FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

FEUP

Smart Mobile Sensing for Measuring Quality of Experience (QoE) in Urban Public Transports

João Miguel Guerra Vieira

Mestrado Integrado em Engenharia Informática e Computação

Supervisor: Maria Teresa Galvão Dias (Prof. Dr.)

July 20, 2011

Smart Mobile Sensing for Measuring Quality of Experience (QoE) in Urban Public Transports

João Miguel Guerra Vieira

Mestrado Integrado em Engenharia Informática e Computação

Approved in oral examination by the committee:

Chair: António Fernando Vasconcelos Cunha Castro Coelho
External Examiner: João Paulo Pereira
Supervisor: Maria Teresa Galvão Dias

July 20, 2011

Abstract

In our current society which is characterized by mobility, individuality and comfort requirements, there is a need for real-time information and services that make people's life easier. In Public Transportation (PT), immersed in the city dynamics, live accurate data is crucial for enhancing the travelling experience. Lately, the number of smartphone applications for trip planning, mobile ticketing and validation has grown considerably, leading to an increase in comfort. However, these solutions still do not take into account the user's affective state or his Quality of Experience (QoE). Emotion, that is fundamental to the human experience, influencing rational decision making, is rarely taken into account. This subjective matter is left out of the PT choosing criteria, until now marked by objective factors, such as trip duration, service delay, cost and number of changes.

Thus, in this work we present a new solution to identify and understand patterns of user satisfaction based on the collection of affective and context information during daily commutations. The interaction loop comprehends three steps: the collection of sensorial data; analysis and pattern extraction in the cloud; and delivery of service. The collection of data permits the definition of personal preference profiles looking at the correlations between dependent variables, e.g. noise can be inversely proportional to relaxation. Matching these profiles against live context data from the PT web services and the opportunistic sensing network allows to find and suggest more comfortable routes to the customer.

This smart application is comprised of two platforms: the Mobile Sensing Platform (MSP), the front-end, built on a smartphone application, responsible for gathering all the user related data and interacting with him; and the Cloud2Bubble (C2B), the reasoning system in the cloud, being developed in the Imperial College London. The focus of this project lies in the MSP architecture and implementation - the proof of concept PTSense Android application - and its integration with the C2B. The current growth of pervasive systems allows the use of everyday devices, specially smartphones, wireless sensors and wearable computers to smoothly collect multi-modal sensorial data during the journeys. In our solution, we make use of the smartphone sensors, environment sensors and personal wearable sensors to collect such data, while the user input serves as a solid validation of the information received.

In this work we present the design, implementation and testing of the PTSense smart mobile application, with special focus on its feasibility. For that, a Usability Test and a Feature test were conducted during the development. The results concluded a solid, understandable design for urban user adoption and validated the concept by showing signs of moderate correlation between variables of the sensed data and the user feedback. The project is not yet finished, but the well-thought ground is ready to support the future development. In the end, we believe this solution can provide a better informed decision and enhanced QoE to customers and, so, adding value to service providers.

Resumo

Na nossa sociedade atual que se caracteriza pela mobilidade, individualidade e requisitos de conforto, verifica-se uma crescente necessidade de informação em tempo real e serviços que tornem a vida das pessoas mais fácil. No contexto dos Transportes Públicos (TP), estes dados são cruciais para melhorar a experiência de viajar. Recentemente, o número de aplicações móveis para o planeamento de rotas, emissão de bilhetes e validação móvel tem crescido consideravelmente, levando a um aumento do conforto. No entanto, estas soluções ainda não tomam em conta o estado afetivo dos utilizadores ou a sua Qualidade da Experiência (QoE). As emoções, que são fundamentais para a experiência humana, influenciando a tomada racional de decisões, raramente são tomadas em conta. Esta questão subjetiva é deixada de fora no critério de escolha de TP, até agora caracterizado por fatores objetivos, como a duração da viagem, tempo de atraso, custo e número de comutações.

Assim, neste trabalho apresentamos uma nova solução para identificar e compreender os padrões de satisfação dos utilizadores com base na recolha de informação de contexto e estado afetivo durante as comutações diárias. A interação compreende três etapas: a recolha dos dados sensoriais; análise e extração dos padrões na *Cloud*; e prestação do serviço. A recolha de dados permite a definição de perfis pessoais de preferências olhando para as correlações e dependências entre as variáveis. A combinação desses perfis com dados do contexto em tempo-real provenientes de serviços *web* dos TP e da rede de *Opportunistic Sensing* permite encontrar e sugerir rotas mais confortáveis para o cliente.

Esta aplicação inteligente é composta por duas plataformas: a Plataforma de Mobile Sensing (PMS), criada sobre uma aplicação móvel, responsável por agregar todos os dados relacionados com o utilizador e interagir com o mesmo; e o Cloud2Bubble (C2B), o sistema de inferências na *cloud*, a ser desenvolvido no Imperial College Londres. O foco deste projeto reside na arquitetura e implementação da PMS - a aplicação Android PTSense como prova de conceito - e na sua integração com o C2B. O atual crescimento dos sistemas ubíquos permite o uso de aparelhos comuns, especialmente *smartphones*, sensores *wireless* e roupa "inteligente" para facilmente recolher dados sensoriais multimodais durante as viagens. Nesta solução, esta recolha é feita através dos sensores do *smartphone*, sensores de contexto e sensores de vestuário pessoais, enquanto o *input* do utilizador serve de validação dos dados recebidos.

Neste trabalho apresentamos o desenho, implementação e teste da aplicação móvel PTSense, com especial ênfase na sua viabilidade. Para isso, foram realizados dois testes de Usabilidade e de Funcionalidades durante o desenvolvimento. Os resultados permitiram concluir uma arquitectura sólida e compreensível para adoção urbana e validar o conceito através de sinais de correlação moderada entre as variáveis dos dados sensoriais e do *feedback* do utilizador. O projeto ainda não está terminado, mas o desenho bem pensado está apto a integrar e suportar o desenvolvimento futuro. No final, acreditamos que esta solução pode proporcionar uma decisão mais informada e melhorar a QoE dos clientes e desta forma, gerar valor aos prestadores de serviços.

Acknowledgements

Seventeen years in, it is finally the time to thank all those people who made this quest an adventure. Through ups and downs, highways and country roads I gained some of the best education and wisdom, friends, experiences and memories that turned me the man I dreamt to be. For that I was blessed with glorious persons to whom I shall show my appreciation. Ultimately, they helped me to embrace a new world outside, and that is why I will write in different languages.

Tenho de começar pelos meus pais, Clara e Manuel, que me ensinaram os primeiros passos. Que com a sua dedicação, inteligência e discernimento sempre me motivaram a querer mais e trabalhar para o sucesso. Por manterem o suporte, alegria e entusiasmo mesmo quando eu quis levantar outros voos, sair de casa e do país. Sinto-me privilegiado pela confiança e liberdade que me deram e que me proporcionaram experiências únicas, desde criança até adulto. O maior dos obrigados e a esperança de continuar a alegrar-vos o mais que puder.

Ao meu irmão, David, que sempre me serviu de exemplo como irmão mais velho. Aos tempos de crianças e divertimento e aos tempos de gente crescida e profissionalismo. Desejo-te também o melhor futuro.

Voltando ao trabalho. Este projecto não teria sido possível sem a participação e autonomia da minha coordenadora, Prof. Teresa Galvão, e do Pedro Maurício Costa. Eles, que me incentivaram e me deram a conhecer provavelmente a área ciêntifica que atualmente mais me interessa. Um grande agradecimento pela liberdade que me deram e pela enorme (!) experiência adquirida, mesmo estando longe. Este projecto e as suas ideias permitiram-me expandir horizontes e ultrapassar fronteiras.

To all my friends, that are too many, whom in different ways also contributed to my success.

To start, my oldest friends, that kept me smiling on all the nonsense discussions and "activities" we had. Even after years and distance you keep me missing you. You all have been amazing and I hope we'll stay that way. Special thanks to Francisca, Manuel, Gonçalo C., Gonçalo I., Rita S., Rita C. and the rest of the "Copos" family. Even more special to Afonso, who accompanied me during the whole university years as a flatmate, in Porto and Budapest. For all the work and crazy experiences we spent together.

The FEUP family, "Tios e primos", without whose knowledge, support and motivation I would be just another one. To these people that showed me there can be work with joy and even in the hardest moments - deliveries! - the happiness never gave up. You are the best workmates I could ever wish for in my future life. Thank you all, and also best wishes for you. See you in success!

The Erasmus family. All that tremendous people that made this year the best of my life. I cannot express how much I loved Budapest, living there with all that family - which is pretty much what you are - and the unforgettable moments. My small country became bigger and with you I'm a citizen of the world. Some of you I will see more often, but I hope we'll all keep in touch - I'm sure we will. Thank you for the best memories. See you soon.

Ad una persona speciale, Egle, la cui felicità e unicità è sufficiente a farmi sorridere. Il cui entusiasmo, passione, energia e sogni ancora mi stupiscono e mi affascinano. Un grande grazie per essere sempre al mio fianco e per aver creduto in me durante tutto quest'anno passato insieme.

Finally, to the rest of my family and friends. And the last, but not the least, to FEUP and all my teachers, for the exceptional education you offered me. I'm proud of graduating here.

João Miguel Guerra Vieira
July 2012

À minha Família,
cuja fé e apoio sempre me acompanharam
nas minhas decisões difícieis.

Contents

List of Figures

List of Tables

LIST OF TABLES

Abbreviations

API	Application Programming Interface
BLE	Bluetooth Low Energy, more commonly known as Bluetooth Smart, included in the Bluetooth 4.0 standard
C2B	Cloud2Bubble, the cloud computing complementary project also under development in the Imperial College London
EEG	Electroencephalography, recording of electrical activity along the scalp
GPS	Global Positioning System
GSR	Galvanic Skin Response, measuring the electrical conductance of the skin
GUI	Graphical User Interface, sometimes only referred as UI
HCI	Human-Computer Interaction
HR	Heart Rate
ICS	Android Ice Cream Sandwich, version 4.0 of Android OS
ICT	Information and Communication Technology
IoT	Internet of Things
IPv6	Internet Protocol version 6
IT	Information Technology
JSON	JavaScript Object Notation, a data representation format
MSP	Mobile Sensing Platform, the sensing platform underlying the mobile application
NFC	Near-Field Communication, close-range wireless communication technology
OOP	Object-Oriented Programming
OS	Operative System
PT	Public Transportation
QoE	Quality of Experience
QoS	Quality of Service
RAD	Rapid Application Development
REST	REpresentational State Transfer. RESTful web services are described in section 4.2.2
RFID	Radio-frequency identification
SOA	Service Oriented Architecture
SOAP	Simple Object Access Protocol
URI	Unified Resource Identifier, used to identify a resource usually over a network
WMSN	Wireless Mobile Sensor Networks, or just WSN
WPAN	Wireless Personal Area Network

Chapter 1

Introduction

When it gets to choosing between services, providers or products it is from the common knowledge that everyone chooses the one which makes them feel comfortable, engaged or simply on time. So, the Quality of Service (QoS) is no longer the only mean of evaluating a service or product but the user perspective is coming to take part of it too. The Quality of Experience (QoE) is now an important part of the planning, development and evaluation of not only services transmitting multimedia data, but also for services involving multi-modal human-machine interaction [MEK+09]. By assessing the users' satisfaction and understanding their personal preferences we are closer to providing him a better environment and service. For example, some prefer noisy environments when they are in bars while others do not; some do not bother with reckless driving or ambient music in public transports while others do.

Personalized services are not new, they started a long time ago to treat customers better. The concept is that every people is different and so the applications, services or treatment should be directed to the customer's preferences in order to improve their QoE. Understanding them, though is more difficult. Thus, in this project it is proposed a Mobile Sensing Platform (MSP) built on a smartphone application that helps the collection of user related data, such as affective and context data. The idea is that looking at the correlation between these variables we are able to define patterns and later user profiles. In this document we will present the solution for this problem in the context of Public Transportation. The reason is because PT lacks subjective factors in the choosing criteria while providing a rich sensorial environment at the same time. By adding personalized classification based on the current conditions to the alternative list, we provide the commuters with a better informed decision that finally leads to more comfortable rides and experiences. The satisfaction of the costumers ultimately has impact on the service providers by improving its overall comfort, reliability, adoption and reputation.

However, there is still no simple solution for the collection of this data as of these days. Thus, in this project we present a smartphone prototype based on the vastly conducted multidisciplinary research. It makes use of the sensorial information collected from a user's smartphone's sensors and environment and body sensors to assess affective and contextual data. This mobile application will be the front-end of a cloud computing infrastructure where predictions are made to support the suggestion of more comfortable routes. With this framework we propose a solution for enhancing QoE in an autonomous and less unobtrusive way with the potential to create user-centered services.

1.1 Context

This project is to be integrated with the Cloud2Bubble project, currently being developed by Pedro Maurício Costa in the Imperial College London. That work [CPCG12] consists of the whole QoE enhancing framework based on the assessment of user affective data and environment variables to allow the recommendation of new services through individual user profiles. In practical terms, this framework can be applied in the most multimodal services where the user can perceive the quality of service in a satisfactory criteria. However, to test it, Public Transportation gives us a rich environment where noise, harsh movements and number of people, for example, define user's comfort, preferences and, lately, decisions. The context of this project then relies in building a front-end smartphone application to collect data in public transportation and feed the C2B platform. In fact, both projects define a smart mobile application using the benefits of Cloud computing.

Public Transportation provides a rich context for a QoE enhancing solution. Immersed in the city dynamics, commuting between services can be fastidious and sometimes frustrating on certain conditions. It provides plenty of sensorial data that can help to assess - affective data - and understand - context data - the patterns of satisfaction of customers. Testing is also easy to prepare and it covers a wide range of users, all the urban community. Besides the rich environment, there is also missing features in the PT services that further explain this choice. Those are part of the motivation and are discussed in section 1.3.

In terms of scientific fields, this project started with the very broad fields of Human-Computer Interaction (HCI), Affective Computing, Mobile Computing, Usability and Quality of Experience. With the research and maturation of the solution, newer fields were "discovered". HCI was narrowed to Pervasive and Ubiquitous Computing which in turned branched to Mobile Sensing, Smartphone Sensing, (Wireless) Sensor Networks and the Internet of Things (IoT). Affective Computing opened doors to Wearable Computing, Emotion Assessment and Physiological signal's analysis. At the same time, the concept of Smart Mobile Computing and Cloud Computing have always been present in this project. In the C2B, there is also another very important field, Machine Learning, that will be referred but not covered in this work.

This project grown to be very complex and ambitious, a multidisciplinary project on the cutting edge of technology. Specially since advances in these areas are going different directions, while in this project we try to integrate them in building a forward thinking application.

1.2 Project

This project, although it is to be integrated or merged with another ongoing project, started from the ground. The idea of collecting affective information to assess QoE and personalize the PT choosing criteria received plenty of brainstorming before it met the final pervasive solution. As any project built from scratch should do, all the software development process was firmly followed. It was used the Rapid Application Development (RAD) given the testing and prototyping aspect of the project. RAD allows the incremental prototypes to be quickly tested providing new insightful ideas and improvements for the next iteration. Therefore, the solution developed was incremental and tracked the following steps, covered in order in this document:

1. Research related to the state of the art, context and related work, which gave the insights for the conceptual solutions;

2. Brainstorming and defining the data flow of the application and interaction with the C2B project, i.e. using collected data to deliver better services. The three stages of the interaction loop: Sense, Analyse and Suggest;

3. Definition of the architecture around both platforms, how and when they interact. Building an Service Oriented Architecture (SOA) web service using Cloud Computing and REpresentational State Transfer (REST) interface. The data is sent through JavaScript Object Notation (JSON);

4. Definition of the architecture of the sensing platform to collect data using by redefining the research in step 1. This platform is called the Mobile Sensing Platform and takes the focus of this work. It defines a sensor network around smartphone sensors, environment sensors and personal wearables sensors that gathers all the data in the smarphone and transmits it to the cloud. It comprehends the following steps:

   - **Context-Aware Smartphone Sensing** – Collection and extraction of features from the smartphone's sensors;
   - **Wireless Collection of Sensor Data** – Wireless communication with other available sensors that collect affective and also context data related to the user and the environment;
   - **User Validation** – User feedback or reviews through direct input;
   - **Cloud Communication** – Transmission and reception of information from the cloud, from and to the user.

5. Implementation of a mobile application, PTSense. Starting from the functional detailed prototype to the coding in Android;

6. Iterative testing both in the functional prototype phase and in the first stage of the implementation.

Because this project is a partnership between FEUP and the Imperial College London, the development was conducted remotely between Porto, London and Budapest, my former location. Brainstorming was made in short meetings or over calls while user testing, in both London and Porto, was in person.

1.3 Motivation and Goals

Within large metropolitan areas, the growing adoption of Public Transportation reaches mass levels in some occasions. As so, providers battle to better serve their costumers and every day commuters. However, in most of the cases they no longer desire, nor are they capable, to expand the current infrastructures, due to economical and environmental reasons. Instead, the optimisation of resources is preferred. Measuring and enhancing Quality of Experience in such services has the potential to draw more costumers due to its higher level of comfort and satisfaction as well as easier routes to one's workplace, home or every spontaneous destination.

Generally speaking, conventional methods of assessing and evaluating a users' Quality of Experience in all kinds of service lack autonomy and invisibility. They are usually related to large community polls or interrogating individuals which not only burdens the user but also carries too much man work and they are unlikely to scale to mass adoption [LML+11]. On the other hand, although emotion is part of the interaction loop, it is not taken into account when assessing satisfaction. The choosing criteria of transportation means is based on objective factors, such as duration of trips, service delays, cost and number of changes. Adding personalization to this criteria permits customers to choose a service also based on current conditions and their own preferences. Figure 1.1 presents an example where this information would help a user to choose waiting 3 more minutes for the next bus and having much better conditions, e.g. less crowded or noisy, than reaching the destination fast but frustrated about the noisy people inside the first bus. Therefore, as stated in the introductory remarks of this chapter, assessing, understanding and improving the QoE provides an accurate mean to better serve the customers.

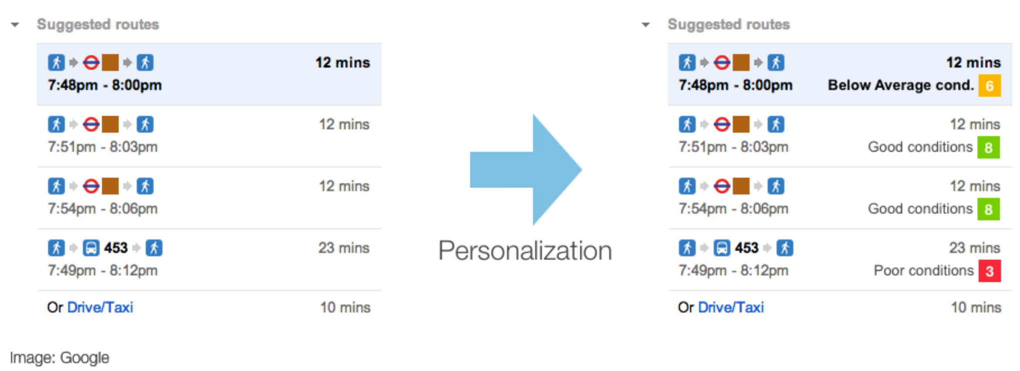


Figure 1.1: Example of adding personalized classification to the PT alternatives choosing criteria

The main goal of this work is to provide a proof of concept tool to assess QoE, during PT journey, in a more autonomous and ubiquitous way, promoting sustained usage over the long-term. This tool should use the collected information to deliver a personalized service to the customer, as shown in Figure 1.1. Further objectives comprehend the details of such platform based on the new technology tendencies, such as mobile computing, which also bring a new motivation.

The number of intelligent systems that ease and help with every day life situations is growing and in the near future it is expected to be so integrated into our routines that nobody can doubt it, much like electricity nowadays. The concept of Smart Spaces where sensors and actuators communicate to help us feel safer or give us less work is not new, nor the Mobile Sensor Networks (MSP) for tracking packages around the globe, nor the real-time routing smartphone applications. However is the Smartphone that is grabbing much of the attention and hype over the last years. It serves not only as the key computing and communicating mobile device of choice, but nowadays it can offer a whole new set of capabilities and opportunities [LML+10]. Today's devices are programmable and software is easily deployed through application delivery channels like the Apple AppStore or Google Android Market, but more importantly, they also come with a rich set of embedded sensors, such as accelerometer, gyroscope, digital compass, global positioning system (GPS), microphone, camera, barometer, thermometer and relative humidity sensor. The collectivity of this sensors is enabling new applications in a wide variety of domains, such as healthcare [Gin] and context-aware [ACRC09] and evolving its mobile sensing research area. This supports the motivation for a mobile application based on the users smartphone. Also, urban public transportation users need dynamic real-time information such as delays and fast free access to services everywhere so they do not miss connections in short time crowded commutations. In this context, mobile applications are useful in improving the user experience and comfort.

The concrete objective of this project lies on the demonstration - a proof of concept - of the feasibility of a multi-disciplinary and advanced system to assess and improve the QoE in the particular context of Public transportation, using a smart mobile phone application. The initial prototype should then be improved through research on the field. Objectively, the goals for this work are as follow:

- Development of a prototype of an application for smartphones that allows the collection of affective data in public transportation;

- Assist in the collection of experimental data using the developed prototype by a sample of users in a controlled environment;

- Research related to improving the prototype through the use of new information and communication technologies

1.4 Structure of the document

The structure of the document is as follows. Chapter 2 presents a critic analysis of the state of the art in the specified research areas and the previous related work. Chapter 3 discusses the conceptual idea and workflow of the solution, architectured in Chapter 4. In Chapter 5 and Chapter 6 we present the details of the Mobile Sensing Platform and its implementation - the PTSense mobile application -, respectively. Chapter 7 covers the user tests conducted and its results, followed by the concluding remarks presented in Chapter 8.

Chapter 2

Literature Review

Through the next chapter it will be described the current State Of the Art in the areas related to this project and address some of the commercial and research innovative works. The technologies used to develop the system will also be presented and the way they integrate with each other and the whole solution. For each subject I will discuss choices and directions that can be applied to our framework and comparisons between alternatives.

The innovative aspect of this project along with its wide architecture makes it hard to objectively compare to any known or current work in progress. Therefore, it gathers bits and pieces of every technology presented next and pursuits a new intelligent and complex architecture to efficiently fulfil our objectives. Also, as some of this research issues are relatively new there are some gaps and holes for which we propose alternatives.

In sum, current works cannot solve our problem as of now, but we propose a framework, even if theoretical, that integrates these innovative topics, giving new insights and opportunities for this areas to evolve and in the future, correctly implementing our solution.

2.1 Introduction

Measuring emotions is not a new topic, neither wireless communication or even smartphone sensing. However, they are hot and with an increasingly ongoing development. Although the concept is not new, it is a fact that some years ago few people would know that we would be doing payments with our smartphone or use it to measure our health and living habits. Importantly, that this evolution would grow so fast with high user adoption and countless research centers in a wide range of new fields of study.

Regarding the multi-disciplinary aspect of the project I will start to discuss the assessment and taxonomy of Quality of Experience. Here we will cover mainly two ways to assess that: by measuring the emotional state of the user through direct querying, called Participatory Sensing and in an intelligent and autonomous way using sensors in devices, known as Opportunistic Sensing;

and on the other hand by measuring environment conditions that allows the system to understand the context and how it affect the afore mentioned affective state, the so called Context-Awareness.

More technically I will discuss the overview of the Human-Computer Interaction paradigm that focus in bringing usability and interoperability between a common user and physical devices that increasingly surround us everywhere. Following, the Pervasive and Ubiquitous Computing talks about the way computing modules integrate our everyday life in an unobtrusive way or are getting so standard that we do not notice them any more. This leads to the use of Sensors, being them Wearable Computers or embedded in our Smartphones (i.e. Smartphone Sensing), but all of them devices that we carry around all day. Regarding mobile sensors, I will discuss in more detail the current and available Android sensors, and what Physiological data can we obtain from wearable accessories. Finally it is discussed the integration of all this sensorial information in a Wireless Mobile Sensor Network (WMSN) built over a Wireless Personal Area Network (WPAN). Also I will address the Wireless Communication Technologies used to communicate with devices, such as the Bluetooth Smart or BLE.

Finally, given the context of the project I will also discuss the current position of the Public Transportation regarding the enhancement of Quality of Experience and what intelligent infrastructures can be found already that meet our requirements. Also, because this project integrates with the Cloud2Bubble, I will briefly describe this Cloud Computing mechanism.

2.2 Quality of Experience

Quality of Experience (QoE) determines ultimately the satisfaction of a user when emerged and interacting with any service, application or product. This user-product interaction, also referred to as User eXperience (UX), has a beginning and an end and inspires a behavioural and emotional change in the meanwhile [Cos11]. Therefore, we can say that emotion is at the heart of the human experience: it affects the plans to interact with products, how the interaction takes place and the outcomes and results of the said interactions. Thus, it is important to understand which parameters are required to model, reason and infer QoE.

QoE is usually confused with Quality of Service (QoS) or Usability, which tends to focus on task efficiency and effectiveness. In fact, these two are closely related: user perception and satisfaction comes in a multitude of different aspects, each of which may (or may not) be influenced by the performance of individual service components [MEK+09, MZA11]. In other words, QoS affects QoE, but not determines it. The term quality, on the other hand, can be defined as the "result of appraisal of the perceived composition of a unit with respect to its desired composition" [Jek05]. It means it requires a perception and a judgement process to take place inside the user, comparing expectations and perceived outcome of the said interaction. Thus, measurement of QoE usually relies on subjective interaction, whereas QoS can be quantified by a person external to the interaction process, e.g. by the system developer.

QoE is a broad construct that encompasses the above mentioned subjectiveness or user related facts such as his/her behavioural, cognitive and psychological states along with the context in which the products and services are provided to the user. As so, context and environment conditions are influencing factors in the user's satisfaction as well as interaction performance that happens when both communicate (see Figure 2.1).

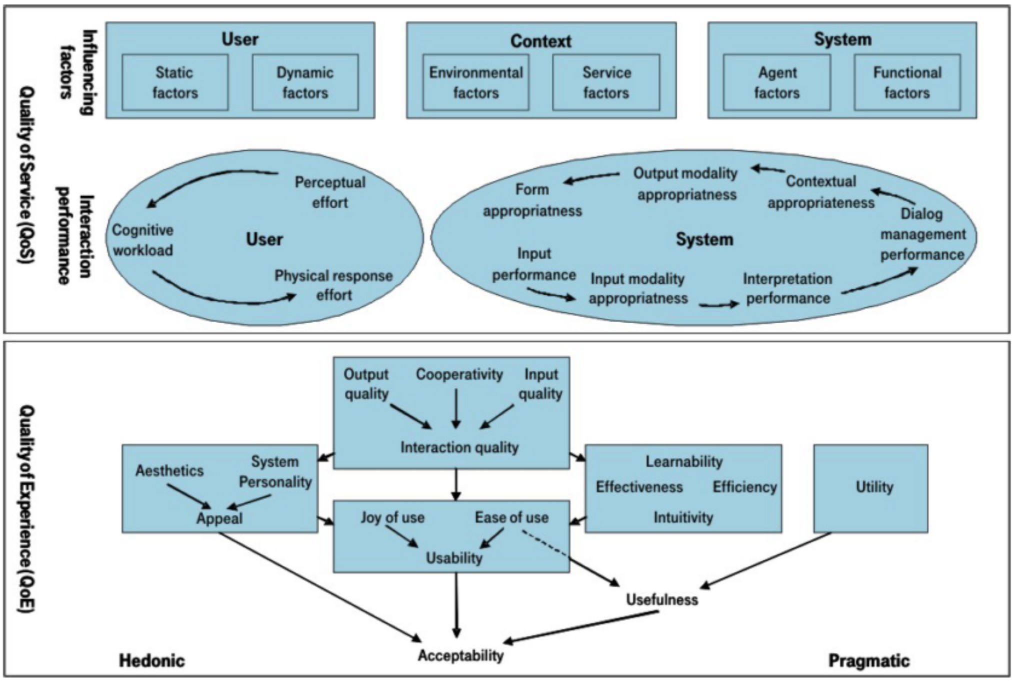


Figure 2.1: Taxonomy of QoS and QoE aspects of multimodal human-machine interaction [MEK+09]

In summary, QoE is a personal representation of what a user thinks and perceives about the service/application/tool. Formally, we define QoE as "a multifaceted phenomenon that depends on a person's likes/dislikes towards a particular object or service where his/her likes/dislikes are defined by his/her personal attributes related to behaviour, cognitive abilities, objects' attributes, the underlying QoS and the environment surrounding that person" [MZA11].

The Public Transportation area gives us a rich multi-modal environment where the context conditions (e.g. luminosity, temperature, etc.) and the interaction with the user (e.g. harsh driving, number of people, human collisions inside the vehicle, etc.) affects users emotion and satisfaction. By measuring and mashing the variety of influencing factors with the users state we are able to assess their QoE individually through pattern recognition and modelling. This concludes that PT provides a good service to apply the proposed QoE enhancement framework.

2.2.1 Environment Conditions

For the context of this project Context-Awareness represents the knowledge of the physical environment that surrounds the user, such as the space, acoustic and lighting conditions. This is particularly important in mobile computing environments where users' behaviour is dynamic since they use applications in different scenarios and social contexts [MZA11].

Unfortunately, Context-Awareness is often still reduced to localization or activity recognition applications, while individual habits and preferences are out of focus. Though, in this work, we give special importance to this vision of context-awareness as surroundings conditions are major influences to the perception of the quality of the experience.

On the other hand, in Public Transportation the environment is constantly changing, even in everyday routines. It is then important to include this aspect in our sensing framework, along with the affective data from the user.

2.3 Public Transportation

Public transport plays an important role in our current society which is characterized by mobility, individuality and comfort constraints. It is common opinion that public transport offers a high level of comfort but lacks individual flexibility and services compared to individual transport [MLE07]. For example, while navigation systems in cars offer a high level of personalization, comfort and a high degree of integration with the car electronics [LMET09], there are no comparable solutions and services available in public transport, e.g. relate air condition or sound volume.

On the other side, people on the move have a grown need for real-time information and services anytime and anywhere. Lately, the number of mobile trip planning solutions, mobile ticketing and validation for smartphone users has grown, but the move to the personal comfort and quality of experience area is far. The latest PT improvements mostly relate to efficiency:

- **Route plannners** – Most applications (web and mobile) focus on the intermodal (i.e. combination of different traffic means) and commuting aspects of the trips. Route planning, in fact, influence the overall user experience by improving the QoS: reducing journey duration and commuting mistakes;

- **Mobile Ticketing** – Buying tickets through the smartphone before getting to the station to avoid queues and waiting lines also enhance the experience;

- **Mobile Validation** – Making the validation electronically tries to transform the smartphone a wallet by grouping services in a single object. The Touch& Travel [Gem08] (see Figure 2.2) is an example.

All these applications are easing the travelling experience and eliminating concerns in changing connections or navigation in large stations in limited time. This is what for some time has made users reluctant to choose public transport [MLE07]. Integrated systems [LMET09] have tried to merge all these techniques based in user localization tracking through GPS, WLAN, or GSM [BDE09], personal habits (e.g. work schedule) and preferences (e.g. buying tickets by the window). The goal of such system is to guide a unique traveller along their complete journey,

Figure 2.2: Touch & Travel Near-Field Communication (NFC) smartphone validation system used in DB Bahn [Gem08]

from the ticketing to commuting, increasing their comfort and reducing their concerns against using public transport. However, although the knowledge of mobility habits and real-time localization is useful for pro-actively provide services, it still lacks the user's preferences on environment conditions.

Beside objective and rational criteria like destination, travel time or price, subjective sensual criteria play also an important role in people selection of traffic means [MLE07]. The comfort of rides has been identified as one of the top criteria that affect customers' satisfaction with public transportation systems, and it has been shown that comfort is an important consideration for passengers that use public transportation [LCCL10]. The Comfort Measuring System (CMS) [LCCL10] tries to define a simple classification of comfort using smartphone sensing and statistics. This, however, is based on a general ideal of the average conditions, and still lacks individuality; different people can like different riding aspects (e.g. harsh movements) or environment conditions (e.g. less crowded). Still, there are some variables that can not be measured given its subjectivity, such as relaxation levels.

In [MLE07] it is defined a wide and concrete list of requirements needed to built ubiquitous transportation applications. Situation detection and Pro-activity were already discussed. Nevertheless, there is another one that deserves some attention: access to dynamic real travel information. It is more reliable than timetables as replanning often occurs due to delays or conditionings. Usually, this information is readily available for trains, subways and many buses and typically is already integrated in most real-time routing solutions.

In sum, there are applications that provide situation detection, pro-activity and real time travel information. As part of the requirements of ubiquitous transportation applications, it is also important to include them in our solution. However, the contents of the information provided to the user are still lacking personalization regarding how the users prefer the journeys.

2.3.1 Smartphone Applications in Urban Public Transportation

Given the context of the project, we need to cover smartphone public transportation applications more in depth. Currently, such applications cover mainly 1) off-line trip planning with static information on timetables and connections, 2) real-time checking of line status and delays, and 3) pointing closest stations/stops through localization systems (e.g. GPS).

The London Tube [Pre] application for iPhone, shown in Figure 2.3, covers all of these topics, creating a seamless user experience for bus and tube travellers. This application allows planning trips from one station to another, giving distance and estimated times for each connection; consulting maps; browsing the current state of every line or station; localizing the closest tube stations and bus stops using the built-in GPS; and finally a pushing notification system that informs users about disrupted lines and delays. All of this is supported by Transport For London (TFL)'s web services and maps. Although, this application brings many common ideas with the prototype we designed, such as push notifications and context-awareness through smartphone sensors, it still lacks individual personalization and introduction of subjective comfort levels in trip-planning. Also, while the notification system here is passive, i.e. it only indicates problems in the lines, we pretend ours to be active, giving (better) alternatives to the user.

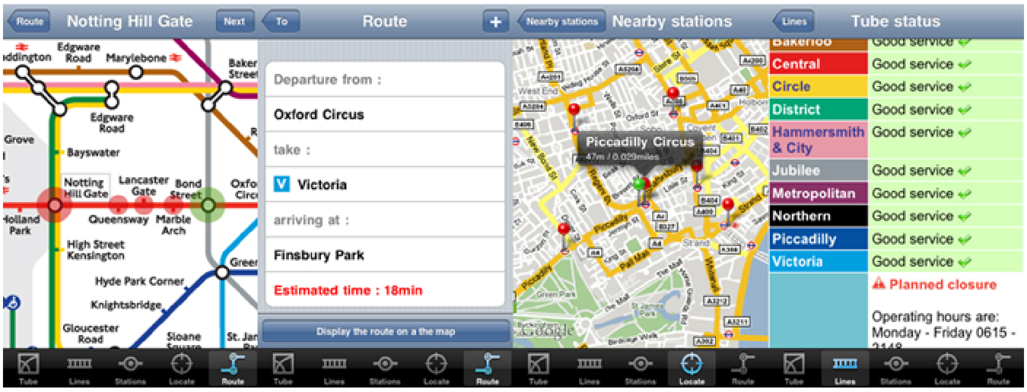


Figure 2.3: London Tube App for iPhone: 1) routing stations on map; 2) trip planning; 3) nearby stations; 4) line status[1]

In Berlin, Germany, VBB Farhinfo [Gmb] presents a good multimodal planning trip with transportation network content from different providers, while also giving real-time information (e.g. destination and time) about next vehicles. Google Maps already provide such features too, as shown in Figure 1.1. Pubtran, Travel+, Travel Deluxe and BusGuru are examples of more route planning solutions, specially for London.

