Human Centric Situational Awareness

A Thesis Submitted to
Department of Computer Science and Engineering
In partial fulfillment of the requirements for the degree of Master of Science

Submitted by Mohamed Abo El Fotouh Abdel Samie

Under the supervision of Dr. Sherif G. Aly

December 2014
Acknowledgements

I would like express my great gratitude and deep appreciation to my thesis adviser Dr. Sherif G. Aly for his continuous support, guidance and for always pushing me to go the extra mile to complete this thesis.

I am greatly thankful to my family who always did their best to show me support and tolerance to be able to complete my thesis. I am truly grateful for my wife and sister for their help, encouragement and understanding.

Finally, I would like to show my appreciation to my father for his constant patience and understanding. He was always the one I looked to for encouragement meant to give the mental support I need.
Table of Contents

Abstract ........................................................................................................................................... 8

1. Introduction .................................................................................................................................. 9
   1.1 Context ...................................................................................................................................... 9
   1.2 Context Awareness .................................................................................................................... 9
   1.3 The Need for Context Awareness .............................................................................................. 10
   1.4 The Need for High level context ............................................................................................... 11
   1.5 How to Reach High Level Data ................................................................................................. 12
   1.6 Challenges with Context Aware Applications ............................................................................ 12

2. Problem Definition ......................................................................................................................... 14

3. Thesis contribution ........................................................................................................................ 15

4. Literature Review ............................................................................................................................ 16
   4.1 Defining Context Aware Systems .............................................................................................. 16
      4.1.1 Introducing Context awareness ......................................................................................... 16
         4.1.1.1 Sensing Context ............................................................................................................ 17
         4.1.1.2 Modeling Context ......................................................................................................... 19
         4.1.1.3 Acting ............................................................................................................................ 20
      4.1.2 Categorization of Context aware systems ............................................................................ 21
      4.1.3 Context Aware Architecture ............................................................................................... 22
   4.2 Classifications, Middleware and Models .................................................................................... 25
      4.2.1 Human Perspective based context acquisition .................................................................... 27
      4.2.2 Situation Templates ............................................................................................................. 31
      4.2.3 A Multilayered Uncertainty Model for Context Aware Systems ........................................ 33
   4.3 High Level Context Aware Applications .................................................................................. 35
      4.3.1 Cenceme ............................................................................................................................ 35
      4.3.2 Darwin Phones .................................................................................................................... 38
      4.3.3 ContextPhone ..................................................................................................................... 42
      4.3.4 SmartProfile ........................................................................................................................ 46

5. Research Methodology .................................................................................................................. 49
   5.1 Overview .................................................................................................................................... 49
   5.2 Defining Situations and Sensors ............................................................................................... 50
   5.3 Survey Implementation and Results .......................................................................................... 51
      5.3.1 Survey Research Objective ................................................................................................. 51
      5.3.2 Survey Major Research Questions ..................................................................................... 51
Table of Figures

Figure 1: Abstract layered architecture for context-aware systems [10] ......................................... 16
Figure 2: List of possible sensors [23] ............................................................................................. 18
Figure 3: Real-world situations related to sensor data [8] ................................................................. 18
Figure 4: Context Aware System Architecture [6] ........................................................................... 22
Figure 5: Grouping Similar Context .................................................................................................. 23
Figure 6: Simple form of the context life cycle [45] ........................................................................ 24
Figure 7: model types used in 109 applications [45] ...................................................................... 25
Figure 8: Using Human Perspective based context Acquisition [23] ............................................... 27
Figure 9: Sensor Based Context Acquisition [23] ......................................................................... 28
Figure 10: Human Based Context Acquisition [23] ...................................................................... 28
Figure 11: Notification Modes on Users' cell phones [23] ............................................................... 29
Figure 12: Notification Modes users' frequently employ [23] .......................................................... 29
Figure 13: Situation Templates [26] ............................................................................................... 31
Figure 14: Situation Recognition Component [26] ...................................................................... 32
Figure 15: Context Ladder [39] ...................................................................................................... 34
Figure 16: Context Facts and source data for meeting situation [39] ............................................. 34
Figure 17: Flow of Information Ceneme System [19] .................................................................... 36
Figure 18: The Layers of Ceneme System [19] .............................................................................. 37
Figure 19: Darwin Model [17] ........................................................................................................ 38
Figure 20: Voice Recognition without Darwin Components [17] ................................................... 40
Figure 21: Voice Recognition with Darwin Components Enabled [17] ........................................... 40
Figure 22: ContextPhone Architecture [57] .................................................................................. 42
Figure 23: Context-Contacts Application Interface [57] ................................................................ 44
Figure 24: Context-Media Application Interface [57] .................................................................... 44
Figure 25: Information flux between sensors [11] ......................................................................... 46
Figure 26: Learning Processes performed by ICSM & ILSM [11] ..................................................... 47
Figure 27: Experiment results hits & Misses [11] .......................................................................... 47
Figure 28: Research Methodology Flowchart .................................................................................. 49
Figure 29: Party Scenario.................................................................................................................. 54
Figure 30: Meeting Scenario............................................................................................................ 55
Figure 31: In Car Scenario............................................................................................................... 55
Figure 32: Sleeping Scenario .......................................................................................................... 56
Figure 33: Theatre Scenario ............................................................................................................ 56
Figure 34: Situational Awareness System Prototype ...................................................................... 58
Figure 35: Accelerometer Coordinate System .................................................................................. 61
Figure 36: Inference Engine............................................................................................................ 67
Figure 37: System Dynamic Interface ............................................................................................. 68
Figure 38: Devices used in Experiment (Samsung Galaxy Tab 2 & Motorola Moto G) .................. 71
Figure 39: Confidence level with and without context weights ....................................................... 72
Figure 40: Experimental results ..................................................................................................... 75
Figure 41: Experimental Trial Results for SLEEPING and CAR Scenarios ........................................ 76
Figure 42: Experimental Trial Results for MEETING and THEATRE Scenarios ................................. 77
Figure 43: Experimental trial results for PARTY scenario .............................................................. 78
Figure 44: R.A.P Context Model [46] .............................................................................................. 90
Figure 45: State Space [42] ............................................................................................................ 93
Figure 46: Context Edges [42] ........................................................................................................ 94
Figure 47: Learning States [42] ...................................................................................................... 94
Figure 48: The Sensay Architecture [52] ......................................................................................... 96
Figure 49: Context-aware prompting system architecture [16] ...................................................... 98
Figure 50: Experiment Tasks [16] .................................................................................................. 99
Figure 51: Sensor Deployment [16] ................................................................................................ 99
Figure 52: Information flow in EmotionSense [31] ................................................................. 100
Figure 53: Emotion Clustering [31] ............................................................................................... 101
List of Tables

Table 1: Sensor attributes and values................................................................. 50
Table 2: Activity Hypothesis.............................................................................. 53
Table 3: Sensor formats and limitations.......................................................... 60
Table 4: System Validation versus Darwin Phones & Smart Profile.................. 79
Abstract

Context awareness is an approach that has been receiving increasing focus in the past years. A context aware device can understand surrounding conditions and adapt its behavior accordingly to meet user demands. Mobile handheld devices offer a motivating platform for context aware applications as a result of their rapidly growing set of features and sensing abilities. This research aims at building a situational awareness model that utilizes multimodal sensor data provided through the various sensing capabilities available on a wide range of current handheld smart phones. The model will make use of seven different virtual and physical sensors commonly available on mobile devices, to gather a large set of parameters that identify the occurrence of a situation for one of five predefined context scenarios: In meeting, Driving, in party, In Theatre and Sleeping. As means of gathering the wisdom of the crowd and in an effort to reach a habitat sensitive awareness model, a survey was conducted to understand the user perception of each context situation. The data collected was used to build the inference engine of a prototype context awareness system utilizing context weights introduced in [39] and the confidence metric in [26] with some variation as a means for reasoning. The developed prototype’s results were benchmarked against two existing context awareness platforms Darwin Phones [17] and Smart Profile [11], where the prototype was able to acquire 5% and 7.6% higher accuracy levels than the two systems respectively while performing tasks of higher complexity. The detailed results and evaluation are highlighted further in section 6.4.
1. Introduction

1.1 Context

Researchers and scholars from different fields may define context in ways that best expresses their understanding of the concept of context. But a definition for context we find most suitable to this research would be information that describes the state or surrounding environment of a mobile device user. According to [59], the three main characteristics of context as previously defined are: where you are, whom you are with, and what resources are around you. As such, context is categorized into three fields [3][21][33]:

- **User context**: Defines the situation from the user’s standpoint covering factors such as user’s location, movement, user profiles and social situation.

- **Physical context**: Contains information regarding the surrounding physical environment, which is gathered through sensor-based measurements such as: lighting, time, temperature, noise and humidity level.

- **Computing context**: Includes all available computing and application resources such as processors, devices reachable for user input and output, other nearby resources such as printers or workstations.

1.2 Context Awareness

For decades, computation has been centered on the needs of the machines rather than the needs of the humans using these computational devices. We have been required to learn their languages and interact only on their rules. But more than twenty years ago Mark Weiser envisioned a world where computers blend into our daily environment, in which devices have perceptive and adaptive components. Meaning an environment with built in computing devices with the ability to characterize and understand the surrounding context and then react accordingly by adapting to meet the needs of its users [4][40].
Context can be defined as: “any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves”. [2][43]

A context aware application can be regarded as a system that makes use of context information to adapt by providing relevant information or services to the users of this system in a constantly changing execution environment. This can only be done by sustaining system awareness towards the surrounding environment’s conditions and events [2].

The research conducted in [15] segmented context aware services into 3 main components; the Scene, context aware service and context aware system. The scene was described as the abstract form of relevant context entity within the user’s environment. A context aware service was mentioned as a type of service that utilizes context information to adapt its current state accordingly. Finally, the context aware system was described as a system that acquires information from diverse sources, conducts normalization and modeling to the collected information and applies the required services.

1.3 The Need for Context Awareness

The digitalization of people’s lives should not require more attention from the user’s side, but on the contrary as computing has evolved from stationary and dedicated devices into mobile, multimodal and multiuser devices; it should also evolve into pervasive context aware devices and applications. “The most profound technologies are those that disappear.”[38] Meaning that easy integration of computational devices into people’s everyday life is needed to realize the pervasive computing vision. To reach this vision computing must shift to context aware applications which can perceive the
surrounding environment and adapt accordingly serving the best interest of the actors of this environment [21][53].

The need for context awareness is vital for constructing a pervasive environment. A system or application is categorized as context aware if it is able to extract, infer and utilize context information to adapt its functionality to the accurate context of use [14].

1.4 The Need for High level context

Context can be classified into two categories: low-level context and high-level context. Lower level context information can be obtained directly from sensors and is considered as a form of raw data. High level context information on the other hand can be obtained through the combination of lower level context data to reach a more abstract form. Furthermore, lower level context aware systems are only concerned with acquisition of sensor data and then abstraction of this data by matching the perceived information to a suitable context [53]. This level of context awareness may be suitable for tasks such as location awareness. But, as mobile devices’ features have evolved in terms of capabilities, storage, applications and sensors, it is fair to assume that the level of sophistication of mobile applications is also in a dire need to expand. High level context awareness integrates and combines lower level data provided by sensors to increase perception and understanding of context information.

Although recognizing the state and features of different attributes of a specific environment is the first stage of situational awareness [41]. The need for high level context awareness is increasing exponentially as a result of the rapid expansion of mobile applications’ capabilities. Pervasive and ubiquitous computing research is now focused on higher level applications that require; context reasoning, situation detection and activity recognition which in turn require more abstract high level information.
1.5 How to Reach High Level Data

The process of acquiring high level context information goes as follows [60]:

1. **Acquisition of raw data:**
   The process of receiving raw data from sensors, which is considered as the attributes of a context situation such as location, temperature, sound level, bandwidth and acceleration.

2. **Aggregation and feature extraction:**
   The process of extracting metadata such as variance or mean values from a set of raw data.

3. **Interpretation**
   Mapping raw data to meaningful and relevant context information sets. This step realizes the move from low-level data to useful context information.

