

University of Trás-os-Montes e Alto Douro

A communication channels dynamic switching model for always-connected availability of service oriented mobile applications

Tese de Doutoramento

Diogo Pires de Azevedo

Supervisor: Professor Doutor Hugo Paredes

Co-supervisor: Professor Doutor Benjamim Fonseca



Vila Real, 2017

A communication channels dynamic switching model for always-connected availability of service oriented mobile applications

Tese de Doutoramento

Diogo Pires de Azevedo

Supervisors:

Doutor Professor Hugo Paredes, Professor Associado com Agregação da Escola de Ciências e Tecnologia da Universidade de Trás-os-Montes e Alto Douro;

Doutor Professor Benjamim Fonseca Professor Associado com Agregação da Escola de Ciências e Tecnologia da Universidade de Trás-os-Montes e Alto Douro;

President of the Jury

Doutor Vitor Manuel de Jesus Filipe, Professor Associado com Agregação da Escola de Ciências e Tecnologia da Universidade de Trás-os-Montes e Alto Douro;

Members of the Jury

Doutor Mário Marques Freire, Professor Catedrático da Faculdade de Engenharia da Universidade da Beira Interior;

Doutor João Manuel Pereira Barroso, Professor Associado com Agregação da Escola de Ciências e Tecnologia da Universidade de Trás-os-Montes e Alto Douro;

Doutor José Benjamim Ribeiro da Fonseca, Professor Auxiliar com Agregação da Escola de Ciências e Tecnologia da Universidade de Trás-os-Montes e Alto Douro;

Doutor Hugo Alexandre Paredes Guedes da Silva, Professor Auxiliar com Agregação da Escola de Ciências e Tecnologia da Universidade de Trás-os-Montes e Alto Douro;

Doutor Jorge Miguel Sá Silva, Professor Auxiliar da Faculdade de Ciências e Tecnologia da Universidade de Coimbra;

Doutor João Paulo Silva Barraca, Professor Auxiliar da Universidade de Aveiro;

Doutor João Paulo Pereira de Sousa, Professor Adjunto da Escola Superior de Comunicação, Administração e Turismo do Instituto Politécnico de Bragança.

This work was supported by Project "NanoSTIMA: Macro-to-Nano Human Sensing: Towards Integrated Multimodal Health Monitoring and Analytics/NORTE-01-0145-FEDER- 000016" financed by the North Portugal Regional Operational Programme (NORTE 2020), under the PORTUGAL 2020 Partnership Agreement, and through the European Regional Development Fund (ERDF).



Resumo

A evolução dos dispositivos móveis levou a uma nova geração de aplicações que exploram a sua mobilidade intrínseca, proximidade com o utilizador, e dependência de dispositivos manuseados. No entanto, no cenário tecnológico atual, as empresas enfrentam problemas associados com o desenvolvimento de aplicações móveis e a rápida integração dos seus sistemas corporativos para garantir serviços de valor acrescentado aos seus clientes. Para além disso, existem ainda problemas com a disponibilidade e qualidade das comunicações, estabelecendo-se como um obstáculo ao fornecimento de aplicativos altamente integrados em ambientes sempre conectados.

O problema de conectividade nos dispositivos móveis é generalizável e neste sentido pode ser criada uma abstração com reduzidas implicações no desenvolvimento e adaptação de aplicações existentes para criar a disponibilidade real das aplicações em ambientes altamente variáveis.

Para preencher esta lacuna, é proposto um modelo dinâmico e genérico para a mudança automática de canais de comunicação garantindo os princípios da transparência em ambientes distribuídos através de: (1) a concepção de uma camada de comunicação intercambiável para as aplicações móveis; (2) assegurar a transparência dos canais de comunicação; (3) uma framework para o desenvolvimento de aplicações móveis em ambiente distribuído para garantir a transparência das comunicações móveis.

Neste sentido e de forma a suportar o modelo proposto foi feita uma pesquisa junto de utilizadores de dispositivos móveis de várias idades e nacionalidades que permitiu aferir a sua importância e enquadrar a implementação do caso de estudo que validou o modelo e a arquitetura propostos usando metodologias de engenharia de software.

Palavras-Chaves

Aplicações Móveis, sempre disponíveis, mobilidade, disponibilidade, Mobile Cloud Computing

Abstract

Mobile devices led to a new generation of applications that exploit their intrinsic mobility, proximity to the user, and dependence on handled devices. However, in the current technological landscape, companies face problems associated with the development of mobile applications and rapid integration with their enterprise systems to ensure value-added services to its customers. Furthermore, there are still problems with the availability and quality of communications, establishing itself as an obstacle to the provision of highly integrated applications on always-connected environments.

The connectivity problem in mobile devices is generalizable and in this sense an abstraction can be created with reduced implications in the development and adaptation of existing applications to allow the services availability of the applications in highly variable environments.

To fill this gap, we propose a generic model for dynamic switching of communication channels to ensure the principles of transparency in distributed environments by: (1) designing an interchangeable communication layer for mobile applications; (2) ensuring the transparency of the communication channels; (3) proposing a framework for the development of mobile applications in distributed environment to guarantee the upper cited communication transparency.

Therefore and in order to support the proposed model a research is presented with mobile users of various ages and nationalities as a way of motivation for the current thesis, which allowed to measure its importance and framed the implementation of the case study that validated the proposed model and architecture using software engineering methodologies.

Keywords

Mobile applications, always-connected, mobility, availability, Mobile Cloud Computing

*“The world never stops. No matter how exhausted you feel, it keeps dragging you on and on
and there's no rest.”*

— Mark O'Sullivan, My Dad Is Ten Years Old

To my parents, sister, grandmother's, nephew and lady love

Acknowledgments

I would like to thank Professor Hugo Paredes and Professor Benjamim Fonseca for their supervision throughout my PhD. I thank them for their wise counsels, their efforts in trying to always keep me motivated, and their guidelines to always keep me focused on the right path.'

I want to especially thank my family, who was always been present, sharing the joys and encouraging me in the toughest times, to my parents and lady love for their continuous encouragement.

I thank all of those who in some way contributed to this PhD.

Table of Contents

A communication channels dynamic switching model for always-connected availability of service oriented mobile applications.....	iii
Resumo.....	vii
Abstract	ix
Acknowledgments.....	xiii
Index of Figures	xix
Index of Tables	xxi
Acronyms.....	xxiii
Chapter 1. Introductory Remarks.....	1
1.1. Introduction	2
1.2. Objectives	4
1.3. Thesis and Contributions.....	5
1.4. Methodology	7
1.5. Organization	10
Chapter 2. The Mobile Ecosystem: the devices, the applications and the communications	13
2.1. Introduction	14
2.2. Handled Mobile Devices.....	16
2.3. Mobile Applications: “Apps”.....	18
2.4. Communication Channels.....	24
2.5. Evolution of Networked Devices.....	32
2.5.1. Internet of Things.....	33
2.5.2. Web of Things	35
2.6. Conclusion	36
Chapter 3. Adaptation of Mobile Applications: a Literature Review	37
3.1. Introduction	38
3.2. Methodology for bibliographic review.....	38
3.3. Architectures for mobile applications development	39
3.4. Emerging Mobile Technologies: the applications-communications fusion.....	43
3.4.1. Mobile Computing and Pervasive Computing.....	44
3.4.2. Cloud Computing and Mobile Cloud Computing.....	46
3.4.3. Mobile Edge Computing and Fog Computing.....	47

3.5.	Conclusion	48
Chapter 4. Software as a Service for Mobile (SaaS4Mobile) Architecture		51
4.1.	Introduction	52
4.2.	Real world application scenarios.....	52
4.2.1.	SOSPhone.....	53
4.2.2.	MyCarMobile.....	55
4.2.3.	M3Medical	56
4.3.	SaaS4Mobile	58
4.3.1.	Life Cycle	58
4.3.2.	Planning and Design	59
4.3.3.	Installation and Tests	60
4.3.4.	Execution.....	62
4.3.5.	Maintenance	65
4.4.	Conclusion	66
Chapter 5. A model for Dynamic Switching of Communication Channels on Mobile Applications 67		
5.1.	Introduction	68
5.2.	Distributed Model Architecture	68
5.3.	Conceptual Model.....	70
5.4.	Model Logical Workability.....	72
5.5.	Conclusion	74
Chapter 6. A prototype for an always connected Mobile App: the Twitter Mobile App 77		
6.1.	Introduction	78
6.2.	Motivation Questionnaire.....	78
6.2.1.	Results Analysis	79
6.3.	The Twitter Mobile App	83
6.3.1.	Server Side.....	83
6.3.2.	Client Side.....	87
6.3.3.	Results Analysis – Unity Tests.....	99
6.4.	Conclusion	106
Chapter 7. Final Remarks and Future Work.....		107
7.1.	Final Remarks	108
7.2.	Future Work.....	110
References		113

Appendix 1 – Questionnaire..... 127
Appendix 2 – XML Services..... 131
Appendix 3 – XML Priorities..... 135
Appendix 4 – XML Rules..... 137
Appendix 5 – XSD Rules 139
Appendix 6 – XSD Priorities 141
Appendix 7 – XSD Services 143

Index of Figures

Figure 1: Design Science Research Cycles [Hevner and Chatterjee, 2010]	7
Figure 2: Genesis and evolution of Rich Mobile Applications (<i>source: Adapted from Abolfazli, et al., 2014</i>)	19
Figure 3: Evolution of mobile technology by generation, 1980 onwards adapted from [Intelligence G.S.M.A., 2014] AND 5G	27
Figure 4 IoT vs WoT based in [Guinard and Trifa, 2016]	36
Figure 5: Evolution of mobile devices novel paradigm	44
Figure 6: SOSPhone application interface	54
Figure 7: MyCarMobile application interfaces	55
Figure 8: M3Medical application interface	57
Figure 9: Application components interaction (mobile application, communication mediation service and third parties)	59
Figure 10: Planning, design, installation and tests phase's life cycle	61
Figure 11: Mediation service components (mobile applications, application package and routing package)	61
Figure 12: Execution phase life cycle	63
Figure 13: Runtime Engine – Functional Core Components Interface	64
Figure 14: Distributed model architecture	69
Figure 15: Model interaction within the distributed model architecture	71
Figure 16: Model cycle representing how rules are related to service and communication channels	71
Figure 17: Model logical workability when a service is requested	73
Figure 18: Participants degree	79
Figure 19: Relation between the importance of mobile devices while travelling and frequency of travelling abroad	80
Figure 20: Relation between the importance of use applications even without Internet and the time spent on the Internet	81
Figure 21: Relation between the time spent on the Internet and frequency of travelling abroad	82
Figure 22: Relation between the time spent on the Internet and the way users access the Internet abroad	83
Figure 23: GSM Modem – HUAWEI Mobile Connect Model E180	84
Figure 24: FormConnect – Server communication ports	84
Figure 25: GSManager – Server interface and runtime communication services response	85
Figure 26: TwitterServices – Server interface of the requested services content (twitter users and posts)	86
Figure 27: Mobile Device – HUAWEI Model U8510-1 with 2.3.3 Android OS Version	87

Figure 28: XML Services	88
Figure 29: XSD Services	89
Figure 30: XML Priorities	90
Figure 31: XSD Priorities	91
Figure 32: XML Rules	92
Figure 33: XSD Rules	93
Figure 34: Application Homepage with Internet Connection	94
Figure 35: Application Homepage without Internet Connection	95
Figure 36: Model Activation Confirmation	96
Figure 37: Application twitter Personal Information and features (post and search)	97
Figure 38: Twitter Personal Information on the Web	97
Figure 39: Twitter Post example	98
Figure 40: Twitter Post example on the Web	98
Figure 41: Twitter Search example	99
Figure 42: Testing approaches of unity tests, from [Whittaker, 2000]	100

Index of Tables

Table 1: Methodology process for bibliographic review39

Table 2: Taxonomy of RMA development problems and solutions, adapted from Abolfazli, et al. 2014.....23

Table 3: Evolution of technology generations in terms of services and performance (source: Adapted from Intelligence G.S.M.A. 2014)32

Table 4: Application settings of IoT technology’s application [Manyika, et al., 2015]34

Table 5: Mobile Devices Characteristics36

Table 6: Comparison of features provided by the novel paradigms48

Table 7: Switching assignments results102

Table 8: Availability assignments results104

Table 9: Time of response assignments results.....105

Acronyms

Acronym	Expansion
1G	First Generation
2G	Second Generation
3G	Third Generation
3GPP	3rd Generation Partnership Project
4G	Fourth Generation
5G	Fifth Generation
C2D	Cloud to Device
CDMA	Code Division Multiple Access
CPU	Central Processing Unit
D2C	Device to Cloud
D2D	Device to Device
DSP	Digital Signal Processing
ETSI	European Telecommunications Standards Institute
I&T	Integration Testing
IoT	Internet of Things
ITU	International Telecommunication Union
GPRS	General Packet Radio Service
GPS	Global Positioning System
GSM	Global System for Mobile Communication
HSDPA	High-Speed Downlink Packet Access
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
LTE	Long-Term Evolution
M2M	Machine-to-Machine
MCC	Mobile Cloud Computing
MMS	Multimedia Message Service
MNO	Mobile Network Operators
M3Medical	My Mobile Medical Appointment
NFC	Near Field Communication
OCC	Occasionally Connected Computing

OSI	Open System Interconnection
P2M	Point-to-Multipoint
P2P	Point-to-Point
PAN	Personal Area Networks
PC	Personal Computer
PoC	Push to talk Over Cellular
PPP	Point-to-Point Protocol
PSAP	Public-safety answering point
QoS	Quality of Service
RFID	Radio-frequency identification
RMA	Rich Mobile Applications
SMS	Short Message Service
TDMA	Time Division Multiple Access
UMTS	Universal Mobile Telecommunications System
USB	Universal Serial Bus
VoIP	Voice over Internet
W-CDMA	Wideband Code Division Multiple Access
WAP	Wireless Application Protocol
WAN	Wide Area Network
WoT	Web of Things
WDP	Wireless Datagram Protocol
WLAN	Wireless Local Network

Chapter 1. Introductory Remarks

In this introductory chapter the major motivation and problems of this research are addressed. The main objectives are presented, a summary of the thesis and the major contributions of the research are described. The detail of the research methodology applied in the course of the study is introduced, by summarizing the major steps undertaken. The chapter ends with the presentation of the thesis structure, including a brief description of each chapter.

1.1. Introduction

The advent of laptops and smartphones has created a multitude of resources for mobile systems, continually changing elements such as Central Processing Unit (CPU), keyboard, monitor, power/autonomy available and communication channels. Mobile wireless Internet has been rapidly developed [Fangxiong and Zhiyong, 2014] and has increased further with the introduction of new wireless networks that allow access to information “anywhere at anytime”. These technological advances in mobile devices and the inclusion of the possibility of wireless communication enable mobile users to make use of more advanced mobile applications [Jing and Karen, 1998]. Small and medium sized business have their services delivered over the Web, because it has more affordable fees, and greater efficiency in delivering services to massive number of receivers [Huang, et al., 2010]. However, in the current technological landscape, companies face problems associated with the speed of development of mobile applications and rapid integration with their enterprise systems in order to provide value added services to its customers. These problems translate into high costs and loss of competitiveness leading to a slow approach of time-to-market. Thus, construction of specific applications for the diverse number of mobile frameworks and platforms integrated with specific services becomes costly, slow and individualized [Fangxiong and Zhiyong, 2014] [Bhagwan, et al., 2003] [Couderc and Kermarrec, 1999].

Internet-distributed systems offer a serious platform for stable, long-lived, flexible applications development. Internet-scale applications are motivating new modes of work and company integration such as teleworking and virtual enterprises. Simultaneously, the Internet is expanding to incorporate traditional services associated with services provided to stream video calls over IP, mobile and TV. In this sense, the Internet is becoming part and parcel of everyday life, which manifests a clear need to provide always-available applications and services to user context across different network types. However, a major challenge for such mobile distributed applications is to provide stability in the face of constant change, providing application developers and users alike with a stable architecture whose elements and locations may change over time. Furthermore, there is an increasing realization that within such flexible working structures those applications will have to support some form of mobility, both to handle explicitly mobile nodes and to provide reconfiguration and redeployment of application components around the Internet. These systems must be able to support the mobility of users and their mobile devices as they move spatially using different networks and hardware

specifications. The current location of the mobile user determines the execution context that will be available, since the mobile networks are heterogeneous by nature. Thus, it can be said that the fundamentals for mobile distributed applications are (i) the current location, which in turn determines (ii) the current context which requires the feature and (iii) adaptation of the application to the variety of possible configurations within a mobile environment. Hence, mobile users goals are also limited due to the device reachability being highly variable, the location of host and connection capabilities being frequently changing, limited wireless bandwidth, and poor internet connections that affect the connection with centralized servers and even access to web applications [Mei, et al., 2008] [Fernando, et al., 2013] [Kaur and Makkar, 2016].

One effective tool when mobile users require searching services, such as information, location, data, images and so on, is the Cloud [Ding, et al., 2013]. The Cloud is known to be a promising solution for Mobile Cloud Computing (MCC) because of mobility, communication, portability and others [Ding, et al., 2013] [Forman and Zahorian, 2014]. Briefly, in [Ding, et al., 2013] [Satyanarayanan, 1996] [Ali, 2009], it is defined that MCC provides mobile users with the data processing and storage services in clouds, and mobile applications do not need a powerful configuration because all complicated computing modules can be processed in the cloud. These new advances in mobile computing devices allow us to gather new mobility approaches to develop a new set of adaptable mobile applications depending on the environment limitations. At the same time, the growth rate of existing technology environment don't match with the increasing customer requirements, the massive growth data, the heterogeneity of networks, and the distributed resources [Li, et al., 2010] [Jie, 2010]. The major problems associated with mobile applications for mobile users, companies and developers are: mobile applications must be "always-connected" and mobile applications services must be available "anywhere at anytime" [Ding, et al., 2013] [Satyanarayanan, 1996] [Leporini, et al. 2011]. The availability – the quality of being present or ready for immediate use [Bhagwan, et al., 2003] – of communications services (roaming, speed data network, and so on), also constitute itself as an obstacle to the provision of valued-added services for highly integrated applications in pervasive environments. This seemingly simple definition of availability can conceal tremendous complexity, since the mobile applications services availability is related to a set of attributes, such as the services itself, communication channels, environment limitations, and so on. Thereafter, the most prominently cited feature of mobile services is the time efficiency –

that results from the ubiquitous nature of Internet and is directly related with services availability enabling consumers to use services “anytime and anywhere” [Nysveen, et al. 2005].

Mobile applications must be designed, in all its aspect, with mobility and adaptability in mind [Couderc and Kermarrec, 1999]. Many researchers consider the ability to adapt the most appropriate way of dealing with frequent changes of resources and services that occur in a mobile environment [Katz, 1994] [Davies, et al., 1997]. In particular, it would be very useful for users to move from one place to another having all his data and applications available, in the same state, and without any intervention by mobile users (a real ubiquitous application). Therefore, this research focus on the management of data communication channels that at each moment depending on the environment limitations, could result in a better availability of the mobile applications services, providing therefore “always-connected” mobile applications and mobile applications services available “anywhere at anytime”.

1.2. Objectives

In the present context of mobile devices paradigm there is a need to have access “anywhere and anytime” to data services. In this sense, the main objectives of this research is the development of a dynamic model for management of the data communication channels that depending on the environment limitations, could result in a better availability of the existing mobile applications services. The generalization of the model would allow a mobile application to have their services available “anywhere at anytime” and give the better response to the mobile user’s needs, which leads to the customer satisfaction.

In order to validate the model, experiments were conducted, through case studies where a mobile application based on the proposed model was developed.

The primary objective achievement of this thesis is made through the establishment of partial objectives to be realized in various stages of the overall development. A partial list of targets to the primary objective is presented:

- Study of the main features of mobile devices and analysis of the major problems associated with the mobile applications services availability. Definition of key concepts related with the paradigm of mobile devices, mobile applications and mobile communication channels.

- State of the art study of models that allow the development, the effectiveness and the availability of mobile applications services.
- Conceptual model evaluation roles in mobile applications development. Identification of the main models for the implementation of mobile applications.
- Specification and characterization of a platform for the rapid development of "always connected" mobile applications, which aims to address the inherent needs in distributed mobile applications development.
- An innovated and dynamic model for the management of mobile applications services is proposed, through a dynamic layer manager of communication channels of mobile devices, which ensures services availability on mobile applications regardless of the environment limitations.
- Model specification of the dynamic layer communication channels management used by mobile devices in mobile applications, enabling the creation of rules and priorities to ensure services availability.
- Implementation of the dynamic model to ensure the development of a mobile application based on the model specification.
- Plan, perform and assess experiments in a prototype to validate the proposed dynamic model.
- Results analysis and roadmap for future work, which allows the system development and encompass potential evolution that have not been totally explored, taking into account the expected standards and technological evolution.

1.3. Thesis and Contributions

The major contribution of this thesis is a model and its prototype implementation that represent a solution to fulfil the existing gap in mobile applications services availability. Thereafter, the solution proposed in this work solves the problems presented, providing a model for a service brokering communication that enables the rapid integration of enterprise applications with mobile applications, using an adaptive and transparent communication layer to ensure full connectivity and reduce costs associated with maintaining this type of platforms. Furthermore, by taking advantage of the dynamic communication channels it is now possible to have an always-connected application despite of the lack of Internet connection.

In this research a set of unanswered questions must be addressed for the mobile community in order to solve existing mobile communication issues. Therefore a list of research questions is hereby presented and further solved as the document evolves:

(RQ1) – What is the need for a device to be always-connected?

(RQ2) – What is the dependence of people on the perception of connectivity?

(RQ3) – It is possible for a device to be virtually always connected?

(RQ4) – What is the ability to generalize the existing connectivity problems with mobile devices in highly instable environments, by abstracting the communication layer from developers and ensure that the priority services are still available?

(RQ5) – Can the mobile applications features or services be classified as principal and secondary, to guarantee the availability of those that have higher priority?

(RQ6) – It is possible to define a model that adapts the mobile application features based on the switching of the available communication channels?

(RQ7) – What are the model implications in applying it to existing frameworks and to real world applications scenarios, and also in terms of performance issues?

The answer to these research questions will be primarily used to address the proposed thesis:

The connectivity problem in mobile devices is generalizable and in this sense an abstraction can be created with reduced implications in the development and adaptation of existing applications to allow the services availability of the applications in highly variable environments.

In summary, a generic model for dynamic switching of communication channels is proposed to ensure the principles of transparency in distributed environments by: (1) designing an interchangeable communication layer for mobile applications; (2) ensuring the transparency of the communication channels; (3) proposing a framework for the development of mobile applications in distributed environment to guarantee the upper cited communication transparency.

1.4. Methodology

In this thesis a generic model for dynamic switching of communication channels in mobile applications is proposed where the research methodology followed was the Design-Science Paradigm [Hevner and Chatterjee, 2010]. This paradigm follows three life cycles – Figure 1 (The Relevance Cycle – bridges the contextual environment of the research project with the design science activities, The Rigor Cycle – connects the design science activities with the knowledge base of scientific foundations, experience, and expertise that informs the research project and The Design Cycle – iterates between the core activities of building and evaluating the design artifacts and processes of the research) and addresses important unsolved problems in unique or innovative ways or solved problems in more effective or efficient ways [Hevner and Chatterjee, 2010] [Von, et al., 2004].

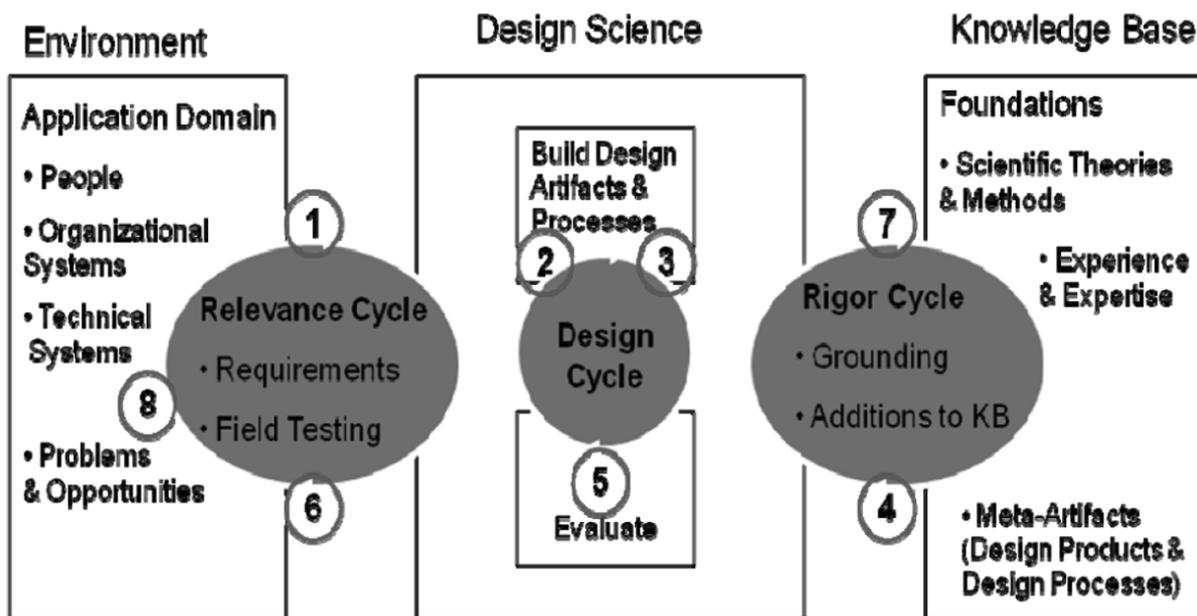


Figure 1: Design Science Research Cycles [Hevner and Chatterjee, 2010]

1. **The Rigor Cycle.** Provides past knowledge base of scientific research project, theories and methods to ensure its innovation. It considers both the state of the art of the solution domain, models matching, and the methods used to substantiate the research.

In our case, the theoretical basis is addressed in Chapter 2, while the state of the art is provided by means of the analytical review summarized in Chapter 3.

2. ***The Design Cycle***. Generates design alternatives and evaluates the alternatives against requirements until a satisfactory design is achieved. The design alternative is to obtain an artifact, such as models, practices, and so on. The evaluation is to obtain essential feedback to refine the artifact in the following interaction of the design alternative.

In this research, the design alternative is summarized in Chapter 4 and 5, while the evaluation phase is presented in Chapter 6.

3. ***The Relevance Cycle***. Initiates design science research with an application context that provides the problem to be addressed as inputs and defines acceptance criteria for the ultimate evaluation of the research results.

In our particular case the application domain is the mobile devices environment whose services availability problem is tackled by means of a dynamic model for mobile communication channels. The specification and requirements for improvement of services availability in mobile devices is addressed in Chapter 5. Further, this environment is used for the validation of the proposal, as detailed in Chapter 6.

The design process of the *Design-Science Paradigm* methodology is a sequence of expert activities that produces an innovative artifact that addresses research through the building and evaluation of the artifact designed to meet the identified needs [Couderc and Kermarrec, 1999]. The evaluation of this artifact then provides feedback information and a better understanding of the problem in order to improve both the quality of the artifact and the design process. This built-and-evaluate loop is typically iterated a number of times before the final artifact is generated.

Therefore, in order to tackle this research, the *Design-Science Paradigm* proposes seven guidelines for the construction of a good design-science research [Von, et al., 2004], namely:

Guideline 1 – *Design as an Artifact*:

As product of this thesis, several artifacts have been obtained, being the foremost, the model for dynamic switching of communication channels for improvement of mobile devices services availability, presented in Chapter 5. However, other artifacts also obtained are the distributed architecture, described in Chapter 4, the case study developed and the assessment for the evaluation of the model through case studies, described in Chapter 6.

Guideline 2 – *Problem relevance*:

The problem relevance of the model focus on the limitations of client-server connectivity in distributed mobile applications, in order to improve services availability in always-connected mobile applications.

Guideline 3 – *Design evaluation:*

The utility, quality, and effectiveness of the model must be rigorously demonstrated via well-executed evaluation methods, therefore two design evaluation methods were chosen to assess and validate the proposed model, such as a Questionnaire addressed in Chapter 6 and a controlled experiment to study the model behaviour on changing communication environments (for example, the availability of services during a loss of internet connection), as described in Chapter 6, which have been developed based on the proposed model).

Guideline 4 – *Research contributions:*

In this thesis, a clear and systematized identification of the contributions is presented in the previous sub-section and in Chapter 7, based on novelty and significance of the obtained artifacts. In summary, the design artifact contribution of the proposed solution is the model itself, which tackle the unsolved-problem of mobility on mobile applications and lead their development process to contribute to the improvement and efficiency of services availability on always-connected mobile applications. However, the model itself does not compose all contributions of the proposed solution because the ability to transparently and dynamically switch between communications channels on distributed client-server mobile applications with pre-built rules and adaptability of the mobile applications to show only the features available depending on the environment limitations, without any intervention by the mobile users, are also important contributions.

Guideline 5 – *Research rigor:*

Design science research relies upon the application of rigorous methods in both the construction and evaluation of the design artefact, the model itself, therefore it is imperative to understand why the artifact work or does not work to enable new artifacts to be constructed that exploit the former and avoid the latter. Additionally, the proposed artifacts have all been formally described in Chapters 4, 5 and 6.

Guideline 6 – *Design as a search process:*

The search for an effective artifact requires utilizing available means to reach desired ends while satisfying laws in problem environments. So, to design this artifact, a cycle will need to be followed, such as: observe, identify problems, solutions posed and implemented, while generating design alternatives and then test those alternatives against the requirements/constraints. In this sense, the required background knowledge of the application domain, mobile devices and the solution domain, models matching, are concisely presented respectively in Chapters 2, 3, 4 and 5.

Guideline 7 – *Research Communication:*

As proof of validity of the contributions, they were presented and published in international conferences, and therefore, verified by the research community. The contributions address a heretofore unsolved problem dealing with the design of a model to support the current limitations of services availability on always-connected mobile applications in distributed environments, thereby this thesis was written to support the communication research, to validate the background and state of the art, to assess the model itself, and to validate and assess the model through case studies.

1.5. Organization

In addition to the current chapter that presents the motivation, objectives and contributions of the research, and methodology this thesis presents a set of seven chapters organized as follows:

- **Chapter 2**

The Mobile Ecosystem: the devices, the applications and the communications

Presentation of the current mobile devices paradigm, by identifying its main features, communication channels and lines of inquiry. Definition of key concepts associated with the theme of mobile devices paradigm.

- **Chapter 3**

Adaptation of Mobile Applications: a Literature Review

Summary of the main architectures for the development of distributed environments, technological solutions for its implementation and related models. Alongside and given particular emphasis on distributed architectures, including mobile architectures and dynamic models for switching of communication channels.

- **Chapter 4**

An Architecture for Software as a Service for Mobile (SaaS4Mobile)

General description and implementation details of the architecture that supports the dynamic model, designed to implement the model and allow its validation. The presentation of the architecture follows the principles defined by the architectural views, starting to be given a general perspective of the system, being progressively detailed each of its components up to the level of its implementation. The main technological solutions used in the implementation are also addressed.

- **Chapter 5**

A model for Dynamic Switching of Communication Channels on Mobile Applications

Model for dynamic switching of mobile communication channels on the development of mobile applications, in order to guarantee transparency and at the sometime improve the efficiency of services availability. The model is based on services, communication channels and a set of rules defined by developers to interconnect the priority of the available communication channels and the supported features of the installed services on the mobile application.