In Porto, Portugal, iMetroPorto [dP] was the first national mobile application for public transportation. However, still behind the previously mentioned solutions. This can simply route users from station to station and access maps and network lines and stations off-line. Much more recently, OPT created MOVE-ME, an application similar to the ones above stated, but the first in

[1]Copyright 2011, Presselite

Portugal of its type. It provides route planning on map, real-time localization with the closest stops information and next services on a specific stop. Although this data is based on timetables and not sensed. Therefore, if a bus is delayed there is no mean of knowing it.

The number of similar applications are endless. After trying some of them, we can point that the routing mechanisms, for as intelligent as it can be, use off-line information. It does not include real-time conditions or delays, for example. This is a very important gap that we try to cover on our proposed solution, with special focus on the environment conditions.

2.4 Human Computer Interaction

Human Computer Interaction is a very broad field of research concerned with the design, evaluation and implementation of interactive systems for human usage. Since the early years of Computer Science developers care about the way humans interact with physical objects provided with computational resources. Despite the technological evolution of such systems and devices, areas such as usability, understandability and learnability of interactive systems still occupy HCI's main attention and are the major criteria for enhancing user experience.

As in the name, interaction between both parts is specially important; neither it intends to be unidirectional, in the form of a monologue between the parts, but rather a conversational experience [Cos11]. Only this way is possible to create seamless and fluent applications that eases the access to computation in one way and user data in the other. The origin and development of Graphical User Interfaces (GUI), for example, which serve as the interface between both parts rose this area to another levels. While this is now found in most of the solutions, a new tendency of making this interaction even more fluent and invisible is making its appearance. Computing is moving from the big screen of personal computers to a multiplicity of devices, allowing the interaction to take place everywhere at any time. Ubiquitous Computing was born under this concept. This new branch of HCI will be discussed in section 2.6.

Although the concerns addressed by HCI have evolved to a much wider spectrum, the research in the area was originally guided by the following criteria [HBC+96]

- the performance of tasks by humans and machines;
- human capabilities to use machines (e.g. learnability of interfaces);
- the structure of communication between human and machine;
- algorithms and programming of the interface itself;
- engineering concerns that arise in designing and building interfaces;
- the process of specification, design, and implementation of interfaces;
- design trade-offs.

Although the visible computer side point of view in the list above, lately, the user has been grabbing more attention as the number of user-centered applications grows. Not only desktop applications followed this direction but the introduction of mobility to computing devices (or computing to mobile devices) and its mass adoption largely influenced this perspective. Mobile Computing, specially in smartphones and tablets, then opens new visions and future to HCI. However, trends in HCI development are shaped by a variety of factors, ranging from the miniaturisation of hardware components, to the specialization of applications originating new interacting paradigms [HBC+96]. Current and future developments are, thus, aligned with a set of characteristics that govern the field of study [Cos11]. Concerning the direction of this project, some of them are:

- **Ubiquitous Communication** – communication made available ubiquitously, accessible virtually in every location where the user chooses to travel;
- **High Functionality Systems** – increasing number of functions available, challenging the traditional way of learning;
- **Embedded Computation** – availability of computational capabilities increasingly integrated in everyday's appliances;
- **User Tailorability** – applications tailored to personal use, extended to the development of applications to different user's domains;
- **Information Utilities** – proliferation of general and specialised services in the public and private domains.

In the following sections it will be discussed in more depth the objective areas of HCI that cover the context and topic of this project.

2.5 Affective Computing

As defined in [Gra10], Affective Computing is the field of study concerned with understanding, recognizing, and utilizing human emotions and other affective phenomena in the design of technological systems. Research in the area is motivated by the fact that emotion pervades human life emotions, motivate and shape our individual thoughts and social behaviour. Regarding the main goal of this project of improving user experience it is then essential to assess, recognize and understand user's emotions when interacting with the public transportation service.

Emotion, as stated in section 2.2, is fundamental to the human experience, influencing cognition, perception, and everyday tasks such as learning, communication, and even rational decision making [Cos11]. It was not until recently that its role was valued in interactions within the computer science field, when Affective Computing arose. Up until then it had been largely ignored as an important component in the human experience, which resulted in a lack of a broader understanding in this area, often leading to frustrating experiences for people [Cos11].

Emotion has been studied as a science at least since Aristotle and an enormous body of theoretical and empirical work exists across a wide range of disciplines, including neuroscience, ethology, psychology, anthropology, sociology, economics, art, and literature. Computer scientists and engineers are newcomers to the game, and bring new perspectives and new tools to the challenge of recognizing, understanding, and shaping human emotions [Gra10]. Affective Computing has then emerged, since the coining of the term by Picard in 1995, as a cohesive interdisciplinary field.

Affective computing has seen its biggest adoption in the growing number of applications in healthcare, motivated by over a decade of findings from neuroscience, psychology, cognitive science, and the arts about how emotion influences human health, decision making, and behaviour. The process might involve sensors gathering users' physiological data or behaviour, leading to the recognition of emotional information and requiring the extraction of meaningful emotional patterns. The techniques used range from speech and facial recognition to physiological data such as skin temperature and electrodermal activity.

Despite this area being still at an early stage, there are already some serious mature discoveries on how to assess emotion, such as the referred physiological data. However, until now, this kind of technology has been used primarily in healthcare applications which don't really care about usability and mass adoption, but rather in continuous and precise collection of sensorial data from patients, resulting in bulk and intrusive devices [Emo] and do not support customization or multiple sensor nodes. The commercial solutions also add proprietary software and protocols, making them impractical for widespread use in affective computing and medical research. However, new advances are trying to spread this technology in controlling ordinary people's lifestyle [FDG+10, LML+11, CCH+08]. Their main concern lies in the usability, trying to pack the same sensing physiological technology in small and lightweight wearable systems that can be used comfortably on a daily basis. This allows these devices to enter a new market of urban usage, allowing to understand emotion's in a wide range of experiences.

2.5.1 Physiological Data and Wearable computing

Emotions can be expressed via several channels and various features can be analysed to assess the emotional state of a participant. Most studies focus on the analysis of facial expressions or of speech [CDCT+01]. These types of signals can however (more or less) easily be faked; in order to have more reliable emotion assessments, it is preferred to use spontaneous and less controllable reactions as provided by physiological signals. Physiological signals can be divided into two categories: those originating from the peripheral nervous system (e.g. heart rate-HR, electrodermal activity-EDA, galvanic skin response-GSR), and those coming from the central nervous system ElectroEncephalograms-EEG). From such data there are models that classify emotions [CKGP06]. This topic however goes beyond the context of this project, since the classification of emotions given the design of the solution, is to be done in the C2B.

The relative big size of some of these physiological solutions and its operability, specially EEG (see Figure 2.4), makes is impossible to use in unobtrusive environments. As so, widespread use of affective sensing has been limited due to several practical factors, such as lack of comfortable wearable sensors, lack of wireless standards, and lack of low-power affordable hardware. The comfort is specially relevant to urban sensing. The ability to measure changes in the referred physiological levels comfortably, and continuously, without injecting cumbersome wires or boxes into people's activities, has the potential to revolutionize health therapies, fitness monitoring and cognitive assistance services.

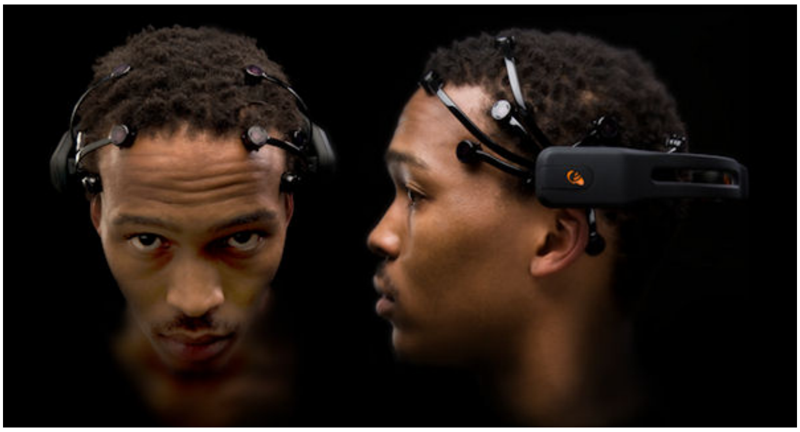


Figure 2.4: Emotiv EPOC introduces emotion-based HCI[2]

The technological advances in this new area of Wearable Computing is starting to shows signs of maturity. New devices being commercialized care about ergonomics and visual appearance, allowing users to carry them continuously. A big advantage for Affective Computing [CCH+08].

Figure 2.5: Basis B1 watch allies technology with fashion[3]

The iCalm wearable sensor [FDG+10], developed in the MIT was one of the first systems concerned with the usability of a physiological sensor device. Moreover, it already connects to the users mobile phone to transmit data through wireless communication. Still the design for user adoption is still far from aesthetically appealing. On the other side, the Basis B1 watch [Sci] (see Figure 2.5) incorporates several sensorial information, such as HR, GSR, temperature and 3D

[2]Copyright 2011, Emotiv
[3]Copyright 2011, BASIS Science

accelerometer into a good-looking and completely ordinary wearable accessory. Thus, making it a useful yet unobtrusive solution for measuring the affective state. The way the technology moves with the design of appealing devices is what drives user adoption. This product is, however, yet to be commercialized. It is under development and from the latest official announcements, the company said they would be integrating Bluetooth wireless communication into the device. More importantly, making the Application Programming Interface (API) open for use by any application. This is one development we are eagerly looking for, since it provides all the features we envision in our design.

The Wahoo Fitness Blue HR [Fit] (see Figure 2.6) already introduces the new wireless communication technologies powering low power and smartphone integration, transmitting real-time sensed data between the devices. It measures HR and also has offers and open API. However, the chest strap style, although it is enough for testing in this project, it fails in the user adoption aspect.

Other solutions tried to incorporate physiological assessment in earplugs to read blood pressure [PKG+09] and electroencephalography (EEG) headsets to control mobile phones for hands-free human-mobile phone interaction [Emo, CCH+10]. The later, however, are way too intrusive for ordinary users and daily living which cannot meet public transportation commuters needs.

Figure 2.6: Wahoo Blue HR sends physiological data to the smartphone wirelessly[4]

From this research in the fields of Affective and Wearable Computing, we can conclude that some devices provide already some testing possibilities. The Wahoo solution provides HR data, Bluetooth Smart communication and open API, though the shape is not correct. The Basis B1, when finished will provide the same features, plus GSR data and thermometer, into a completely ordinary watch.

[4]Copyright 2011, Wahoo Fitness

2.6 Pervasive and Ubiquitous Computing

Over the past few decades, computers have become increasingly inherent to people's daily lives. Business people, researchers and computer aficionados are no longer the sole users of computers, but people everywhere now use the Internet. This migration is due to the ever more affordable computers and internet access everywhere and in consequence it creates a massive database of all types of information technologies, applications, media and data accessible by everyone. Nowadays, this defines is a new paradigm of computing. As addressed in a survey in Ubiquitous Computing [Wes11], this new area will enable a world where the monumental computer on a desk ceases to exist as a central link to information, but perhaps a wallet would hold all of a user's personal information as well providing access to the internet.

Under the field of HCI, Ubiquitous or Pervasive Computing brings the process of information into everyday objects and activities, leveraging the computing environment to a level where a device acts as a portal into an application-data space [Cos11]. Integrating these intrusive yet invisible systems into the human environment allows the creation of intelligent applications that come to, more often than not, help living. Computation will move into every area of our lives. Instantaneous information and computation will be distributed over an array of small wireless networked devices connected to the internet. These can be embedded in daily artefacts such as appliances, light switches, stereos, cellular phones, and watches. This capability will revolutionize computation, allowing it to take place anywhere and at any time. Rather than accessing data only via a monitor and keyboard, one might access data via voice-activated commands and view it on a neighbouring wall. Computation will be everywhere [Wes11] – in "things" (see Figure 2.7).

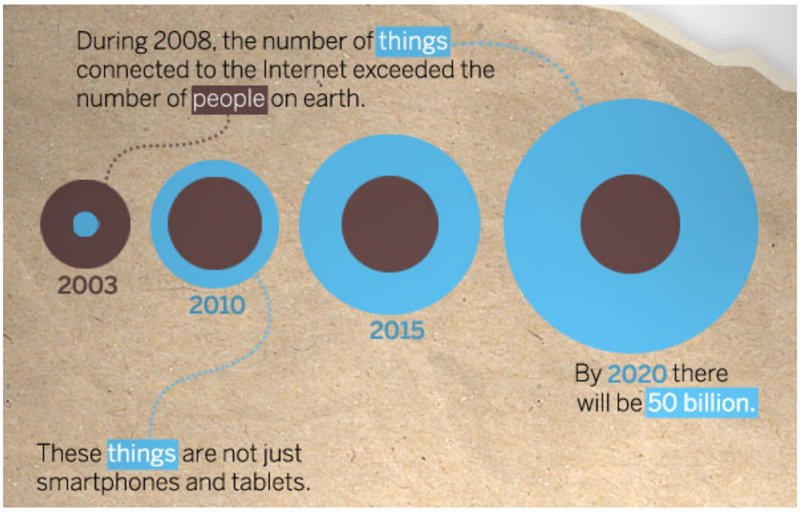


Figure 2.7: Part of Cisco's infographic on the Internet Of Things (IoT)[5]

The goal of Ubiquitous computing as envisioned by its creator, Mark Weiser, is to recess computers into the background of life, much like electricity has been. We no longer think about

[5]Copyright 2011, Cisco Systems

using electricity, we just do. He states a similar goal for UC: “Its highest ideal is to make a computer so embedded, so fitting, so natural, that we use it without thinking about it”. His initial research led to creating a new field of computer science that speculates on a physical world filled and invisibly interwoven with sensors, activators, displays, and computational elements embedded seamlessly in the everyday objects of our lives [WGB99].

What Weiser envisioned 10 years ago, is happening now. For example, a user may have a refrigerator that tells them when items inside expire. Imagine that user then walking into a grocery store and having all the sale items that match their shopping list display the price and location of each deal on the shopping cart’s display. Also you could turn on your microwave from work so that you could have a hot meal when you walked in the door. A car could be able to signal to your garage door to open when it was in your driveway [Wes11]. The possibilities are endless.

The wireless network of sensors, discussed ahead in the section 2.7.2, became more solid and widely available with the growth of the processing and sensing capabilities of current mobile devices, also known as Smartphones. Devices started to communicate with each others creating an ubiquitous network of smart processing. Thus, for this interaction to be effective and invisible it needs to be done wirelessly. This intelligent intercommunication between all kinds of objects or “things” gave rise to the new and so discussed concept of Internet of Things (IoT). Complementing the UC paradigm, the IoT is more concerned in making physical objects communicate than to interact with the user. Both, however, have the common goal of taking some of the burden in daily human activities. Even Google entered the game with the Google Wallet [Gooc] and Near Field Communication (NFC), allowing one to move all of the information contained in his wallet to his Smartphone and, for example, paying with it.

The use of such framework and sensor network in the context of Public Transportation, or even in any other area, enables the collection of affecive data, environment variables and transmitting them to the user’s smartphone wirelessly. Is this integration of technologies and building a solid sensing framework that is missing nowadays and will soon start to make its appearance on the market.

2.7 Mobile Sensing

To introduce the concepts of Smartphone Sensing and Wireless Mobile Sensor Networks it is necessary to firstly introduce and define Mobile Sensing and Sensor Networks.

Sensors are the common name given to the devices that measure a defined parameter (e.g. a balance measure weight, a thermometer temperature). The use of such sensors can report as back as the first mechanical devices arose and nowadays they are everywhere. They began to integrate our lives specially by reducing the size of such devices and incorporating them in ordinary objects. For this reason, networks are becoming ubiquitous, driven in particular by advances in wireless technologies and capabilities of sensors [MGT10]. Sensor Networks are then composed of geographically dispersed sensors that work together to monitor physical or environmental conditions, such as air pressure, temperature, or pollution. The application of Sensor networks is wide, including industrial process monitoring and control, environment and habitat monitoring, healthcare, home automation, and traffic control [TWDS11].

Wireless connectivity between sensors and access to the Internet brought another dimension to Mobile Sensing. Tracking parcels (see Figure 2.8), for example, can now be made through RFID tags that are autonomously read and tracked through web services [Sch10]; in the medical context we can collect multi-modal physiological information from multiple sensors distributed along the body [RACB11]. As so, collection, communication and aggregation of data from various sensors on the spot increasingly helps to assess multi-modal data either from the context and the user. For its importance and use in this project, Wireless Mobile Sensing Networks (WMSN) will be discussed in section 2.7.2.

Figure 2.8: Dash7-based RFID tags have been used in military's parcel tracking [Sch10]

Recently, smartphones have also attracted the attention of networking and ubiquitous computing research communities due to their potential for acting as mobile sensor nodes for city-wide sensing applications. This opened doors to the wide-scale adoption of such services, through their growing number and precision of embedded sensors. However, it is not only a sensing platform, but also the aggregation node of such information (as shown in Figure 2.6). The smartphone uses sensor data in the creation of appealing visual interfaces, logging, inquiry of over-time data, access to web services and other purposes that will be largely addressed in section 2.7.1.

In order to develop usable applications that reach mass adoption it is also needed to address some concerns as in the Mobile Sensing Platform system [CCH+08]

- package multi-modal sensors into single small devices;
- either integrates into a mobile device, such as a cell phone, or wirelessly transmits data to an external device;
- and for small devices to be able to function on a usable level, they also need to consume little power [Wes11].

2.7.1 Smartphone Sensing

Mobile phones have been adopted faster than any other technology in human history. As of 2011, the number of mobile phones subscribers exceeds 5.6 billion. Today's smartphones are rapidly becoming the central computer and communication device in people's lives. We carry them around the clock and save inside a handful of information, which gives space for people-centric computing, be it either personal, group or community-scale sensing applications.

Up until recently mobile sensing research such as activity recognition and real-time localization required specialized mobile devices [CCH+08] and GPS systems, respectively. Software was intellectual property to the manufacturers and applications had to be downloaded manually. However, all of this is changing due to important technical advances. [LML+10] points four key points for the world-wide spreading of sensing applications:

- the availability of cheap embedded sensors initially included in phones to drive the user experience (e.g. the accelerometer's initial purpose was to change the display orientation) is allowing plenty of other possible sensing applications (e.g. using the accelerometer to play games);

- smartphones are now open and programmable. In addition to sensing, phones come with computing and communication resources that offer a low barrier of entry for third-party programmers;

- each phone vendor now offers an app store allowing developers to deliver new applications to large populations of users across the globe. This is transforming the deployment of new applications and, importantly, allowing the collection and of data far beyond the scale of what was previously possible;

- the mobile computing cloud enables developers to offload mobile services to back-end servers, providing unprecedented scale and additional resources for computing on collections of large-scale sensor data.

In the last version of the Google Android platform - 4.0 Ice Cream Sandwich (ICS) - we can find support for a handful of sensors such as, accelerometer, digital compass, gyroscope, GPS, microphone, camera, barometer, temperature and relative humidity [Gooa]. These sensors are not only useful in driving the user interface and providing location-based services, but they also represent a significant opportunity to gather data about people and their environments. For example, accelerometer data is capable of characterizing the physical movements of the user carrying the phone [MLF+08], while in combination with location-based sensors (GPS, Wi-Fi, radio frequency) can characterize the way people and communities interact with the locations they inhabit [MPL+11]. More ubiquitous applications use plenty of sensorial data to acknowledge the context of the user, specially in indoors spaces [ACRC09]. However, the smartphone gathers

Figure 2.9: Current smartphones power rich sensors and useful data

even more daily useful data, as seen in Figure 2.9, such as synchronized calendar data, and communications and connectivity records. GINGER.io [Gin], an healthcare application, uses such information to find social behaviour anomalies, e.g. reduced number of calls.

In sum, sensor-based smartphone applications are growing and being used in a wide variety of domains, such as healthcare [CMT+08], social networks [MLF+08, MPL+11, Facebook], environmental monitoring [MRS+09], context-awareness [ACRC09, MLF+08] and transportation [TWDS11, LCCL10], and give rise to a new area of research called Smartphone Sensing. The research in this field is done in universities research groups or labs (e.g. the Dartmouht's Smartphone Sensing Group [Uni]) but more often in the commercial world, making it hard to track design details and technology developments.

Building smartphone applications for different contexts means different architectures and sensing paradigms. However, [LML+10] presents a generic sensing architecture, shown in Figure 2.10), that comprises three building blocks:

- **Sense** – where the mobile phone collect raw sensor data from its embedded sensors; In this subject, a new concept of continuous sensing is emerging. It refers to the almost permanent collection of sensed data, but also brings concerns on processing capabilities of the smartphone and the amount of raw data transmitted to the cloud. To address this issue, a solution is to reduce this transmission to short summaries after feature extraction;

- **Learn** – in this stage, information is extracted from the sensor data by applying machine learning and data mining techniques. These operations occur either directly on the phone, in the mobile cloud, or with some partitioning between the phone and cloud. Classification and extraction of features is subjective and application-dependant, but there are already some models to classify activity and comfort levels from accelerometer movements [ACRC09, MLF+08, LCCL10], fingerprinting sound based on intervals of sound amplitude from the microphone [ACRC09, MLF+08] (see Figure 2.11), and light and color classification from the phone's camera pictures and orientation [ACRC09].

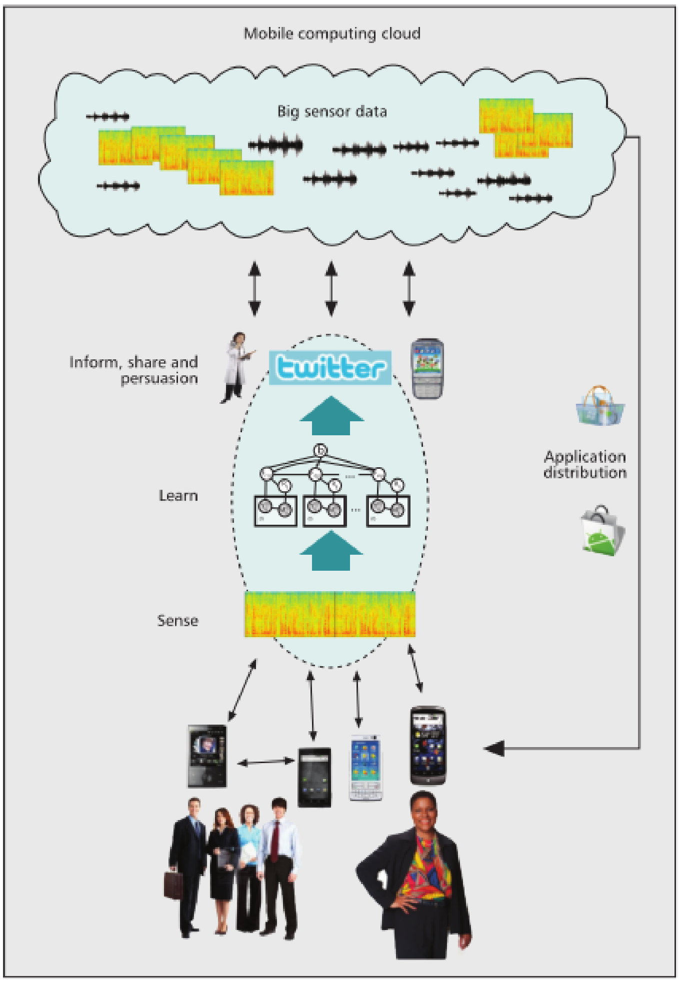


Figure 2.10: A mobile phone sensing architecture [LML+10]

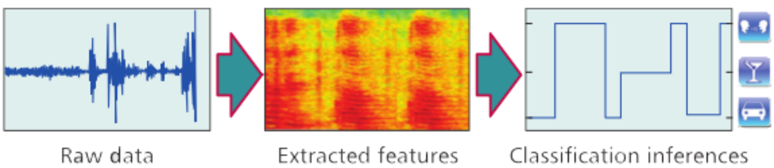


Figure 2.11: Raw audio data captured from mobile phones is transformed into features allowing classification through learning algorithms [ACRC09]

Labelling and matching with other sensorial information is done through machine learning algorithms and techniques, such as supervised and semi-supervised [CCH+08, ACRC09, LML+10]. However, this computation phase, along with emotion assessment, overpasses the border of this project since it is done in the cloud;

- **Inform, Share and Persuasion** – the last step is where most applications might diverge. For example, a Personal sensing application will only inform the user, whereas a group or community sensing application may share the information with the broader population. Here, privacy is a very important consideration as well.

Personal sensing applications are designed for a single individual, and are often focused on data collection and analysis. Using this personal sensing technology, applications are able to provide users with more informed decisions across a spectrum of services. Typical scenarios include tracking a person's exercise patterns and return feedback on weight loss. This example shows that personal sensing applications generate data for the sole consumption of the user and are not shared with others [CMT+08]. An exception is healthcare applications where limited sharing with medical professionals is common [LML+10].

At the same time researchers are discussing how much the user should be actively involved during the sensing activity (e.g. taking the phone out of the pocket to collect a sound sample or take a picture). That is, should the user actively participate and directly insert data, known as Participatory Sensing, or, alternatively, passively participate letting the application collect his own data for public or semi-public consumption, known as Opportunistic Sensing. For affective and context awareness applications to be effectively user-friendly and objectively understand patterns of behaviour and likings, it is necessary that the system receive feedback from the user whenever the user can or wants to give it [Wes11]. Given the emotional aspect of this work and the rather unpredictable and data-noisy environments in public transportation this issue is very important and will be addressed shortly in section 2.7.1

Developing sensing applications that does not violate usability and human-computer interaction principles is not an easy task, though. The survey on Mobile Phone Sensing [LML+10] adds serious concerns in developing such applications regarding personal privacy, processing capabilities and large amount of shared data without jeopardizing user experience, battery issues and resource-sensitive reasoning with noisy data. When these barriers are overcome the world of mobile sensing can advance quickly in the areas of healthcare, social networking, transportation and energy. To speed up this process, it is important that new smartphone applications address these issues, as it planned for this project.

Sensing Paradigms

One common issue to the different types of sensing applications is to what extent the user is actively involved in the sensing system. For that we can define two sensing paradigms: participatory sensing, where the user actively engages in the data collection activity (i.e. the user manually determines how, when, what, and where to sample) and opportunistic sensing, where the data collection stage is fully automated with no user involvement. However, it is likely that many applications will emerge that represent a hybrid of both these sensing paradigms.

The benefit of opportunistic sensing is that it lowers the burden placed on the user, allowing overall participation by a population of users to remain high even if the application is not that personally appealing. However, often these systems are technically difficult to build. On the other side, when such difficulties are overcome, intelligent urban systems based in opportunistic sensor networks [CEL+06] have the ability to scale to very large metropolitan areas. This paradigm is specially effective when used in people or community-based Wireless Mobile Sensor Networks communication (see section 2.7.2) to aggregate multi-modal data without needing user input.

Participatory sensing, which is gaining interest in the mobile phone sensing community [LML+10], but also adds a bigger effort to the user. For example, manually inserting service satisfaction (e.g. Starbucks satisfaction poll) or improving data collecting through sampling (e.g. taking a picture). The research application Mappiness [MM] (see Figure 2.12), for instance, uses this paradigm to assess UK's level of happiness. This application randomly prompts the user asking how they feel, their company and adding the opportunity to add camera pictures to this feedback. An advantage of Participatory Sensing is that complex operations can be supported by delegating the intelligence to the user who can solve the problem more efficiently. One drawback of participatory sensing is that the quality of data is dependent on participant enthusiasm to collect sensing data. Another found disadvantage is the intrusive aspect when asking the same questions over and over again about thoughts and emotions, because some might see it as useful others might see it as burdensome.

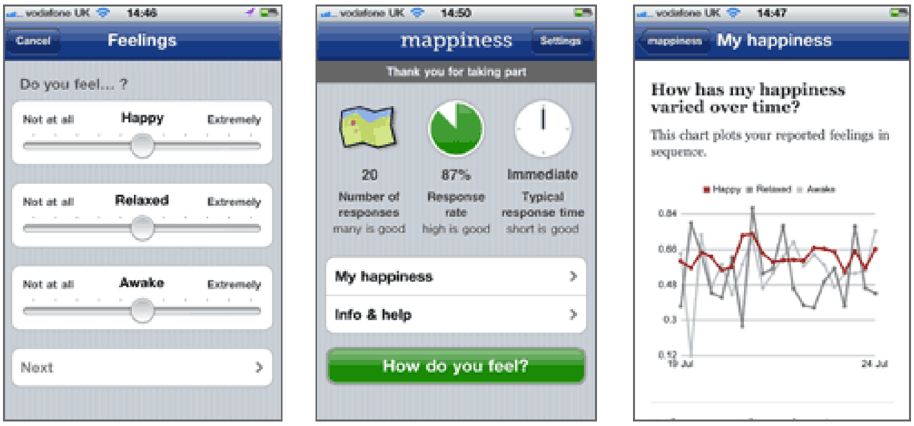


Figure 2.12: Mappiness maps happiness and emotions across UK from participatory sensing[6]

Both views have advantages and to our application the more correct approach is to combine them. The autonomous sensing of affective data (either from wearable computers or smartphone's sensors) cannot entirely and objectively assess the satisfaction level of the user, e.g. sensors cannot measure crowded places, they can feel a changes in the user's biosignals but do not understand why. For that, it is important the minimum input from the user, to add more information to the reasoning system or simply as validation of the opportunistically collected data.

2.7.2 Wireless Mobile Sensor Networks

In short, Wireless Mobile Sensor Networks are sensor networks as described in the beginning of this section that communicate wirelessly. They are a set of distributed wireless independent

[6] Copyright 2011, LSE mappiness.org.uk

autonomous sensors and microcontrollers that report physical data such as temperature, sound, vibration, or pressure. Typically, these networks are built over a wireless mesh topology to transmit the data being collected [Wes11].

Opposite to normal static sensor networks, e.g. smart houses, the dynamic of mobile sensors and the alleged accessibility to services everywhere in a pervasive environment, as Weiser envisioned, makes the wireless infrastructure a powerful yet under development area. Over the past few decades many wireless communication technologies arose, starting from connecting laptops to routers - IEEE 802.11's Wi-Fi - to connecting mobiles to hands-free kits - Bluetooth. Nowadays the growth of wirelessly connected objects is almost exponential, as shown in Figure 2.13, mostly cause by the aim of application development at ubiquitous solutions. However, it was in the last few years that some concerns on technical details, such as frequency, bandwidth/spectrum usage, maximum connectivity distance, power and antenna's size led to the origin of new low power low bandwidth wireless standards. Technologies like ZigBee, DASH7, ANT and Bluetooth Smart (or Bluetooth Low-Energy, BLE). Their specifications vary which makes them recommended or preferred in different contexts while competing in others.

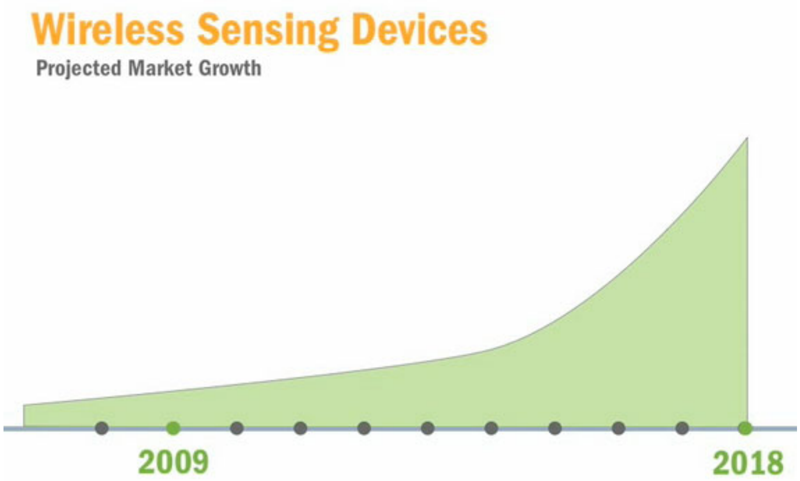


Figure 2.13: The projected number of wireless sensing devices in the future

In terms of multi-modal Personal Sensing applications, specially those that measure movements, mobility patterns and physiological signals, WMSN are becoming extremely important in building networks of multiple body sensors, known as Body Area Networks [FDG+10]. This concept helped the IEEE 802.11 protocol - Wireless LAN - evolve to the IEEE 802.15, designed for Wireless Personal Area Networks (WPAN's) [Hei09]. As the name suggests WPAN's are networks for interconnecting devices around an individual's surroundings - in which the connections are wireless. The key concept in WPAN technology is known as plugging in. Ideally, selected devices when in close proximity are able to connect with each other as if they had cables connecting them. This concept truly fits a design for our solution where multiple sensors around the traveller collect and communicate different information to a central computing node, lets say his smartphone.

The multitude and characteristics of devices and interactions as described in [CEL+06] is enormous. We can have static sensors communicating with mobile sensors, static sensors communicating with gateways as well as mobile sensors and above that gateway communication to access sensed data from different networks. At a higher level, we can combine these interactions to create a massive wireless sensor network. As shown in Figure 2.14 this network can be divided in:

- Environment Sensor Networks;
- Body Sensor Networks;
- Object Sensor Networks;
- other Wireless Sensor and Actuator Networks (WS& AN).

When all these parts communicate and share information through web services, it will create an such intelligent system that knows where and how the user is and all sorts of real-time information. This concept, together with the IoT, define, in the scientific community, the Future Internet. As a research work, we can design a prototype solution following these innovative steps and leveraging the sensor networks power. In fact, all the previous research fits into this concept seamlessly providing an insightful direction for our solution.

