incase if there is a change. In our case the mobile phone negotiate with the service provider using the GSM (Global System for Mobile communication) network or the telecommunication network. The service provider then traces the antenna coverage or the BTS (Base Transceiver Station) location. To identify each BTS or sector of a BTS within the LAC if not within a GSM network, it uses a GSM cell ID, the unique number of the BTS. After getting the Antenna coverage it map the information with the database of the OpencellID and return the coordinate or longitude and latitude value of the current location.

- **Publish event information**

Event Information provided by the user will be send to the server to be saved in the database for a latter use.

6.2.2 J2me Communicating with the Web Server

The mobile side of the application needs to communicate with the server in order to provide any advising services. J2me application will send a request to connect with the web server. If the j2me application communicating with the web server is successful we will get the HTTP/1.1 200 OK message.

6.2.3 Server Side of the application

In the server side of the application we do all the processing and providing of the services.

A. Web Server

In the web Server side all the methods that process the service are defined here. When committing some command on the j2me mobile side application, the commands call a tread and in the tread there is a code that will call the http post request and response method that pass the URL as a parameter. In the server side we use apache web server by initiating it using vertigo server.

B. Information extraction from web/ content mining from the web

What we do first is extracts the data from different websites that provide event information using web mining techniques, for the prototype we use movie sites of Edna mall Mati multiplex (http://www.ednamall.info/) and the Internet movie database/IMDB (http://www.imdb.com/) for extracting movie events and for the suggestion of point of interests (POI) specifically for restaurants we use (http://www.ethiopianrestaurant.com/) and for malls/shops we try to use the ET yellow page but since it is not well organized we store list of malls and supermarkets as a sample manually.

Since we are concerned only about the content on the websites, we use web content mining to extract the data. Web content mining is very similar to text mining and data mining the difference is that in web content mining the data is either semi-structured or unstructured whereas in data mining the data is more structured. Some of the methodologies used in data mining are used in the web content mining since both deals with information extraction.

Using web scraping/screen scraping we can convert the semi-structured data from a website to a more structured format that can be stored in a relational database. Since each website stores and presents the data differently, the data we require might not always be present in the same location, and the same method cannot be used to extract the data across various websites. Every web page should be analyzed to exactly determine the location of data. Some websites display information that is not required. It is critical to extract only the information that is required and scrap the rest of the data before storing the data in the database. Otherwise, it would involve a lot of overhead to process the data every time it is fetched from the database. Some of the techniques used to implement web scraping involve the use of a DOM/HTML parser.

There are different java libraries like html parser, Jtidy, Jaunt API, Jsoup, html unit and some others to do html parsing. For this project we used Jsoup. Jsoup is a java library that provides various methods to extract and manipulate the data from an HTML document. Jsoup reads the HTML document and parses it similarly to the DOM (Document Object Model) parser to identify various nodes. These nodes are represented in a tree structure, and this tree can be traversed and the required data can be extracted using various methods provided by Jsoup. Data can be extracted from that page and stored in the database in a structured format for future use. We choose Jsoup because it is well suited for web scraping and it works nicely on modern web applications that use Jquery and other Ajax features and use div tags extensively. It is also very forgiving if the target web applications have missing java scripts. Jsoup is a very efficient API and to use all its methods, we can import its jar file into our Java program. Figure 6.12 shows extracted movie data of cinema halls, show time and title from ”Edna mall Mati multiplex” show time html page.

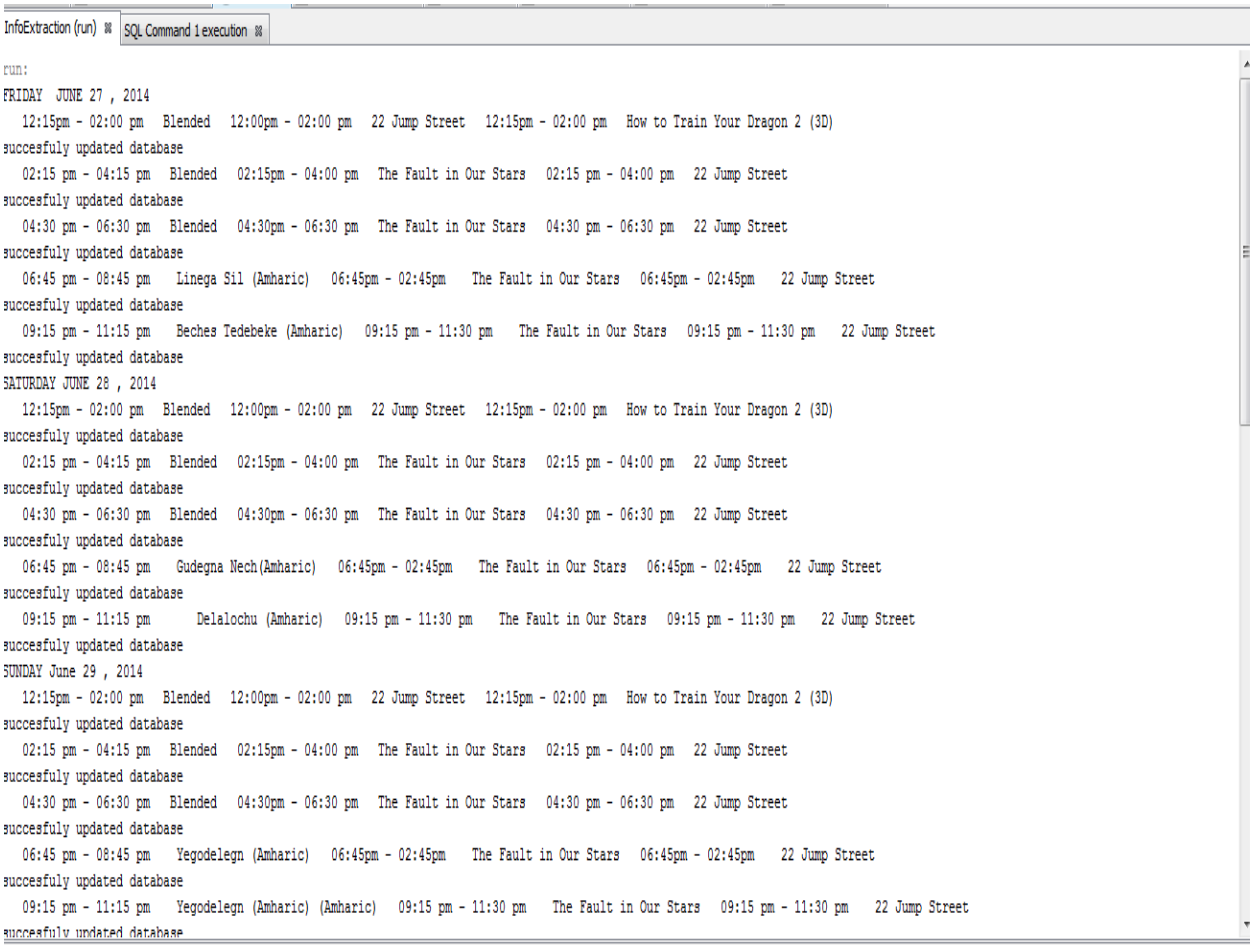


Figure 6.12 Jsoup outputs of extracted movie data from Mati multiplex

We also extract the IMDB database for having information of a movie director, actor, description and mainly genre. This information are required for suggesting the user since most found web sites of movie cinema halls did not provide the information and since user select their preference of movie by genre.

```
loopCounter 1
url to download: http://www.imdb.com/search/title?at=0&sort=moviemeter&title_type=feature&year=2014,2014&start=1
Movie [2] Title: The Maze Runner
Movie [2] Description: Thomas is deposited in a community of boys after his memory is erased, soon learning they"re all trapped in a maze that will require him to join forces with fellow
"runners" for a shot at escape.
Movie [2] Directors: Wes Ball Dylan O'Brien Kaya Scodelario Will Poulter
Movie [2] Genre: Action | Mystery | Sci-Fi | Thriller
Movie [2] Length: 113 mins.
succesfuly updated database
Movie [3] Title: The Hunger Games: Mockingjay - Part 1
Movie [3] Description: Katniss Everdeen is in District 13 after she shatters the games forever. Under the leadership of President Coin and the advice of her trusted friends, Katniss spreads her
wings as she fights to save Peeta and a nation moved by her courage.
Movie [3] Directors: Francis Lawrence Jennifer Lawrence Josh Hutcherson Liam Hemsworth
Movie [3] Genre: Adventure | Sci-Fi
Movie [3] Length: ***UNKNOWN***
succesfuly updated database
Movie [4] Title: Guardians of the Galaxy
Movie [4] Description: A group of space criminals must work together to stop a fanatic from destroying the galaxy.
Movie [4] Directors: James Gunn Chris Pratt Vin Diesel Bradley Cooper
Movie [4] Genre: Action | Adventure | Sci-Fi
Movie [4] Length: 121 mins.
succesfuly updated database
Movie [5] Title: The Giver
Movie [5] Description: In a seemingly perfect community, without war, pain, suffering, differences or choice, a young boy is chosen to learn from an elderly man about the true pain and pleasure
of the "real" world.
Movie [5] Directors: Phillip Noyce Brenton Thwaites Jeff Bridges Meryl Streep
Movie [5] Genre: Drama | Sci-Fi
Movie [5] Length: 97 mins.
succesfuly updated database
Movie [6] Title: Tusk
Movie [6] Description: When podcaster Wallace Bryton goes missing in the backwoods of Manitoba while interviewing a mysterious seafarer named Howard Howe, his best friend Teddy and
girlfriend Allison team with an ex-cop to look for him.
Movie [6] Directors: Kevin Smith Justin Long Michael Parks Haley Joel Osment
Movie [6] Genre: Comedy | Drama | Horror
Movie [6] Length: 102 mins.
succesfuly updated database
Movie [7] Title: Edge of Tomorrow
Movie [7] Description: An officer finds himself caught in a time loop in a war with the alien race. His skills increase as he faces the same brutal combat scenarios, and his union with a Special
Forces warrior gets him closer to defeating the enemy.
```

Figure 6.13: Extracted information from the IMDB website page

C. Matching

All the event information extracted from the user and the event provider and the preference information of the user must be matched to advise the user about events.