4. **Utilizing the generated high level context information in an application**

1.6 Challenges with Context Aware Applications

Creating a reliable context aware system is not an easy task a number of challenges face the implementation of this type of application. The main obstacle is defining and inferring context information correctly, as context is a collection of parameters that define a certain situation. Any misperception of parameters will in turn lead to incorrect adapting behavior on the system’s part. Also, the complexity of networked systems may cause a burden on developers of pervasive applications. Other setbacks may occur due to incomplete sensor data which would lead to prediction of some information which may also lead to misinterpretation of a specific situation [29][36].
The context aware system architecture introduced in [54] illustrates means to dealing with uncertainty of context aware system inference due to collection of fuzzy or inaccurate context information, through displaying system confidence of inference data.
2. Problem Definition

Having reviewed and analyzed a large sum of high level context awareness models and applications we discovered that most of these applications and models widely depend on a very limited set of sensors as a source of data. The limited set of sensor data in situational awareness negatively affects the accuracy of system inferred context. The lack of sufficient sensor data produces incomplete and incorrect context information which in turn leads to misperception and incorrect adapting behavior.

Another common drawback in a majority of current context aware applications and models that we feel attributes to lower accuracy rates is the negligence of localized preferences of users in different geographical regions. Meaning no localized information sets are utilized in the development of context aware applications reviewed in this research, the general notion is to build context aware applications independent of local environments and user behavior.


3. Thesis contribution

The main contribution of this thesis is:

1. Attempting to increase the current accuracy levels of existing situational awareness models through the utilization of more relevant sensation available on smart phones. The combination of several diverse sensors that individually capture a small aspect of an environment should produce a clearer representation that better characterizes a situation. As means of evaluation, the proposed model’s resulting outcome will be benchmarked against the experimental results of the Darwin Phone model illustrated in detail in [17] and the Smart Profile application described in [11]. A more detailed account of both implementations will be covered in section 4.3 of this document.

2. Enhancing Situational awareness models with habitat sensitive information. In the sense that context perception will consider variations in context meaning throughout different geographical regions and context reasoning will be influenced based on the preferences of users and their behavior in a specific geographical area.
4. Literature Review

4.1 Defining Context Aware Systems

4.1.1 Introducing Context awareness

A computer system can be defined as context aware if it has the ability to extract, interpret and use context data to adapt its functionality to the current context of use [32].

In addition any context aware application may support three general kinds of features: 1) presentation of information and services to a user, 2) automatic execution of a service and 3) tagging context information for later referencing [2].

Context aware applications normally use various sensors to infer the user’s activity. One of the main obstacles facing such applications is the methodology of detecting the appropriate context form noisy and ambiguous sensor data.

Any form of adaptation as a result of context awareness must be relevant to the user’s expectation, or otherwise will be considered as intrusive behavior. As a result sensor data which acts as the seed for any context aware application must be accurate to avoid any misinterpretation. In addition, any context aware application needs a strong modeling mechanism to match the provided sensor data to an appropriate context.

A context aware pervasive system consists of three basic functionalities: Sensing, Reasoning and Acting [34].

![Figure 1: Abstract layered architecture for context-aware systems [10]](image)
4.1.1.1 Sensing Context

Sensors provide means to collect data or information about the physical world. This knowledge provides computer systems with a mechanism to infer actions most suitable for the physical situation at hand. A combination of multiple sensors may reveal more information for the computer system to reason with, constructing a more comprehensive and accurate image of the physical world [24][34].

Gathering sensor data can be done through a number of ways illustrated in [24][48] as:

- Device-databases (e.g. calendars, to-do-lists, address books and profile information).
- Direct input to the application running (notepad and taking notes).
- Active environments (active badges, IR-networks and cameras).
- Sensing context using sensors (Sensor Badges, GPS, cameras, microphones, etc.)

Another means of identifying sensors as highlighted in [45] and [30] is through categorizing sensor types into three groups; Physical, virtual and logical sensors.

- **Physical Sensors**: This type of sensor outputs the data itself and is usually the most common sensor type. (ex. Accelerometer, Microphone and Light sensors)
- **Virtual Sensors**: This type of sensor collects data from other sources and outputs it as sensor data. (ex. Calendar, twitter status, email)
- **Logical Sensors**: Created through a combination of physical and virtual sensors to provide a high level service. (ex. web service for weather information)
The perceptual capabilities can be located on the device itself, embedded into the environment or on other devices that share the same context.

<table>
<thead>
<tr>
<th>Sensor Type</th>
<th>Possible Attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Light Sensors</td>
<td>Ambient light, indoors brightness, outdoors, ...</td>
</tr>
<tr>
<td>Accelerometers</td>
<td>Motion, vibration, physical state, ...</td>
</tr>
<tr>
<td>Video Cameras</td>
<td>Facial expressions, gaze, behavior, presence, ...</td>
</tr>
<tr>
<td>Microphones</td>
<td>Speech, noise, music, decibel levels, ...</td>
</tr>
<tr>
<td>Motion Detectors</td>
<td>Presence, single or multiple users, ...</td>
</tr>
<tr>
<td>Pressure Sensors</td>
<td>Pressed, occupied, hand gestures, ...</td>
</tr>
<tr>
<td>RFID</td>
<td>Location, activity, situation, ...</td>
</tr>
<tr>
<td>GPS</td>
<td>Location, ...</td>
</tr>
<tr>
<td>Environmental Sensors (temp, humidity, etc.)</td>
<td>Weather and other environmental conditions</td>
</tr>
<tr>
<td>Event Monitors</td>
<td></td>
</tr>
<tr>
<td>Action Listeners</td>
<td>GUI events, schedules, notifications, errors, updates, ...</td>
</tr>
<tr>
<td>Data Loggers</td>
<td></td>
</tr>
<tr>
<td>Agents, Processes</td>
<td></td>
</tr>
</tbody>
</table>

**Figure 2: List of possible sensors [23]**

Sensor data can be used to map out real life situations; the research carried out in [8] gives a detailed account of different sensors and an illustration of how sensor data can be utilized to identify different attributes of a situation. Figure 3 illustrates examples of sensor values in real world situations.

<table>
<thead>
<tr>
<th>Situation</th>
<th>Sensor Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>User sleeps</td>
<td>It is dark, room temperature, silent, type of location is indoors, time is “nighttime”, user is horizontal, specific motion pattern, absolute position is stable</td>
</tr>
<tr>
<td>User is watching TV</td>
<td>Light level/color is changing, certain audio level (not silent), room temperature, type of location is indoors, user is mainly stationary</td>
</tr>
<tr>
<td>User is cycling</td>
<td>Location type is outdoors, user is sitting, specific motion pattern of legs, absolute position is changing.</td>
</tr>
</tbody>
</table>

**Figure 3: Real-world situations related to sensor data [8]**
4.1.1.2 Modeling Context

This refers to the process where raw sensor data is modeled to reflect physical entities in an environment which could be manipulated. This modeling intends to build knowledge concepts from the information provided by the sensors in an environment. Significant information should be interpreted from raw data attained by the sensors.

According to [53] low level context data provided by sensors may be modeled to high level context information by a number of methods:

- **One-to-One**: when one low level context value matches one high level context dimension. The sensed aspects of a context are combined and compared with a model to provide a value.
- **Context Fusion**: When several low level context values match one high level context dimension.
- **Context Fission**: When one low level context value matches several high level dimensions.

A context state is defined as a group of attributes (i.e. dimensions) values at time t. Each dimension has a value space that represents its range of values for a specific context situation for example the age range for a specific user might range within 20-50 years old. These values could either contain a discrete number of elements or a continuous range of elements. In addition some context dimensions have a higher relevancy to a certain context situation; as a result a weight is assigned to each dimension that reflects its importance for each context situation [53].

Another approach is to segment the context model into elements of user, device, activity, time and location to create a context awareness model that tailors the services of an application to adapt to user needs or preferences. The user is identified as a collection of user information both permanent such as date of birth, preferences or gender and information that describes his identity such as job role [12]. The device element represents the general specifications of a given mobile device including
available sensors, storage capacity and operating system. As for the Activity it is defined as the set of tasks and roles conducted by the user at a given point of time. The time and location elements enable the system to react in a timely manner while matching user’s location to nearby services taking vicinity of available services into account [12].

4.1.1.3 Acting
Once context information has been collected and situations have been recognized, appropriate actions are taken as an adaptation to the users’ or the environment’s needs.
4.1.2 Categorization of Context aware systems

A system is context aware if it uses context to provide relevant information and services to the user where relevancy depends on the users’ task.

In [53], the authors proposed an extended definition to context aware computing stated as follows:

“Context aware computing aims to enable better service delivery through adapting use, access, structure and behavior of information, services, applications and physical resources with respect to available context information.”

In an aim to further describe the domains of usage for context aware computing, the following categorization of context aware applications is introduced [53]:

1. **Context based filtering and recommendation of information and services:**
   Sorting through available context information and recommending an appropriate service that meets the given context. E.g.: finding the nearest physical resource in an office.

2. **Context based presentation and access of information and services:**
   Presenting available relevant information and directly accessing the appropriate service that meets the given context. E.g.: choosing voice when screen display is not accessible.

3. **Context based information and service searching:**
   Utilizing context information in searching services. E.g.: location aware search engine (which utilizes context information “Location” to filter retrieved search results).

4. **Context based service and application modification/configuration:**
   Changing the status of an existing feature according to acquired context information.
5. **Context based actions:**
   The actions taken to effect the surrounding environment as means of adapting to the available context information.

6. **Context based resource allocation:**
   The allocation of various resources such as “physical memory” for the use of variable entities in an environment.

### 4.1.3 Context Aware Architecture

Context aware systems are normally composed of a number of elements handling the following issues:

- Context Discovery
- Context Management
- Context Representation
- Adaptation

Cádiz et al. [6] proposed a context aware architecture which addresses the elements above, creating a foundation for the design of context aware systems.

![Figure 4: Context Aware System Architecture [6]](image-url)
**Context Discovery:** This component is responsible for gathering raw context data from different sensors and translating this data into information of useful nature.

**Context Management:** As the amount of gathered information may be huge due to the variety of sensors and high availability of information sources, the data acquired could be ambiguous and at some point conflicting. This component is responsible for conflict resolution and processing raw data into consistent meaningful information.

**Context Representation:** This component focuses on grouping context information of similar nature together in families of contexts.

![Diagram](image)

**Figure 5: Grouping Similar Context**

The previous figure illustrates an example on grouping similar contexts together forming a family of related context information.

**Behavior Adaptation:**

The Final component of a context aware system is in charge of adapting the application’s behavior to meet the users’ needs in a constantly changing environment.
Another alternative to looking at context aware systems is the concept of context life cycle mentioned in [45] where in essence most of the existing context aware models follow four phases as part of the context life cycle.

![Context Life Cycle Diagram]

**Figure 6: Simple form of the context life cycle [45]**

Figure 6 illustrates the main components of the context life cycle [45]:

- **Context Acquisition**: Context is gathered from a collection of source being either physical or virtual sensors.
- **Context Modelling**: The collected data needs to be modeled and represented in specific formats to give it meaning.
- **Context Reasoning**: Modelled data requires processing to extract high level context information from the initial low level sensor data.
- **Context Dissemination**: Distribution of context information to the user.
4.2 Classifications, Middleware and Models

Detailed surveys on various context aware applications and models published in recent years conducted in [9] [45] [13] identify the most commonly used decision models and reasoning techniques. The survey’s results in [9] and [45] account for 109 different context aware applications published between 2003-2009 within three major conferences: Computer-Human Interaction (CHI), Ubiquitous Computing (Ubicomp), and Pervasive. Figure 7 details the results of this survey.

![Figure 7: model types used in 109 applications [45]](image_url)

The most widely implemented reasoning model in context aware applications is developer specified rules where majority of user preferences is encoded within these rules. In general rule based reasoning models apply a straightforward an IF-THEN-ELSE approach. Rules are a simple method of mapping human thinking [45][9]. The research to create the PersonisAD framework described in [37] is an account of rule based systems implementation. The results published in [7] also give a detailed overview of different available context awareness middleware.
The second reasoning model is Decision trees, mostly popular for its simplicity in implementation. “A decision tree infers an output by deciding on a specific input feature at each node as it traverses down and returns a decision once it reaches a leaf.” [9]

Naïve Bayes is a classifier that implements Bayes Theorem to define the probability of system outputs taking into consideration the collected inputs. While Hidden Markov Models “are Bayesian probabilistic classifiers that model the probability of a sequence of hidden states given a sequence of observations”. These two models are not widely utilized by context aware systems due to their higher complexity. [9]

For the remainder of this section a number of context awareness models will be showcased and discussed in details. The first model introduced is a human perspective based data acquisition technique will be illustrated to highlight the benefits and down points of employing a human based approach in comparison with a sensor based approach for context acquisition. In the following section the use of situation templates in the inference of context will be discussed. In addition to a detailed description on the use of a multilayered uncertainty model that introduces the concept of context weights to calculate system confidence. Finally, the research described in [27] introduces an interesting approach to mining context aware information regarding user preferences on smart phones through collecting historical data from a context log on mobile devices.
4.2.1 Human Perspective based context acquisition

In context aware systems the accuracy of sensor data generated from various sensors is a vital aspect of the effectiveness of such a system. In some cases accurate data that reflects real life situations is not always available, as a result of this the need for some mechanism to increase the accuracy of the input sensor data. This method proposes using a human perspective based approach to complement incomplete sensor data.