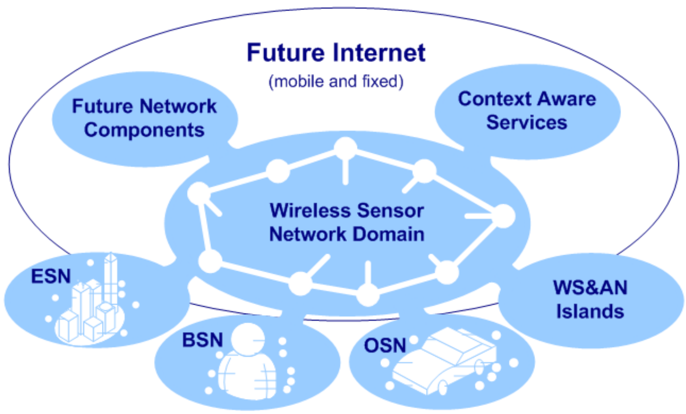


Figure 2.14: Overview of the Future Internet based on a massive wireless sensor network

Mobile Phone in WMSN

Nowaday's mobile phones or smartphones are, as discussed in section 2.7.1, by themselves a computing and sensing device that is already inherent to living habits. As so, they can be easily integrated as a node in the afore mentioned WSN's, building distributed applications that take advantage of its processing capabilities and internet connection.

Due to widespread use of smartphones, it is now possible to develop large-scale sensor networks using their technology and deploy applications on end-user devices. These applications gather sensor data and report readings back to servers. Therefore, they act not only as a sensing node, but also as a blackboard sensor knowledge base, helping the processing and sharing of such data. [TWDS11] shows how smartphones can help reduce the development, operation, and maintenance costs of small wireless sensor networks, while also enabling these networks to use web services, high-level programming APIs, and increased computing capability.

By combining web services and the advanced computational power of smartphones, applications can contain real-time information filtered from individual users, such as location, social relations, or preferences. Data from multiple users can be combined and with web services data to create powerful applications reporting real-time, location-aware content [TWDS11]. Therefore, allowing the expansion to a large group or community sensing applications, specially in the social context: mapping crowds and activities [CEL+06]. People like to know what is hot, where the people is, what is happening and what is not, in real-time. In Public Transportation, however, there is no actual use of such networks and this is where we can fit our solution. With a sensor network in PT, it would be easier to know live service conditions.

2.8 Related Work: Cloud2Bubble

This work follows and tries to integrate with Pedro Maurício Costa's previous work [CPCG12] and prototype: the Cloud2Bubble cloud-computing module. This infrastructure (see Figure 2.15) is capable of aggregating a number of data sources from both user and environment and generating user-tailored services with the potential of enhancing QoE. The Cloud2Bubble framework facilitates the development of system design and development in smart environments and it addresses the disconnection between user-generated data collection and personalised service delivery.

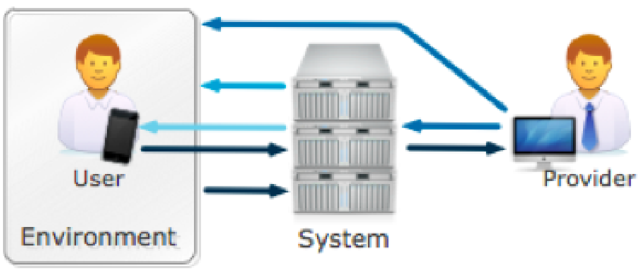


Figure 2.15: Interactions in the Cloud2Bubble infrastructure [Cos11]

2.8.1 Interaction Loop

The interaction process that takes place between user and system forms a continuous loop of interaction composed of user-data collection and service delivery, supported by the collection of environment state. The stream of environment data is transformed into a representation of the environment state. This environment state is then aggregated together with the stream of user data, resulting in a unique user profile. This profile describes the environment conditions that correspond to positive and negative user responses. Such a profile is continuously improved over time, leading to the identification of increasingly accurate patterns of satisfaction and/or dissatisfaction. Individual profiles enable the delivery of user-tailored services, i.e. recommendations of suitable alternatives or notification warnings.

2.8.2 World Modelling

The world modelling is based on two main concepts: groups of co-located environment sensors, defined as sensor Cloudlets, and personal mobile devices capable of interacting with users, defined as virtual Bubbles. Cloudlets and Bubbles are organised in a hierarchical structure, reflecting their physical relationship, which is then used for environment state representation by the framework. Moreover, such a structure facilitates a systemic integration of existing technology and infrastructures already present in the environment, as well as scalability for future additions.

2.8.3 Architecture

This framework is also currently under development. It combines a set of modules that facilitate the development of smart systems and enable a QoE-oriented approach:

- **Domain Management** – enables the dynamic modelling of an environment, including the relationship between different components, ranging from personal smartphones to environment-deployed sensors and third-party services;

- **Rule Engine** – implements an adaptive event-driven solution capable of processing a large number of changes in the environment and adapting itself to the requirements posed by different environment states;

- **Reasoner Module** – it aggregates different data sources into individual user profiles, which describe personal preferences. Such profiles are then used to take action in the environment, providing users with relevant information.

2.9 Conclusions

From this chapter we have seen that a new era of user-centered applications is on the rise. The HCI paradigm is moving towards mobility and mobile services are slowly but steadily becoming a part of people's everyday live. Specially caused by the need for services that make their lives easier by blending smoothly with the situations they find themselves in every day.

Quality of Service (QoS) was until now the main criteria to evaluate user experience, but a new attention to Quality of Experience (QoE) focused on the perceived quality by the user is serving as a supporting guideline for new applications. In Public Transportation, trip planning applications supported by advanced routing algorithm based in real-time dynamic information combined with e-ticketing and mobile validation are providing more comfortable rides. However, individualization regarding commuters preferences and personal satisfaction is still missing. It has been shown that beside objective and rational criteria like trip duration, delays, cost and number of changes, subjective sensual criteria also plays an important role in people's selection of traffic means. The comfort of rides, largely affected by the customers' affective state, has been identified as one of the top criteria that affect satisfaction with public transportation systems. Assessing and understanding emotions will then open doors to personal enhancement of QoE. It is in this context that the Affective Computing field of study gives already some innovative insights and solutions. Wearable physiological sensors are important improvements to measure changes in the emotional state of the user, but are rather new, complex or, more important, too intrusive as of now. Some commercial solutions are, however, in pursuit of urban mass adoption by combining technology and style allowing the collection of such data to take place everywhere.

Finding patterns of satisfaction is complex. Changes in affective state are only substantial when mashed against context information such as surrounding conditions. This way it is possible to understand the influencing factors in the user experience, enabling the creation of personal profiles based on user satisfaction or QoE. Lately, context-aware applications are becoming popular in terms of social connectivity, indoors localization and others, with its main resource being the personal smartphone, but not in PT.

Today's smartphones are rapidly becoming the central computer and communication device in people's lives, but it also powers a rich sensorial platform that as been used in a wide variety of domains, such as healthcare, social networks, environmental monitoring, context-awareness and transportation. Smartphone adoption extended the area of Ubiquitous Computing allowing computation to happen unobtrusively and interacting with other systems and devices. For developers it opened new doors given its programmable aspect, the rich embedded sensors, the easy to deploy mobile applications markets and its internet connectivity acting as a portal to the Cloud Computing world. Therefore giving processing capabilities and real-time information to mobile applications as never seen before.

Mobile Sensing allows the combination of multi-modal data collected from a multiplicity of sensors. By carrying/affixing one or more personal sensors people can monitor certain aspects of their own body. Sensors embedded in the home or public infrastructure can provide information about the environment surrounding people. Data collected from both personal body sensors and environmental sensors can be correlated to provide a trace of context-rich personal health information. Ultimately allowing real-time monitoring or mid-term analysis for a better delivery of service to the user. Communication infrastructures like new low-power low-bandwidth wireless technology standards supports a new concept of ubiquitous systems called Wireless Mobile Sensor Networks. In WMSN mobile sensing is brought to the level of invisibility, making the user experience and HCI transparent and more fluent, specially by integrating everyday objects such as the smartphone. It can interoperate with static and mobile sensors wirelessly and communicate collected data to the cloud. In the future, computing will be so integrated in our lives that we won't notice it, but it will growingly affect our daily experiences.

Ubiquitous computing as defined by Mark Weiser will not exist for a few decades. There are too many issues that need to be resolved before computation can exist everywhere. Currently technology can handle a simplified ubiquitous environment in a controlled setting, but does not have the infrastructure or mass appeal needed to disperse inexpensive network-enabled computers.

As seen in the chapter, a solution to fulfil the Cloud2Bubble framework currently being developed does not exist. Instead we found a collection of isolated solutions for almost every aspect of the proposed system. Thus, by connecting the current state of the art in these areas we will be able to define a sensing platform and build a proof of concept for enhancing QoE in public transportation. The collection of user affective and context data is the point where this work integrates the whole project.

Chapter 3

Designing the Solution

3.1 Introduction

Following the Rapid Application Development method proposed for the project development, the first goal is to gather all the knowledge acquired during research and build a solid solution to accomplish the objectives. Since the main goal is a proof of concept prototype, the design phase takes a bigger and longer stage in the development and is important to understand the concept and architecture behind the smartphone implementation. Specially, since the pursued approach drove to a cutting edge solution, using innovative models, tools, paradigms and architectural techniques on top of a user-centered design.

In this chapter, it is presented the proposed methodology for collecting all the context and emotional information we need for assessing QoE. More, understanding the choice of this data and how it is useful for our solution. For that, it will be introduced a high abstraction level representing the flow of throughout the interaction loop. In other words, it describes the method used to create a knowledge base of user preferences from the collected data and using those preferences to deliver a personalized service. Regarding the design of an ubiquitous solution for urban use, we will also be addressing the principles and key features that characterize this approach.

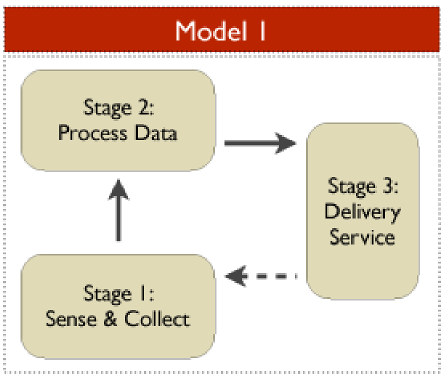


Figure 3.1: Abtract model of the information flow

3.2 An Ubiquitous Approach

During the research, the fields of Affective Computing (Section 2.5) and Ubiquitous Computing (Section 2.6) opened new insightful horizons on the advancements of pervasive technology that we could apply to create a seamless, fluent and useful application. This approach of a pervasive system lets the interaction between the Cloud2Bubble framework and the user to be done through a front-end application in a completely unobtrusive way. It can be seen as a middle layer in this communication, collecting data from the user and transmitting to the cloud and vice-versa.

The process of developing such solution started with brainstorming. Since the goals of the project were not restrictive, the prototype solution was flexible enough to follow any desired direction in the collecting of user-related data. It was during the research phase that the solution started to emerge, which in turn lead to further investigation. The brainstorming started with three critical questions that ultimately define our problem:

- What variables should be measured, tracked and collected from the user that gives us affective and context-awareness information during journeys?;
- How to extract that information and from which sources?;
- How to transmit it between user devices and the existing infrastructure in the cloud?

3.2.1 What To Measure?

One of the key objectives described in Section 1.3 is the introduction of emotional preferences in the PT choosing criteria. The research into the field of Affective Computing and QoE told us that emotions are a good indicator of what the user perceives from a particular experience [CPCG12]. And, thus, it helped to understand that by measuring a user's emotional state during the course of an experience we can more autonomously assess the quality of that experience. Emotions are useful to understand in which moments the traveller felt happy, stressed or calm in a journey. Nevertheless, this dimension can not solely explain its reason - what makes the user feel happy? And irritated?. Context information plays the combining role here. The environment affects - but is not the responsible for - the quality of a journey, as described in Section 2.2, alongside the performance of the service, for example. These two dimensions - affective and context - make it possible to stablish a weak cause-effect relationship based on which environment conditions make the user's emotional state change, as shown in Figure 3.2. It is weak given the numerous factors that influence the experience, such as, listening to music, the context or the journey (leisure vs work), landscape, conversations, etc. Therefore, although, some of relationships might be simple, others can be hard to determine. The solution lies in using complex reasoning algorithms possible to evolve in the future with the availability of more data in order to reach more accurate results.

The particular variables that measure affective state and context conditions depend on the tools available. As an urban application we look for collecting such data almost invisibly, as a pervasive solution should pursue. As so, besides those two dimensions - affective and context - we

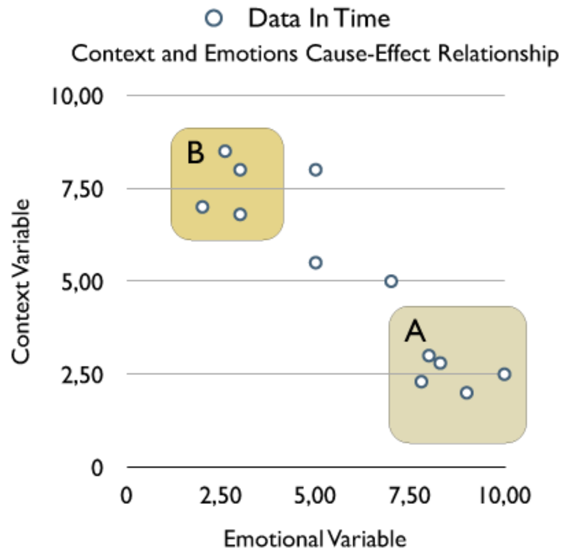


Figure 3.2: From the cause-effect relationship between context and affective variables is possible to distinguish what the user likes (A) from what he does not (B)

also defined two sources of data: the traveller and the vehicle/environment. The exact variables to measure in each component are described in the Mobile Sensing Platform chapter (Chapter 5). For now, in this higher abstraction level, it is enough to know we will measure context and affective data from the user and from the environment. Moreover, that this data will feed the reasoning system in C2B to understand the cause-effect relationships between the two domains.

3.2.2 How To Measure?

On the user's side, to offer him a natural casual interaction, the solution should use mobile and unobtrusive devices: no common urban traveller will go around with bulky devices on their heads or straps around their chests. One thing they carry all the time, though, is their smartphone. Smartphones, as discussed in section 2.7.1, are powerful computing and sensorial resources that are so integrated with our life styles that they are not intrusive any more. Also, as a sensing device, the embedded sensors provide plenty of user-related environment information, i.e. context surrounding the user.

As for gathering emotions, the smartphone may also be effective. Although smartphones do not have any tool for ubiquitously measuring affective data, they have the best and most accurate tool for direct input - the user interface. Giving the user the ability to insert his perception of the experience adds great value to the accuracy of the emotions. It can serve as a strong validation to other emotional data sources or as the full source of affective data. Besides the smartphone, we intended to gather additional affective information, but at the same time less obtrusively - without user input. Reading and assessing affective data is hard but as we saw in section 2.5, innovations in this area are making it possible, specially advances in wearable technology. In Section 2.5.1

it was presented some already commercialized products that can measure physiological data in a pervasive environment. Intelligent clothing accessories can nowadays incorporate such small and invisible sensors that makes them prone to mass adoption. Some of the solutions, such as the Basis B1, are used to measure physiological signals with ease and style. Thus, this concept will be used in the future work.

In summary, regarding the user, we will collect data from his smartphone which he carries everyday. This data will be of two types: close context information and direct input of emotional state. Also, for more objective affective information we will also gather data from wearable sensors. As of now, we only suggest using physiological signals since they are the only incorporated in unobtrusive sensors. As soon as newer urban-style sensors for measuring other domains, e.g. EEG, are available they can be integrated in this solution.

On to the environment's side, it relates to the installation of static sensing devices inside the vehicles. Such sensors collect general context data that can influence the experience of every user using the service at the same time. Environment conditions in the vehicle are closely related to those measured by the user's smartphone. However, they can be used to understand the overall experience rather in some isolated points. Measuring on the vehicle's side is specially important in three situations:

- when there are no passengers inside the vehicle to transmit context data to the system through their smartphones;
- soften the noise in a particular position with a general overview of the experience, i.e. broader context information;
- collect data that the smartphone can not, e.g. the estimated number of people inside the vehicle.

3.2.3 How To Transmit It?

The transmission of data has two stages. The first stage regards the gathering of all data from the above mentioned data sources into a sensor database in the smartphone. Structuring the flow of data helps to maintain the consistency and atomicity of the data collected in a particular journey. From the Section 2.7 we learned that the smartphone has the best connectivity and process capabilities of all the sensing devices. Therefore, the smartphone will take also this responsibility of managing the database of sensed data before transmitting it to the system. It connects with the environment and wearable sensors and groups all of their data along with its embedded sensor data in time-based clusters.

The second stage is the transmission of that whole database of sensed variables from the journey to the cloud. The smartphone acts as a client in a Service Oriented Architecture (SOA) that will be described in detail in Chapter 4. For now, the goal is to understand that the smartphone collects the data, in what we defined the Mobile Sensing Platform (Chapter 5), and transmits it to a server in the cloud, using its internet connectivity. There it will be processed and delivered back as a service to the smartphone, adding value to the user experience, as we will see in the next section.

3.3 Interaction Loop

After an introductory description of the ubiquitous solution envisioned, here it is presented the information flow between the two participants of the interaction - the user and the system. In the previous section we addressed the methodology for collecting and transmitting data. In this section, it is discussed what the system will do with the data and how to make good use of it to serve the customer. The system refers to the smartphone front-end application and the Cloud2Bubble platform. Both are combined in this loop to create a smart mobile application providing an intelligent service to the user. Therefore, some parts take place in the smartphone while others in the cloud infrastructure, though, this will be better understood in the Architecture chapter (Chapter 4).

Figure 3.3 shows the three stages of this interaction. The first stage corresponds to the collection of data on the user or client's side through a set of devices, including his smartphone, as described in the previous section. In the second stage the information will be processed in the server, in order to understand the correlation between the sensed variables and to create user profiles. Still in the server, various types of real-time information is gathered from PT web services, which, combined with the user profiles provides the opportunity to deliver the service to the customer. In the third stage, the service is delivered to the user. This application's service is to facilitate a better informed decision to the user. This is done by suggesting alternatives that would please him, not only in terms of duration and number of changes, but also context conditions. This is based on his profile preferences. The goal of the service delivery stage and the, for what matters the overall application, is to help the user choose the best alternative for his journey.

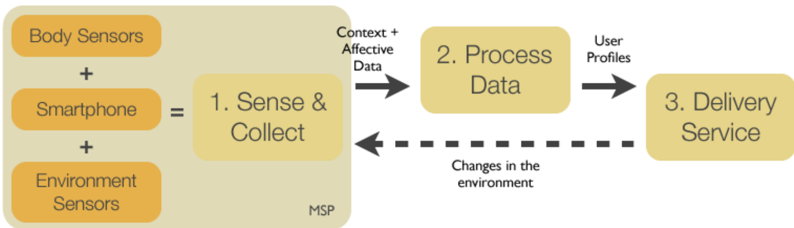


Figure 3.3: Information flow in the three stages of the interaction loop

3.3.1 Stage 1: Sense & Collect

It is in this stage that the Mobile Sensing Platform performs its sensing and aggregation of data from multiple sources. The data is collected in the smartphone from its embedded sensors and user interface, environment sensors and body sensors. The type has two dimensions: context and emotions. Collecting happens during the course of the trip and is sent to the server when it finishes. At this time, there are two possibilities:

- the system responds with a prediction, based on the transmitted sensor data, that tries to classify the journey for the user. This classification happens in the C2B by running a prediction based on the model that defines the user profile. This model is created and updated through machine learning. This process is not part of this project, and should be referred to Section 2.8 or [CPCG12]. That prediction or inference can, however, be edited by the user, resulting in the additional validation of the data that is transmitted back to the server to update/correct the model;

- if the system is not able to make a prediction, because it does not have enough data for an accurate model, it responds with a feedback request. This feedback is based on a simple interface where affective and rational data is inserted.

The details of such interaction is further explained in Chapter 5 and demonstrated in Chapter 6.

3.3.2 Stage 2: Process Data

The second stage takes place in the Cloud2Bubble platform immediately after the sensor data is received, as introduced in Stage 1. Here, all the information received is analysed in order to find patterns of satisfaction between the two domains. This is done by training the sensor data with the user feedback using machine learning techniques and defining models. Since the data is identified to be from a particular user, we have the opportunity to create and maintain the profile for that user. Every time more data is generated and received, these models are corrected and profiles are enhanced, which in turn allows more accurate predictions.

As a simple case scenario, if a user, during his trips, classifies his level of happiness and relaxation of good when the sensed sound level is low, i.e. calm environment, the model will reflect this relationship. The next time the user travels, if the sensed data indicates low levels of sound volume, then the output of the prediction system will be the labelling/classification of the journey as happy and relaxed. This is a simple scenario. In the practical use it is much more complex. However, as already stated, for further explanation of this machine learning and prediction process the Cloud2Bubble documentation should be referred.

3.3.3 Stage 3: Deliver Service

Delivering the service means adding value to the customer through the enhancement of his quality of his experience in riding public transportation. The main focus in this stage of the interaction is to gather reliable real-time PT data from their web services. Understanding when the current conditions do not please the customer, i.e. the prediction gives bad classifications, provides the mean to offer the service in the situations the user needs better informed journey alternatives. Better informed decision means planning a trip knowing the current conditions in the different alternatives. For each alternative is shown our improved choosing criteria with the service lines that compose the route, their durations and changes, and a classification of the current environment conditions, which include delays, number of people, noise, etc. The classification is based on the user's profile where his PT environment condition's preferences are saved (see Figure 1.1).

How the system delivers such service to the customer will be discussed throughout this document. In more detail in the Mobile Sensing Platform chapter (Chapter 5) and in the PTSense mobile application chapter (Chapter 6).

3.4 Designing Goals and Principles

In Chapter 2 it was introduced many of the key aspects of ubiquitous systems. This solution, also pursuing a completely innovative and pervasive system also shares those same principles. Invisibility, autonomy and mobility enable interaction with the system everywhere, making it more prone to user adoption [Wes11, CEL+06] by not interfering in everyday human experiences. This is the idea behind the IoT [MGT10], where devices intelligently communicate with devices to ease complex activities without burdening the user. Therefore, these are also the main guiding principles in the design of our smart solution so that travellers get a customized service without noticing it.

3.4.1 Unobtrusive Sensing

Healthcare applications usually prefer accuracy of readings over device size. However, in urban community-sized applications and contexts, like Public Transportation, unobtrusive sensing is essential, given the uncomfortability of carrying bulky devices all day and in crowded environments. Thus, here we will use a new generation of wearable computers that combines style and effective sensing capabilities in attractive accessories making sensing almost unnoticeable. The Basis B1 watch [Sci] is a good example of such accessories. It provides HR and GSR readings, Bluetooth connectivity and open API to communicate with the smartphone.

The smartphone embedded sensors and small static sensors will also be used to capture a considerable amount of data invisibly [ACRC09, MLF+08]. Thus, one of the main advantages of this design is how little the user is requested to participate. All of the sensing is done underneath the user interface and the complexity of the sensing platform is hidden. The user is only requested to see what really matters to him - predictions, inferences and route suggestions. By doing this we are providing him with support and advice without burdening him constantly. He can even forget this application, as long as it warns him when he really needs.

3.4.2 Wireless Communication

Not only should the devices be unnoticeable but it is also important for them to communicate wirelessly given the inconvenience of wires. Even more, the new low-power low-bandwidth protocols also provide the means to save big amounts of battery resources, specially when the platform consistently uses this technology. From the wireless technologies presented in Section 2.7.2 the ones chosen were Bluetooth, specially the new Bluetooth Smart version, and Wi-Fi. Bluetooth Smart was preferred to other low power technologies given its past in the mobile phone connectivity. Most of the accessories for smartphones used this former technology. Regarding the Bluetooth Smart, its growth is visible. Although it is a very recent technology, we suspect a wider adoption in the recent years. The introduction of this technology in the iPhone 4S, New iPad, Motorola Droid RAZR and more recently the Samsung Galaxy S III confirms this idea [SIG]. Also sensors, such as the Wahoo Blue HR are starting to use this technology. As of Wi-Fi, it provides a secure, simple and standard technology for communication with the cloud and environment sensors.

3.5 Conclusion: Ubiquitous User-Centered Design

In this chapter it was presented the brainstorming that evolved helped building the concept of an innovative pervasive solutions. Architectural details apart, it was introduced and explained the way we are going to assess quality of experience and use it to provide better services to the customer. We answered three objective questions which ultimately define the problem: what to measure, how and how to transmit it. We discovered that assessing emotional states and collecting affective and context data is essential to understand how the traveller liked the journey. These two dimensions correlate to determine which environment variables influence changes in emotional states and are ultimately used to define the user profiles.

To gather this information it was envisioned an approach based on ubiquitous design principles. An ubiquitous solution that hides the sensing platform by using wireless connectivity and unnoticeable sensors is more prone to user adoption. To collect the context and affective data it was identified the source points of data and designed the mobile sensing platform. These sources of context and emotional data are:

- **The user** – from the smartphone: close context data and direct emotional input. From wearable sensors: affective data in the form of physiological inputs;
- **The vehicle** – from static sensors: overall environment vehicle conditions.

Once collected, the data is sent to the server for creating and maintaining the user profiles, which comprises the second stage of the Interaction Loop. The third stage of this interaction happens still in the server. Real-time PT information is accessed and compared with user profiles in order to discover the best alternative journey for the user's routines in case the service is conditioned. To do so, it facilitates the information of the service lines and commutations, with the special focus on the classification of the current environment conditions for each service. This classification, as introduced in Figure 1.1, is based on the user profile model which holds his PT preferences. Choosing the best alternative ultimately improves his QoE.

Chapter 4

Architecture

4.1 Introduction

After describing the interaction of the whole system the next step is to understand how it is physical implemented. The whole project was quickly envisioned as a distributed system running over a cloud computing module, the Cloud2Bubble infrastructure, that provides services to the front end mobile application - the service consumer. This architecture is based on a new generation of smart mobile applications that use cloud computing capabilities to provide intelligent context-aware services. The focus of this project lies only on the front-end of such application, the smartphone interface, and its interaction with the cloud. The Mobile Sensing Platform is the responsible for generating user data, while the C2B works as a web service with the reasoning and analytical power to process that data. This architecture serves many principles found in client-server architecture and provide several features that make it a solid approach in this context.

In this chapter we introduce and explain how this system implements a Service-Oriented Architecture Web API bringing scalability, interoperability and granularity as key advantages to our solution. In Figure 4.1 we can also see that both entities work on the interaction loop for an integrated solution, as referred in Section 3.3.

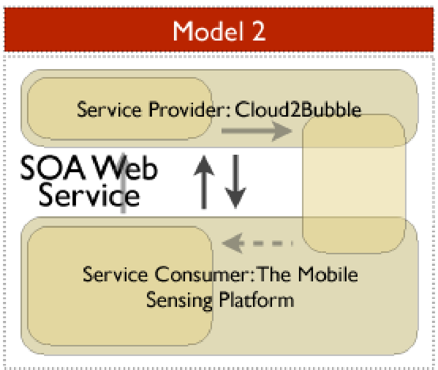


Figure 4.1: Simple Service-Oriented Architecture of the solution using the C2B for remote cloud computing

4.2 Cloud computing and SOA Web Service

The key point in our solution design, from its early concept, was organizing the resources in a distributed system, specially for two reasons:

- The Cloud2Bubble framework is supposed to be a generic service provider that works in several contexts, one of which is Public Transportation. Therefore, its services can be business functionalities shared by many applications;
- Not overloading the mobile application with the processing of large amounts of data while it could be done in a much bigger, capable platform.

The best approach to connect both entities was to use the C2B structure as a web service, accessible by the mobile application through the HTTP protocol. Web services make functional building-blocks accessible over standard internet protocols independent of platforms and programming languages. As so, it was defined that the C2B would implement such computing resources and offering them to the smartphone application. This methodology is already used by most of the recent mobile applications. It gives access to web resources everywhere at any time using a simple internet connectivity, which largely extends the functionality of a smartphone application. It uses not only core procedures, but also powerful cloud computing functions.

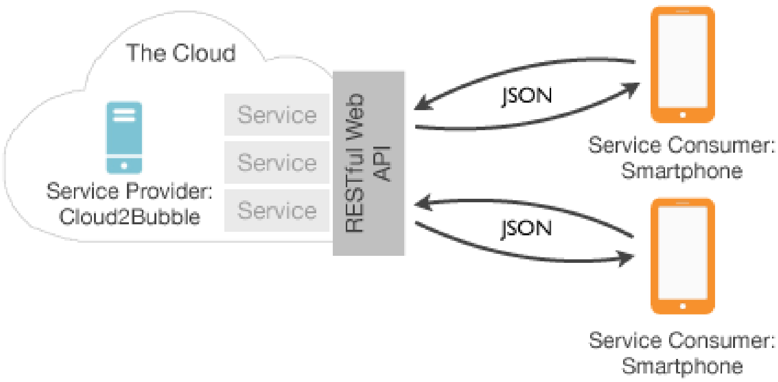


Figure 4.2: C2B provides RESTful web services as abstractions, reusable functionality and interoperability to consumers

More generally, this web service solution implements a Service-Oriented Architecture (SOA). Rather than defining an API, SOA defines a simple interface in terms of protocols and functionality. This keeps all the business functional blocks abstract from the user. In fact, the mobile application does not need to know the complexity underneath the C2B module. The key is independent services with defined interfaces that can be called to perform their tasks in a standard way, without a service having knowledge of the calling application, and without the application having or needing knowledge of how the service actually performs its tasks. Thus, in practical terms, the Mobile Sensing Platform and C2B are loosely coupled and the first uses the second's functionalities or services as an abstraction (see Figure 4.2).

Cloud computing, on the other side, refers to the delivery of computing and storage capacity as a service. It entrusts services with a user's data, software and computation over a network, where usually a complex system is installed. Cloud computing and SOA are complementary activities and together they offer a cloud service where cloud computing's platform and storage service can provide a value-added for SOA efforts [Rai09].

SOA also promotes the goal of reusable services, which perfectly serves the objective of the whole framework. Individual functions or services (shared services) can be called by different applications for different purposes. The services implemented in our service provider - the C2B - can therefore be run on various distributed platforms and be accessed across networks. Since ideally we want this framework to scale to more contexts and platforms its interoperability is fundamental. Thus, this architecture supports the future work.

4.2.1 Cloud2Bubble and the Mobile Sensing Platform

The Cloud2Bubble is in the cloud and offers the called web services over the internet. This component and its specific implementation comprehends a parallel project and [CPCG12] should be referred to better understand its operability. For the context of this project, it enough to know that it is hosted in the Google App Engine and delivers computing and storage capacities as a service to the Mobile Sensing Platform in the smartphone, our only service consumer. The characteristics and interface are identical to the identified above for a general SOA web service.

The Mobile Sensing Platform is also called the web service client and simply uses the HTTP protocol to access the RESTful web service of C2B. This means the smartphone can use any sort of internet connection, such as Wi-Fi or mobile data, e.g. 3G. Also, MSP accesses the resources in the C2B without knowing its implementation, a characteristic similar to Object Oriented Programming (OOP). For this reason, the specification regarding the cloud processing is not covered in this document, since, in the client's point of view, it is treated as a black-box interface. Nevertheless, the MSP needs to know the set of services available in the cloud. Given the small size of the web service and reduced number of shared functionalities, this interface is known *a priori* by the client without the need of a service broker.

4.2.2 RESTful Web Service

Objectively, this application is a SOA web service that is exclusively used on the web. This characteristic makes it a good example to use RESTful web services. A RESTful web service (also called a RESTful web API) is a web service implemented using HTTP and the principles of REST. It also relies on the built-in HTTPS protocol which provides a reasonable level security and privacy. By relying exclusively on the internet protocol, RESTful interfaces are easily scalable and simple to implement.

The RESTful interface of the Cloud2Bubble web service is a collection of resources, each defined by four aspects:

- the base unique resource identifier (URI) for the web service – `http://cloud2bubble.appspot.com/api/<service>`;
- the internet media type of the data supported by the web service – JSON;
- the set of operations supported by the web service using HTTP methods (e.g., GET, PUT, POST, or DELETE);
- the API is hypertext driven, following the RESTful principles.

The main point here is that a REST architecture provides a standard interface between the client and the server that provides civilized access to resources. Similarly to a browser accessing and retrieving information from an remote resource, the MSP posts and gets data to/from the Cloud2Bubble web services. The key feature of this REST communication is the possibility of sending objects of data back and forth. The information gathered from the sensors is then encapsulated into a single serializable object that is understandable on the server side. JSON objects serve perfectly for this purpose.

As a case scenario, when the smartphone application transmits sensor data to the server it sends a POST request to `http://cloud2bubble.appspot.com/api/trip/data` with the data encapsulated into a JSON object. To send a feedback, on the other hand, it does the same but this time the JSON object represents the user feedback input and the POST request is to a different resource: `http://cloud2bubble.appspot.com/api/trip/feedback`.

4.2.2.1 JSON Encoding

JSON objects define a set of key-value pairs that are easily encoded and decoded. This allows to build a comprehensive structure from the sensor data structures defined in the application code. Later, on the server side, this same objects can be constructed with the exact same information.

Since JSON is a standard and widely used content type in RESTful communications, it provides an easy tool for clients and web service to transmit data. The communication requests are independent of platform and language, as long as the correct data - well-formed JSON objects - is sent to the correct resources, serving interoperability issues. Anticipating its implementation, the following represents a user feedback data in a JSON object:

```
{
   "comment":"Nice ride!",
   "inputs":{
      "reliable":"7.9",
      "ambience":"5.7",
      "happy":"4.4",
      "relaxed":"4.0",
      "noisy":"7.7",
      "fast":"7.9",
```

```
        "crowded":"7.3",
        "smoothness":"6.5"
    },
    "device_id":"358049040983327",
    "trip_info":{
        "reviewed":false,
        "end_time":"2012-05-30 18:59:36",
        "database_id":2,
        "origin":"Polo Universitario",
        "service":"Metro do Porto",
        "start_time":"2012-05-30 18:38:57",
        "line":"D",
        "destination":"D. Joao II"
    }
}
```


4.3 Conclusion: New Generation Services

In this chapter we took a look at the physical implementation of this system and the interaction between the Cloud2Bubble framework and the Mobile Sensing Platform. The distributed computing model between our components was defined since the beginning of the project, which allows expansion to other contexts and takes advantage of resources of a much more capable platform than smartphones - the cloud.

The approach converged into a Cloud computing architecture implementing a SOA web service, also using a RESTful Web API as an interface between the provider and consumer. Both cloud computing and SOA share concepts of service orientation which focus on the definition of service interfaces and predictable service behaviours. SOA represents an open, agile and extensible architecture comprised of autonomous, interoperable and possibly reusable services implemented as web services. It can establish an abstraction of business logic and technology, resulting in a loose coupling between these domains.

Cloud computing, on the other side, represents a profound change in the way industry provides Information Technology (IT) resources. The emphasis is to leverage the network to outsource IT functions. Cloud computing provides access to virtualized IT resources that are simple to use and accessed over the web. The Cloud2Bubble is housed in the Google App Engine, providing this SOA web service functionality, also called a cloud Service.

To communicate with the server, clients use a RESTful interface which has gained widespread acceptance as a simpler alternative to SOAP. It is a key design model that embraces a stateless client-server architecture in which the web services are viewed as resources and can be identified by their URLs, making use of the HTTP protocol flexibility and internet standards. Thus, the MSP can post and get data from the service provider easily over an internet connection, everywhere at any time.