- **Chapter 6**

A prototype for an always connected Mobile App: the Twitter Mobile App

Validation of the dynamic model and supported architecture through the implementation of case studies that were the basis for conducting experiments. The assessment of the dynamic model and architecture is given based on qualitative and quantitative measurements of services availability on a mobile application.

- **Chapter 7**

Final Remarks and Future Work

Overview of the most relevant points of the research. Critical analysis and assessment of the model and architecture. The conclusions of the work are addressed, with particular emphasis on the achievement of the proposed objectives. Indication of potential lines for future research in order to continue the work.

Chapter 2. The Mobile Ecosystem: the devices, the applications and the communications

This chapter introduces the key concepts associated with applications and communication channels for mobile devices. First of all, in section 2.1, the context and evolution of mobile devices used in the context of this thesis is provided. Then, in section 2.2, a brief description of the mobile devices field is introduced, by presenting its characterization. In section 2.3 the most common communication channels used by mobile devices are described. Then, in section 2.4, the mobile applications paradigm is addressed, followed by their characteristics and main features. Finally, a short summary in section 2.5 concludes this chapter.

2.1. Introduction

“We always overestimate the change that will occur in the next two years and underestimate the change that will occur in the next ten. Don’t let yourself be lulled into inaction”. Bill Gates

Nowhere can this quote be more true than with respect to the rise of the mobile Internet. In 2007, fixed broadband had just surpassed dialup as the main form of Internet access; one billion users accessed the Internet, the majority from developed countries; and it would be another two years before the iPhone was launched and four years before the first 4G network was deployed.

Mobile devices and mobile applications are currently at the top of innovation, and it has been one of the major advances in technology for the last years. The first set of smartphones were often just for email or were limited to stylus interaction, due to their unsophisticated touch screens and system capabilities [Fuggetta, et al., 1998]. The Short Message Service (SMS), used by mobile devices and provided by Mobile Network Operators (MNO) has changed the way people communicate everywhere in the world. Market researchers presented a globally delivered of 21 billion SMS per day in 2014 [Gerpott, 2015]. The popularity of SMS has even led to concerns about an excessive integration of this service in people’s daily lives, particularly among younger MNO customers [Lu, et al., 2011] [Skierkowski and Wood, 2012]. Communication services are the most elementary and well-known type of mobile services. While voice-to-voice applications traditionally were the primary service in wireless technology, SMS were slowly becoming the predominant standard [Igarashi, et al., 2005] [Reid and Reid, 2004]. The possibilities of SMS were extended by the introduction of Multimedia Message Service (MMS) that allow for the attachment of audio, video, and pictures in addition to the traditional textual message.

The benefits from the advancements of mobile devices, mobile platform, and wireless network technologies [Wang, et al., 2009] [Borst, et al., 2009] [Kao, et al., 2007] [Atmaca, et al., 2009] [Moreno and Oliveira, 2009], are making the market of mobile devices growing dramatically: an increasing number of people are using at least one mobile phone, or even tablet PC, daily. Unlike traditional cellphones, users are not only capable of making phone calls or sending and receiving short messages, but can also install applications from the online markets. Apple iOS, Google Android and Windows Phone [Hall and Anderson, 2009] [PCs, 2008] [Chang, et al., 2009] are three major examples of this type of mobile device platforms [Kao, et al., 2012]. Apple opened its App Store in 2008 and started out with just 500 apps but within 3 months it

had seen 100 million downloads and the number of apps jumped to 3000. By March 2013, Apple hit the 50 billion-download mark, and 85 billion towards the end of 2014. As for apps, Apple in 2015 has around 1.4 million iOS apps available for download. Google launched the Android Market in 2008 with only a handful of apps, but by 2012 the number of apps surpassed 500,000 for the first time and the total number of downloads reached 25 billion. Google's store managed to surpass Apple's, in terms of the number of apps available, towards the end of 2014. As for 2015, the Wall Street Journal has reported that Google Play had 70% more app downloads than Apple's App Store in the first quarter of 2015, but Apple's app revenue was about 70% higher than on Google Play [Peña-López, 2015].

Today, the changes that have taken place would have been hard to fathom ten years ago. We highlight a few key statistics and trends [Peña-López, 2015]:

- 3 billion Internet users by May 2015.
- Mobile Internet penetration is forecast to reach 71% by 2019.
- Usage per device is forecast to more than triple by 2019.
- 192 countries have active 3G mobile networks, which cover almost 50% of the global population.
- Smartphone sales are the majority of mobile handsets sold worldwide; tablet sales will soon exceed total PC sales.
- While there are at least five mobile platforms, Android has an 84% share of smartphones, and 72% of tablets.
- There are well over 1 million apps available, which have been downloaded more than 100 billion times.
- Time spent using apps exceeds time spent on mobile browsers, and in the US, at least, exceeds time spent on desktop and mobile browsers combined.

2.2. Handled Mobile Devices

Falling under the category of mobile devices they serve multiple purposes for today's electronic consumers. Years ago, cellular phones, also called cellphones or mobile phones, served one purpose only: they let users make voice communications. Today, mobile phones and similar devices are equipped with customized software, Internet access, digital cameras, portable music players, Global Positioning System (GPS) functions and many more options. Once again technological advancements make common terminology such as mobile phone, Personal Digital Assistant (PDA), smartphone, tablets and wearables difficult to decipher as each type of device changes constantly and features traditionally belonging to one type of device are now found on others.

A mobile device is defined as a small computing device, typically small enough to be handled, and having a display screen with touch input and/or a miniature keyboard and weighing less than 2 pounds (0.91 kg). However, a mobile device is also a handheld computing device that has an Operating System (OS), and can run various types of application software, known as apps. Most handheld devices can also be equipped with Wi-Fi, Bluetooth, Near Field Communication (NFC) and GPS capabilities that can allow connections to the Internet and other devices, such as an automobile or a microphone headset or can be used to provide location-based services. Increasingly mobile devices also contain sensors like accelerometers, compasses, magnetometers, or gyroscopes, allowing detection of orientation and motion.

Mobile Phone

A mobile phone is more frequently called a cellular phone, cellphone or feature phone. These communication devices connect to a wireless communications network through radio waves or satellite transmissions. Most mobile phones provide voice, SMS, MMS, and newer phones may also provide Internet features such as Web browsing, instant messaging capabilities and e-mail. In an effort to provide parity with smartphones, modern mobile phones have also incorporated support for 3G connectivity, touchscreens, and access to popular social networking services. However, their functionality and support for third-party software are still relatively limited in comparison to smartphones.

Smartphone

Smartphones allow users to store information, e-mail, and install applications, along with using a mobile phone in one device. A smartphone's feature is usually more oriented towards mobile phone options than the PDA-like features. Any mobile device that has more than basic mobile phone capabilities can actually be filed under the smartphone category of devices. A complication in distinguishing between smartphones and mobile phones is that over time the capabilities of new models of mobile phones can increase to exceed those of phones that had been promoted as smartphones in the past. For example, today's mobile phones typically also serve as a PDA and portable media player and have capabilities such as cameras, touchscreen, GPS navigation, Wi-Fi and mobile broadband Internet access, and even mobile gaming.

Personal Digital Assistant

A typical PDA can function as a mobile phone, fax sender, Web browser and personal organizer. These devices are usually pen-based, which requires the use of a stylus rather than a keyboard for input. A smartphone is considered to be the combination of the traditional PDA and mobile phone, with a bigger focus on the mobile phone part. These handheld devices integrate mobile phone capabilities with the more common features of a handheld computer or PDA.

Tablets

A tablet is a mobile computer with a touchscreen display. It is equipped with sensors, cameras, microphone, accelerometer, and the touchscreen display uses the recognition of finger or stylus gestures replacing the usage of the mouse and keyboard. Tablets are typically larger than smartphones and can be classified into several categories according to the presence and physical appearance of keyboards. Slates and booklets do not have a physical keyboard and typically feature text input performed through the use of a virtual keyboard projected on a touchscreen-enabled display. Hybrids and convertibles do have physical keyboards, although these devices typically also make virtual keyboards available.

Wearables

Wearable technology, wearable's, fashionable technology, wearable devices, tech togs, or fashion electronics are clothing and accessories incorporating computer and advanced electronic technologies. The designs often incorporate practical functions and features, but may also have a purely critical or aesthetic agenda. Wearable devices such as activity trackers are a good example of the Internet of Things (IoT), since they are part of the network of physical objects or "things" embedded with electronics, software, sensors and connectivity to enable objects to exchange data with a manufacturer, operator and/or other connected devices, without requiring human intervention.

Mobile Devices

Cellphones, PDA's, Smartphones, Tablets and Wearable's can also be called of Mobile Devices if they follow a set of characteristics, such as: OS, Support of Apps, Connectivity (Wi-Fi, Mobile Data, GSM and Bluetooth) and Hardware. These features/characteristics of mobile devices, namely the installation of apps and the connectivity, are essentials for this thesis in order to apply the purposed model.

2.3. Mobile Applications: “Apps”

The introduction of mobile devices brought a new type of portable computer with a number of advanced features, which are accessible through apps distributed through online stores. It is clear that the mobile Internet play a key role in bringing the billion users online. Mobile Internet has already leap-frogged fixed access in many countries because of limitations in the coverage of the fixed network, and the availability of mobile Internet access significantly outpaces adoption today. The mobile Internet is therefore central to realising the Internet Society vision that “*The Internet is for everyone*”. Furthermore, the benefits of the mobile Internet are arising from new innovative services based on mobile access to the Internet, using all the features embedded into the smart devices, and accessed through apps. These services enable social inclusion, interaction with government, and commerce, among other applications. These innovations are already driving a further evolution of the Internet that has been in a state of constant change since its founding. Figure 2 [Abolfazli, et al., 2014] represents the gradual development of candidate application architectures from early desktops until recent years from left to right that ends with Rich Mobile Applications (RMA).

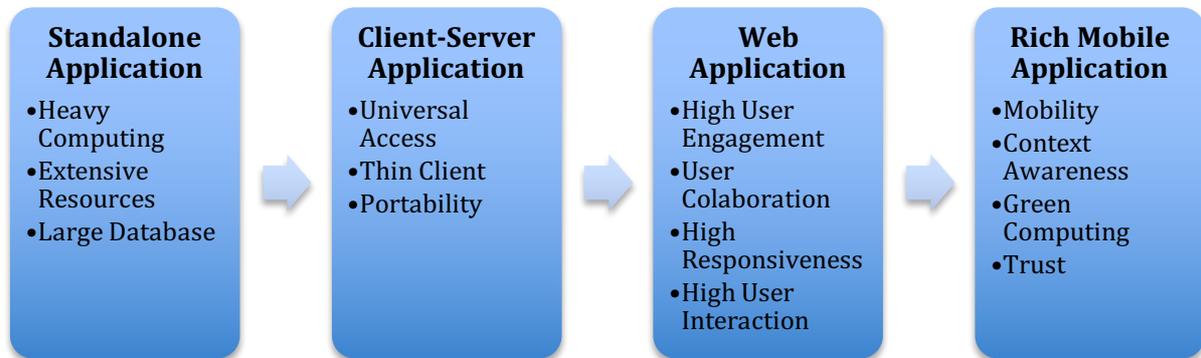


Figure 2: Genesis and evolution of Rich Mobile Applications (*source: Adapted from Abolfazli, et al., 2014*)

These apps, taking advantage of the advanced features of smart devices and the full mobility of users, have provided benefits in every part of our lives:

- Entrepreneurs have created billion dollar opportunities developing and selling apps worldwide.
- Users' livelihoods can be enhanced as farmers, fishers, and others use apps to increase their output and earnings.
- Opportunities abound for general education, and to help learn a trade through the mobile Internet.
- Users with disabilities can rely on mobile services to communicate, work, and shop, among other activities.
- Governments are increasingly using the mobile Internet to communicate with their citizens and make information available. They are accomplishing it by creating smart cities that deliver infrastructure and communications more efficiently.
- A variety of mobile healthcare applications are emerging for users to track their own fitness or enable remote diagnostics.
- Smart devices can be used to send automatic alerts when the user's personal security is in danger.
- As with many new electronic devices, the mobile Internet is used in everyday life.

- The sensors in smart devices, including not just location, but also barometers, accelerometers, and others, enable them to be a part of the emerging IoT.

Small and medium sized business have their services delivered over the Internet, because it has more affordable fees and greater efficiency in delivering services to massive numbers of receivers [Huang, et al., 2010]. However, in the current technological landscape, companies face problems associated with the speed of development of mobile applications and rapid integration with their enterprise systems in order to provide value added services to its customers. These problems translate into high costs and loss of competitiveness leading to a slow approach of time-to-market. Also building mobile applications brings problems associated with it because of the number of different mobile platforms in the market and building different applications for each native language is very expensive [Charland and Leroux, 2011] [Pradhan, 2011], time consuming [Pradhan, 2011], and user demanding [Chu, et al., 2004] [Lange, et al., 2011].

Creating an interesting application that can be executed on different mobile devices within a very short time is a significant issue of application development. However, developers used to usually write mobile applications in a specific programming language for a specific mobile platform; that is, iPhone applications in Swift [Apple, 2017], and Android applications in Java or C#. Since applications written in Swift could not be executed on the Android platform, developers had to spend twice the time writing the same application in two different languages. Thus, construction of specific applications for the diverse number of mobile frameworks and platforms integrated with specific services becomes costly, slow and personalized [Chu, et al., 2004] [Lange, et al., 2011]. Therefore, due to the heterogeneity of mobile devices, a ubiquitous application is demanded to be developed in various versions to provide consistent quality of user experience, such as a cross-platforms framework solutions that follow a flow driven or model-based approach, capable of deploying for the existing mobile devices, in providing dynamic communication features, and maintaining the reliability of the application depending on the environment. Furthermore, other important aspect of user experience that had influenced the motivation of this research is the goal of mobile users, such as: access to information regardless of their physical location, willing to stay connected while on the move, no time to wait or search.

If the developing process of a mobile application is considered, there are some relevant aspects that need to be explored:

- 1) **Web**: The mobile Web refers to the use of browser-based Internet services from handheld mobile devices, such as smartphones or feature phones, through a mobile or other wireless network.
- 2) **Native**: The term "native app" is often mentioned in the context of mobile computing because mobile applications have traditionally been written to work on a specific device platform. A native app is installed directly on a mobile device and developers create a separate app version for each mobile device. The native app may be stored on the mobile device, or it can be downloaded from a public or private app store and installed on the mobile device. Data associated with the native app is also stored on the device, although data can be stored remotely and accessed by the native app.
- 3) **Hybrid**: Most embedded devices use only software installed directly on the device, but mobile devices often include applications that invoke services over the telephone network or the Internet via a web browser and affect data and displays on the device. Within the mobile computing sector, Web apps are sometimes contrasted with native apps, but the two are not mutually exclusive because many applications contain elements of both native and Web apps.
- 4) **RMA**: are online mobile applications originated from the convergence of mobile cloud computing and imminent communication technologies that deliver rich user experience via high functionality (offline usability, context-awareness, portability and data ubiquity), immersive interaction, and brittle response in secure wireless environment [Abolfazli, et al., 2014].
- 5) **OCC – Occasionally Connected Computing** is a term used in computing for an architecture or framework that permits running some aspects of an application when not connected to the Internet. This is sometimes a feature of a Rich Internet application (RIA) that is now being implemented in RMA (Facebook Mobile App is the most recent example of OCC). Occasionally connected computing is a software architecture based on the idea that an end user should be able to continue working with an Internet application even when temporarily disconnected or when a wireless connection fails or is otherwise unavailable. Apps that continue to work while “offline” or “disconnected” have been important since laptops first became common. A classic example is a salesperson that wants to enter data

while travelling, and then have that data automatically sync with a server when they get back to the office. Even with Wi-Fi and cellular broadband, there are many scenarios (airplanes, Wi-Fi only devices, and remote areas) where people still need to work without a connection. To build such OCC applications there are three core capabilities to be aware, such as browser application caching pages (allows a manifest to be created listing pages that should be cached and made available offline), local storage (retains local web app data using a key/value system that works similarly to browser cookies) and local database (removes the 5MB limit of local storage and allows data to be indexed so that multiple properties can be queried quickly).

The distinction between mobile Web applications and native applications is becoming increasingly blurred because mobile browsers gain direct access to the hardware of mobile devices (including accelerometers and GPS chips), and the speed and abilities of browser-based applications improve. Persistent storage and access to sophisticated user interface graphics functions may further reduce the need for the development of platform-specific native applications. Mobile Web access today still suffers from interoperability and usability problems. Interoperability issues stem from the platform fragmentation of mobile devices, mobile operating systems, and browsers. Usability problems are centred on the small physical size of the mobile phone form factors (limits on display resolution and user input/operating). Furthermore, web applications are slower to execute when compared to native applications because they absolutely require compatible web browsers that are dependent of the bandwidth, for example a drawback very usual is if a browser vendor decides not to implement a certain feature, or abandons a particular platform or operating system version, current users will have huge problems. All the development process of native applications takes a lot of time, money, there are just a few solutions, and applications and services provide a wide variety of functionalities. In order to access these applications from different devices via different modalities, the actual user interface implementation has several challenges [Oviatt, et al., 2004], such as the set of mobile devices that are supported (iOS, Android and WindowPhone).

From web to native mobile applications there are metrics that are transversal to all mobile applications, such as architecture, communication technology, portability, trustworthy, usability and online connectivity. These metrics are summarized in Table 1 and explained bellow.

Metrics/App	Native	Web	Hybrid	RMA	OCC
Architecture	Multi-tier, smartphone-nested applications	Multi-tier, web application	Multi-tier, web application	Multi-tier, smartphone-nested applications	Multi-tier, web application
Communication technology	Synchronous/Asynchronous communication				
Portability	Low in standalone applications	High in browse-based applications	High in browse-based applications	Low in standalone applications	High in browse-based applications low in standalone applications
Trustworthy	Low due to wireless connectivity, Internet and mobility	Medium due to wired connectivity and Internet	Medium due to wired connectivity and Internet	Low due to wireless connectivity, Internet and mobility	Medium due to limited services availability
Usability	Intermittent online and limited offline (low bandwidth and limited storage)	Online and offline (high bandwidth and large storage)	Online and offline (high bandwidth and large storage)	Intermittent online and limited offline (low bandwidth and limited storage)	Intermittent online and limited offline (low bandwidth and limited storage)

Table 1: Taxonomy of RMA development problems and solutions, adapted from Abolfazli, et al. 2014

Web and native applications are multi-tier online applications, in web applications the logic and data layers are located in the back-end server and the presentation is migrated to the client machine. However, in native applications is kept a partial copy of the application in the mobile device and the remaining components are distributed among private servers or public resources, particularly clouds. There is also the particular case of native applications being used locally, which allows them work offline, but in case of using remote services the working process is similar to web applications. Both web and native applications leverage asynchronous communication mechanism, from a technological communication perspective, for enhancing the quality of communication in online applications [Campeato and Nilson, 2011] [Abolfazli, et al., 2014].

In the absence of portability trait of mobile applications, developers should write different versions for a single mobile application platform, which imposes financial costs and increases development time [Sanae et al, 2014]. If trustworthy considered, the security and privacy are crucial for mobile devices users including location information, online behaviour, and stored data at a time when mobile hacking and malware-contaminated applications are increasing drastically [Cachin and Schunter, 2011] [La Polla et al, 2013 [Abolfazli, et al., 2014]. Therefore,

enhancing security and privacy provision in resource-constrained mobile devices tranquilize users and enrich their mobile computing experience, but decrease computing performance due to extra security-related computations and controlling overhead [Almenares et al, 2013] [Abolfazli, et al., 2014]. In terms of application usability, offering online and offline application functionality and access to content in web and native is requested to fulfil user requirements and deliver user experience. A reliable communication facilitates online and offline functionalities and content delivery in immersive computers. However, wireless communications of mobile devices are comparatively low-bandwidth, error-prone, and intermittent with long WAN latency [Abolfazli, et al., 2014]. Disconnecting from the network restricts application usability in the absence of offline functionality and local data storage. Moreover, multi-tier architecture of native applications enables distribution of task between native device and remote servers resulting in seamless ubiquitous functionality and data access that can realize the true feel of real time well-connected mobile applications to mobile users. In the event of network failure, techniques like widgets push messaging, and partial cached data are just-in-time reactions that convey a sense of connectivity to users.

2.4. Communication Channels

Mobile Web refers to the use of browser-based Internet services from handheld mobile devices, such as smartphones or feature phones, through a mobile or other wireless network. Traditionally, access to the World Wide Web has been via fixed-line services on laptops and desktop computers. However, the Web is becoming more accessible by portable and wireless devices. The shift to mobile Web access has been accelerating with the rise since 2007 of larger multi-touch smartphones, and of multi-touch tablet computers since 2010. Both platforms provide better Internet access, screens, and mobile browsers or application-based user Web experiences than previous generations of mobile devices have done.

Recent years have seen a surge in mobile data traffic, based on the increase in the number of users, and data consumption generated by users through all the possibilities of the mobile Internet. Traffic will continue to grow at a fast pace as mobile network deployments expand to cover entire countries, and as networks upgrade to 3G and 4G technologies and beyond. At the same time technologies at the edge, namely 5G are being prepared to attack the markets.

The availability and growth of mobile Internet services is critically dependent on access to a finite resource – radio frequencies or spectrum. All devices offering mobile Internet operate in

a similar way, generating and transmitting a signal at a specific radio frequency. Once transmitted from a device, it is captured by the nearest base station and then transferred to the Internet. Return traffic is transmitted from the base station to the receiving device, which then decodes the signal into the required data. Spectrum is at the heart of many commercial services, including radio and television, government services including safety networks and military communications, and personal services such as ham radio. In many cases, but not all, a frequency band must be made available on an exclusive basis for a particular service, and often to individual providers of the service, in order to prevent interference and provide an incentive to invest in the service. As spectrum is a scarce resource, and given that the spectrum most attractive for mobile cellular and Internet services has often been assigned to other existing services, spectrum management is a critical policy and regulatory issue. The needs for mobile services must be balanced against the needs of other services within a country, and against the international need to coordinate spectrum use and create standards for equipment.

Mobile communication channels can be defined as a connection between two nodes or multiple nodes. There are different types of communication channels used by mobile devices nowadays, such as Wi-Fi, GSM, Bluetooth, NFC, Radio-frequency identification (RFID) and others. Such communications channels are capable of providing different types of information and can be categorized by data or by voice and text only.

The first wireless Internet access became available in 1991 as part of the second generation (2G) of mobile phone technology. Higher speeds became available in 2001 and 2006 as part of the third (3G) and fourth (4G) generations. In 2011, 90% of the world's population lived in areas with 2G coverage, while 45% lived in areas with 2G and 3G coverage, [ITU, 2011] and 5% lived in areas with 4G coverage. By 2017 more than 90% of the world's population is expected to have 2G coverage, 85% is expected to have 3G coverage, and 50% will have 4G coverage [Ericsson, 2012]. Some mobile services allow more than one device to be connected to the Internet using a single cellular connection using a process called tethering [Ergen, 2009]. Tethering is connecting one device to another. In the context of mobile phones and tablet computers, tethering allows sharing the Internet connection of the phone or tablet with other devices such as laptops. Connection of the phone or tablet with other devices can be done over wireless LAN (Wi-Fi), over Bluetooth or by physical connection using a cable, for example through USB. If tethering is done over Wi-Fi, the feature may be branded as a mobile hotspot. Mobile hotspot is a feature present in smartphones nowadays which lets users convert their

smartphone into a portable router. A barrier to mobile wireless Internet use is the coverage provided by the mobile phone networks. This may mean no mobile phone service or that service is limited to older and slower mobile broadband technologies. Customers will not always be able to achieve the speeds advertised due to mobile data coverage limitations including distance to the cell tower. In addition, there are issues with connectivity, network capacity, application quality, and mobile network operators' overall inexperience with data traffic.

Wireless communication systems and services are now extensively deployed throughout the world. The most widespread cellular telephony standard is the Global System for Mobile Communication (GSM). Likewise, the most widely deployed wireless local network (WLAN) standard for Internet access is IEEE 802.11 commonly known as Wi-Fi. The Internet is a collection of disparate computers and networks coupled together by a web of interconnections using standardized communications protocols. Its vast reach as a result of its wide and increasing availability and easy access protocols characterizes the Internet. Unfortunately, the heterogeneous nature of the Internet makes it difficult for the hardware and software that implement it to add functionality. The Open System Interconnection (OSI) network model usefully describes networked data communication, such as the Internet, as a series of logical layers or protocol layers. The OSI model was basically developed to simplify network complexity, facilitate network training and introduce easy network troubleshooting. The OSI model is not real network architecture, because it does not really specify the services and protocols each layer should use. It rather describes what the layers must do. Each layer provides services to the layer above it, and shields the layer above it from details of lower layers. Each layer is configured to communicate with other similar level layers. In general, computers at network nodes (e.g., clients and servers) implement higher-level processes including application layer, presentation layer, and session layer processes. Lower level processes, including network layer, data link layer and physical layer operate to place data in a form suitable for communication across a raw communication channel or physical link. Between the higher and lower level processes is a transport layer that typically executes on a machine at the network node, but is highly dependent on the lower level processes [Bora, et al., 2014].

Mobile Technology by Generation

In the beginning mobile service providers had limited opportunities to offer interactive data services, however the appearance of Wireless Application Protocol (WAP) allowed the interactivity to support Internet and Web applications. WAP is a technical standard for

accessing information over a mobile wireless network through a WAP browser, which describes a protocol suite allowing the interoperability of WAP equipment, and software with different network technologies, such as GSM and code Division Multiple Access (CDMA). The bottom-most protocol in the suite, the Wireless Datagram Protocol (WDP), functions as an adaptation layer that makes every data network look a bit like UDP to the upper layers by providing unreliable transport of data with two 16-bit port numbers (origin and destination). On native IP bearers such as GPRS, Universal Mobile Telecommunications Systems (UMTS) packet-radio service, or Point-to-Point Protocol (PPP) on top of a circuit-switched data connection, WDP is in fact exactly UDP. WDP provides transaction support (reliable request/response) adapted to the wireless world and supports more effectively than TCP the problem of packet loss, which occurs commonly in 2G wireless technologies in most radio conditions, but is misinterpreted by TCP as network congestion. Finally, this protocol (WDP) suite allows a terminal to transmit requests that have a Hypertext Transfer Protocol (HTTP) or Hypertext Transfer Protocol Secure (HTTPS) equivalent to a WAP gateway; the gateway translates requests into plain HTTP.

The last few years have witnessed a phenomenal growth in the wireless industry, both in terms of mobile technology and its subscribers. There has been a clear shift from fixed to mobile cellular telephony, especially since the turn of the century. With all the technological advances, and the simultaneous existence of the 2G, 2.5G, 3G and 4G networks, the impact of services on network efficiency have become even more critical. Figure 3 shows the evolution of mobile technology by generation, 1980 onwards.

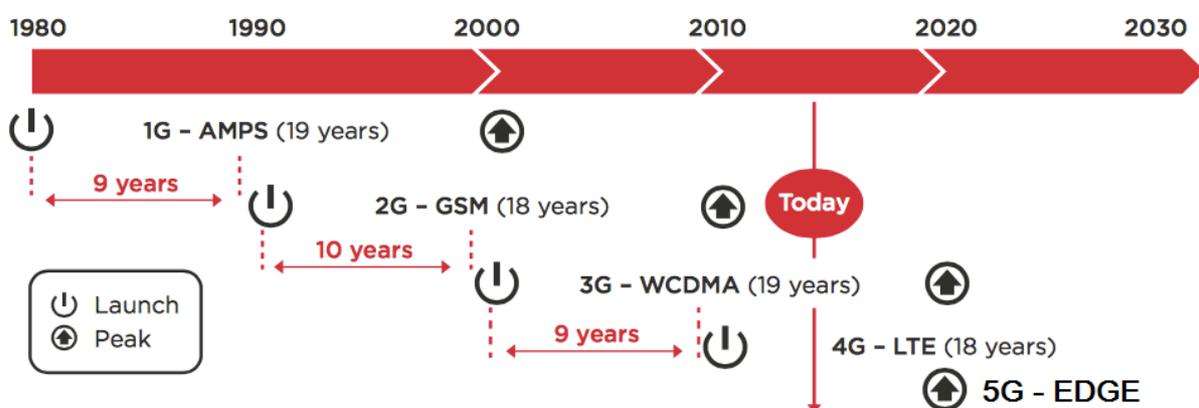


Figure 3: Evolution of mobile technology by generation, 1980 onwards adapted from [Intelligence G.S.M.A., 2014] AND 5G

1G

Commercial wireless systems and services have undergone rapid development and deployment, since 1G cellular telephone systems that were introduced in the early 1980s [Stüber, 2011]. 1G mobile systems used analogue transmission for speech services. This system offered handover and roaming capabilities but the cellular networks were unable to interoperate between countries. This was one of the inevitable disadvantages of first-generation mobile networks. [Kumar, et al., 2010].

GSM core network

GSM core network is the component of a GSM system that carries out call switching and mobility management functions for mobile phones roaming on the network of base stations. It is owned and deployed by mobile phone operators and allows mobile devices to communicate with each other and telephones in the wider public switched telephone network. Its features consists of the circuit-switched core network, used for traditional GSM services such as voice calls, SMS, circuit switched data calls, WAP, MMS and Internet access.

2G: GSM

2G systems were developed and introduced in the early 1990s and still in widespread operation today. The 2G cellular systems are based on either Time Division Multiple Access (TDMA) or CDMA technologies, and were initially designed to carry circuit-switched voice and data [Stüber, 2011]. 2G systems use the GSM, which is a circuit-switched network ideal for delivery of voice but with limitation for sending data. However, in 2000 the introduction of General Packet Radio Service (GPRS) added packet-switched functionality to start the delivery of Internet on mobile devices. Therefore the 2G systems later became known as 2.5G systems since they were enhanced to provide packet-switched data in addition to circuit-switched voice.

2,5G: GPRS

The 2.5G networks break both digital voice and data into small parts, and cross both onto the network simultaneously in a process called packet switching, allowing the mobile phone to have a voice connection and a data connection at the same time. Such process concedes web browsing on the mobile phone, however a personal computer (PC) can also tap into this service if connected through a mobile phone. GPRS is a packet oriented mobile data service, which extends the GSM Packet circuit switched data capabilities and permits a best-effort service,

implying variable throughput and latency that depend on the number of other users sharing the service concurrently, as opposed to circuit switching, where a certain quality of service (QoS) is guaranteed during the connection [Ainscough, 2001]. The GPRS accomplish the following services: SMS messaging and broadcasting, "Always on" internet access, MMS, Push to talk Over Cellular (PoC), Instant messaging and presence, Internet applications for mobile devices through WAP, Point-to-point (P2P) and Point-to-Multipoint (P2M). Furthermore, if SMS is used through GPRS, it is possible the transmission of 30 SMS messages per minute, instead of using GSM that only achieves 6 to 10 SMS messages per minute [Ainscough, 2001]. Moreover, if TCP/IP is used, the mobile phone can have one or more IP addresses allocated, where the GPRS will store and forward the IP packets to the mobile phone even during handover. The TCP handles any packet loss (e.g. due to a radio noise induced pause).

2,75G: EDGE

EDGE is a superset to GPRS and can function on any network with GPRS deployed on it. EDGE does not require hardware or software changes to be made in the GSM core network, however EDGE-compatible transceiver units must be installed and the base station subsystem needs to be upgraded. If the operator already has this in place, which is often the case today, the network can be upgraded to EDGE by activating an optional software feature. Today all major chip vendors support EDGE for GSM.