D. Notification Manager

When every time event information that match with the specified context parameter is found the matching manager will invoke the notification manager to start. Notification manager accept the event information and push the message to the client.

In J2me push technology included in the MIDP 2.0 (push registry API in MIDP 2.0). By this technology we can to activate a midlet remotely, beside the AMS (Application Manager System) that is responsible for installing, managing, starting and stopping an application. There are 3 ways to start a midlet

- User request
- Incoming network connection and
- Schedule alarm

In this application we use the Incoming network connection. In order for a midlet to be notified of an incoming network connection, it must register these requests with the AMS and then the AMS monitor the port activity. The midlet will listen to the message through specified connection type and port number. Figure 6.14 shows a coming event notification.

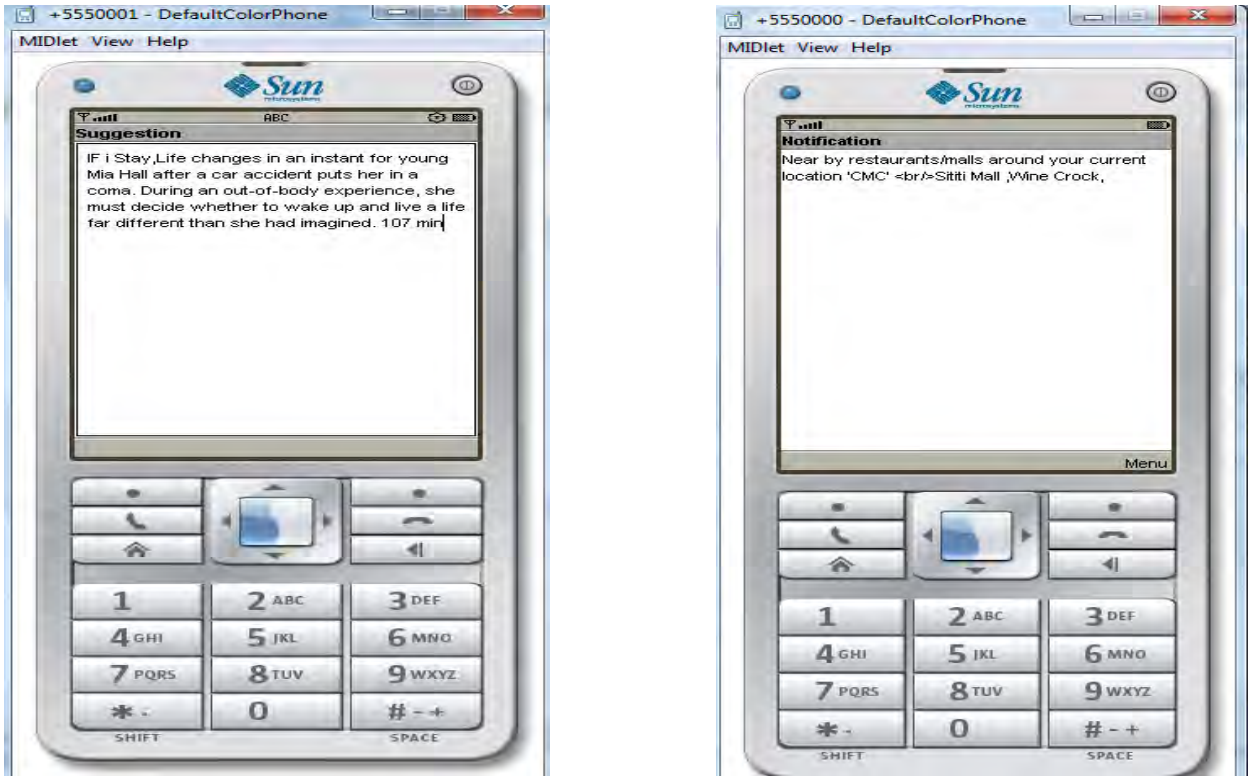


Figure 6.14: Notification

6.3 Prototype Usability Test

Usability testing tests the software features that are how easy it is to use the software, how easy it is to learn the software and how convenient is the software to end user. The unique features of mobile devices and wireless networks pose a number of significant challenges for examining usability of mobile applications, including mobile context, multimodality, connectivity, small screen size, different display resolutions, limited processing capability and power, and restrictive data entry methods. Because of the above challenges traditional guidelines and methods used in usability testing of desktop applications may not be directly applicable to a mobile environment. Therefore, it is essential to develop and adopt appropriate research methodologies and tools to evaluate the usability of mobile applications [28].

6.3.1 Design and usability qualities taken into consideration

Since we are mainly concerned about the application, general objective that is developing a mobile phone based pervasive context-aware event advisor, that performs the core functionality of the application, we did not emphasize much on the user interface (UI) of the application that may impose a major impact on the user experience beside the challenges posed by the feature of the mobile devices and the wireless network. Some design and usability qualities taken into consideration are:

- Menus and buttons are clearly and consistently labeled to help the user navigation, learnability and memorability.
- To minimize cognitive load back track and easy access to earlier pages is supported.
- We try to typically fit content page information on one screen to avoid scrolling even though this is not succeeded on some event categories, publishing and sign up page that require detail information.

6.3.2 Usability evaluation and testing of the mobile application prototype

Usability testing (or GUI navigation), focuses on the external interface and the relationships among the screens of the application. In usability testing basically the testers test the ease with which the user interfaces can be used. We perform a usability test on the application prototype to identify whether the application built is user-friendly or not and to address the several question that is needed for designing the application. We plan our test using a 1 page usability test plan

dashboard as shown in (Annex B). We follow the proposed framework for the design and implementation of usability testing of mobile applications by [28] as shown in Figure 6.15.