In this study a user centered approach is proposed for data acquisition to create rules and learning models for context awareness. Information received from the user directly can be used to create accurate context awareness models while keeping the user’s preference in mind [23].

Figure 8: Using Human Perspective based context Acquisition [23]

The previous figure shows how human perspective is used to generate ontology and awareness models. Using this approach has a number of advantages including:

- User preferences, environmental characteristics, and other relevant criteria can be directly taken into consideration during the design of awareness models.
- Allows data collection to truly represent the correct scenarios and use cases.
- Takes considerably less time to collect data compared to sensor based data acquisition.
- Data can be gathered for large scenarios without disrupting users.

Figure 9: Sensor Based Context Acquisition [23]

Figure 10: Human Based Context Acquisition [23]
The figures illustrate the major modifications that were carried out on the context aware system design to employ the user based acquisition technique as opposed to sensor based acquisition.

The research at hand implemented this approach on an application for interruption-aware cell phones. A comprehensive research was conducted to collect user experience about use of cell phones under different scenarios. The research conducted illustrated the various modes available on cell phones and the most frequently used ringer profiles used [23].

![Figure 11: Notification Modes on Users' cell phones [23]](image1)

![Figure 12: Notification Modes users' frequently employ [23]](image2)
The implemented system integrated sensor data with human based context data resulting in an application that follows a number of awareness models. The system can change the ringer usage on a given cell phone according to the surrounding context gathered through sensors integrated on the cell phone and it’s matching with the previously set human experience [23].

This method utilizes human perspective context acquisition which greatly depends on user experience and preferences in the context reasoning process. The method also lacks any means of updating context definitions which will negatively affect its performance in any constantly changing environment. Furthermore, Situations on which user perspective was based are not easily reproduced which affects later context reasoning. Finally, the research was created under the assumption that the raw sensor data would be completely accurate.
4.2.2 Situation Templates

This research introduces the idea of providing the spatial model infrastructure with a situation recognition component based on generic situation templates. “A situation template is an abstract, machine readable description of a certain situation type, which could be used by different applications to evaluate their situation” [26].

A situation template is composed of:

- An accurate description of context information considered relevant to a given situation type.
- A description of how to infer the existence of a solid situation from given data or values.

Each situation template is part of a larger template library which could be referenced by various applications to evaluate their current situations.

Figure 13: Situation Templates [26]
A situation type is described in [26] as the representation of an explicit, recurring condition or event in the real world and can act as means of assessment for the adaption process of context-aware applications. The ideal description of those specific conditions including various parameters such as, their thresholds, and guidelines of how to infer the occurrence of a situation from these parameters is referred to as Situation Template.

“Providing a situation template of a certain situation type with concrete data and processing it allows for the diagnosis of the existence of a particular situation, a situation token [26].”

The research separated between two uncertainty metrics probability metric and confidence metric to address the issue of uncertainty of context ranging from sensor values, inferred context and situation recognition process. [26]

- **Probability Metric**: is the probability value that represents the occurrence of a situation from the recognition-process view. A Higher value indicates a higher assumption of the occurrence of a certain situation.
- **Confidence Metric**: is normalized value between 0-1, which represents the quality of the utilized situation-template. The higher value indicates that the template will detect the situation with high levels of correctness.

![Figure 14: Situation Recognition Component [26]](image)
The previous figure illustrates how the situation recognition component is integrated into the existing spatial model infrastructure. An example of an established situation template is illustrated to infer a situation “whether a meeting takes place in the room X” or not. The templates initially identified are stored in a template repository. While each situation maps to a specific object that is attached to a situation template [26].

The user is using a simple meeting planner application that initiates a situation aware workflow. After the workflow is initiated, it looks up the situation of all available meeting rooms and reserves the first free room. The user is able to input feedback regarding the correctness and accuracy of the situation recognition component.

### 4.2.3 A Multilayered Uncertainty Model for Context Aware Systems

This research presents a model that collects context uncertainty and delivers methods to capture the uncertainty level of a given situation. The research introduced the notion of context ladder which starts by grouping source data of a context aware system into 2 groups; Physical sensor readings as temperature which may be uncertain depending on precision and accuracy and profiled data such as calendar inputs which may be inaccurate due to human error. The second layer of the context ladder is context facts which are generated from source data; each fact has a quantifiable confidence level that is defined by the source data uncertainties. Finally, each situation is assigned a confidence level through combining the confidence of the underlying context facts with appropriate context weights that represent the importance of a specific context fact to the occurrence of a situation. [39]
The study carried out a demonstration utilizing the meeting as a sample situation with 3 data sources listed in the figure below.

Each context fact will have an accompanying confidence level affected by the uncertainties and the context weights. To determine the situation confidence level the research combined probabilistic logic and context weights reflecting the importance of each context to the occurrence of the situation. The following equation was utilized to generate the confidence level $\sigma$ where $\beta$ represents the different context weights and $\gamma$ is the confidence for each respective context fact; [39]

$$\sigma = \frac{\sum_{1 \leq i \leq 3} \gamma_i \times \beta_i}{\sum_{1 \leq i \leq 3} \beta_i}$$

As a conclusion the research utilized the notion of context weights to showcase the contribution of each context fact to the confidence level of a given situation.
4.3 High Level Context Aware Applications

4.3.1 Cenceme

This application is a personal sensing system that enables users to share their current status and context on social networks. The application captures the users’ context through utilizing the sensors found on commercial smart phones. The system defined the sensed status in terms of the user’s physical activities (walking, sitting, running, etc), feelings (Sad, Happy, etc), habits (at the gym, coffee shop, etc) and surroundings such as (bright, dark, cold, hot). The application then updates the user’s social network accounts (facebook, twitter, IM) with his current status [18][19][20].

The application is built upon the concept of “sensing presence” which captures the user’s status in terms of activities, feelings, habits and surroundings. The system utilizes the mixture of hardware sensors built in any commercial smart phone with virtual software sensors that infers context information from the user’s online status. Other building blocks of the cenceme application include [18][19]:

- **An analysis engine**: This module identifies the type of sensing presence from the acquired data.

- **Data Storage repository**: This module is responsible for storing various sensor data for easy access.

- **Services Layer**: Facilitates the sharing of context information between different users, while being subject to adjustable user privacy settings.

The Cenceme application provides the following services:

- **Life Patterns**: contains all the information that reflects the user’s interests based on his current and past activities.

- **My Presence**: This reflects the current sensing presence of the user.

- **Friend Feeds**: Provides a constant status feed for different friends using the same application.
- **Social Interaction:** Utilizes data inferred from sensing presence to build valid assumptions regarding the interactions of various users.

- **Significant Places:** Users can identify certain locations as places of interest associated with their normal activities.

- **Other Social Services:** These services include searching for users, comparisons between different users and their profiles [19].

The Cenceme architecture is divided upon three main layers: receiving raw sensor data from physical and virtual sensors, inferring low level context information from the raw data, and finally sharing the learned context information on various social networks with the user’s friends.

![Flow of Information Cenceme System](image)

**Figure 17: Flow of Information Cenceme System [19]**

Once more the main aim of the Cenceme application is to make use of the large number of sensors available on mobile phones nowadays to collect information about the users of such devices. The application employs this acquired data to construct information patterns about users indicating their recurring interests, activities, habits or feelings. Finally the application uses the huge reach of social networks to broadcast the gathered context information about a specific user to all his friends on a certain online social medium [19].
The Cenceme application is divided into different classes of devices mainly a back end server and a mobile device. The consumer devices are considered as the source of the raw data through collecting information through the attached sensors. The back end server is perceived as the core of the Cenceme architecture by conducting all the computation required to transform raw sensor data into low level context information. Finally the information is shared through social networks and various multimedia tools [19].

This application reaches high level context awareness, through utilizing a number of sensors to collect raw data and then generates an inference about some defined user activities. As a setback for the implementation of this application the reasoning process accuracy rates are relatively low as the application’s classifiers produced a large number of false positive inferences. Another point is that reasoning accuracy varies in different context environments; accuracy is fairly high when located in an indoor environment and decreases outdoors. Finally the application is able to infer the occurrence of certain user activities such as; walking, sitting, running and in conversation, but does not include capabilities for situational awareness with a larger number of parameters.
### 4.3.2 Darwin Phones

Darwin presents a collaborative reasoning system that utilizes sensors available on a mobile device to automatically infer various aspects of a person's life while achieving better accuracy and scalability, at lower cost to the user.

Darwin combines three different computational steps to achieve its goal: 1) Evolution, 2) Pooling and 3) Collaborative Inference.

**Figure 19: Darwin Model [17]**

- **Classifier Evolution**: Is an automated approach using self-evolving classification models over time such that the classifiers are robust to the variability in sensing conditions common to mobile phones.

- **Model Pooling**: An approach that allows mobile devices to seamlessly exchange different models, if the model is obtainable from a different phone, thus, allowing mobile devices to promptly increase their classification abilities; that is, if a certain device doesn’t have an existing model for a certain user or occasion, there is no need recreate the classifier as it should be already available on a different device.
• **Collaborative Inference**: Collects and combines different classification outcomes from various devices to reach a more accurate reasoning with higher confidence towards the inferred result. After pooling all the devices in the same location will share a common set of classifiers. At which all devices are able to execute the same inference algorithm in parallel and the final result is the combination of all phones’ output [17].

**Darwin Operations [17]:**

- **Step 1**: Every Mobile device creates a number of models for the events to be sensed in the seeding phase. As time goes by the seed models are used to gather new data and evolution takes place. The reason for this step is that, by increasingly gathering new samples, the models will recruit data in different environments and hence, become more robust to environmental variations.

- **Step 2**: When a number of mobile devices are co-located they exchange their models so that each device will contain its own original model in addition to the co-located device’s model. Different devices may share their knowledge base enabling them to undergo larger classification tasks. For example: in the case of speaker recognition, going from recognizing the owner of the phone to recognizing all the people around in conversation.

- **Step 3**: “Collaborative inference exploits this diversity of different phone sensing context viewpoints to increase the overall fidelity of classification accuracy [17]”. 
Experimental Results:
A voice recognition experiment was conducted to illustrate the need for Darwin’s evolve-pool-collaborate model. Three people are walking on a busy street and are engaged in a discussion. The application for speaker recognition runs on each of their mobile devices without the Darwin components; meaning no available classifier evolution, no pooling, and no collaborative inference algorithms are available for the trial. In addition the training of voice classifiers for the three people was conducted in a quiet indoor setting [17].

![Figure 20: Voice Recognition without Darwin Components [17]](image1)

The experiment resulted in a 63%, 61% and 52% accuracy rates for phones 1, 2 and 3 respectively. The poor performance was a result of the classification model being trained indoor. This was reflected in a poor performance outdoors due to operation in a noisy outdoor setting. This highlights the challenge presented when a mobile device is sensing in a different environment [17].

![Figure 21: Voice Recognition with Darwin Components Enabled [17]](image2)
The experiment was conducted a second time with the classifier evolution component intact. This second scenario had more noise than the first trial, which resulted in lower initial accuracy before running the evolution when compared to the first experiment. The experiment shows the accuracy improvement as the amount of data sent from the phone to the backend for re-training grows. The accuracy of voice recognition increases over time as the classification evolves to reach a more precise outcome.

On implementation this model was applied on an application for voice recognition which confined the capabilities of the reasoning process through depending only on sensors related to sound context. This limited sensation caused drop in accuracy levels during outdoor trails of the application.
4.3.3 ContextPhone

ContextPhone, a software platform built for smart phones which helps bridge the gap between the functionalities offered by mobile operating systems and the actual needs of context aware application development. The platform consists of four interconnected modules [57]:

![ContextPhone Architecture](image)

**Figure 22: ContextPhone Architecture [57]**

*Sensors*: Means for acquisition of context data from various sources in the surrounding environment. In ContextPhone a number of physical and virtual sensors which can be found on an off the shelf smart phone are utilized for data gathering [57]:

- **Location**: GSM cell identifier and GPS.
- **User Interaction**: Application use when active, Idle and active phone state, Phone profile for the alarm and battery charge state.
- **Communication Behavior**: Calls, attempted calls, SMS and Content of messages sent or received.
- **Physical Environment**: Availability of Bluetooth devices and networks
Communications: “ContextPhone supports both local (Infrared, Bluetooth) and wide area (GSM, GPRS) communications. The Communications package has both protocol implementations and service abstractions on top of these.” [57]

Customizable versions of built-in applications: ContextPhone provides customizable versions of already built-in smart phone applications. ContextPhone provides a number of context-aware applications that support its platform such as Context Contacts application.