3G – UMTS

UMTS is a third generation mobile cellular system for networks based on the GSM standard. Developed and maintained by the 3GPP (3rd Generation Partnership Project), UMTS is a component of the International Telecommunications Union in IMT-2000 standard set and compares with the CDMA2000 standard set for networks based on the competing CDMA technology. UMTS uses wideband code division multiple access (W-CDMA) radio access technology to offer greater spectral efficiency and bandwidth to mobile network operators.

The third generation (3G) cellular system were introduced after the year 2000 and allowed simultaneous use of speech and data services with higher data rates. These data rate capabilities supplemented by geolocation information gave rise to location-dependent services. The umbrella term for the third generation radio technologies developed within 3GPP is the UMTS. However, UMTS is not just about radio: the radio access network connects to the core network

that is an evolution from the GSM core. 3GPP has expanded its capabilities, in principle allowing most services to be delivered over either 2G, GSM/EDGE or 3G.

3,5G

A further evolution is the 3.5G technologies High-Speed Downlink Packet Access (HSDPA), which provides speeds of multiple Megabits per second. Several of the mobile network operators that provide 3G or faster wireless Internet access offer plans and wireless modems that enable computers to connect to and access the Internet. These wireless modems are typically in the form of a small USB based device or a small, portable mobile hotspot that acts as a WiFi access point (hotspot) to enable multiple devices to connect to the Internet. WiMAX based services that provide high speed wireless internet access are available in some countries and also rely on wireless modems that connect to the provider's wireless network.

4G - LTE

In 4G systems, the circuit-switched infrastructure is abandoned and only a packet-switched network is provided, while 2.5G and 3G systems require both packet-switched and circuit-switched network nodes. Currently, 4G cellular systems have been introduced and use voice over Internet (VoIP) and multimedia applications with ultra-broadband (gigabit peak speed) access.

LTE, an abbreviation for Long-Term Evolution, commonly marketed as 4G LTE, is a standard for wireless communication of high-speed data for mobile phones and data terminals. It is based on the GSM/EDGE and UMTS/HSPA network technologies, increasing the capacity and speed using a different radio interface together with core network improvements [Cox, 2012]. The different LTE frequencies and bands used in different countries will mean that only multi-band phones will be able to use LTE in all countries where it is supported. The goal of LTE was to increase the capacity and speed of wireless data networks using new digital signal processing (DSP) techniques and modulations that were developed around the turn of the millennium. A further goal was the redesign and simplification of the network architecture to an IP-based system with significantly reduced transfer latency compared to the 3G architecture. The LTE wireless interface is incompatible with 2G and 3G networks, so that it must be operated on a separate radio spectrum.

5G

Mobile data traffic in the years to come foresee an extreme increase. This unprecedented growth demands a significant increase of wireless network capacity. Even if the current evolution of 4G systems and, in particular, the advancements of the LTE standardization process foresees a significant capacity improvement with respect to 3G systems, the European Telecommunications Standards Institute (ETSI) has established a roadmap toward the fifth-generation (5G) system, with the aim of deploying a commercial system by the year 2020 [Barbarossa, et al., 2014].

Bluetooth

Bluetooth is a wireless technology standard for exchanging data over short distances from fixed and mobile devices, and building personal area networks (PANs). It can connect several devices, overcoming problems of synchronization. Bluetooth networks often operate as a single connection, or a Bluetooth network may involve many devices. Bluetooth also allows for a scheme known as Bluetooth pairing where devices can quickly be associated. The Bluetooth specification defines a variety of forms of Bluetooth network connection that may be set up. In this way Bluetooth networking is a particularly flexible form of wireless system for use in a variety of short-range applications.

Table 2 presents the evolution of mobile generation technology in terms of services and performance.

Generation	Services	Network	Key differentiator	Weakness
1G	Analogue phone calls	PSTN	Mobility	Poor spectral efficiency, major security issues
2G	Digital phone calls and messaging	PSTN	Secure, mass adoption	Limited data rates – difficult to support demand for internet/e-mail
2,5G	Phone Calls, messaging and data	GSM TDMA	MMS Internet	Security issues
2,75G	Phone Calls, messaging and data	WCDMA	Better security and data rate	Low data rates
3G	Phone Calls, messaging and data	Packet Network	Better Internet experience	Real performance failed to match hype, failure of WAP for internet access
3,5G	Phone Calls, messaging and broadband data	GSM TDMA	Broadband internet, applications	Tied to legacy, mobile specific architecture and protocols
4G	All-IP services (including voice, messaging)	Internet	Faster broadband internet, lower latency	Security and handovers
5G	Dynamic Information access, wearable devices with AI capabilities	Internet	Edge-Computing	Under study

Table 2: Evolution of technology generations in terms of services and performance (source: Adapted from Intelligence G.S.M.A. 2014)

2.5. Evolution of Networked Devices

An emerging topic of technical, social, and economic significance is the IoT. Everyday objects, consumer products, cars, industrial and utility components, and sensors are being combined with Internet connectivity and data analytic capabilities changing our daily communities [Rose, et al., 2015]. Projections for the impact of IoT on the Internet and economy are impressive, with some anticipating as many as 100 billion connected IoT devices and a global economic impact of more than \$11 trillion by 2025 [Rose, et al., 2015]. At the same time the IoT raises significant

challenges about the hacking of Internet-connected devices, surveillance concerns, and privacy. Therefore technical challenges remain and new policy, legal and development challenges are emerging.

IoT take advantage of advancements in computing power, electronics miniaturization, and network interconnections to offer new capabilities not previously possible. The impact of IoT is already being discussed from new market and business opportunities to concerns about security, privacy and technical interoperability. On the other side, IoT possibilities in society, industry, and commerce is already widely recognized [Michael and James, 2014] [Marco and Karin, 2014].

The recent focus of IoT researchers are mainly on communication issues and on enabling interoperability [Luigi, et al. 2010]. Furthermore, a great effort has been also been made in devoting approaches to facilitate resources integration and services towards a distributed view for the IoT. For instance, the evolution of IoT to the Web of Things (WoT) vision [Dominique, et al., 2011], [Heuer, et al., 2015], which promotes the provision of resources in an IoT network with Web Services, and therefore making it possible to develop distributed and coordinated IoT services by using standard Web technologies.

2.5.1. Internet of Things

The IoT is a system of physical objects that can be discovered, monitored, controlled or interacted with by electronic devices which communicate over various networking interfaces, and eventually can be connected to the wider Internet. There are different definition to IoT, where the [Spanoudakis and Moraitis, 2015] links the IoT back to cloud services:

“The Internet of Things (IoT) is a framework in which all things have a representation and a presence in the Internet. More specifically, the Internet of Things aims at offering new applications and services bridging the physical and virtual worlds, in which Machine-to-Machine (M2M) communications represents the baseline communication that enables the interactions between Things and applications in the cloud.”

The various definitions of IoT emphasize different aspects from different points and use cases, but all of them focus on network connectivity and computing capability extension of objects, devices, sensors, and everyday items. Such extension allows devices to generate, exchange, and

consume data with minimal human intervention. Some applicability's examples of IoT can be found in Table 3.

Setting	Description	Example
Human	Devices attached or inside the human body	Devices (wearables and ingestibles) to monitor and maintain human health and wellness; disease management, increased fitness, higher productivity
Home	Buildings where people live	Home controllers and security systems
Retail Environments	Spaces where consumers engage in commerce	Stores, banks, restaurants, arenas – anywhere consumers consider and buy; self-checkout, in-store offers, inventory optimization
Offices	Spaces where knowledge workers work	Energy management and security in office buildings; improved productivity, including for mobile employees
Factories	Standardized production environments	Places with repetitive work routines, including hospitals and farms; operating efficiencies, optimizing equipment use and inventory
Worksites	Custom production environments	Mining, oil and gas, construction; operating efficiencies, predictive maintenance, health and safety
Vehicles	Systems inside moving vehicles	Vehicles including cars, trucks, ships, aircraft, and trains; condition-based maintenance, usage-based design, pre-sales analytics
Cities	Urban environments	Public spaces and infrastructure in urban settings; adaptive traffic control, smart meters, environmental monitoring, resource management
Outside	Between urban environments (and outside other settings)	Outside uses include railroad tracks, autonomous vehicles (outside urban locations), and flight navigation; real-time routing, connected navigation, shipment tracking

Table 3: Application settings of IoT technology's application [Manyika, et al., 2015]

The projection about the potential impact of IoT projects more than 24 billion Internet-connected objects by 2019 [Rose, et al., 2015]. More optimistic, Morgan Stanley, projects 75 billion networked devices by 2020. Looking further to the future, Huawei forecasts 100 billion IoT connections by 2025, with a financial impact on the global economy may be as much as \$3.9 to \$11.1 trillion [Manyika, et al., 2015]. Collectively, these predictions point to an extreme significant growth and influence of IoT. However, for the IoT to become a reality, we will need

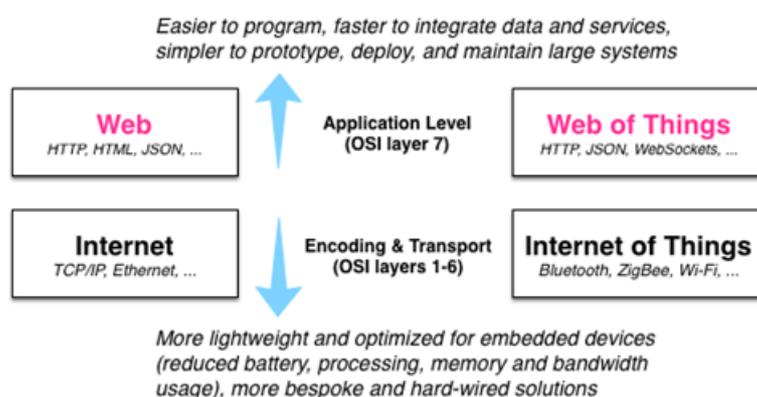
a single universal application layer protocol for devices and applications to talk to each other, regardless of how they are physically connected.

2.5.2. Web of Things

WoT approaches are very likely representing a keystone technologies in the future of IoT, where a number of different approaches are being proposed to support engineering of IoT systems and applications, such as middleware [Lina, et al., 2015] [Mrissa, et al., 2015] and programming approaches [Jacob, et al., 2015] [Latronic, et al., 2015]. However, it's still missing a common and unifying approach supporting their design and development (abstractions, models and methodologies).

Therefore, instead of creating another protocol from scratch like many IoT projects have been doing, the WoT reuse and leverage readily available and widely popular Web protocols and standards to make data and services offered by objects more accessible to a larger community. This approach effectively breaks the ongoing “one device, one protocol, one app” pattern used by the IoT. The future to push down to each tiny device the exact same technology that helped, for example Facebook or Google to scale to millions of concurrent users, without compromising on security or performance.

The WoT only deals with the highest OSI Layer (7), see Figure 4, which handles applications, services and data. Working on such a high level of abstraction makes it possible to connect data and services from many devices regardless of the actual transport protocols used. In contrast, the Internet of Things does not advocate a particular application level protocol and is usually focusing on the lower layers of the OSI stack.



2.6. Conclusion

Networking interruption and intermittency cause several issues, especially lack of connectivity, excessive native mobile resource consumption, and frequent application interruption and failure which lowers quality of computing services and degrades user experience [Satyanarayanan, 2005] [Abolfazli, et al., 2014]. Also, it originates redundant costs on cloud-connected mobile users and encumbers reliability. There are solutions, which allow the fast switching between network communications, such as data or voice and text. However, the development and the way that mobile applications are seen must change, since traditional client-server mobile applications are seen as a client and a server, and not as whole, which provides different user experiences. Finally, the development mobile applications should support the characteristics presented in Table 4, by combining the available hardware with the existing connectivity options.

OS	Connectivity Wi-Fi, Bluetooth, RFID, NFC, GSM, 3G, 4G	Hardware Processor, Memory, Disk Space, Antenna, SD Card	APPs Web, Native, Hybrid, OCC and RMA
iOS	Yes	Yes (but No SD Card)	Yes
Android	Yes	Yes	Yes
Windows Phone	Yes	Yes	Yes

Table 4: Mobile Devices Characteristics

While the IoT has been busy resolving networking problems, the WoT relies exclusively on application level protocols and tools. Mapping any device into a Web mindset making WoT agnostic to the physical and transport layer protocols used by devices. Finally, we can consider that mobile may be losing track against IoT, especially if the WoT approach is followed. WoT uses already available protocols to communicate, which mixing the Edge paradigm with the WoT, a world communication could be possible, making all devices and objects interconnected.

Chapter 3. Adaptation of Mobile Applications: a Literature Review

This chapter introduce the evolution of mobile devices novel paradigm and the related work. First of all, in section 3.1, the context and evolution of mobile architectures is provided. Then, in section 3.2, the applications communication fusion are addressed (comparison of mobile computing and pervasive computing, cloud computing and mobile cloud computing, and mobile edge computing and fog computing), followed by their characteristics and main features. Finally, a short summary in section 3.5 concludes this chapter.

3.1. Introduction

In recent years, massive developments of applications have been targeted for mobile devices in various categories, such as entertainment, health, games, business, social networking, travel and news. The reason for such abundance is related to mobile computing since it provides to mobile users the right tools, when and where it is needed – irrespectively of their movement – supporting therefore location independence. Furthermore, in the current technological ecosystem we are faced with several studies related to mobile application development. These might include a wide range of architectures, models and frameworks that are used to develop, manage or to give the better user experience, and can go from mobile computing to edge computing.

3.2. Methodology for bibliographic review

The literature review used to address the research of mobile devices, services, communication and features follows a process of manual search of scientific articles and books. To get to the main references, the articles, books, and technical reports of the sample were selected because they were also cited in other reviewed articles. The presented research addressed follows the process presented in Table 5. The research was performed through Google Scholar, ACM Digital Library and Web of Science search engines using terms such as “edge computing”, “applications”, “always-connected”, "mobility," "availability," "cloud computing," "middleware", "services", among others and also by adding the term "mobile" to the search. Thus, a list of broad-spectrum references was collected, and their titles and abstract were read, sometimes even in full, in cases where the nature of the research was not clear. In the case of Google Scholar references, the number of citation was also a selection criterion to organize the results. The references of each publication were analyzed and allowed to recognize other studies related to the mobile communications field that were not found in the repository of the search engines research. The literature review consisted of a complete reading of the references. In this process, it was also taken into consideration the background research list by the authors cited that were related to the subject of mobile communications.

Review Questions	<ul style="list-style-type: none"> • What are the existing communication models approaches? • What are the architectures for developing distributed mobile applications? • How have the emerging mobile technologies evolved in terms of mobile communication fusion?
Research Databases	<ul style="list-style-type: none"> • Google Scholar • ACM Digital Library • Web of Science
Analysis of titles and abstracts	<ul style="list-style-type: none"> • 149 selected publications • 4 excluded publications since it was not possible to obtain more information about them
Acquisition and reading of publications:	<ul style="list-style-type: none"> • 67 selected publications
Total number of surveys included in the review	<ul style="list-style-type: none"> • 10 selected publications

Table 5: Methodology process for bibliographic review

3.3. Architectures for mobile applications development

One major concern in mobile connectivity is how to efficiently access cloud-based applications when using a resource-constraint mobile device, where essentially applications require a continuous Internet connection that is difficult to obtain in challenged environments (lack of communication, interference access networks and even areas with high costs of Internet roaming) [Pal, 2016]. Some researchers have started to explore and analyse existing solutions in order to understand the relevance, innovation and future prospects of this field. For example, [Pal, 2016] explore the emergence of extending cloud-based applications with mobile opportunistic networks in challenged environments and observe how local user's social interactions and collaborations help to improve the overall message delivery performance in the network. On the other hand, [Tsai, et al., 2010] gives an overview survey of current cloud computing architectures, discusses issues that current cloud computing implementations have and proposes a Service-Oriented Cloud Computing Architecture so that clouds can interoperate with each other.

Furthermore, focusing in mobile cloud computing related work it has been identified three communication protocols used to communicate, namely: Wi-Fi, Mobile Data (3G and 4 G) and Bluetooth [Fernando, et al., 2013]. In the Wi-Fi mobile clouds it were identified several researchers, such as: Spectra [Flinn, et al., 2002], Chroma [Balan, et al., 2003], Cuckoo [Kemp,

et al., 2010], Cloudlets [Satyanarayanan, 2009], MAUI [Cuervo, et al., 2010], CloneCloud [Chun, et al., 2011], MobiCloud [Huang, D., et al., 2010], Hyrax [Marinelli, 2009], Virtual cloud [Huerta-Canepa and Lee, 2010] and Scavenger [Kristensen, 2010]. In Mobile Data, there is less research under this perspective and the ones that can be found also use Wi-Fi technology, therefore are not exclusive to Mobile Data communication, namely: Cuckoo [Kemp, et al., 2010], MAUI [Cuervo, et al., 2010] and CloneCloud [Chun, et al., 2011]. Finally, using the Bluetooth protocol to communicate, there is Cuckoo [Kemp, et al., 2010] and MMPI [Doolan, et al., 2008]. However these solutions are focusing on the development of mobile cloud, the experimental results presented are mainly for energy consumption analysis [Cuervo, et al., 2010] [Fernando, et al., 2013].

There are more research focusing on addressing other gaps of mobile cloud computing, however their main focus is the computation offloading of the mobile devices, namely for dependency relation to optimize the energy and execution time [Singh and Madan, 2016]. Mobility offloading decision can be altered due to unstable connectivity of mobile networks therefore a novel and robust solution is proposed by [Deng, et al., 2015] to solve the problem of mobile computation-offloading due to workflows mobile services that occurred to fulfil their complex requirements; [Rego, et al., 2016] presents a framework to support a method-based offloading technique for applications of different mobile platforms; to reduce the energy consumption and computational cost, [Guo, et al., 2016] propose a data offloading and task allocation scheme for a cloudlet-assisted ad hoc mobile cloud in which the master device who has computational tasks can access resources from nearby slave devices or the cloudlet; [Singh and Madan, 2016] proposed a system based upon the user's moving path mobility, which will assume the user's region to finish the process and reduce the response time as well as improve the load balancing; [Reiter, et al., 2016] shows a solution that enables a dynamic use of external resources and assures that security-critical computations are offloaded to trusted resources only; [Kaur and Makkar, 2016] implemented a distributed application processing to leverage the limitations of resources on mobile devices by outsourcing the application processing load to cloud server nodes entirely or partially; finally is proposed by [Liu, et al., 2016] an iterative decoupling algorithm with high efficiency to obtain near-optimal offloading decisions for energy saving.

Finally, when focusing on mobile devices and the intent they deliver services, a variety of communication procedures can be found, such as a "*Device to Cloud*" (D2C), "*Cloud to Device*"

(C2D) and “*Device to Device*” (D2D) [Pal, 2016]. In D2C architecture approach, mobile devices connect to the cloud through an Internet connection and the data storage and processing happens outside the mobile devices; In C2D approach applications run inside the mobile devices and are executed at the edge of the mobile network; the D2D approach is done through a direct communication between nearby devices that aim to improve information availability and energy efficiency [Pal, 2016] [Alam, et al., 2015]. Researchers have been developing different architecture approaches that follow these classifications models:

a. Device to Cloud

[Li, et al., 2009] propose mobile cloud architecture to deploy mobile computations to the remote cloud servers, by executing the data processing outside the mobile devices and ensuring a secure communication between the user’s mobile devices and the cloud-based applications through a trust-based mechanism. However, a major concern is the available network bandwidth for seamless data offloading between the mobile agents and the cloud servers.

‘Volare’ is an architecture presented in [Papakos, et al., 2010] that monitors the resources and context of the mobile device and dynamically adapts cloud-based services accordingly with the user’s preferences at runtime. It provides a better service provision to the users with a cost benefit approach and efficient bandwidth utilisation during high network traffic. However, an environment to gain benefit from these advantages in the absence of an Internet connection is missing. [Samimi et al., 2006] brings other architecture to extend the infrastructure clouds service to the places with wireless Internet connections. It aims to provide efficient services at places that are away from the traditional wired infrastructures through an automatic communication between the users within a network. Unlike the ‘Volare’ [Papakos, et al., 2010], this architecture explores the service provision during the time of disconnection, where the router redirects the communication paths between the users, regardless of their network connectivity for direct communication, device or application.

[Shen, et al., 2010] introduces an architecture named ‘E-Recall’ for personal multimedia information management, searching and sharing for mobile devices, based on the coordination between the mobile devices and cloud-based applications. Unlike ‘Volare’ [Papakos, et al., 2010] it requires a higher bandwidth of network for service delivery, but it works seamlessly in dynamic heterogeneous networks. However, ‘E-Recall’ requires a constant Internet connection for communication with one another. [Ou, et al., 2007] presents an architecture that supports

dynamic offloading mechanisms in a wireless mobile communication network, by exploring offloading mechanisms in failure circumstances. Unlike, [Samimi, et al., 2006] architecture, the major limitation is based on wireless mobile communication networks which may suffer from connection problems in challenged environments and any disconnections during the offloading execution treated a task as a failure.

[Klein, et al., 2010] presents an architecture based on the concept of Intelligent Radio Network Access (IRNA) to provide an intelligent network access strategy to the mobile users. It deals with the dynamics and heterogeneity of the available networks based on the user's application requirements. The advantage of this architecture is for heterogeneous network environments since it uses a Uniform Resource Identifier (URI) to give to the user a faster interaction with the service provider.

b. Cloud to Device

[Huang, D., et al., 2010] presents 'MobiCloud', which provides cloud-based applications to the local mobile nodes through a cloud-based mobile augmentation approach. This architecture supports a particular data offloading using the maximum advantage of each mobile node in the system that uses cloud-based applications. However, it is influenced by bandwidth and latency. [Miluzzo, et al., 2012], explores 'mCloud' where mobile devices become a core component for executing cloud-based applications. It splits the cloud-based computations by dividing a task into smaller subtasks to other mobile devices according to the execution's requirements. However, this architecture does not mention how to collaborate seamlessly for information exchange in a challenged environment.

[Shi, et al., 2012] presents a cloud-based application that runs inside the mobile devices according to the execution's requirements and user's application requirements. It allows making offloading decisions in variable connectivity and enables an efficient resource management mechanism. However, it requires a constant Internet connection for a communication establishment between the mobile devices and cloud servers. [Verbelen, et al., 2012] presents a 'cloudlets' approach, where cloud-based applications have moved nearer to the user's mobile devices in the form of 'small data storage'. Cloudlets are used for mobile devices due to its widely dispersed and decentralised Internet infrastructure and the ability to provide a high bandwidth of network utilisation [Pal, S. (2016)]. The proposed architecture reduces the bottleneck of wireless communications and improves latency, since it introduces a

‘pocket cloudlet’ where full or a part of the cloud-based applications is stored inside the mobile device.

c. Device to Device

[Mtibaa, et al., 2013] presents a scalable and autonomous Mobile Device Centric (MDC) approach based on the user’s social network relations. The social network relations help to avoid unwanted communications by identifying foreign users during the content offloading between the devices. [Pedersen and Fitzek, 2012] presents an architecture that explores technical and social aspects of communication. In this architecture, mobile devices create their own networks for sharing information between each other, and explore a user’s mobility during information exchange. However the offloading data to the cloud servers is difficult with the absence of an Internet connection.

[Jin and Kwok, 2010] explore architecture for collaborative information sharing between users who have similar interests. It is based primarily on the combined information received from multiple users located in a close physical proximity, where each user handles a part of the information. However, how to enhance this architecture in places without a network infrastructure is lacking in this research. [Huerta-Canepa and Lee, 2010] presents an architecture that helps user to dynamically locate an alternative route to communicate with the cloud servers during disconnection of a network service. It explores the alternative route by locating users in a physical proximity with a stable connection to offer. However, this architecture does not consider user’s mobility, device’s processing capacity and user’s privacy related issues when communicating with the nearby users.

3.4. Emerging Mobile Technologies: the applications-communications fusion

The growing trend of mobile computing is the pervasive computing (also called ubiquitous computing). Pervasive computing means, “existing everywhere” because the devices are completely connected and constantly available. Researcher goals of pervasive computing field is to create smart products that communicate unobtrusively through the Internet and where the data they generate is easily available. Furthermore, one of the main characteristics of pervasive computing environments is mobility where users are able to work seamlessly regardless of their movement [Fernando, et al., 2013]. However, mobility brings problems associated, namely:

low connectivity, energy consumption and resource scarceness [Satyanarayanan, 1996] [Fernando, et al., 2013].

In recent years, these problems have been addressed through Cloud Computing where applications are delivered as services over the Internet through data centers [Armbrust, et al., 2009] [Fernando, et al., 2013]. Cloud computing brings benefits associated with it, however, the average network latency and jitter increases because it implies a large separation between end users and the cloud in the centralization of resources [Satyanarayanan, 1996] [Roman and Mambo, 2016]. Therefore, various novel paradigms have emerged where the deployment of cloud computing capabilities are at the edge of the network, such as mobile cloud computing, mobile edge computing, and fog computing [Beck and Maier, 2014] [Vaquero and Roderomero, 2014] [Wang and Wang, 2015] [Manco, et al., 2015] [Garcia, et al., 2015] [Roman and Mambo, 2016]. These edge paradigms have various differences, such as different protocols and interfaces, personal cloudlets (user-owned edge data centers), and infrastructure providers that focus on the mobile network operators, among others [Roman and Mambo, 2016]. The evolutions of these novel paradigms are represented in figure 5 and are described in detail in the following sections.

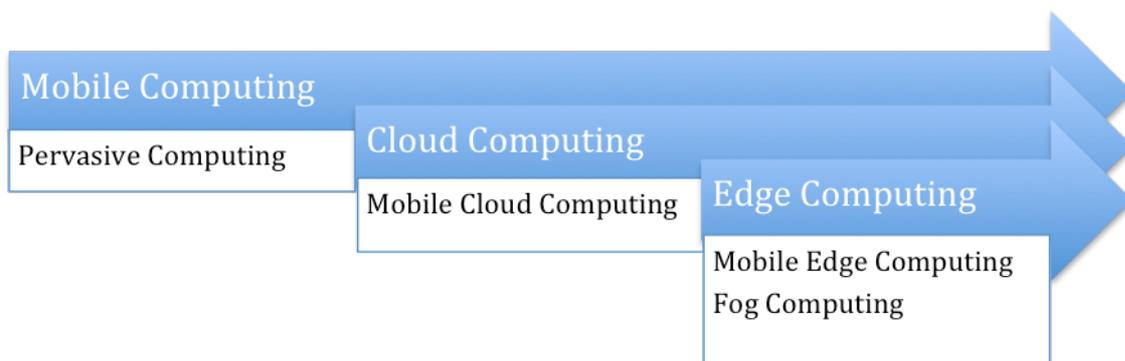


Figure 5: Evolution of mobile devices novel paradigm

3.4.1. Mobile Computing and Pervasive Computing

The most effective and convenient communication tools that are increasingly becoming an essential part of human life, and are not bounded by time and place, are the mobile devices

[Dinh, et al., 2013]. Hereby, a powerful trend in the development of IT technology, commerce and industry is led by the rapid progress of mobile computing [Dinh, et al., 2013]. Mobile computing is identified by variations in user needs, computing and communication resources [Alia, et al., 2007]. There are four constraints that characterize mobile computing [Satyanarayanan, 1996], namely mobile elements are resource-poor relative to static elements, mobility is inherent hazardous, mobile connectivity is highly variable in performance and reliability, and mobile elements rely on a finite energy source. The new area of mobile computing is increasingly mobile devices with more capable processing speed and store, faster wireless network and lower latency [Bahl, et. al., 2012]. However, there are many challenges in terms of bandwidth, limited battery capacity, constraints of wireless networks, mobility and the device limitation, which significantly drawback the improvement of mobile services [Satyanarayanan, 1996] [Dinh, et al. 2013]. There are work being done in mobile computing that explore its capabilities and act in various scientific fields. For example, some researchers, take benefit of mobile computing by focusing on providing educational opportunities for students for access course content, with interaction with instructors and colleagues wherever they are located [Cavus and Ibrahim, 2008] [Kukulaska-Hulme and Shield, 2008] [Richardson and Lenarcic, 2008] [Shih, 2007]. On the other hand there are researchers focusing on the technical aspects, such as [Alia, 2013] that presents a distributed resource management framework and mechanism to maintain an up to date resource model at runtime. This model leverage of runtime needs, resources, adaptation capabilities of the software, performs context monitoring, adaptation planning, and dynamic reconfiguration.

The growing trend of mobile computing and the increase of mobile devices and wireless Internet represent the first truly mobile pervasive computing device [Campell and Choudhury, 2012]. In pervasive environments devices are completely connected and constantly available, in order to communicate unobtrusively through the Internet and where the data they generate is easily available. On the other hand, one of the main characteristics of pervasive computing environments is mobility where users are able to work seamlessly regardless of their movement [Fernando et al., 2013]. However, there are inherent problems associated with mobility, such as low resources, finite energy, low connectivity and network bandwidth [Satyanarayanan, 1996] [Hung, et al., 2012]. According to Tim O'Reilly the future belongs to mobile devices through services that respond in real time to information provided by users, devices and sensors [Siegele, 2008] [Fernando, et al., 2013]. Real time applications demand intensive computing

resources because of the high levels of responsiveness. In order to facilitate the development of distributed mobile applications in pervasive environments Hydra [Satoh, 2005] presents a network set of pervasive computers where the application follows the location change, current task and number of people. Following the user's movement allows it to support the user mobility by moving the mobile agent component through the corresponding servers that will contact the mobile application service.

A barrier that needs to be overcome in order to realize and full potential of mobile computing and pervasive computing is in fact intrinsic to mobility and not merely temporary technological deficiency [Fernando, et al., 2013]. In the last years, mobility problems have been addressed through the cloud computing, where the applications are delivered as services over the Internet and the hardware that process it is located in data centers [Vogels, 2008] [Armbrust, et al., 2009].

3.4.2. Cloud Computing and Mobile Cloud Computing

Cloud Computing, also called demand computing, utility computing or pay as you go computing, is a collection of computing resources shared to serve multiple consumers, which are available over the network and accessed through standard mechanisms [Fernando, et al., 2013], where its main objective is to offload the computation to remote resource providers. Therefore, Cloud Computing can be also defined as the aggregation of computing as a utility and Software as a Service [Vogels, 2008] where the applications are delivered as services by data centers over the Internet [Armbrust, et al., 2009]. Cloud computing is an emerging paradigm for cost efficient and reliable service provisioning [Guest Editorial, 2015] with a diversity of deployment models and service models, provided by the Cloud Computing paradigm, which are in fact the key strengths of Cloud Computing and can be described in terms of the services offered by cloud service providers, such as: from public to private clouds, Infrastructure as a Service [Carolan, et al., 2009], Platform as a Service and Software as a Service [Roman and Mambo, 2016].

Cloud computing is an emerging paradigm for cost efficient and reliable service provisioning, where the rich availability of energy-harvesting and resource-constrained mobile computing devices is beginning to converge with the great opportunity offered by the powerful cloud computing services hosted by virtualized data center resources. Mobile Cloud Computing has been introduced has a potential technology for mobile services alongside with the explosive

growth of mobile applications and Cloud Computing [Dinh, et al., 2013]. Mobile Cloud Computing is based under the concept of Cloud Computing, where adaptability, scalability, availability and self-awareness are requirements that need to be met in a Cloud [Mei, et al., 2008] [Fernando, et al., 2013] [Kaur and Makkar, 2016]. Mobile Cloud Computing refers to an infrastructure where both the data storage and the data processing occur outside of the mobile device. Mobile cloud applications move the computing power and data storage away from mobile devices and into the cloud, bringing applications and mobile computing not only to smartphone users but also to a much broader range of mobile subscribers. Hence, the obstacles presented in mobile computing are overcome by integrating the Mobile Cloud Computing concept. Mobile Cloud Computing focuses on the notion of mobile delegation due to the limited resources available to mobile devices [Roman and Mambo, 2016] where both the data storage and the data processing occur outside of the mobile device.