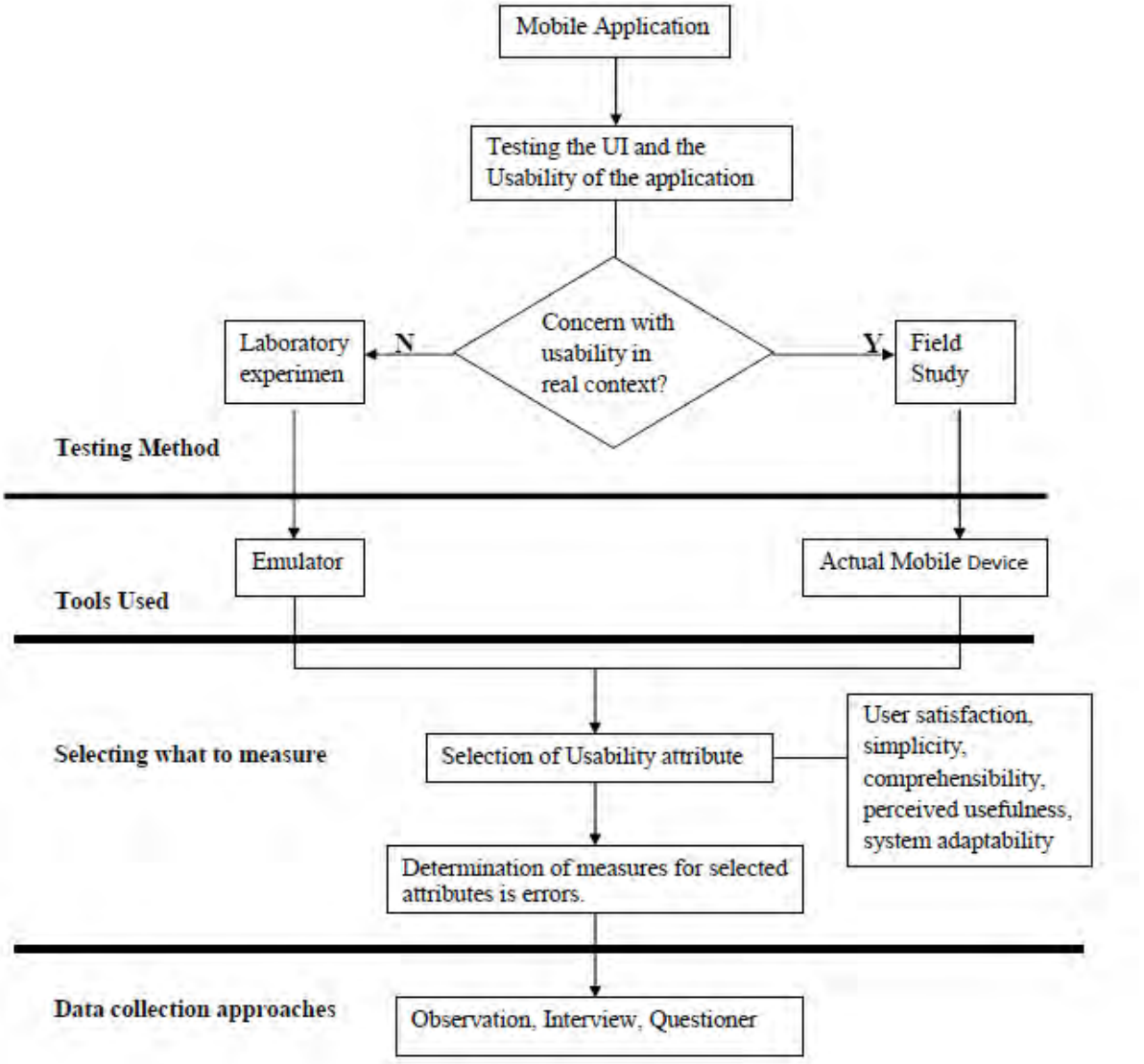


Figure 6.15:A framework for the Design and Implementation of Usability Testing of Mobile Applications (modified to test PCEA)

The usability tests have been performed both in laboratory experimental environment and with field studies. In the experimental environment, human participants are required to accomplish specific tasks using a mobile application in a controlled laboratory setting, while a field study allows users to use mobile applications in the real environment.

To perform the usability test we select five participants (n=5) ranged from 24 to 45 (mean age=34.6 years), three of whom were women and two men. According to Nielsen [25] five participants will discover 80% of the problems in a system. He argued the best results come from; testing no more than 5 users and elaborating usability tests are just a waste of resources. Nielsen and Landauer [9] showed that the number of usability problems found in a usability test with *n* users using $N(1-(1-L)^{n})$ equation where *N* is the total number of usability problems in the design and *L* is the proportion of usability problems discovered while testing a single user. Every time the number of users increased there is something new you find out and also there are some overlaps in what you learn. The overlaps increase when the number of the users increased and after the fifth user, you are wasting your time by observing the same findings repeatedly but not learning much new. In any case, a small amount of users, that is, generally fewer than 10 subjects, is sufficient for any formative evaluation of usability [20].

First, we conducted a laboratory experiment to evaluate the user interface. In the experiment, an emulator on laptop computers was used. The tasks given to participants were to 1) Sign up into the application, 2) select their setting and event preference 3) sign in into the application and update their preferences, 4) publish event information to the system and 5) check if there is any event information pushed for them according to their preference. Participation in the study lasted approximately one hour and was conducted in our department computer laboratory. It consisted of the series of tasks that we mention above. All participants were tested individually. After being welcomed by the experimenter, participants were told that they were to take part in a usability test and were to work with a prototype of a mobile phone based pervasive context-aware event advisory application in short PCEA or event advisor. The user interviewed and answers a questionnaire (Annex C) after finishing the tasks. We also collected comments on the prototypes and preference data and evaluations in the form of the PCEA usability test questionnaire completed by the users after the test. Any user action that did not lead to the successful completion of a task we defined as error. Errors were classified into two main categories, navigation errors and comprehension errors. Navigation errors occurred when participants didn‟t move as expected. Comprehension errors occurred when participants didn‟t understand the design of the interface. Table 6.2 shows the types of errors.

Table 6.2 Types of errors

Types of Errors	No. of errors occurred
Navigation	3
Comprehension	2
Total	5

To test the UI of the application we use a black box testing or input/output driven testing techniques. The output result point out, 98% of the task performed in the testing session meet the expected result.

The captured data are generally informative and useful for analyzing user performance and finding faulty designs of applications that frustrate users. However, using emulators omits some important aspects of actual mobile devices and mobile context. For example, it alleviates the problems of long transmission latency caused by limited bandwidth in real wireless networks, inefficient input mechanisms, and the changing wireless environment, potentially leading to untruthful user perception and satisfaction. Generally the laboratory experimental usability test performed in a controlled and specific environment that ignores mobile context (e.g. location) and unreliable connection of wireless networks. So, we perform a field study to evaluate the usability of the application in the real environment. For the field study we use the same participants that participate in the laboratory experiment test. The participants were asked to use the mobile application on an actual mobile device that support j2me platform specifically Nokia mobile device and to perform the five tasks mentioned above and to change their location for the experiment purpose.

During the usability tests to measure the quality of the application we consider qualitative usability attributes out of the nine generic usability attribute i.e. learnability, efficiency, memorability, errors, user satisfaction, effectiveness, simplicity, comprehensibility, learning performance and the additional commonly used attributes such as user perceived usefulness and system adaptability.

Participants were asked to complete an SUS (System Usability Scale) questionnaire, so as to explore their experiences when interacting with the prototypes. A crucial feature of the SUS lies in the fact that asks the user to evaluate the system as a whole, rather than specific aspects. All 10 questionnaire statements having been processed, the overall SUS score for each participant

presented in Table 6.7. To calculate the SUS score, first we summed the score contributions of the items 1, 3, 5, 7 and 9. The score contributions of these items are their scale position (the user response) minus one. We then summed the score contributions of the other items: five minus their scale position. Finally, we multiplied the sum of the scores by 2.5, to obtain the overall score with a range between 0 to 100.

Table 6.3 Overall SUS Score

Participants	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	SUS score
P1	3	4	3	4	3	3	3	3	3	4	82.5
P2	2	3	2	3	2	3	3	3	2	3	65
P3	4	4	3	3	3	4	3	4	3	3	85
P4	3	3	3	4	2	3	3	3	3	4	77.5
P5	4	4	2	3	2	4	3	3	3	4	80
Mean											**78**

The survey results showed the overall satisfaction. Sauro [21] reports that a mean value over 74 is level B, value above 80.3 is level A. An average value of below 51 is level F (fail). The PCEA prototype with an average value of 78 belongs to level B.

During the test the qualitative usability attribute have been evaluated through interviewing the participants:

- User satisfaction: the attitude of users towards using the application
- Simplicity: the degree of comfort with which users find a way to accomplish tasks
- Comprehensibility: how easily users can understand content presented on the mobile device
- Perceived usefulness: to what extend the application has met its implementation objectives
- System adaptability: how the system adapts to the user requirements.

Out of the five participants the two users mentioned they did not receive any notification. In other hand the three participants get notifications and one of the participants mentioned he received the restaurant and mall notification after he already changes his current location.

The feedback from the participants on the event advisor application prototype was in overall positive, although some participants also gave critical comments. Participants tended to believe the overcrowded pages will be considered as burden if the user is filling them while walking or doing some other activities and they also point out this may not cause much problem since most information providing to the system is done in the first use of the application. They also mention if there was a method the user query about entertainment and point of interests whenever they want to beside the application search for them they believe doing this make them feel they got much control on the application.

CHAPTER 7: CONCLUSION AND FUTURE WORKS

7.1 Conclusion

In this work, we specify the system requirement and analysis, the system design and a prototype of a context aware event advisor application that will advise or suggest the user what he/she prefers with minimum destruction. The proposed architecture introduces a context aware event advisor that utilizes the static and dynamic context of the user and the event information. The application takes advantage of the web as well as the users themselves to provide the event information required for the advising purpose.

The application retrieves and acquires the user schedule and the user location from the user mobile phone and because of J2me mobile phone manufacturers impose some accessing restriction on the phone personal information management database it impose difficulties in retrieving the content of the calendar value when it comes to real phone application testing. The prototype has a context analyzing and matching component to analyze the user contextual data as well as the event information data to filters out the suitable event information to provide personalized suggestions.

We perform a usability test both in laboratory experimental environments and with field studies to identify whether the application built is user friendly or not and to address the several questions that is needed for designing the application. We conduct an SUS (System Usability Scale) questionnaire, so as to explore the users experience when interacting with the prototype or the user satisfaction in general and we get a level B, above 74 usability scale score.

7.2 Future Work

By acquiring the schedule of the user we only analyze the date value inserted. Doing this only allows us to identify if the user is busy or free on that specific date but this will not show what exactly the user is doing or having, and we believe that knowing this kind of specific information help to be exact what the user is doing or his/her location and will maximize the context awareness of the application. As a future work we suggest if a developer uses natural language processing in order to get the scheduled content.