System Services: This module of ContextPhone is responsible for complimenting the smart phone system with any lacking features such as; Automatic startup, crash and recovery mechanism and an error log. [57]

Three applications were developed to utilize and illustrate the usage domain of ContextPhone on off the shelf smart phones. The applications make use of the context data provided through the ContextPhone platform to reach context awareness.

1. **ContextLogger: Studying patterns of mobility**
   The application records mobility of sensor and user data. “ContextLogger receives notifications of context changes from the sensors and customizable applications, writes this data in a local file, and periodically uploads the files to the researchers’ server via the background file upload” [57].

2. **ContextContacts: Automatic context-sharing for human-to-human communication**
   This application lets users automatically represent and exchange context information. As opposed to building agents that decide whether the user being called is in an interruptible state. The application facilitates the exchange of context information regarding the user’s state between two mobile devices carrying the same application. ContextContacts consists of three main elements [57].
• **Presence publisher**: gathers relevant sensor data from and broadcasts it to other users via the Jabber channel.

• **Presence Listener**: receives data broadcasted by other devices via the Jabber channel.

• **Customizations of applications**: Integrates the information into the interface of the applications.

![Image](image_url)

**Figure 23: Context-Contacts Application Interface [57]**

3. **ContextMedia: Sharing mobile media**

This application exploits the idea of situated media, which is media that includes a description of the situation it was taken in. ContextMedia utilizes different sensors to interpret media. Users can utilize the Jabber channel to inform other users of shared media in real time [57].

![Image](image_url)

**Figure 24: Context-Media Application Interface [57]**
This model is more focused on inference and sharing of context data through various applications, the main goal is creating means to facilitate context reasoning. The application uses multiple sensors for capturing context data. The application does not utilize the huge sensation capabilities available on most current smart phones; instead the application is confined to the use of a very limited set of sensors to acquire context data.
4.3.4 SmartProfile

This application personalizes a user’s phone sound mode based on data retrieved from the user’s calendar and location. The system includes two main modules; Information Inferring Module (IIM) and Personalization Module (PM). The IIM utilizes the data collected from smartphone sensors and generates a user profile, while the PM uses the gathered information to personalize the sound state of the phone accordingly.

![Figure 25: Information flux between sensors][11]

Moving forward with system development the system was further segmented into more specific sub-modules; Inferring Calendar Sub-Module (ICSM), Inferring Location Sub-Module (ILSM) and a Vibrate Personalization Sub-Module (PVSM). On change of device state the inferring Sub-Modules are triggered and updated information is sent to the personalization Module. The ICSM retrieves data from the user’s calendar to understand his behavior pattern, while the ILSM Module analyzes the relation between a user’s location and his routine. The below diagrams illustrate the flow of process. [11]
A prototype for the application was developed using Java on Android technology and was initially tested by collecting data from 16 subjects who had the application running continuously on their smart phones. The system had the objective of correctly personalize the sound profile of the users based on the systems current context. The system resulted in a 77.4% accuracy level based on the data collected from the user devices. [11] The number of hits and misses for the system are represented in the following figure.
The system utilized a very limited set of sensors to identify user context. The trial experiments were conducted only among 16 users which generally is a very small sample size to properly assess the accuracy of the application.

Further research highlighted in [1] also focuses on employing sensors on a given smartphone for activity recognition proposes. The research introduces a different model where no human interaction is required, no additional hardware needed only a normal smartphone and can infer up to 7 different activity types.
5. Research Methodology

5.1 Overview

This research aims at constructing a situational awareness model that identifies the occurrences of predefined context scenarios using an off the shelf smart phone. In addition, the research utilizes localized information to complement the inference capabilities of the developed model through understanding user preferences and behaviors in a specific geographical region. The following figure represents the steps undertaken to implement this research.

![Research Methodology Flowchart](image)

Figure 28: Research Methodology Flowchart
5.2 Defining Situations and Sensors

As an initial step, 5 different situational scenarios were selected for the implementation of the proposed model, the selection of the scenarios focused on real life events with regular occurrences which would be easy to replicate for experimental purposes.

The selected scenarios are:

- Driving
- Meeting
- Party
- Sleeping
- Theatre

Each context scenario will be inferred by gathering various parameters that identify a specific situation. The parameters of each situation are viewed as the set of multimodal sensor data that is obtained from various sensors found on an off the shelf smart phone. A list of sensor types and their attributes were identified and initial value ranges were assigned to each sensor type as a hypothesis to validate the users’ preferences. The list of sensors selected and their attributes are displayed in table 1.

<table>
<thead>
<tr>
<th>Sensor Attribute</th>
<th>Sensor</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Surrounding Sound Level</td>
<td>Microphone</td>
<td>High / Medium / Low / None</td>
</tr>
<tr>
<td>Amount of Motion</td>
<td>Accelerometer</td>
<td>High / Medium / Low</td>
</tr>
<tr>
<td>Mobile ringer usage</td>
<td>System Profile</td>
<td>Vibrate / Ringer / Silent / Other</td>
</tr>
<tr>
<td>Lighting</td>
<td>Light Sensor</td>
<td>High / Medium / Dark / All options valid</td>
</tr>
<tr>
<td>Changes in Absolute position</td>
<td>GPS</td>
<td>Rapidly changing / Regularly changing /</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Stable / Stationary</td>
</tr>
<tr>
<td>Time of Day</td>
<td>System Clock</td>
<td>Morning / Afternoon / Night / All options</td>
</tr>
<tr>
<td></td>
<td></td>
<td>valid</td>
</tr>
<tr>
<td>Day of the Week</td>
<td>System Calendar</td>
<td>Weekday / Weekend</td>
</tr>
</tbody>
</table>

Table 1: Sensor attributes and values
5.3 Survey Implementation and Results

As the next step of this research and in order to understand the wisdom of the crowd a localized survey was conducted with 155 subjects as part of the human perspective context acquisition to determine the users’ preferences and experience in the case of each previously defined context situation. Each participant identified the sensor attributes with the highest priority in relevance to each one of the predefined context scenarios. The context model was complimented with this localized survey to reach a habitat sensitive approach, in the sense that the context acquisition will be based on the preferences of users located in Egypt.

5.3.1 Survey Research Objective

Primary research objective was to answer the major research questions and validate or negate the alleged hypothesis. The research was conducted in the form of a survey/Questionnaire as a research instrument. Primary targets for data collection were fellow researchers, Random mobile users, experts in the field of mobile computing, telecom equipment suppliers and university students.

5.3.2 Survey Major Research Questions

1. What are the factors that indicate the occurrence of each one of the following context situations:
   a. Meeting
   b. Party
   c. In Car
   d. Movie Theatre
   e. Sleeping
2. What is the level of importance of each factor?
3. What are the parameters of each factor that would specify the previously mentioned activities?

5.3.3 Survey Research Method

The survey was implemented using an analytical predictive research method, where the research involves collecting and analyzing data, applying statistical tests, using surveys/questionnaires to gather numerical data. To finalize the survey hypothesis testing was required and therefore this research is considered as quantitative.

5.3.4 Survey Sampling Method

The survey conducted two different methods of sampling to collect data from the proposed target segment:

- **Snowball Sampling**: Uses recommendations to select individuals with a certain range of skills or experiences that have been defined as useful. In this method study subjects recruit future subjects from within their own contacts.

- **Stratified Sampling**: The population is divided into homogeneous subgroups before sampling. Each subgroup from the population will produce representatives.
5.3.5 Initial Activity Hypothesis

<table>
<thead>
<tr>
<th>Activity</th>
<th>Context Parameters</th>
</tr>
</thead>
</table>
| Party    | • Sound Level: High  
|          | • Motion: High  
|          | • Light Intensity: Low  
|          | • Day: Weekend  
|          | • Time: Night  
|          | • Absolute Position: Regularly Changing  
|          | • Phone Ringer: High |
| Meeting  | • Sound Level: Medium  
|          | • Motion: Low  
|          | • Light Intensity: Low  
|          | • Day: Weekday  
|          | • Time: Morning/Afternoon  
|          | • Absolute Position: Stable  
|          | • Phone Ringer: Vibrate/Silent |
| In Car   | • Sound Level: Medium  
|          | • Motion: Medium  
|          | • Light Intensity: Irrelevant  
|          | • Day: Irrelevant  
|          | • Time: Irrelevant  
|          | • Absolute Position: Rapidly Changing  
|          | • Phone Ringer: Irrelevant |
| Theatre  | • Sound Level: High  
|          | • Motion: Low  
|          | • Light Intensity: Low  
|          | • Day: Irrelevant  
|          | • Time: Theatre display schedule  
|          | • Absolute Position: Stationary  
|          | • Phone Ringer: Irrelevant |
| Sleeping | • Sound Level: Low  
|          | • Motion: Low  
|          | • Light Intensity: Low  
|          | • Day: Irrelevant  
|          | • Time: Night  
|          | • Absolute Position: Stationary  
|          | • Phone Ringer: Silent/Vibrate |

Table 2: Activity Hypothesis
5.3.6 Survey Results

This section illustrates the knowledge of the crowd obtained through the comprehensive survey conducted among 155 subjects from different backgrounds and age groups. The aim of the survey was to gather the users’ preferences and experience in dealing with each of the following context scenarios:

- Party
- Meeting
- Sleeping
- In Car
- Theatre

Where each subject answered the major research questions by indicating which of the identified sensor attributes has the highest priority in relevance to each one of the predefined context scenarios.

Each of the following figures shows a graphical representation of the users’ preferences to each context scenario and its defined sensor attributes.

![Figure 29: Party Scenario](image-url)
Figure 30: Meeting Scenario

Figure 31: In Car Scenario
Figure 32: Sleeping Scenario

Figure 33: Theatre Scenario
5.4 Defining the Confidence Metric

A confidence metric was constructed as a normalized value between zero and one, which reflects the quality or the correctness of the system inference. The confidence level of the system towards a given situation is generated from the average probability attributed to each sensor type from the localized survey. The confidence level of a specific situation is influenced by the concept of context weights described in [39] where each sensor type in a given situation scenario is given a weight value that quantifies its importance in interpreting the occurrence of that situation. The confidence equation and context weights assignment as part of the system inference engine will be further illustrated in section 5.5.2.

5.5 System Prototype

Throughout the development of the application multiple trials were conducted to ensure the application utilized sensors that were common across the vast majority of smart phones currently available in the market as to avoid any compatibility limitations in the use of the application on specific devices. In addition, the researcher aimed to employ sensation that can easily be computed and inferred on the client side without the need to run any complex compilation on an application server. This was to further ensure no limitations existed in the use of the application.

The system prototype was established with 2 main pillars:

- **Sensors sub-module**: Responsible for collection and normalization of raw sensor data.
- **The inference engine**: Tasked with applying an inference algorithm to generate the system confidence level towards the occurrence of each context situation.
Each of the system components will be illustrated in details in the following sections. Figure 34 demonstrates the system flow starting from the sensor data input into the system and ending with the final output to the application dynamic interface.

The system start is initiated by the smart phone user upon the occurrence a specific context environment, this is done through clicking on the “START” button available in the application interface. The following details the steps of the process carried out by the system upon launch:

- Upon start of the application the smart phone sensors start collection of sensor data from the surrounding environment for a pre-determined time frame which for the purpose of this research was set to a 2 minute time intervals.

- Each sensor sends the raw data collected to its designated sensors sub-module which will be discussed further in the next section.
- Each sensor sub-module normalizes the gathered data, updates the dynamic interface with every new reading and sends the final value at the end of the time interval to the inference engine.

- The inference engine uses the survey results i.e. user preference defined at the beginning of the research as input to the inference algorithm that identifies the confidence level and the occurrence of a specific context scenario. The inference engine applies a rule based context awareness model utilizing the IF-THEN-ELSE approach as means of reasoning.

- Finally the outcome is sent to the application interface which is in turn visible to the smart phone user.
5.5.1 Sensor Modules

Most Android powered devices have embedded hardware and software sensors that measure multiple surrounding conditions including motion, temperature, sound and location. Our system used 7 different physical and virtual sensors currently available on a wide range of recent versions of smart phones utilizing the Android mobile operating system. For the application to collect raw data from all sensors, a sub-module was created for each sensor type to gather and normalize the sensor readings over a pre-defined time frame into an effective usable format that can be used by the system’s inference engine.