3.4.3. Mobile Edge Computing and Fog Computing

Current mobile architectures are fixed in nature and delivered well applications. However, the 3GPP solutions are based around a static topology, which at the same time enables a degree of mobility within the network. Mobile Edge Computing enables resources to be placed closer to the end-user, taking into account the relative locations of producers and consumers. It is an assumption to place generic compute and storage close to the network edge in a mobile network environment, which extends the cloud to the local environment. There are benefits of deploying cloud services at the edge of mobile networks like 5G, such as low latency, high bandwidth, and access to radio network information and location awareness. The edge environment is embedded in and managed by the operator through a set of APIs where the application operators and developers can explore their capabilities.

Fog computing allows the management and programming of compute, networking and storage services between data centers and the end device. It is defined as a distributed computing paradigm that extends the services provided by the cloud to the edge of the network. Therefore, Fog computing addresses applications requirements that need low latency with a wide geographical distribution, namely by supporting mobility, computing resources, communication protocols, interface heterogeneity, cloud integration and distributed data analytics. So, Fog Computing enhances the service quality to mobile users, when compared to

cloud computing, since it increases data rate and reduced service latency and response time, by downloading data through local connections [Luan, et al., 2015].

There are work being done within the Fog Computing paradigm, for example: [Beck, et. al., 2015] presents an approach that reduce approximately 13% of the power consumption of mobile devices while offloading video to external services co-located with cellular base stations. On the other hand, [Takahashi, et al., 2015] discusses the performance metrics through a study of an abstract model of application execution in multi-tier mobile cloud and examines the trade-offs among them.

3.5. Conclusion

The current and continuing evolution in the novel paradigm architecture of mobile devices are based in a variety of mobile features (Table 6), such as: deployment, hardware, service, network architecture, latency, mobility, and availability. However these architecture provide a high degree of availability, there are some constrains that still need to be addressed, namely in limited conditions where the availability of services cannot be reached [Kaur and Makkar, 2016].

	Cloud Computing	Mobile Cloud Computing	Mobile Edge Computing	FOG Computing
Deployment	Network Core	Network Edge, Devices	Network Edge	Near-Edge, Edge
Hardware	Servers	Servers and Users Devices	Heterogeneous Servers	
Service	Virtualization	Virtualization, Others	Virtualization	
Architecture	Centralized	N-Tier, Decentralized, Distributed		
Mobility	No	Yes		
Latency	Average	Low		
Availability	High			

Table 6: Comparison of features provided by the novel paradigms

Current state of the art and related work through a variety of architecture fails in supporting mobile users while they move across networks or where no Internet connection is available. Therefore and to conclude, there are solutions following the presented architecture (from cloud

computing to fog computing, and so on) but their focus is mainly in performance, battery consumption or even network bandwidth. So neither of them focuses on the mobile devices perspective (available components) and how could the availability of services be solved in limited conditions (for example without internet connection) [Huerta-Canepa and Lee, 2010] [Zhang, et al., 2010] [Kaur and Makkar, 2016]. In this sense, the current state of the art fails in providing solutions that successfully achieve the multi-objective design goals; namely, (1) in providing always connected communication features, (2) the application reliability depending on the environment, and (3) in supporting mobile services applications with information access from the viewpoint of client-server computing.

Chapter 4. Software as a Service for Mobile (SaaS4Mobile) Architecture

This chapter introduces a general description and implementation details of the architecture that supports the dynamic model, designed to implement the model and allow its validation. First of all, in section 4.1, it is presented the support for the current research. In section 4.2 real world application scenarios are introduced and presented. Then, in section 4.3, it is explained the architecture and the contribution of this thesis. In section 4.4, the various phases of the architecture life cycle are addressed and fully explained. Finally, a short summary in section 4.5 concludes this chapter.

4.1. Introduction

The SaaS4Mobile Architecture defines a unified access to services provided by third parties through a mobile interface, governed by iconographic flows with non-verbal interactions. The architecture operates in a distributed environment using a mobile device, which supports a client application and a mediation service, and run the communication mediation modules and integration with external systems entities. The generalization of the solution allows the creation of specific applications for each service to be provided in a common environment, ensuring the sharing of resources, the scale and scope of the solution, enhancing the viability of services. The architecture has added value in providing mobile applications services using rapid prototyping for multiplatform based on specification of the application flow. Transparency of communication associated with independence and dynamic allocation of the communication channels used, ensures access to services provided by external entities through anytime and anywhere mobile applications.

The generalization of the concepts of Software as a Service for Mobile (SaaS4Mobile) Architecture, were the core basis for the provisional patent request, started in July 2012, entitled “Automated method of asynchronous, non-verbal communication mediation through interactive iconographic flows”. The work was funded by the “IAPMEI – Agência para a competitividade e Inovação” under the entrepreneurship program “Passaporte para o Empreendedorismo”. The technology and model developed under this work was licensed by UTAD to a company in October 2014.

4.2. Real world application scenarios

The SaaS4Mobile architecture is a generalization aiming to fit several real world scenarios. Previous work shows some example of application of the architecture, that were used the concept generalization and allow the design to the SaaS4Mobile Architecture. Three applications are presented and described, such as SOSPhone, MyCarMobile and My Mobile Medical Appointment (M3Medical). SOSPhone is an application to fast emergency call, MyCarMobile allows car drivers to call for assistance on the road and M3Medical allows users to manage appointments at health institution. In the following section these three mobile applications will be fully described and the main features explained.

4.2.1. SOSPhone

SOSPhone [Paredes, et al., 2014] is a mobile application that allows for emergency calls without the need of audio features. This mobile application offers the user an iconographic touch interface that allows users to contact the emergency center by choosing the icons that represent the problem that is being reported, and click just a few buttons to answer simple questions such as the number of victims in the event. The result of this selection process is a SMS message that is sent to a server containing the codes corresponding to the selected information together with the coordinates of the sender. In addition to the main features presented in the SOSPhone, the following are also included:

- Built-in emergency flow Protocol – to facilitate integration with existing systems the application allows the user to describe the situation in as much detail as possible, following the emergency protocols used by emergency control centers;
- Bandwidth reduced to communicate with the emergency control center – the use of the bandwidth must be reduced to allow faster communication of emergency;
- Automatic Location of incident (implicit or explicit);
- Identification of the caller;
- User registration is necessary – to help avoid false calls;
- Two-way communication channel between the user and the Public-safety answering point (PSAP) after the report of the emergency situation.

The evaluation of this prototype was designed by conducting interviews with deaf users and those responsible for emergency services in Portugal, namely ANPC, police / 112.pt and INEM.

The implemented solution seeks to exploit an interface based on icons to describe an emergency situation. To proceed to the collection of emergency data (occurrence), the application flow is managed by the protocol defined by the emergency services. The communication between the caller and the PSAP is performed using a SMS message, which is encoded throughout the description of the occurrence and includes location information provided by the mobile device. Figure 6 shows some application images.

The iconographic touch interface was implemented using icons of reference symbols, taking into account the best practices of mobile applications (<http://www.w3.org/TR/mwabp/>), and the necessary disability elements for the devices. The icons were grouped by severity as the various

emergency situations, so that the display is not overloaded with too much information, allowing better use by people with low mobility.

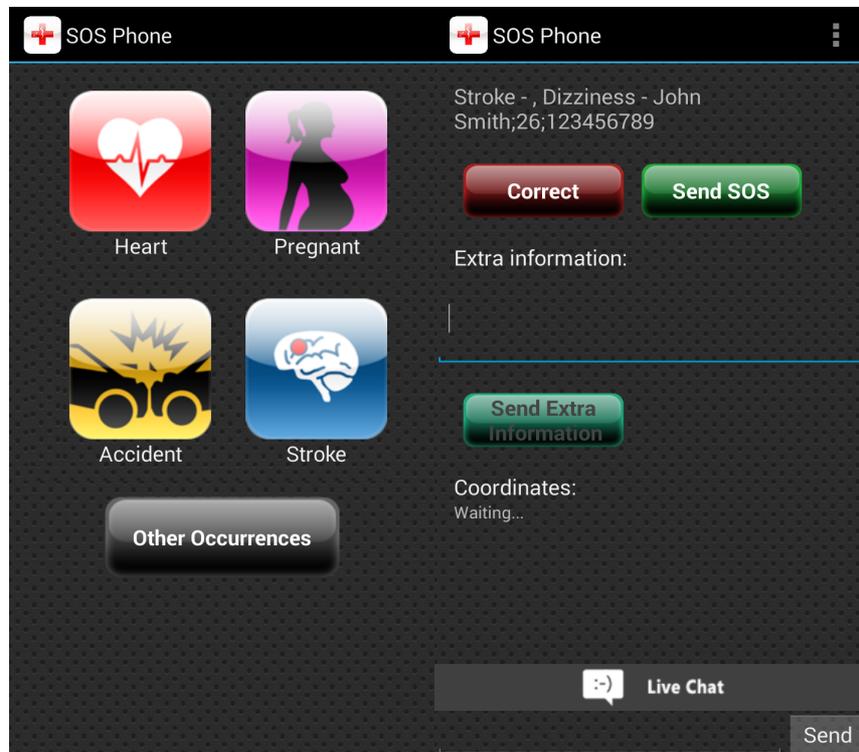


Figure 6: SOSPhone application interface

In order to ensure a low communication bandwidth to the emergency control center the prototype sends only one SMS message. The construction of the SMS message is made by the process, based on the letters and numbers shown in each of the options selected, taking into account the limitation of 160 characters. The user registration information is concatenated to this information, including name, age and social security number. Location information is also included in the SMS message if the GPS is available.

User registration is required and is executed when the mobile application is started for the first time. Normally, a phone is personal, so the recorded data will be collected only once. However, the log data can be changed at the last emergency call step before sending the SMS.

Regarding the installation of emergency flow protocol, the prototype supports the inclusion of mechanisms that allow the definition of dynamic flows without changing the application source code, allowing the adaptation of flows using dynamic updates in the mobile application through networks data (3G / Wi-Fi).

4.2.2. MyCarMobile

One of the communication problems that deaf people face in their day to day, especially those who have driving license, occurs when they get stuck on the road due to a malfunction in the car or even in case of an accident and need to communicate with the assistance traveling. A survey conducted with the Portuguese deaf community revealed that 80% have driving license and at least 56% required at least once to call the travel assistance [Paredes, et al., June 2014].

This survey also revealed that about 64% of service companies in travel have communication support for deaf, however, in 56% of occasions the deaf faced communication problems (36% did not answer, so this number may be even higher) , causing approximately half of the cases were taken.

The results presented indicate that current communication solutions between deaf and travel assistance are not efficient and are dependent on the help of others. In this context, the availability of a mobile application that uses the iconographic communication can help address this gap in the daily lives of the deaf.

This mobile application is titled MyCarMobile [Paredes, et al., June 2014] and uses interactive graphical information such as icons, diagrams and buttons to allow users to describe problems with the car and report it to a roadside assistance service. In this application are considered three categories of occurrence: the car crash, have an accident with the car and other problems. The mobile application interface example is presented in Figure 7.



Figure 7: MyCarMobile application interfaces

In Figure 7, the leftmost interface is the first screen, where the user can select one of 3 buttons corresponding to 3 basic situations: breakdown (“Avaria”), accident (“Acidente”) and other situations (“Outras Situações”). If the first option is selected, the interface changes to the second

one, which allows specifying the location of the breakdown: engine (“motor”), doors (“Portas”), inside view (“Vista interior”) and wheels (“Rodas”). The third interface shows the inside view, and the fourth is the one for the case of accident, enabling to specify the damaged areas. The fifth and sixth interfaces show refinements in the description of the problem, such as reporting broken glasses, requesting hauling or sending GPS coordinates.

The MyCarMobile follows the same characteristics and principles as SOSPhone. The mobile applications throw SMS allows to call for assistance in any kind of car trouble and also ask for assistance (everything in the same flow through iconographic touch).

4.2.3. M3Medical

In general, the deaf face various problems of communication when dealing with public services, however emerge even more critical difficulties when it comes to health and safety issues. The M3App is an application for mobile devices that allows users to effecting medical appointments, selecting iconographic icons, following the same interaction and architectural principles of SOSPhone. Again, despite their inspiration from the specific needs of the deaf it is also suitable for any other user, because of its convenience and the fact that is very intuitive. This mobile application avoids time-consuming phone calls and obviate the need for a person to have to move to a specific health center for an appointment.

The M3App has characteristics similar to SOSPhone and the MyCarMobile, here are some more specific features:

- Possibility to use location services to help the user find the closest health services;
- Ability to choose criteria to filter the desired location – use the current location, based on the address of the person and the preferences of health facilities, physicians, regions, cities, etc.;
- Mediation Service – The M3App is the user interface for a multitude of health services, which are managed by a central mediation service to guide the process of determining the member organizations (hospitals, clinics, health centers);

- Activation on first use – users must fill out a user profile that includes the most relevant identification data, such as name, ID number, address, health system or social security number, or even some more relevant clinical information (such as illness or permanent disability).

In this applications the user has access to the list of future appointments and the whole history of old appointment, which can be useful to help scheduling future appointments. When making a new appointment it is possible to choose user own profile to make an appointment for himself, or to someone else if necessary (for this it is needed to manually provide all the personal details). When the user finish filling out all the necessary information to make an appointment, this information is sent to the mediation service through the most convenient network communication channel. The mediation service processes this information and sends it to the health care unit proper, which later will have to confirm or negotiate alternatives to the request made. Communication between the mediation service and the health unit can be done through web, SMS or email. Figure 8 shows an image of the M3App prototype.



Figure 8: M3Medical application interface

4.3. SaaS4Mobile

Entities wishing to provide a service aimed at small groups in society face difficulties in terms of financial sustainability of the business, combined with the scale factor of existing solutions. The SaaS4Mobile aimed at solving this problem since it is a generic solution that can be used for a wide variety of services sharing a supporting infrastructure and maintenance. In this sense, from the perspective of the entity providing the service manager, the system aims to ensure the availability of the service in a short period of time and guarantee the minimization of costs associated with their development and maintenance, maximizing the financial viability of the service. Taking into account the requirements of universal access, most products are not developed because of its high cost of implementation and the small target market. The paradigm shift introduced by the conceptual model of automatic mediation of nonverbal asynchronous communication for interactive iconographic flows through the aggregation and rapid development of services, aims to minimize these problems.

From the point of view of users the goal is to ensure access to services that are limited to use due to temporary or permanent restrictions and special/limited needs. These restrictions are typically related to the availability of services when the communication channels are restricted to be used. The availability of an intuitive interface and easy learning that meets user needs, and mobile routines is another goal of the system. Therefore, the interface device should follow the interface standards in order to ensure consistent use with other applications. The transparency of the communication layer is another major objective and also a requirement of the mobile application, providing the most efficient communication channel at anytime, without user intervention to contact the mediation service and send the user request. Finally, the updates to the application flow are, according to the default options automatically performed without the need for user intervention and where the necessary communication conditions are met. These updates include the redefinition of the application flow that will be provided by the companies that provide the service.

4.3.1. Life Cycle

The model consists of three application components that interact with each other (Figure 9): a mobile application, a communication mediation service and services provided by third parties. Through the universal access interface of the mobile application, the access to the services provided by external entities is displayed according to a pre-defined flow. Communication with

external parties is mediated by the mediation service. The mediator gateway ensures standardization of information between external entities and mobile applications and ensures a common platform shared by several applications. This process involves the conversion of data formats, adapting the communication channels and forwarding of messages.

Each solution will have a life cycle with four phases (Figure 10 and 12):

- Planning and design;
- Installation and testing;
- Execution;
- Maintenance.



Figure 9: Application components interaction (mobile application, communication mediation service and third parties)

4.3.2. Planning and Design

The planning and design is the first stage of the SaaS4Mobile life cycle and involves the architect, the managers and users, similar to the project design stage. Analysing the service requirements to be implemented based on feedback from potential users and service managers carries out the first step. The architect uses the information gathered from the analysis of requirements and develops the application interaction flow, further he selects the mobile platforms where the application is executed and identifies the required services of external entities. Along with these actions the forwarding message in the mediation service is defined,

for example: the flow of messages received by the mobile device to external entities and vice versa. Furthermore, among the various communication channels supported by the system, the communication channels required by the services application are selected. A tool available to the architect supports the definition of message processing at the level of the mediation service and its referral to external entities. The application allows the graphical definition of the information flow, based on existing/registered entities in the system, and the definition of the routing process that can be manual or semi-automatic. In manual mode, the architect defines the whole process. In the semi-automatic mode, the system detects the characteristics of each of the external entity and suggests the architect the most appropriate streams based on the analysis made and the system knowledge. To ensure the development of a flow of information, the tool allows the definition of: states (interaction and action states); conditional structures (if, case); cyclic structures (while, repeat) and variable (types of simple and complex data structures). A set of actions to set states, in particular: send messages, receive messages, information on the system's location, continuously record the information and read persistent information are also available to the architect. Additionally, this tool provides a set of interaction patterns – icons, location setting, multiple choice, and choice of date – appropriate to the specific characteristics of the mobile interfaces in order to ensure that users have a quick learning and easy adaptation to use the application. The planning and design phase are complete with the creation of an application package containing the flow of interaction, routes, and the selection of mobile platforms and communication channels to be used by the application. This application package is sent to the administrator to start the installation and tests phase.

4.3.3. Installation and Tests

The installation and tests phase (Figure 10) are the system administrator's office. The tools provided for the administrator include monitoring management tools. Part of their duties is to ensure the management of the installation and maintenance of centralized infrastructure in the mediation service. At this level, the system and all peripheral services are controlled. Based on the application package supplied, the administrator connects the flow and the functional core of the mobile application through the creation of an application to the client that is subjected to the application markets of each selected mobile platforms. The operation of the functional core of the mobile application will be further detailed. Furthermore, at this stage, the package is compiled in the mediation service (Figure 11) based on routes and communication channels

defined by the application package created in the previous phase.

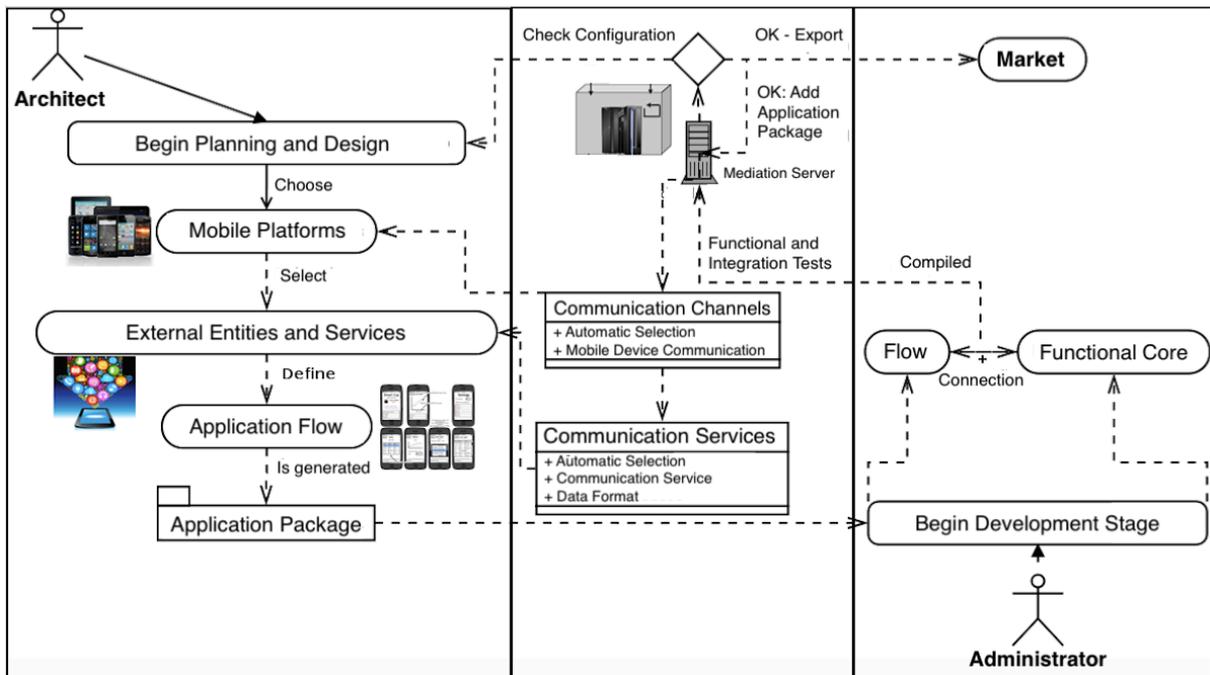


Figure 10: Planning, design, installation and tests phase's life cycle

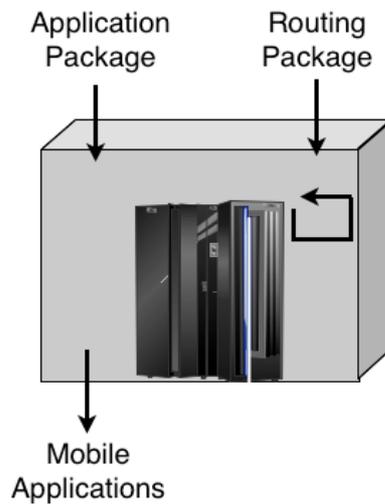


Figure 11: Mediation service components (mobile applications, application package and routing package)

This application package is inserted in the mediation service and are performed a set of functional and integration tests. After this phase and the availability of the applications in markets, the service is ready to use and the service execution phase starts. The administrator also manages the registration of external entities to which requests are sent. At this level, it is a requirement the administrator to define the most appropriate integration policies for every situation, making use of standards/models for integration with mobile applications, together with external entities. It should be noted that due to the generalization of the system could coexist several integration levels and different levels of services, which affect the response time to the user.

4.3.4. Execution

The execution phase (Figure 12) is characterized by the availability of the service to users and is started after the application is installed on mobile devices. At this stage, all system components work together: the mobile device, the mediation service and the external entities. An analysis of the operation of the mobile device and the mediation service is made in the execution phase, since the operation of the external entities is the service provider's liability. The use of system services begins with the user interaction with the application installed on the mobile device by following the flow of interaction defined by the service and the request is created for the entity. A functional core divided into four functional blocks, as shown in Figure 10, executes the mobile application. The structural simplicity of the execution core is intended to minimize the intermediates and thus guarantee minimum performance levels in its implementation taking into account the limitations of the equipment (especially on mobile devices with less features).

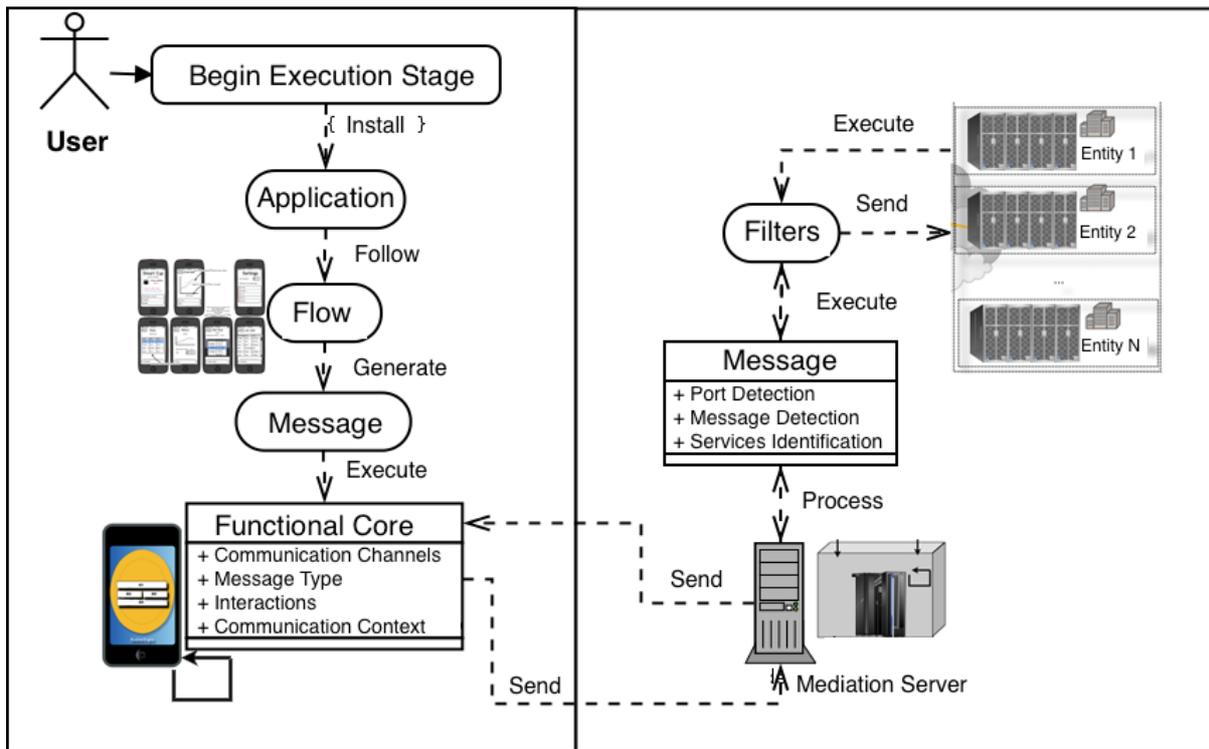


Figure 12: Execution phase life cycle

At the bottom of the execution core is made a data connection via an XML parsing function blocks. The XML functional blocks contain information exchange and therefore its availability to the system is of the responsibility of the block that interprets and adapts the data at run time. It should also be noted that the application flow protocols are defined in XML, according to a particular XML schema. The UILoader functional blocks and UIRender take care of the necessary adjustments so that data is used in real-time. The UILoader is responsible for support and sending events structures based on the protocol defined by the application flow. On the other hand, the UIRender performs the verification of user interfaces, validating the declaration of a single distributor of the event. The necessity of having a single shipping record for the event is related to the operation of the runtime environment, as will be shown later. This functional block also carries the user interfaces based on files described. Finally, the functional block has a running component that performs sending events between the interface and the user, and is specified in the protocol application flow. The basic operation of this block is a GenericDispatcher that dispatches events triggered in the user interface for the states transitions defined by the application protocol flow. Actions can occur in the case of dynamic method invocation, which are secured by means of reflection mechanisms.

At the end of customer interaction is sent a request to the mediation service (Figure 13) using one of the available communication channels. The channel is selected in consideration of the connection (available channel) interactions and the request type. When detecting reception of the request, the mediation service processes the message to obtain the service identifier associated with the received request. Filters are applied, based on the service and communication channel that has received the message, to the content in order to reconstruct the message to an intermediate format. Then, the modules associated with the service are activated to make the processing and forwarding the message through the defined application package. The mediation service accompanies the transformation and the message submission phase to the destination port, in accordance with the service rules – defined in the development of the application during the installation of the service in the mediation service and according with the requirements of the characteristics of each service. The message is re-encoded with the format described in the integration policy set between the mediation service and the external entity. Once it is confirmed that the message was successfully sent is recorded in the system this action, and if necessary, feedback mechanisms are activated to receive the answer. In this case, to send the message response, the mechanism that is processed is reversed to the reception in order to send the response to the user.



Figure 13: Runtime Engine – Functional Core Components Interface

4.3.5. Maintenance

In parallel to the execution phase follows the maintenance phase and evolution of the solution. To ensure high availability of the solution, these steps can be performed in parallel, requiring the system to ensure version control that determines not only the type of application, but also its version, always ensuring the availability of the current version and previous application until their records have been updated for all clients. The maintenance of an application can be made to the client side at runtime level or at the application package, which is hosted and carried by the mediation service. The evolution of the system is ensured by the developer, to which the system is now considered a low level, under the auspices of each of the functional blocks. From this perspective, the system is seen by Application Programming Interfaces (API) of the various components in order to evolve the solution to the needs of the market, according to the new requirements of service providers. To achieve this goal, it is a requirement that the system has a set of API references to ensure their development, including:

- API reference for the installation of translation modules and integration with external entities;
- Reference API for creating the communication adapters;
- API for creating actions at the level of the mobile device interface;
- API for models and standards interactions.

The application updates that modify the interaction of the flow are performed within the client application. At the core level application updates are considered in the evolution of the solution. Therefore, the client application update consists of:

- Creating a new application flow;
- Update package compilation;
- Application package installation of the update on the update service at the mediation service;
- Update notification on installed applications;
- When the necessary communication conditions to perform the update are available, it makes the download and installs the application package update locally;
- Dynamic loading of the new application flow.

With regard to updates on the mediation service, this also includes changes in the shipping flows of the application. Moreover, the evolution of the mediation service includes the development and delivery of new features. Updating the logic of sending a service includes:

- Creation of a new logic of submission;
- Compilation of updating the shipping package;
- Compilation of the update of the application package for the mediation service based on the application package submitted updates;
- Installation of the application updates the mediation service;
- Test the upgraded installation.

The evolution of the solution comprises updating the execution core of the mobile application, providing new features associated with the application, and error correction. This process involves the programmer who develops new or fixes existing bug. It also involves the administrator that after the release of the new execution core regenerates all applications hosted on the new core-based platform and makes the updates available in commercialized market. This process ends when users manually perform the update of the mobile application.

Similarly, the development is carried out for the mediation service, leaving the programmer to implement new features or bug fixes, and the administrator to install new application modules and testing before going to production.

4.4. Conclusion

The SaaS4Mobile Architecture is a quick method for developing native mobile applications always-connected, through a platform of rapid prototyping by simply setting the application flow with integration with services always available through the mediation service and the dynamic communication channel. This solution focuses on providing a generic service that provides interactive solutions based on iconographic flows to non-verbal communication, such as an integrated system of mediation and communication between the different entities. Finally, this architecture was designed in order to support the presented model of this research. The big picture is that the model can be supported by the SaaS4Mobile and therefore allow developers to implement mobile applications based on this new solution, and construct always available mobile applications.

Chapter 5. A model for Dynamic Switching of Communication Channels on Mobile Applications

This chapter introduces a model for dynamic switching of mobile communication channels on the development of mobile applications, in order to guarantee transparency and at the same time improve the efficiency of services availabilities. First of all, in section 5.1, a brief introduction of the key concepts associated with the model is presented. Then, in section 5.2, a distributed architecture to implement the model is introduced. In section 5.3 the conceptual model is addressed and fully explained. Further, in section 5.4, the model logical workability is analysed, which explains the model interaction components. Finally, a short summary in section 5.5 concludes this chapter.

5.1. Introduction

Mobile Cloud Computing (MCC) embrace several issues, such as Low bandwidth, availability [Huerta-Canepa and Lee, 2010] [Zhang, et al., 2010] [Kaur and Makkar, 2016] and heterogeneity [Dinh, et al., 2013]. Service availability becomes a more important issue in MCC than the CC with wired networks. Mobile users may not be able to connect to the cloud to obtain a service due to traffic congestion, network failures, and no signal availability [Dinh, et al., 2013]. The research presented herein proposes a generic model for dynamic switching of communication channels to ensure the principles of transparency in distributed environments. This innovative and dynamic model for the management of mobile applications services, through a dynamic layer manager of communication channels of mobile devices, ensures services availability to mobile users regardless of the environment limitations. Such communications channels are capable of providing automatically an always-connected feature to the mobile application independently of the environment limitations. This model achieves the multi-objective design goals, in the perspective of client-server computing, by means of a set of pre-defined rules for each service, therefore it chooses the best available communication channel and at the same time it adapts the interface and the available content (based on the defined rules) on the mobile application.

5.2. Distributed Model Architecture

Currently, mobile applications follow the new paradigm of distributed mobile applications and must be seen as a whole, such as mobile applications, communication channels, servers, and external services. The proposed model follow the distributed model architecture presented in Figure 14. There exist distributed scenarios in which the need for dynamism, mobility, and adaptability, has to be addressed with highly dynamic approaches. These scenarios present different challenges and difficulties: efficient access to heterogeneous and distributed data sources, dynamic load balancing, unstable connections and communication failures, and so on. To tackle these challenges different approaches have appeared, such as mobile agent technologies and middleware solutions.

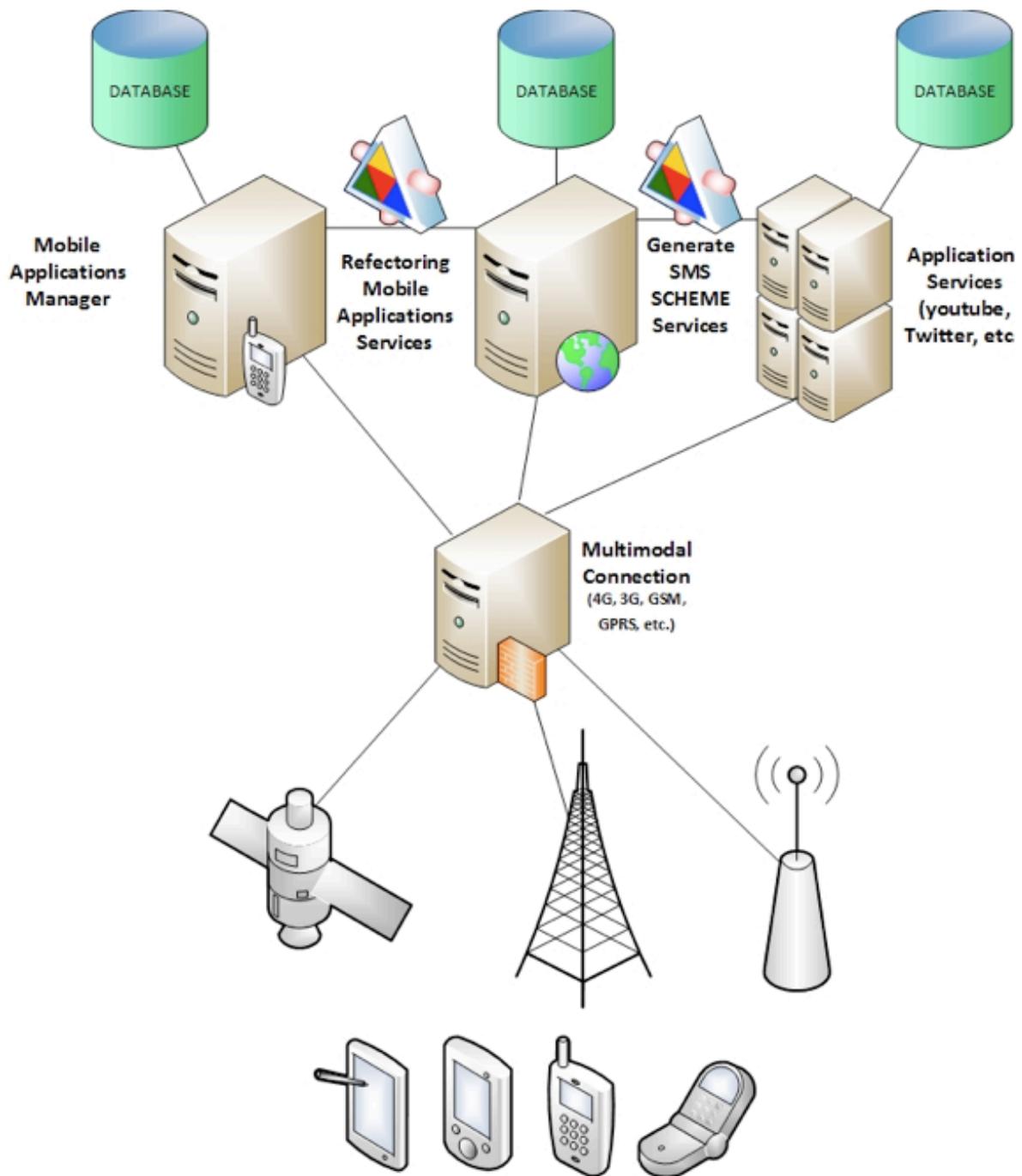


Figure 14: Distributed model architecture

In this scenario, the communication channels will process the connection between the mobile devices and centralized mediation servers, by means of a multimodal connection (4G, 3G, 2G, GSM, GPRS, DTMF, and so on). Services availability, capabilities and offered modalities are characterized by the current context, personal preferences, available devices and changes [Chu, et al., 2004], therefore enabling mobile multimodal applications needs to consider the users'

mobile situation and provide an “always-connected” application despite of the current limitations, such as the lack of Internet connection. For example, in case of having only channels available like Dual-Tone Multi-Frequency – DTMF or Global System for Mobile Communications – GSM, the application will request to the server the required service data information through a short message scheme. The server process the information in the web server through the Mobile Application Manager and the Application Services and then refactors the mobile application services information by generating and encapsulating a SMS SCHEME Services to be processed by the mobile application.

Following this approach, the proposal focus on the limitations of client-server connections in distributed mobile applications through a system that prevents communication loss. If a communication service required to the integrity of the mobile applications fails it will need dynamic communication channels capable of adapting to all environments, and capable to fragment and replace themselves while maintaining the integrity, functionality and transparency of the communication channels. Hence, such dynamic communications channels will be part of the whole life cycle of mobile applications: planning and creation, installation and testing, implementation, and maintenance and evolution.

5.3. Conceptual Model

The proposed model includes (see Figure 15), services available at the database and reached by the server. The communication layer that is responsible for all “Send and Request of Data” between the application, the services, and the server is transparent to the user, but is incorporated in all applications. These methods are available for the developer that wants to make use of a specific service on the development process. Each specific service has rules pre-configured in the server, however taking into consideration these rules the developer can add new rules and specific priorities for the services running developing application. All configurations are also recorded in the Communication Layer to work at runtime.

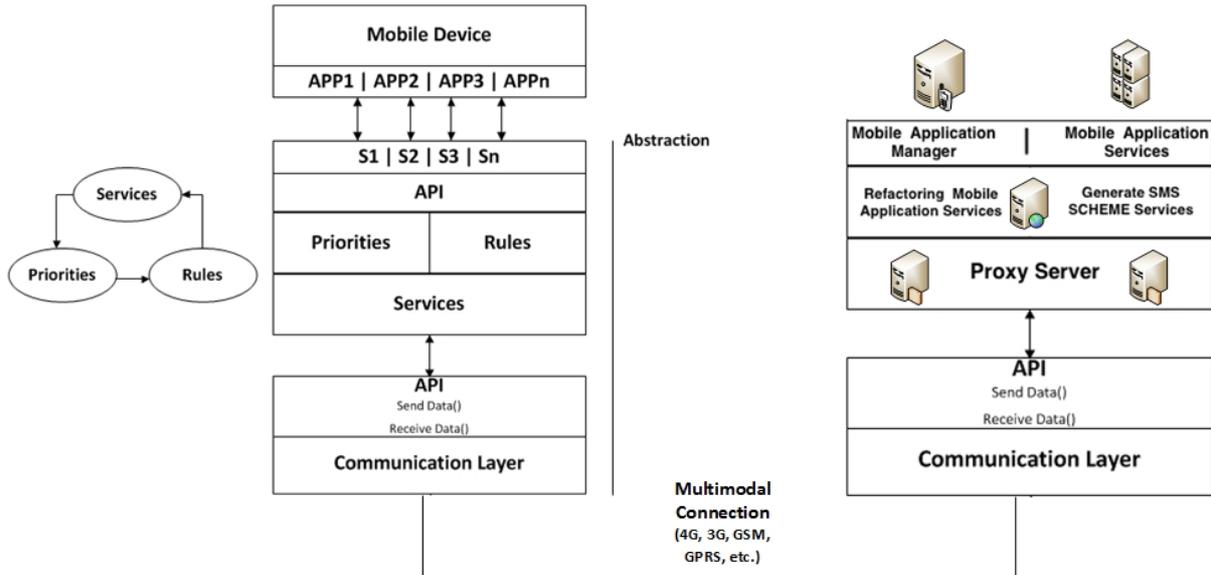


Figure 15: Model interaction within the distributed model architecture

However, to implement this model and to define how services will be available to mobile end users some requirements for the configuration of the communication Layer are needed, such as: (1) Network Type; (2) available Services and their associated Rules and Priorities (see Figure 9); (3) Connection verification (based on the priorities and rules of the specific service); (4) Services and communication Configuration (server, data access, communication type, other data and features); (5) an API (SEND and REQUEST DATA); and (6) Alert mechanism for end users.

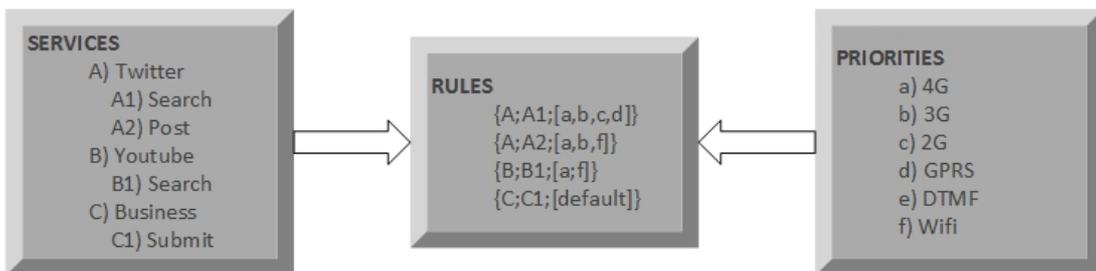


Figure 16: Model cycle representing how rules are related to service and communication channels

The Services, Rules and Priorities (Figure 16) are interconnected and are mandatory in order for the communication layer to communicate with the server, to request the proper data based on pre-configured features, and to know when and how it should adapt to the services running in the applications. In this scenario, Services, e.g. YouTube, Twitter, Business processes, and so on, are available at the mobile application and at the server manager; the developer in the server for each specific Service defines Priorities, which are related to how the availability of each service in the mobile application can be obtained depending on the environment

limitations, for example each service has referenced in the Priority scheme the hierarchy of the communication channels that can successfully sustain the availability of the specific service; Rules are defined for each specific Service in the server by the developer, which, based in specific pre-built conditions linked to the Priority scheme, represent the behaviour of the mobile application, how it should adapt to the environment limitations and the refactoring scheme of each service representation. In the developers perspective the model presents a complete abstraction method, since the model will be integrated into a development framework, being transparent to developers and mobile users.

5.4. Model Logical Workability

Figure 17 represents how the model will behave, which has two important points, the Mobile Application and the Server. The APP Run is the initialization process of the application, where the requirements and data are checked by the communication layer taking into account the services associated with the application and the pre-defined settings (Figure 16). The model has 3 major components in the point of view of the Mobile Application, such as the Application itself, the Services running in the application, and the communication layer that is responsible for all the communication transparency and send and request of data. In the Connection Logic, there are different approaches that are needed to be tackled, such as:

- The application has all its features available.
- All requirements are not available for its operation and therefore the application is not initialized, instead it sends a warning message to the user.
- Has just a few available requirements and is initialized by sending a warning message to the user.
- Taking into consideration that it only has some available requirements to work the application restructures itself and show only the available features, by hiding services, buttons, and so on.

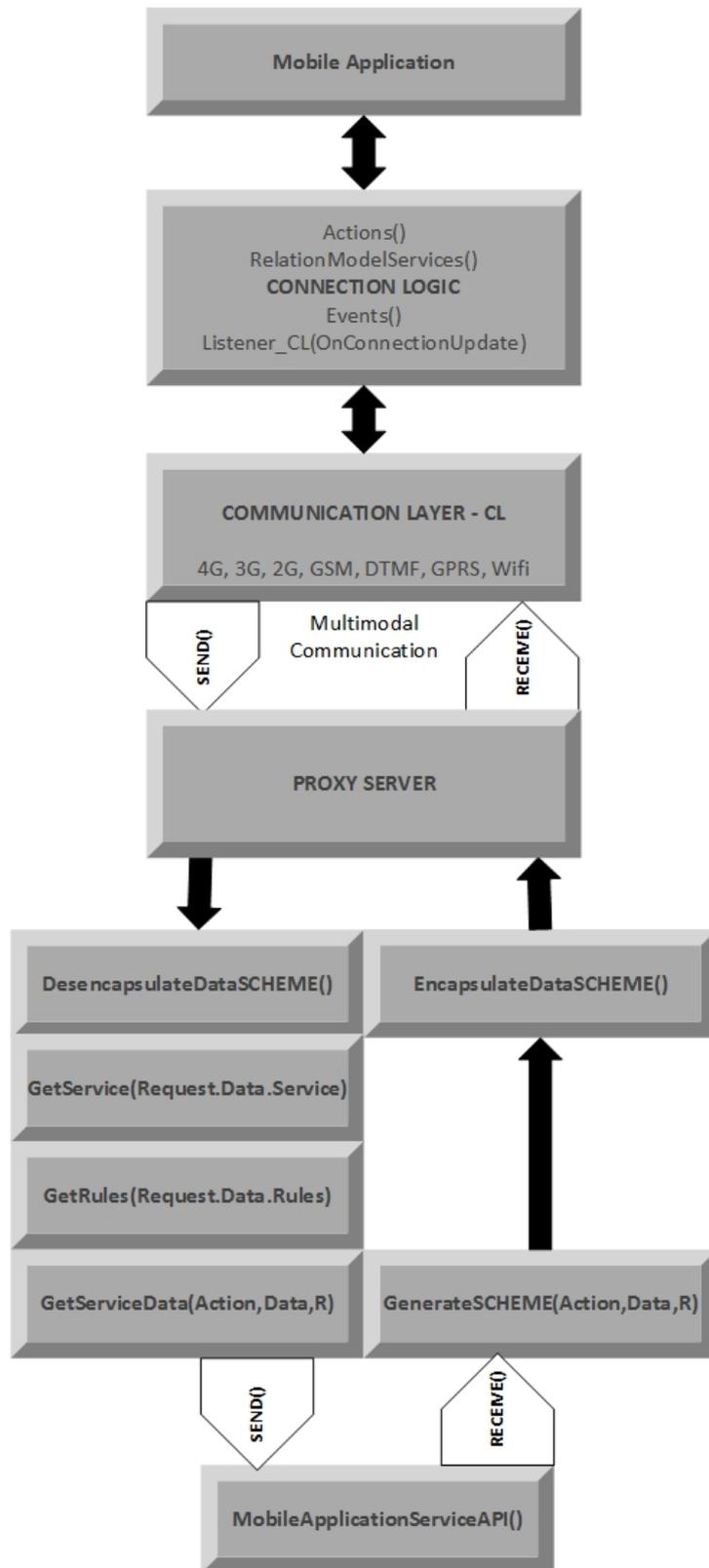


Figure 17: Model logical workability when a service is requested

The Listener_CL is the runtime process responsible to manage the communication channels changes, the RelationModelServices is responsible for checking the service settings and choosing the best configuration. After managing this configuration the communication layer sends the request through a multimodal channel to the server asking for the specific data within the specified configuration. The server gets the request and process the configuration (DesencapsulateDataSCHEME) based on the communication channel received, the service request (GetService), and attributes of the specific service (GetRules). After loading the configuration settings the server sends the request to the MobileApplicationServicesAPI. When the request is returned a data scheme is generated, encapsulated and sent through the requested communication channel to the mobile device. In the client side, the communication layer receives the data scheme and sends it to the connection logic to process the scheme configuration, which will make the service information available to the mobile application interface.

5.5. Conclusion

In this thesis, a model for dynamic switching of mobile communication channels is presented. A new perspective of distributed mobile applications is introduced based in the proposed model, which allows the refactoring at run time of mobile applications services depending on the environment limitations. Furthermore, the main objectives of this research are summarized, which were conducted through the establishment of partial objectives to be realized in various stages of the overall development. A partial list of targets to the primary objective that also represent the future work, are enumerated:

- An innovative and dynamic model for the management of mobile applications services is proposed, through a dynamic layer manager of communication channels of mobile devices, which ensures services availability to mobile users regardless of the environment limitations.
- Model specification of the dynamic layer communication channels management used by mobile devices in mobile applications, enabling the creation of rules and priorities to ensure always-available services.
- Implementation of the dynamic model to ensure the development of mobile applications, based on the model specification. The model will be implemented and assessed in an

application running services like Twitter, Youtube or other business process, in order to address the performance availability of these services while running the proposed model.

Chapter 6. A prototype for an always connected Mobile App: the Twitter Mobile App

This chapter validates the dynamic model and supported architecture through the implementation of case studies that were the basis for conducting experiments with mobile users. First of all, in section 6.1, two case studies are introduced. Then, in section 6.2, the first case study through a form of questionnaire is presented and the results analyses are discussed. In section 6.3, the second case study – development of an application and server based on the presented model – is addressed, followed by the unity tests results analysis. Finally, a short summary in section 6.4 concludes this chapter.

6.1. Introduction

Two approaches were conducted to support and prove the current research: further a study to support the need and motivation of the current research, and finally a case study to validate and implement the proposed solution. The first study was made through a questionnaire and supports the presented motivation described in the state of the art. The case study was made through a mobile applications implemented based on the proposed model.

In the first case study the purpose of the questionnaire is to conduct a study on the behaviour of mobile applications and the availability of services in ubiquitous environments. The study aims to analyse the main needs that mobile users face in the use of applications in different environments. The participants were volunteers of various age groups and the information collected is used only for the intended study, without the use of personal data.

In the second case study the goal of the mobile application is to validate and analyse the presented model in a real case scenario, where the Twitter service was implemented in the mobile applications following the guidelines presented in Chapter 5. In this scenario a server was also developed so that the mobile applications could get and post requests.

In the following sections both case studies are described.

6.2. Motivation Questionnaire

The questionnaire (Appendix 1 – Questionnaire) aims to collect information on the behaviour of mobile users in different environments. In this regard, participants should follow a set of questions drawn up with the objective of analysing the difficulties encountered in the use of applications in roaming. Filling out the questionnaire does not imply the prior use of any application, focusing on the experience of users. The completion of the questionnaire had an approximate duration of 10 minutes. Participants may contribute to a better efficiency of the involved study, primarily in the collection of problems that occur when they are moving abroad, which is directly associated and will allow the grounds of the need to ensure the availability of services even with limited mobile communications.

In the questionnaire were involved a total of 105 participants, between 14 and 58 years old, from 13 nationalities. The majority of the participants were graduated and master students, being respectively the former 50% and the latter 23,4% (Figure 18).

The major objective of the questionnaire focused in four important aspects:

- Importance of mobile devices;
- Use of mobile applications;
- Using mobile applications while travelling;
- Use of mobile applications even without Internet access.

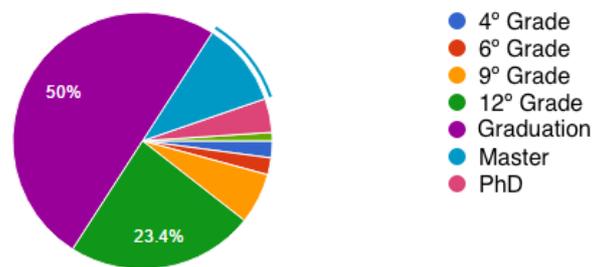


Figure 18: Participants degree

6.2.1. Results Analysis

Taking in consideration the main goals of the questionnaire the data gathered is presented and the results analysed. Various correlations were made with the data gathered, which allowed to filter and support important information.

One important motivation role of the presented investigation was the importance of mobility on mobile applications when users travel abroad. Therefore, it was made the correlation analysis of the importance of mobile devices for mobile user while they are travelling abroad (Figure 19). This correlation allowed understanding that the users that most consider mobile devices important are those who travel more often. The results show that participants travelling more than once per year aboard are those that consider that mobile devices are important (46,15%) and very important (53,85%).

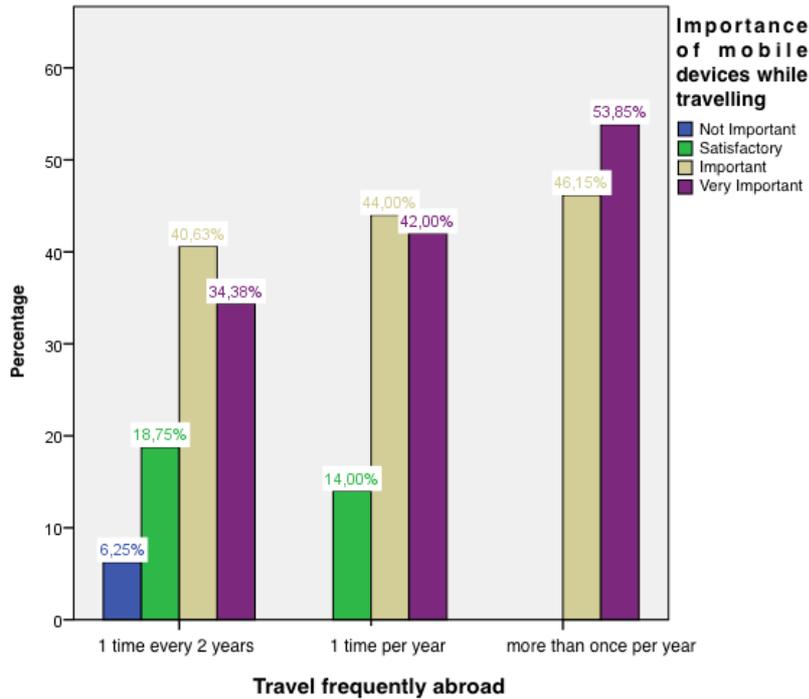


Figure 19: Relation between the importance of mobile devices while travelling and frequency of travelling abroad

Another research goal, as motivation, of the current research was the need for having always available services on mobile applications even without Internet connection. This was undoubtedly the great contribute of the research hereby described, since it brings huge benefits for mobile users (can access services on limited conditions), entities (provide always available services for their costumers) and developers (access to an API model to manage communication requirements). The result analysis of this important contribute for the mobile community (Figure 20), showed that the users that spent between 2 and 6 hours or even more time over the Internet while abroad are those who consider that is important to have access to mobile applications services even without Internet connection. It can be understood from this results that expert users – those that spent more time on their mobile device – are the ones that more access mobile applications services and therefore have the need to continuously interact with the mobile services.

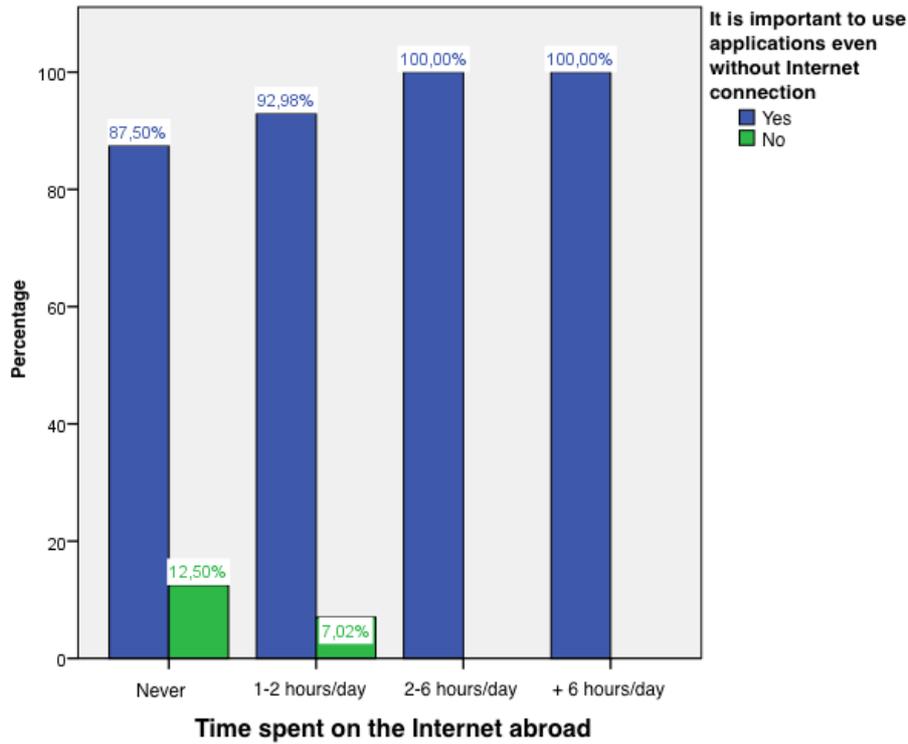


Figure 20: Relation between the importance of use applications even without Internet and the time spent on the Internet

Further, Figure 21 is presented to give more emphasis and support for mobile users that travel more often abroad are those who spent more time on the Internet. Therefore, it makes even more necessary to provide access to mobile services everywhere and at every time. Since, mobile users that travel more than one per year abroad are those who spent between 2 to 6 hours per day on the Internet abroad (53,85%).

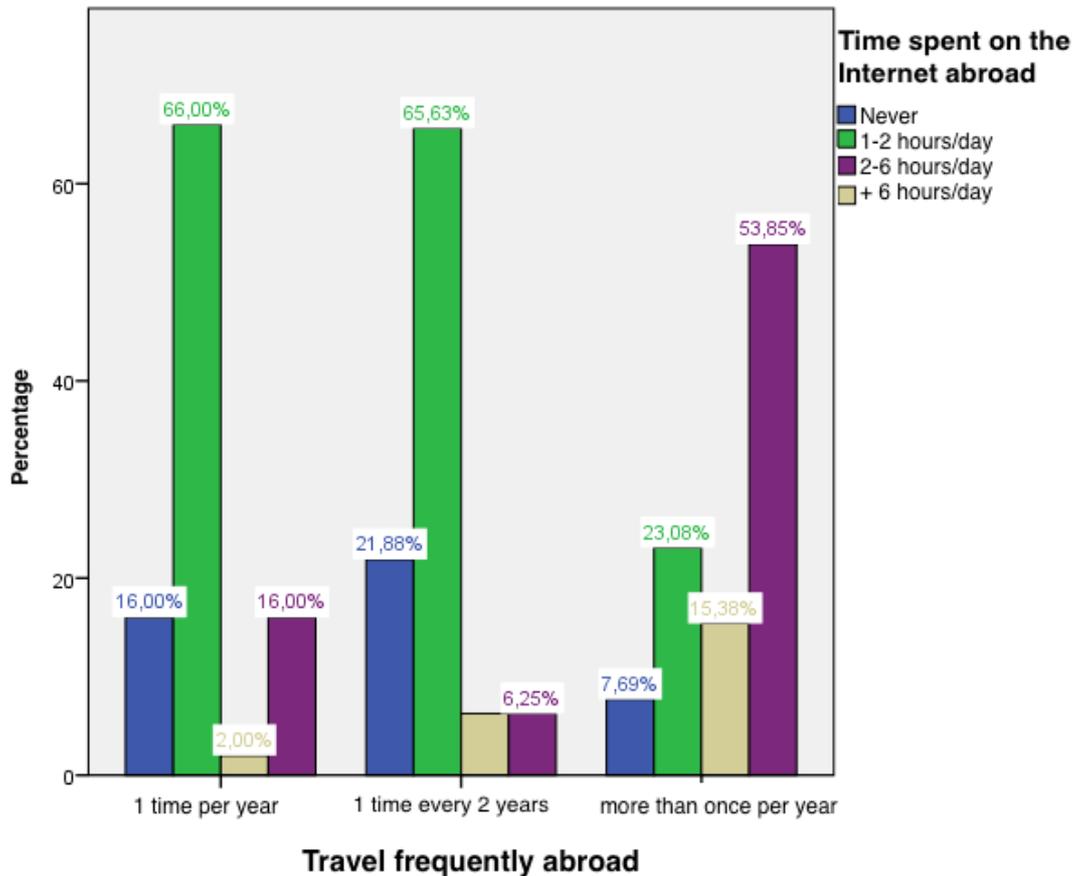


Figure 21: Relation between the time spent on the Internet and frequency of travelling abroad

Finally, to understand how mobile users gain access to mobile Internet while abroad when compared to the time that they spent on the Internet, Figure 22 was elaborated based on the questionnaire responses. The results show that from the various Internet access modes, namely: Hotspots, public places with Internet access, local operators and mobile data of the national carrier; the local operators and the national carrier are the preferred for all kind of users, where the users that least use Internet abroad have an average of 56% of using all different modes of accessing the Internet. On the other hand, the results showed that the more expert users – spent between 2 to 6 hours or more on the Internet – prefer to use local operators (average of 16,6%) or national carrier (average of 12,55). Therefore, from non-experts to expert’s mobile users their main goal is to have access to Internet, in order to use their mobile applications services. While the non-expert users just want to have access to the Internet, the expert users are more careful choosing their way to connect, which can also means that they want to access with security and privacy to their services information.

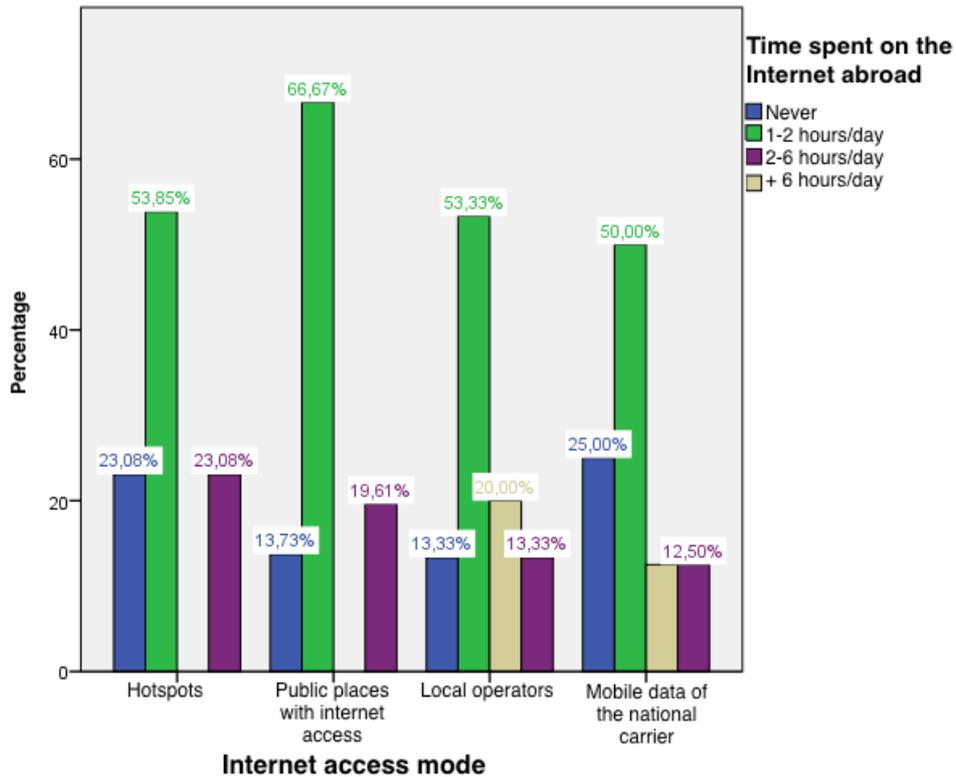


Figure 22: Relation between the time spent on the Internet and the way users access the Internet abroad

6.3. The Twitter Mobile App

In this research a model for dynamic switching of communication channels to guarantee the availability of mobile services is presented in Chapter 5. In Chapter 4, is addressed a distributed architecture to support the proposed model. Therefore, in this case study, a mobile application and a server have been implemented to assess and support the addressed model. In the client side a mobile applications supporting the Twitter service was implemented based on rules, services and priorities, as defined in Chapter 5. The mobile application contacts through different available communication channels the mediated server through requests, and the server side deals with the mobile applications requests and contact the associated services.

In the following sections the server, the mobile applications and the technical aspects are described.

6.3.1. Server Side

The server side is responsible to handle request from a mobile application, through three available channels, such as: Internet, Bluetooth and GSM. The server is running in a Windows

8 OS virtual machine and was developed with WindowsForm components from Visual Studio 2015. The form application server supports Universal Serial Bus (USB) communication to install and contact a HUAWEI Mobile Connect Model E180 (Figure 23), which is used as a GSM Modem. Furthermore, the server side has three major components, namely: FormConnect (Figure 24), GSMManager (Figure 25) and HandleRequests.



Figure 23: GSM Modem – HUAWEI Mobile Connect Model E180

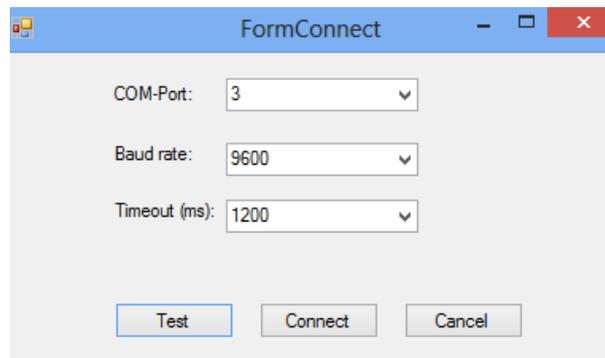


Figure 24: FormConnect – Server communication ports

FormConnect (Figure 24) is the first configuration done at the server side when is running up. It is at this point that the GSM Antenna is configured with the necessary communication requirements, namely: COM-port, Baud rate and Timeout. After setting up the GSM Antenna and successfully connects to the server is presented interface GSMManager (Figure 25).

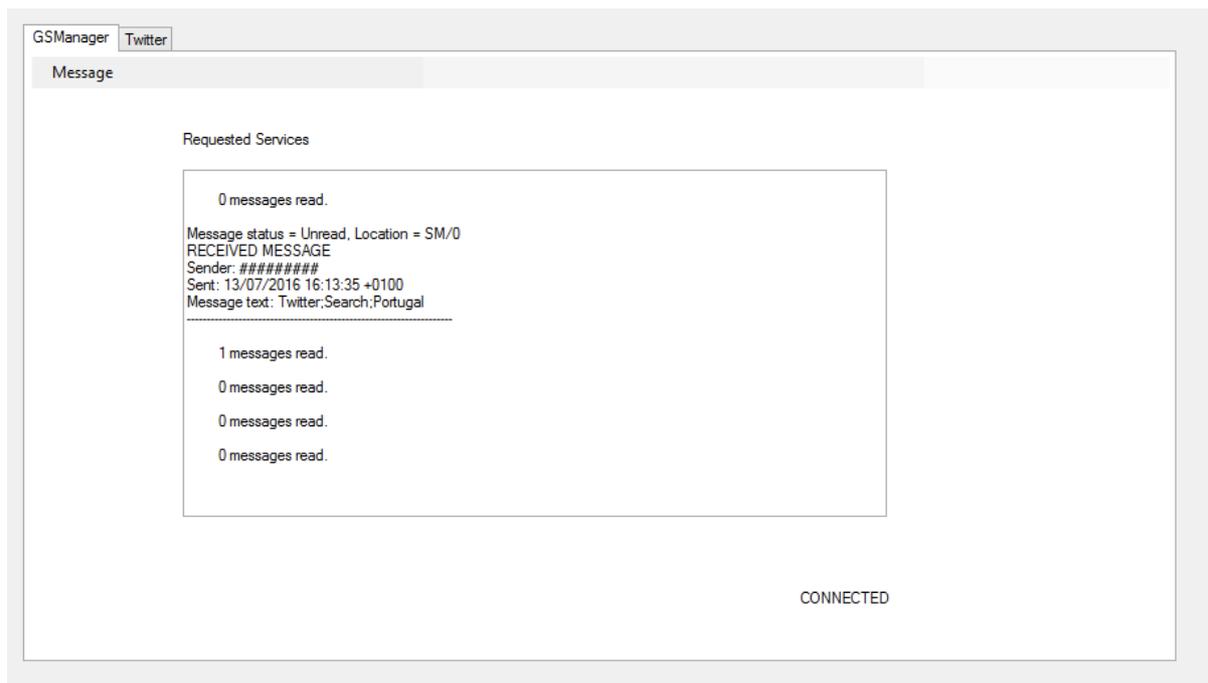


Figure 25: GSMManager – Server interface and runtime communication services response

The GSMManager is a run time interface feature, exclusive to the GSM channels, responsible to pop up the GSM requests received from the Mobile Applications. In the interface is presented the detailed information based on the request. If a request is successfully receive the contact information (sender, date and others personal information) and the message content is displayed in the mobile application. At the content of the request an encoded message is received, containing the service, feature and action.

Therefore, all requests handled from the mobile application to the server and vice versa are 64 bits base encoded, so a script to encode and decode was developed both in the mobile application and at the server. An example of the message structure, for the developed case study, is the following: “Twitter;Search;Portugal”, however in an encoded in a 64 bits base message: “VHdpdHRlclg==;U2VhcmNo;UG9ydHVnYWw=”.

Finally, the HandlRequests is a generic script developed at the server and mobile applications, responsible to receive and send all requests, from different communication channels (Internet, GSM or Bluetooth) and contact the specific service request (however, in this particular case study it only handles Twitter requests). To handle the Twitter request and contact the Twitter servers the Linq2Twitter¹ API is used. Linq2Twitter is an open source 3rd party LINQ Provider for the Twitter micro-blogging service. It uses standard LINQ syntax for queries and includes method calls for changes via the Twitter API². The Twitter API micro-blogging service includes two RESTful APIs (communicate over HTTP with the same HTTP verbs: GET, POST, PUT, and DELETE). The Twitter REST API methods allow developers to access core Twitter data (update timelines, status data, user information, search and trends). This particular service is only developed and handled at the server side. An example of the information gathered from Linq2Twitter can be found at Figure 26.

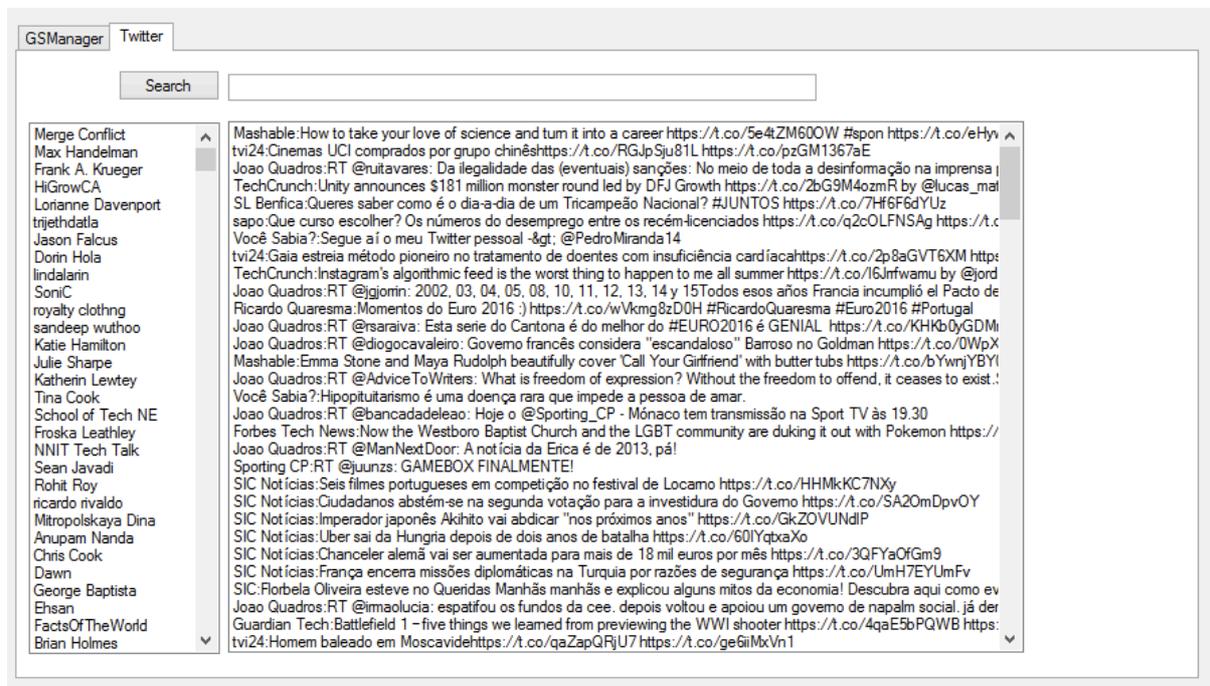


Figure 26: TwitterServices – Server interface of the requested services content (twitter users and posts)

¹ Linq2Twitter – <https://linqtotwitter.codeplex.com/>

² Twitter API – <https://dev.twitter.com/>

TwitterServices (Figure 26) is the Twitter service interface that presents the last Twittes posted by Twitter users. It is an interactive interface since it is possible to search specific Twittes. However, all twitter requests are handled at runtime and therefore there is no time spent to pop up the particular request.

6.3.2. Client Side

The client side handles the mobile application and related services for mobile users. The mobile application was developed with Xamarin Studio Version 6.0.2 based on the model addressed in Chapter 5. However, the model presented in Chapter 5 is a generic model for all mobile OS platforms, for this particular case study it was only implemented an Android prototype. The mobile device used to install and assess this application was an HUAWEI model U8510, with the 2.3.3. Android OS Version, as shown in Figure 27.



Figure 27: Mobile Device – HUAWEI Model U8510-1 with 2.3.3 Android OS Version

It was addressed various technical aspects in order to developed this case study. Following the generic model (Figure 16), presented in Chapter 5, the Twitter service was chosen with three features: Personal Information, Post and Search; then the necessary communication channels supported by the mobile device were identified. For each service feature the required communication channels were crossed referenced. Therefore, in order to develop this mobile application, based on the generic model to guarantee always available services, a set of XML files were defined and build in the mobile application. As defined in the generic model, the XML files are composed with three important roles, namely: Services (Twitter was used in this case study) – Appendix 2, Priorities (available communication channels) – Appendix 3 and

Rules (each service feature associated with the communication channels priority of use) – Appendix 4. For a better readability the XML files are presented in XML graphical format (Figure 28, 30 and 32) and in XSD format (Figure 29, 31 and 33).

The XML Services is responsible to gather all available services at these mobile applications. For each service, is defined the name and URL, followed by the corresponding features (functions) and the related technical aspects, namely input and output parameters. For example, as showed in Figure 28 and 29 – XML Services, the Service 1 – Twitter has two functions “Search” and “Post”, which has input parameters, namely the name (searchString) and data type (string); and also the output parameters the name (tweetList) and the type (array[string]). This format is similar to other services: Service Details, Service Features, Input Parameters and Output Parameters. It is very important to define the XML correctly since it needs to communicate with the Server that will receive the requested service based on the format defined.

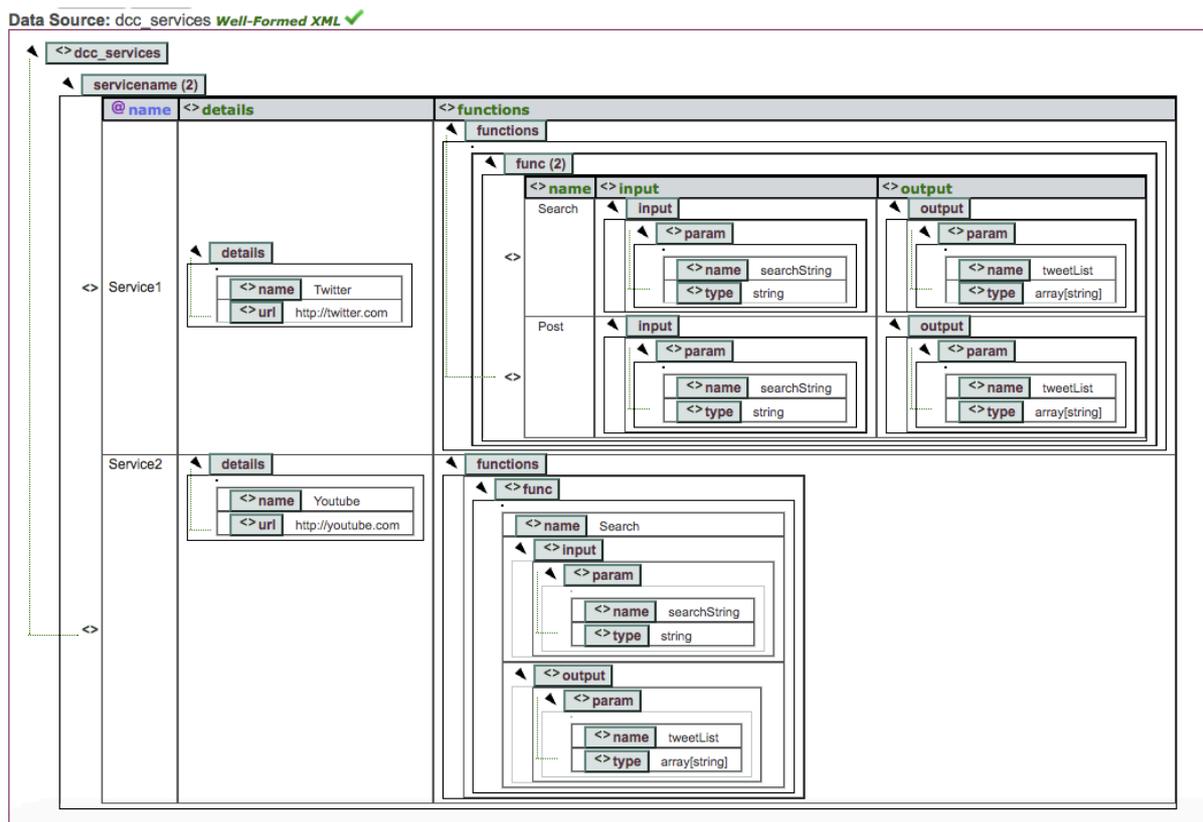


Figure 28: XML Services

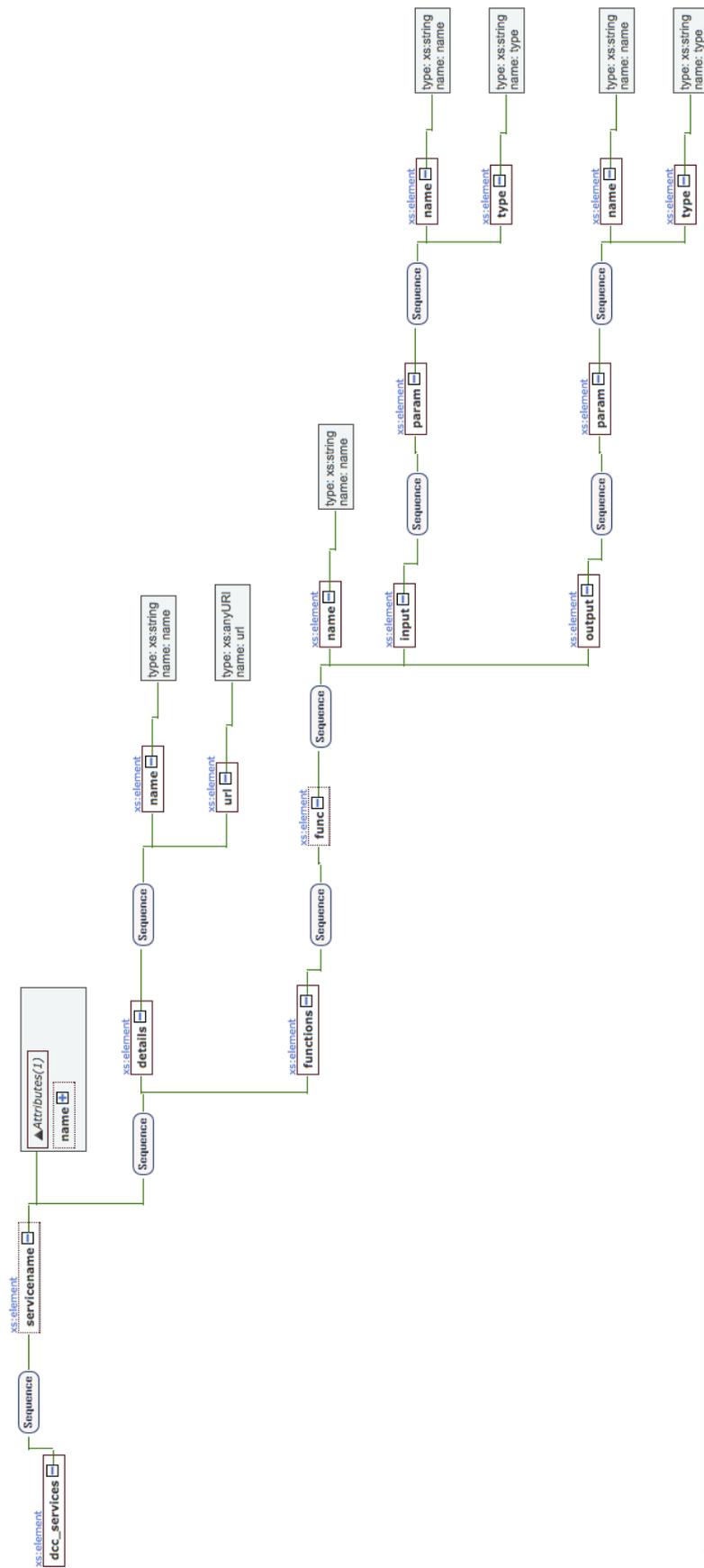


Figure 29:

XSD Services

The XML Priorities – Figure 30 and 31 is responsible to define the available communication channels that will be used by the mobile application. Furthermore, it represents the default priority list to be used and the communication channels information type, for example: the Wifi channel uses information data with the template “*servicename;func_param[param]*”.

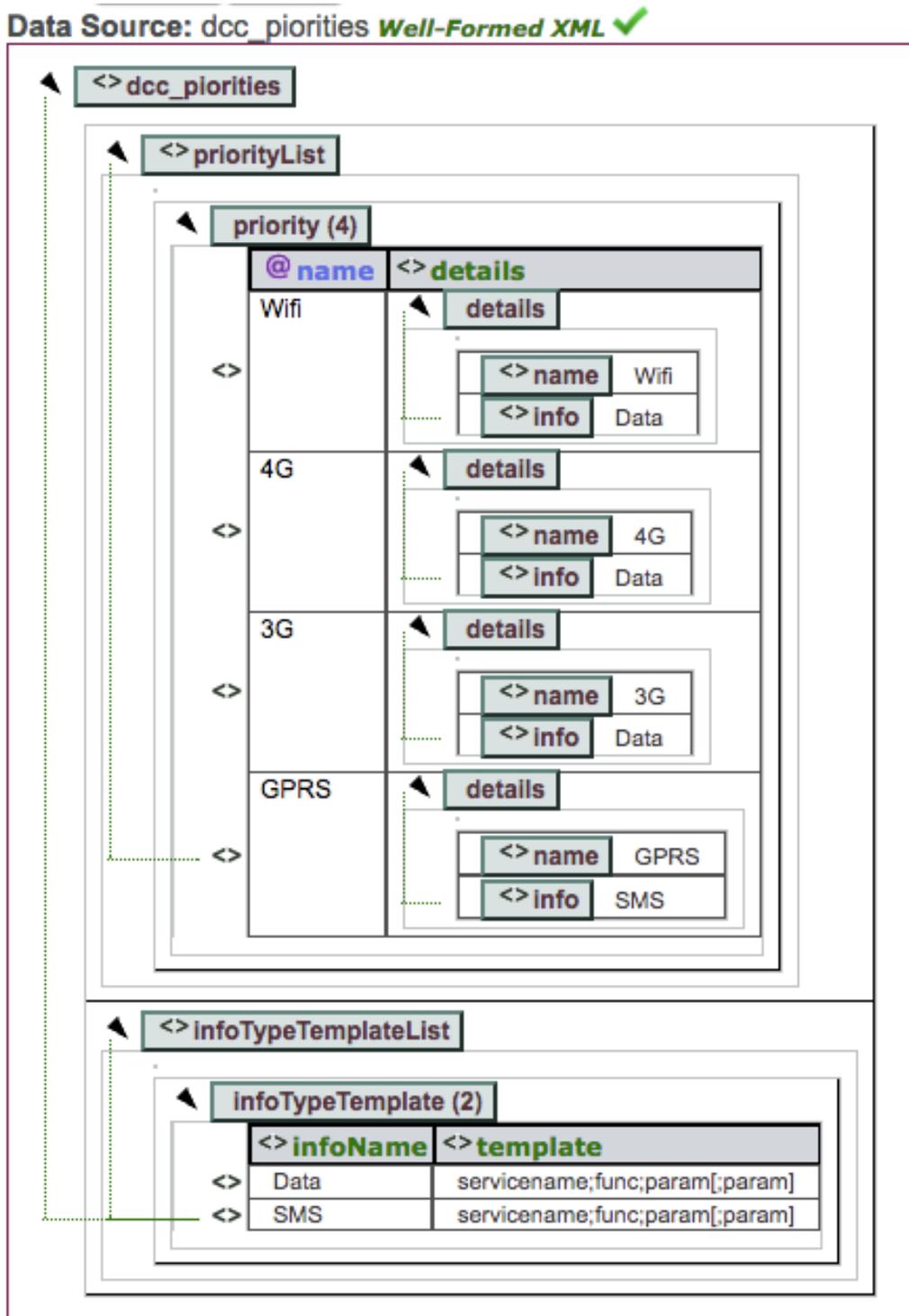


Figure 30: XML Priorities

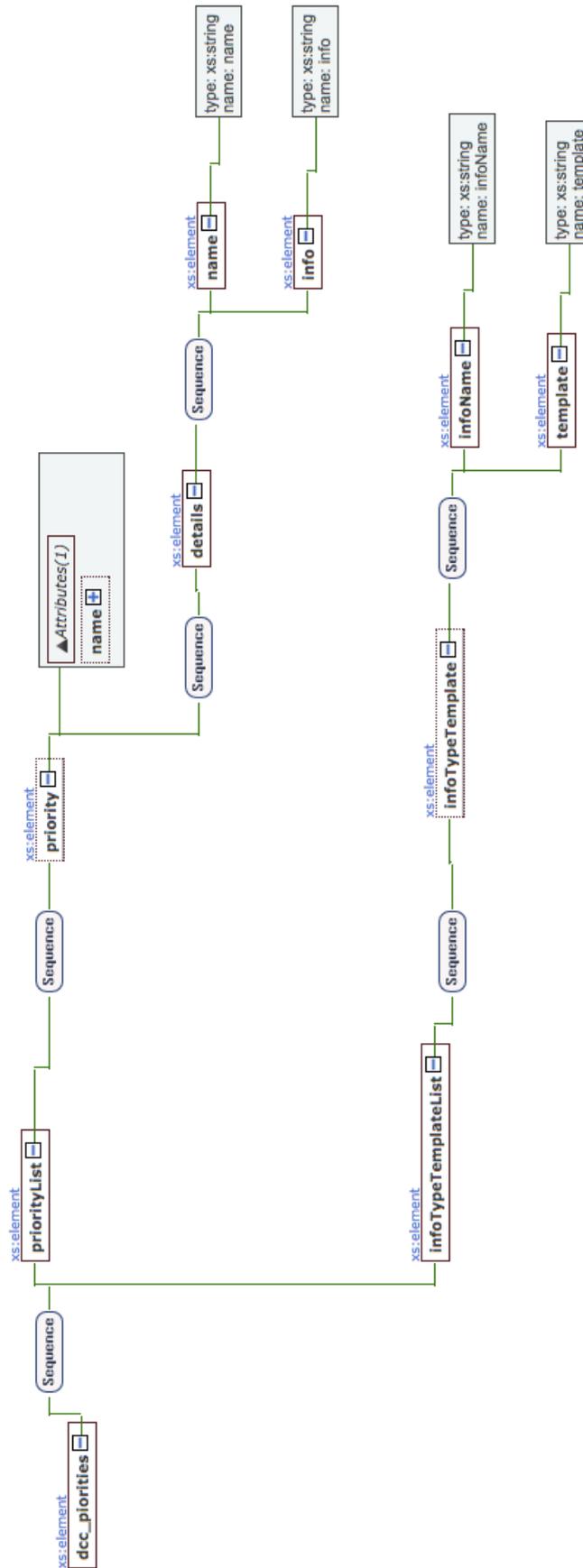


Figure 31: XSD Priorities

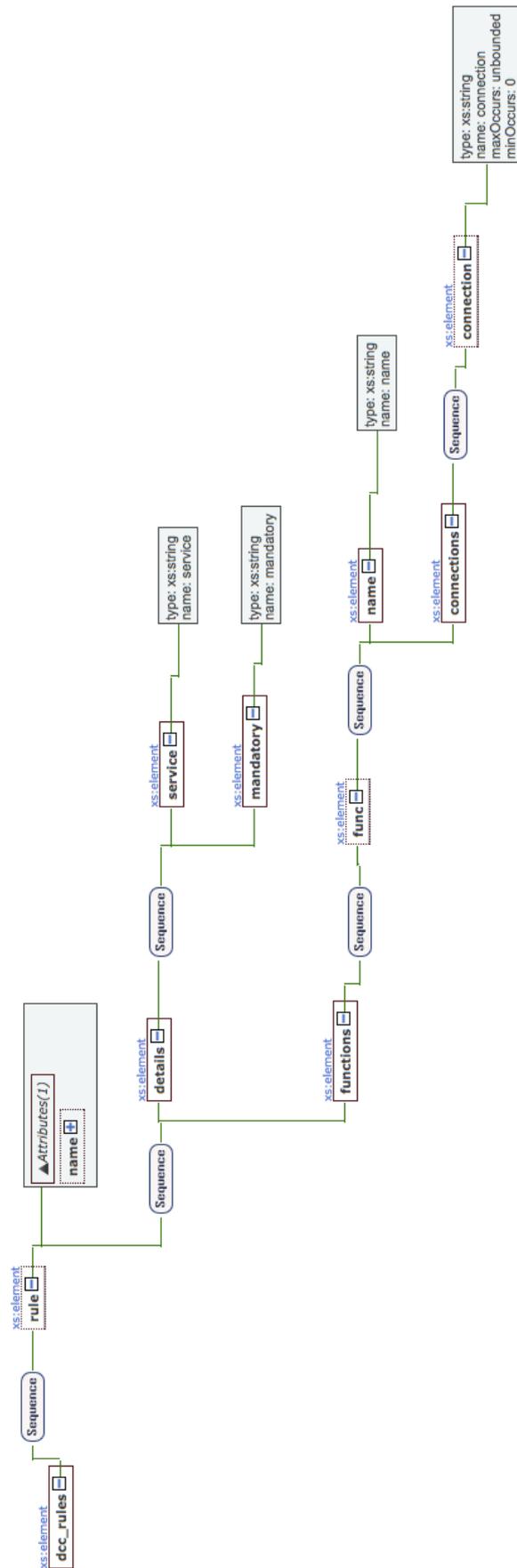


Figure 33: XSD Rules

After defining the XML files and configuring all features at the Server side, the mobile application is ready to be designed and implemented. The mobile application needs to be designed based on the XML files rules and the communication needs to follow the defined format to send request to the server side. As presented in the previous section, all request between the server and the mobile application are encoded 64 bits, therefore the mobile application is responsible to decode and encode every messages. The format used is the same presented in the server side, for example: “Twitter;Search;Portugal”, however it is encoded in a 64 bits message: “VHdpdHRlcg==;U2VhcmNo;UG9ydHVnYWw=”.

The first thing done at the mobile application was to build in the XML files, and make necessary changes so that the mobile application can handle communication changes events based on the defined XML Rules. At this point, this is the most important feature to implement in the mobile application, since it will determinate if the application is able to understand the mobile environment changes at run time and adapt is content.

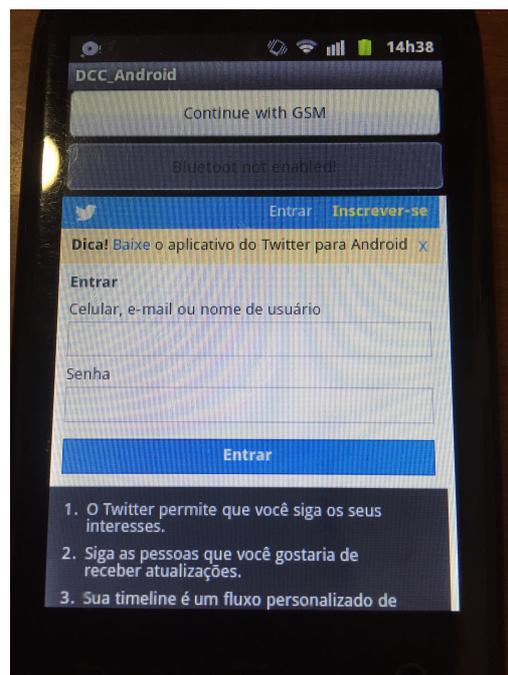


Figure 34: Application Homepage with Internet Connection

In order to present the mobile application content and model in this case study, the interface design is presented in Figure 34 and 35. However, for a perspective of interpretation the set of communication channels, defined in XML files, are presented at the homepage of the application (Figure 34 and 35) allowing the user to choose the ones that he prefers – in case of it being available to use. It is important to notice that depending on the communication channel

used, not only service feature can be present (since it follows the XML rules for each available communication channel). On the other hand, this model is loaded at runtime and in case of some communication channel not being available it will not be showed, for example, if the internet connection or Bluetooth is not available the GSM channels will be used, but the user will not see the interface as showed in Figure 35, the application will simply present the available features for the GSM channel. In case of Figure 35, there were no Internet connection and the mobile device did not support Bluetooth, so based on the defined rules the GSM channel will be used. In this particular case, the available features for the Twitter services were the Personal Information, Search and Post.

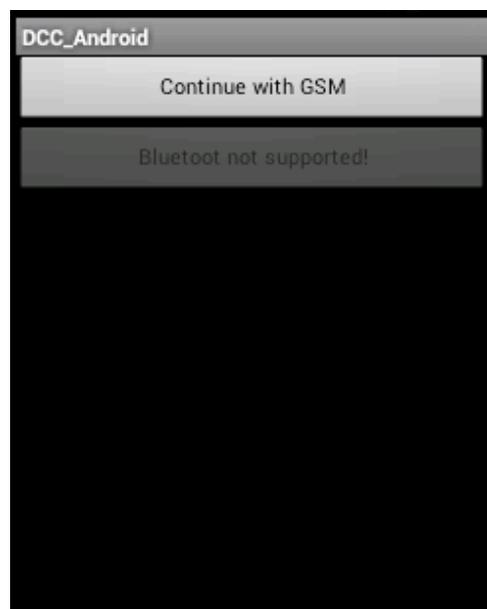


Figure 35: Application Homepage without Internet Connection

The message presented in Figure 36 is displayed in case of the user chooses to “*Continue with GSM*”, however this kind of information will be made displayed upon the installation of the application on the mobile device, taking out the need to be always displayed when the GSM channel is used.

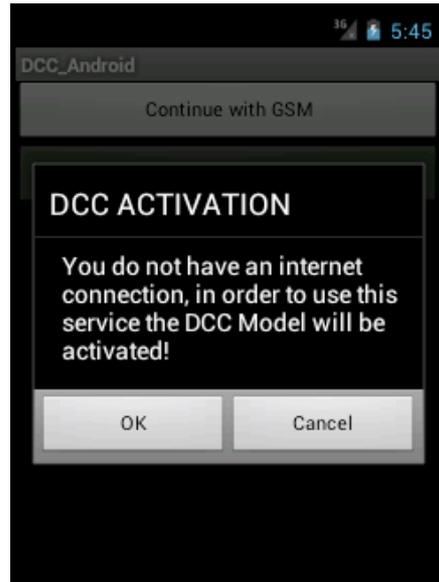


Figure 36: Model Activation Confirmation

If the mobile user accepts the “*DCC ACTIVATION*” the application adapts its content, makes available the defined features and sends a request (64 bits encoded) to the server asking for the user personal information: “*Twitter;Personal;ID*”. At this point, the mobile application is working with the GSM channel and all request between the applications the server are done based on encrypted SMS’s. In case of changes on the environment conditions the application will at runtime change the communication channel and adapt its content once more, to notice that for this case study it only could change to Bluetooth or to Wifi. The mobile application based on GSM channels – Figure 37 already shows the user personal information and can be compared with the Web browser version in Figure 38. The personal information content is composed with: username, followers and tweets; however more information could be displayed.

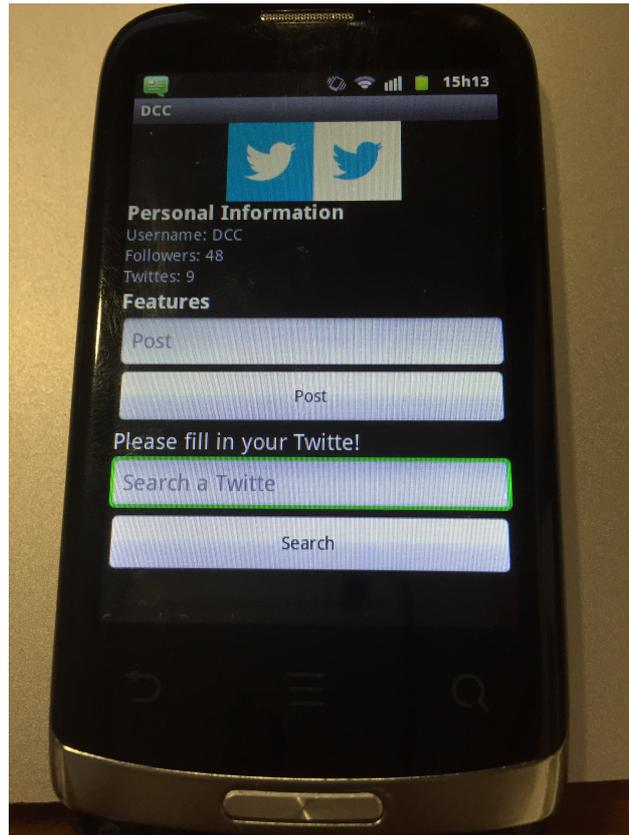


Figure 37: Application twitter Personal Information and features (post and search)



Figure 38: Twitter Personal Information on the Web



Figure 39: Twitter Post example

The mobile user can now perform a post at the Twitter service – Figure 39. In this case it was done a Post with the content “Portugal” with the following encoded request to server: “VHdpdHRlcg0K; UG9zdA==;UG9ydHVnYWw=”. The application waits the server response and in case of successfully post the Tweet it display the decoded message “You Tweet Portugal with Success!”. Figure 40 shows the Tweet performed in a Web browser version.

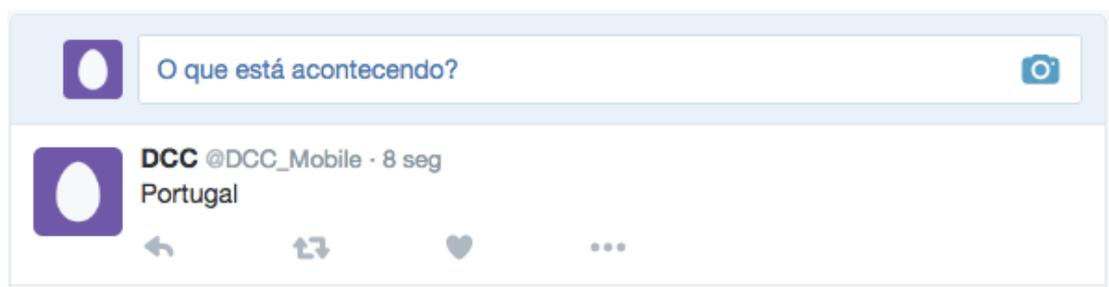


Figure 40: Twitter Post example on the Web

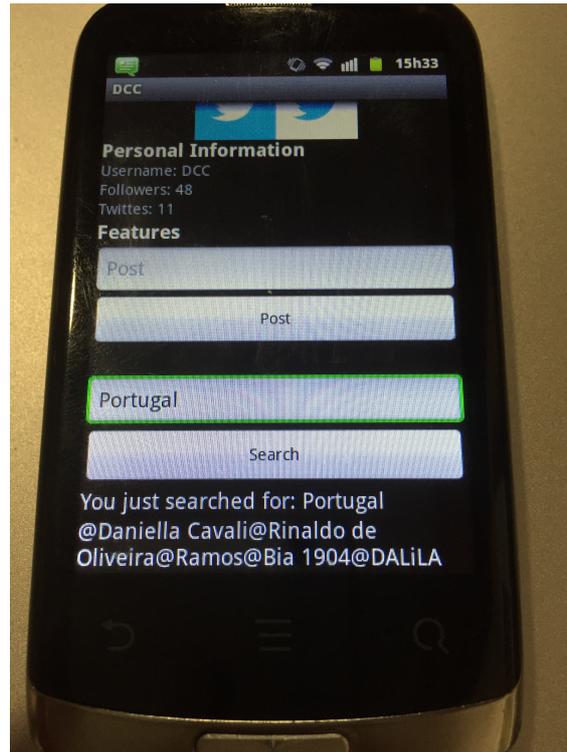


Figure 41: Twitter Search example

Furthermore, mobile users are also able to Search for Twitter users that have tweeted a specific content. In this case the request “*VHdpdHRlcg0K;U2VhcmNo;UG9ydHVnYWw=*” was sent to the server side. The server side decoded the request and through the Linq2Tweet API sent a request to the Twitter Service asking for all users that have tweeted the content “*Portugal*”. The server side encode the specific content and send a response to the mobile application, which decoded the message and displayed the content, has seen in Figure 41.

More features and content can be implemented and made available to mobile users. However, they have to be defined at the XML files and the server side needs to be configured to support the new features or services.

6.3.3. Results Analysis – Unity Tests

In regard to the software field, testing is essential to ensure its quality and reliability, based on a clear system specification. Unit testing is the tests of units that constitute a system, such as functions, modules or components, verifying if they behave as expected, that is, whether their output is accurate, where the units in question should be fixed when it isn't. Therefore, unity test is the process of executing a software system to determine whether it matches its

specification and executes in its intended environment [Whittaker, 2000]. In a subsequent development stage, integration testing (I&T) is also common to be made, i.e., the phase in software testing in which individual software modules are combined and tested as a group [Ould and Unwin, 1986].

The benefits of testing are the detection of problems at an early stage of implementation, which helps to find bugs or errors in the system specification. It also helps to better understand the code and to make changes, since the system will be responsible for verifying if the changes made break existing behaviours [Cleary, 2014].

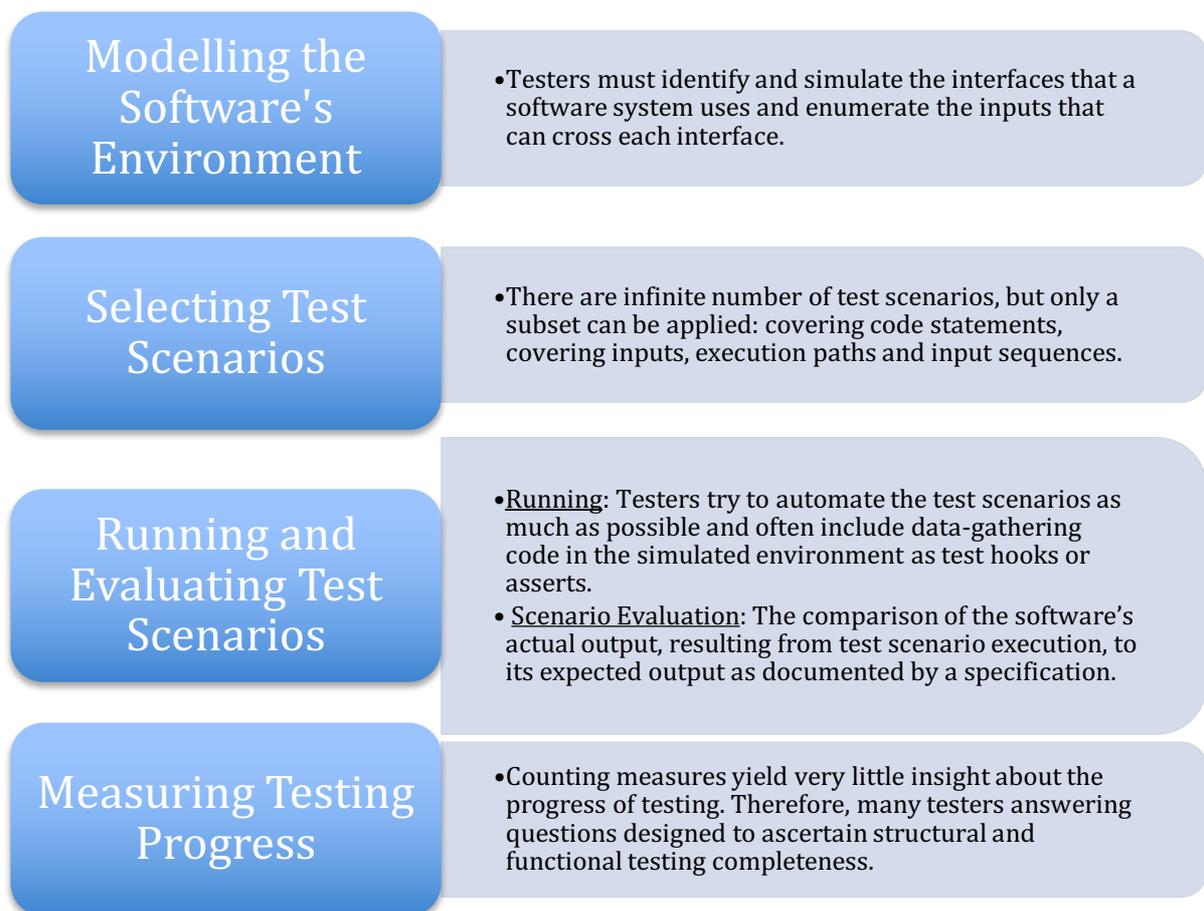


Figure 42: Testing approaches of unity tests, from [Whittaker, 2000]

To get a clearer view of some of software testing's inherent difficulties, we can approach testing in four phases (Figure 42): Modelling the Software's Environment, Selecting Test Scenarios, Running and Evaluating Test Scenarios, and Measuring Testing Progress.

So, for assess the case study presented in this chapter a set of unity tests have been performed, namely:

- **Switching** – The ability to switch from communication channels (Wifi, GSM and Bluetooth) at start up and runtime, based on the available and defined rules.
- **Availability** – Service features (Personal Information, Search and Post) are available at start up or at runtime, independently of the communication channel used (Wifi, GSM and Bluetooth).
- **Time of response** – Waiting time for request and response of services features, for each individual communication channel used (Wifi, GSM and Bluetooth).

Switching Assignments:

- **SA1** – Check if it **loads** the **XML files** (Services, Priorities and Rules) at **start up**;
- **SA2** – Check if it **interpreters** the **Rules XML** files at **start up**;
- **SA3** – Check if **exists** an **Internet** connection at **start up**;
- **SA4** – Check if it **adapts** the application interface **content** based on the **Internet** Connection at **start up**;
- **SA5** – Check if **exists** a **GSM** connection, in case of non-Internet connection, at **start up**;
- **SA6** – Check if it **adapts** the application Interface **content** based on the **GSM** connection at **start up**;
- **SA7** – Check if **exists** a **Bluetooth** connection, in case of non-Internet or GSM connection, at **start up**;
- **SA8** – Check if it **adapts** the application Interface **content** based on the **Bluetooth** connection at **start up**;
- **SA9** – Check if it **interpreters** the **Rules XML** file at **runtime**;
- **SA10** – Check if it **adapts** the application interface **content** based on the **Internet** Connection at **runtime**;
- **SA11** – Check if **exists** a **GSM** connection, in case of non-Internet connection, at **runtime**;
- **SA12** – Check if it **adapts** the application Interface **content** based on the **GSM** connection at **runtime**;

- **SA13** – Check if exists a **Bluetooth** connection, in case of non-Internet or GSM connection, at **runtime**;
- **SA14** – Check if it **adapts** the application Interface **content** based on the **Bluetooth** connection at **runtime**;

Assignments	Output	
	Expected	Result
SA1	Positive	✓
SA2	Positive	✓
SA3	Positive	✓
SA4	Positive	✓
SA5	Positive	✓
SA6	Positive	✓
SA7	Invalid	✓
SA8	Invalid	ⓘ
SA9	Positive	ⓘ
SA10	Positive	✓
SA11	Positive	✓
SA12	Positive	✓
SA13	Invalid	ⓘ
SA14	Invalid	ⓘ

Table 7: Switching assignments results

Symbol's labels:

✓ – The result is the expected.

ⓘ – The result is the expected, but is limited to the Bluetooth range and therefore has not been implemented (see Section 6.4 and Chapter 7).

✗ – The result is not the expected.

Availability Assignments:

- **AA1** – Twitter **personal information** is **available** with **Internet** connection at **start up**;
- **AA2** – Twitter **search** is **available** with **Internet** connection at **start up**;
- **AA3** – Twitter **post** is **available** with **Internet** connection at **start up**;
- **AA4** – Twitter **personal information** is **available** with **GSM** connection at **start up**;
- **AA5** – Twitter **search** is **available** with **GSM** connection at **start up**;
- **AA6** – Twitter **post** is **available** with **GSM** connection at **start up**;
- **AA7** – Twitter **personal information** is **available** with **Bluetooth** connection at **start up**;
- **AA8** – Twitter **search** is **available** with **Bluetooth** connection at **start up**;
- **AA9** – Twitter **post** is **available** with **Bluetooth** connection at **start up**;
- **AA10** – Twitter **personal information** is **available** with **Internet** connection at **runtime**;
- **AA11** – Twitter **search** is **available** with **Internet** connection at **runtime**;
- **AA12** – Twitter **post** is **available** with **Internet** connection at **runtime**;
- **AA13** – Twitter **personal information** is **available** with **GSM** connection at **runtime**;
- **AA14** – Twitter **search** is **available** with **GSM** connection at **runtime**;
- **AA15** – Twitter **post** is **available** with **GSM** connection at **runtime**;
- **AA16** – Twitter **personal information** is **available** with **Bluetooth** connection at **runtime**;
- **AA17** – Twitter **search** is **available** with **Bluetooth** connection at **runtime**;
- **AA18** – Twitter **post** is **available** with **Bluetooth** connection at **runtime**;

Assignments	Output	
	Expected	Result
AA1	Positive	✓
AA2	Positive	✓
AA3	Positive	✓
AA4	Positive	✓
AA5	Positive	✓
AA6	Positive	✓
AA7	Invalid	ⓘ
AA8	Invalid	ⓘ
AA9	Invalid	ⓘ
AA10	Positive	✓
AA11	Positive	✓
AA12	Positive	✓
AA13	Positive	✓
AA14	Positive	✓
AA15	Positive	✓
AA16	Invalid	ⓘ
AA17	Invalid	ⓘ
AA18	Invalid	ⓘ

Table 8: Availability assignments results

Symbol's labels:

✓ – The result is the expected.

ⓘ – The result is the expected, but is limited to the Bluetooth range and therefore has not been implemented (see Section 6.4 and Chapter 7).

✗ – The result is not the expected.

Time of response Assignments:

- **TA1** – Time needed to gather Twitter **Personal Information** with an **Internet** connection;
- **TA2** – Time needed to **search** a Tweet with an **Internet** connection;
- **TA3** – Time needed to **post** a Tweet with an **Internet** connection;
- **TA4** – Time needed to gather Twitter **Personal Information** with a **GSM** connection;
- **TA5** – Time needed to **search** a Tweet with a **GSM** connection;
- **TA6** – Time needed to **post** a Tweet with a **GSM** connection;
- **TA7** – Time needed to gather Twitter **Personal Information** with a **Bluetooth** connection;
- **TA8** – Time needed to **search** a Tweet with a **Bluetooth** connection;
- **TA9** – Time needed to **post** a Tweet with a **Bluetooth** connection;

Assignments	Output Time of Request and Response	
	Expected	Result (Average ms)
TA1	Low	1426 ✓
TA2	Low	800 ✓
TA3	Low	1480 ✓
TA4	Medium	31926 ⓘ
TA5	Medium	31300 ⓘ
TA6	Medium	31980 ⓘ
TA7	Invalid	–
TA8	Invalid	–
TA9	Invalid	–

Table 9: Time of response assignments results

Symbol's labels:

✓ – The result is the expected.

ⓘ – The result is the expected, but duo to unexpected conditions the response time is higher (see Section 6.4).

✗ – The result is not the expected.

6.4. Conclusion

With the unity tests analyse there were unexpected behaviours occurring in the system, duo or based in different conditions and intervenient. Although the results of the tasks performed are all valid, there are some warnings/conditions that show the need to revise the system specification, since, in those situations, the application is not supporting the requirement specifications. It is a cyclical process, where the tests purpose is exactly for finding these situations to improve the overall behaviour of the system.

Bluetooth communication settings in big distance did not work, the model should be prepared to communicate with other mobile devices using the mobile application developed with the proposed model, so that those mobile devices could be used as a bridge to contact the server when a Bluetooth communication is required. In this sense, and following the Bluetooth requirements, with the case study presented, the mobile device needed to be near by the server, however in terms of mobility that is not achievable, and therefore the test results were considered invalid. Furthermore, the time of response for GSM communications presented are all above the 30000 milliseconds, which can be considered a high response time. However, since it uses SMS to communicate the test performance were restricted to the limited allowed by the SIM card, which needed to be reset after 20/25 SMS received. With such restrict conditions, to the number of SMS's allowed, it made also that some of service requests could not be received by the server or by the mobile device making the service unreachable. Therefore, these were the reasons that made the GSM response time being higher than expected. The GSM communication response time is based in: request a feature service through SMS; contact the required service by the server; and response through SMS to the mobile device.

Chapter 7. Final Remarks and Future Work

This chapter presents an overview of the most relevant points of the research, a critical analysis and assessment of the model and architecture. First of all, in section 7.1, the conclusions of the work are addressed, with particular emphasis on the achievement of the proposed objectives. Finally, in section 7.2, indication of potential lines for future research in order to continue the work is pointed out.

7.1. Final Remarks

The core of this research is a model and a prototype implementation that can represent a solution to fulfil the existing gap in mobile applications services availability. A set of research questions were introduced in section 1.3 of the current document. All of them were taken into consideration during the evolution of the current research and are now answered in detail:

(RQ1) – What is the need for a mobile device to be always-connected?

In section 6.2 the results of the questionnaire are introduced, which allowed to understand that mobile users want to have access to information despite of their location. They just want to fill that they are connected to the world.

(RQ2) – What is the dependence of people on the perception of connectivity?

Mobile users want to be always-connected to the Internet in order to have access to their services and applications (Chapter 6). Most of them do not have the skills or are aware of the mobile features to allow them to switch or to gain access to mobile channels, hereby they are dependent of the available communication channels and therefore dependents on the perception of connectivity. Furthermore, this perception will be study in future work by making test with real users.

(RQ3) – It is possible for a device to be virtually always connected?

Based on the current research a device connectivity may be very unstable and variable depending on the environment limitations (Chapter 2 and 3). However, and in extreme situation where no Internet connectivity is available, switching model mechanisms can be used based on pre-defined rules making the minimum services features available.

(RQ4) – Ability to generalize the existing connectivity problems with mobile devices in highly instable environments, by abstracting the communication layer from developers and ensure that the priority services are still available?

Applying the presented model in an architecture as the presented in Chapter 4 – SaaS4Mobile and by exposing an API, mobile developers can be abstracted by a communication layer and further setup their application services through different communication channels.

(RQ5) – The mobile applications features or services can be classified as principal and secondary, to guarantee the availability of those that have higher priority?

In the current mobile ecosystem, mobile applications expose various services through different features. However, most of them are nice to have features and mobile users may even not be aware of such functionalities. Hereby, there are main features – the most used by users and therefore with higher priority – that can be considered the principal services that should be made always available (for example, consider a mobile user that wants to post in Facebook: one with just a message and other with a video. In a situation with no Internet connectivity, however with the proposed model implemented, the post feature will only make available the post for message but not the one for videos).

(RQ6) – It is possible to define a model that adapts the mobile application features based on the switching of the available communication channels?

In chapter 5 is presented a model for dynamic switching of mobile communication channels on the development of mobile applications, in order to guarantee transparency and at the same time improve the efficiency of services availabilities. This model achieves the wanted results through a set of pre-defined rules for each service, therefore it chooses the best available communication channel and at the same time it adapts the interface and the available content (based on the defined rules) on the mobile application.

(RQ7) – What are the model implications in applying it to existing frameworks and to real world applications scenarios, and also in terms of performance issues?

Exposing an API to be deployed into the existing frameworks will allow a rapid integration and developing through an abstraction communication layer. In section 4.2 real world application scenarios are addressed to show the potential on applying the proposed model in mobile

application. Finally, in section 6.4 the possible issues and performance loss are addressed, such as the delay in response that could occur.

The exposed research questions allowed to prove the proposed thesis of this document, namely:

The connectivity problem in mobile devices is generalizable and in this sense an abstraction can be created with reduced implications in the development and adaptation of existing applications to allow the services availability of the applications in highly variable environments.

Therefore, the solution proposed in this project solves the problems presented in the state of the art, by providing a model for a service brokering communication that enables the rapid integration of enterprise applications with mobile applications, using an adaptive and transparent communication layer to ensure full connectivity under limited communication conditions. Furthermore, by taking advantage of the dynamic communication channels it is now possible to have an always-connected application despite of the lack of Internet connection. In summary, it is presented an interchangeable communication layer for mobile applications to ensure the transparency of the communication channels through a framework for the development of mobile applications in distributed environment to guarantee the upper cited communication transparency.

Thus, and in spite of this thesis only presenting a small contribution to a bigger and better practical realisation of what is intended to improve the service availability for mobile users, the model specified herein is governed by based that literature considers feasible and reliable, which could become relevant to research in this field.

7.2. Future Work

The conceptualised model in the context of this thesis is intended to contribute to the evolution and ongoing discussion of best technological solutions that, in an increasingly decisively way to improve the mobile services availability.

This manuscript describes only a proof of concept for the Android platform, therefore will also be required an implementation to other existing mobile platforms (iOS and Windows Phone). Furthermore, is also intended to address the SaaS Mobile presented in Chapter 4 and an API so that mobile developers can easily address the proposed contribution model. Taking advantage of this API other communication channel should also be supported by the server and mobile applications. To prove and assess the model in extreme situation and conditions it will be address an application with more services features through tests with real user in a bigger scale. Assessing the presented model in such conditions can be done through different techniques namely questionnaires for users, questionnaires for observation and other unity tests.

References

- Abolfazli, S., Sanaei, Z., Gani, A., Xia, F., & Yang, L. T. (2014). "Rich mobile applications: genesis, taxonomy, and open issues". *Journal of Network and Computer Applications*, 40, 345-362.
- Ainscough, D. G. (2001). "The Evolution and Future of Mobile Communication Systems". *Viitattu*, 15(1), 2015.
- Alam, M. M., Arbia, D. B., & Hamida, E. B. (2015, April). "Research trends in multi-standard device-to-device communication in wearable wireless networks". In *International Conference on Cognitive Radio Oriented Wireless Networks* (pp. 735-746). Springer International Publishing.
- Ali, M. (2009, December). "Green cloud on the horizon". In *IEEE International Conference on Cloud Computing* (pp. 451-459). Springer Berlin Heidelberg.
- Alia, M., Hallsteinsen, S., Paspallis, N., & Eliassen, F. (2007, June). "Managing distributed adaptation of mobile applications". In *IFIP International Conference on Distributed Applications and Interoperable Systems* (pp. 104-118). Springer Berlin Heidelberg.
- Almenares, F., Arias, P., Marin, A., Diaz-Sanchez, D., & Sanchez, R. (2013). "Overhead of using secure wireless communications in mobile computing". *IEEE Transactions on Consumer Electronics*, 59(2), 335-342.
- Apple (2017). Retrieved from <https://developer.apple.com/swift/> at May 2017
- Armbrust, M., Fox, A., Griffith, R., Joseph, A. D., Katz, R. H., Konwinski, A., & Zaharia, M. (2009). "Above the clouds: A berkeley view of cloud computing".
- Atmaca, S., Ceken, C., & Erturk, I. (2009). "A new QoS-aware TDMA/FDD MAC protocol with multi-beam directional antennas". *Computer Standards & Interfaces*, 31(4), 816-829.
- Bahl, P., Han, R. Y., Li, L. E., & Satyanarayanan, M. (2012, June). "Advancing the state of mobile cloud computing". In *Proceedings of the third ACM workshop on Mobile cloud computing and services* (pp. 21-28). ACM.

- Balan, R. K., Satyanarayanan, M., Park, S. Y., & Okoshi, T. (2003, May). "Tactics-based remote execution for mobile computing". In Proceedings of the 1st international conference on Mobile systems, applications and services (pp. 273-286). ACM.
- Barbarossa, S., Sardellitti, S., & Di Lorenzo, P. (2014). "Communicating while computing: Distributed mobile cloud computing over 5G heterogeneous networks". IEEE Signal Processing Magazine, 31(6), 45-55.
- Beck, M. T., & Maier, M. (2014). "Mobile Edge Computing: Challenges for Future Virtual Network Embedding Algorithms". In Proceedings of the 8th International Conference on Advanced Engineering Computing and Applications in Sciences (ADVCOMP) (pp. 65-70).
- Beck, M. T., Feld, S., Fichtner, A., Linnhoff-Popien, C., & Schimper, T. (2015, February). "2Me-volte: Network functions for energy-efficient video transcoding at the mobile edge". In Intelligence in Next Generation Networks (ICIN), 2015 18th International Conference on (pp. 38-44). IEEE.
- Bhagwan, R., Savage, S., & Voelker, G. M. (2003, February). "Understanding availability". In International Workshop on Peer-to-Peer Systems (pp. 256-267). Springer Berlin Heidelberg.
- Bora, G., Bora, S., Singh, S., & Arsalan, S. M. (2014). "OSI reference model: An overview". International Journal of Computer Trends and Technology (IJCTT, 7(4).
- Borst, C., Wimböck, T., Schmidt, F., Fuchs, M., Brunner, B., Zacharias, F., & Rink, C. (2009, May). "Rollin'Justin-Mobile platform with variable base". In ICRA (pp. 1597-1598).
- Cachin, C., & Schunter, M. (2011). "A cloud you can trust". IEEE Spectrum, 48(12), 28-51.
- Campbell, A., & Choudhury, T. (2012). "From smart to cognitive phones". IEEE Pervasive Computing, 3(11), 7-11.
- Carolan, J., Gaede, S., Baty, J., Brunette, G., Licht, A., Remmell, J., & Weise, J. (2009). "Introduction to cloud computing architecture". White Paper, 1st edn. Sun Micro Systems Inc.
- Cavus, N., & Ibrahim, D. (2008). "A Mobile Tool for Learning English Words". Online Submission.
- Chang, Y. F., Chen, C. S., & Zhou, H. (2009). "Smart phone for mobile commerce". Computer Standards & Interfaces, 31(4), 740-747.

Charland, A., & Leroux, B. (2011). "Mobile application development: web vs. native". *Communications of the ACM*, 54(5), 49-53.

Chu, H.; Song, H.; Wong, C.; Kurakake, S.; Katagiri, M.: "Roam, a seamless application framework", *Journal of Systems and Software*, 2004.

Chun, B. G., Ihm, S., Maniatis, P., Naik, M., & Patti, A. (2011, April). "Clonecloud: elastic execution between mobile device and cloud". In *Proceedings of the sixth conference on Computer systems* (pp. 301-314). ACM.

Cleary, S. (2014). "Concurrency in C# Cookbook". " O'Reilly Media, Inc."

Couderc, P., & Kermarrec, A. M. (1999). "Improving level of service for mobile users using context-awareness". In *Reliable Distributed Systems, 1999. Proceedings of the 18th IEEE Symposium on* (pp. 24-33). IEEE.

Cox, C. (2012). "An introduction to LTE: LTE, LTE-advanced, SAE and 4G mobile communications". John Wiley & Sons.

Cuervo, E., Balasubramanian, A., Cho, D. K., Wolman, A., Saroiu, S., Chandra, R., & Bahl, P. (2010, June). "MAUI: making smartphones last longer with code offload". In *Proceedings of the 8th international conference on Mobile systems, applications, and services* (pp. 49-62). ACM.

Davies, N., Wade, S. P., Friday, A., & Blair, G. S. (1997). "Limbo: A tuple space based platform for adaptive mobile applications". In *Open Distributed Processing and Distributed Platforms* (pp. 291-302). Springer US.

Deng, S., Huang, L., Taheri, J., & Zomaya, A. Y. (2015). "Computation offloading for service workflow in mobile cloud computing". *IEEE Transactions on Parallel and Distributed Systems*, 26(12), 3317-3329.

Dinh, H. T., Lee, C., Niyato, D., & Wang, P. (2013). "A survey of mobile cloud computing: architecture, applications, and approaches". *Wireless communications and mobile computing*, 13(18), 1587-1611.

Dominique Guinard, Vlad Trifa, Friedemann Mattern, and Erik Wilde (2011). "From the internet of things to the web of things: Resource-oriented architecture and best practices". In

Dieter Uckelmann, Mark Harrison, and Florian Michahelles, editors, *Architecting the Internet of Things*, pages 97–129. Springer Berlin Heidelberg, 2011.ss

Doolan, D. C., Tabirca, S., & Yang, L. T. (2008, November). “Mmpi a message passing interface for the mobile environment”. In *Proceedings of the 6th International Conference on Advances in Mobile Computing and Multimedia* (pp. 317-321). ACM.

Ergen, M. (2009). “Mobile broadband: including WiMAX and LTE”. Springer Science & Business Media.

Ericsson, (2012). “Ericsson Mobility Report”, [Online], Retrieved on January 7, 2012, from <http://www.ericsson.com/res/docs/2012/ericsson-mobility-report-november-2012.pdf>.