Also accessing the calendar/schedule of the user can also be used as another advantage of getting event information. This is by extracting schedule of the user from the calendar and filters out

event specific information and uses it for suggesting other users about that event. The other thing we allow users to publish event information that he/she knows so that it reaches other users interested in that specific event information i.e. the user that would like to be suggested about it. But in this research work, we assume and consider each user of the application as a trusted user that will not broadcast false information and this can‟t be always the case so we suggest as a future work to make more out of the application by including some mechanism like [59] use to eliminate traffic to eliminate spam event information from reaching the other users. Here, security and privacy issue are not dealt and we advise in the future they must be dealt with specific detail.

REFERENCES

[1]C. Biancalana, F. Gasparetti, A. Micarelli, and G. Sansonetti. "An approach to social recommendation for context-aware mobile services."*ACM Transactions on Intelligent Systems and Technology (TIST),* volume 4, no. 1, pp.10, 2013.

[2]C. Sapateiro, N. Baloian, P. Antunes, and G. Zurita. "Developing collaborative peer-to-peer applications on mobile devices." In *Computer Supported Cooperative Work in Design, 2009. CSCWD 2009. 13th International Conference on*, pp. 396-401. IEEE, 2009.

[3]M. Raento, A. Oulasvirta, R. Petit, and H.Toivonen. "ContextPhone: A prototyping platform for context-aware mobile applications." *Pervasive Computing, IEEE, volume* 4, no. 2, pp. 51-59, 2015.

[4]M. Satyanarayanan, "Pervasive computing: Vision and challenges." *Personal Communications, IEEE* volume 8, no. 4, pp. 10-17, 2001.

[5]G .Chen , and D. Kotz. "*A survey of context-aware mobile computing research"*.Vol. 1, no. 2.1. Technical Report TR2000-381, Dept. of Computer Science, Dartmouth College, 2000.

[6]J. Zhu, P. Chen, H. Pung, M. Oliya, S. Sen, and W. Wong. "Coalition: A platform for context-aware mobile application development." *UbiCCJournal* , pp. 722-735, 2011.

[7]P.Kopipad, J.Mantyjarvi, J.Kela, H.kerane and E.Juhani Malm, VTT technical research center, Finland, 2003.

[8]R.devaul, J.Gips and M.Sung, "The context aware cell phone project", MIT Media Lab, 2001.

[9] J. Nielsen, and T. Landauer, "A mathematical model of the finding of usability problems," *Proceedings of ACM INTERCHI'93 Conference* (Amsterdam, The Netherlands, 24-29 April 1993), pp. 206-213.

[10] A.K. Dey and G.D. Abowd, „Towards a better understanding of context and context-awareness", Proceedings of the Workshop on the What, Who, Where, When and How of Context-Awareness, ACM Press, New York, 2000.

[11] G. Adomavicius and A. Tuzhilin, Context-Aware Recommender Systems, University of Minnesota and New York University, 2011.

[12] B. Schilit , N. Adams, and R.Want. "Context-aware computing applications."In *Mobile Computing Systems and Applications, 1994.WMCSA 1994. First Workshop on*, pp. 85-90. IEEE, 1994.

[13] G. Chen, and D. Kotz. "*A survey of context-aware mobile computing research"*.Vol. 1, no. 2.1. Technical Report TR2000-381, Dept. of Computer Science, Dartmouth College, 2000.

[14] M. Weiser, "The computer for the 21st century."*Scientific american* 265, no. 3, pp. 94-104, 1991

[15] A. Asthana, M.Cravatts, and P.Krzyzanowski, An indoor wireless system for Systems and personalized shopping assistance. In Proceedings of IEEE Workshop on Mobile Computing Applications, Santa Cruz, California, December 1994.

[16] H. Yan, and T. Selker. "Context-aware office assistant." In *Proceedings of the 5th international conference on Intelligent user interfaces*, pp. 276-279. ACM, 2000.

[17] W. Schwinger., C. Grün, B. Pröll, W. Retschitzegger, and A. Schauerhuber. "Context-awareness in mobile tourism guides–A comprehensive survey." *Rapport Technique.Johannes Kepler University, Linz*, 2005.

[18] D. Zhang, Z. Yu, and C.Yau Chin. "Context-aware infrastructure for personalized healthcare."*Studies in health technology and informatics,s* volume 117, pp.154-163,2005.

[19] A.Wadood, A.Nadeem, N.Khan and H.Khan, Mobile Development – Discovery Document, Zigron, INC., 2006-2008.

[20]H. Petrie and N. Bevan. "The evaluation of accessibility, usability and user experience" *In: The Universal Access Handbook, C Stepanidis (ed), CRC Press*, 2009, pages 299–315

[21] J. Sauro, A practical guide to the System Usability Scale (SUS): Background, benchmarks & best practices. Denver, CO: Measuring Usability LLC.201.

[22]M. Weiser, and J. Brown. "The coming age of calm technology."In *Beyond calculation*, pp. 75-85.Springer New York, 1997.

[23] R. Katz, D. Long., M. Satyanarayanan , S. Tripathi, Workspaces in the Information Age, In Report of the NSF Workshop on Workspaces in the Information Age, Leesburg, VA, October, 1996.

[24] M. Satyanarayanan, "Fundamental challenges in mobile computing."In *Proceedings of the fifteenth annual ACM symposium on Principles of distributed computing*, pp. 1-7.ACM, 1996.

[25] J.Nielsen, “Why You Only Need to Test With 5 Users”. Jakob Nielsen's Alertbox, March19, 2000.

[26] T.Moron and p. Dourish, “Special Issue of Human-Computer Interaction on context aware Computing”, IBM Almaden Research Center and University of California, Irvine, Volume 16, 2001.

[27] D. A Norman, “The invisible computer”. MA: MIT Press, Cambridge, 1998.

[28] D. Zhang, B. Adipat, Challenges, Methodologies, and Issues in the Usability Testing of Mobile Applications, International Journal of human-computer, 2005.

[29] A. K Dey, and J. Häkkilä. "Context-awareness and mobile devices."*User interface design and evaluation for mobile technology volume 1,* pp.1205-217,2008

[30] M. Baldauf, S. Dustdar, and F. Rosenberg. "A survey on context-aware systems."*International Journal of Ad Hoc and Ubiquitous Computing* volume 2, no. 4, pp. 263-277,2007

[31] A. Singh and M. Conway, “Survey of Context aware Frameworks – Analysis and Criticism”, UNC-Chapel Hill ITS, University of North Carolina, 2006.

[32] T. Winograd, "Architectures for context."*Human-Computer Interaction* volume 16, no. 2, pp. 401-419,2001.

[33] A. Kushwaha and V. Kushwaha, "Location based services using android mobile operating system." *International Journal of Advances in Engineering &Technology* volume 1, no. 1, pp. 14-20,2011.

[34] S. Yau, F. Karim, Y. Wang, B. Wang, and Sandeep and K.S. Gupta "Reconfigurable context-sensitive middleware for pervasive computing."*IEEE Pervasive Computing,* volume 1, no. 3, pp. 33-40, 2002.

[35] T. Gu, H. Keng Punga, D. Qing Zhang. "A service-oriented middleware for building context-aware services."*Journal of Network and computer applications* volume 28, no. 1, pp. 1-18,2005.

[36] A. K.Dey and J. Häkkilä, "Context-awareness and mobile devices."*User interface design and evaluation for mobile technology* volume 1, pp. 205-217,2008.

[37] G. Kortuem, M. Bauer and Z. Segall, "NETMAN: the design of a collaborative wearable computer system." *Mobile Networks and Applications*volume 4, no. 1, pp. 49-58,1999.

[38] S. Long, R. Kooper, G.D. Abowd and C.G. Atkeson, "Rapid prototyping of mobile context-aware applications: The cyberguide case study." In *Proceedings of the 2nd annual international conference on Mobile computing and networking*, pp. 97-107.ACM, 1996.

[39] J. Pascoe, D.R. Morse and N.S. Ryan. "Developing personal technology for the field."*Personal Technologies*volume 2, no. 1, pp.28-36,1998.

[40] K. Cheverst, N.Davies, K.Mitchell and A.Friday. "The role of connectivity in supporting context-sensitive applications."In *Handheld and Ubiquitous Computing*, pp. 193-207.Springer Berlin Heidelberg, 1999.

[41] N. Davies, Nigel, Keith Cheverst, Keith Mitchell, and Adrian Friday. "Caches in the air': disseminating tourist information in the GUIDE system." In *Mobile Computing Systems and Applications, 1999.Proceedings.WMCSA'99. Second IEEE Workshop on*, pp. 11-19. IEEE, 1999.

[42] J. Healey , and R. Picard. "StartleCam: a cybernetic wearable camera." In *Wearable Computers, 1998. Digest of Papers. Second International Symposium on*, pp. 42-49. IEEE, 1998.

[43] R. Picard and J. Healey. "Affective wearables."*Personal Technologies* volume 1, no. 4, pp. 231-240,1997.

[44] A. Schmidt. "Implicit human computer interaction through context."*Personal technologies* volume 4, no. 2-3, pp. 191-199, 2000.