The sensor sub-modules tapped into the Android sensor framework which provides different classes and Interfaces to help developers conduct sensor related commands. The below table describes the different sensors, system formats, normalized values and some of the sensor limitations we faced while gathering the sensor data.

<table>
<thead>
<tr>
<th>Sensor Type</th>
<th>Sensor data format</th>
<th>Normalized Value</th>
<th>Limitations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accelerometer</td>
<td>X,Y,Z acceleration force coordinates</td>
<td>Number of motions measured</td>
<td>Low sensitivity to slight motion</td>
</tr>
<tr>
<td>Microphone</td>
<td>Sound Amplitude</td>
<td>Average sound readings converted to Decibel equivalent</td>
<td>Maximum reading of 32,767 Amplitude value (equivalent to 100 Decibels)</td>
</tr>
<tr>
<td>GPS</td>
<td>Location Longitude and Latitude</td>
<td>Difference in Absolute location</td>
<td>Only works outdoors</td>
</tr>
<tr>
<td>Ambient Light</td>
<td>Light intensity in Lux</td>
<td>Average of light readings</td>
<td>NA</td>
</tr>
<tr>
<td>System Clock</td>
<td>Time</td>
<td>Day time or Night time</td>
<td>NA</td>
</tr>
<tr>
<td>System Date</td>
<td>Date</td>
<td>Weekend or weekday</td>
<td>NA</td>
</tr>
<tr>
<td>Ringer</td>
<td>Ringer setting</td>
<td>NA</td>
<td>NA</td>
</tr>
</tbody>
</table>

Table 3: Sensor formats and limitations
Each Sensor module converted the raw sensor readings into meaningful information to match the conducted localized survey’s data sets. Each sub-module used a different mechanism to gather, maintain and convert the raw data and the following sections detail the mechanics of each sub-module and the conversions used for each sensor.

5.5.1.1 Accelerometer Sensor Sub-Module

As detailed previously the accelerometer in the Android platform normally measures the forces applied on a mobile device in all 3 axis X, Y and Z in m/s$^2$. In the survey results motion was mapped as fuzzy intervals of “None”, “Low”, “Medium” and “High” motion. Further experimentation was conducted to assign these intervals to meaningful numeric values listed below:

- 0 is equivalent to “No Motion”
- 1-10 is equivalent to “Low Motion”
- 11-20 is equivalent to “Medium Motion”
- >20 is equivalent to “High Motion”

The Accelerometer Sensor Sub-Module converted the initial readings of the accelerometer to total number of motions applied on a smart phone in a given time frame through creating a counter that was applied upon any change in device orientation.

Figure 35: Accelerometer Coordinate System
**5.5.1.2 Microphone Sensor Sub-Module**

The built-in Microphone sensor on Android gathers data in Amplitude format, while the survey results once again referred to sound level as “None”, “Low”, “Medium” and “High”. The research utilized the findings detailed in [50] which was a study conducted on different sounds and their effect on the human ear, while listing the decibel equivalent readings. The Microphone Sensor sub-module converted amplitude values gathered by the sensor to their decibel equivalent derived from [50].

- 0-40 Decibel is equivalent to “No Sound”
- 41-79 Decibel is equivalent to “Low Sound”
- 80-85 Decibel is equivalent to “Medium Sound”
- 85-90 Decibel is equivalent to “High Sound”

This sensor specifically faced some limitations due to the restricted capabilities of the typical built in microphone on off the shelf smartphones. The microphone on multiple devices used in the testing of the application could only distinguish sound levels of up to 90-95 decibels and any volume above that range was not defined. The quality of sensation of the microphone also varied in terms of sensitivity to sound level from one device to the other, but was within a conservative margin that allowed the research to carry out the experiments detailed in section 6.

**5.5.1.3 GPS Sensor Sub-Module:**

The GPS sensor gathers the longitude and latitude values of the current smartphone location, while the survey results mapped out the absolute change in position. In this case the GPS sensor Module calculates the approximate change in absolute position between the starting point of the experiment and the ending location in a given time frame which reflects the speed of movement and the change in the mobile device’s position. The `distanceBetween()` built-in function was utilized to deliver this result. The
function takes the longitude and latitude of the start and end locations as parameters for its calculations.

- 0 movement or GPS not working are equivalent to “Stationary/Stable”
- Between 1 - 400 meters is equivalent to “Regularly Moving”
- > 400 meters is equivalent to “Rapidly Moving”

5.5.1.4 Ambient Light Sensor Sub-Module:

The ambient light sensor collects data in lux, while the results of the survey conducted identified light values as “Dark”, “Medium” and “High” Light intensity. The researcher defined the real life equivalent values of light intensity through the typical illuminance values study conducted in [56], then the Light sub-module was created to map the sensor readings in lux to their equivalent real life situations and the below segmentation was created.

- <50 lux is equivalent to “Dark”
- 51-2000 lux is equivalent to “Medium Light Intensity”
- >2000 lux is equivalent to “High Light Intensity”

5.5.1.5 System Clock Sensor Sub-Module

The System clock is used to retrieve current time in hours, minutes and seconds format and the Clock Sensor Sub-Module converts the reading into one of two defined values either “Day time” or “Night time” based on the retrieved timing and the actual Sunset and Sunrise real life timings.

- Between 5 AM – 5 PM is equivalent to “Day time”
- Between 5:01 PM – 4:59 AM is equivalent to “Night time”
5.5.1.6 System Calendar Sensor Sub-Module

The System calendar is used to retrieve current day of the week and that is converted to one of two defined values either “Weekend” or “Weekday” by the sensor sub-module.

- Friday and Saturday are defined as “Weekend”
- Sunday, Monday, Tuesday, Wednesday and Thursday are defined as “Weekday”

5.5.1.7 Phone Ringer

The Phone ringer sensor is used to collect the current sound profile of the used smartphone and is classified into one of three defined profiles “Normal”, “Silent” and “Vibrate”.

Each sub-module handles the collection and normalization tasks of a specific sensor type through managing sensor availability, checking for sensor capabilities, handling event listeners, acquiring raw sensor data and running normalization functions. [51] Sensor sub-Modules finally trigger the inference engine after both the normalization of data and the defined time frame is complete.

The sensor Sub-Modules are also linked to the application’s dynamic interface and with every new change such as new GPS coordinates, increase in motion or fluctuation in light intensity. The sensors sub-modules update the interface in real time for the smart phone user to be able to track changes in the surrounding environment in real time.
5.5.2 Inference Engine

The system’s main component is the inference Engine which is responsible for collecting normalized sensor data values from the different sensor modules and inferring the current situation based on the outcome and probabilities generated by the survey conducted at the beginning of the research among 155 subjects. The engine applies a rule based context awareness model with an IF-THEN-ELSE approach for reasoning. The inference engine then generates a numerical value that identifies the confidence level of the system towards the occurrence of a given situation in the current context.

To define the confidence level of the system towards the occurrence of each of the 5 pre-defined situations, the inference engine attributes 5 probability values mapped from the localized survey to every sensor reading one for the occurrence of each situation respectively. For each situation the inference engine combines the different probabilities for all 7 sensor readings to generate a confidence level for that situation. The situation with the highest confidence level above a 50% threshold is deemed to be the correct situation by the system.

As previously clarified the notion of context weights described in [39] was applied in our inference engine with some variation to capture each survey subject’s belief towards the importance of each sensor type in the identification of different context scenarios. In essence context weights further quantify each context factor’s (Sound level, light, motion, etc.) contribution to the occurrence of the situation relative to the other sensor data. In efforts to segment the various sensor readings and their importance in the occurrence of each situation into groups of varying context weights we referred to our initial localized survey which assigned all the sensor types per individual situation into 3 groups of importance “high”, “medium” and “low” with the survey subjects being aware that each group would be multiplied by a factor to quantify its contribution to the overall confidence of each situation.
In order to define appropriate context weights for each situation and sensor reading, the researcher resorted to further experimental research conducted using different context weight thresholds that will be detailed further in the experimental methodology section. The conclusion was reached to set the factors to a ratio of 1 : 1.05 : 1.1 for sensors with low importance, sensors with medium importance and for sensors with high importance in the occurrence of each situation respectively.

An equation was developed to take into account the relative importance of each sensor data set to the overall occurrence of each situation. In every trial the inference engine would calculate the confidence level for all 5 situations to generate an accurate reasoning on the user’s current situation based on the situation with highest confidence level. The equation calculates the weighted average of normalized sensor readings between 0-1 for each context scenario to determine the confidence level for each of the predefined scenarios.

\[
CL_c = \frac{\sum W \times S_c}{n}
\]

In certain cases the survey results deemed some of the sensors Irrelevant to the occurrence of a given situation an example being the time of day for the “IN CAR” scenario, in such cases the inference algorithm disregarded these sensor readings and included only the sensor types that were listed as relevant to each situation. This mechanism allowed the system to generate a confidence metric based only on relevant
data sources highlighted by the users which contributed to the overall accuracy of the system results.

The inference engine uses the previously mentioned equation to calculate the confidence level of each situation by applying reasoning influenced by the data extracted from the localized survey which represents the user’s confidence towards each sensor parameter. The system would then generate the results of the context reasoning with the relevant confidence level.
5.6 User Interface

The system was built with a simple user interface that captures the readings dynamically on the screen as each trial run is in progress and at the end of the defined time frame the results would be displayed. Each of the sensor types’ readings would be displayed in real time as they change according to the environment variations, which gave the researcher the ability to trace and resolve any anomalies in the sensor readings. The interface displayed all the different sensors vertically and at the bottom of the screens all the normalized final sensor data would be displayed as means of validation to the inferred context situation.

![Figure 37: System Dynamic Interface](image-url)
5.7 Utilized Technology

In order to experimentally validate the research hypothesis, a context awareness application was built to utilize a large range of virtual and physical sensors available on an off the shelf smart phone to increase the accuracy levels of current available situational awareness models. The system was built using the Java programming language on the Android SDK to support the Android operating systems of Android 4.0 and above.
6. Validation

6.1 Experimental Setup

This section illustrates the details of the experiment setup to validate the developed system prototype described in the previous section against the research hypothesis; utilizing an off the shelf smart phone with a large range of sensors and habitat sensitive context information to increase situational awareness applications’ accuracy levels. The experiment will use an In-Situ experimental technique to conduct the experiment, where a real application is tested in a real environment.

The experiment applied a 2 minute time frame as the duration permitted for continuous data collection by the sensors and normalization functions by the sensor sub-modules for each trial run. This period allowed enough time for the system to form an accurate overview of the surrounding environment attributes. Initial experiments were conducted to test other time frames of 30 seconds and 1 minute, but results indicated insufficient data sets that generated inaccurate results from the inference engine. Thus the decision to carry out the experiment with a 2 minute time frame as it proved to be a minimum adequate time interval for data collection to allow for reliable system reasoning.

Prior to running the actual experiment a number of pilot test runs were conducted to ensure the system is working correctly and this led to minor changes in the inference engine with regards to the original user preference for context situations. These minor contradictions in our opinion were the result of fuzzy definitions in the survey phase that might have caused slight confusion in interpretation, nonetheless the pilot testing conducted created a solid foundation for the decision to apply minor corrective changes to a limited number of user preferences.
The experimentation to validate the developed system prototype was conducted using a Samsung Galaxy tab2 mobile device running the Android 4.0 platform and a Motorola Moto G mobile device running the Android 4.0 platform while allocating the trial runs between both devices to 70% and 30% respectively. Both devices contain an Accelerometer, Ambient light sensor, Microphone and GPS which allowed for seamless collection of data across either device and although some inconsistencies in the sensor readings did exist they had a minor effect on the overall system inference results.

![Figure 38: Devices used in Experiment (Samsung Galaxy Tab 2 & Motorola Moto G)](image)

### 6.2 Defining the Context Weights

As illustrated in the previous section the system utilized the notion of context weights to quantify the importance of some sensors in identifying specific context situations based on the results of the localized survey. The research questions in the initial survey asked the subjects to assign every sensor type in each specific context situation to one of 3 defined groups low importance, medium importance and high importance with no defined numeric weight for each group.
The research conducted further experimentation on various context weight ranges to understand the implications of different weights on the inference capabilities of the developed system. Experiments were conducted to test 3 different sets of context weights to identify the optimum setting for the system:

- No context weights used
- Ratio of 1 : 1.05 : 1.1 used for Low, Medium and High importance respectively.
- Ratio of 1: 1.1 : 1.2 used for Low, Medium and High importance respectively.