Fangxiong, W., & Zhiyong, J. (2004, July). “Research on a distributed architecture of mobile GIS based on WAP”. In *Proceedings of XXth ISPRS Congress* (pp. 12-23).

Fernando, N., Loke, S. W., & Rahayu, W. (2013). “Mobile cloud computing: A survey”. *Future Generation Computer Systems*, 29(1), 84-106.

Flinn, J., Park, S., & Satyanarayanan, M. (2002). “Balancing performance, energy, and quality in pervasive computing”. In *Distributed Computing Systems, 2002. Proceedings. 22nd International Conference on* (pp. 217-226). IEEE.

Forman, G. H., & Zahorjan, J. (1994). “The challenges of mobile computing”. *Computer*, 27(4), 38-47.

Fuggetta, A., Picco, G. P., & Vigna, G. (1998). “Understanding code mobility”. *IEEE Transactions on software engineering*, 24(5), 342-361.

Garcia Lopez, P., Montresor, A., Epema, D., Datta, A., Higashino, T., Iamnitchi, A., & Riviere, E. (2015). “Edge-centric computing: Vision and challenges”. *ACM SIGCOMM Computer Communication Review*, 45(5), 37-42.

Gerpott, T. J. (2015). “SMS use intensity changes in the age of ubiquitous mobile Internet access—A two-level investigation of residential mobile communications customers in Germany”. *Telematics and Informatics*, 32(4), 809-822.

Guinard, D., & Trifa, V. (2016). “Building the Web of Things: With examples in Node.js and Raspberry Pi”, ISBN:1617292680 9781617292682.

Guo, X., Liu, L., Chang, Z., & Ristaniemi, T. (2016, June) "Data offloading and task allocation for cloudlet-assisted ad hoc mobile clouds". *Wireless Networks*, 1-10. doi:10.1007/s11276-016-1322-z

Hall, S. P., & Anderson, E. (2009). "Operating systems for mobile computing". *Journal of Computing Sciences in Colleges*, 25(2), 64-71.

Heuer, J., Hund, J., and Pfaff, O., (2015). "Toward the web of things: Applying web technologies to the physical world". *Computer*, 48(5):34–42, May 2015.

Hevner, A., & Chatterjee, S. (2010). "Design research in information systems: theory and practice" (Vol. 22). Springer Science & Business Media.

Huang, D., Zhang, X., Kang, M., & Luo, J. (2010, June). "MobiCloud: building secure cloud framework for mobile computing and communication". In *Service Oriented System Engineering (SOSE), 2010 Fifth IEEE International Symposium on* (pp. 27-34). IEEE.

Huang, Y., Su, H., Sun, W., Zhang, J. M., Guo, C. J., Xu, J. M., & Zhu, J. (2010). "Framework for building a low-cost, scalable, and secured platform for Web-delivered business services". *IBM Journal of Research and Development*, 54(6), 4-1.

Huerta-Canepa, G., & Lee, D. (2010, June). "A virtual cloud computing provider for mobile devices". In *Proceedings of the 1st ACM Workshop on Mobile Cloud Computing & Services: Social Networks and Beyond* (p. 6). ACM.

Igarashi, T., Takai, J., & Yoshida, T. (2005). Gender differences in social network development via mobile phone text messages: A longitudinal study. *Journal of Social and Personal Relationships*, 22(5), 691-713.

Intelligence, G. S. M. A. (2014). "Understanding 5G: Perspectives on future technological advancements in mobile". London, UK.

ITU - International Telecommunication Union (2011). "The world in 2011: ICT facts and figures". Geneva, Statistics retrieved February 20, 2012. <http://www.itu.int/ITU-D/ict/facts/2011/index.html>.

Jacob Beal, Danilo Pianini, and Mirko Viroli (2015). "Aggregate programming for the internet of things". *IEEE Computer*, 48(9):22–30, 2015.

Jie, Z. (2010). "Research about the Model of Library Information Resources Sharing in the" Cloud Computing". *Journal of Intelligence*, 2, 185-187.

Jin, X., & Kwok, Y. K. (2010, December). "Cloud assisted P2P media streaming for bandwidth constrained mobile subscribers". In *Parallel and Distributed Systems (ICPADS), 2010 IEEE 16th International Conference on* (pp. 800-805). IEEE.

Jing, Jin, and Karen Huff (1998). "Adaptation for Mobile Workflow Applications." *Proceedings of Workshop on Modeling and Simulation in Wireless Systems, Montreal, Canada*. 1998.

Kao, Y. W., Peng, P. Y., Hsieh, S. L., & Yuan, S. M. (2007, November). "A client framework for massively multiplayer online games on mobile devices". In *Convergence Information Technology, 2007. International Conference on* (pp. 48-53). IEEE.

Kao, Y. W., Lin, C., Yang, K. A., & Yuan, S. M. (2012). "A Web-based, Offline-able, and Personalized Runtime Environment for executing applications on mobile devices". *Computer Standards & Interfaces*, 34(1), 212-224.

Katz, R. H. (1994). "Adaptation and mobility in wireless information systems". *IEEE Personal Communications*, 1(1), 6-17.

Kaur, P., & Makkar, G. (2016, April-June) "Augmenting the Computational Capabilities of Mobile Devices with Cloud Computing". *Advances in Computer Science and Information Technology (ACSIT)* p-ISSN: 2393-9907; e-ISSN: 2393-9915; Volume 3, Issue 3; April-June, 2016, pp. 152-156

Kemp, R., Palmer, N., Kielmann, T., & Bal, H. (2010, October). "Cuckoo: a computation offloading framework for smartphones". In *International Conference on Mobile Computing, Applications, and Services* (pp. 59-79). Springer Berlin Heidelberg.

Klein, A., Mannweiler, C., Schneider, J., & Schotten, H. D. (2010, May). "Access schemes for mobile cloud computing". In *2010 eleventh international conference on mobile data management* (pp. 387-392). IEEE.

Kristensen, M. D. (2010, March). "Scavenger: Transparent development of efficient cyber foraging applications". In *Pervasive Computing and Communications (PerCom), 2010 IEEE International Conference on* (pp. 217-226). IEEE.

Kukulska-Hulme, A., & Shield, L. (2008). "An overview of mobile assisted language learning: From content delivery to supported collaboration and interaction". *ReCALL*, 20(03), 271-289.

La Polla, M., Martinelli, F., & Sgandurra, D. (2013). "A survey on security for mobile devices". *IEEE communications surveys & tutorials*, 15(1), 446-471.

Lange, L., Blotny, A., & Magedanz, T. (2011, February). "An Agile Service Platform for Telecommunication Environments". In *New Technologies, Mobility and Security (NTMS), 2011 4th IFIP International Conference on* (pp. 1-6). IEEE.

Latronico, E., Lee, E.A., Lohstroh, M., Shaver, C., Wasicek, A., and Weber, M.s (2015). "A vision of swarmlets". *Internet Computing, IEEE*, 19(2):20–28, Mar 2015.

Leporini, B., Buzzzi, M. C., & Buzzzi, M. (2011). "INTERACTING with the iPad via VoiceOver: accessibility and usability issues". In *Proc. of the Workshop on Mobile Accessibility at INTERACT*.

Li, X., Zhang, H., & Zhang, Y. (2009, December). "Deploying mobile computation in cloud service". In *IEEE International Conference on Cloud Computing* (pp. 301-311). Springer Berlin Heidelberg.

Li, L., Li, X., Youxia, S., & Wen, L. (2010, October). "Research on mobile multimedia broadcasting service integration based on cloud computing". In *Multimedia Technology (ICMT), 2010 International Conference on* (pp. 1-4). IEEE.

Lina Yao, Q.Z. Sheng, and S. Dustdar (2015). "Web-based management of the internet of things". *Internet Computing, IEEE*, 19(4):60–67, July 2015.

Liu, K., Peng, J., Li, H., Zhang, X., & Liu, W. (2016). "Multi-device task offloading with time-constraints for energy efficiency in mobile cloud computing". *Future Generation Computer Systems*, 64, 1-14.

Lu, X., Watanabe, J., Liu, Q., Uji, M., Shono, M., & Kitamura, T. (2011). "Internet and mobile phone text-messaging dependency: Factor structure and correlation with dysphoric mood among Japanese adults". *Computers in Human Behavior*, 27(5), 1702-1709.

Luan, T. H., Gao, L., Li, Z., Xiang, Y., & Sun, L. (2015). "Fog computing: Focusing on mobile users at the edge". *arXiv preprint arXiv:1502.01815*.

- Luigi Atzori, Antonio Iera, and Giacomo Morabito (2010). "The internet of things: A survey". *Computer Networks*, 54(15):2787–2805, 2010.s
- Manco, F., Martins, J., Yasukata, K., Mendes, J., Kuenzer, S., & Huici, F. (2015). "The case for the superfluid cloud". In 7th USENIX Workshop on Hot Topics in Cloud Computing (HotCloud 15).
- Manyika, James, Michael Chui, Peter Bisson, Jonathan Woetzel, Richard Dobbs, Jacques Bughin, and Dan Aharon. (2015) "The Internet of Things: Mapping the Value Beyond the Hype." McKinsey Global Institute, June 2015.
- Marco Iansiti and Karin Lakhani, (2014) "Digital ubiquity: How connections, sensors, and data, are revolutionizing business". *Harvard Business Review*, 2014.
- Marinelli, E. E. (2009). "Hyrax: cloud computing on mobile devices using MapReduce" (No. CMU-CS-09-164). Carnegie-mellon univ Pittsburgh PA School of computer science.
- Mei, L., Chan, W. K., & Tse, T. H. (2008, December). "A tale of clouds: paradigm comparisons and some thoughts on research issues". In *Asia-Pacific Services Computing Conference, 2008. APSCC'08. IEEE* (pp. 464-469). Ieee.
- Michael Porter and James HeppelMann (2014) "How smart connected products are transforming competition" *Harvard Business Review*, 2014.
- Miluzzo, E., Cáceres, R., & Chen, Y. F. (2012, June). "Vision: mClouds-computing on clouds of mobile devices". In *Proceedings of the third ACM workshop on Mobile cloud computing and services* (pp. 9-14). ACM.
- Moreno, E. D., & de Oliveira, J. I. F. (2009). "Architectural impact of the SVG-based graphical components in web applications". *Computer Standards & Interfaces*, 31(6), 1150-1157.
- Mrissa, M., Medini, L., Jamont, J.-P, Le Sommer, N., and Laplace, J. (2015). "An avatar architecture for the web of things". *Internet Computing, IEEE*, 19(2):30–38, Mar 2015.s
- Mtibaa, A., Harras, K. A., & Fahim, A. (2013, December). "Towards computational offloading in mobile device clouds". In *Cloud Computing Technology and Science (CloudCom), 2013 IEEE 5th International Conference on* (Vol. 1, pp. 331-338). IEEE.

Nysveen, H., Pedersen, P. E., & Thorbjørnsen, H. (2005). "Intentions to use mobile services: Antecedents and cross-service comparisons". *Journal of the academy of marketing science*, 33(3), 330-346.

Ou, S., Yang, K., Liotta, A., & Hu, L. (2007, June). "Performance analysis of offloading systems in mobile wireless environments". In *2007 IEEE International Conference on Communications* (pp. 1821-1826). IEEE.

Ould, M. A., & Unwin, C. (1986). "Testing in software development". Cambridge University Press.

Oviatt, S., Coulston, R., & Lunsford, R. (2004, October). "When do we interact multimodally? Cognitive load and multimodal communication patterns". In *Proceedings of the 6th international conference on Multimodal interfaces* (pp. 129-136). ACM.

Pal, S. (2016). "Extending cloud-based applications in challenged environments with mobile opportunistic networks" (Doctoral dissertation, University of St Andrews).

Papakos, P., Capra, L., & Rosenblum, D. S. (2010, November). "Volare: context-aware adaptive cloud service discovery for mobile systems". In *Proceedings of the 9th International Workshop on Adaptive and Reflective Middleware* (pp. 32-38). ACM.

Paredes, H., Fonseca, B., & Barroso, J. (2014, June). "Developing Iconographic Driven Applications for Nonverbal Communication: A Roadside Assistance App for the Deaf". In *International Conference on Universal Access in Human-Computer Interaction* (pp. 762-771). Springer International Publishing.

Paredes, H., Fonseca, B., Cabo, M., Pereira, T., & Fernandes, F. (2014). "SOSPhone: a mobile application for emergency calls". *Universal Access in the Information Society*, 13(3), 277-290.

PCs, U. M. (2008). "Emerging technologies mobile-computing trends: Lighter, faster, smarter". *About Language Learning & Technology*, 3.

Pedersen, M. V., & Fitzek, F. H. (2012). "Mobile clouds: the new content distribution platform". *Proceedings of the IEEE*, 100(Special Centennial Issue), 1400-1403.

Peña-López, I. (2015). *Global Internet Report 2015*. "Mobile evolution and development of the Internet".

- Pradhan, D. (2011): 'Cross-platform Mobile and Tablet Application', Helsinki Metropolia University of Applied Sciences, Bachelor of Engineering Information Technology, Bachelor's Thesis, 20 November 2011.
- Rego, P. A., Costa, P. B., Coutinho, E. F., Rocha, L. S., Trinta, F. A., & de Souza, J. N. (2016). "Performing computation offloading on multiple platforms". *Computer Communications*.
- Reid, D., & Reid, F. (2004). "Insights into the social and psychological effects of SMS text messaging". URL (consulted 17 July 2006): <http://www.160characters.org/documents/SocialEffectsOfTextMessaging.pdf>.
- Reiter, A., & Zefferer, T. (2016, March). "Flexible and Secure Resource Sharing for Mobile Augmentation Systems". In 2016 4th IEEE International Conference on Mobile Cloud Computing, Services, and Engineering (MobileCloud) (pp. 31-40). IEEE.
- Richardson, J., & Lenarcic, J. (2008). "Text messaging as a catalyst for mobile student administration: the trigger experience". *International Journal of Emerging Technologies and Society*, 6(2), 140-155.
- Roman, R., Lopez, J., & Mambo, M. (2016). "Mobile Edge Computing, Fog et al.: A Survey and Analysis of Security Threats and Challenges". arXiv preprint arXiv:1602.00484.
- Rose, K., Eldridge, S., & Chapin, L. (2015). "The internet of things: An overview". *The Internet Society (ISOC)*, 1-50.
- Samimi, F. A., McKinley, P. K., & Sadjadi, S. M. (2006). "Mobile service clouds: a self-managing infrastructure for autonomic mobile computing services". In *Self-Managed Networks, Systems, and Services* (pp. 130-141). Springer Berlin Heidelberg.
- Sanaei, Z., Abolfazli, S., Gani, A., & Buyya, R. (2014). "Heterogeneity in mobile cloud computing: taxonomy and open challenges". *IEEE Communications Surveys & Tutorials*, 16(1), 369-392.
- Satoh, I. (2005, July). "Dynamic deployment of pervasive services". In *ICPS'05. Proceedings. International Conference on Pervasive Services, 2005.* (pp. 302-311). IEEE.

Satyanarayanan, M. (1996, May). "Fundamental challenges in mobile computing". In Proceedings of the fifteenth annual ACM symposium on Principles of distributed computing (pp. 1-7). ACM.

Satyanarayanan, M. (2005), "Avoiding Dead Batteries," in IEEE Pervasive Computing, vol. 4, no. 1, pp. 2-3, Jan.-March 2005. doi: 10.1109/MPRV.2005.5

Satyanarayanan, M., Bahl, P., Caceres, R., & Davies, N. (2009). "The case for vm-based cloudlets in mobile computing". IEEE pervasive Computing, 8(4), 14-23.

Satyanarayanan, M. (2015). "A brief history of cloud offload: A personal journey from odyssey through cyber foraging to cloudlets". GetMobile: Mobile Computing and Communications, 18(4), 19-23.

Shen, J., Yan, S., & Hua, X. S. (2010, October). "The e-recall environment for cloud based mobile rich media data management". In Proceedings of the 2010 ACM multimedia workshop on Mobile cloud media computing (pp. 31-34). ACM.

Shi, C., Habak, K., Pandurangan, P., Ammar, M., Naik, M., & Zegura, E. (2014, August). "Cosmos: computation offloading as a service for mobile devices". In Proceedings of the 15th ACM international symposium on Mobile ad hoc networking and computing (pp. 287-296). ACM.

Shih, Y. E. (2007). "Setting the new standard with mobile computing in online learning". The International Review of Research in Open and Distributed Learning, 8(2).

Siegele, L. (2008). "Let it rise: A special report on corporate IT". Economist Newspaper.

Skierkowski, D., & Wood, R. M. (2012). "To text or not to text? The importance of text messaging among college-aged youth". Computers in Human Behavior, 28(2), 744-756.

Singh, A., & Madan, N. (2016). "A Region Based Offloading Mechanism in Mobile Cloud Computing Environment". Singh Arshdeep, Madan Neena, International Journal of Advance researchs, Ideas and Innovations in Technology. ISSN: 2454-132X (Volume2, Issue3)

Spanoudakis, N. and Moraitis. P. (2015) "Engineering ambient intelligence systems using agent technology". Intelligent Systems, IEEE, 30(3):60-67, May 2015.

Stüber, G. L. (2011). "Principles of mobile communication". Springer Science & Business Media.

Takahashi, N., Tanaka, H., & Kawamura, R. (2015, March). "Analysis of process assignment in multi-tier mobile cloud computing and application to Edge Accelerated Web Browsing". In *Mobile Cloud Computing, Services, and Engineering (MobileCloud)*, 2015 3rd IEEE International Conference on (pp. 233-234). IEEE.

Tsai, W. T., Sun, X., & Balasooriya, J. (2010, April). "Service-oriented cloud computing architecture". In *Information Technology: New Generations (ITNG)*, 2010 Seventh International Conference on (pp. 684-689). IEEE.

Vaquero, L. M., & Rodero-Merino, L. (2014). "Finding your way in the fog: Towards a comprehensive definition of fog computing". *ACM SIGCOMM Computer Communication Review*, 44(5), 27-32.

Verbelen, T., Simoens, P., De Turck, F., & Dhoedt, B. (2012, June). "Cloudlets: bringing the cloud to the mobile user". In *Proceedings of the third ACM workshop on Mobile cloud computing and services* (pp. 29-36). ACM.

Vogels, W. A. (2008, October). "Head in the Clouds—The Power of Infrastructure as a Service". In *First workshop on Cloud Computing and in Applications (CCA'08)(October 2008)* (Vol. 5).

Von Alan, R. H., March, S. T., Park, J., & Ram, S. (2004). "Design science in information systems research". *MIS quarterly*, 28(1), 75-105.

Wang, Y., Lin, J., Annavaram, M., Jacobson, Q. A., Hong, J., Krishnamachari, B., & Sadeh, N. (2009, June). "A framework of energy efficient mobile sensing for automatic user state recognition". In *Proceedings of the 7th international conference on Mobile systems, applications, and services* (pp. 179-192). ACM.

Wang, Y., Chen, R., & Wang, D. C. (2015). "A survey of mobile cloud computing applications: perspectives and challenges. *Wireless Personal Communications*", 80(4), 1607-1623.

Whittaker, J. A. (2000). "What is software testing? And why is it so hard?". *IEEE software*, 17(1), 70-79.

Zhang, L., Ding, X., Wan, Z., Gu, M., & Li, X. Y. (2010, June). "WiFace: a secure geosocial networking system using WiFi-based multi-hop MANET". In Proceedings of the 1st ACM Workshop on Mobile Cloud Computing & Services: Social Networks and Beyond (p. 3). ACM.

Appendix 1 – Questionnaire

12/08/2016

The behavior of mobile applications and services in ubiquitous environment

The behavior of mobile applications and services in ubiquitous environment

Questionnaire prepared under the Course Project Bachelor of Information Technology and Communication of the academic year 2015/2016.

Performed by:
-Andreia Sousa
-Fernando Cerqueira

Advisors:
-Prof. Hugo Paredes

Co-Advisors:
-Prof. Diogo Azevedo.

* Required

Acceptance Letter

Researchers: Andreia Sousa e Fernando Cerqueira, Bachelor in Information and Communication Technologies (UTAD)

Purpose: The purpose of the questionnaire is to conduct a study on the behavior of mobile applications and the availability of services in ubiquitous environments. The study aims to analyze the main needs that mobile users face in the use of applications in different environments.

Participants: Participants are volunteers of different age groups.

Confidentiality: The information collected is only for the present study, without use of personal data.

Procedure: The questionnaire aims to collect information on the behavior of mobile users in different environments. In this regard, participants should follow a set of questions drawn up with the objective of analyzing the difficulties encountered in the use of applications in roaming. Filling out the questionnaire does not imply the prior use of any application, focusing on the experience of users. The completion of the questionnaire has an approximate duration of 10 minutes.

Benefits: Participants may contribute to a better efficiency of the involved study, primarily in the collection of problems that occur when they are moving abroad, which is directly associated and will allow the grounds of the need to ensure the availability of services even with limited mobile communications.

Risks: The procedure does not cause any injuries or risks.

Compensation: No compensation in completing this questionnaire, but there are benefits for the scientific community with regard to ensuring the availability of mobile services.

Disclaimer: Participation in this study is completely voluntary. Volunteers are not linked to any obligation and they may complete the questionnaire as they wish.

Questions or later doubts: For questions about any of the topics described above, you can send an email to the investigator (s): fernando.cerqueira@outlook.pt ; Andreia_Sousa_04@hotmail.com

Declaration

I was informed about the goals of this project, as well as the procedures involved in this experience. I reserve the right to leave the experiment at any stage thereof and not with my consent for their use, all

information previously obtained is destroyed or my identity withdrawal.

1. Accept the following terms? *

Mark only one oval.

- Yes
- No Stop filling out this form.

Travels

2. How do you usually plan your travels ? *

Check all that apply.

- Travel Websites
- Family / Friends
- Travel Websites
- Other:

3. How often do you usually travel abroad? *

Mark only one oval.

- Once a year
- Once every 2 years
- More than Once a year
- Other

4. How would you describe the use of mobile devices on a trip? *

Mark only one oval.

	1	2	3	4	
Bad	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Very Good

Cell Phones

5. Rate the importance of cell phone today? *

Mark only one oval.

	1	2	3	4	5	
	<input type="radio"/>	Very important				

6. What kind of phone you have? *

Mark only one oval.

- Smartphone
- Cell phone with physical keyboard
- Other

7. Do you use your mobile applications on a daily basis? **Mark only one oval.*

- Yes
 No

8. Do you use to download applications on your phone ? **Mark only one oval.*

- Yes
 No

9. How do you download them ? **Mark only one oval.*

- Play Store
 App Store
 Windows Store

10. You think important powers access both applications, when no Internet? **Mark only one oval.*

- Yes
 No

11. Spend the most time on the Internet? **Check all that apply.*

- Social Networks
 Mobile Messaging Apps (WhatsApp, Viber...)
 Researches
 Scope professional
 Other

12. How often use the internet abroad? **Mark only one oval.*

- Never
 1 to 2 hours/day
 2 to 6 hours/day
 More than 6 hours/day

13. When you are abroad and you usually communicate with friends/colleagues/family? **Check all that apply.*

- Skype
 Mobile Messaging Apps (WhatsApp, Viber...)
 Calls from the national operator
 Calls from the local carrier
 Other

14. As you usually access the Internet abroad? **Check all that apply.*

- Adhere to local operators to get access
- Hotspots
- Cafes/Restaurants/Locais Públicos com acesso WI-FI
- Mobile data through my national operator
- Other

15. Are you aware of the tariffs for the use of mobile data abroad by your national carrier? **Mark only one oval.*

- Yes
- No

16. Do you think that a reduction in the cost of mobile data abroad by your national operator would be an option to consider when you are traveling? **Mark only one oval.*

- Yes
- No

Personal Data

17. How old are you? *

.....

18. What is your education? **Mark only one oval.*

- 4th year
- 6th year
- 9th year
- 12th year
- Graduation
- Master's degree
- PhD

19. What is your nationality? *

Powered by
 Google Forms

Appendix 2 – XML Services

```
<?xml version="1.0" encoding="UTF-8"?>
<dcc_services>
  <servicename name="Service1">
    <details>
      <name>Twitter</name>
      <url>http://twitter.com</url>
    </details>
    <functions>
      <func>
        <name>Search</name>
        <input>
          <param>
            <name>searchString</name>
            <type>string</type>
          </param>
        </input>
        <output>
          <param>
            <name>tweetList</name>
            <type>array[string]</type>
          </param>
        </output>
      </func>
      <func>
        <name>Post</name>
        <input>
          <param>
            <name>searchString</name>
```

```

    <type>string</type>
  </param>
</input>
<output>
  <param>
    <name>tweetList</name>
    <type>array[string]</type>
  </param>
</output>
</func>
</functions>
</servicename>
<servicename name="Service2">
  <details>
    <name>Youtube</name>
    <url>http://youtube.com</url>
  </details>
  <functions>
    <func>
      <name>Search</name>
      <input>
        <param>
          <name>searchString</name>
          <type>string</type>
        </param>
      </input>
      <output>
        <param>
          <name>tweetList</name>

```

```
<type>array[string]</type>  
</param>  
</output>  
</func>  
</functions>  
</servicename>  
</dcc_services>
```


Appendix 3 – XML Priorities

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<dcc_priorities>
```

```
<priorityList>
```

```
<priority name="Wifi">
```

```
<details>
```

```
<name>Wifi</name>
```

```
<info>Data</info>
```

```
</details>
```

```
</priority>
```

```
<priority name="4G">
```

```
<details>
```

```
<name>4G</name>
```

```
<info>Data</info>
```

```
</details>
```

```
</priority>
```

```
<priority name="3G">
```

```
<details>
```

```
<name>3G</name>
```

```
<info>Data</info>
```

```
</details>
```

```
</priority>
```

```
<priority name="GPRS">
```

```
<details>
```

```
<name>GPRS</name>
```

```
<info>SMS</info>
```

```
</details>
```

```
</priority>
```

```
</priorityList>
```

```
<infoTypeTemplateList>
  <infoTypeTemplate>
    <infoName>Data</infoName>
    <template>servicename;func;param[;param]</template>
  </infoTypeTemplate>
  <infoTypeTemplate>
    <infoName>SMS</infoName>
    <template>servicename;func;param[;param]</template>
  </infoTypeTemplate>
</infoTypeTemplateList>
</dcc_priorities>
```

Appendix 4 – XML Rules

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<dcc_rules>
```

```
<rule name="Rule1">
```

```
<details>
```

```
<service>Twitter</service>
```

```
<mandatory>Yes</mandatory>
```

```
</details>
```

```
<functions>
```

```
<func>
```

```
<name>Search</name>
```

```
<connections>
```

```
<connection>Wifi</connection>
```

```
<connection>4G</connection>
```

```
<connection>GPRS</connection>
```

```
</connections>
```

```
</func>
```

```
<func>
```

```
<name>Post</name>
```

```
<connections>
```

```
<connection>Wifi</connection>
```

```
<connection>4G</connection>
```

```
<connection>GPRS</connection>
```

```
</connections>
```

```
</func>
```

```
</functions>
```

```
</rule>
```

```
<rule name="Rule2">
```

```
<details>
```

```
<service>Youtube</service>
<mandatory>No</mandatory>
</details>
<functions>
  <func>
    <name>Search</name>
    <connections>
      <connection>Wifi</connection>
      <connection>4G</connection>
    </connections>
  </func>
</functions>
</rule>
</dcc_rules>
```

Appendix 5 – XSD Rules

```
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
```

```
  <xs:element name="dcc_rules">
```

```
    <xs:complexType>
```

```
      <xs:sequence>
```

```
        <xs:element name="rule" maxOccurs="unbounded" minOccurs="0">
```

```
          <xs:complexType>
```

```
            <xs:sequence>
```

```
              <xs:element name="details">
```

```
                <xs:complexType>
```

```
                  <xs:sequence>
```

```
                    <xs:element type="xs:string" name="service"/>
```

```
                    <xs:element type="xs:string" name="mandatory"/>
```

```
                  </xs:sequence>
```

```
                </xs:complexType>
```

```
              </xs:element>
```

```
            <xs:element name="functions">
```

```
              <xs:complexType>
```

```
                <xs:sequence>
```

```
                  <xs:element name="func" maxOccurs="unbounded" minOccurs="0">
```

```
                    <xs:complexType>
```

```
                      <xs:sequence>
```

```
                        <xs:element type="xs:string" name="name"/>
```

```
                        <xs:element name="connections">
```

```
                          <xs:complexType>
```

```
                            <xs:sequence>
```

```
                              <xs:element type="xs:string" name="connection"
maxOccurs="unbounded" minOccurs="0"/>
```

```
        </xs:sequence>
    </xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
<xs:attribute type="xs:string" name="name" use="optional"/>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:schema>
```

Appendix 6 – XSD Priorities

```
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
```

```
  <xs:element name="dcc_priorities">
```

```
    <xs:complexType>
```

```
      <xs:sequence>
```

```
        <xs:element name="priorityList">
```

```
          <xs:complexType>
```

```
            <xs:sequence>
```

```
              <xs:element name="priority" maxOccurs="unbounded" minOccurs="0">
```

```
                <xs:complexType>
```

```
                  <xs:sequence>
```

```
                    <xs:element name="details">
```

```
                      <xs:complexType>
```

```
                        <xs:sequence>
```

```
                          <xs:element type="xs:string" name="name"/>
```

```
                          <xs:element type="xs:string" name="info"/>
```

```
                        </xs:sequence>
```

```
                      </xs:complexType>
```

```
                    </xs:element>
```

```
                  </xs:sequence>
```

```
                <xs:attribute type="xs:string" name="name" use="optional"/>
```

```
              </xs:complexType>
```

```
            </xs:element>
```

```
          </xs:sequence>
```

```
        </xs:complexType>
```

```
      </xs:element>
```

```
    <xs:element name="infoTypeTemplateList">
```

```
      <xs:complexType>
```


Appendix 7 – XSD Services

```
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
```

```
  <xs:element name="dcc_services">
```

```
    <xs:complexType>
```

```
      <xs:sequence>
```

```
        <xs:element name="servicename" maxOccurs="unbounded" minOccurs="0">
```

```
          <xs:complexType>
```

```
            <xs:sequence>
```

```
              <xs:element name="details">
```

```
                <xs:complexType>
```

```
                  <xs:sequence>
```

```
                    <xs:element type="xs:string" name="name"/>
```

```
                    <xs:element type="xs:anyURI" name="url"/>
```

```
                  </xs:sequence>
```

```
                </xs:complexType>
```

```
              </xs:element>
```

```
            <xs:element name="functions">
```

```
              <xs:complexType>
```

```
                <xs:sequence>
```

```
                  <xs:element name="func" maxOccurs="unbounded" minOccurs="0">
```

```
                    <xs:complexType>
```

```
                      <xs:sequence>
```

```
                        <xs:element type="xs:string" name="name"/>
```

```
                        <xs:element name="input">
```

```
                          <xs:complexType>
```

```
                            <xs:sequence>
```

```
                              <xs:element name="param">
```

```
                                <xs:complexType>
```

```

    <xs:sequence>
      <xs:element type="xs:string" name="name"/>
      <xs:element type="xs:string" name="type"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="output">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="param">
        <xs:complexType>
          <xs:sequence>
            <xs:element type="xs:string" name="name"/>
            <xs:element type="xs:string" name="type"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>

```

```
</xs:element>
</xs:sequence>
<xs:attribute type="xs:string" name="name" use="optional"/>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:schema>
```