Finally, in terms of advantages of this new generation services we can say that both cloud computing and SOA can support good engineering practices by enabling fundamental concepts such as abstraction, loose coupling, and encapsulation. Both models rely on the definition of clear and unambiguous interfaces, predictable performance and behaviour, interface standards selection, and clear separations of functionality. Complementing storage and computing capacity with an SOA approach over the HTTP protocol with a RESTful interface helps to improve manageability, granularity, flexibility and scalability in our solution and its future work.

Chapter 5

The Mobile Sensing Platform

5.1 Introduction

The previous chapters focused on the overview of the smart application and the methodology used to collect and transmit data between both components - the Mobile Sensing Platform and the Cloud2Bubble. The conceptual architecture and workflow was also covered. As this point, the dynamics of the whole framework are defined. The next stage of the development is detailing the Mobile Sensing Platform, which is the focus of this project. It refers solely to the smart mobile application prototype for interacting with the user, seen in Figure 5.1. It facilitates the sensor data collection, aggregation and transmission, and the delivery of service, as introduced in Section 3.3.

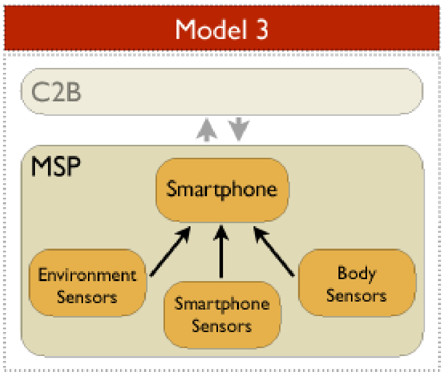


Figure 5.1: The sensor network centralized on the smartphone in the Mobile Sensing Platform

The MSP operates on the users' side and is expected to immerse them in the new generation Internet of Things (IoT). This concept of a massive sensorial network, discussed in Section 2.7.2, provides a mean for collecting huge amounts of data that characterize the dynamics of the city. In this chapter, we will cover the importance of sensor networks starting from the personal network that supports the MSP until the opportunistic sensing aspect of the application. The proposed architecture for the MSP leverages the power of WSN making the collection of data much more practical. Nevertheless, much of this technology is emerging and, thus, not freely or easily available, which brings some restrictions on the testing and implementation. Thus, this platform proposes a conceptual architecture that may take some time to mature, but its complete implementation was not part of this projects goals. As a proof of concept, MSP reaches its objectives and its initial implementation is covered in the PTSense Android application chapter (Chapter 6).

The first part of this chapter addresses the key factors that make the MSP and Cloud2Bubble an IoT system. Here we explain the fundamentals for implementing a reliable sensor network. Following the same flow of ideas, it will be explained the technical details of its design. In this chapter, the main focus is the individual behaviour of the isolated components of the MSP, highlighting the smartphone, and the specific data generated and collected.

5.2 Potential of an Internet of Things System

The world of connected devices is finally taking a big turn. More and more objects are becoming embedded with sensors and gaining the ability to communicate. This rising concept is moving the biggest companies in the world (IBM, Cisco, Google, Oracle, McKinsey, etc.) on evolving the internet into a global community of connected users and devices. "When objects can both sense the environment and communicate, they become tools for understanding complexity and responding to it swiftly" [CLR10]. This is the key goal of this system. Specifically, it falls into the IoT category of Sensor Driven Decision Analytics, meaning it aims at assisting human decision-making through deep analysis and data visualization.

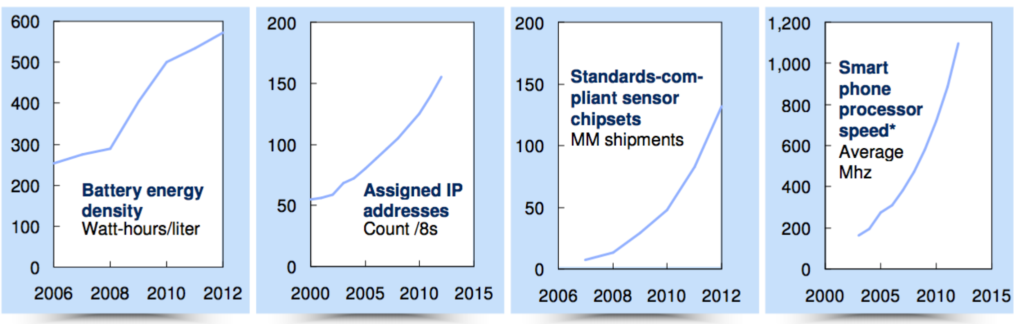


Figure 5.2: McKinsey & Company's study shows technology trends are enabling the emergence of an IoT[1]

Several factors contributed to the choice of this embedded sensing architecture in our system [CLR10] (see Figure 5.2):

- Improved power management and miniaturization of sensors allow them to be integrated into more physical devices. They are also gaining new capabilities while costs, following the pattern of Moore's Law, are falling;

[1]Copyright 2010, McKinsey & Company

- Advances in wireless networking technology and the greater standardization of communications protocols make it possible to collect data from these sensors almost anywhere at any time;

- Massive increases in storage and computing power, some of it available via cloud computing, make number manipulations possible at very large scale and at declining cost.

Our MSP is based on a WSN that uses the technologies from the above list. This solution has advantages in terms of smartphone battery resources, ubiquitous data transmission and sensors, aggregation of multi-modal data and strong analytical capabilities. In a world of mobility this proves to be a good solution for urban public transportation use. Figure 5.2 shows that IoT is based on evolving technologies that are rising accordingly and will grow even more in long term with the world wide adoption of the Internet Protocol version 6 (IPv6) protocol. This signs show that this architecture may have a proper implementation in the future and succeed with the maturity of the IoT.

5.3 Sensing over a Sensor Network

Mobile Sensors Networks, as introduced in Chapter 2, are a great source for single-user multi-modal data or for creating large community sized sensor databases in the context of social networks and location-awareness applications. Our solution, works on both perspectives: on the user level it collects data from multiple sources consisting of a personal sensing model, and on a community level it gathers data from multiple users in an opportunistic sensing architecture. The latter means that data collected from the individual users regarding the service's environment conditions - not personal affective data - may be used to map a real-time network of the service's state. Imagine hundreds or thousands of people reporting current conditions. This real-time data is then used for all the other commuters about to use that PT service. Its benefits are tremendous. For example, it allows a customer to choose a bus instead of the metro because at the moment he knows the metro is more crowded than normal, even thought the bus ride can consume more time.

Back to the MSP personal sensing architecture, it is comprised, as seen in Figure 5.3, of various components that acquire different data:

- **Smartphone** – an Android application that 1) collects context data from the embedded sensors, 2) gathers data from other sensors and 3) communicates with the web service;

- **Environment Sensors** – sensors placed in the vehicle to collect ambient contextual data;

- **User Sensors** – wearable computers used for measuring affective data, in particular physiological signals.

All these devices communicate wirelessly. As referred in Section 3.4.2, they will be connected through Bluetooth Smart and Wi-Fi. Sensing over a sensing network with this specifications gives plenty of features to the MSP:

- Communication of devices within a personal space;
- Market applicability and device interoperability given the wireless standard;
- Low power consumption;
- Low implementation cost;
- Low barrier to market entry given the growing number of smartphone users;
- Create a complex sensing device from small parts and acquire data that could not be collected with a single device.

Some constraints still remain, though. There is still not a standard or open interface that allows an ordinary communication between devices, although Bluetooth Smart and Bluetooth Smart Ready devices are a good promise. Also, the number of wireless communicating sensor devices is still extremely low. Therefore, the wireless communication with sensors has not been tested during the course of this project and stays the most important goal for future work. More on this in the Conclusions chapter (Chapter 8).

In the following sections the components of the MSP are described, regarding the specific information collected.

5.3.1 Smartphone Sensing

The smartphone is the central node of the network given its processing and connectivity capabilities. It acts not only as a sensing node, but also as a knowledge base with sensorial data from every other sensor and as a communication interface with the cloud using internet connection.

The latest version of the Android platform (4.0 Ice Cream Sandwich) offers support for a handful of sensors such as, accelerometer, digital compass, gyroscope, GPS, microphone, camera, barometer, temperature and relative humidity [Gooa]. The smartphone can collect plenty of useful data, but the most interesting aspect is that since it is a personal device, also the data is personal. By tracking record of a particular device's data, we are in fact monitoring a particular user. Nevertheless, privacy is guaranteed through the wireless technology protocol. The smartphone accesses the web service through HTTP using its standard secure protocol. For sensor communication the same concept applies, it transmits data to Bluetooth paired devices, similarly to connecting a personal hands-free kit. However, to avoid switching requests, e.g. answer a prediction to other user, the requests use the unique device id in its JSON contents.

Besides sensing and cloud communication, the other goal of the smartphone is to gather affective data by using participatory sensing. It means the user provides more data to the system than the previously autonomously sensed. This data, as explained in Section 3.2.2, is important to validate the other emotional data collected or to serve as the full source of affective data. Sensed data, specially in noisy and crowded environments is always subject to interferences that do not reflect the actual circumstances. The mood of the traveller is affected by several external factors, as

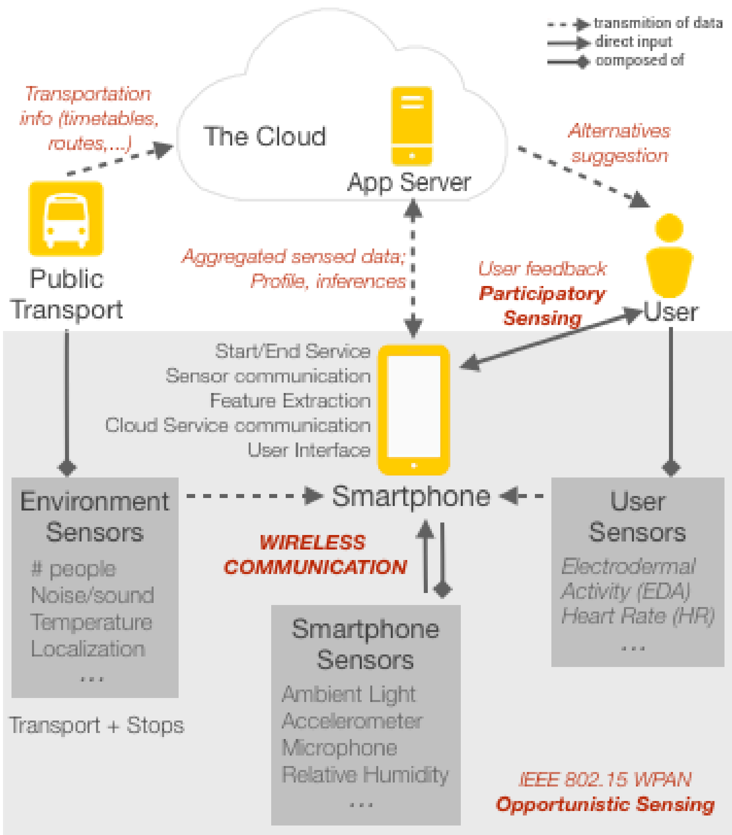


Figure 5.3: The sensing platform gathers data from environment, smartphone and user sensors along with user input and communicates it to the cloud infrastructure

already stated. Thus, making the user review the system predictions (see Section 3.3.1) validates the collected information.

Reviewing is as simple as looking at the system prediction received after a trip and rationally edit it. This prediction shows classifications of variables, e.g. noise, made by the reasoning module in C2B, based on the user's previous reported feedbacks. The model used for such functionality, as briefly explained in that same section, represents one of the services that MSP requests to the C2B through their RESTful interface (see Section 4.2.1). The smartphone application does not care about the way the predictions are created, it just receives them and presents to the user. This is better understood if looked at their practical interface, shown in its implementation, in Section 6.3.3.2.

During the first uses of the application it is normal that the prediction module may not be able to create an inference - the model is still very weak. In those situations, the request for a prediction fails and the smartphone instead requests the feedback of the trip to the user directly. Inserting a feedback, similarly to the review, is equally simple. The user is presented with a set of stimulus - context and affective - that he needs to classify. This information is then sent to the C2B for improving the module with the sensed data. The appearance of the feedback is also shown in the next chapter, particularly in Figure 6.8.

5.3.2 Context-Awareness & Static Sensors

Environment sensors, unlike smartphone sensors, read data that influences not only the QoE of one person, but from everyone that uses the same transportation service. It is important to have an overview of the general conditions that are not modified by subjective criteria. Also, some of this readings may be sent directly to PT web services and accessed in the cloud (see Figure 5.3). That is what happens already with the position and delay time of vehicles in various public transportation systems, which we assume in this work. The explanation for using these sensors was presented in Section 3.2.2.

In the vehicle sensors there are two types of real-time sensing based on the data they collect:

- data transmitted to PT web services for their own use, which happens nowadays. Such data, since it is not received from the smartphone's reports, the C2B accesses it in the cloud (see Figure 5.3). This data includes location, delay time, and estimated number of people;
- data transmitted directly to the smartphone: ambient temperature, background noise, oscillation levels and vehicle/line information. There is also the possibility of sending this information to the web directly, which allows the system to know current environment conditions of a vehicle even when there is no smartphone inside.

Such architecture is completely dependent of the public transportation services and its future infrastructures. In this solution we proposed the installation of such sensors given the possibility to measure useful context-related in smart cities. Furthermore, its operability could also differ between the two possibilities above, but until a possible testing environment we include both alternatives in this specification.

5.3.3 Affective Data & Body Sensors

The widespread use of affective sensing has been limited due to several practical factors, such as lack of comfortable wearable sensors, lack of wireless standards, and lack of low-power affordable hardware [FDG+10]. However, these constraints are slowly disappearing with the technological advances in this area and the commercialization of new light and stylish products [Sci]. As so, we allow the use body sensors in our solution to add useful affective data to the reasoning system and later understand which conditions make the emotional state change.

Wearable sensors are, probably, the most third-party-dependent module in the Mobile Sensing Platform. Craft new specific sensors to incorporate our solution is completely out of the options, because of two reasons: the commercial solutions are fastly converging to meet our requirements; and it would break the WPAN's principle of plugging in. MSP should work with any wearable sensor meeting our requirements and not only with specific devices. Therefore, in the future we expect to use commercial solutions, such as the Basis B1 (see Section 2.5.1). The requirements for a wearable device integrate our application are:

- measure any sort of affective data, including HR, GSR, EDA, EEG, etc.;

- be accurate, but not healthcare-like accurate;
- wirelessly connectable, ideally through Bluetooth Smart;
- open API, providing a programmable interface to communicate with the smartphone;
- small, comfortable and appealing, since it is supposed for urban adoption and to be used during through long periods of time. Watches and bracelets are a good example.

5.4 Aggregation and Processing of Data

After collecting every variable that can influence our perception of the quality of the service, this system intelligently defines profiles of satisfaction through a complex learning process. The raw sensed data from the various types of sensors is aggregated in the smartphone where some of its features are extracted. After pre-processed it is transmitted to the cloud for feeding the machine learning mechanism and improving the model for a particular user, which again is beyond the context of the sensing platform and this document.

5.4.1 Feature Extraction

Feature extraction is fundamental for 1) manipulating data into understandable, meaningful values, such as calculating oscillation level based in the three accelerometer variables, and 2) freeing connectivity resources by not transmitting huge amounts of raw data to the server continuously.

Regarding the resources, the smartphone groups intervals of sensed data from every sensor in clusters based on time. Clusters of 2 seconds of information saves processing and connectivity resources, as opposed to process and transmit every new generated data. The typical sensor read rate is a new value per 200ms which would give thousands of entries for a single journey. We use intervals of 2 seconds where all the data collected in the meanwhile is aggregated in a single value and then added to a data structure with the rest of the sensor data. In a typical scenario, this would result in 900 readings multiplied by the number of sensors available for a trip of 30 minutes, which given the storage of current smartphones is a reasonable amount of data. Besides reducing storage and connectivity resources this also provides reliable data since 2 seconds is still a small interval where a fraction of a second event (e.g. someone shouting) has impact on the average value.

For most of the sensor data, the average of the data collected in the 2 seconds is enough. However, there are two different treatments: the calculation of oscillation level and decibels. For the oscilation we use a similar approach of [LCCL10], by first calculating the acceleration at time t following the ISO 2631 standard:

$$a_t = \sqrt{(1.4a_t^x)^2 + (1.4a_t^y)^2 + a_t^{z2}} \tag{5.1}$$

and calculate the acceleration level at time t, i.e., L_t, by:

$$L_t = 20\log\frac{a_t}{10^{-5}} \tag{5.2}$$

This L_t is the information added to the database, instead of the three values from the accelerometer.

For the sound, since the Android only provides a function to get the maximum amplitude in one interval, we calculate an approximate power, i.e. G_dB, in decibels as [2]:

$$G_dB = 20\log\frac{Amp_t}{Amp_{ref}} \tag{5.3}$$

where Amp_{ref} is a reference value that we set as the threshold for the readable sound in the smartphone, so the maximum possible sound level, which Android provides.

5.4.2 Classification

The classification takes place in the cloud computing and although it was not yet implemented it is described next. The main goal of classification in converting metric variables, such as decibels and centigrade degrees, to a simple 1-10 numeric scale based on historical sensed data. Briefly, if a user rates a noise level of 60dB with an 8 mark and 80dB with 6 during initial feedbacks, later a reading of 70dB would result in a 7 mark. It is also used an algorithm to qualitatively classify variances to the profile, i.e. one variable is good, below average or poor compared to the variable in the profile. In the previous example, the 100dB reading would be classified as poor (4), assuming 60db and 8 is the average. With such algorithms the visual interface is simplified to hide the complexity beneath and enhance usability, as will see in Chapter 6. Over time, the classification gets more accurate as more data and feedbacks are received, as shown in Figure 5.4.

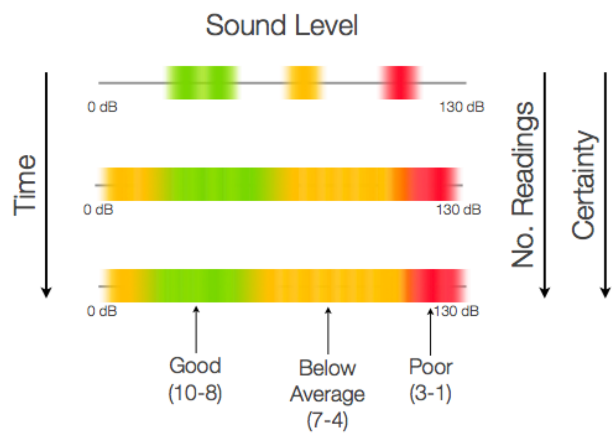


Figure 5.4: Classification algorithm. Over time and number of readings the classifications get more accurate

[2] See http://en.wikipedia.org/wiki/Decibel

5.4.3 Pattern Recognition and Profiling

When the data is transmitted to the server, it also comes with an unique identifier of the client, i.e. the smartphone. Unique identification of data, asides from the privacy issue discussed in section 5.3.1, is essential for 1) associating sensed data with the respective feedback and 2) updating the specific user profile.

To the user, a profile is a set of variable range parameters (e.g. preferred noise level between 56dB-63dB, meaning values in the middle were classified with higher marks) and list parameters (e.g. preferred means of transportation are metro, tram, bus) that characterize the preferred conditions of an individual, allowing the inference mechanism to compare them to current conditions or newly sensed data. Its appearance, in the PTSense application (Figure 6.10, helps to understand.

The way the profiles are created, represented and update in the cloud have been discussed in previous sections, particularly in Section 3.2.1 and 3.3.2. At a high level it consists in mapping the relationships between affective, including feedbacks, and context variables in a machine learning model. Since data is aggregated in time-based clusters, this means we can look at those correlations at a specific time. Variations are easily tracked over the time line of the journey and by overlapping different variables we are able to find these relationships. Figure 5.5 shows and example of this overlap. In the implementation, though, all the cluster data is fed to the machine learning which teaches the model with that set of data. Newer and more complex reasoning systems, such as the Google Prediction API [Goob], allow to update the model without needing to run the teaching process again with a new set of data. Although out of scope, this is a possible proposed tool for the Cloud2Bubble project.

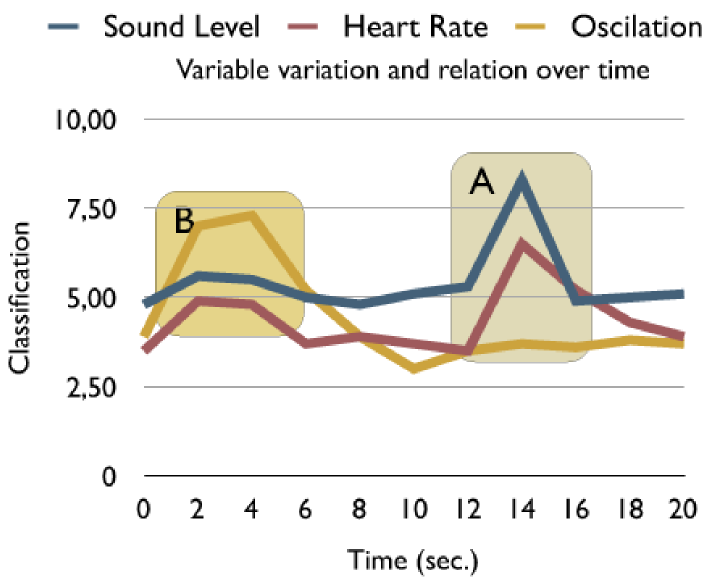


Figure 5.5: Some variations in environment directly influence emotional state, while other relations are more complex

Profiles also have information on personal routines, calculated autonomously from smartphone sensing data or given through direct user input. Routines display days of the week, start time of journeys and start and destination stops. The smartphone application will give the user the opportunity to insert routines, or to let the system identify by itself (see Section 6.3.3.3).

5.5 Conclusion: Building the Internet of Things

In this chapter it was detailed the architecture inside the Mobile Sensing Platform. This component comprehends the client side of the framework for this particular context of Public Transportation. It derived from the concept of the Internet of Things and by doing so shares the same principles and advantages. The most important for the reliability and operability of this solution are:

- Improved power management, capabilities, cost and miniaturization of sensors integrated into more physical devices;
- Advances in wireless networking technology and the standardization make it possible to collect data anywhere at any time;
- Massive increases in storage and computing power make number manipulations possible at very large scale and at declining cost.

Aside from this advantages, there are another number of indicators that the Future Internet will follow this trend of a global network where devices communicate amongst themselves. This was also the approach taken on the implementation of the MSP and its integration with the Cloud2Bubble framework. Under this approach, this technology fall under the IoT category of Sensor Driven Decision Analytics, meaning it aims at assisting human decision-making through deep analysis and data visualization.

This system comprehends sensing over a sensing network, at the personal application level and at the community size. The first represents the sensing platform to ubiquitously collect data from the user surroundings through multiple sensors. The second corresponds to an opportunistic sensing architecture. By collecting data from every user we can create a complete network of live service's state. This lets the application know real-time conditions of every line and provide useful information to the user.

The smartphone is used to collect sensor data that is transmitted securely to the cloud, but also to request the system prediction review or trip feedback from the user. The future of static sensors depends on the future service provider's infrastructure and operability. And wearable sensors are the most dependent component of the MSP, given the plugging in aspect of this solution that should let any third-party sensor interact with the application. However, newer sensors are converging to meet our requirements and will soon give the opportunity to test them.

After the data collection, it is aggregated and processed still in the smartphone. The data is grouped in time-based clusters of 2 seconds, out of which features are extracted. For most of the variables, the average is calculated, but to extract the oscillation and sound power, the formulas were presented. The classification and profiling that take place in C2B were also discussed.

The Mobile Sensing Platform rests on the foundations of the IoT and its sensing paradigm and model provides an innovative solution that uses rising technology, making it a solid approach for smart cities. The maturity of the technology is most likely to improve this solution and so, the future work will be easier and still on the edge. Finally, the proposed architecture covers all the objective of this project and provides plenty of further development to the initial idea. The next step is starting the implementation of this big solution, which is covered in the next section.

Chapter 6

PTSense App

6.1 Introduction

The PTSense Android application is the implementation of the Mobile Sensing Platform specified in the previous chapters. As the only point of interaction between the system and the user and responsible for a set of complex activities and communication with all the other devices and platforms, it is important to opt for a systematic approach in designing a user-centered application. After having defined the requirements and interactions within the architecture of the system, as shown in the previous chapter, we started by designing the user interface (UI) based on usability and simplicity given the target market of a common urban user. This kind of users want easy access to the most important features and the application to talk with them when it needs input. Taken this into account, a very detailed functional prototype was developed and tested with potential users in order to address usability issues and validate the UI. This was the first step to start implementing a simple yet functional application for everyday use. For the course of this chapter, prototype is referred to the interactive UI tool used for the usability testing, while the actual implementation is referred as PTSense application.

In this chapter we will talk about the implementation of the PTSense Android application, with special focus on the efficiency and the user. The prototype that shaped the Android implementation was an important step to ease the future work. The structure that it defines provides customers with an application that "talks" with them, leveraging the next generation services and a smart mobile applications.

This mobile application is not fully finished, as it was not the purpose of the project. Its main goal was to simply design and create a simple prototype for the collection of data. As so, in the end of this chapter it is shown that the development of such prototype went beyond the focus of the project. It provides a clean, commercial-like application that achieves all the objectives and adds space for the future development. It also provided the opportunity to test it in real environments, i.e. public transportation journeys, and take some important results for the upcoming steps of this project. The testing procedure and results are discussed in the User Testing chapter (Chapter 7).

6.2 Prototyping the Mobile Application

The RAD process followed in the development of this application had special focus in its implementation. It let us create prototypes and test them with supervisor and users to validate features. The key aspect is that this prototypes get more rich and solid as the development goes further, allowing not only the validation of implementation but also trying new features and take informed conclusions.

When starting to prototype an application, the tools used are essential for the user/customer/-client to have the best interpretation possible of how the application will look when finished. The use of black and white frames and sketches although providing a really fast tool for transmitting ideas do not attract that much attention of the client or do not reflect the actual appearance of the application. For this project, though, that was fundamental, since the purpose of the prototype was design and test the usability and interface, while at the same time showing its future features. Therefore, we opted to create a very realistic prototype combining all the features with an almost final visual aspect which gives a very clear idea of the final aspect of the application.

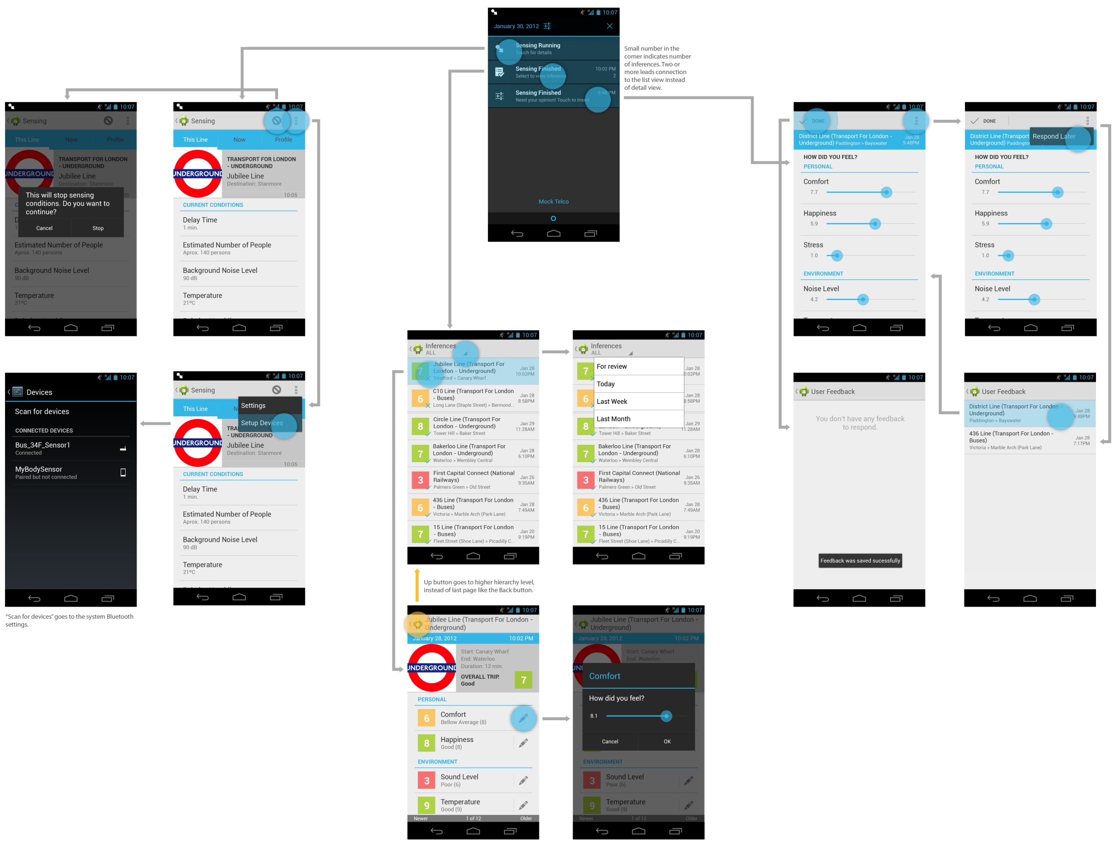


Figure 6.1: First prototype using Photoshop and Android ICS Design Guidelines

To start prototyping it was essential to first extensively read all the Android Design Guidelines [1] for the new Android Ice Cream Sandwich so that it would be according to the new platform standards. This prototype was made in Adobe Photoshop CS5, using a set of tool templates for this new Android version, resulting in a very accurate and real-feeling prototype, as shown in Figure 6.1. These tools were found online at the TechieTalks website [2], before the release of the Google official stencil, now available in the Android Design website. The used templates, although very useful and complete, do not provide all the wanted widgets or features, which had to be designed manually in Photoshop for the complete integration with the desired appearance. Figure 6.1 was the first interaction sketch made and is best viewed in the digital version of this document or in the external deliverable object, given its enormous resolution.

After validating this first interactions prototype, the rest of the screens were made and finally all these images were imported to the Mac OS's Keynote application to make an interactive prototype. This was achieved by creating hyperlinks over the buttons areas to the correspondent screen, giving the flow of the real mobile application, as shown in Figure 6.2. When saved as a clickable PDF presentation and opened in the smartphone it accurately simulates the application as installed on the device. Clicking and navigating within the application on the smartphone's PDF viewer gave us a wonderful idea of the overall and final interaction of the application. This final version of the prototype is available as a deliverable together with this document.

Before start implementing, this functional prototype was tested with ordinary users which made us change some details in the real Android application. This test is addressed in the User Testing chapter (Chapter 7).

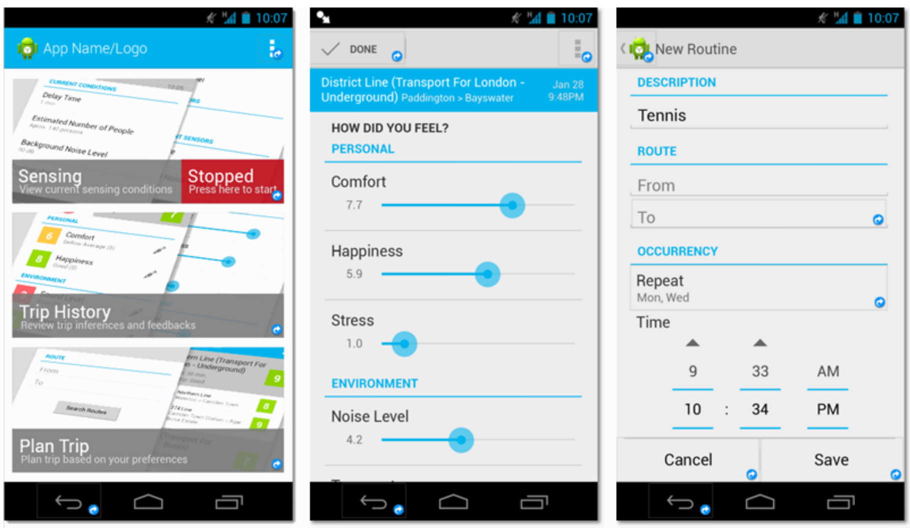


Figure 6.2: Using hyperlinks in Keynote presentation makes an interactive prototype to use with a PDF viewer in smartphone

[1] Android Design, http://developer.android.com/design/index.html
[2] TechieTalks, [Android 4.0] Ice Cream Sandwich GUI Design Kits in PSD, http://techietalkz.com/2011/12/13/android-ics-gui-design-kits-in-adobe-photoshop-format/

The prototyping phase was long but provided an accurate interface for the Android implementation. A benefit of this approach, as we will conclude, was the extraordinary simplification and less time needed in the coding phase.

6.2.1 Android ICS

The choice of the Android platform was decided in the early steps of the project, in the research stage. It was specially driven by the convenience of the developers - my mobile phone is Android - and its availability in the community - the number of Android devices is much bigger. Nevertheless, I did not know the Android framework, neither iOS, before this project, which added another learning phase between the research and the prototyping.

To build the prototype was essential to know the capabilities of the framework, but more importantly to follow its design guidelines. This approach is important to make this application integrated and in accordance not only with the system interface but also with all the range of applications you find in the Google Play store. The choice for the latest version of the Android platform – Ice Cream Sandwich – is obvious for new applications being developed, not only to be visually actual but also because of the new features it provides. For the PTSense, the ICS was preferred because it provides new sensor capabilities, e.g. ambient temperature and relative humidity, and also a better user interface, specially the useful action bars buttons, tabs and the more accessible context menu.

This first step of acknowledging the standards in the design was very important, since we were going to develop this application for the new Android operative system, to start, and then provide the backward compatibility for the older versions. The Ice Cream Sandwich is very distinctive from older versions and has very different features that we wanted to add to our application, providing an architecture supporting the future development. As shown in Figure 6.1 and Figure 6.2 it was used the new set of widgets interfaces, action bar and tabs, color palettes and interactions (e.g. back and up buttons) for the Ice Cream Sandwich, that were provided in the referred used template (Section 6.2).

6.2.2 Notification System and Usability

When designing an application we always have to consider the target customer and its use. The desired characteristics that were introduced in Chapter 3 for the Mobile Sensing Platform are autonomy and ubiquity in collecting information without interfering with the user's primary activities, and informing him of useful information. Therefore, the structure of the application is based on background services that perform the complex tasks and the use of notifications to notify the user of their state. The interface of the main application activities, i.e. screen, is useful and should also be meticulously designed, but when thinking on practical use, on a normal day, the expected interaction is through the notification system. The User Manual for Tests in Appendix F gives a practical use example (in its Section 2.2) for understanding this interaction through notifications.