For each set of context weights 3 trial experiments were carried out for each context scenario. The experiment results showed an obvious increase in confidence level upon integrating context weights within the inference algorithm which can be clearly shown in the previous graph Figure 39 where the Y-axis represents the confidence level % and the X-axis maps out the different context scenarios. The use of context weights at a ratio of 1: 1.05 : 1.1 for the low, medium and high importance levels showed an average of 4% increase in confidence level generated by the system as opposed to not using context weights.

Figure 39: Confidence level with and without context weights
Upon increasing the context weights to 1:1.1:1.2 the overall confidence level of the results did in fact increase further, but the results were also accompanied with a major increase in wrong reasoning where wrong context scenarios were identified as correct. This was caused by the inflation of some of the sensor attributes due to the high multiplication factor of up to 20% used in this case. The conclusion reached from conducting this experiment was that the context weights used in this research required a maximum threshold as not to affect the integrity and accuracy of the system results. As a result the context weights were set as 1:1.05:1.1 for the duration of the experimentation.

6.3 Experimental Testing Trials

After finalizing the experimental setup and defining the accurate context weights intervals to be used for testing, the researcher set out to validate the research hypothesis through conducting an accurate experiment of the developed system prototype and inference engine. The experiment was carried out through 20 trial runs conducted for each of the predefined context scenarios adding up to 100 experimental trials in total.

An Experiment results form shown in Appendix C was created to capture the readings, circumstances and system inference for each trial run. The form included normalized sensor readings, Context Scenario Identified by the system, Actual Context Scenario and the Confidence level. The researcher also looked at the confidence levels of the other context scenarios to better understand and enhance the inference engine based on resulting system reasoning. The form also helped the researcher identify some of the common factors that might affect the inference capabilities of the application, the following being the most common:

- Accuracy of GPS readings
- Accelerometer sensitivity specifically while driving a car
- Sensitivity of microphone
The experiment for testing the system prototype was designed to capture real life situations where a subject would carry the mobile device with the installed application to an actual party, theatre, meeting, car trip or while sleeping. In each situation the subject was asked to act normally within the context situation and to apply his own preferences when it comes to conducting these situations in his day to day activities. Each experiment result and readings would be captured and preserved by the dynamic interface of the application and in most cases when applicable the experiments would be conducted under the monitoring of the researcher.

The experimental trials resulted in an overall confidence level of 66.8% across all 5 predefined context scenarios. Although generating a higher confidence level might have been possible, that would have negatively affected the overall accuracy of the system. A detailed account and evaluation of the experimental results will be covered in the next section.
6.4 Evaluation and Results

This section of the research thesis aims at showcasing the results and evaluation of the experiment carried out to validate the research hypothesis. The section will also demonstrate a comparison between the context awareness model proposed in this thesis and other context models and applications described in section 4.3 of this document. After running 100 trial experiments using the developed mobile context awareness application the research gathered the resulting data to give an overview on the accuracy and confidence level of the proposed model and supporting prototype application.

The overall results proved highly successful with an 85% accuracy rate which represents the overall percentage of correct inferences generated by the system. This was accompanied by an average 66.8% confidence level towards all context situations inferred by the system, which represents the system’s certainty to the occurrence of a specific situation. The below graph in figure 40 illustrates the overall results segmented by situation type, given that some situations attributed more positively than others towards the overall accuracy and confidence of the system.

![Accuracy and Confidence Graph](image_url)

Figure 40: Experimental results
As highlighted in the previous graph the system had the highest accuracy in identifying both the “SLEEPING” and the “IN CAR” context scenarios accompanied by the highest confidence level generated by the system’s inference engine. The researcher’s evaluation for this outcome is the identifying characteristics of both context scenarios defined by the research survey where the highest attributing sensor value to the occurrence of an “IN CAR” situation was the rapid change in absolute position, which was mostly unique to this situation. The second context scenario “SLEEPING” was also uniquely identified by the very low sound levels, very low light sources and no motion.

![Confidence Level vs Trial Number](image)

**Figure 41: Experimental Trial Results for SLEEPING and CAR Scenarios**

On the other hand the system reacted slightly different to the “MEETING” and “THEATRE” context scenarios as a result of their complexity due to lack of unique identifying sensor values. The “MEETING” and “THEATRE” situations were often confused with other context scenarios sharing similar context profiles. Another aspect that contributed to the results of this experiment was the irregularities in sensor readings between the different devices used in conducting the experiment. Never the less the use of multiple sensor attributes enabled the system to maintain a satisfying accuracy level of 75% and 85% with a confidence level of 63% and 63.5% for the “MEETING” and “THEATRE” scenarios respectively.
Another context scenario that proved to be most challenging in reasoning with high levels of confidence was the “PARTY” context scenario. This is greatly attributed to the uncertain representation to some of the sensor attributes in the initial survey results. This Resulted in more complexity within the inference engine to define the “PARTY” scenario with a high probability of occurrence when compared to other scenarios. The overall accuracy level generated by the system for the situation was 85% which is significantly high even though the confidence level was at 57%.

Figure 42: Experimental Trial Results for MEETING and THEATRE Scenarios
This research model was also initially designed taking into account the user preferences and behaviors derived from a localized survey, which enabled the system to support local habits and largely attributed to the high accuracy levels presented by the system. It is worth noting at this point that applying the same system to a different culture without adapting the inference engine to the preferences of the new user groups might generate a different outcome.

As stated previously in this section the research aims to benchmark the generated results against previous related work namely Darwin Phones [17] and Smart Profiles [11] the following table captures the comparison between all 3 models and identifies how the model introduced in this research proves to be more superior.
### Human Centric Situational Awareness

- **System Functionality**: Developed to identify the occurrence of 5 predefined context scenarios.

- **Sensors Utilized**: Microphone, GPS, Ambient Light, Accelerometer, System Clock, Calendar and phone ringer

- **System Capabilities**: Able to identify the occurrence of each situation with no pre-setup required, the system acts as a plug and play application compatible with any Android device with the minimum sensor requirements.

- **Experiment Setup**: 100 trials were conducted for the 5 predefined scenarios utilizing various test subjects and 2 separate devices.

- **Performance**: The System generated an 85% overall average accuracy rate for 5 different context situations.

<table>
<thead>
<tr>
<th>Human Centric Situational Awareness</th>
<th>Darwin Phones</th>
<th>Smart Profiles</th>
</tr>
</thead>
<tbody>
<tr>
<td>Developed to identify the occurrence of 5 predefined context scenarios.</td>
<td>Developed as a voice recognition application in multiple context environments.</td>
<td>Developed to identify user behaviors and habits and adapt phone sound profile to match user preference.</td>
</tr>
<tr>
<td>Microphone, GPS, Ambient Light, Accelerometer, System Clock, Calendar and phone ringer</td>
<td>Microphone</td>
<td>System Clock, user calendar and GPS</td>
</tr>
<tr>
<td>Able to identify the occurrence of each situation with no pre-setup required, the system acts as a plug and play application compatible with any Android device with the minimum sensor requirements</td>
<td>The system require classifier training for devices before implementation and different training environments could results in wrong readings in real life.</td>
<td>The system is able to create a user profile based on use habits, but requires data collection from the devices before implementation for duration of 2-4 weeks.</td>
</tr>
<tr>
<td>100 trials were conducted for the 5 predefined scenarios utilizing various test subjects and 2 separate devices.</td>
<td>3 trial runs conducted on different scenarios where 2 types of mobile devices were used.</td>
<td>Experiment conducted with data gathered from 16 subjects.</td>
</tr>
<tr>
<td>The System generated an 85% overall average accuracy rate for 5 different context situations.</td>
<td>The system generated an overall accuracy rate between 80%-85% for correct voice recognition.</td>
<td>The system presented a 77.4% accuracy level</td>
</tr>
</tbody>
</table>

**Table 4: System Validation versus Darwin Phones & SmartProfile**

These results show that when benchmarked against other context awareness models/platforms our system through utilizing a higher number of sensors available on an off the shelf smartphone while enhancing its reasoning engine with habitat sensitive information was able to successfully deliver more complex tasks with higher levels of accuracy then previous models or applications that purely relied on a limited sensation ability to carry out a simple reasoning task. This research’s proposed model generated
accuracy levels of 85% as an average across 100 trial experiments for a task of identifying 5 different complex context situations, while the 2 other systems delivered accuracy levels of 80% and 77.4% to carry out simple tasks such as voice recognition or setting the phone sound profile to user preference.

It is also worth noting that the system proposed in this research does not require any pre-setup or training before the system can correctly be activated. While on the other hand both of the other presented models/platforms required intensive pre-work sometimes up to 4 weeks of classifier training or data collection to apply correct reasoning. In addition, the limited number of trial runs causes high doubts towards the accuracy and precision of the supplied results of both models.
7. Conclusion and future work

Surveying literature on the subject of context-awareness has revealed to a great extent that the current research on the topic shows shortfall in the area of situational awareness. Most current research resorts to using a limited set of sensors as means of gathering context information from surrounding environments. Context reasoning and inference might produce inaccurate outcomes as a result of depending on incomplete context data from a limited set of sources. Utilizing a small amount of sensation as an input to context aware systems will in turn convey a partial representation of the actual surrounding environment.

This research constructed a situational aware context architecture that makes use of a fairly higher amount of sensation which can be found on an off the shelf smart phone. The proposed architecture used seven various virtual and physical sensors to accurately represent the occurrence of a specific predefined context situation. A quantitative research approach was employed using surveys as a tool for data gathering to understand the wisdom of the crowd. The conducted survey expressed the user’s experience regarding the context parameters that identify a certain situation. The survey was applied to 155 subjects generating statistical data that varied in some cases from the initial activity hypothesis; the results of the survey were used to compliment the researcher’s initial assumptions to reach an optimum collection of context parameters.

The research also devised an approach that compliments situational awareness with habitat sensitive context reasoning. In the sense, that context inference will consider variations in the meaning of context in different geographical regions. For example, the factors and parameters that convey the occurrence of a meeting might differ from one region of the world to the other.

A prototype application was developed and the notion of context weights was utilized in an effort to quantify the importance of relevant sensors to the occurrence of
specific situations based on the results supplied from the localized survey. Experimental research was conducted to identify the optimum context weights intervals as to compliment the inference algorithm while maintaining the integrity of the reasoning results generated by the built prototype.

An experiment detailed in section 6 was setup to validate the research hypothesis and test the developed prototype mobile system illustrated in section 5.5 against 5 predefined context situations with variable dimensions. The results showed an overall accuracy rate of 85% with a confidence level of 66.8% successfully exceeding the research initial benchmark ambitions set by previous research works in the same domain. The prototype carried out more complex tasks with higher levels of accuracy.

For future work we aim to increase the time intervals used for the experimental trials to give the inference engine a larger set of data points, which should in turn increase the overall system accuracy and confidence. Another aim is to revise and update the localized survey with more detailed research questions to help generate a more accurate inference equation and to allow for more concrete system reasoning in the future. In addition, the researcher believes that adding a larger number of sensors in the future such as proximity and calendar appointments can further increase the overall system accuracy.
References


15. Chu Weijie; Mo Tong; Cui Jie; Wang Yuan; Xu Jingmin; Li Weiping; Lin Huiping, "A Context-Aware Services Development Model," In international joint conference on Service Sciences, May 2012.


APPENDIX A: Background Literature
Context Awareness Models

Resources, Actors and Policies Model (RAP)

RAP is a context awareness model for representation of general purpose contexts. The model is built to recognize the actors, resources and policies that compose any given environment [46].

The basic model is defined as a triple set \( C = <R, A, P> \) where,

- \( R \) is the set of resources available in a given environment.
- \( A \) is the set of actors that may interact with available resources.
- \( P \) is a set of context related policies that govern the use of resources by actors.

The RAP model is established with different actual contexts by filling the sets with actual context specific components. The mapping results in a precise context model \( C_s = <R_s, A_s, P_s> \) which permanently reflects a certain context situation.

![Figure 44: R.A.P Context Model [46]](image)

a) Resources:

A context resource is considered as a physical / virtual entity which generates or processes context information [46]. Resources can be divided into:

- **Passive Resources:** aim at capturing and storing context specific data, they are the resources attached in to the physical space in which the
actors are interacting. Such resources include Cameras, Noise Sensors and Orientation sensors embedded within the environment.