[45] W. Schwinger, C. Grin, B. Prll1, and W. Retschi tzegger, "A light-weight framework for location-based services." In *On the Move to Meaningful Internet Systems 2005: OTM 2005 Workshops*, pp. 206-210. Springer Berlin Heidelberg, 2005.

[46] R. Schmohl, and U. Baumgarten. "Context-aware computing: a survey preparing a generalized approach." In *Proceedings of the International MultiConference of Engineers and Computer Scientists*, vol. 1, pp. 744-750. 2008.

[47] T. Berners-Lee, R. Cailliau, World-Wide Web, C.E.R.N. CH - 1211 Genève 23 [An nvited talk at the conference: Computing in High Energy Physics 92, Annecy, France, 23-27 September 1992.

[48] T. Berners-Lee, R. Cailliau, A. Luotonen, H. Frystyk Nielsen and A.Secret, "the world wide web", Vol.37 no.8, communication of the ACM, , August 1994.

[49]G. Geleijns, "Information Extraction from the Web using a Search Engine", Philips Research Laboratories in Eindhoven, the Netherlands, 2008

[50] T.Strang and C.LinnhoffPopien, "A context modeling survey."In *Workshop Proceedings*. 2004.

[51]F. Ay, "Context Modeling and Reasoning using Ontologies", University of Technology Berlin, July 2007.

[52] M.Perttunen, J.Riekki and O.Lassila , "Context representation and reasoning in pervasive computing: a review." *International Journal of Multimedia and Ubiquitous Engineering, vol* 4, no. 4, 2009.

[53] D.R Mendoza, "Using Ontologies in Context-Aware Services Platforms", thesis for a Master of Science degree in Telematics from the University of Twente, Enschede, The Netherlands, November, 2003.

[54] C. Bettini, O.Brdiczka, K.Henricksen, J.Indulska, D.Nicklas, A.Ranganathan, D.Riboni,' "A survey of context modelling and reasoning techniques."*Pervasive and Mobile Computing* volume 6, no. 2, pp. 161-180, 2010.

[55] J. van Sinderen, T. van Halteren, M.Wegdam, B. Hendrik, E. Henk Eertink, ."Supporting context-aware mobile applications: an infrastructure approach."*Communications Magazine, IEEE* volume 44, no. 9, pp.96-104, 2006.

[56] A.Boytsov. "*Context reasoning, context prediction and proactive adaptation in pervasive computing systems"*, Department of Computer Science, Electrical and Space Engineering Luleå University of Technology, SE-971 87 Luleå, Sweden, June 2011.

[57] M.Miraoui, C.Tadj, C.ben Amar, "Architectural Survey Of Context-aware System In Pervasive Computing Environment", *Ubiquitous Computing and Communication Journal*, Volume 3 Number 3.

[58] P. Flor´een, M. Przybilski, P. Nurmi, J. Koolwaaij, A.Tarlano, M.Wagner, M. Luther, F. Bataille, M. Boussard, B. Mrohs, S. Lau, "Towards a Context Management Framework for MobiLife", *Proc. 14th IST Mobile & Wireless Summit* 2005 (2005): 20-28.

[59] F.Khoshnood, M.Mahdavi and M. Kiani sarkaleh, Designing a Recommender System Based on Social networks and location based system, University of Guilan, International Journal of Managing Information Technology (IJMIT) Vol.4, No.4, November 2012, Rasht, Iran.

[60] B.N.Schilit, N.ADAMS and R. WANT,"Context-aware computing applications", In Proceedings of the Workshop on *Mobile Computing Systems and Applications, 1994.WMCSA 1994. First Workshop on*, pp. 85-90. IEEE, 1994.

[61] H.Costa, B.Furtado, D.Pires, L.Macedo, A.Cardoso, "Context and intention-awareness in pois recommender systems." In *6th ACM Conf. on Recommender Systems, 4th Workshop on Context-Aware Recommender Systems, RecSys*, vol. 12, p. 5, 2012.

[62]B. Rao, AND L. Minakakis, "Evolution of mobile location-based services."*Communications of the ACM* volume 46, no. 12, pp. 61-65,2003.

[63] H. Al Tair, M. J. Zemerly, M. Al-Qutayri and M. Leida , ."Architecture for context-aware pro-active recommender system."*International Journal Multimedia and Image Processing (IJMIP), volume* 2, no. ½, pp. 15-133, 2012.

[64] G. Adomavicius, R. Sankaranarayanan, S. Sen, and A. Tuzhilin. "Incorporating contextual information in recommender systems using a multidimensional approach."*ACM Transactions on Information Systems (TOIS)* 23, no. 1, pp. 103-145, 2005.

[65] J. Indulska, J.Ma, L. Yang, T. Ungerer, and J. Cao," Location-based recommendation system using Bayesian user‟s preference model in mobile devices". *In Ubiquitous Intelligence and Computing, Eds. Lecture Notes in Computer Science Series*, vol. 4611. Springer Berlin / Heidelberg, 120–129, 2007.

[66] K. cheverst, N. Davies, k. Mitchell and A. Friday, "Experiences of developing and deploying a context-aware tourist guide: the GUIDE project." In *Proceedings of the 6th annual international conference on Mobile computing and networking*, pp. 20-31.ACM, 2000.

[67] M. Bohmer, M. Prinz, and G. Bauer, "Contextualizing mobile applications for context-aware recommendation." *adj. proc.Of Pervasive* 2010, 2010.

[68] Y. Lin, J. Jessurun, B. Vries, and H. Timmermans, "Motivate: Context-aware Mobile Application for Activity Recommendation", Eindhoven University of Technology Eindhoven, Netherland, 2011.

[69] T. Hofer, W. Schwinger, M. Pichler, G. Leonhartsberger, J. Altmann, "Context-awareness on mobile devices – the hydrogen approach", Proceedings of the 36th Annual Hawaii International Conference on System Sciences, 2002 pp.292–302.