The notifications, as shown in Figure 6.3 at the top, are of three types:

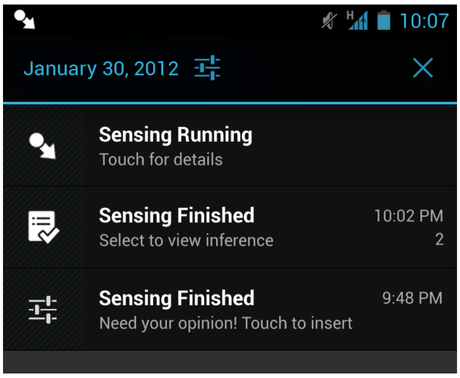


Figure 6.3: Notifications detail of Figure 6.1 (top)

- **Sensing Running** – indicates when the application is actually sensing conditions and leads to the application's sensing screen when clicked (see screens on the left of Figure 6.1);

- **Sensing Finished with Feedback** – informs the user that he has a feedback to respond about his previous trip. Also, if there is more than one it is automatically grouped into a single notification. If there is only one feedback, this leads to the actual feedback, otherwise to the list of pending feedbacks (see screens on the right of Figure 6.1);

- **Sensing Finished with Inference** – informs that the system was able to make a prediction of the previous journey's conditions and the user's emotional state and just needs the user to review, confirming or changing its classifications. Similarly to the previous notification, if there is only one inference it leads to that inference review screen, otherwise, to the list of pending reviews (see screen in the center of Figure 6.1).

By interacting with the notifications the user is guided to the correct screen of the application to finally insert his input. This approach addresses the usability issues of understandability and ease of use. Also, the list and detail views, defined in the Android Design guidelines, and its tab structure provides access to all the features in no more than 3 finger clicks.

6.3 Implementation

Creating a very detailed prototype using real tools or templates makes the implementation phase becomes much easier. Given the high fidelity prototype, the implementation was the translation of images to code, so to say, with the changes and feedback from the usability test conducted.

The implementation, however, is much more than building the user interface in Android. It is defining which actions should run in the background and when, and what should be transmitted between activities; what should be stored, where and when; and when to transmit data to the cloud platform. All this aspects are discussed in the following sub sections.

6.3.1 Background Services

Background Services are services that run in the background and do not have a user interface. They can interact with the activities of the application and during their lifespan they perform actions until they finish or are told to stop. In the PTSense application we use this kind of services to collect data from the sensors while allowing the user to focus on his primary activities. Also, the communication with the cloud service should be done without compromising the experience. The Figure 6.4 shows a schematic interpretation of the interactions between these services and the rest of the application and/or external activities.

It shows the Sensing Activity that is started from the home screen also starts the Sensing Service. This service keeps running in the background while the user interacts with other activities. When the Sensing Activity is brought back to the front again, the service starts

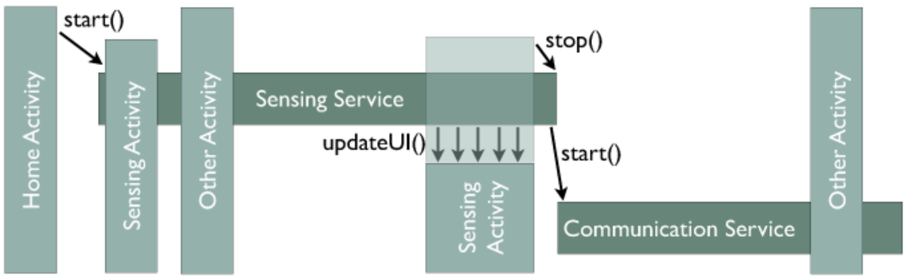


Figure 6.4: Services run in background while user interacts with activities

6.3.1.1 Sensing Workflow

The Sensing Service starts when the user decides to start sensing. This is done from the Home Activity, as shown in Figure 6.4 and starts along with the Sensing Activity. This activity, or just the Sensing Now Fragment (tab), works as the interface for the service, showing the user what variables the sensors are reading at the moment. However, when the user exits the application and starts other activities, either from the PTSense or external, the service keeps running in the background as normal. Because the user cannot see this service running we added the notification Sensing Running, as discussed in Section 6.2.2. When the user switches back to the Sensing Activity, it starts showing what the sensors are reading again, as seen in Figure 6.7.

The Sensing Activity and the Sensing Service are two separate components and they need to communicate in order to shown in the screen what the sensors are reading. To do that, we make the Sensing Service broadcast intents every half a second (500 milliseconds) with the information of the new readings. The Sensing Activity, on the other side, registers a listener for that type of intent and updates its interface with the data on it. This almost instantaneous process is unnoticeable to the user and builds a fluent interface.

The service is stopped when the user decides to stop, either from the Sensing Activity or from the Home Activity, in which situation send an intent to stop the service. In this initial version, it was also added a timer of 1 hour, after which it automatically stops the sensing, which keeps this process from running indefinitely if the user forgets to stop. The complete specification of the sensing mechanism is discussed in Section 6.3.3.1.

6.3.1.2 Server Communication

When the sensing finishes, the PTSense tries to send the collected data to the C2B computing resources over the interface specified in Section 4.2.2. The same happens after the user responds to a feedback or inference. In these situations we also use another service that runs in a process apart from the application - the C2BClient. The reason is simple: the device may not have internet at all times and trying to transmit data without connection would interrupt the foreground activities. The solution however is somewhat complex: if the device does not have internet connectivity, either Wi-Fi or mobile data, in these moments, then the application keeps listening/waiting for this connectivity. When it finally gets it, since the transmission takes place in a different process, the PTSense application does not need to be started in order to transmit the data, just the C2BClient service. And as soon as all the pending data is sent it self-terminates and the internet connectivity changes are ignored since we do not need internet any more. This transmission algorithm is described in Figure 6.5.

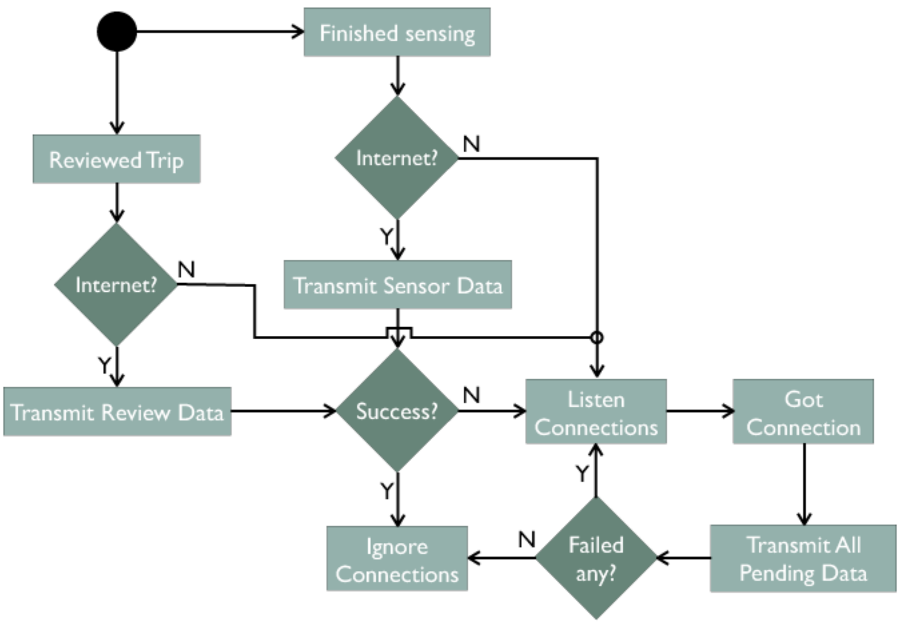


Figure 6.5: The connectivity algorithm to transmit data to server

6.3.2 Database

The database of the smartphone is used to store temporary data before transmitting to the Cloud2Bubble web service. This plays an important role both on persistence and battery. On the one hand, all the data is kept in the smartphone until it is not successfully sent to the server, so that when the device does not have internet connection or the transmission fails the data is not lost. On the other hand, by keeping it in the local database for the whole trip and performing the transmission all at once in the end we are saving battery resources compared to multiple short Wi-Fi connections throughout the journey. Also, this last option might not be possible some of the times, since there is no connectivity in the underground, for instance, for continuous transmission.

The data store in the database is, as shown in Figure 6.6:

- **SensorData** – from all the sensors available during the journey. An entry is identified by tripinfo_id plus the time (view Section 6.3.3.1);
- **TripInfo** – the information regarding a journey, associated to the sensed data and the review. All columns, except the "reviewed" tag, are needed to identify a unique journey;
- **Feedback** – the user feedback of the journeys as a set of stimulus classifications.

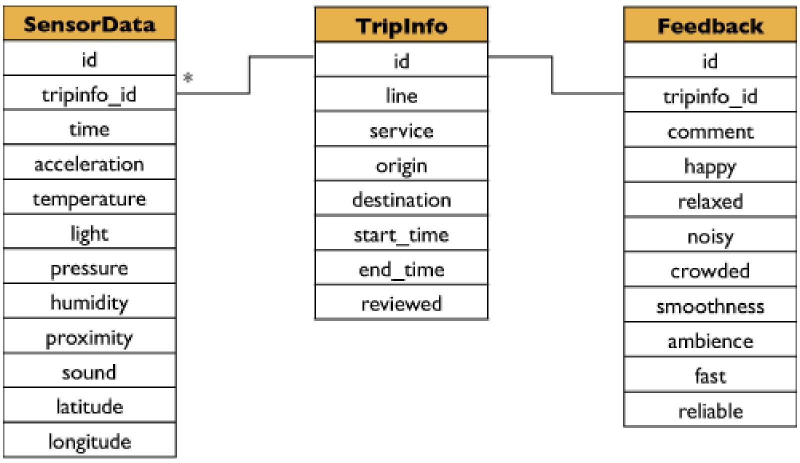


Figure 6.6: Schema of the database in the smartphone

In order to identify the data from multiple users in the web service it is also needed the unique identifier of the smartphone as well, both for the sensor data and the feedbacks, as discussed in Section 5.4.3.

6.3.3 User Interface and Features

In the following sub sections we will discuss the features of the application that are visible to the user and explain its details, always covering the efficiency and usability point of view.

Before we start, Figure 6.7) shows the implemented interface for the PTSense and it is quickly perceived that it is very close to the prototyped application. From the Home screen, screen 1) in that figure, we can easily access all the features, such as:

- start and stop sensing;

- view current conditions and live readings;
- review previous trips;
- plan journey based on your preferences;
- and from the context menu, view and edit profile and change settings.

6.3.3.1 Sensing

In this section we will discuss the first of the three options in the Home screen - Sensing. When the application is not sensing, this option is not selectable since there is no information to show about what currently sensing conditions. This explains the gray tone, which otherwise is similar to the other options. The only possible action in this situation is to start sensing the current journey. This is as easy as clicking the "Start" button, highlighted in that option menu, which prompts the user to insert the information for that trip, as seen in screen 2) of Figure 6.7.

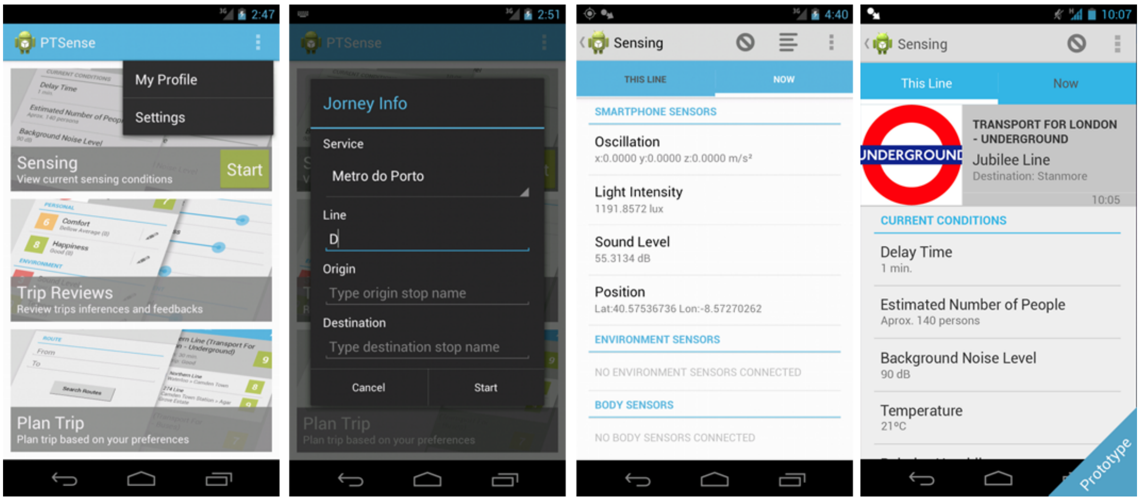


Figure 6.7: 1) Home Screen, 2) Journey Information, 3) Sensing Now, 4) Sensing This Line

In the journey information screen it was added the Auto-complete feature, since we want to make the experience really easy. So the user can start typing the service lines or stops from the selected service and it will show a list of options without the user needing to finish, as seen in Section 2.2 of Appendix F. As of now, this information is hand-held but the idea is to fetch all of the lines and stops data from the service providers. However, in the future we would like to make it even more autonomously by starting and stopping automatically, but we will cover that in the Future Work section of the conclusions (Section 8.4).

Another feature that was added after the usability test, is the possibility to edit the journey information whenever the user wants. It does not need to be totally inserted at the beginning: the user can always update this information during the journey, useful for situations where the destination is not exactly decided when the user starts the trip.

Regarding the sensing screen, it is divided in two tabs: the Sensing Now and Sensing This Line (see Figure 6.7). The first shows all the data being collected from the sensors at the moment. As explained in Section 6.3.1.1, this information is captured from the intent broadcasted by the Sensing Service. In this screen the sensor data is divided in the three source types of the Mobile Sensing Platform sensors: smartphone, environment and body sensors. The Sensing This Line tab gives current information about the line the user is currently in. Regarding the action bar, there are three more options: finish the trip, update journey information or setup external devices, which do not need further explanation. The third option, though, is not yet implemented.

Efficiency and Data Collection Details

Regarding the collection of data, the latest version of the PTSense only collects data from the smartphone sensors. As seen in Figure 6.6, we use the Microphone, Accelerometer, Light, Ambient Temperature, Relative Humidity, Proximity and Pressure Sensors, and GPS. All this variables provide different information of the context conditions or, in case of the proximity sensor, support them. This sensor's information lets the C2B discard useless data entries when its value is close to 0, meaning it is inside a pocket or bag.

The data received from the sensors is represented by float variables which are easily used by the processing module, the DataProcessor class. The data processing regarding the feature extraction was described in Section 5.4.1.

Besides the feature extraction, another issue we had in mind was the efficiency, which covers aspects like battery, storage and processing in the smartphone. Regarding the battery:

1. the data is fully transmitted at the end of the trip (view Section 6.3.2);
2. the Location Manager only update GPS location every 30 seconds;
3. sensor updates are received with the normal (slowest) delay (200 ms).

Regarding the storage:

1. use of reusable bounded buffers, that allows the sensors to insert data while the processing mechanism retrieves it. So it is putting and taking at the same time without needing to reallocate more space;
2. the data is grouped in clusters of 2 seconds that are small enough to trace all the events of the trip while also freeing some space. If we used all the data from the sensors it would result in 10 times more space (2000/200);
3. the sound samples are stored in the external SD card and deleted after the trip;
4. the pending data in the database is also deleted when transmitted to the cloud.

All the above characteristics also reflect improvements in the processing resources needed. This saves resources for other activities the user might be currently using.

6.3.3.2 Trip Reviews

Trip reviews are the reviews made and to be done of the previous journeys, either by the user - Feedback - or by the system - Inference. The Feedback List, shown in 2) of Figure 6.8) shows all the pending feedbacks the user has to respond in a descent order, i.e. newest first. When a user correctly responds to a feedback review it is transmitted to the server, as discussed in Section 6.3.1.2, and removed from this list.

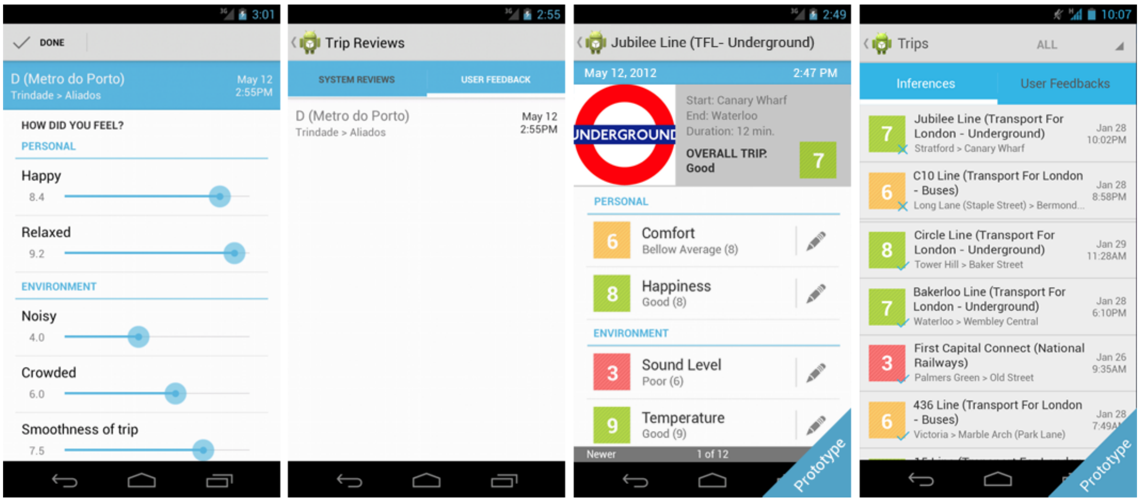


Figure 6.8: 1) Feedback, 2) Feedback List, 3) System Inference, 4) System Inferences List

Responding to a feedback represents the rational user review of that corresponding trip. As shown in 1) of Figure 6.8, this is as easy as dragging a set of sliders according to the user's opinion. The emotional model used here is based on the Valence-Arousal model (see Figure 6.9) to identify the emotional state of the user regarding that trip. Here it is represented by the Happy-Relaxed relation. Then, we have a list of environment stimulus that are easily classified over a well known 1-10 continuous scale. This is the useful information, along with the sensed data, that is given to the machine learning mechanism in order to train the user's model.

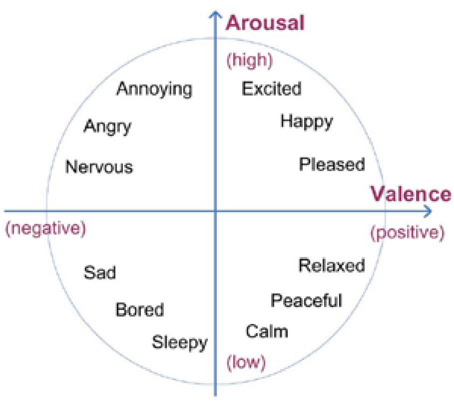


Figure 6.9: The Valence-arousal emotional model

On the other side, in 3) and 4) of Figure 6.8 we have the prototypes for the System Inferences part. The later shows the list of all the inferences, with the pending reviews on top, while in the action bar it is possible to filter based on time. The first shows a particular inference or prediction made by the system when it can predict with a certain level of certainty how did the trip go. The classifications are made quantitatively, i.e. with numbers, qualitatively, i.e. with qualities, and with colors for the users to understand easily, following the classification model explained in Section 5.4.2. So when the trip finishes, the user can see both the feedback or the review screens, but either way the user has always the last word by letting him change classifications with which he does not agree, clicking the pencil edit button. This alterations are re-sent to the C2B and the model is updated with the correction, as explained in Section 5.4.3.

6.3.3.3 Profile and Settings

Checking the profile or edit some settings are not actions you would do every day. So, and still following the Android Design guidelines, we grouped this options in the context menu of the Home screen.

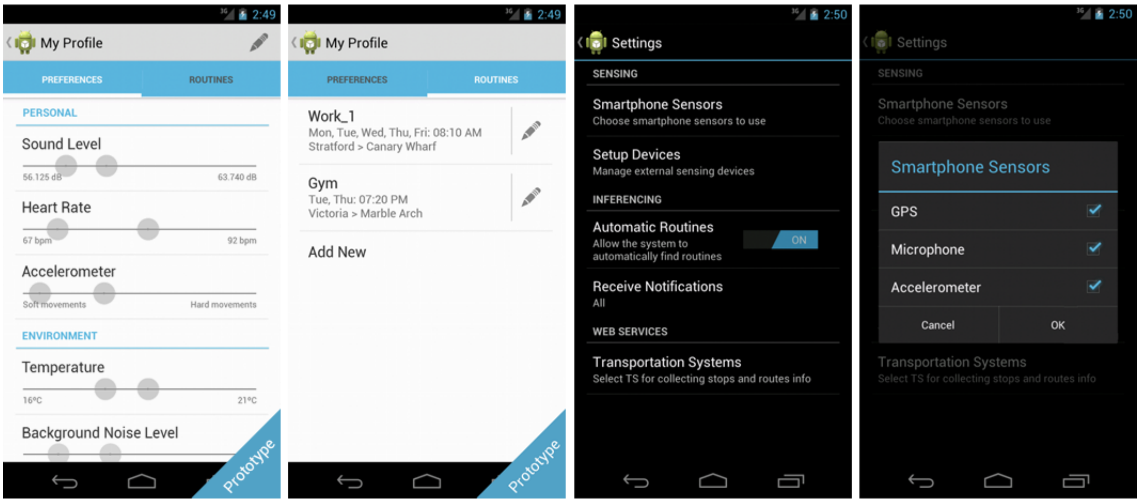


Figure 6.10: 1) My Profile, 2) Routines, 3) Settings, 4) Sensor Selection

Regarding the profile, the idea, as it has been expressed over this document, is to define and update automatically using the data collected and trip reviews. Nevertheless, the user can always check and edit his profile in the cloud (screen 1) in Figure 6.10) and help to give the application more accurate information on his preferences. Again, we opted for the sliders to maintain UI cohesion. Also in this are, the user can edit his routines. They can be defined manually in the Routines tab of the Profile screen (2) in Figure 6.10) or let the application find them automatically using the historical data from the previous trips, i.e. journeys and times, set in the Settings.

The Settings, as shown in 3) and 4) in Figure 6.10, provide simple tweaks to the application. The user can define which sensors to use and setup the external sensors, as discussed in Section 6.3.3.1; turn on the automatic routine identification for the system to try to identify his routines automatically; define which notifications to receive, i.e. all, none or only feedbacks; and select which service providers to use in the journey information screen. The list of sensors only displays the sensors available on the device, from those listed in Section 6.3.3.1.

6.3.3.4 Plan Trip and Pop-ups

Knowing the customer's routines is where the system can provide useful information by checking the current conditions on those trajectories and say something to the user if they are not according to his likings. This alert, as shown in 1) of Figure 6.11, informs the current state of the line that we usually take, and provides the user with a list of alternatives that currently have better conditions. This list of alternatives, 2) in Figure 6.11, provides a similar information to what we can have these days, but also the overall conditions on that trip and all of its segments. This defines the improved choosing criteria, similarly to the pursued goal expressed in Figure 1.1. With this real-time information the user has all the facts needed to choose a better journey for himself. Nevertheless, the list presents the better classified routes first.

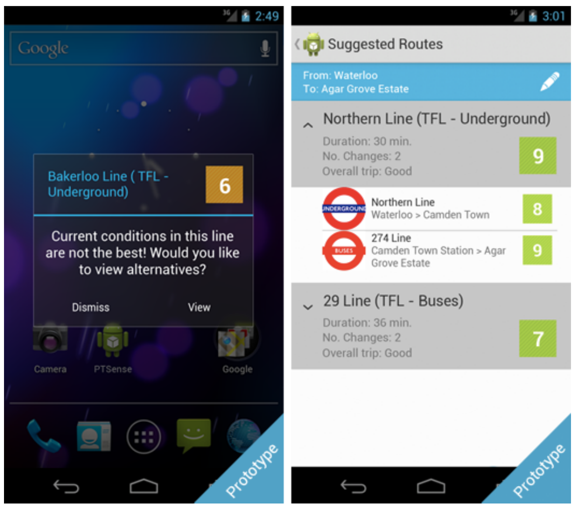


Figure 6.11: 1) Pop-up alert, 2) Alternative Suggestion

Pro-actively suggesting alternatives to the user and showing him all the real-time information he needs to make a better decision is what enhances the experience in choosing and riding public transportation. This goal, that goes beyond of the project, enters the ambitious commercial scene where such innovative and unique application could succeed. It does not only collect data but also, elegantly, delivers a useful service to the customer.

6.3.3.5 Unimplemented Modules

All the screenshots seen here were taken from the last version of the PTSense app, except for those with the "Prototype" sign, which represent a slightly updated version of the prototype. However, to publish the application for user testing, it had to be aesthetically finished. All the features that were not yet implemented at this time were substituted with a default screen, as shown in Figure 6.12. This fills the implementation gaps with a short summary of the expected features to be added in the future.

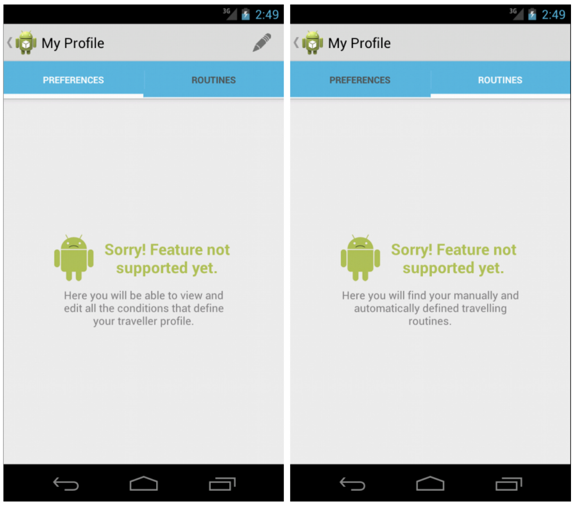


Figure 6.12: Default screen for not yet implemented features in the latest version

6.3.4 Backward Compatibility

Since the application was completely developed for Android ICS (API 14+), the last step was to allow its operability in older versions of the operative system (OS). Some of the features used were new in the ICS or different than older API's, such as the Action Bar, the Fragments, Sensors, Dialogs, Sliders and Multiselect Preference List. Gladly, Google provides a library package to allow this backward compatibility, making this classes available to lower API's. This solved most of the problems, but the use of the ActionBarSherlock [3] with some tweaks, made it possible to convert the whole application to versions of the Android as old as the 2.1, as seen in Figure 6.13. Most of the interface is the same, even with the new Action Bar, but the version dependent elements, such as the Dialogs, we opted to maintain for the consistency with the system, as seen in that figure.

However, because of this many changes in the interface, and specially because of sensor restrictions, i.e. the Ambient Temperature, Relative Humidity and Pressure Sensors are only available in ICS, it had to be made two different projects and packages. Nevertheless, this was not a problem for publishing, as discussed in Section 6.3.6.

6.3.5 Debugging

The implementation was followed by continuous debugging and testing, mainly in two ways: in physical devices connected to the computer and on emulator, since we cannot have all the physical devices to test all versions. To start, in terms of UI, all the versions in the range of 2.1 to 4.0.4

[3] ActionBarSherlock, http://actionbarsherlock.com/

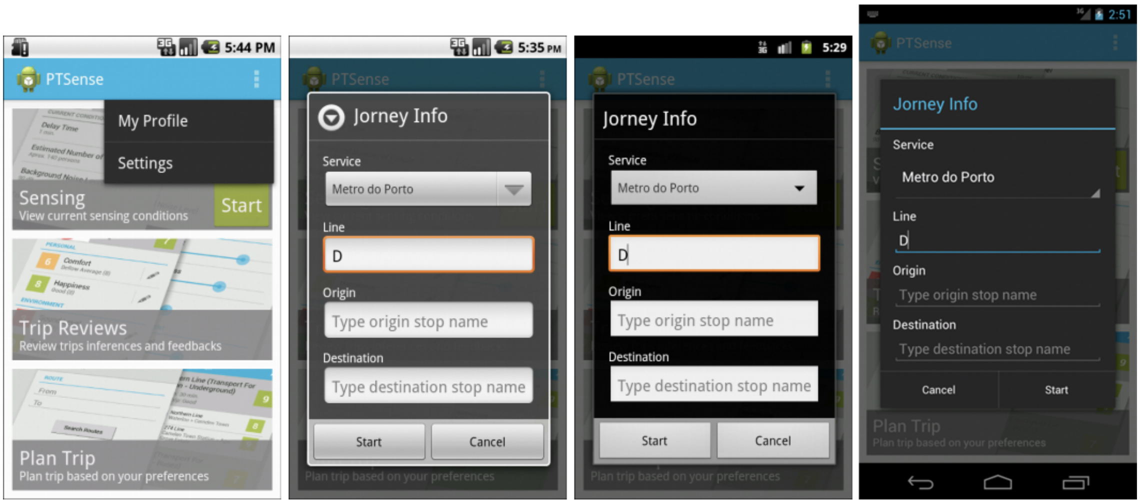


Figure 6.13: 1) Home screen in Android 2.1, 2), 3) and 4) Start Dialog in Android 2.2, 2.3.3 and 4.0.4, respectively

were tested in the emulator, where it was possible to ensure a correct interface for every user (see Figure 6.13). The more intensive debugging, though, was made testing the application on a real device with sensors and tracing logs in real time. From these logs, that are defined in certain points of the application, we can understand the flow of actions being taken and ensure the correct behaviour of the application.

Another debugging tool was the ADB (Android Debug Bridge) that allows to get access to the smartphone's data in the computer. This tool was specially important to visualize the state of the database during the interactions in the computer's Terminal application.

Finally, testing the communication and creation of JSON objects followed the same principles: use of logs reporting the content of objects being sent to the RESTful resources; and by tracking the reception in the Cloud2Bubble over instant messaging with developers of that project (see Section 2.8).

6.3.6 Publishing on Google Play

The final step of the implementation was to publish it online in the Google Play Store. The main purpose was to make it easy to access and install for the user testers in the last Feature Test, covered in Section 7.3. Given the multiple API versions available in the market, this store provides all the support to develop applications for as many devices as possible. Since ICS and its numerous changes, it is possible to publish an application with two different packages targeting different versions of the system, as seen in Figure 6.14. This information was consulted before starting the backward compatibility process, which encouraged the creation of two packages: one for the latest developments and completely untouched with support compatibility code; and another for older versions where its understandability is not a problem. This, however, brings the problem of duplicating the features in both projects. In the future, with the increase number of devices with ICS and later versions, the second version might be dropped.

The PTSense can be found on the Google Play Store on the mobile or online at https://play.google.com/store/apps/details?id=com.cloud2bubble.ptsense.

Figure 6.14: The Google Play Store allows multiple packages for same app

Also, for future work, the version codes follow a simple standard:

1. first 2 digits represent the minimum API version;
2. next 2 digits represent the screen size range supported (1-Small, 2-Normal, 3-Large and 4-Extra Large);
3. last 3 digits represent the actual version id.

6.4 Conclusion: Usability, Good Structure and Availability

In this chapter we followed the development of the PTSense mobile application from its early beginning. This application was prototyped using very accurate tools to design a close to final visual interface, providing an almost real experience of interaction with the interactive prototype. The tools used were Photoshop CS5 with a set of template tools from the Android ICS, that allowed to create the customized interfaces, one by one; and Apple's Keynote to make an interactive PDF with hyperlinks over the buttons. When visualized on the smartphone's screen, it simulates the real application. This prototype was used for the Usability Test, discussed in the next chapter. The choice of the Android ICS relates to the powerful sensor capabilities it provides and also to create a current and elegant commercial-like application that would attract user's attention and willing to use.

The architecture of the PTSense application relies on the combination of Background Services and the User Interfaces. These services run in the background letting the user keep focus on his primary activities while providing him a shortcut, i.e. notifications, to interact with the application whenever he wants or is needed. The collection of data from the sensors and the transmission of data to the server are the two most important tasks running as services. The first, however, is able to communicate with the interface by broadcasting intents with the sensor data collected. It was also presented the connectivity algorithm to transmit the data to the server.

Section 6.3.3.1 discussed the details of collecting the information for every sensor, while in Section 6.3.3.2 it was described the procedure to completely review a trip. Usability concerns were a constant throughout the development of this application and in this module they are easily perceived. The use of notifications for trip review's requests and the visual interface for the correspondent detailed views provide a simple yet useful information to the system. The interface relies on dragging sliders to classify stimulus and on quantitative, qualitative and color classifications for the predictions.

The user model used to identify the emotional state of the user is based on the Valence-arousal model that, with the classification of the environment conditions provides the useful information to train the personal model in the cloud. This provides the C2B with enough data to understand the relationships between affective and context variables, as proposed in the design phase of the project.

The application is not completely finished and the missing features are discussed in the Future Work section (Section 8.4). However for testing and publishing purposes they were filled with a short description of its future functionality. These gaps support the future developments, which was strongly taken into account during the whole application architecture. To test, we also provided the backward compatibility to older versions, as seen in Section 6.3.4.

Finally, the application designed and implemented during this project achieves its objective of building a solid architecture and design for the upcoming work. It even goes beyond the goals, which covered a simple prototype for data collection. The advances led to a solid smartphone application that could possibly be used in the future by the urban community. It provides a seamless integration with the proposed pervasive architecture, specially its communication with external sensors and the communication with the C2B platform. The last is already correctly implemented and tested. Finally, this proof of concept provides an elegant solution to collect the sensor data from the smartphone, the other objective for this project, which is covered in the next chapter.

PTSense App

Chapter 7

User Testing

7.1 Introduction

Testing a product is the first step to improve it. It serves to validate the implemented features or to identify the problems that need to be addressed to make it better. As a proof of concept, this project's goals need to be tested in the urban environment, bot only to verify its feasibility but also to stablish a solid ground for future work. Given our user-centered approach it is important to assess the final consumer's opinion during the whole development process. For that, two tests were conducted in different stages of the process: a Usability Test, in the beginning, with the prototype developed, and a Feature Test, in the end, with the implemented PTSense application.

In this chapter we are going through these two tests, discussing their purpose, the participant selection, the actual test procedures and the results obtained. Both these tests provided interesting results that are also discussed here.

7.2 Usability Test

A Usability Test is a technique used in user-centered designs to evaluate a product by testing it on users. It focuses on measuring the product's capacity to meet its intended purpose, giving direct input on how real users use the system [Nie94]. By understanding how users understand and use the product or application the developers can make an informed evaluation and address eventual issues in later versions of the application.

Testing the application in terms of usability in its early stages is vital to the design and development process. The sooner the errors are found or the application's architecture reaches maturity the better. Usability testing gives the possibility to iteratively increment the design with opportunistic user ideas. Many organizations, though, skip usability testing, assuming that it is costly, time consuming and unnecessary. However, this is the best way to test the idea or even the prototype and reducing extraordinary costs of correcting errors in late phases of the project. Therefore, it is essential to give users access to early prototypes or proof of concepts and let them explore and test it.

There are various types of Usability Testing depending on the project development phase or the type of test object used. Low-fidelity sketches are easy to build and useful for brainstorming, while high-fidelity prototypes help identify details and a more realistic expectation, though they need more time to develop [WTL02]. This test was conducted during the design stage, where we wanted to assess the usefulness of the solution we envisioned and specially the mobile application prototype. In this phase, the main objective was to identify problems with the interaction of the application and also the understandability of its features. For this reason it was chosen a very high-fidelity, which would provide very detailed feedback and lead to an improved interface and higher user satisfaction. At the same time this prototype leveraged the PTSense implementation, as explained in Section 6.2.

7.2.1 Participant Selection

One of the main concerns when starting a Usability Test is to determine the number of participants needed to collect enough feedback and data to be able to detect a significant number of errors. Jakob Nielsen, known as the father of Usability, refers that elaborating heavy usability tests is a wasteful of resources. The best results come from testing no more than 5 users and running as many small tests as you can afford [Nie00] (see Figure 7.1).

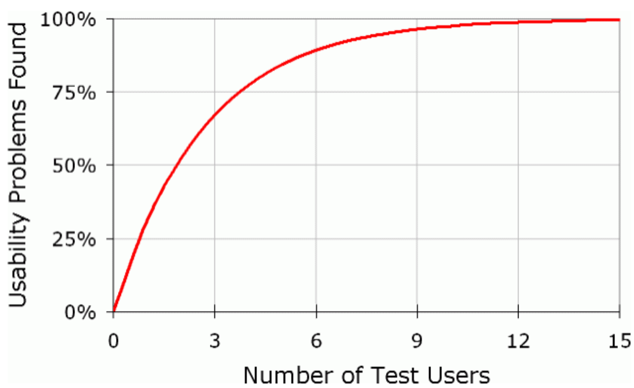


Figure 7.1: Number of users vs. usability problems found (5 is enough) [Nie00]

Following Nielsen's research we then selected 6 participants from the London metropolitan area using the social networks. They were all users of Public Transportation and represented potential users of this application.

7.2.2 Procedure

In order to get the best results out of a usability test, the first is to ensure that the best possible method for testing is used. Generally, the best method is to conduct a test where representative participants interact with representative scenarios. Therefore, we chose a set of representative participants, created a realistic prototype and defined a testing process with several scenarios, which

	Age	Profession	Nationality	PT Use	Choosing Criteria	Smartphone
1	22	IT Analyst	Portuguese	Tube, bus, river bus	TD > NC > WT > TT	Yes, Android
2	24	PhD Student	Greek	Bus	TD	Yes, Symbian
3	36	Researcher	Spanish	Bus, metro	TD > NC	No
4	27	CEO	British	Train, tube, bus	TD > E	Yes, iOS
5	24	Student	British	Tube, bus, rail	TD > TT > WD > NC	Yes, iOS
6	28	PhD Student	Greek	Tube, bus	TD	Yes, iOS

TD: Trip duration; NC: Number of changes; WT: Waiting time; TT: Type of transportation; WD: Walking distance; E: Efficiency

Table 7.1: Usability test participants with some demographic and context information

can be found in Appendix B. This scenarios would cover all the main features of the application through real life situations.

The test consisted of the several steps:

1. greeting and introduction to testers;
2. inform participants of the purpose of the test and its procedure. This information was read to the participants that thereafter signed a Consent Form (see Appendix C);
3. completion of a pre-test questionnaire in MS Excel, covering demographic and context related questions (e.g. use of smartphone, use of public transportation) (see Appendix D);
4. small introduction to the prototype's home screen and Android ICS;
5. execution of the test cases and scenarios from the test script;
6. completion of a post-test questionnaire MS Excel, covering understandability of questions, easiness to achieve objectives, and opinions (see Appendix E);
7. final remarks, thank the participants and offer of a small reward.

The practical test phase was recorded, which as stated in the consent form, including voice, mouse movement and clicks. Also the duration of tasks was recording using a stopwatch. The environment with which the participants interacted is shown in Figure 7.2. During the execution of the test cases the participants were asked to talk aloud their thoughts and justify the steps. The tester only participated to clear and further explain some cases, when asked, and encouraging the dialog to understand the participant's thoughts. Following usability testing guidelines, the tester was impartial and explored all the user's thinking.

The test was conducted on the 1st and 2nd of April, in the Imperial College London with the 6 participants shown in Table 7.1. I was the tester, with the supervision of Pedro, the developer of C2B. The language was English with the exception of one participant, with whom was in Portuguese.

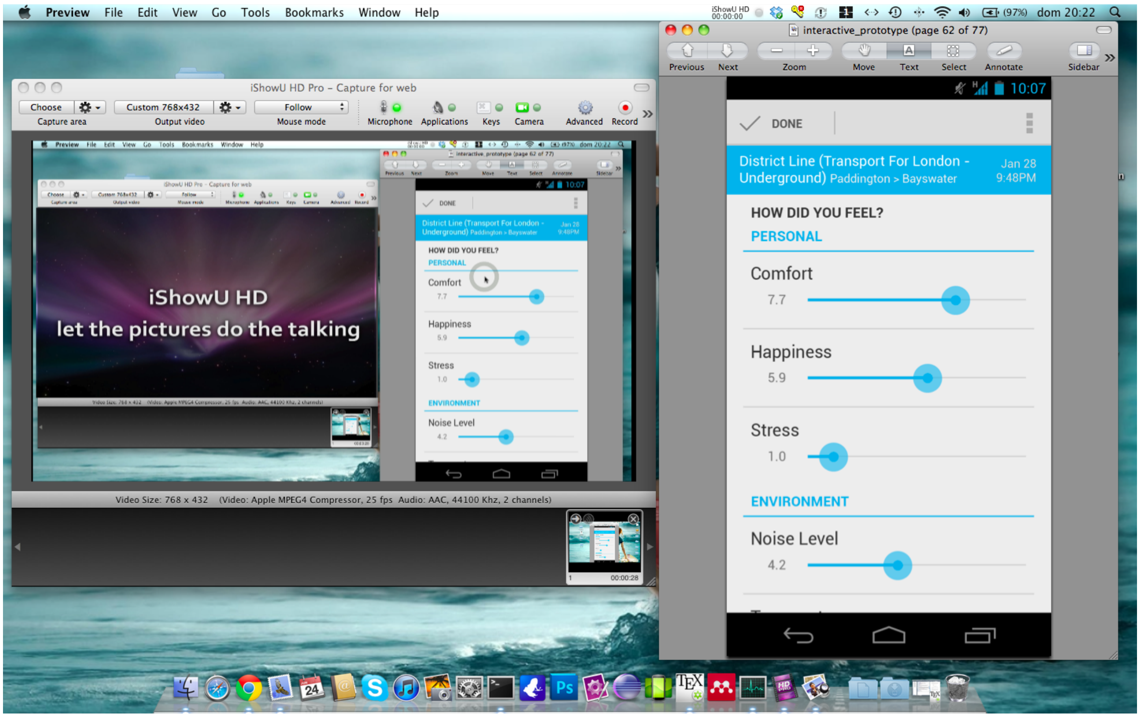


Figure 7.2: The recorded test environment consisted simply of the prototype (on the right)

7.2.3 Results

Since the main goal of this test was understanding how users would interact with the application, the results were mostly taken from the actual interaction with the prototype and the respective recordings of tasks. These were reviewed and from the dialogues, time spent and mouse clicks the relevant notes were taken regarding the user's concerns, doubts, suggestions and problems.

In general, we noted that all the participants completed the test without any major problems. Also, we realized that the understandability would improve with the time spent navigating in the application with the help of real life scenarios, i.e. in the first tasks they would take more time while in the last ones the navigation it would be almost automatic. This is understandable since the first moments are usually spent in understanding the architecture of the application and what are the options available. Some of the participant's concerns were also related to the Android design guidelines on the new version of the OS, specially the context menus, the action bar and the back or up buttons. This is due to the fact that any of them has used Android ICS applications before.

Nevertheless, it was possible to find some common mistakes and suggestions from the participants regarding the interface. For example, situation 1) in Figure 7.3 shows that initially the Start and Stop buttons expressed the state of the application (e.g. Started or Stopped Sensing) with inverted colors, i.e. red when the user had to start and green to stop, which tricked the participants. The solution, changed in the Android implementation, was to make it reflect actions (e.g. Start in green and Stop in red) and giving a more button-like feel. Still in situation 1) it is shown that the second option's text changed from ”Trip History” to ”Trip Reviews” as suggested by a

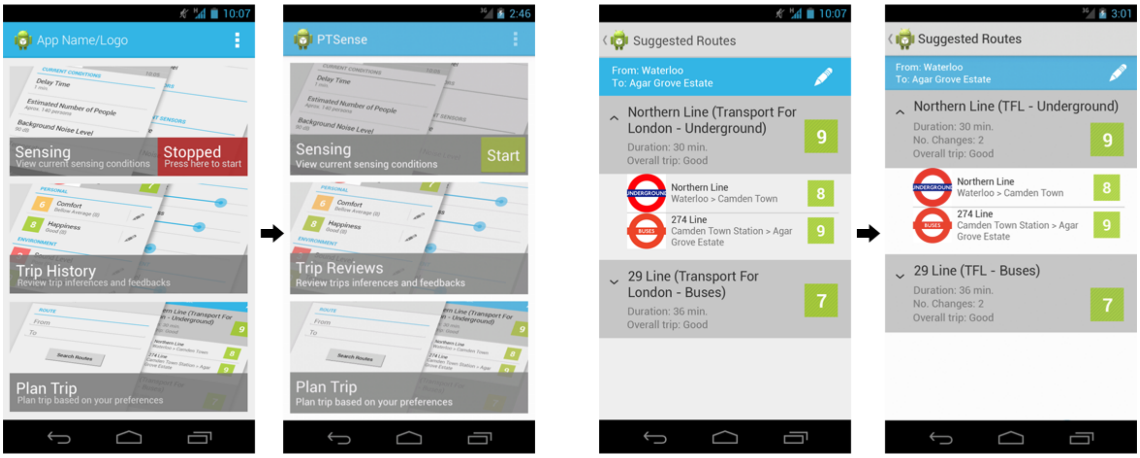


Figure 7.3: Changes in GUI derived trom the Usability Test: 1) Home, 2) Alternative Suggestions

participant, because the word "History" could mean only previous reviewed trips while inside that option there are trips to be reviewed. Also, the same participant, noted that feedbacks are in fact reviews. Hence the name change.

Situation 2) presents a designing flaw that some of the participants found: in the list of toggled trips, when it was minimized the user could not see the number of lines/changes, having to expand all the trips to correctly view all the journey plan. The solution was to add the number of changes to the header of those toggle views, as suggested by the participants.

Finally, we also added a button to add the possibility to change trip information during the journey, because the commuter is either busy when entering or leaving the vehicle or because he can change ideas during the trip.

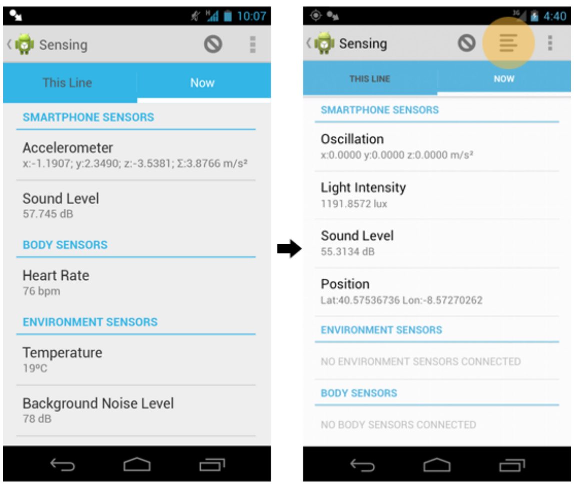


Figure 7.4: Adding the possibility to change trip information during journey

Two of the participants were also confused about the meaning of the numbers and colors in the inference screen, while the rest understood it completely at start. Nevertheless, in the end or after some explanation they realized its meaning and said it makes sense. Thus, no changes were made in this case.

From the pre and post-test questionnaires, we could also take three other relevant conclusions:

- the majority of the participants already used some smartphone applications for journey planning, specially for timetables, service maps, times for next services or view journey alternatives;
- all of the participants were just Satisfied, i.e. classified 2 out of 5, about the public transportation service in London;
- all of the participants Agreed, i.e. classified 4 out of 5, that the tasks were clear and the application helped to complete them. Also they Agreed they understood the application and it was easy to navigate in it.

From these tests it was concluded that the application appealed to the participants and helped them to complete the tasks. In general, the participants were satisfied with the idea and hopping to see it in the market, or more of its kind. In technical terms it allowed to address some aspects of the interface - some of them design flaws or errors and some were new ideas.

7.3 Feature Test

The Feature Test is a validation of the features implemented in a specific version of a product, i.e. if it provides the expected functionality. As the Usability Test, it is important to make it as early as possible and throughout the development, in order to identify and correct errors without any extraordinary cost. As so, following the goal to build a solid base for further development, the Feature Test was preferred to continuing the development to other phases. This test then covers the first stage of the implementation - the collection of data from the smartphone sensors. It puts the implemented Android application to test with real PT users on their daily lives. The test was conducted in Porto, for the duration of two weeks.

In this chapter we are going to discuss the selection of the participants and the procedure of publishing and monitoring the data. The user manual documentation to support testers will be briefly discussed followed by the results obtained.

7.3.1 Participant Selection

The participant selection phase in this test was slightly different from the previous and followed a more rigorous process:

1. it was created a group on the social network Facebook entitled "Quality of Experience in Public Transportation". People, friends and strangers, from the UK, Portugal or other places, were invited within the application or by email. This group reached 650 people;

2. a survey on QoE, elaborated on the context of the C2B project, was publish in this group for the people to respond. From the 650, 170 responded;

3. from these 170 participants, we used the results to select 20 people for the test following a not so restrict criteria: users of public transportation in Porto, with Android smartphones and different age groups and mobility patterns;

4. a confirmation email and test procedure explanation was sent to the selected participants from which only 10 replied affirmatively and participated in the test.

7.3.2 Procedure

The test procedure, that started with the participant selection was, as stated, transmitted to the participants via email. To start, the application was published online on the Android Play Store as discussed in Section 6.3.6. The instructions to download and install the application from this source were explained in the User Manual (see Section 7.3.3) which was sent as an attachment in one of the emails.

Another setup issue was to keep track of who is participating and to address individual problems. For this, we had to match the participants to their smartphones, since the data received in the server comes only with the smartphone ID information. Therefore, the participants were asked to do a dumb test journey and insert in the comments section of the feedback their unique user identification that was individually provided through email. This way we could know that user with $u\_id = PTSense\_Exp\_1$ was making the trips with the smartphone with $s\_id = 357191040087682$. This procedure was explained both through email and also in the User Manual.

After the initial setup the users had a day to try the application and ask questions to the developers or report problems. During this time, some problems regarding older Android versions were reported and corrected in newer releases until the application was ready to be used by everyone. During the test, from the moment it started until the end, the developers did not interfere in the process. The results were simply received and monitored on the server side, accessible online at:

- http://cloud2bubble.appspot.com/api/data/sense?date=1-06-2012 , for the sensing data received on June 1st, 2012, for example;

- http://cloud2bubble.appspot.com/api/data/feedback?date=1-06-2012 , for the feedback received on that same day.

After the completion of the test, all the results for each day were converted from JSON to XML and imported to an MS Excel worksheet where they were later treated (see Section 7.3.4).

7.3.3 User Manual

The User Manual for Tests, found in Appendix F, is a support tool for testers/participants with instructions and guidelines on working with the application and correctly complete the test. It provides relevant information for the specific test and product version to all the participants. There they could find information on:

- how to download and install PTSense on their smartphones;
- an overview of the features available in the test version;
- a practical example of its usage;
- how to start the test;
- and finally contact information for providing feedback on errors, suggestions or questions.

7.3.4 Results

The results for the two weeks of user travelling and sensing were saved in the server database in its original format of JSON in the public places indicated in Section 7.3.2. After importing all the information to a workbook we were able to filter the important data given the setup process defined. The first trip from a new smartphone id in which the feedback comment had a user id indicated a new participant in the test. By having all the device ids associated to participants, it was possible to filter all the information - sensor data and feedbacks - from those ids, as shown in Figure 7.5. The selection headers, as seen in the same figure, were created automatically when importing the XML files.

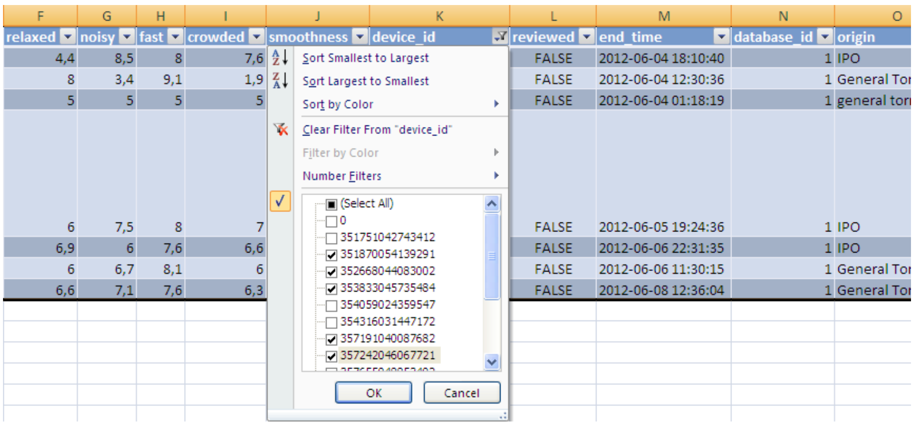


Figure 7.5: Filter headers in Sensor data and Feedback data help to filter information by device and journey

To get the information related to a specific user and trip, the end time, which is unique, was added to the device filter. Filtering the monstrous sensor data and feedbacks with the same filter (device and end time) it is possible to match the affective and context information, as desired. For each journey we aggregated the feedback classifications with the average of the sensor data throughout the trip, as shown in Figure 7.6.

When all the information for each individual and each trip was grouped, we were able to take some early conclusions:

	Transport	Reliable	Ambience	Happy	Relaxed	Noisy	Fast	Crowded	Smoothness	Sound	Acceleration
trip1	Metro do Porto	8,5	2,3	4,2	7,7	4,1	4,4	7,8	7,7	64,34340	48,81674
trip2	Metro do Porto	8,7	3,5	6,7	5	8,4	3,9	7,5	7,3	63,36623	49,77407
trip3	Metro do Porto	8,2	6,4	4,9	4,7	4,6	6,3	3,7	7,8	60,49086	50,12936
trip4	Metro do Porto	7,6	2,1	7,7	6,6	4	6,5	6,7	4,2	66,77927	47,72006
trip5	Metro do Porto	9,8	6,8	6,7	2,8	7,4	4,5	2,4	4,7	59,93356	49,66536
trip6	Metro do Porto	7,8	2,1	7,6	3,5	4,7	2,6	6	7	67,88399	47,08543
trip7	Metro do Porto	6,3	5,8	4,1	8	3,4	3,7	2,4	7,5	66,44577	50,54641
trip8	Metro do Porto	8,2	5	8,9	7,9	1,4	6,5	2,2	8	68,54195	50,26483
trip9	STCP	1,7	8,3	7,2	7	2,2	6,8	6,1	2,7	68,84926	49,83132

Figure 7.6: Groupping average sensor data and feedback classification for each journey

- 50% of the smartphones only had 2 out of 5 useful sensors (Accelerometer and Microphone);
- the number of trips was surprisingly low, with only 3 participants making more than 10 journeys with useful data. However, one of those reported a significant amount of 34 trips;
- we were able to identify daily routines based on routes and time.

From this grouped information we traced the correlation graphs between some of the variables for all the trips, either matching sensor data with feedback variables or between feedback variables. We also calculated the linear correlation coefficient, R, and coefficient of determination, R^2.

R, sometimes referred to as Pearson product-moment correlation coefficient, measures the strength and the direction of a linear relationship between two variables. In a linear relationship, two variables vary proportionality, i.e. if values for x increase then values for y also increase or decrease, in a positive or negative correlation, respectively. The closer R is to 1 or -1 the stronger the relation is. Given the social sciences and human behaviour context of the data collected in this project, a value less than $0,3$ denotes a weak correlation, around $0,5$ represents a moderate correlation and above $0,8$ a strong correlation [Mat].

R^2, is useful because it gives the proportion of the variance of one variable that is predictable from the other variable. It is a measure that allows to determine how certain one can be in making predictions from a certain model or graph. These are the type of models that are used in the inference module in C2B to predict the certain variables from other known variables. "The coefficient of determination is a measure of how well the regression line represents the data. If the regression line passes exactly through every point on the scatter plot, it would be able to explain all of the variation. The further the line is away from the points, the less it is able to explain" [Mat].

The results shown in Figure 7.7 represent this correlations with the calculated R and R^2 parameters.

From these graphs we could already identify some signs of moderate correlations, given by the distribution of values along the regression line. Although they are not very strong, probably because of the low number of trips and feedbacks or not so reliable/serious feedbacks, we believe that with more time they would turn more solid. Nevertheless, we can already identify some patterns:

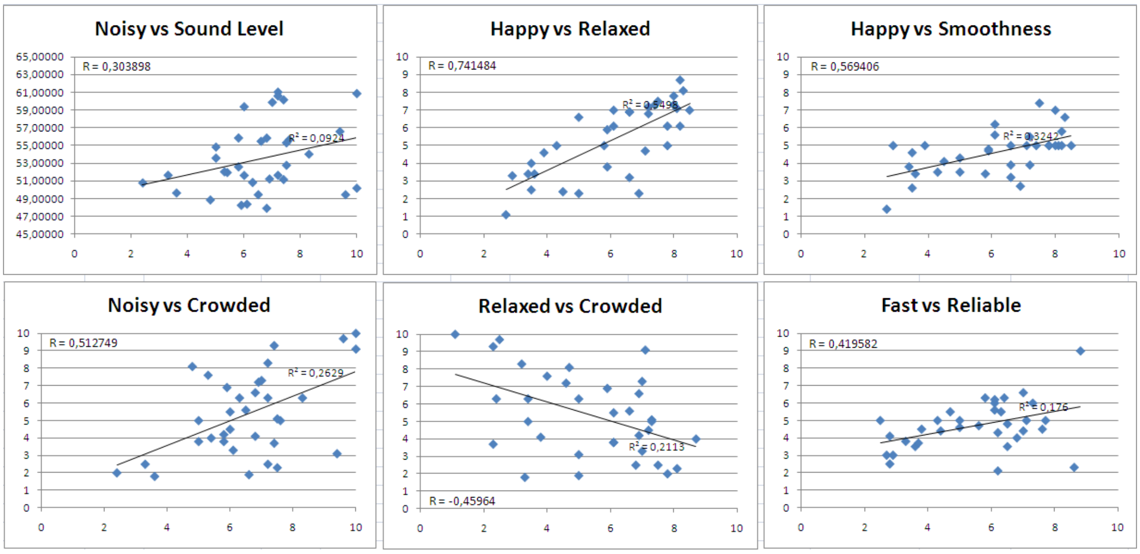


Figure 7.7: Results of the Feature Test show moderate correlations but low determination between variables with two week data

- the only correlation we find with sensor data is the relation between the noisy feedback and the sensed sound level. Even though the correlation coefficient is lower than the rest, it represents a good sign or potential correlation. It means that with a low certainty we can predict a service is noisy from the sound level data received;
- feedback variables alone provide some data for modelling the profiles. For example, we can guess that the user is happy when the trip is smooth and therefore relaxed, or that normally when the service is fast it is also reliable. These correlations have good coefficients, both R and R^2;
- the crowded variable influence noisy and relaxed classifications in different ways, as expected. From the graphs we realize that the more crowded a vehicle is, the more noisy it will be, and on the other side, the less relaxed the user is.

7.4 Conclusion: Validating and Improving Solution

Testing is an important phase in Software development that many people and companies forget. It reduces changes of errors in the public release version of a product and more importantly the cost of correcting those errors in later phases. By testing and validating throughout the project we were able to improve our initial prototype into a more solid and useful product and validating the features implemented in the PTSense application.

The Usability Test permitted to collect data on the participant's success, speed of performance and satisfaction when interacting with the prototype. The findings gave us (designers and developers, at the same time) a broader understanding of how ordinary people would use the application, which helped change some of the application's interface and design. In general, the participants did not have problems with the application, were satisfied with the appearance and even interested in the future developments. Their statements showed that the introduction of such application in the mobile application market for public transportation would greatly improve the experience of commuting. The current condition aspect is extremely useful and the way the application collects data and shows predictions and suggestions to the user is interesting.

The Feature Test, in a more restricted process, provided the information needed to identify the correlation between variables of the sensed data and the feedback classifications. As one of the main goals of this project, this test was successfully concluded without problems. The results, filtered from the huge amount of collected data, showed signs of moderate correlations between sensor and feedback variables and also just from the feedback. Nevertheless, the coefficient of correlation, R, and the coefficient of determination, R^2, calculated from each user data set and particular variable relationships, are low to medium. These were expected to be higher, meaning stronger relationships, but we suspect that with more time and readings, they would become stronger, as thought in the early phases of the development (see Section 5.4.2).

In the end, adding testing phases to the development process showed to be effective and useful. The initial objectives only covered the Feature Test, although the introduction of the Usability Test provided a much more solid approach before the implementation phase. Other than the effects to the solution, these tests also let us, developers of the PTSense and the Cloud2Bubble, practice and gain more insights on formal testing with ordinary users and in different environments - London and Porto.

Chapter 8

Conclusions

The new emerging concept of the HCI, where pervasive computing will invade our daily lives, is growing solid and many believe this is a new era for the internet. Many advances in technology are pointing in that direction, as seen in the Literature Review (Chapter 2). IPv6, ultra low power wireless communication, almost invisible sensor nodes, the definition of complex sensor networks, processing power in mobile devices, specially smartphones, casual yet effective wearable computers and adoption of cloud computing are some of the signs converging to the Internet of Things concept and the design of Smart Mobile Applications for a smarter world.

In this project, given both goals of implementation and research, the design of the solution followed, since the beginning, this new trend. The use of sensor networks would suffice the objective of collecting affective data and at the same time pushing mobile development into newer grounds. Using smartphone sensors is not new, neither cloud computing based mobile services, but its a new emerging powerful architecture that allied with a WPAN for multi-sensor multi-modal data collection and an Opportunistic Sensing architecture gives this project an innovative aspect. This design was first introduced in a paper presented at the SmartApps'12 international workshop in the Pervasive 2012 conference, in Newcastle (Appendix A). Another accepted poster is yet to be presented at the MobileHCI 2012 conference in San Francisco, in September of this year. These publications suggest that this idea is proving to have potential in the short-term and at an international level, alongside with the biggest innovations in the scientific community.

The solution envisioned is ambitious, meaning it is large, somewhat complex and still dependent of other devices and technologies. Because of time and technology access constraints, this project is not yet finished, as it was also not supposed to be. Its main purpose was to build a solid architecture to support the collection of Affective data in public transportation and a seamless integration with the Cloud2Bubble module that is being developed. This proof of concept was tested, either its utility and usability, with a positive feedback from the users. There is still a long way until it is finished, but it provides a good ground and support for the work to come. The innovative vision is on the edge of technology which means that future work most likely will not be outdated.

In fact, some of the missing work still needs further research and maturity of external technologies, such as physiological wearable sensors and Bluetooth Smart integration in more sensors and smartphones.

Regarding QoE, we could not make a complete assessment of this solution's benefits to either users or service providers, since the only feedback we had was a possible feasibility based on the theory. However, we believe that when it is finished and we have feedback from the service providers, it will prove to enhance the quality of the experience for the users, by adding current condition's classification based on personal preferences to the choosing criteria; and on the other side, to service providers, it would give a better understanding of the users preferences and allowing them to re-arrange or re-route people to less crowded services, providing easier and more comfortable rides.

8.1 Work Summary

The work done can be divided in five main steps: the research of the state of the art; designing the solution; prototyping; implementation and testing.

The research, presented in Chapter 2, was the phase that consumed more time given the multidisciplinary aspect of this work and the large quantity of material on this area. From this research and previous related work on the cloud module we envisioned a solution based on the loose ends of the state of the art while at the same time integrating seamlessly with the C2B project in a SOA web service using the cloud computing capabilities (see Chapter 4). This architecture was based on the collection of affective and context data from smartphone sensors, static environment sensors and user body sensors. This information would be gathered in the smartphone and transmitted to the server application responsible to generate models of preferences for each user using machine learning mechanisms. Later it accesses real-time public transportation information, matches against user routines and preferences and transmits a notification to the user's smartphone with better alternatives when there are. This provides the architecture and specification to the Mobile Sensing Platform, discussed in Chapter 3 and Chapter 5 and easily summarized in Figure 5.3.

After the solidification of the architecture, a very realistic prototype of the Android application was built, following the Android design guidelines. It was an interactive PDF that, when viewed on the smartphone's screen, simulated the real application. Its purpose was to test the interface with ordinary users, which gave us useful information on understandability and appraisal to users. In general, it was easy to understand and navigate, but some concerns were commonly pointed and later addressed.

Following the prototype and Usability Test, the implementation phase took place. The PT-Sense Android application was developed using the new Android ICS version and following the UI of the prototype. The use of a notification system and background services gives the application a passive look, as desired. It is not meant to be a foreground application, such as games or calendar, but a background one, such as a music player. The implementation could not be finished completely given time constraints, but provides all the support and guidelines for the future work. However, we were able to test the collection of smartphone sensor data and emotional model based feedback and transmit it to the C2B for analysis. This test took place in Porto during two weeks, where participants used the application during their daily routines, reporting data constantly. The gathered information permitted to identify some moderate correlations between some variables, although we suspect they would get stronger over time, as envisioned. The prototype and PTSense implementation are covered in Chapter 6 while the test procedures and results in Chapter 7.

During this period, two publications were written and one was already presented. All the written deliverables are found in the Appendixes, with the exception of the interactive prototype and PTSense application, which are provided as deliverables with this document.

8.2 Objective Satisfaction

All the goals presented in the Introduction (Chapter 1) were successfully completed and even exceeded. As said, this project consisted mainly in designing a proof of concept architecture and mobile application prototype for assisting in the collection of affective data in urban public transportation. The final pervasive solution and PTSense application surpasses these goals and defines a commercial-like smart application, fully integrated with the C2B project, that collects affective and context data and delivers a personalized service to the user, than can, in fact, improve his QoE in riding public transportation.

Starting the project from the research proved to be a good choice. It allowed to gain knowledge of the technology tendencies and latest developments. From research conducted and presented we realized that the current state of the art in a wide range of fields of study does not provide a standard solution for our problem. Instead, it gives us new isolated directions that can be integrated into the framework being developed. Focusing in the collection of affective and context information we envision a new ubiquitous experience based in wireless sensor communication centered in the user's smartphone. Computation is moving to mobile devices integrated in every days objects giving transparency to Human-Computer Interaction in a more fluent unobtrusive user experience as we pursue.

Without this first step the design of the architecture would not definitely integrate sensor networks and wearable sensors, for instance. The research on improving the solution continued with the addition of the Bluetooth Smart communication between devices, Opportunistic Sensing using sensed data to inform other users and the mechanism to start and stop sensing automatically. Some of this research is not yet implemented but is addressed in this document. The introduction of new Information and Communication Technology (ICT) developments to the design makes it a forward-thinking solution that is capable of following the edge of technology. There is no other mobile application that collects data from wearable sensors or other sensors for the sake of QoE. Even less making that data available in the cloud for other users and building prediction models based on the collected affective and context data. This research applied to the architecture gives the possibility to add personalization based on individual preferences to the choosing criteria when choosing amongst PT alternatives. Thus, providing major contributions in the design and specification of a new distributed sensing paradigm for smart mobile applications.

The other two goals were the implementation of a smartphone application prototype and its test. The first combined additional steps that define the Rapid Application Development process used. Before starting the implementation of the Android application, the prototype helped to design the UI of the application and the usability test to validate and improve it. The implementation of the application following the concept of the prototype and we were able to successfully build a mobile application capable of collecting affective and context data, based on the collection of sensor data and user input. The information collected from this application is useful for defining the user profiles in the cloud, complementing the QoE enhancing framework.

The last objective was to test the developed smartphone application in a controlled public transportation environment. This was addressed in the User Test chapter (Chapter 7), where we describe the whole procedure and results achieved. To the required Feature Test another Usability Test was added. Both tests were complete successfully and provided useful results for the maturity of the application.

8.3 Impact on User Experience

The first objective of this project was to build a solution capable of assessing the QoE in urban public transportation so we could understand the influencing factors and improve them, creating an overall better experience. The objectives were completed and we believe the final solution has the desired impact on the user experience, both creating value for the PT providers and for the commuters themselves.

8.3.1 For Commuters

During the whole project the user had a central focus in the design, implementation and test of the solution. We can define the final solution as a User-centered Design given the importance in creating an application easy to use, understandable and following Jakob Nielsen's usability principles. For this purpose we crafted the application around a very simple interface where all the key features are easily accessible from the main screen. Similar features are also grouped in tabs and processing services run in background. All this design concerns address 3 out of 5 usability principles: Learnability, Memorability and Efficiency [Nie94]. The choice of the quantitative 0-10 scale, qualitative classification and colors to classify stimulus or predictions addresses the same principles, plus the Satisfaction principle of the users when interacting with the application. Finally, during the implementation of the PTSense, error detection and debugging had an important part which covered the last principle of reducing the number of Errors the user would encounter. Addressing all this principles gave the final application a user-friendly interface and interaction which greatly improves the possibility of mass urban adoption.

Asides the usability, the features lying beneath the interface are what, in fact, improve the user experience. By creating user profiles based on preferences we can serve each commuter individually. Adding how a particular route pleases a user better, can ultimately allow him to choose one alternative over another. Over time, this suggestions get more certain and can greatly benefit the user's journey planning. Again, we remember that nowadays the solutions available do not take into account the conditions of the service lines and or the user preferences. In the Introduction we presented (Figure 1.1) one example of what this solution pretends to achieve and the PTSense, when completely finished, will provide this same subjective information (Figure 6.11).

8.3.2 For PT Providers

This solution also brings value to PT providers. To start they get to understand better their user's preferences. For example, if a morning bus normally is full of young people, whom typically do not care about noise or oscillation, and normally that is how the service is, then the PT provider perhaps does not need to worry or change that situation. However, if there is a morning metro full of people that go to work and need to be on time or to be focused, meaning they do not like crowded or noisy services, then maybe it would be better to route some of that people to other less crowded services. This, in fact, leads to a better service for the PT providers while at the same time making a better distribution of the people between the services. We can then say it adds value to PT providers by improving the customer's satisfaction and experience, which is what a service generally aims at.

8.4 Future Work

There is still much development to be done, specially regarding newer technologies that are just now getting mature. As shown in Figure 8.1, technologies and ideas like the Internet of Things, Context-Enriched Services, Cloud Computing, Machine-to-Machine (M2M) Communication Services and Mesh Sensor Networks will take some more years to reach maturity. Nevertheless, the early adoption of this architecture is what makes this project survive in the future and possibly expand.

The development is always being done on both sides of the project: the mobile application and the C2B. Regarding the mobile application there are still 3 missing parts:

- Finish the PTSense interfaces with the system inferences, profile and plan trip features, while the machine learning and prediction mechanism is implemented in C2B;
- Add external sensors, the environment sensors and wearable sensors, using the new Bluetooth Smart technology already available in some sensors and some smartphones. At the same time, in the C2B it will be implemented the model to identify emotional states based on physiological signals;

[1] Copyright 2011, Gartner, Inc.

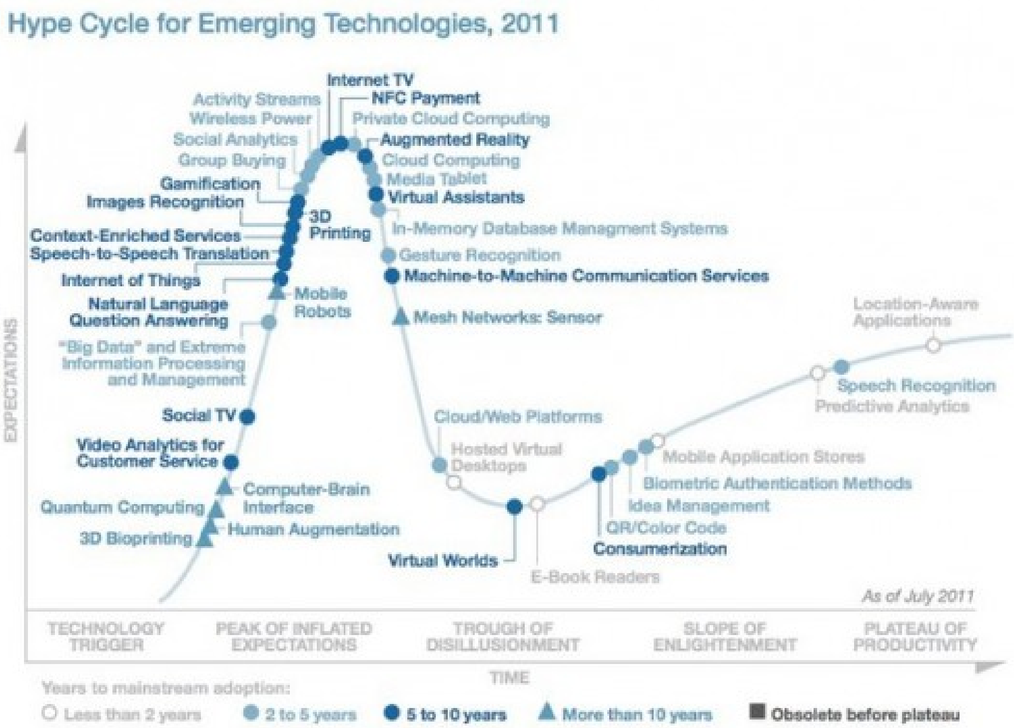


Figure 8.1: Gartner's hype cycle for 2011 shows that covered areas will need years to get mature[1]

- Define the Start and Stop sensing mechanism to start sensing automatically when the user enters the vehicle and stop when he leaves it.

To complete these steps, some more research will have to be done, specially on the physiological signals matter. Another constraint is that most of the innovation that we look forward to access is being conducted specially in companies rather than in an open source environment. Nevertheless, there are some wearable sensors that provide and open API based on the Bluetooth Smart technology that we can use. Finally, the current public transportation test environment lacks infrastructure details used in this architecture such as static environment sensors or mobile validation in stops or vehicles - a possible solution for the third development step above.

8.4.1 System Inferences

This is were the PTSense implementation stopped. To provide the application with the capacity to show predictions and classifications it is first necessary to work on the machine learning and prediction mechanism in the cloud. While it is being developed, the interfaces just need to be coded, since the space for this feature already exist and the prototype interfaces as well, as seen in Figure 6.8. It is also missing the interfaces for the profile and plan trip (Figure 6.10 and 6.11, respectively). When all the UI is done, it is needed to complete the transmission protocol that was left half done. Further development already have the space saved to implement this missing features.

Objectively, is missing:

- when the trip finishes and sends the sensor data, parse response from the server that will tell if it has a prediction or if it needs a feedback. As of now, the server is only answering with and OK code and a feedback is requested to the user;
- if the above server answer is a prediction, start a notification for a system inference review. This includes the system inference screen and list interfaces;
- add inference review changes to a database table and add the information on this table to the transmission algorithm/mechanism already defined;
- create the profile and routines screens and define the synchronization protocol;
- create the alternative suggestion pop-up and interfaces from the information received from the server, parsed from JSON.

When this steps are completed, the application is finished. The next steps are improvements on the efficiency, accuracy and usability.

8.4.2 External Devices

Connecting external devices represents the definition of the interaction protocol between all the sensors connected to the smartphone, either from the environment or wearable computers. This transmission has to be distinguished between the two types of sensors.

Wearable sensors are personal sensors. They are supposed to be paired with the smartphone, similarly to what Wahoo Blue HR [Fit] does, for example. As of now, only the iPhone 4S, the Motorola Droid RAZR and Samsung Glalxy S III are Bluetooth Smart Ready [SIG]. However, this is expected to increase as well as the number of wearable sensors supporting this technology. The physiological sensor that would better fit this solution is the Basis B1 watch [Sci] which reads HR and GSR and was lately officially reported to be supporting wireless communication over Bluetooth and an open source API. The data to be received from these sensors has already a reserved space in the PTSense, missing only the transmission protocol.

Regarding the environment sensors, it is different. They are public sensors that broadcast information to the smartphones inside the vehicle. For this, we believe a Wi-Fi data transmission is the more correct approach, similar to a normal days router. The space for the data collected is also reserved, missing the implementation of the transmission mechanism.

In parallel, in the C2B platform we will need a model to identify emotional states based on the physiological signals. There are already some models, but this will still need more research for reliable and accurate emotional state recognition.

8.4.3 Start & Stop Sensing Mechanism

Finally, for enhancing usability, we want PTSense to start and stop sensing automatically when the user gets in or out of the service. For this, we propose two solutions:

- make the environment sensors send a packet or some sort of intent to the smartphone (similar to the information it receives when there are free wireless networks) to start the sensing service when it gets inside the service;
- use a mobile validation system already used in other countries [Gem08] that would tell the smartphone the user is inside the station or vehicle and should start sensing by itself.

In both alternatives the application receives the information regarding the service, line, start and stop stops and time from these transmitters, removing the burden from the user of inserting the trip information at all times. The application then runs totally in the background, only interaction with the user through the notification system and pop-ups. This is the final step and goal of the pervasive solution, allowing to have a passive system that users do not notice most of the time, but is extremely useful when alerting about better routes.

8.5 Final Words

In a recent Google I/O talk about Smart App Design[2], Travis Green, Product Manager at Google, defined smart applications as "things that automate the repetitive, extract the essential and recommend the useful". So we believe that the importance of this work relates to the conception, design, implementation and test of a smart mobile application supported by a cloud computing platform that provides personalized services to customers. Following that definition, this solution is a smart application since:

- we collect data from sensors, and generate a big amount of live data from participatory sensing and web services in an autonomous way;
- then we extract the essential by our strong analysis in the cloud;
- and finally we move planning and personal thinking to the application, by recommending the useful, simulating a personal advisor.

Smart applications are built to create a smarter world. This multidisciplinary work helps the development a new paradigm of HCI where pervasive systems have the capability to collect sensorial data. Understanding this data and transforming it into the knowledge of routines, mobility patterns and personal preferences creates an intelligent system to enhance the quality of the user's experiences.

Ideally, even in an theoretical point of view, this proof of concept will conduct researchers in the direction of ubiquitous wireless sensor networking through smartphones, the development of attractive wearable computers for urban user adoption and distributed sensing paradigms for multi-modal data collection. This work will provide concepts and insights that meets the concept of the future Internet of Things (IoT) where devices communicate in an intelligent way to satisfy human needs, not only in public transportation but in any kind of service.

[2] http://youtu.be/FJDP_0Mrb-w

Appendix A

SmartApps'12 Paper

Smart Mobile Sensing for Measuring Quality of Experience in Urban Public Transports

João G. Vieira, Teresa Galvão
João Falcão e Cunha
Faculdade de Engenharia, Universidade do Porto
Rua Dr. Roberto Frias, 4200-465 Porto, Portugal
{joao.guerra, tgalvao, jfcunha}@fe.up.pt

Pedro Mauricio Costa, Jeremy Pitt
Imperial College London
South Kensington Campus, London SW7 2AZ
{pm.costa, j.pitt}@imperial.ac.uk

ABSTRACT

In our current society which is characterized by mobility, individuality and comfort requirements, there is a need for real-time information and services that make people's life easier. In Public Transportation (PT), such data is crucial for enhancing the travelling experience. Lately, the number of smartphone applications for trip planning, mobile ticketing and validation has grown considerably, but they do not take into account the user's affective state or his Quality of Experience (QoE). Emotion, that is fundamental to the human experience influencing rational decision making, is rarely taken into account. Thus, in this work we present a new solution to find and understand patterns of user satisfaction based on the collection of affective and context information with and without direct user input. The current growth of ubiquitous systems allows the use of everyday devices, specially smartphones, wireless sensors and wearable computers to smoothly collect multi-modal sensorial data during trips. While the user input serves as a solid validation of the information received. A PT mobile service then uses these preference profiles to suggest more comfortable routes according to the user likings through a smartphone application. This adds a new domain of personal preferences to the usual duration, cost and number of changes in the transportation choosing criteria. In this work we present a mobile sensing platform based on the mobile application responsible for collection trip data and user input, and its communication with the reasoning system in the cloud - the Cloud2Bubble (C2B) framework.

Categories and Subject Descriptors

H.3.0 [**Information Storage and Retrieval**]: General; C.2.1 [**Computer-communication Networks**]: Network Architecture and Design; H.1.2 [**Models and Principles**]: User/Machine Systems

General Terms

Design, Experimentation, Human Factors, Measurement

Keywords

Mobile Computing, Affective Computing, Wireless Mobile Sensor Network, Context-Awareness, Mobile Sensing, Future Internet

1. INTRODUCTION

When one has to choose between services, providers or products he normally opts for the one that makes him feel comfortable, engaged or simply on time. While the quality of the service influences that decision it is also important to assess the perceived quality from the user point of view, i.e. Quality of Experience (QoE) [15]. In Public Transportation, trip planning applications supported by real-time dynamic routing algorithms combined with e-ticketing and mobile validation [12] are making urban travelling easier and more comfortable. However, individualization regarding commuters preferences and personal satisfaction is still missing. Which aside objective and rational criteria like duration or cost, also plays an important role in the selection of traffic means [5]. An intelligent system that understands what the customer prefers allows the delivery of personalized services. It removes the burden from the user to pro-actively select information in generic services making usability effortless. For example, in trip planning, showing first bus routes over metro because the user does not like confusion. As so, assessing and understanding emotions throughout the trips is an important factor that allows the offer of better services to users.

In this work, we introduce a new solution to assess personal satisfaction in a more autonomous, intelligent and ubiquitous way. We present a mobile sensing platform that feeds a web service with user emotional data and context-aware information. Changes in affective state are only substantial when mashed against context data such as activity and surrounding conditions and ultimately validated with the user rational perception. All of this data is collected through a smartphone application and transmitted to the cloud where patterns of user preferences are defined, used and maintained.

In the past decades the Human-Computer Interaction paradigm moved towards mobility and mobile services, which are slowly but steadily becoming a part of people's everyday life [17]. Today's smartphones are settling as the central computing and communication device in people's lives, but they also power a rich sensorial platform that has been used in a wide variety of domains. In our solution we use the set of smartphone sensors to collect context information, such as oscillation, ambient temperature, light and sound conditions, per se and/or communicating with surrounding sensing devices installed in vehicles. The interaction via low-power wireless technology such as Bluetooth Low-Energy (BLE), already available in smartphones, allows this collection of data to be done transparently and unobtrusively. On the other hand, emotion has also a significant impact on User eXperience (UX) [16]; the way customers feel in a trip can help to identify their likings. As so, an affective-oriented approach provides a better understanding of users in relation to their environments. Measuring affective data is done through a new generation of unobtrusive wearable computers that reads physiological signals [18, 2], later comprehended as emotions.

In sum, our solution defines a Wireless Mobile Sensing Network in which the central node, the smartphone, aggregates and extracts features from user and environment data. In addition, it acts as gateway to a cloud-based service, which maintains individual user profiles and offers personalised context-aware services to the customer's smartphone. The work presented here integrates into the Cloud2Bubble project, a framework that aims at enhancing QoE in smart environments, currently being used in the Urban Public Transportation context.

The next chapter introduces the Cloud2Bubble web service and its integration with the on-going project. Chapter 3 presents the sensing platform, while Chapter 4 covers the smartphone application in more detail. Chapter 5 concludes the paper with some words on future work.

2. CLOUD2BUBBLE FRAMEWORK

The smart mobile application presented in this paper is supported by a cloud-based infrastructure: Cloud2Bubble [6]. This infrastructure is capable of aggregating a number of data sources from both user and environment and generating user-tailored services with the potential of enhancing QoE. The Cloud2Bubble framework facilitates the development of system design and development in smart environments and it addresses the disconnection between user-generated data collection and personalised service delivery.

The dissemination of pervasive environments presents a number of opportunities, ranging from healthcare to retailing domains. Moreover, a survey on smartphone sensing [10] refers the usage of smartphone embedded sensors in areas like healthcare [8, 11], social networks, context-awareness [1, 14] and transportation [12]. Mobile information is used to measure some of the user's characteristics and context to promote well-being, social interactions, intelligent localisation or real-time information of services. In PT, trip planning applications supported by real-time dynamic information, buying tickets through the mobile and mobile trip validation [12] are making commuting easier and more comfortable. These services are, however, more concerned in enhancing the quality of service rather than the personal subjective quality of experience.

2.1 Interaction Loop

The interaction process that takes place between user and system forms a continuous loop of interaction composed of user-data collection and service delivery, supported by the collection of environment state. The stream of environment data is transformed into a representation of the environment state. This environment state is then aggregated together with the stream of user data, resulting in a unique user profile. This profile describes the environment conditions that correspond to positive and negative user responses. Such a profile is continuously improved over time, leading to the identification of increasingly accurate patterns of satisfaction and/or dissatisfaction. Individual profiles enable the delivery of user-tailored services, i.e. recommendations of suitable alternatives or notification warnings.

2.2 World Modelling

The world modelling is based on two main concepts: groups of co-located environment sensors, defined as sensor Cloudlets, and personal mobile devices capable of interacting with users, defined as virtual Bubbles. Cloudlets and Bubbles are organised in a hierarchical structure, reflecting their physical relationship, which is then used for environment state representation by the framework. Moreover, such a structure facilitates a systemic integration of existing technology and infrastructures already present in the environment, as well as scalability for future additions.

2.3 Architecture

This framework is currently under development alongside the mobile sensing platform. It combines a set of modules that facilitate the development of smart systems and enable a QoE-oriented approach:

- **Domain Management** – enables the dynamic modelling of an environment, including the relationship between different components, ranging from personal smartphones to environment-deployed sensors and third-party services;
- **Rule Engine** – implements an adaptive event-driven solution capable of processing a large number of changes in the environment and adapting itself to the requirements posed by different environment states;
- **Reasoner Module** – it aggregates different data sources into individual user profiles, which describe personal preferences. Such profiles are then used to take action in the environment, providing users with relevant information.

3. THE MOBILE SENSING PLATFORM

In this section we introduce the mobile sensing platform that collects affective and context data. The creation and maintenance of individual user profiles requires the collection and aggregation of all influencing variables in the perceived quality of every trip the customer does. For this purpose we use an Opportunistic Sensing paradigm [10] to autonomously collect this information from interconnected sensors. However, only this data might lead to erroneous evaluations: although emotions can help determine the satisfaction of a customer, they are also very complex and affected by multiple external factors, such as stress levels (e.g. because of a meeting), company or landscape. To overcome this situations, we also collect information from direct input for more correct readings.

The next sections cover the designing principles, an overview of the architecture developed and implementation details of its components.

3.1 Designing Principles

The key aspect of ubiquitous systems is integrating computation in human interactions everywhere. Invisibility, autonomy and mobility make these systems more prone to user adoption [19, 3] by not interfering in everyday human experiences. This is the idea behind the Web of Things concept [13], where devices intelligently communicate with devices to ease complex activities. Therefore, these are the main guiding principles in the design of our smart solution so that travellers get a customized service without noticing it:

- **Unobtrusive Sensing** – Healthcare applications usually prefer accuracy of readings over device size. However, in urban community-sized applications and contexts, like Public Transportation, unobtrusive sensing is essential, given the uncomfortability of carrying bulky devices all day and in crowded environments. Thus, here we will use a new generation of wearable computers that combines style and effective sensing capabilities in attractive accessories [2] making sensing almost unnoticeable. The smartphone embedded sensors and small static sensors will also be used to capture a considerable amount of data invisibly [1, 14]. Thus, one of the main advantages of this design is how little the user is requested to participate;

- **Wireless Communication** – Not only should the devices be unnoticeable but it is also important for them to communicate wirelessly given the inconvenience of wires. However, the new low-power low-bandwidth protocols also provide the means to save big amounts of battery resources, specially when the platform consistently uses this technology.

3.2 Architectural Overview

Briefly, our solution, as seen in Figure 1, is comprised of various components that acquire different data:

- **Smartphone** – an Android application that 1) collects context data from the embedded sensors, 2) gathers data from other sensors and 3) communicates with the web service;

- **Environment Sensors** – sensors placed in the vehicle to collect ambient contextual data;

- **User Sensors** – wearable computers used for measuring affective data (e.g. physiological signals).

For what each of this components measure see Section 3.3.1.

3.2.1 *Wireless Mobile Sensor Network*

The proposed architecture is built over a Wireless Mobile Sensor Network where every node of the network is an autonomous sensor reporting physical data (see Section 3.3.1) that might influence the user experience. We connect them in a Wireless Personal Area Network (WPAN), in which the key concept is known as plugging in. Ideally, selected devices when in close proximity are able to connect with each other as if they had cables connecting them. This allows the use of multiple sensors (e.g. more than one wearable body sensor) around the user to collect different information onto his smartphone.

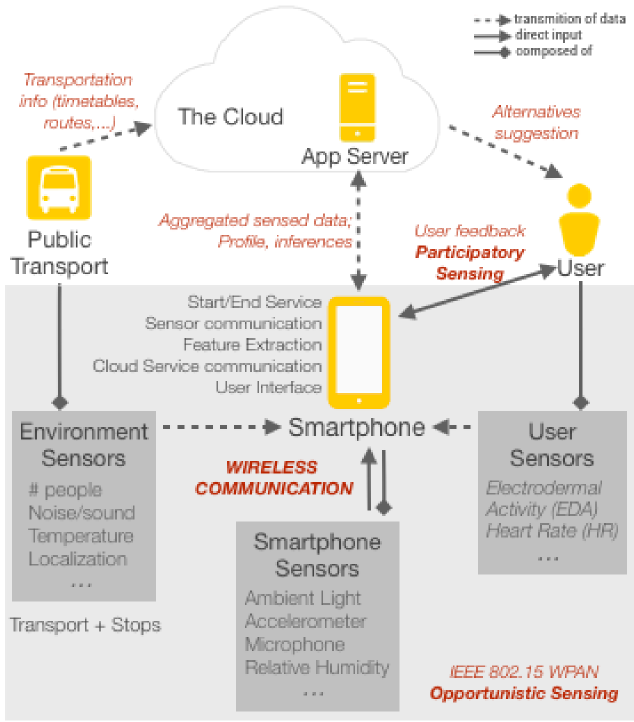


Figure 1: Conceptual architecture; the sensing platform gathers data from environment, smartphone and user sensors along with user input and communicates it to the cloud infrastructure.

For communication purposes we chose the Bluetooth Low-Energy (BLE) included in the Bluetooth 4.0 standard, which is already available in mobile devices. Given the Bluetooth history in the mobile phone community, BLE is expected to receive a bigger commercial adoption as a smartphone standard. The Wahoo Blue HR [18], for example, already uses BLE to communicate heart-rate readings onto an iPhone.

3.2.2 *Smartphone*

The smartphone is the central node of the network given its processing and connectivity capabilities. It acts not only as a sensing node, but also as a knowledge base with sensorial data from every other sensor and communication interface with the cloud throughout internet connection. Finally, for affective and context awareness applications to be effectively user-friendly and identify patterns of behaviour, it is necessary to receive feedback from the users whenever they can or want to give it [19]. As so, given the complexity of emotions and unpredictability of PT conditions, we also defined a participatory sensing module, enabling the traveller to give direct feedback of the trip.

3.2.3 Environment Sensors

Environment sensors, unlike smartphone sensors, read data that influences not only the QoE of one person, but from everyone that uses the same transportation service. It is important to have an overview of the general conditions that are not modified by subjective criteria. Also, some of this readings are sent directly to PT web services and accessed in the cloud (see Figure 1). That is what happens already with the position and delay time of vehicles in various public transportation systems, which we assume in this work.

3.2.4 User Sensors

The widespread use of affective sensing has been limited due to several practical factors, such as lack of comfortable wearable sensors, lack of wireless standards, and lack of low-power affordable hardware [7]. However, these constraints are slowly disappearing with the technological advances in this area and the commercialization of new light and stylish products [2]. As so, we allow the use body sensors in our solution to add useful affective data to the reasoning system and later understand which conditions make the emotional state change.

3.3 Collection and Aggregation of Data

3.3.1 Collection of Sensorial Data

The following sections detail which variables are sensed in each of the isolated components described in the architecture.

Smartphone Sensing The latest version of the Android platform (4.0 Ice Cream Sandwich) offers support for a handful of sensors such as, accelerometer, digital compass, gyroscope, GPS, microphone, camera, barometer, temperature and relative humidity [9]. Smartphone sensing, unlike environment sensors, gives the system useful information about the close surrounding of the user, since it is always with him. The accelerometer, for example, might sense bumping people or careless driving; the microphone and thermometer, the noise level and temperature at the exact traveller's place; GPS, calendar and clock gives the potential for identifying travelling patterns; the music player helps understand the customer's activity. However, some considerations should be taken into account such as using the proximity sensor to know if the smartphone is inside a pocket or bag and cancel useless readings.

As for the participatory sensing module, the smartphone is responsible for prompting the user, at appropriate times, to insert his feedback or review inferences made from the system (see Section 4). Sensed data, specially in noisy and crowded environments is always subject to interferences that do not reflect the actual circumstances. The mood of the traveller is affected by several external factors as already stated. Thus, making the user review such system inferences validates the collected information. On the other side, when the system is not able to infer correctly (e.g. big variances between sensed data and the correlation that defines a user's pattern) the user feedback helps understand the way variables affected his trip. These two interactions - reviews and feedbacks - happen only at the end of every trip and can be answered at any time after that. The appearance of such interactions and information displayed to the user will be described in Section 4.

General Context-Awareness Complementing smartphone sensing, this platform also collects data from static devices implemented in the transportation vehicle. These can sense conditions that the smartphone can not, either because its sensors do not have the best conditions (e.g. inside a bag) or just because it is physically impossible (e.g. estimated current number of people inside a vehicle). There are two types of sensing domains in this component, both of them in real time: ambient temperature, background noise, oscillation levels and vehicle/line information, which are sent directly to the smartphone; and location, delay time, and estimated number of people, which are sent and used in PT web services and will be accessed through the Cloud2Bubble infrastructure, as shown in Figure 1. There is also the possibility of sending all the information over the web, which allows the system to know current environment conditions of a vehicle even when there is no smartphone inside.

Affective Data Until now, the biggest adoption of affective computing systems has been in the healthcare field. Physiological signals such as heart rate (HR) and galvanic skin response (GSR) can give interesting information by looking at its variations and using models [4] to classify emotions. For example, a slow heart rate might mean the user is calm and enjoying the trip, while environmental discrete stimuli (sights, sounds, smells) will evoke changes in skin conductance that symbolize states of stress or joy. Although there are already some models, this research area is still relatively new. In the context of this project we use the simplest classification models available today which will be later improved with further research. However, by using direct user input we are helping the classification of affective state and reducing errors.

3.3.2 Aggregation of Data

After collecting every variable that can influence our perception of the quality of the service, this system intelligently defines profiles of satisfaction through a complex learning process. To start, the smartphone application extracts features from the raw sensed data. Later it transmits this aggregated information to the C2B framework where a machine learning mechanism finds patterns and correlations between variables, which define the user profiles. The algorithms used in this machine learning mechanism are, however, beyond the context of the sensing platform.

Feature Extraction and Classification Feature extraction is fundamental for 1) filtering personal information from raw data, such as voice and conversations for privacy issues, and 2) freeing connectivity resources by not transmitting huge amounts of raw data to the server continuously. The pre-processing in the smartphone mostly groups intervals of sensed data from every sensor in clusters based on time. Clusters of 10 seconds of information saves processing and connectivity resources, as opposed to process and transmit every second of data.

Classification takes place in the cloud computing infrastructure and mainly converts metric variables, such as decibels and centigrade degrees, to a simple 1-10 numeric scale based on historical sensed data. Briefly, if a user rates a noise level of 60dB with an 8 mark and 80dB with 6 during initial feedbacks, a reading of 70dB would result in a 7 mark. It is also

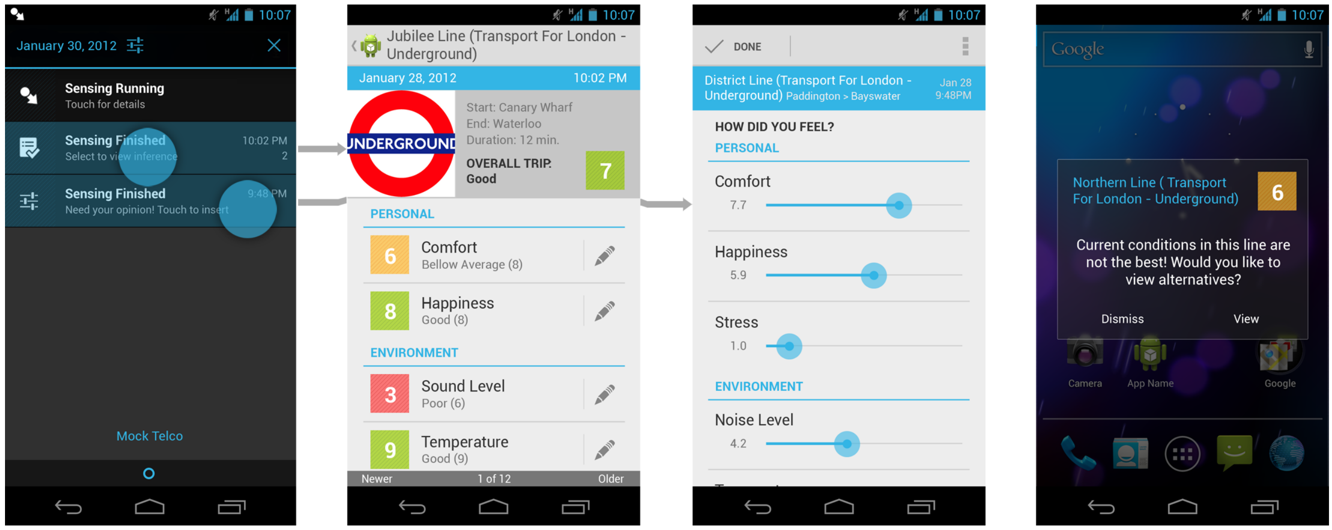


Figure 2: Android application prototype; 1) notification system, 2) user review of system inference, 3) user feedback when system can not make a solid inference and 4) trip suggestions pop-up

used an algorithm to qualitatively classify variances to the profile, i.e. one variable is good, below average or poor compared to the variable in the profile. In the previous example, the 100dB reading would be classified as poor (4), assuming 60db and 8 is the average. With such algorithms the visual interface is simplified to hide the complexity beneath and enhance usability (see Figure 2).

Pattern Recognition and Profiling After classification, these variables are associated with users, defining profiles. Thus, a profile is a series of range (e.g. preferred noise level between 56dB-63dB, because they were classified with higher marks) and list (e.g. preferred means of transportation are metro, tram, bus) variables that characterize the preferred conditions of an individual, allowing the inference mechanism to compare them to current conditions or newly sensed data. Profiles also have information on personal routines, calculated autonomously from smartphone sensing data or given through direct user input. Routines display days of the week, start time of journeys and start and destination stops.

Cloud Server Interface The stronger processing capabilities available in the cloud allied with the access to real-time information from web-services suggests us to use the heavy machine learning and inference algorithms in the C2B module, instead of the smartphone. Communication, as seen in Figure 1, is done through internet connection sharing 1) preprocessed sensed data and user feedbacks from the smartphone to the server and 2) inferences, profiles and current conditions from the server to the smartphone.

4. ANDROID PROTOTYPE

To implement this solution it was developed an Android smartphone application prototype, partially shown in Figure 2. Its key objectives are to gather sensorial data and communicate with the server without burdening the user, i.e. making it transparent and only interacting with him when needing direct solid information - reviews and feedbacks - or to suggest better alternatives. Therefore, we designed our application using notifications that alert the user when it has new relevant information, while most of it runs in the background. Ideally, even starting and ending the sensing service would happen autonomously when entering a transportation service. However, the application also allows him to view live information inside the main activity, such as planning trips, view current conditions in the line, what are the sensors reading now, or view his profile. The sensing module in the smartphone was already described in Section 3.3.1 as well as the appearance user profiles and sensed data in Section 3.3.2. In this chapter we discuss how the information is presented to the user regarding usability principles.

4.1 Notification System

The appplication uses three different notifications (see Figure 2) for notifying the user about on-going activities or to display new information. The first one takes the user to the sensing activity, where he can check current conditions and what is being sensed. Though this is just a visual interface; the sensing is always running until it is explicitly stopped. The second and third appear when a trip or sensing finishes. They bring to screen the review and the feedback activities, as shown in 2) and 3) in Figure 2, respectively. The fourth screen (4)) shows a pop-up message informing that the conditions on the customer is about to take, based on the routines saved in his profile, does not provide the best conditions for the user (or there are alternatives with better conditions).

4.2 Delivery of Service

Combining user routines with current real-time conditions and user preferences gives the system the potential to suggest better services even before the user starts a journey. The intelligence of pro-actively offering better experiences regarding individual preferences is a ground-breaking solution in mobile applications for enhancing QoE - be it in public transportation or in other contexts. The customer can also, however, plan his trip using this applications. The results make use of routing algorithms in PT web services plus the added information on preferences, making them different form user to user.

4.3 Usability

A functional prototype of the GUI was tested with real users, which pointed some issues which were addressed in the implementation phase. The position of the Start/Stop sensing buttons, few information on trip planning and difficulties in finding the routines were the top concerns. However, every participant, after some time, was able to navigate fluently throughout the application and understand its use in real situations. As referred in Section 3.3.2 the user of a known scale and colors made the interface simple and understandable, hiding the complexity of the sensor network beneath. Also, the application is also thought in a way that the more solid a pattern (profile) is the less the user will have to insert data, since inferences will be more oftenly correct.

5. CONCLUSION AND FUTURE WORK

The current ubiquitous computing environment is moving computation to mobile devices integrated in every days objects. Thus, giving transparency to Human-Computer Interaction in a more fluent and unobtrusive User eXperience, as we pursue. Using this paradigm we define a Wireless Mobile Sensor Network to collect influencing variables from autonomous sensors that help understand the likings and preferences of users. In Public Transportation, Quality of Service oriented applications offer easier and more comfortable journeys. However, assessing personal Quality of Experience through the assessment of emotions and environment conditions enables the customization of services to personal preferences. In this paper, we presented a mobile sensing platform based on a smartphone application to unobtrusively collect such multi-modal information. The design and implementation reflects the attention to the users in giving them support for choosing better trip alternatives almost unnoticeable: or by advising them or by customizing trip plans. For PT providers, it stands as a powerful tool to understand the emotional component of their clients QoE. This way we intelligently and invisibly add an emotional dimension to objective parameters, such as time and cost constraints, to the rational decision process of transportation means.

In the future, the prototype will be tested in real systems and situations. At first it will provide the C2B framework with sensorial data from participatory sensing and context-awareness smartphone sensing. Later, it will be refined with the introduction of user and environment sensors to complete the proposed sensing platform. Ideally, this project will conduct researchers in the direction of ubiquitous wireless sensor networking based in smartphones and the development of attractive wearable computers for urban user adoption, transparently measuring emotions and introducing them into the QoE assessment criteria. Thus, this work will provide concepts and insights that meet the concept of the Future Internet where devices communicate in an intelligent way to satisfy human needs, not only in public transportation but in any kind of service.

6. REFERENCES

[1] M. Azizyan, I. Constandache, and R. Roy Choudhury. SurroundSense: mobile phone localization via ambience fingerprinting. In *ACM MobiCom*, 2009.

[2] Basis Science. Basis b1. https://mybasis.com/.

[3] A. T. Campbell, S. B. Eisenman, N. D. Lane, E. Miluzzo, and R. A. Peterson. People-centric urban sensing. In *ACM WICON*, 2006.

[4] G. Chanel, J. Kronegg, D. Grandjean, and T. Pun. Emotion assessment: Arousal evaluation using eegs and peripheral physiological signals. *Classification and Security*, pages 530–537, 2006.

[5] C. Chorus, E. Molin, and B. Van Wee. Use and Effects of Advanced Traveller Information Services (ATIS): A Review of the Literature. *Transport Reviews*, 26(2):127–149, Mar. 2006.

[6] P. M. Costa, J. Pitt, J. F. e. Cunha, and T. Galvao. Cloud2Bubble: Enhancing Quality of Experience in Mobile Cloud Computing Settings. In *The Third ACM Workshop on Mobile Cloud Computing and Services*, 2012.

[7] R. Fletcher et al. iCalm: Wearable sensor and network architecture for wirelessly communicating and logging autonomic activity. *IEEE Information Technology in Biomedicine*, 14(2):215 –223, 2010.

[8] Gingerd. Ginger.io. http://ginger.io/, 2011.

[9] Google. Google developers. http://developer.android.com/.

[10] N. Lane, E. Miluzzo, H. Lu, D. Peebles, T. Choudhury, and A. Campbell. A survey of mobile phone sensing. *IEEE Communications Magazine*, 48(9):140 –150, 2010.

[11] N. D. Lane et al. BeWell: A smartphone application to monitor , model and promote wellbeing. In *Pervasive Health*, 2011.

[12] K.-H. Lüke, H. Mügge, M. Eisemann, and A. Telschow. Integrated solutions and services in public transport on mobile devices. In *IICS*, volume 148 of *LNI*. GI, 2009.

[13] S. Mayer, D. Guinard, and V. Trifa. Facilitating the integration and interaction of real-world services for the web of things. In *Urban Internet of Things – Towards Programmable Real-time Cities (UrbanIOT 2010); IoT 2010*, 2010.

[14] E. Miluzzo et al. Sensing meets mobile social networks: the design, implementation and evaluation of the CenceMe application. In *ACM SenSys*, 2008.

[15] S. Möller, K. Engelbreacht, C. Kühnel, I. Wechsung, and B. Weiss. A taxonomy of quality of service and quality of experience of multimodal human-machine interaction. In *QoMEx*, 2009.

[16] R. Picard. *Affective computing*. The MIT Press, 1997.

[17] D. Saha and A. Mukherjee. Pervasive Computing: A Paradigm for the 21st Century. *Computer*, 36(3):25–31, Oct. 2003.

[18] Wahoo Fitness. Wahoo Blue HR. http://www.wahoofitness.com/wahoo_blue/.

[19] M. T. West. Ubiquitous computing. In *ACM SIGUCCS*, 2011.

Appendix B

Usability Test Script

TEST SCRIPT

Title: Smart Mobile Sensing for Measuring Quality of Experience in Urban Public Transports

Version: 1.0 - Initial Prototype Usability Testing

Date: March 27th, 2012

PERSONA AND CONTEXT:

You are a worker/student in London with a fixed or semi-flexible schedule. Besides, you go to your workplace/school every day by urban public transportation. You downloaded the application so you expect it to suggest you better routes and you were guided through simple explaining steps on how it works. Plus, you have with you a watch that can read heart beats, compatible with the application.

SCENARIO 1:

Today you go to work and you want to collect the data from the journey to support the application's inferences.

Tasks:
1.1. You enter the bus/metro. Start collecting the application data for the trip.
1.2. Connect your smartphone to your watch, so you collect personal affective data.
1.3. Check that the application is receiving data from your personal device.
1.4. While travelling you want to navigate through the application. What is the delay time of the current vehicle you are in?
1.5. You have to make a call so you go out of the application. When you are finished, view your profile.
1.6. You realize your profile is wrong so you want to edit. Change preferred temperature to between 18ºC and 20ºC.
1.7. You arrived to your workplace/school stop. Stop sensing.
1.8. You received one notification... Enter your feedback of the trip.

SCENARIO 2:

You have had the application for some time and gathered enough personal preferences data from your trips. Today you want to go play tennis with a friend but you do not have a car, so you go by public transportation.

Tasks:
2.1. Plan trip from the closest bus/metro stop to the tennis complex. What journey would you choose? Why?
2.2. You realize you will play tennis at this time every week and you do not want to check conditions of the lines every time. Find a way to receive notifications in the following weeks if the conditions are not the best for you.

2.3. You finished your trip and you received one notification... But you are late and do not have time to respond now.
2.4. When you get home you remember to review trip. Review the inference. You do not agree with the comfort inference. Change it.

SCENARIO 3:

You are coming back home from work in the afternoon. However, today there is a special event in the city center making public transportation more crowded than normal. You receive a pop-up message in your smartphone.

Tasks:
3.1. How are you going to get home?
3.2. You have to go shopping first in a different local than your neighborhood. What is the best way to go? (Change destination)

Usability Test Script

Appendix C

Participant Information Sheet and Consent Form

Faculdade de Engenharia da Universidade do Porto
Departamento de Engenharia Informática
Rua Dr. Roberto Frias, s/n 4200-465 Porto PORTUGAL

Tel.: +351 91 597 05 07
Email: joao.guerra@fe.up.pt

PARTICIPANT INFORMATION SHEET

Title: Smart Mobile Sensing for Measuring Quality of Experience in Urban Public Transports

To participants:

My name is João Vieira, I am a MSc student in the Department of Informatics Engineering at Faculdade de Engenharia da Universidade do Porto. I am conducting research into affective ubiquitous computing for enhancing quality of experience in urban public transports in collaboration with Pedro Maurício Costa from the Imperial College London. We are investigating a new way to support people's decision of traffic means based on his/her emotional state and environment conditions. A smartphone application will be developed as an interface with the traveller. A part of exploring these ideas is involving potential 'ordinary' users in the design, usability testing and evaluation of the prototype application. In this study we are exploring design ideias that will better serve the general public.

You are invited to participate in our research and we would appreciate any assistance you can offer us, although you are under no obligation to do so.

Participation involves one visit to our laboratory at the Imperial College London, for approximately 20 minutes. If you agree to participate, you may be asked to perform a number of tasks using our application prototype. The scenarios and tasks will be fully explained. You will be asked to navigate through the prototype to accomplish the referred tasks. The activities you undertake and the time you spend working on each task will be digitally recorded together with synchronized video. You will be asked to fill in a short questionnaire to note your age, education level and existing experience with the tasks and technology and complete a short questionnaire on your experience. This is a test of the application; we are not testing you. If you find something difficult to use, chances are that others will as well. This test is simply a mean of evaluating the application's design and to discover any issues we need to address.

All the questionnaire information you provide and recorded data will remain anonymous and used solely by researchers inside this project. Your name will not be used in any reports arising from this study. The information collected during this study may be used in future analysis and publications and will be kept indefinitely. At the conclusion of the study, a summary of the findings will be available from the researchers upon request.

FEUP	Faculdade de Engenharia da Universidade do Porto Departamento de Engenharia Informática Rua Dr. Roberto Frias, s/n 4200-465 Porto PORTUGAL Tel.: +351 91 597 05 07 Email: joao.guerra@fe.up.pt

If you don't want to participate, you don't have to give any reason for your decision. If you do participate, you may withdraw at any time during the session and you can also ask for the information you have provided to be withdrawn at any time.

If you agree to participate in this study, please first complete the consent form attached to this information sheet. Your consent form will be kept separately from your questionnaire data so that no-one will be able to identify your answers from the information you provide.

Thank you very much for your time and help in making this study possible. If you have any questions at any time you can contact me (joao.guerra@fe.up.pt), Pedro (pm.costa@imperial.ac.uk), or my supervisor, Professor Teresa Galvão Dias (tgalvao@fe.up.pt).

FEUP	Faculdade de Engenharia da Universidade do Porto Departamento de Engenharia Informática Rua Dr. Roberto Frias, s/n 4200-465 Porto PORTUGAL Tel.: +351 91 597 05 07 Email: joao.guerra@fe.up.pt

CONSENT FORM

Title: Smart Mobile Sensing for Measuring Quality of Experience in Urban Public Transports

Researcher: João Miguel Guerra Vieira

I have been given and understood an explanation of this research project. I have had an opportunity to ask questions and have them answered. I understand that at the conclusion of the study, a summary of the findings will be available from the researchers upon request.

I understand that the data collected from the study will be held indefinitely and may be used in future analysis.

I understand that I may withdraw myself and any information traceable to me at any time without giving a reason, and without any penalty.

I understand that I may withdraw my participation during the session at any time.

I agree to take part in this research by completing the session.

I **agree/do not agree** digital and video recordings taken during the session being used research reports on this project.

Signed: ____

Name (please print clearly): ____

Date: ________________________

Appendix D

Pre-Test Questionnaire

PRE-TEST DEMOGRAPHIC QUESTIONNAIRE

Age:	
Profession:	
Nationality:	

PUBLIC TRANSPORTATION

1) For what and which kind?

What	Kind

2) How often?

Everyday	1 p/week	2-3 p/week	Weekdays	When you need		

3) On an ordinary trip, do you have any mean for browsing for alternatives?

4) How often do you have them?

Always	Most of the tin	Sometimes	Never			

5) How do you choose amongst different alternatives?

Trip duration	Waiting time	Type of transportation	Other:		

6) Your opinion on journeys (1 Bad - 5 Excelent):

	Bad	Not satisfactor	Satisfactory	Above average	Excellent
5.1) Comfort:					
5.2) On time:					
5.3) Crowded:					
5.4) Noisy:					
5.5) Easy to plan:					
5.6) Easy to commute:					
5.7) Information to the public					

SMARTPHONE USAGE

7) Do you have a smartphone?	

8) Which Operating System?	

9) What do you use it for?

SMS/call	Email	Camera	Browse Intern	Social apps	GPS	Other:

10) Do you have any app for journey planning/check line conditions?	

11) What is your comfort level with technology?

Uncomfortable	Not very Comfortable	Comfortable	Somewhat Comfortable	Very Comfortable

Appendix E

Post-Test Questionnaire

POST-TEST QUESTIONNAIRE

TASKS

	1) I understand the task					2) The tool helped me to complete tas					Time spent:	Tester Comments:
Tasks	Strongly disagree	Disagree	Neutral	Agree	Strongly Agree	Strongly disagree	Disagree	Neutral	Agree	Strongly Agree		
1.1												
1.2												
1.3												
1.4												
1.5												
1.6												
1.7												
1.8												
2.1												
2.2												
2.3												
2.4												
3.1												
3.2												

PROTOTYPE

	Strongly disagree	Disagree	Neutral	Agree	Strongly Agree
3) I understand how this application works/will wo					
4) I would like to use this application in the future					
5) Navigating through the application was easy					

COMMENTS/RECOMMENDATIONS

Appendix F

User Manual For Tests - PTSense App (PT)

Faculdade de Engenharia da Universidade do Porto
Departamento de Engenharia Informática
Rua Dr. Roberto Frias, s/n
4200-465 Porto PORTUGAL

FEUP

PTSense App
Manual do utilizador para testes

Versão do manual: 1.0
Versão da aplicação a testar: 0.9.4 (Android 2.1+), 0.9.4 (Android 4.0+)

Porto, 29 de Maio de 2012

Índice

Objetivo

O principal objetivo desta experiência é a recolha de dados de utilizadores reais representando uma amostra dos potênciais utilizadores futuros. Os dados recolhidos são obtidos através da aplicação PTSense e enviados para um servidor onde serão posteriormente analisados. Esta informação vai-nos permitir analisar de que forma as condições nos meios de Transporte Públicos (TP) durante as viagens influenciam a percepção de cada utilizador acerca das mesmas. Desta forma, esta experiência leva-nos a perceber se é ou não possível encontrar padrões nas preferências de comforto de cada utilizador.

O que é o PTSense?

1. A aplicação PTSense é uma aplicação móvel que permite ao sistema interagir com os utilizadores. A partir da aplicação é possível recolher ou sentir certas váriáveis do ambiente durante viagens em meios de transporte público. Esta recolha é efectuada através dos sensores do smartphone e, possivelmente de sensores instalados nos veículos. A estes dados é agregada informação acerca do estado emocional do utilizador com base na sua perceção da viagem, introduzida directamente na aplicação no final da viagem.

2. Após terminada uma viagem, toda a informação é enviada para um sistema remoto na web através de conexão sem fios (e.g. Wi-Fi) onde é processada. Isto dará origem a perfis de preferências dos utilizadores com base nas correlações contexto-emocionais encontradas. Desta forma, quanto maior o número de viagens, mais fiável será o perfil.

3. Finalmente, tendo acesso às condições em tempo real nos TP e aos perfis dos utilizadores (incluindo as suas rotinas diárias/semanais) o PTSense informa o utilizador através do seu smartphone de melhores rotas caso as suas preferências não se verifiquem. Por outro lado, permite ainda personalizar a pesquisa de melhores rotas em qualquer momento com base nessa mesma informação.

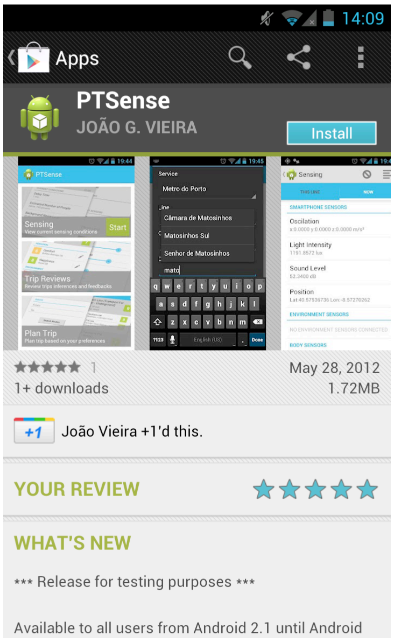
> **Nota: Dado o projeto estar ainda numa fase inicial do desenvolvimento, a aplicação suporta até à data apenas as fases 1 e 2, acima referidas, desta interação.**
>
> **Para mais informações consulte http://www.cloud2bubble.com/case-studies/experiment/.**

1. Instalar

Instalar o PTSense é bastante fácil através da Google Play Store. Para tal basta:

1. Pesquisar pela aplicação “PTSense” na Google Play Store no seu smartphone;
2. Pressionar “Transferir” no topo do ecrã;
3. Ler e aceitar as permissões para inicar a transferência e instalação;
4. Quando terminado, pressionar “Abrir” ou encontrá-la na sua lista de aplicações.

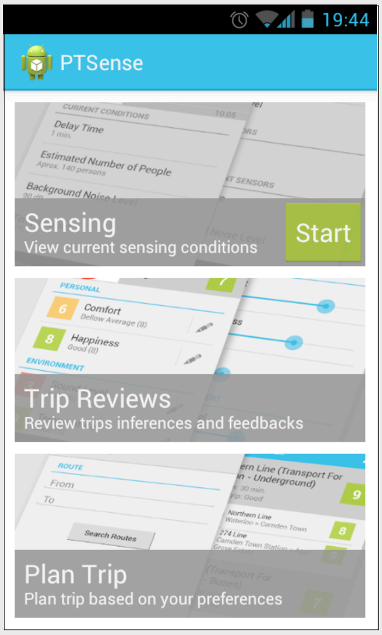
Ou pode também ser feito através do browser em https://play.google.com/store/apps/details?id=com.cloud2bubble.ptsense.

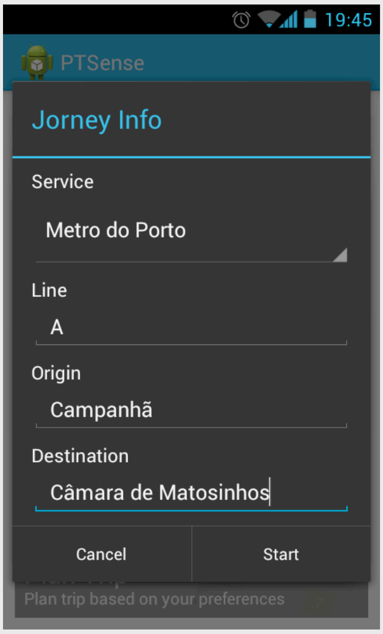


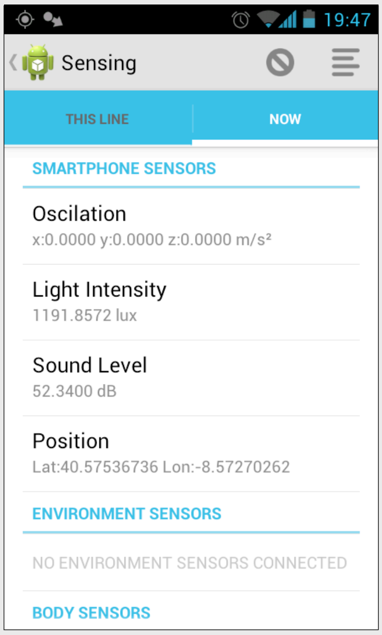
2. Usar a aplicação

A estrutura do PTSense é bastante simples, pelo que as descrições que se seguem explicam as funcionalidades principais, disponíveis nesta versão.

2.1. Visão Geral







1. Página Inicial

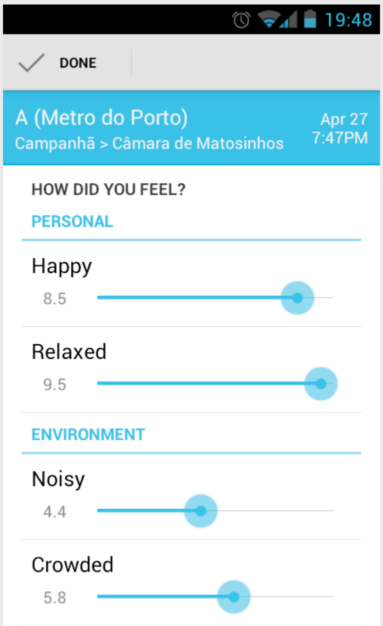
- iniciar a recolha de dados em “Start”
- pressionar a tecla “Menu” no smartphone para aceder a opções extra
- a opção “Sensing” encontra-se inactiva até iniciar a recolha

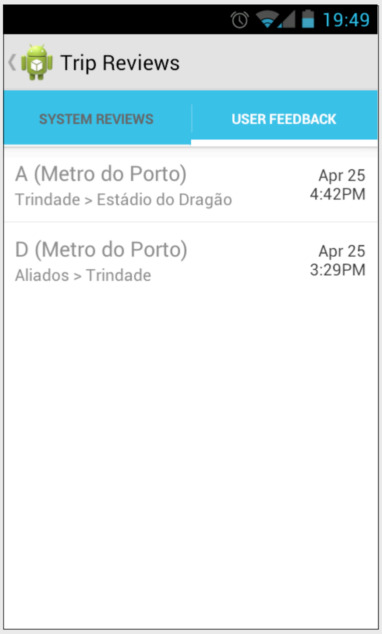
2. Dados da viagem

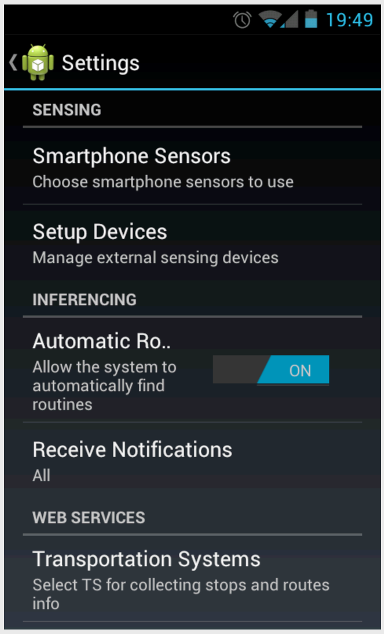
- inserir os dados da viagem
- caixas de texto com Auto-complete
- pode iniciar a viagem sem introduzir todos os dados, mas estes são necessários para a terminar

3. Sensing

- “Now” mostra dados a serem recolhidos pelos sensores activos
- opções no menu superior levam a (7) e (2), respectivamente
- “This Line” não implementado







4. Feedback da viagem

- modificar os sliders consoante a opinião relativa à viagem
- inserir comentário opcional
- registo é gravado quando pressionado “Done”
- pode ser efectuado mais tarde

5. Lista de Revisões

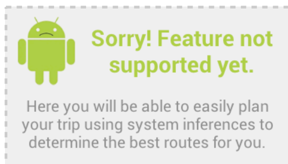
- “User Feedback” mostra as revisões em falta para as viagens listadas
- ordem temporal: mais recentes no topo
- “System reviews” não implementado

6. Definições

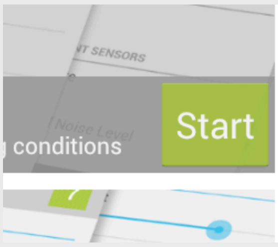
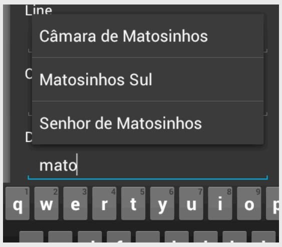
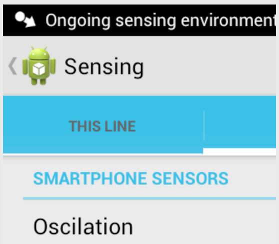
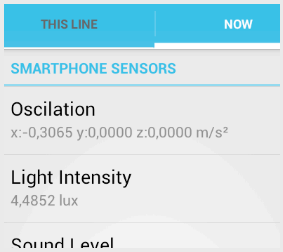
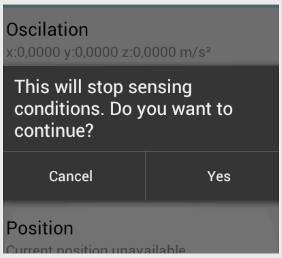
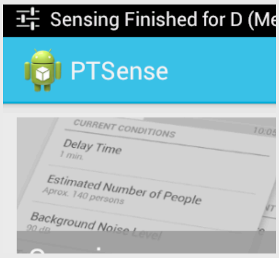
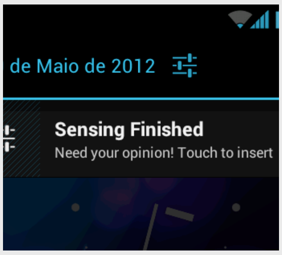
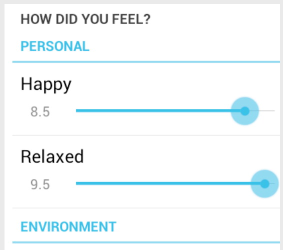
- definir quais os sensores a utilizar
- definir quais a notificações a receber
- apenas as duas opções acima estão disponíveis nesta versão

Nota: As imagens apresentadas dizem respeito à versão para Android 4.0. O aspeto pode alterar em versões mais antigas.

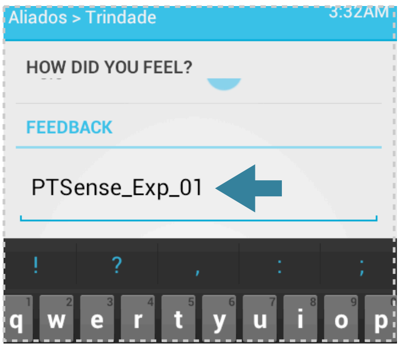
Nota: As partes não referidas da aplicação ou indicadas como não implementadas apresentam esta imagem, seguida de uma breve descrição da funcionalidade futura.



2.2. Exemplo de utilização

			
1. Pressione **"START"** na Página inicial para iniciar recolha de dados no **início** da viagem.	**2.** Insira os **dados da viagem** quando quiser: no início, durante ou no fim.	**3.** A **notificação** indica que a recolha está a ser efectuada	**4.** Acompanhe os dados ou saia da aplicação. Para voltar, pressione na notifi- cação em (3).
			
5. **Termine** a recolha dos dados no **fim** da viagem, na Página inicial ou em "Sensing".	**6.** A **notificação** indica que tem a revisão da úl- tima viagem (ou de mais) a fazer.	**7.** Pressione a notificação para inserir **quando quiser** ou vá à opção **"Trip Reviews"** mais tarde.	**8.** Insira o **feedback** da viagem, com um comentário opcional.

3. Iniciar o teste



De forma a analisar apenas dados relevantes para o teste, enviados pelos participantes, é necessário controlar a recepção dos mesmos, visto que a aplicação está de momento disponível ao público. Para tal é pedido aos participantes que enviem o respectivo CÓDIGO DE UTILIZADOR recebido por email da seguinte forma:

1. Iniciar uma nova viagem e terminá-la (a informação da viagem não é relevante);

2. No feedback desta viagem inserir o CÓDIGO DE UTILIZADOR no campo de comentário, no final, como é indicado na figura ao lado.

4. Contactar

Esta fase do desenvolvimento é crucial para o futuro do projecto. Assim, é necessário atender a todas as imperfeições de forma a minimizar as correcções tardias com o objetivo de criar uma aplicação ao nível esperado pelos utilizadores. Desta forma, agradece-se o contato em caso de **dúvidas, sugestões, erros ou "crashs"** para um dos seguintes contactos:

- **João M. Guerra Vieira**
  joao.guerra@fe.up.pt
  Faculdade de Engenharia da Universidade do Porto

- **Pedro Maurício Costa**
  pm.costa@imperial.ac.uk
  Imperial College London

Boas viagens!

References

[ACRC09] Martin Azizyan, Ionut Constandache, and Romit Roy Choudhury. SurroundSense: mobile phone localization via ambience fingerprinting. In *Proceedings of the 15th annual international conference on Mobile computing and networking*, MobiCom '09, pages 261–272, New York, NY, USA, 2009. ACM.

[BDE09] M.A. Bayir, M. Demirbas, and N. Eagle. Discovering spatiotemporal mobility profiles of cellphone users. In *IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks Workshops, 2009. WoWMoM 2009*, pages 1 –9, June 2009.

[CCH+08] T. Choudhury, S. Consolvo, B. Harrison, J. Hightower, A. LaMarca, L. LeGrand, A. Rahimi, A. Rea, G. Bordello, and B. et al. Hemingway. The Mobile Sensing Platform: An embedded activity recognition system. *Pervasive Computing, IEEE*, 7(2):32 –41, april-june 2008.

[CCH+10] Andrew T Campbell, Tanzeem Choudhury, Shaohan Hu, Hong Lu, Mashfiqui Rabbi, Rajeev D S Raizada, and Matthew K Mukerjee. Neurophone : Brain-mobile phone interface using a wireless eeg headset categories and subject descriptors. *Design*, pages 3–8, 2010.

[CDCT+01] R. Cowie, E. Douglas-Cowie, N. Tsapatsoulis, G. Votsis, S. Kollias, W. Fellenz, and J.G. Taylor. Emotion recognition in human-computer interaction. *Signal Processing Magazine, IEEE*, 18(1):32 –80, jan 2001.

[CEL+06] A. T. Campbell, S. B. Eisenman, N. D. Lane, E. Miluzzo, and R. A. Peterson. People-centric urban sensing. In *Proceedings of the 2nd annual international workshop on Wireless internet*, WICON '06, New York, NY, USA, 2006. ACM.

[CKGP06] Guillaume Chanel, Julien Kronegg, Didier Grandjean, and Thierry Pun. Emotion assessment: Arousal evaluation using eegs and peripheral physiological signals. *Classification and Security*, pages 530–537, 2006.

[CLR10] Michael Chui, Markus Löffler, and Roger Roberts. The Internet of Things. *McKinsey Quarterly*, (2):1–9, 2010.

[CMT+08] S. Consolvo, S. W. McDonald, T. Toscos, M. Y. Chen, J. Froehlich, B. Harrison, P. Klasnja, A. LaMarca, and L. et al. LeGrand. Activity sensing in the wild: a field trial of ubifit garden. In *Proceedings of the twenty-sixth annual SIGCHI conference on Human factors in computing systems*, CHI '08, pages 1797–1806, New York, NY, USA, 2008. ACM.

[Cos11] Pedro M. Costa. A framework for quality of experience enhancing systems. Technical report, Imperial College London, February 2011.

[CPCG12] Pedro Mauricio Costa, Jeremy Pitt, Joao Falcao e Cunha, and Teresa Galvao. Cloud2Bubble: Enhancing Quality of Experience in Mobile Cloud Computing Settings. In *The Third ACM Workshop on Mobile Cloud Computing and Services*, 2012.

[dP] Metro do Porto. imetroporto. http://itunes.apple.com/tr/app/imetroporto/id452703146?mt=8. Last accessed Jan 2012.

[Emo] Emotiv. Emotiv epoc. http://emotiv.com/. Last accessed Jan 2012.

[FDG+10] R.R. Fletcher, K. Dobson, M.S. Goodwin, H. Eydgahi, O. Wilder-Smith, D. Fernholz, Y. Kuboyama, E.B. Hedman, Ming-Zher Poh, and R.W. Picard. iCalm: Wearable sensor and network architecture for wirelessly communicating and logging autonomic activity. *Information Technology in Biomedicine, IEEE Transactions on*, 14(2):215 –223, march 2010.

[Fit] Wahoo Fitness. Wahoo blue hr. http://www.wahoofitness.com/wahoo_blue/. Last accessed Jan 2012.

[Gem08] M. Gemeinder. Touch & Travel – NFC based automatic fare collection using a passive infrastructure, 2008.

[Gin] Gingerd. Ginger.io. http://ginger.io/. Last accessed Nov 2011.

[Gmb] VBB Verkehrsverbund Berlin-Brandenburg GmbH. Vbb farhinfo app. http://www.vbbonline.de/index.php?cat=1&sCat=465. Last accessed Nov 2011.

[Gooa] Google. Android developers. http://developer.android.com/. Last accessed May 2012.

[Goob] Google. Google prediction api. https://developers.google.com/prediction/. Last accessed Jun 2012.

[Gooc] Google. Google wallet. http://www.wahoofitness.com/wahoo_blue/. Last accessed Jan 2012.

[Gra10] Jonathan Gratch. Editorial. *Affective Computing, IEEE Transactions on*, 1(1):1–10, Jan. 2010.

[HBC+96] V. Hewett, Baecker, Card, Carey, Gasen, Mantei, Perlman, and Strong. *Curricula for Human-Computer Interaction*. 1996.

[Hei09] Bob Heile. Overview of 808.15 projects, 2009.

[Jek05] U. Jekosh. *Voice and Speech Quality Perception. Assessment and Evaluation*. Springer, Berlin, 2005.

[LCCL10] Cheng-yu Lin, Ling-jyh Chen, Ying-yu Chen, and Wang-chien Lee. A comfort measuring system for public transportation systems using participatory phone sensing. In *International Workshop on Sensing for App Phones (PhoneSense)*, 2010.

[LMET09] Karl-Heinz Lüke, Holger Mügge, Matthias Eisemann, and Anke Telschow. Integrated solutions and services in public transport on mobile devices. In *IICS*, volume 148 of *LNI*, pages 109–122. GI, 2009.

[LML+10] N.D. Lane, E. Miluzzo, Hong Lu, D. Peebles, T. Choudhury, and A.T. Campbell. A survey of mobile phone sensing. *Communications Magazine, IEEE*, 48(9):140 –150, sept. 2010.

[LML+11] N. D. Lane, M. Mohammod, Mu Lin, X. Yang, Hong Lu, Shahid Ali, A. Doryab, E. Berke, T. Choudhury, and A. T. Campbell. BeWell: A smartphone application to monitor , model and promote wellbeing. In *5th International ICST Conference on Pervasive Computing Technologies for Healthcare*, 2011.

[Mat] MathBits. Correlation coeficient: How well does your regression equation truly represent your set of data? http://mathbits.com/mathbits/tisection/statistics2/correlation.htm. Last accessed Jun 2012.

[MEK+09] S. Möller, K. Engelbreacht, C. Kühnel, I. Wechsung, and B. Weiss. A taxonomy of quality of service and quality of experience of multimodal human-machine interaction. In *International Workshop on Quality of Multimedia Experience 2009. (QoMEx '09)*, pages 7–12, 2009.

[MGT10] Simon Mayer, Dominique Guinard, and Vlad Trifa. Facilitating the integration and interaction of real-world services for the web of things. In *Urban Internet of Things – Towards Programmable Real-time Cities (UrbanIOT 2010); IoT 2010*, 2010.

[MLE07] Holger Mügge, Karl-Heinz Lüke, and Matthias Eisemann. Potentials and requirements of mobile ubiquitous computing for public transport. In Rainer Koschke, Otthein Herzog, Karl-Heinz Rödiger, and Marc Ronthaler, editors, *GI Jahrestagung (2)*, volume 110 of *LNI*, pages 237–246. GI, 2007.

[MLF+08] E. Miluzzo, N. D. Lane, K. Fodor, R. Peterson, Hong Lu, M. Musolesi, S. B. Eisenman, X. Zheng, and A. T. Campbell. Sensing meets mobile social networks: the design, implementation and evaluation of the cenceme application. In *Proceedings of the 6th ACM conference on Embedded network sensor systems*, SenSys '08, pages 337–350, New York, NY, USA, 2008. ACM.

[MM] G. MacKerron and S. . London School of Economics (LSE) Mourato. Mappiness. http://www.mappiness.org.uk/. Last accessed Jan 2012.

[MPL+11] E. Miluzzo, M. Papandrea, N. D. Lane, A. M. Sarroff, S. Giordano, and A. T. Campbell. Tapping into the vibe of the city using vibn, a continuous sensing application for smartphones. In *Proceedings of 1st international symposium on From digital footprints to social and community intelligence*, SCI '11, pages 13–18, New York, NY, USA, 2011. ACM.

[MRS+09] M. Mun, S. Reddy, K. Shilton, N. Yau, J. Burke, D. Estrin, M. Hansen, E. Howard, R. West, and P. Boda. Peir, the personal environmental impact report, as a platform for participatory sensing systems research. In *Proceedings of the 7th international conference on Mobile systems, applications, and services*, MobiSys '09, pages 55–68, New York, NY, USA, 2009. ACM.

[MZA11] Karan Mitra, Arkady Zaslavsky, and Christer Ahlund. A probabilistic context-aware approach for quality of experience measurement in pervasive systems. In *Proceedings of the 2011 ACM Symposium on Applied Computing*, SAC '11, pages 419–424, New York, NY, USA, 2011. ACM.

[Nie94] Jakob Nielsen. *Usability engineering*. Morgan Kaufmann Publishers, 1994.

[Nie00] Jakob Nielsen. Why you only need to test with 5 users, 2000.

[PKG+09] Ming-Zher Poh, Kyunghee Kim, A.D. Goessling, N.C. Swenson, and R.W. Picard. Heartphones: Sensor earphones and mobile application for non-obtrusive health monitoring. In *Wearable Computers, 2009. ISWC '09. International Symposium on*, pages 153 –154, sept. 2009.

[Pre] Presselite. London tube app for iphone. http://www.presselite.com/iphone/londontube/. Last accessed Dec 2011.

[RACB11] M. Rabbi, S. Ali, T. Choudhury, and E. Berke. Passive and in-situ assessment of mental and physical well-being using mobile sensors. In *Proceedings of the 13th international conference on Ubiquitous computing*, UbiComp '11, pages 385–394, New York, NY, USA, 2011. ACM.

[Rai09] Geoffrey Raines. Cloud computing and soa. Technical report, The MITRE Corporation, October 2009.

[Sch10] D. Schneider. Wireless networking dashes in a new direction. *Spectrum, IEEE*, 47(2):9 –10, february 2010.

[Sci] Basis Science. Basis b1. https://mybasis.com/. Last accessed Jun 2012.

[SIG] Bluetooth SIG. Bluetooth smart devices. http://www.bluetooth.com/Pages/Bluetooth-Smart-Devices.aspx. Last accessed Jul 2012.

[TWDS11] H. Turner, J. White, B. Dougherty, and D. Schmidt. Building mobile sensor networks using smartphones and web services: Ramifications and development challenges. In *Handbook of Research on Mobility and Computing: Evolving Technologies and Ubiquitous Impacts*, SAC '11, pages 502–521, Hershey, 2011. IGI Global.

[Uni] Dartmouth University. Smartphone sensing group. http://sensorlab.cs.dartmouth.edu/. Last accessed Dec 2011.

[Wes11] Matthew T. West. Ubiquitous computing. In *Proceedings of the 39th ACM annual conference on SIGUCCS*, SIGUCCS '11, pages 175–182, New York, NY, USA, 2011. ACM.

[WGB99] M. Weiser, R. Gold, and J. S. Brown. The origins of ubiquitous computing research at parc in the late 1980s. *IBM Systems Journal*, 38(4):693 –696, 1999.

[WTL02] Miriam Walker, Leila Takayama, and James A. Landay. High-fidelity or low-fidelity, paper or computer? choosing attributes when testing web prototypes. In *Proc. Human Factores and Ergonomics Society 46th Annual Meeting*, pages 661–665, 2002.