– **Active Resources:** can interact directly with the context and modify the context state. This type is a set of context resources attached to the actors that provide actor interactions related information such as user’s PDA.

\[ R = R_A \cup R_S \]

Each context resource has a specific identity that can be described by three features:

- **Resource Properties:** identifies the set of context information that a resource can offer.
- **Resource Services:** specifies the functionality provided by the resource.
- **Resource Influence Zone:** It is the 3D physical space in which the resource presence can be sensed [46].

b) **Actors:**

An actor is a physical or virtual entity that interacts directly with the context or utilizes the context resources to achieve its needs.

An Actor is characterized by:

- **Actor Resources:** The set of resources linked to a specified actor such as Position elements (RFID tags).
- **Context Request:** Identifies actor preferences associated with the context it will interact with.
- **Context Contract:** Identifies the actor’s privileges and tasks within a defined context [46].

c) **Policies:**

A policy represents a set of rules that must be fulfilled by actors or resources available in the context influence zone.
The model proposes a solid method for representing context as sets of actors, resources and policies in a pervasive environment. In the researcher’s opinion the model does not illustrate any means for scalability for specific context sets or policies. This would negatively affect the inference accuracy in constantly changing environments as there are no means to update or evolve existing policies or context definitions. Another point is that the model was designed for use within a fixed physical spaces or contained environment with no means of compatibility for mobile applications.
Self-Adapting Context Definition

The meaning of sensor data is defined relevant to the application. The application uses feedback to learn this meaning and identifies contexts based on it. Based on this concept an application centric meaning is defined as a mapping “M” from sensor data “Sd” to actions “A” [42].

\[ M: Sd \rightarrow A \]

The application recognizes environmental situations using its sensors. Sensor data is perceived as a state space in order for the application to reason about and compare situations. The sensor readings of n sensors form a unique point in the n-dimension space.

**Figure 45: State Space [42]**

From the standpoint of the application, a context is a combination of states inside the state space that share the same meaning. Where similar states should be located close to each other.

Different contexts are defined in terms of context edges. An edge is the bordering area between two different contexts. Hence, if neighboring states within the state space have different meanings then a context edge should exist between them. The meaning of a context is the same as any of the states it abstracts; it should influence the application in the same manner any of its states would [42].
The previous figure illustrates a two dimensional state space where 2 contexts exist. Every state in Context A should have the same meaning and the same goes for Context B. The meaning of a context is the same as that of the states it abstracts.

A layer of abstraction is introduced between the contexts and the sensor states called learning states. Learning states are used to define contexts instead of completely relaying on sensor states. A Learning state correctly represents the sensor states beneath it if the proper action is consistent across the learning state [42].

In the previous figure ls2 is considered to be inconsistent as it contains two regions with different required actions. “The size of the inconsistency is the distance between the context edge and the nearest learning-state boundary. A context edge based on these learning states is inaccurate by this amount.” [42]
“Learning states are compared to their neighbors, again using our similarity metric. Dissimilar neighbors have a context edge between them, while similar neighbors are in the same context. Context edge locations are compared to existing context definitions. If the bounds of existing definitions do not match the identified edges they are updated.” [42]

This Self adapting context definition proposes a means to automatically evolve context definitions to reach the most accurate context reasoning. The research however, only addresses the problem of adaptable context definitions without any reference to representation of context or means of interpretation. In addition this proposed model depends mainly on sensor based data acquisition for needed context information with no regard to any user context.
High Level Context Aware Systems

SenSay

SenSay (sensing & saying) is a context-aware mobile device that adapts its behavior based on its user's surrounding environment. It adapts to regularly changing environmental and physical states. To provide context information SenSay utilizes sensors that identify the light, motion, and sound attributes. The sensors are attached to parts of the human body with a central core, called the sensor box [52].

The sensay application provides the user with four different states of activity:

- **Uninterruptable:** If the user is engaged in a high priority activity such as a meeting or lecture and cannot be interrupted. The mobile device should be able to infer such a state and act accordingly.
- **Active:** If the user is engaged in communication blocking activities such as those associated with high noise, loud music or high levels of physical activity. Sensay is able to dynamically adjust its ringer volume and vibration mode accordingly.
- **Idle:** is defined as a time period when the user is not engaged in any high priority or communication blocking activities but rather is in an interruptible state.
- **Normal:** When the phone is not placed in any of the previously mentioned states by the decision module, the phone arrives at this default state. With no suggestions made by the user, the phone’s ringer and vibrate modes are set to their default values.

The sensay architecture is composed of five modules sensor box, sensor module, decision module, action module, and phone module.

![Figure 48: The Sensay Architecture][1]
Sensay Components [52]:

- **Sensor Box**: Contains all available sensors and is responsible for collection of raw sensor data.
- **Sensor Module**: responsible for querying the sensor box periodically and returning that data to the decision module. This querying is applied through a simple communication protocol.
- **Decision Module**: requests collection of sensor data and the calendar data of the user. Based on these inputs, the module defines the current state of the user and issues corresponding actions by the phone.
- **Action Module**: is responsible for issuing changes in setting and operation on the mobile phone and is controlled by the decision module. In addition it is also responsible for some basic operations on a given mobile device such as:
  - ringer control: off/low/medium/high
  - Vibrate control
  - Send SMS to caller
  - Make call Suggestions
  - Access electronic calendar
- **Phone Module**: provides access to the mobile phone operating system and user interface.

This application uses a number of wearable sensors to understand the users’ surrounding context and adapt a mobile device’s profile accordingly. The issue with this system is that the application was based upon context data derived from a number of wearable sensors which creates limitations on the use cases of this application. In addition the system does not offer any means of evolving the policies or rules which govern the context reasoning process which indicates that context definitions are not scalable and this also creates limitation on the inference process when system is introduced to a new environment.
**Context Aware Prompting System for Improving Physical Activity**

This section describes a system built using a smartphone application that infers certain physical activities and builds a physical activity condition, which is then utilized to notify the user with certain physical activity recommendation based on his inferred context. The system is built to utilize one sensor data type; specifically, the built in accelerometer. Then, it sends this data to a remote server where the data is processed and 3 types of higher level context data sets are generated; activity, activity level and activity duration. [16]

![Context-aware prompting system architecture](image)

**Figure 49: Context-aware prompting system architecture [16]**

The activity recognition process includes four parts [16]:

- Signal preprocessing: which is done on the server side,
- Time window selection: which is the time interval for the data collection,
- Feature extraction
- Classifier: which is a threshold based algorithm.

The system also utilizes historical data to understand physical activity patterns and set a suitable exercise plan. The final module of the system is the projection which
generates exercise advice to the user based on a set of rules derived from the WHO recommendations for physical activity.

The system was tested using 3 subjects who were required to perform a set of tasks presented in the following figure such as; walking, fast-walking and running for duration of 30 days. The results proved the system to have high accuracy in terms of activity recognition.

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>Duration(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sit-to-stand</td>
<td>From initially seated position to stand up and remains standing.</td>
<td>10</td>
</tr>
<tr>
<td>Stand-to-sit</td>
<td>From initially standing position to sit down and remains seated.</td>
<td>10</td>
</tr>
<tr>
<td>Stand-to-sit-to-lying</td>
<td>From initially standing to sit down on bed slowly and then lying to bed.</td>
<td>10</td>
</tr>
<tr>
<td>Lying-to-sit-to-stand</td>
<td>From initially lying to seated on bed slowly, then stand up and remain standing.</td>
<td>10</td>
</tr>
<tr>
<td>Sit-to-stand and walking</td>
<td>From initially seated to stand up, then walks with normal speed for 15 m, then sit down to chair.</td>
<td>15</td>
</tr>
<tr>
<td>Walking-to-fast-walking</td>
<td>Walking to fast-walking on running machine from 1 km/h to 8 km/h each velocity remains 50 seconds.</td>
<td>400</td>
</tr>
<tr>
<td>Running</td>
<td>Running on running machine from 6 km/h to 9 km/h, each velocity remains 50 seconds.</td>
<td>200</td>
</tr>
</tbody>
</table>

**Figure 50: Experiment Tasks [16]**

The system is built with a goal to infer simple physical tasks such as walking and running and not to deduce high level context situations which would suggest the high level of accuracy given the high system limitation in utilizing only one sensor to reach a conclusion. Another down side of this application is that the smart phone has to be set in a specific position and used as a wearable device in order for the system to work properly. Finally conducting the experimental trials on only 3 subjects is considered to be a significantly small sample size to validate the system’s accuracy.

**Figure 51: Sensor Deployment [16]**
**Emotion Sense**

EmotionSense is a mobile platform for social psychology studies based on mobile devices. The platform presents a number of features including the ability to sense emotions and activities of an individual, as well as verbal and proximity interactions among members of social groups.

“The EmotionSense system consists of several sensor monitors, a programmable adaptive framework based on a logic inference engine, and two declarative databases Knowledge Base and Action Base.”[31]

**Figure 52: Information flow in EmotionSense [31]**

The system contains [31]:

- **EmotionSense Manager**: Responsible for starting all the sensor monitors, the inference engine, and instantiating the Knowledge Base.
- **Sensor monitors**: Accelerometer monitor, Bluetooth Monitor and Location Monitor.
- **Action and Knowledge base**: Responsible for storing the inferred facts from the raw sensor data produced from different sensors. Facts are logged to Knowledge base by monitors, which are later used to generate actions by the inference engine.
- **Inference Engine**: Follows a set of rules to adapt the system behavior at run time based on changes in current activity or location of the current mobile device user.

- **Speaker Recognition Subsystem**: Contains features such as speech analysis and Silence detection.

- **Emotion Recognition Subsystem**: Ability to distinguish between 14 different narrow emotions with some degree of accuracy. [31]

<table>
<thead>
<tr>
<th>Broad emotion</th>
<th>Narrow emotions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Happy</td>
<td>Elation, Interest, Happy</td>
</tr>
<tr>
<td>Sad</td>
<td>Sadness</td>
</tr>
<tr>
<td>Fear</td>
<td>Panic</td>
</tr>
<tr>
<td>Anger</td>
<td>Disgust, Dominant, Hot anger</td>
</tr>
<tr>
<td>Neutral</td>
<td>Neutral normal, Neutral conversation, Neutral distant, Neutral tete, Boredom, Passive</td>
</tr>
</tbody>
</table>

**Figure 53: Emotion Clustering [31]**

This context aware application focuses on understanding and adapting to user emotions and activities. The application implements physical and virtual sensors to reach a conclusion about the user’s current emotional state. But, the application is not able to detect situations the user might be engaged in. In addition this application acquires context data through the use of a limited set of sensors which will affect the accuracy of the application.
APPENDIX B: Conducted Survey
Overview:

The purpose of this survey is to reach a better understanding of the elements present in a given environment that would be of benefit in inferring the context present in that environment.

We are focusing on five specific contextual situations:

- Party
- Meeting
- In Car
- Sleeping
- Movie Theatre

For each context, 10 elements that may contribute to the identification of the contextual situation are identified:

1. Sound Level
2. Sound Type
3. Motion
4. Time
5. Day
6. Ringer Usage
7. Lighting
8. Absolute Position
9. Number of people in vicinity
10. Location

The following survey should help the researcher define the importance of each factor in relation with each of the previous contexts.
Context (PARTY)

Kindly assign the following factors to ONLY ONE of the columns below according to their importance in defining a PARTY.

1. Sound Level
2. Sound Type
3. Motion
4. Time
5. Day
6. Ringer Usage
7. Lighting
8. Absolute Position
9. Number of people in vicinity
10. Location

<table>
<thead>
<tr>
<th>High Importance</th>
<th>Average Importance</th>
<th>Low Importance</th>
<th>Irrelevant</th>
</tr>
</thead>
<tbody>
<tr>
<td>(0.6)</td>
<td>(0.3)</td>
<td>(0.1)</td>
<td>(0)</td>
</tr>
</tbody>
</table>

In your opinion, kindly assign a value to each of the following characteristics based on its importance in determining the presence of a PARTY. The values should be: (HI: High Indicator, AI: Average Indicator, LI: Low Indicator, I: Irrelevant.)