APENDECIES

Annex A: Sequence Diagram

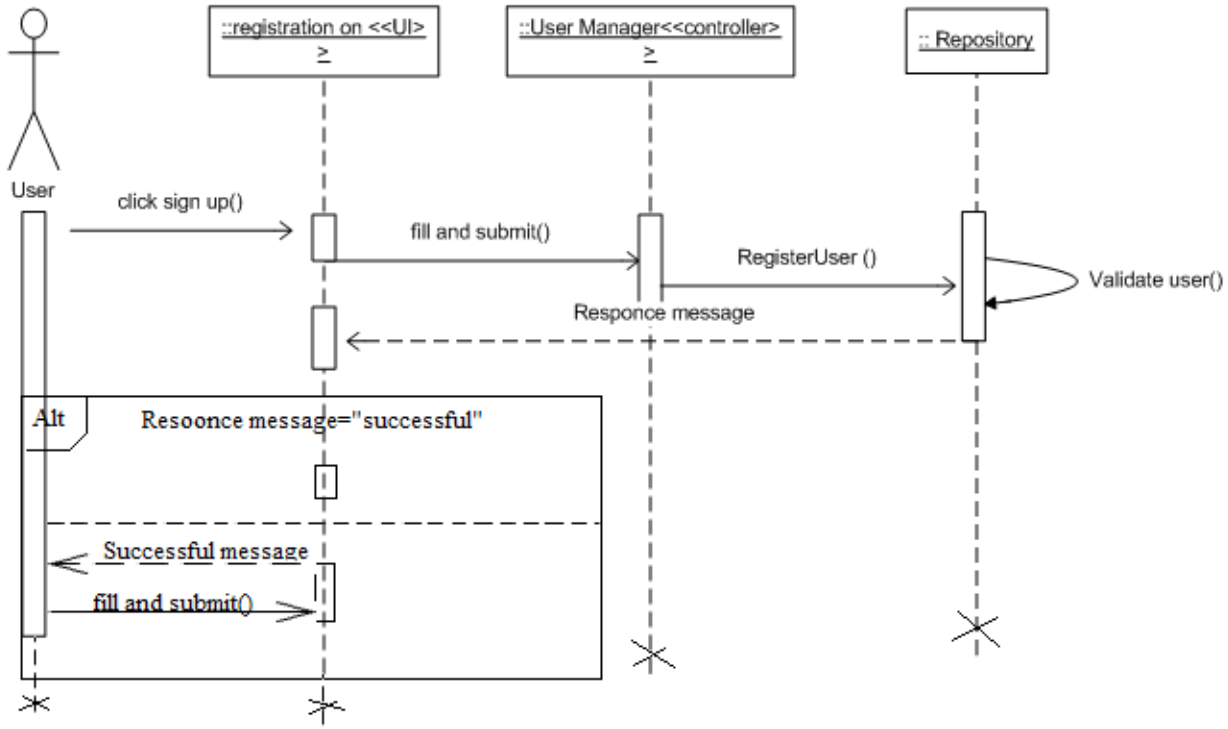


Figure A-1: Sequence diagram of users' registration

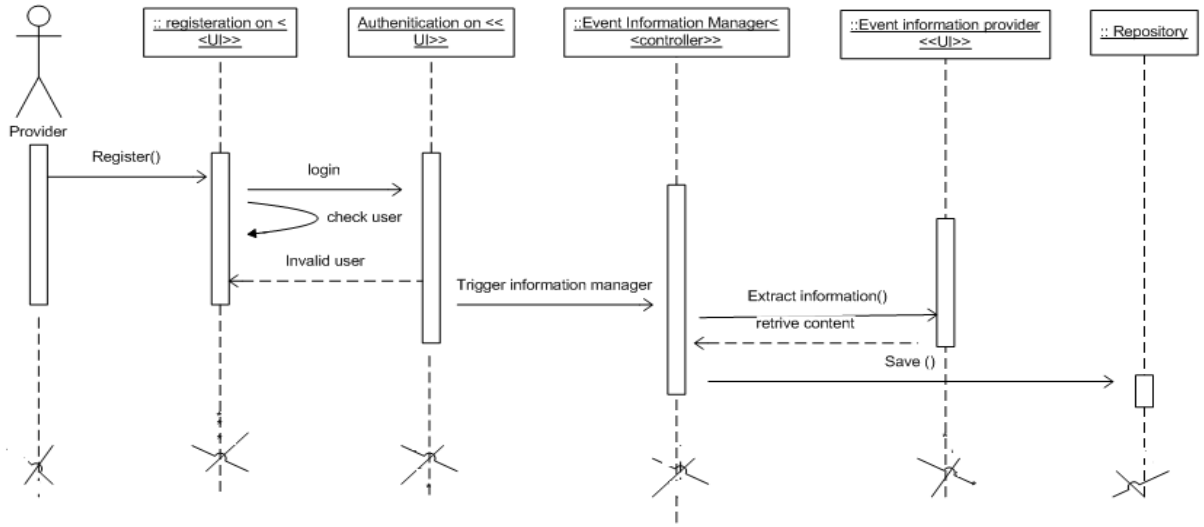


Figure A-2: Sequence diagram of information extraction

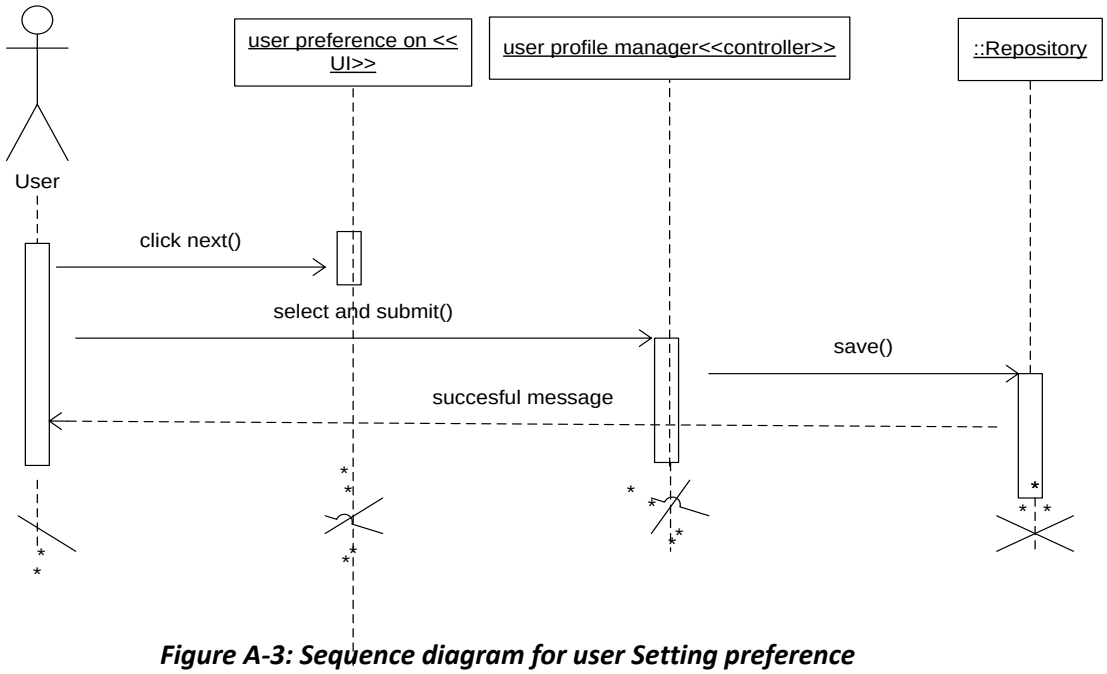
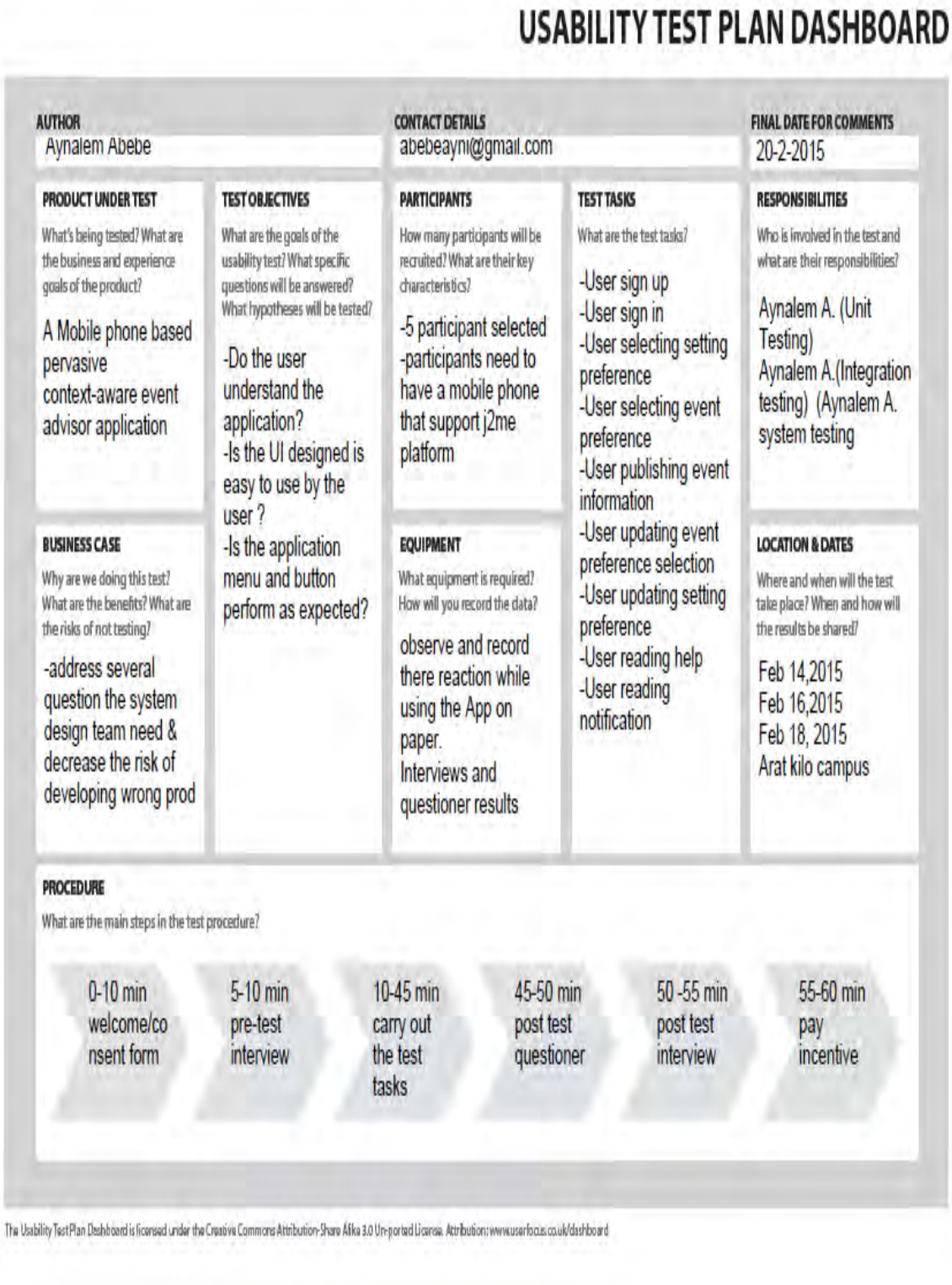


Figure A-3: Sequence diagram for user Setting preference

Annex B: Usability Test Plan Dash Board



USABILITY TEST PLAN DASHBOARD

AUTHOR
Aynalem Abebe

CONTACT DETAILS
abebeayni@gmail.com

FINAL DATE FOR COMMENTS
20-2-2015

PRODUCT UNDER TEST
What's being tested? What are the business and experience goals of the product?

A Mobile phone based pervasive context-aware event advisor application

TEST OBJECTIVES
What are the goals of the usability test? What specific questions will be answered? What hypotheses will be tested?

-Do the user understand the application?
-Is the UI designed is easy to use by the user ?
-Is the application menu and button perform as expected?

PARTICIPANTS
How many participants will be recruited? What are their key characteristics?

-5 participant selected
-participants need to have a mobile phone that support j2me platform

TEST TASKS
What are the test tasks?

-User sign up
-User sign in
-User selecting setting preference
-User selecting event preference
-User publishing event information
-User updating event preference selection
-User updating setting preference
-User reading help
-User reading notification

RESPONSIBILITIES
Who is involved in the test and what are their responsibilities?

Aynalem A. (Unit Testing)
Aynalem A.(Integration testing) (Aynalem A. system testing

BUSINESS CASE
Why are we doing this test? What are the benefits? What are the risks of not testing?

-address several question the system design team need & decrease the risk of developing wrong prod

EQUIPMENT
What equipment is required? How will you record the data?

observe and record there reaction while using the App on paper.
Interviews and questioner results

LOCATION & DATES
Where and when will the test take place? When and how will the results be shared?

Feb 14,2015
Feb 16,2015
Feb 18, 2015
Arat kilo campus

PROCEDURE
What are the main steps in the test procedure?

0-10 min welcome/consent form
5-10 min pre-test interview
10-45 min carry out the test tasks
45-50 min post test questioner
50 -55 min post test interview
55-60 min pay incentive

Annex C: Usability Test Questionnaire

*I would like to **thank you** for generously volunteering your time to participate in this usability testing. Your input will be **invaluable** in the development of the Application PCEA (Mobile phone based Pervasive Context-aware Event Advisor). We hope that you found it to be an interesting and enjoyable experience!*

System Usability Scale

		Strongly Disagree				Strongly Agree
1.	I think that I would like to use this system frequently	1	2	3	4	5
2.	I found the system unnecessarily Complex	1	2	3	4	5
3.	I thought the system was easy to use	1	2	3	4	5
4.	I think that I would need the support of a technical person to be able to use this system	1	2	3	4	5
5.	I found the various functions in this system were well integrated	1	2	3	4	5
6.	I thought there was too much inconsistency in this system	1	2	3	4	5
7.	I would imagine that most people would learn to use this system	1	2	3	4	5
8.	I found the system very cumbersome to use	1	2	3	4	5

9. I felt very confident using the System

1	2	3	4	5

10. I needed to learn a lot of things before I could get going with this system

1	2	3	4	5

Name:

E-mail:

Date:

Comment

Annex D: Sample Code

A. Sample code for getting location of a user using CellID.

```
try {
        //Specify the retrieval method to Online/Cell-ID
        int[] methods = {(Location.MTA_ASSISTED | Location.MTE_CELLID | Location.MTE_SHORTRANGE
| Location.MTY_NETWORKBASED)};
        // Retrieve the location provider
        provider = LocationUtil.getLocationProvider(methods, null);
        location=provider.getLocation(50000);
        coordinates=location.getQualifiedCoordinates();
        f.deleteAll();
        f.append("Latitude:"+coordinates.getLatitude()+"\n");
        f.append("Longitude:"+coordinates.getLongitude());
    } catch (NoClassDefFoundError ex)
    {
        f.deleteAll();
        f.append("Cell-ID retrieval not supported on this device");

    } catch(InterruptedException e)
    {
        f.deleteAll();
        f.append("Location retrieval was interrupted");
    } catch(LocationException x)
    {
        f.deleteAll();
        f.append("Location could not be retreived");
    }
```

B. Sample HTTP request and response code for j2me/PHP server communication.

```
httpConnMy = (HttpConnection) Connector.open(url2);

        // Set the request method and headers

        httpConnMy.setRequestProperty("Content-Type", "application/x-www-form-urlencoded");
        httpConnMy.setRequestProperty("User-Agent", "Profile/MIDP-2.0 Configuration/CLDC-1.1");
        httpConnMy.setRequestProperty("Content-Language", "en-US");
        httpConnMy.setRequestProperty("Accept", "application/octet-stream");
        httpConnMy.setRequestProperty("Connection", "close");
        httpConnMy.setRequestMethod(HttpConnection.POST);
        httpConnMy.setRequestProperty("If-Modified-Since", "2 oct 2014 7:14:31 GMT");
        // httpConnMy.setRequestProperty("Cookies","PHPSESSID=bgdol19r8m8ijgvkev3sodu3v0;username="+m.tfusername+";password="+m.tfpassword+";");
        String formData = st;
        byte[] data = formData.getBytes();
        httpConnMy.setRequestProperty("Content-Length", Integer.toString(data.length));
        os = httpConnMy.openOutputStream();
        os.write(data);
        os.close();
        // Getting the output stream may flush the headers
        if (httpConnMy.getResponseCode() == HttpConnection.HTTP_OK) {
            String str;
            is = httpConnMy.openInputStream();
            int length = (int) httpConnMy.getLength();
            if (length != -1) {
                byte incomingData[] = new byte[length];
                is.read(incomingData);
                str = new String(incomingData);
            } else {
                ByteArrayOutputStream bytestream = new ByteArrayOutputStream();
                int ch;
                while ((ch = is.read()) != -1) {
                    bytestream.write(ch);
                }
                str = new String(bytestream.toByteArray());
                bytestream.close();
            }
            serverResponse = str;
            System.out.println("response = " + serverResponse);
            //fmpr.append(serverResponse);

            if (serverResponse.startsWith("YOUR REGISTRATION IS COMPLETED...")) {
                //m.tfusername.setString("welcome" + tfusername.getString());
                fmR.append("welcome" + username1.getString());

            } else {
                fmR.append("SORRY...YOU ARE ALREADY REGISTERED USER...");
                m.fmR.removeCommand(nextup);
            }
        }
    } catch (IOException ioe) {
        System.out.println("Caught IOException: " + ioe.toString());
    } finally {
        if (is != null) {
            try {
                is.close();
            } catch (Exception error) {
                /*log error*/
            }
        }
        if (httpConnMy != null) {
            try {
                httpConnMy.close();
            } catch (Exception error) {
                /*log error*/
```

C. Sending user movie preference sample PHP code.

```
        $deleteUserPreference = "delete from user_movie_preference where user_id = " . $_SESSION['user_id'] . ";
         // execute delte query
        $dresult = mysql_query($deleteUserPreference);
    }
    $saved = false;
       // for each movie genre save user preference
       while ( $row = mysql_fetch_array($result) ) {
         // get values from the user response
          if(isset($_POST[$row[1]])){
         echo '<input type="checkbox" value="'."$row[1]".'" name="'.$row[1].'" checked="checked" >'.$row[1].'</input>';
        $saved = true;
        $movie_preference_insert = "insert into user_movie_preference (user_id, genre_id)values('".$_SESSION['user_id']."','".$row[0]."')";
        $insert_result = mysql_query($movie_preference_insert);
        }
        else
        {
            $old_preference = "";
            if(!isset($_POST['hdnPreference'])){
            $movie_preference_select = "select * from user_movie_preference  where user_id ='".$_SESSION['user_id']."' and genre_id='".$row
[0]."'";
            $selec_result = mysql_query($movie_preference_select);
            if(mysql_num_rows($selec_result) == 1)
                $old_preference = 'checked="checked"';
            }
                echo '<input type="checkbox" value="'.$row[1].'" name="'.$row[1].'" '. $old_preference.'>'.$row[1].'</input>';

        }


//                  (($preference == "true")?'checked':'').'">'.$row[1].'</input>';
         /*
         if($preference == "true"){
            //save user preference
            $insert = "insert into user_movie_preference(user_id, genre_id) values('".
```

D. Sample PHP code for matching movie event.

```
//"    -- ms.location,".
"       msi.date,".
"       msi.time".
" from ".
"(SELECT  imdb_movie_info.movietitle, ".
"       (select GROUP_CONCAT(ref_movie_gener.genre_name)".
"        from  tbl_movie_genre, ref_movie_gener".
"        where  imdb_movie_info.movieid = tbl_movie_genre.movieid".
"               and ref_movie_gener.genre_id = tbl_movie_genre.genre_id".
"       ) genre,".
"       (select GROUP_CONCAT(ref_movie_gener.genre_name)".
"        from  tbl_movie_genre, ref_movie_gener, user_movie_preference".
"        where  imdb_movie_info.movieid = tbl_movie_genre.movieid".
"               And".
"               ref_movie_gener.genre_id = tbl_movie_genre.genre_id".
"               And".
"               user_movie_preference.genre_id = tbl_movie_genre.genre_id".
"               And".
"               user_movie_preference.user_id = '$user_id'".
"       ) upm,".
"       Duration,".
"       Description,".
"       movieid".
" FROM   imdb_movie_info) t, movie_schedule_info msi".
" where t.upm is not null".
"    and".
"    t.movietitle = msi.movietitle".
"    AND".
"    cast('20140930' as date) = ".
"    cast(".
"          concat(".
"          substring(msi.date,-4) ,".
"              (CASE WHEN upper(substring(trim(substring(msi.date,instr(msi.date, ' ')+1)),1,3)) = 'JAN' THEN '01'".
"              WHEN upper(substring(trim(substring(msi.date,instr(msi.date, ' ')+1)),1,3)) = 'FEB' THEN '02'".
"              WHEN upper(substring(trim(substring(msi.date,instr(msi.date, ' ')+1)),1,3)) = 'MAR' THEN '03'".
"              WHEN upper(substring(trim(substring(msi.date,instr(msi.date, ' ')+1)),1,3)) = 'APR' THEN '04'".
"              WHEN upper(substring(trim(substring(msi.date,instr(msi.date, ' ')+1)),1,3)) = 'MAY' THEN '05'".
"              WHEN upper(substring(trim(substring(msi.date,instr(msi.date, ' ')+1)),1,3))= 'JUN' THEN '06'".
"              WHEN upper(substring(trim(substring(msi.date,instr(msi.date, ' ')+1)),1,3)) = 'JUL' THEN '07'".
```

Annex E: User Manual

With the PCEA event advisor mobile app you can get services of suggestion about entertainment areas you prefer according to your contexts. In the application you can get information about movies, music‟s, art, sport, festivals and also you can get restaurant, coffee, juice and bar houses that are located near you and serve your food and drink choices.

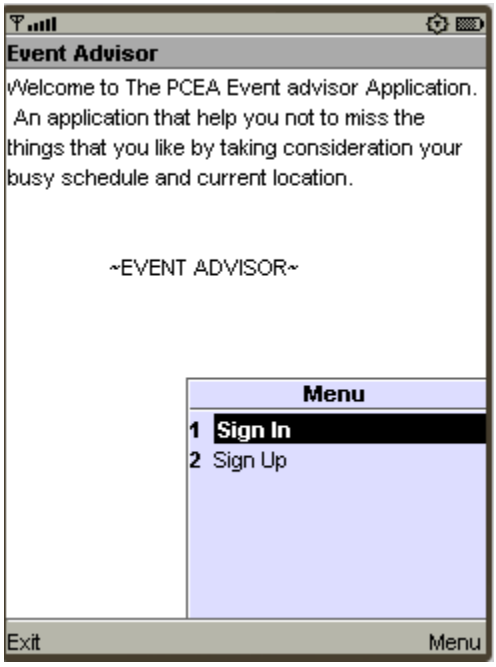
The app is supported by mobile phone that support j2me platform. It is compatible for Nokia S40 phones specifically Nokia Asha 201.

- ***How to get the PCEA event advisor mobile application on your mobile phone***

  To use the PCEA mobile app first you need to download the application to your mobile phone from www.eventadvisor.hints.me. To download the application on your mobile device appropriately, follow the instruction on the webpage.

- ***How to use the PCEA event advisor mobile application on your mobile phone after installing it***

  When you run the app you will see the application welcome page that have a general description about the application and a menu.



From the menu select „**sign up'** if you don‟t have a user account for the application or select „**sign in'** if you already have an account. A user account for the mobile app is required in order

for the system recognize you and provide you the service and for you to provide your preference selection.

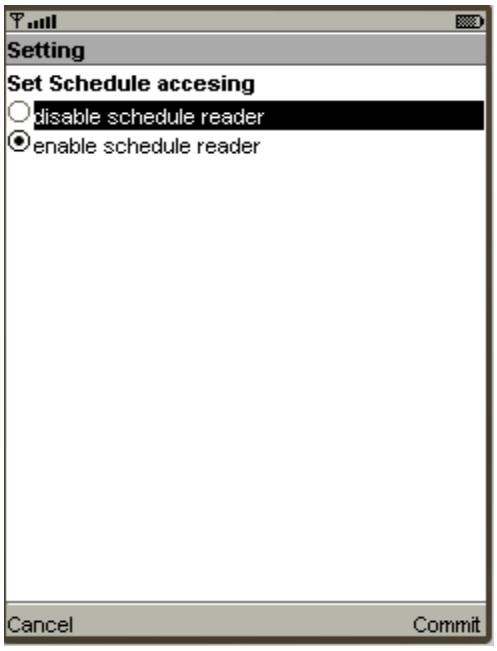
- To sign up to the PCEA event advisor mobile app all you have to do is fill the required information.
- To sign in to the PCEA event advisor mobile app you need to provide user name and password.

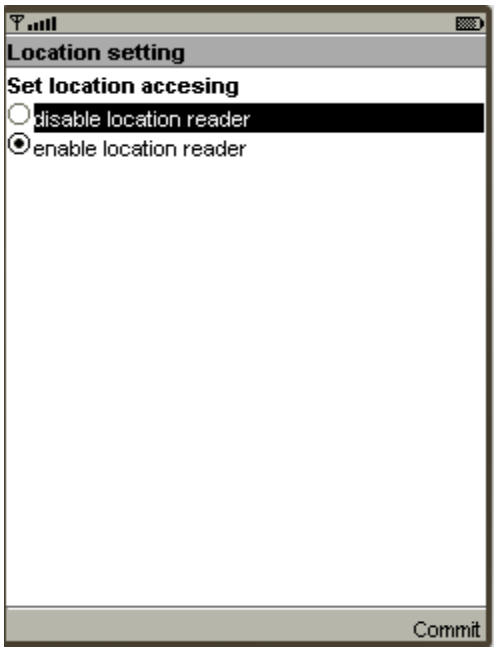
➢ ***How to select your setting preference***

The application let you to specify your preference for allowing the application to access your location and schedule information. After signing up there is a response page that let you know if there is an error in signing up or your attempt is successful.

- If it is not successful there will be back button on the left corner that let you return back and fill the required information and submit again.

- If it is successful there will be a next button in the right corner. Clicking the button will lead you to the setting preference selecting page. Here you can to select whether to allow or disallow dynamic data reading by selecting the radio buttons.





Note: It is preferable if you allow location and schedule accessing for a better suggestion and a better experience with the application.

- ***How to select your event preference***

  After registered to the application you need to provide your preference selection around the event categories the application provides. Here you follow simple step since the application make you choose by clicking check boxes.

  - The Commit button after the location setting page will led you to the first preference selecting page or event categories.

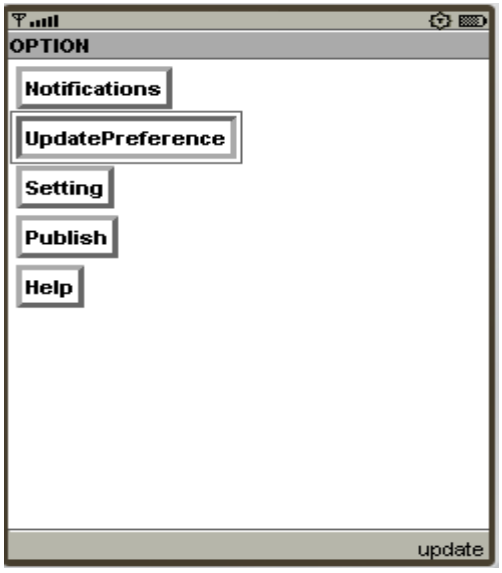
- Check the checkboxes and click the next button until you finish the event categories and then click submit that‟s found in the „**shop**‟ page to finish inserting your event preference and submit the information.

The „**Back**‟ button in each event categories page will let you go back to the previous page.

- ***How to update your event preference***

  If you want to edit your event preference after submitting, you can sign in to the system and perform the action.

  - After signing in you can see the option page
  - From the option page select update preference by clicking update button that‟s found on the page right corner



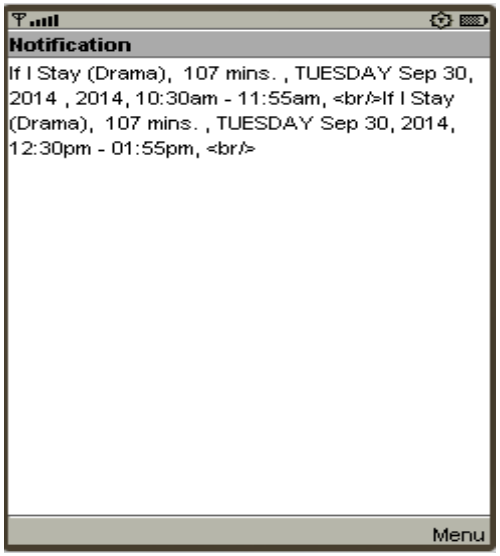
- The application page will ask you to check whether you want to update your preference or not if you select yes it will lead you to the first event category preference page and if no will return you back to the option page.

- ***How to update your setting preference***
  - Signing in to the application
  - On the „**Option**" page select „**setting**" button
  - Update your preference
- ***How to publish event information to the system***

  You can use this button if you know any event information that you want to share with other users.

- Signing in to the application
- On the „**Option**" page select „**publish**" button
- Fill the required information
- Click „**publish**" button
- Click „**cancel**" button if you want to cancel publishing the event information

- ***How to get help about the application***
  - Select „**Help**" button on the option page
- ***How to read an event notification that recently pushed***
  - Select „**Notification**" button on the option page



Declaration

I, the undersigned, declare that this research project is my original work and has not been presented for a degree in any other university, and that all source of materials used for the research project have been duly acknowledged.

Declared by:

Name: _____

Signature: _____

Date: _____

Confirmed by advisor:

Name: _____

Signature: _____

Date: _____

Place and date of submission: Addis Ababa, February 2015