1- Sound Level:

   a) High      b) Average      c) Low      d) None

2- Sound Type:

   a) Speech      b) Noise      c) Music      d) Other: ________

3- Motion:

   a) High      b) Average      c) Low      d) None
4- Time:

<table>
<thead>
<tr>
<th></th>
<th>a) Day</th>
<th>b) Evening</th>
<th>c) Night</th>
<th>d) All options are valid</th>
</tr>
</thead>
</table>

5- Day:

<table>
<thead>
<tr>
<th></th>
<th>a) Weekend</th>
<th>b) Weekday</th>
</tr>
</thead>
</table>

6- Ringer Usage on your mobile phone:

<table>
<thead>
<tr>
<th></th>
<th>a) High</th>
<th>b) Silent</th>
<th>c) Vibration</th>
<th>d) Silent &amp; Vibration</th>
</tr>
</thead>
</table>

7- Light:

<table>
<thead>
<tr>
<th></th>
<th>a) Bright</th>
<th>b) Medium</th>
<th>c) Dark</th>
<th>d) All options are valid</th>
</tr>
</thead>
</table>

8- Absolute Position:

<table>
<thead>
<tr>
<th></th>
<th>a) Rapidly</th>
<th>b) Regularly</th>
<th>c) Stable</th>
<th>d) Stationary</th>
</tr>
</thead>
</table>

9- Number of people in vicinity of observer (less than 5m):

<table>
<thead>
<tr>
<th></th>
<th>a) High</th>
<th>b) Medium</th>
<th>c) Low</th>
<th>d) None</th>
</tr>
</thead>
</table>

10- Location:

<table>
<thead>
<tr>
<th></th>
<th>a) Office</th>
<th>b) Home</th>
<th>c) Other: ____________</th>
</tr>
</thead>
</table>

105
Context (MEETING)

Kindly assign the following factors to ONLY ONE of the columns below according to their importance in defining a Meeting.

1. Sound Level
2. Sound Type
3. Motion
4. Time
5. Day
6. Ringer Usage
7. Lighting
8. Absolute Position
9. Number of people in vicinity
10. Location

<table>
<thead>
<tr>
<th>High Importance (0.6)</th>
<th>Average Importance (0.3)</th>
<th>Low Importance (0.1)</th>
<th>Irrelevant (0)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

In your opinion, kindly assign a value to each of the following characteristics based on its importance in determining the presence of a MEETING. The values should be: (HI: High Indicator, AI: Average Indicator, LI: Low Indicator, I: Irrelevant.)

1- Sound Level:

a) High  b) Average  c) Low  d) None

2- Sound Type:

a) Speech  b) Noise  c) Music  d) Other: _________

3- Motion:

a) High  b) Average  c) Low  d) None
4- **Time:**

<table>
<thead>
<tr>
<th>a) Day</th>
<th>b) Evening</th>
<th>c) Night</th>
<th>d) All options are valid</th>
</tr>
</thead>
</table>

5- **Day:**

<table>
<thead>
<tr>
<th>a) Weekend</th>
<th>b) Weekday</th>
</tr>
</thead>
</table>

6- **Ringer Usage on your mobile phone:**

<table>
<thead>
<tr>
<th>a) High</th>
<th>b) Silent</th>
<th>c) Vibration</th>
<th>d) Silent &amp; Vibration</th>
</tr>
</thead>
</table>

7- **Light:**

<table>
<thead>
<tr>
<th>a) Bright</th>
<th>b) Medium</th>
<th>c) Dark</th>
<th>d) All options are valid</th>
</tr>
</thead>
</table>

8- **Absolute Position:**

<table>
<thead>
<tr>
<th>a) Rapidly</th>
<th>b) Regularly</th>
<th>c) Stable</th>
<th>d) Stationary</th>
</tr>
</thead>
</table>

9- **Number of people in vicinity of observer (less than 5m):**

<table>
<thead>
<tr>
<th>a) High</th>
<th>b) Medium</th>
<th>c) Low</th>
<th>d) None</th>
</tr>
</thead>
</table>

10- **Location:**

<table>
<thead>
<tr>
<th>a) Office</th>
<th>b) Home</th>
<th>c) Other: ___________</th>
</tr>
</thead>
</table>
## Context (IN CAR)

Kindly assign the following factors to ONLY ONE of the columns below according to their importance in defining an IN CAR.

1. Sound Level
2. Sound Type
3. Motion
4. Time
5. Day
6. Ringer Usage
7. Lighting
8. Absolute Position
9. Number of people in vicinity
10. Location

<table>
<thead>
<tr>
<th>High Importance</th>
<th>Average Importance</th>
<th>Low Importance</th>
<th>Irrelevant</th>
</tr>
</thead>
<tbody>
<tr>
<td>(0.6)</td>
<td>(0.3)</td>
<td>(0.1)</td>
<td>(0)</td>
</tr>
</tbody>
</table>

In your opinion, kindly assign a value to each of the following characteristics based on its importance in determining if the user is IN CAR. The values should be: (HI: High Indicator, AI: Average Indicator, LI: Low Indicator, I: Irrelevant.)

1- **Sound Level**:

<table>
<thead>
<tr>
<th>a) High</th>
<th>b) Average</th>
<th>c) Low</th>
<th>d) None</th>
</tr>
</thead>
</table>

2- **Sound Type**:

<table>
<thead>
<tr>
<th>a) Speech</th>
<th>b) Noise</th>
<th>c) Music</th>
<th>d) Other: ________</th>
</tr>
</thead>
</table>

3- **Motion**:

<table>
<thead>
<tr>
<th>a) High</th>
<th>b) Average</th>
<th>c) Low</th>
<th>d) None</th>
</tr>
</thead>
</table>
4- Time:

<table>
<thead>
<tr>
<th>a) Day</th>
<th>b) Evening</th>
<th>c) Night</th>
<th>d) All options are valid</th>
</tr>
</thead>
</table>

5- Day:

<table>
<thead>
<tr>
<th>a) Weekend</th>
<th>b) Weekday</th>
</tr>
</thead>
</table>

6- Ringer Usage on your mobile phone:

<table>
<thead>
<tr>
<th>a) High</th>
<th>b) Silent</th>
<th>c) Vibration</th>
<th>d) Silent &amp; Vibration</th>
</tr>
</thead>
</table>

7- Light:

<table>
<thead>
<tr>
<th>a) Bright</th>
<th>b) Medium</th>
<th>c) Dark</th>
<th>d) All options are valid</th>
</tr>
</thead>
</table>

8- Absolute Position:

<table>
<thead>
<tr>
<th>a) Rapidly</th>
<th>b) Regularly</th>
<th>c) Stable</th>
<th>d) Stationary</th>
</tr>
</thead>
</table>

9- Number of people in vicinity of observer (less than 5m):

<table>
<thead>
<tr>
<th>a) High</th>
<th>b) Medium</th>
<th>c) Low</th>
<th>d) None</th>
</tr>
</thead>
</table>

10- Location:

<table>
<thead>
<tr>
<th>a) Office</th>
<th>b) Home</th>
<th>c) Other: ___________</th>
</tr>
</thead>
</table>
## Context (SLEEPING)

Kindly assign the following factors to ONLY ONE of the columns below according to their importance in defining that a person is **Sleeping.**

1. Sound Level
2. Sound Type
3. Motion
4. Time
5. Day
6. Ringer Usage
7. Lighting
8. Absolute Position
9. Number of people in vicinity
10. Location

<table>
<thead>
<tr>
<th></th>
<th>High Importance (0.6)</th>
<th>Average Importance (0.3)</th>
<th>Low Importance (0.1)</th>
<th>Irrelevant (0)</th>
</tr>
</thead>
</table>

In your opinion, kindly assign a value to each of the following characteristics based on its importance in determining that a person is **SLEEPING.** The values should be: (HI: High Indicator, AI: Average Indicator, LI: Low Indicator, I: Irrelevant.)

### 1- Sound Level:

<table>
<thead>
<tr>
<th></th>
<th>a) High</th>
<th>b) Average</th>
<th>c) Low</th>
<th>d) None</th>
</tr>
</thead>
</table>

### 2- Sound Type:

<table>
<thead>
<tr>
<th></th>
<th>a) Speech</th>
<th>b) Noise</th>
<th>c) Music</th>
<th>d) Other:__________</th>
</tr>
</thead>
</table>

### 3- Motion:

<table>
<thead>
<tr>
<th></th>
<th>a) High</th>
<th>b) Average</th>
<th>c) Low</th>
<th>d) None</th>
</tr>
</thead>
</table>
4- Time:

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>a) Day</td>
<td>b) Evening</td>
<td>c) Night</td>
<td>d) All options are valid</td>
</tr>
</tbody>
</table>

5- Day:

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>a) Weekend</td>
<td>b) Weekday</td>
</tr>
</tbody>
</table>

6- Ringer Usage on your mobile phone:

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>a) High</td>
<td>b) Silent</td>
<td>c) Vibration</td>
<td>d) Silent &amp; Vibration</td>
</tr>
</tbody>
</table>

7- Light:

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>a) Bright</td>
<td>b) Medium</td>
<td>c) Dark</td>
<td>d) All options are valid</td>
</tr>
</tbody>
</table>

8- Absolute Position:

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>a) Rapidly</td>
<td>b) Regularly</td>
<td>c) Stable</td>
<td>d) Stationary</td>
</tr>
</tbody>
</table>

9- Number of people in vicinity of observer (less than 5m):

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>a) High</td>
<td>b) Medium</td>
<td>c) Low</td>
<td>d) None</td>
</tr>
</tbody>
</table>

10- Location:

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>a) Office</td>
<td>b) Home</td>
<td>c) Other: ___________</td>
</tr>
</tbody>
</table>
**Context (MOVIE THEATRE)**

Kindly assign the following factors to ONLY ONE of the columns below according to their importance in defining that a person is in a **MOVIE THEATRE**.

1. Sound Level
2. Sound Type
3. Motion
4. Time
5. Day
6. Ringer Usage
7. Lighting
8. Absolute Position
9. Number of people in vicinity
10. Location

<table>
<thead>
<tr>
<th>High Importance (0.6)</th>
<th>Average Importance (0.3)</th>
<th>Low Importance (0.1)</th>
<th>Irrelevant (0)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

In your opinion, kindly assign a value to each of the following characteristics based on its importance in determining the presence of a person in a **MOVIE THEATRE**. The values should be: (HI: High Indicator, AI: Average Indicator, LI: Low Indicator, I: Irrelevant.)

**1- Sound Level:**

- a) High
- b) Average
- c) Low
- d) None

**2- Sound Type:**

- a) Speech
- b) Noise
- c) Music
- d) Other: ________

**3- Motion:**

- a) High
- b) Average
- c) Low
- d) None
<table>
<thead>
<tr>
<th>4- Time:</th>
</tr>
</thead>
<tbody>
<tr>
<td>a) Day</td>
</tr>
<tr>
<td>b) Evening</td>
</tr>
<tr>
<td>c) Night</td>
</tr>
<tr>
<td>d) All options are valid</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>5- Day:</th>
</tr>
</thead>
<tbody>
<tr>
<td>a) Weekend</td>
</tr>
<tr>
<td>b) Weekday</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>6- Ringer Usage on your mobile phone:</th>
</tr>
</thead>
<tbody>
<tr>
<td>a) High</td>
</tr>
<tr>
<td>b) Silent</td>
</tr>
<tr>
<td>c) Vibration</td>
</tr>
<tr>
<td>d) Silent &amp; Vibration</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>7- Light:</th>
</tr>
</thead>
<tbody>
<tr>
<td>a) Bright</td>
</tr>
<tr>
<td>b) Medium</td>
</tr>
<tr>
<td>c) Dark</td>
</tr>
<tr>
<td>d) All options are valid</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>8- Absolute Position:</th>
</tr>
</thead>
<tbody>
<tr>
<td>a) Rapidly</td>
</tr>
<tr>
<td>b) Regularly</td>
</tr>
<tr>
<td>c) Stable</td>
</tr>
<tr>
<td>d) Stationary</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>9- Number of people in vicinity of observer (less than 5m):</th>
</tr>
</thead>
<tbody>
<tr>
<td>a) High</td>
</tr>
<tr>
<td>b) Medium</td>
</tr>
<tr>
<td>c) Low</td>
</tr>
<tr>
<td>d) None</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>10- Location:</th>
</tr>
</thead>
<tbody>
<tr>
<td>a) Office</td>
</tr>
<tr>
<td>b) Home</td>
</tr>
<tr>
<td>c) Other: ___________</td>
</tr>
</tbody>
</table>
APPENDIX C: Experimental Results Form
Experiment Results Form

<table>
<thead>
<tr>
<th>Scenario Name</th>
<th>Light Intensity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Experiment Day</td>
<td>Motions</td>
</tr>
<tr>
<td>Location</td>
<td>Sound level</td>
</tr>
<tr>
<td>Experiment Time</td>
<td>Ringer</td>
</tr>
</tbody>
</table>

Description of Change:

Results for Selected context:

<table>
<thead>
<tr>
<th>Context Scenario Identified by System</th>
<th>Actual Context Scenario</th>
<th>Confidence Level</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Results for the other scenarios:

|                                      |                         |                  |
|                                      |                         |                  |
|                                      |                         |                  |
|                                      |                         |                  |

General Comments: