Universidade de Aveiro
Departamento de Eletrónica, Telecomunicações e Informática
2014

Mário Jorge Marques da Silva

Dispositivos móveis na recolha eletrónica de dados em estudos clínicos

Mobile devices for electronic data capture in clinical studies

Universidade de Aveiro
2014

Departamento de Eletrónica,
Telecomunicações e Informática

Mário Jorge Marques da Silva

Dispositivos móveis na recolha eletrónica de dados em estudos clínicos

Mobile devices for electronic data capture in clinical studies

Dissertação apresentada à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Engenharia de Computadores e Telemática, realizada sob a orientação científica do Doutor Ilídio Oliveira, Professor auxiliar do Departamento de Eletrónica, Telecomunicações e Informática da Universidade de Aveiro.

Dedico este trabalho aos meus pais e amigos.

o júri / the jury

presidente / president

Professor Doutor Augusto Marques Ferreira da Silva

Professor auxiliar do Departamento de Eletrónica, Telecomunicações e Informática da Universidade de Aveiro

vogais / examiners committee

Professora Doutora Alexandra Isabel Cardador de Queirós

Professora Coordenadora s/ Agregação da Escola Superior de Saúde da Universidade de Aveiro

Professor Doutor Ilídio Castro Oliveira

Professor auxiliar do Departamento de Eletrónica, Telecomunicações e Informática da Universidade de Aveiro

agradecimentos /
acknowledgements

Em primeiro lugar gostaria de agradecer ao meu orientador, Professor Doutor Ilídio Oliveira, por todo o apoio, técnico e moral, que me levou a conseguir concluir este trabalho. Pela sua disponibilidade, simpatia e excelente carácter humano. Ao meu co-orientador, Ivo Pinheiro que me auxiliou e disponibilizou material técnico para a realização deste trabalho.

Um obrigado à iUZ por toda a disponibilidade fornecida e por ter cedido um dispositivo médico para testes. Um obrigado a todos os colegas com quem tive a oportunidade de trabalhar na iUZ pela sua sempre boa disposição e ajuda, principalmente ao Bruno Garcia e ao Gil Millasseau pela sua ajuda na parte gráfica da aplicação. Um obrigado a todos os colegas de laboratório pela sua sempre boa disposição e ajuda. Por todos os concelhos e vivências partilhados que me ajudaram a crescer ao longo deste ano.

Agradeço toda a ajuda dos meus colegas e companheiros ao longo destes anos, principalmente ao Eng. Tiago Magalhães e ao Eng. João Pinto, que me ajudaram e aconselharam nos momentos mais difíceis.

Também quero deixar um agradecimento muito especial aos meus pais por terem sempre apoiado neste grande desafio. Sem eles toda a minha vida académica não teria sido possível.

Palavras Chave

Estudos clínicos, Recolha electrónica de dados, Dispositivos móveis, Android, ISO/IEEE 11073.

Resumo

Os dispositivos móveis, incluido os comuns *smartphones* e *tablets*, estão a ser cada vez mais usados em cenários de *mHealth*, em que o dispositivo é usado para a recolha de dados médicos diretamente ou atuando como um agregador para sensores médicos. Tais aplicações permitem captura e persistência eletrónica dos dados, prevenindo problemas com a inserção manual.

A disponibilidade de *smartphones* e *tablets*, por um lado, e sensores vestíveis/dispositivos médicos, por outro, cria uma oportunidade para usar a captura de dados de saúde com dispositivos móveis também em aplicações de estudos clínicos.

Nesta dissertação, propomos uma applicação móvel para a participação em estudos clínicos, desenvolvida em Android, incluindo a captura eletrónica de dados em contextos ambulatórios. Além do suporte comum para preenchimento de questionários, a aplicação utiliza o protocolo ISO/IEEE 11073 para comunicar com dispositivos médicos compatíveis.

O trabalho foi concebido para se integrar com a plataforma de estudos clínicos uEDC (desenvolvida pela iUZ Technologies). O uso experimental dos sistemas mostra que o *front-end* móvel consegue com sucesso suportar diferentes dispositivos e protocolos de estudo, totalmente integrados com o *backend* uEDC.

Keywords Clinical studies, Electronic data capture, Mobile devices, Android, ISO/IEEE 11073.

Abstract

Mobile devices, including common smartphones and tablets, are being increasingly used for mHealth scenarios, in which the device is used to capture health values directly or acting as a hub for health sensors. Such applications allow a machine-to-machine capture and persistence of data, avoiding problems with manual data entry.

The availability of smartphones and tablets, on one side, and wearable sensors/medical devices, on the other, creates an opportunity to use mobile data capture of health values also in clinical studies applications.

In this dissertation, we propose a mobile front-end for clinical studies participants, developed in Android, including electronic data capture in ambulatory contexts. Besides the common questionnaire filling support, the front-end relies on the ISO/IEEE 11073 standard to directly obtain values from compliant medical devices.

The work has been designed to integrate with the existing clinical studies platform uEDC (developed by iUZ Technologies). Early usage of the system shows that the mobile front-end can successfully support different devices and study protocols, fully integrated with the uEDC backend.

Contents

List of Figures

List of Tables

List of Abbreviations and Acronyms

AHD	Application Hosting Device
API	Application Programming Interface
BLE	Bluetooth Low Energy
CRF	Case Report Form
CROs	Clinical Research Organizations
CSV	Comma-Separated Values
DETI	Departamento de Eletrónica, Telecomunicações e Informática da Universidade de Aveiro
DHACA	Digital Health and Care Alliance
ECRF	Electronic Case Report Form
EDC	Electronic Data Capture
FDA	U.S. Food and Drug Administration
FLOS	Free/Libre and Open Source
HDP	Bluetooth Health Device Profile
ICMJE	World Health Organization International Committee of Medical Journal Editors
ICS	Ice Cream Sandwich
IEEE	Institute of Electrical and Electronics Engineers
ISO	International Organization for Standardization
IP	Internet Protocol
IPC	Inter-Process Communication
iUZ	iUZ Technologies
JSON	JavaScript Object Notation
MDER	Medical Device Encoding Rules
MDS	Medical Device System
DIM	Domain Information Model
mHealth	Mobile Health
NDK	Native Development Kit
NLM	U.S. National Library of Medicine
OS	Operating System
OSGi	Open Service Gateway initiative
OSI	Open System Interconnect
PAN	Bluetooth personal area network
TCP	Transmission Control Protocol
PHDC	Personal Health Device Class
PM-Store	Persistent Metric Store
QRcode	Quick Response Code
REST	Representational state transfer
SMS	Short Messaging Service
UI	User Interface
UML	Unified Modeling Language
USB	Universal Serial Bus
WHO	World Health Organization
XML	Extensible Markup Language

CHAPTER 1

Introduction

The U.S. National Library of Medicine (NLM) defines that every clinical trial has a protocol, or action plan, for conducting the trial. The plan describes what will be done in the study, how it will be conducted, and why each part of the study is necessary. Each study or protocol has its own rules, e.g who can take part, if some volunteers with some disease are needed or if the target is a healthy person [1]. The industry use the Case Report Form (CRF) for saving the medical data about the patient but this CRF normally relies heavily on paper documents. Numerous reports indicate that one can improve the efficiency by replacing traditionally paper CRF with Electronic Case Report Form (ECRF) [2] [3] [4]. With this ECRF arises a new possibility, the use of Electronic Data Capture (EDC) Systems. These EDC systems help on the collection of clinical data and will replace the paper documents by electronic records.

Our world is becoming increasingly more "mobile". In the last few years the market penetration of mobile devices has risen exponentially, which has led the medical field to begin using them in order to capture health values by (integrating) health data sensors. Such applications allow a machine-to-machine capture and persistence of data, avoiding problems with manual data entry. The availability of smartphones and tablets, on one side, and wearable sensors/medical devices, on the other, leads to an opportunity to use mobile data capture also in clinical studies applications, as EDC Systems.

In this dissertation, we propose a mobile front-end for clinical studies participants, developed in Android, including electronic data capture in ambulatory contexts. Besides the common questionnaire filling support, the front-end relies on the International Organization for Standardization (ISO) /Institute of Electrical and Electronics Engineers (IEEE) 11073

standard to directly obtain values from compliant medical devices. This dissertation was hosted by iUZ Technologies (iUZ) , which proposed the topic, according with the collaboration model with the companies in practice by Departamento de Eletrónica, Telecomunicações e Informática da Universidade de Aveiro (DETI) .

1.1 OBJECTIVES

Motivated by the availability of smart mobile devices for health applications, our main objective is to create a mobile front-end that integrates into existing EDC Systems, in particular the platform uEDC (developed by iUZ), with smartphones that use Android Operating System (OS) . With this solution, the patient can actively participate in a clinical trial in an ambulatory context. Mobile technologies have the potential to improve every step of the clinical trial [5].

Our objective is to focus on data capture using mobile devices, so the patient can stay home and participate on the study without losing time on waiting queues on hospitals or requiring proximity to an investigator's site. This solution should also retrieve data from medical devices, e.g retrieve medical data from a blood pressure monitor, using the ISO/IEEE 11073 standard. Patients can use the data to manage their own health, including sharing the data with their doctor or using the data into spreadsheets to see the evolution. This solution opens new possibilities on clinical trials and on the management of our health.

1.2 STRUCTURE

This dissertation is organized in seven chapters:

Chapter 1 presents the motivation and objectives of the present work that lead to uEDC Mobile.

Chapter 2 describes the related state of art containing an overview of clinical trials, how they work and how the EDC Systems are used on clinical trials. This chapter describes the use of mobile devices on health, the Mobile Health (mHealth) , and the current systems used on clinical trials. Still in this chapter, we survey existing solutions for the communication between the medical devices and mobile devices.

Chapter 3 describes the requirements and use cases for the uEDC Mobile.

Chapter 4 presents the system architecture and the changes needed in uEDC to integrate

with mobile devices. This chapter present the rationales and the structure used in the present work.

Chapter 5 describes the implementation of uEDC Mobile, which solutions were used and which decisions were made. In addition, we present how each requirement was implemented, in high level, and the features present on uEDC Mobile. The chapter presents a complementary application that does the communication between the mobile devices and the medical sensors as a proof of concept.

Chapter 6 presents the validation activities made on uEDC Mobile and the results. In addition, we present tests made on the communication between the mobile device and medical sensors.

Chapter 7 discusses the accomplishments of uEDC Mobile and future work to give continuity to this project.

CHAPTER 2

State of Art

With the evolution of electronic systems, arises the possibility of changing the paper-based clinical trials to systems that collect medical information, the EDC systems. The EDC systems help to reduce costs and time on clinical trials and clinical studies, improving the data collection and management. To improve the clinical trials, we need to know how they work on the modern systems.

2.1 clinical trials & clinical studies concepts

The World Health Organization (WHO) defines clinical trials as any research study that prospectively assigns human participants or groups of humans to one or more health-related interventions to evaluate the effects on health outcomes [6]. More specifically, the description of a clinical trial follows a pre-defined protocol and contains various information about the study, e.g the target, the characteristics of patients that can participate, details about the institutions that accomplished the study, etc. Overall, the clinical trials are mandatory protocols that describe medical research on humans and among the most valuable sources of medical practice evidence [7].

Currently, there are several clinical trial registries all over the world. ClinicalTrials.gov [8] is a registry developed and maintained by the NLM and the U.S. Food and Drug Administration (FDA) . It adopts the standard proposed by World Health Organization International Committee of Medical Journal Editors (ICMJE) as the data format, to promote the sharing of clinical trial data and the interaction all over the world [9].

CASE REPORT FORM & ELECTRONIC CASE REPORT FORM

A CRF is a paper-based questionnaire used in clinical trial research. The CRF is the tool used by the sponsor of the clinical trial to retrieve data from each participant. But before being sent to the sponsor, this data is usually de-identified by removing all the information that has a direct relation with the patient, e.g the patient's name, medical record number, and is given the patient a unique study/case number [10].

The responsible of the study designs a CRF that accurately represents the protocol of the clinical trial, as well as managing its production, monitoring the data collection and auditing the content of the filled-in CRF. The size of CRF can be short (one hour) or can take periods of weeks or months. Each CRF can have hundreds of pages, which implies a huge waste of resources. The application of ECRF solves most of these problems, i.e. the processing and visualization of data will be faster.

ELECTRONIC DATA CAPTURE

Traditionally, the clinical practice, research studies, and quality control exercises use paper-based case report forms or questionnaires for collecting data. J. Shah et al. [11] indicate that "despite being simple, manual data entry into spreadsheets and subsequent verification is time-consuming, tedious, and prone to data errors". With the explosion of digital information on medicine, the EDC systems have emerged as an alternative to paper-based systems.

An EDC system is normally web-based software that stores patient data collected in clinical trials. In the article [12] it is indicated that normally paper source documents are used to record the data before being saved in an ECRF, but nowadays the data are inserted directly into the system, where the paper documents phase almost disappear.

According to Forte Research Systems [12] the Clinical Research Organizations (CROs) are choosing EDC software as the preferred tool for managing clinical trial data. Also, Forte Research Systems says that EDC systems are now a standard and have been used to carry out both simple and complex trials in all phases of clinical research [12].

2.2 MOBILE DEVICES IN MHEALTH APPLICATIONS

Mobile devices, including common smartphones and tablets, are being increasingly used as measure systems. Figure 2.1 shows some possible application fields of smartphones as

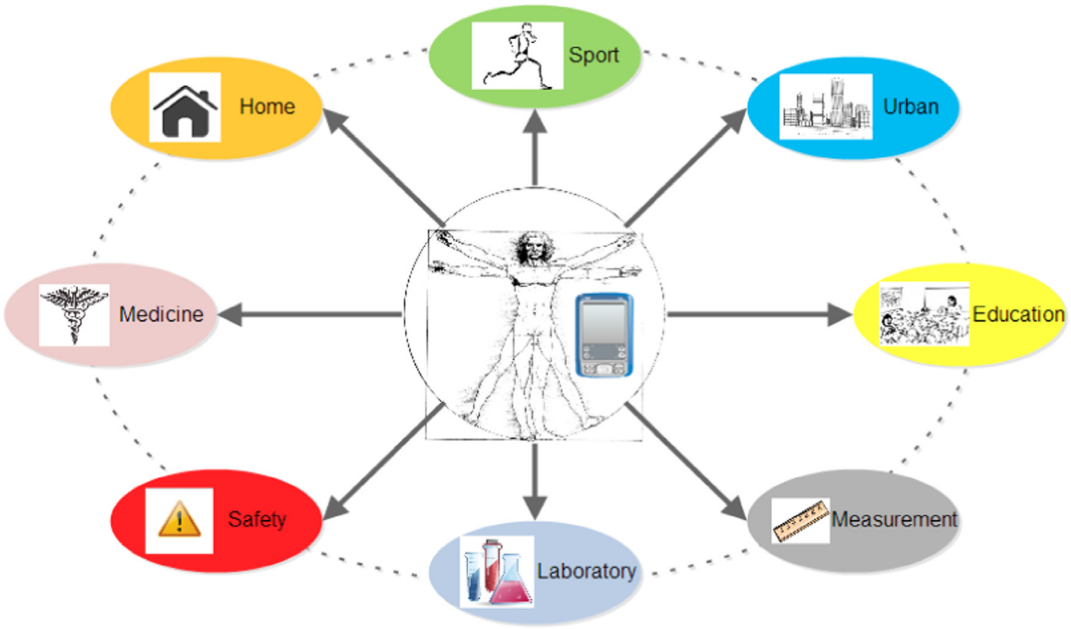


Figure 2.1: Application fields of smartphones as measurement systems [13]

measurement systems. In medicine, exists one field that describe the application of mobile devices on health, the mHealth.

MHEALTH

The WHO define mHealth as "mobile phone technology for health-related purposes". Thomas J. Betjeman et al. [14] identify that "this relatively new, dynamic, and rapidly evolving field includes the development and study of mobile phone applications such as Short Messaging Service (SMS) , voice calling, and wireless data transmission to collect or disseminate health-related information or to direct care". Recently, there has been an explosion in mHealth activities globally [15], indicating a strong investment from industry.

Kevin Patrick [16] uses the figure 2.2 to show the application of the mobile technologies by providers and consumers for "monitoring health status or improving health outcomes, including wireless diagnostic and clinical decision support". It also indicates that the "mHealth technologies supports new methods for collecting biological, behavioral, or environmental data and the outcomes of interventions. These include sensors that monitor phenomena with higher precision, improved sampling frequency, fewer missing data, greater convenience, and in some cases, lower cost than traditional measures". These new technologies, mainly the use of mobile devices as measurement systems, are changing the way that health interventions are conducted, motorization and prevention, improving health outcomes.

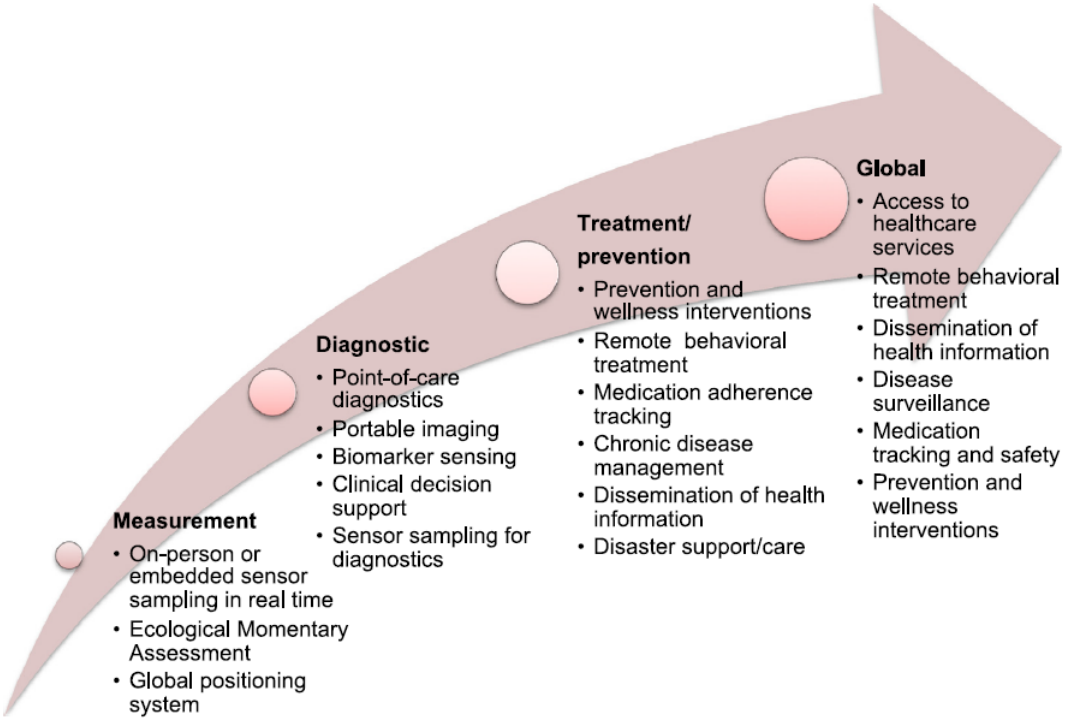


Figure 2.2: Continuum of mHealth tools [16]

Joong-Yeol Park et al. [17] shows that this technology and its estimated market value has grown rapidly. The global mHealth market was estimated to be worth 1.2 billion dollars in 2011 and is expected to increase to 11.8 billion dollars by 2018.

In terms of OS used on mobile devices, the dominating OS is the Android OS. Gartner [18] made a market research about the smartphones sales in the fourth quarter of 2012, in which the Android OS dominates the sales when compared to the iOS, Microsoft and others. It was approximately 70% of Worldwide Smartphone Sales to End Users. Other study [19] made on second quarter of 2013 shows an evolution of the sales for Android OS, approximately 10% more than the fourth quarter of 2012. This numbers demonstrates the dominance of Android OS and, currently if we make an application for Android OS, we will address more users.

2.3 COMMUNICATION BETWEEN MOBILE DEVICES AND MEDICAL SENSORS

The evolution of mobile devices has notably changed the technologies used in the healthcare community, especially in home healthcare environments. With this expansion emerges the need for systems that are capable of retrieving medical data from different sources. These

sources normally have different proprietary formats, leading to lack of interoperability between them. Nowadays, the problem of these new systems is the integration with different biomedical data sources into a single platform [20].

One solution for this problem is an implementation of middleware that allows the communication of multiple devices with single platform. Among the standards related to interoperability between devices and medical systems, A. Fioravanti et al. [20] shows that the healthcare industry is progressively drawing special attention to standard developed by ISO and IEEE. This standard is commonly referred as x73 or ISO/IEEE 11073, where a set of standards is set to address the interoperability between sensors and medical systems.

With the release of the Bluetooth Health Device Profile (HDP) a strong impulse was given to achieving interoperability among wireless devices [20] [21]. The HDP is a standardized specification for Bluetooth communication between medical devices. Normally, it refers to the ISO/IEEE 11073 protocol specification because its the only protocol currently supported and the HDP architecture is clearly aimed towards ISO/IEEE 11073 needs. HDP goes as far as having persistent channels, which goes hand in hand with ISO/IEEE 11073 persistent sessions. Universal Serial Bus (USB) has Personal Health Device Class (PHDC) that have the same objective as HDP: to provide suitable transport channels for ISO/IEEE 11073 [22].

A. Fioravanti et al. [20] shows that the market standardization is facing obstacles from being easily accessible to every device and platform, even if the state of art in this field is quite developed. However this trend is just changing, already existing some alliances that are trying to change this problem, using ISO/IEEE 11073 standard on support [23].

ISO/IEEE 11073

The primary goals of the ISO/IEEE 11073 standards are to “provide real-time plug-and-play interoperability for patient-connected medical devices and facilitate the efficient exchange of vital signs and medical device data, acquired at the point-of-care, in all health care environments" [24].

This is only an overview based on Antidote [22] to understand how the standard works. The standard is huge and all the information is detailed in [25].

The ISO/IEEE 11073 is divided in two device types: Agents and Managers. Typically, the Agents are data producers (the medical sensors) and Managers are the data collectors. The ISO/IEEE 11073 is based upon the idea that agents are low-powered and have few processing resources, so most of the burden of ISO/IEEE 11073 is put on managers.

Communication model

The standard ISO/IEEE 11073 is transport-agnostic and is based on Open System Interconnect (OSI) for networking, meaning that it can be carried by almost any packet-based technology, such as Transmission Control Protocol (TCP) /Internet Protocol (IP) , Bluetooth, USB, etc. The standard ISO/IEEE 11073 fills the following top three layers:

- Presentation, is provided by Medical Device Encoding Rules (MDER) ;
- Session is provided by communication model;
- Application is where useful data finally flows from agent to manager.

The ISO/IEEE 11073 session layer supports session persistence even upon disconnection at transport level, i.e. the transport can be disconnected due to a failure or to save power. The session persistence is optional but most of the devices do not implement the persistence session, because it is simpler if the device terminates the session as soon as transport disconnects.

State Machine

The ISO/IEEE 11073 Session Layer is governed by a state machine, in which the basic states for the manager are:

- Disconnected
- Connected
  - Associating
  - Associated
    - Waiting for configuration
    - Checking configuration
    - Operating
  - Disassociating
  - Unassociated

As the indentation suggests, some states imply others, e.g a device must be associated in order to be operational. The hierarchy is more complex than the illustrated. The associations may persist even upon transport disconnection. A more realistic state machine map can be found in the figure 2.3.

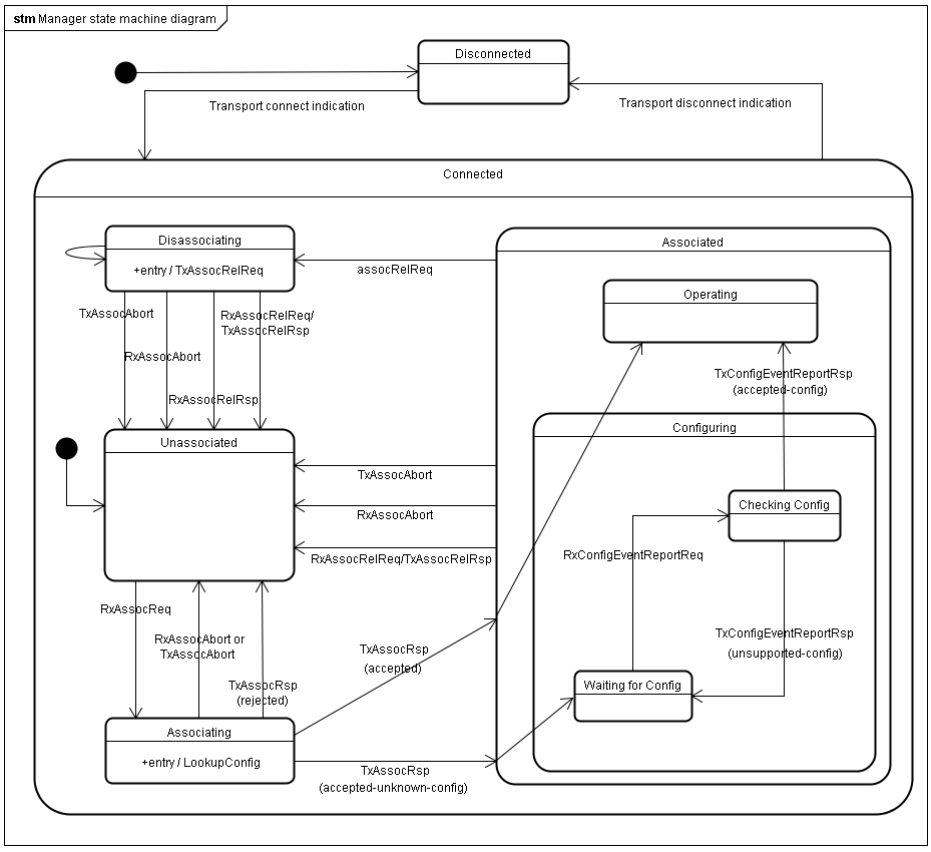


Figure 2.3: Manager State Machine on ISO/IEEE 11073 [22]

Domain Information Model

Domain Information Model (DIM) defines an idealized structure of data inside of each agent device. The DIM is a kind of "object-oriented" structure which means that the "classes" define each DIM. Each class has a set of permissible attributes but depending on class, some attributes are mandatory, others optional, and others forbidden and so on.

A physical agent device has a number of objects that instantiate such "classes" and each object have a number of attributes. Each agent contains exactly one Medical Device System (MDS) object. The MDS has a number of attributes, e.g. System ID, manufacturer information, specialization. To retrieve these MDS attributes, the manager uses a Get operation and in some situations, can be written using Set operations. The manager can discover the MDS attributes in several phases, normally in the phase of configuration and measurement, but there are some crucial MDS attributes that are sent at the association phase.

Configuration and Event reports

After association establishment the agent will send the configuration to the manager. In this configuration is specified the number of objects that exists in MDS and other information, e.g their types, handles, units, etc. When the value of a MDS object is changed (e.g. the Pulse and Oximetry numeric objects of an oximeter are updated by sensor data), the agent sends another Event Report with measurement data to the Manager. It is here that the application finally gets some sensor data.

The event report packet can only be interpreted (decoding the information) by the Manager. This is the reason why the Manager must discover the configuration before entering the operating phase. These mechanisms provide prompt extensibility where the manager does not need to have any preconceptions about agent's MDS, he will learn everything at the configuration phase.

The ISO/IEEE 11073 has two types of configurations: standard and extended. The standard configurations are set on the specialization documents defined by the standard. The Manager may know some standard configurations in advance, so on the association phase he does not need to learn anything because he already knows the standard configurations of the agent.

The extended configuration can have any format, so the Manager must learn it during the configuration phase. Once it learns the configuration, it can save it locally (like a cache), skipping the configuration phase in future associations. The cached configuration is only valid for that device, if another device comes along, even if it has the same manufacturer and model, the manager has to learn the extended configuration again.

Standard configurations have uniform codes across all devices, e.g. 0x0190 is a standard configuration for oximeter, which is defined in ISO/IEEE 11073 documents. Extended configurations have codes above 0x4000 where they cannot be correlated across devices. For example if device A has config-id 0x5000 and other device B that has the config-id 0x5000 too, the manager cannot assume that they share the same configuration.

Specializations

The standard ISO/IEEE 11073 has additional documents that define device specializations, apart of the ISO/IEEE 11073-20601 base document. Every specialization document defines which objects are mandatory in agent's MDS and what they mean.

For example, the document ISO/IEEE 11073-10404 defines the oximeter specialization. It

defines that an oximeter must have at least two Numeric objects: one for oximetry (whose unit is %) and one for pulse (whose unit is beats per minute (bpm)). The manufacturer can add more objects, but he must respect the objects required. This guarantees that an oximeter will always produce those two objects and the manager can rely on them.

Each specialization document specifies one or more standard configurations. These configurations are rather bare-minimum and last-resort configurations. For example, none of oximeter standard configurations has a Persistent Metric Store (PM-Store) , but a good oximeter is expected to store measurements, so to use that it must have a PM-Store and use an extended configuration.

CONTINUA HEALTH ALLIANCE

Signove [22] describes the Continua Health Alliance as a "non-profit association that promotes the standardization of personal health devices, envisioning a market of standard, affordable and readily connectable sensors". In order to accomplish that, the Continua chose the standard ISO/IEEE 11073 as the main protocol stack. The Continua has a "Continua-compliant Manager" (typically a software running on PC) that guarantees compatibility to every Continua-certified device.

But not every Continua device implements ISO/IEEE 11073, there are some technologies that do not support the standard. For example, the Bluetooth Low Energy (BLE) devices have very low power requirements, in which the standard ISO/IEEE 11073 is inappropriate. This can be resolved with a transcoding scheme, which translate the messages from native device protocol to ISO/IEEE 11073 messages. This solution is a kind of proxy that does the role of an agent. Actually, the Continua has a transcoding standard for BLE devices where it is possible to apply some intermediate software that emulates the role of an agent, sparing the Manager software from non-11073 protocols [22].

IMPLEMENTED SOLUTIONS FOR ISO/IEEE 11073

Some implementations exist on the market for this standard. The Digital Health and Care Alliance (DHACA) [26] has a list with the most popular implementations [27].

OpenHealthTools Agent/Manager

The Stepstone project has the goal to provide "a stepping stone for those looking to build health and wellness solutions using embedded technology, service oriented architecture,

and open standards". The Stepstone project "leverages the Open Service Gateway initiative (OSGi) standard for Java to enable dynamic interaction with medical devices and their integration with backend systems and processes". The project is available in [28].

s3group solution

The S3Group has "an Embedded Agent Stack that enables turning a personal health device, communicating over a proprietary protocol, into a Continua Bluetooth personal area network (PAN) Device able to communicate with any Continua-compliant Application Hosting Device (AHD) ", which is available in [29].

Openhealth Project

The OpenHealth Project is a "Free/Libre and Open Source (FLOS) implementation of a multiplatform manager device written in Java according to the ISO/IEEE 11073-20601. Manager application is designed to work in DalvikVM over android platform", which is available in [30].

Signove Antidote IEEE Agent/Manager

The Signove Antidote is a set of libraries that implement ISO/IEEE 11073-20601 stack for medical devices. Main goals are portability, simple usage and simple integration with applications, which is available in [31] [32].

OXPlib

The OXPlib project is "an implementation of the ISO/IEEE 11073-20601 written in C++ and Java". The library supports the operation of the manger component defined in ISO/IEEE 11073-20601, which is equivalent to the Continua Health Alliance's PAN client component. The project is available in [33].

The solution of Signove Antidote IEEE seems to be the most indicate implementation for this dissertation because the Antidote library has been tested with multiple mobile devices, tested with multiple medical devices and has good support [34]. An other reason is that it can be installed in older versions of Android [35] (version 2.3) and we can test our application without needing a medical device. The library can be used on Linux in which an agent and a manager are available. The Antidote supports all the devices certificated by Continua Health Alliance, even the devices that do not support the standard ISO/IEEE 11073. This solution supports transcoding via a plug-in architecture, in which any developer can write new plug-ins to support more devices [22].

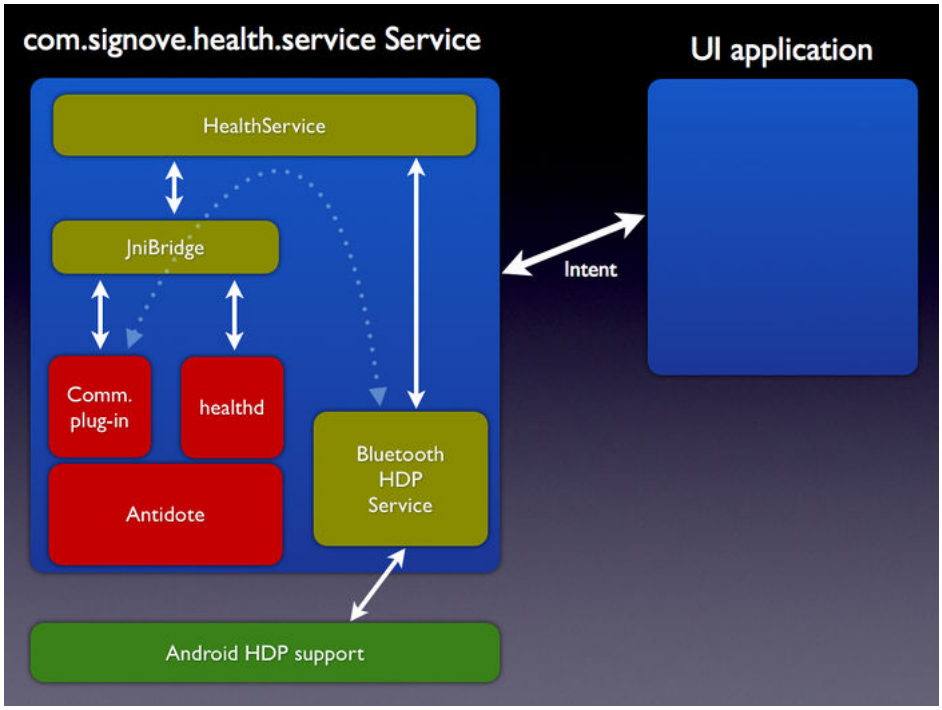


Figure 2.4: Architecture of Antidote [22]

The Antidote never forces the client application to deal directly with MDS types. It encapsulates every piece of data (measurements, configuration, MDS attributes, etc) on DataList structures. Normally, the manager will be a service process separated from the application User Interface (UI) , in which each DataList is sent via Inter-Process Communication (IPC) , and needs to be encoded. Anticipating this need, Antidote offers Extensible Markup Language (XML) encoders. Those formats have decoding facilities in every major language and framework. On the other hand, if the manager wants/needs to access MDS directly, there are no barriers (beyond the complexity of standard ISO/IEEE 11073).

Antidote distribution is implemented in C language and can be built in Android, using Native Development Kit (NDK) [36]. The Antidote is divided in several components as shown on figure 2.4. The blocks in red on figure 2.4 are where the components responsible for standard ISO/IEEE 11073 and Manager are. The other components are the parts written in Java [37] that take care of services Application Programming Interface (API) , HDP communications and timers.

To use the Antidote library we do not need to change anything, we can use library via native IPC available in Android (intent) [38]. The developer can use the Intent to communicate with the service, where he can import this library and use it on any mobile application without much effort.

Analyzed features				
	ClinCapture [39]	Medrio [40]	BioClinica [41]	REDcap [42]
Creation CRF Online	Yes	Yes	Yes	Yes
Manager and Deploy CRF	Yes	Yes	Yes	Yes
Validations on CRF	Yes	Yes	Yes	Yes
Dashboard	Yes	Yes	Yes	Yes
CFR Part 11 Compliant	Yes	Yes	Yes	Yes
Mobile Support	N/A	N/A	Windows 8 APP for data visualization	N/A
Free	Yes, exist plans for extra features	Free for Universities	N/A	Free for REDcap consortium partners

Table 2.1: Analysed features on each EDC System

2.4 ELECTRONIC DATA CAPTURE SYSTEMS WITH MOBILE DEVICES

Before starting to implement a solution for this problem, we did an analysis on the best-known EDC systems to see if some of them already support communication directly from medical sensor or if some have native solutions for mobile. The EDC solutions that were analysed are: ClinCapture [39], Medrio [40], Bioclinica [41] and RedCap [42]. We made a feature analysis of each solution as described the table 2.1.

All these solutions are based on web pages, which offer the possibility of being used on smartphones, but without a native solution, the communication between mobile devices and medical devices will be hard or impossible to accomplish. On the other hand, some solutions already exist to show the communication with medical sensors, but none of the solutions are directly connected with clinical trials or with mobile devices. The solutions analysed are:

- HealthVault [43], developed by Microsoft;
- QualCommLife [44], developed by Qualcomm Life;

These solutions allow the communication between mobile device and cloud, but not the retrieval of data from medical sensors using smartphones. The HealthVault is a cloud service where connected applications, i.e. websites, computer software, and mobile apps, can help the user to see and share their medical information. The HealthVault support communication with 210 [45] third-party devices, but to retrieve the data it is required the download a Microsoft application called HealthVault Connection Center to connect the device to the platform [46]. Some applications that use this solution only allow retrieve data from the sensors that are inside of smartphone, i.e. accelerometer sensor, but nothing more. The QualCommLife use another solution, they have one hub where the data is retrieved from medical sensors and

send to the cloud.

None of these solutions have a communication between EDC systems, mobile devices and medical sensors. Using the standard ISO/IEEE 11073 on smartphones to communicate with mobile devices, using this information to fill-in ECRF on smartphone and send the ECRF to EDC Systems (via cloud solutions), we can attain one possible solution.

CHAPTER 3

Requirements of uEDC Mobile

This chapter presents the requirements of uEDC Mobile and which adaptations must be fulfilled on the uEDC System, before it is possible to use the uEDC Mobile.

3.1 OVERVIEW

Before presenting the proposed solution, we need to understand the flow of the EDC system that is already implemented. The EDC system used is the uEDC product, developed by iUZ Technologies. This EDC system has different instances: when iUZ receives a CRF in paper, prepares a corresponding ECRF, in other words, each instance of uEDC is a different ECRF. All these instances have the same flow, that can be described using the figure 3.1, using Unified Modeling Language (UML) representation [47].

The flow of each instance of uEDC is simple, the responsible of the study has to advertise the study and build a list with the possibles candidates. After this step, the responsible of the study asks each possible candidate if s/he wants to participate on the study. If the answer is yes the responsible will add a new participant on uEDC System and the new participant reads the document that describes all the information about the proceedings of the study. When these two steps are completed, the participant on the study receive a identification number and can immediately answer the study. After all the answers are completed, the participant will end their participation on the study. On the other hand, the responsible of the study

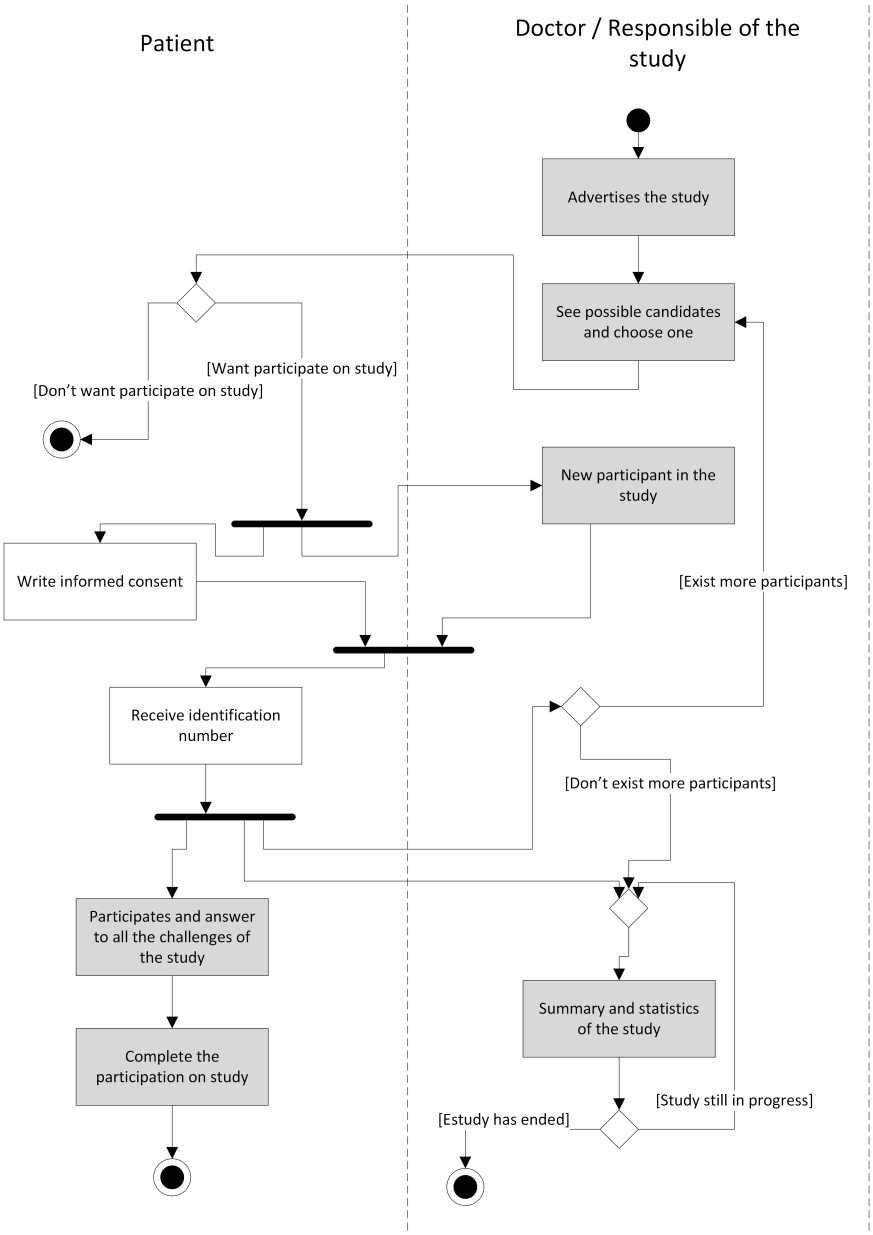


Figure 3.1: Business Model - The areas filled have support of software solution

will continue to check the other candidates in order to gather more participants. At same time, the responsible of the study can see statistics and summary. When the responsible of the study has all the information, one can end the study.

Currently, the full workflow is supported in a web application, which is convenient for deployment, but cannot accommodate the direct access to medical devices.

In this work, we introduce a new module, the uEDC Mobile to support part of the workflow in mobile devices. Before using the uEDC Mobile, the participant must enroll on the study. The enrollment of the patient implies one visit to the doctor because some clinical requirements should be checked [48].

3.2 ENROLLMENT ON STUDY

Each study has its requirements and if the patient does not comply with all requirement s/he cannot participate in the study. Figure 3.2, shows the use case of enrolling in the study. For the patient to enroll in the study, s/he needs to go to the doctor to see if s/he complies with the requirements [48].

Before participating in the study, the patient has to read a written informed consent. This is needed to ensure acceptance of all the conditions of the study.

Each patient that is enrolled in the study has one identification number that identifies the process file and not the patient, because all the answers given by the patient must be anonymous and should not be directly associated with him/her [48] [10]. This use case is only possible on uEDC System, without this step fulfilled, it is not possible to use the uEDC Mobile.

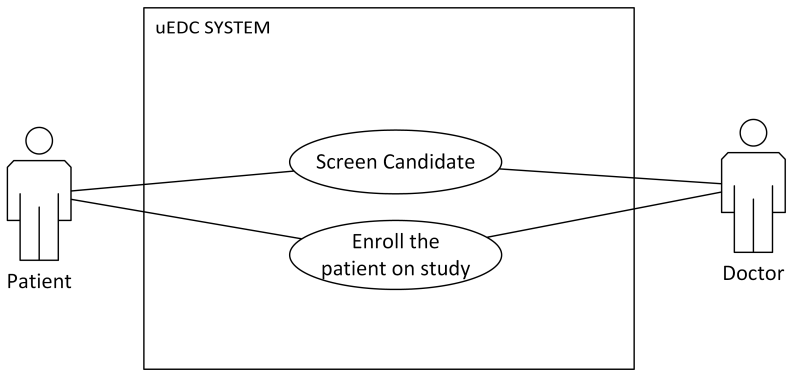


Figure 3.2: Enrollment use case

3.3 UEDC MOBILE USE CASES

After the enrollment, the patient will receive one identification number that will be used to start the study. That identification number is encoded in some specificity type, in this case, it is encoded using Quick Response Code (QRcode) . The QRcode is a machine-readable optical label that contains information about the item to which it is attached [49].

The uEDC Mobile will read that identification number and, with that number, will ask the uEDC System for information that is needed to start the process of configuration of the study. After this first step, the patient can use the uEDC mobile to participate in the study, as described in the figure 3.3.

uEDC mobile should support the following functional:

Check-in in a new study

The check-in has to be simple and intuitive to use. The study can be answered by older people or for people who are not used to mobile devices. To get the study it is mandatory to have some type of Internet connectivity. The check-in is only available to patients that are already enrolled on the study.

Answer the questions in study

The system has to guarantee the possibility of answer the questions that can occur at different moments in time. Every time one group of questions is completed or changed

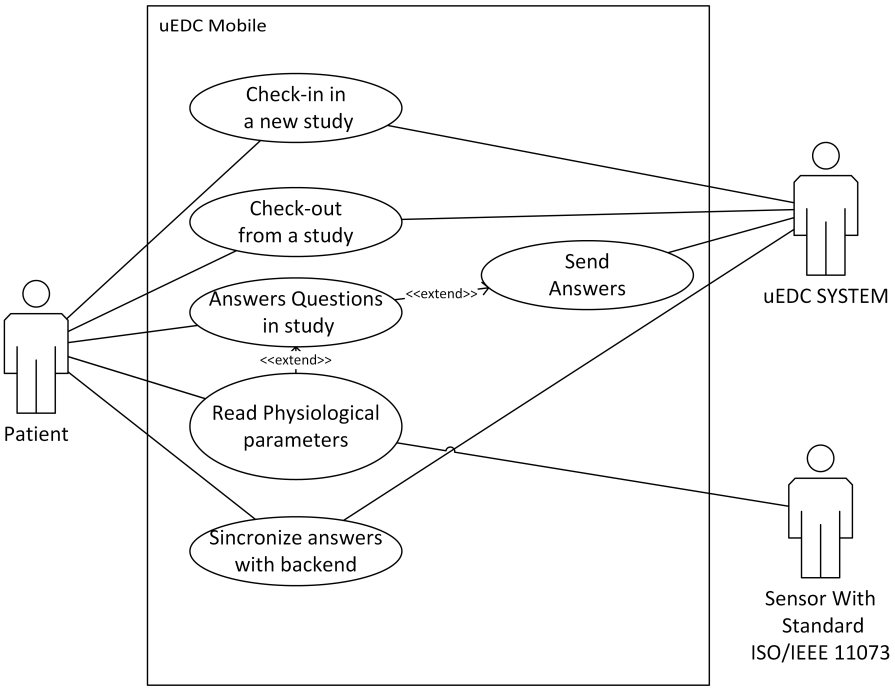


Figure 3.3: uEDC mobile use case

they can send that data to uEDC System.

Synchronization with backend

If the mobile device has some type of Internet connectivity, the system will try to synchronize the answers with uEDC System. With this functionality it is possible to answer the questions in multiple platforms. For example in the web browser when we are at home or mobile device when we are traveling.

Read physiologic parameter

Some studies need medical data read from the subject's body. The system will retrieve the medical data directly from the device. If that is not supported or the patient does not have that kind of medical sensor, the system will ask for manual insertion of values.

Check-out from a study

After the patient answers all the questions, it is possible to finalize the study, ending the participation of the patient in the study. To do this is mandatory to have some type of Internet connection.

Security measure

uEDC Mobile must support some functionalities that give to user a possibility to add an extra protection for use the application, e.g. the possibility of add a pin (pin that exists on SIM card).

The following non-functional requirements should be addressed:

Target devices requirements

To use this application it is required one smartphone or tablet that has the Android version 4.+ because it is used the HDP profile that is only available on version 4.+. In order to read the QRcode, the smartphone has to have one camera installed.

Data integrity

uEDC Mobile must save all the data internally without a external access, e.g. save on internal database.

Privacy

The elimination of any direct association with the user in the data. Each user has a random number that is not associated with personal information (email, name, etc) that exists on the smartphone/tablet.

Usability

uEDC Mobile must support the integration between medical devices that communicate via Bluetooth, in specificity, sensors that have the standard ISO/IEEE 11073

CHAPTER 4

uEDC Mobile System

This chapter presents the uEDC system and uEDC mobile system. uEDC System is a web platform for clinical studies management by iUZ Technologies. The uEDC mobile system is a new module to extend the uEDC for mobile. This chapter focuses on uEDC System/uEDC mobile System architecture.

4.1 OVERVIEW

The core of uEDC System is a product already in market, and the goal is to make an extension to work with the existing system. Figure 4.1 presents the actual architecture of EDC System, following a very common architectural pattern, the 3-tier Architecture [51].

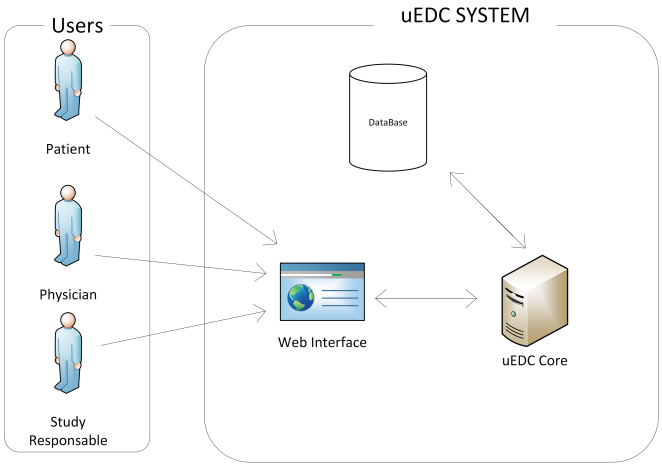


Figure 4.1: High-level uEDC System without the mobile component

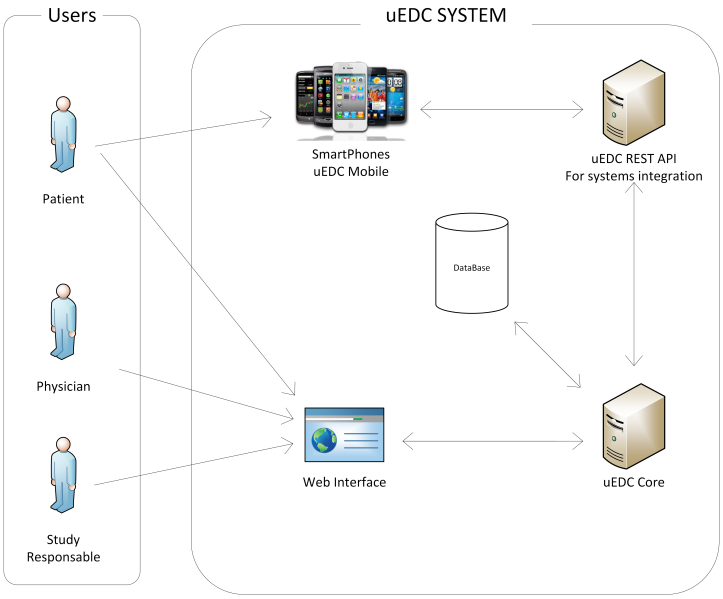


Figure 4.2: uEDC System with mobile devices

The architecture 3-tier is divided in three layers or tiers [52]. In uEDC System, each tier is represented by:

- Client tier – Web interface;
- Business logic tier - uEDC Core;
- Database tier - Database.

To integrate with this architecture we are going to use a Representational state transfer (REST) API [53][54]. This REST API allows for communication between the uEDC System and the mobile devices. One possibility is shown on figure 4.2, which consists in the creation of a module that exports some of the functionalities available of the uEDC Core.

This new module is a REST API that responds to requests made by the uEDC mobile. With this module, the uEDC Mobile can easily be used on the uEDC System. The creation of this new module is responsibility of iUZ to ensure all the security necessary on the project.

STRUCTURE OF STUDY

As explained in chapter 3, each instance of uEDC System is an ECRF. uEDC System divided these ECRF into three groups :

Observations

Each study has multiple observations. These observations represent a point in time. Normally the study is not filled all at once, i.e. the first observation can be completed

today and the following only after 6 months. The protocol of the study defines the interval of each observation.

Evaluations

Each observation has one or more evaluations. Each evaluation has a direct relation with what is evaluating, e.g. if we are dealing with eligibility, demographic, medical data, etc. The same evaluation can be inside of multiple observations.

Groups

Each evaluation has one or more groups. Each Group has a number of questions that are associated with the Evaluation, e.g. the Evaluation medical data can have one group that has the questions about the blood pressure and other group with the questions about the height.

FUNCTIONALITIES OF THE API DEVELOPED BY IUZ

The iUZ developed an API REST that allows the following iterations:

- HttpGET patients/patientId

  This request returns the cases associated so far with the patientID. This case is a copy of ECRF on uEDC system plus the answers made by the patient.

- HttpPOST cases/caseId/answers

  This request sends the answers done by the patient. Each patient has associated one caseId that identifies the patient on that study to avoid a direct association with the patient.

- HttpPut cases/caseId/answers

  This request adds new answers and terminates the observation. Normally, it is used to terminate the observation.

Therefore, the uEDC Mobile uses this API REST to communicate with the uEDC System allowing the integration between these two systems.

4.2 UEDC MOBILE ARCHITECTURE

In the previous sections, we explained how the uEDC system works (in high level) and the changes needed to enable the interaction with uEDC mobile. This section presents the internal workflow of uEDC mobile system and interaction of each component to guarantee the requirements that are on chapter 3.

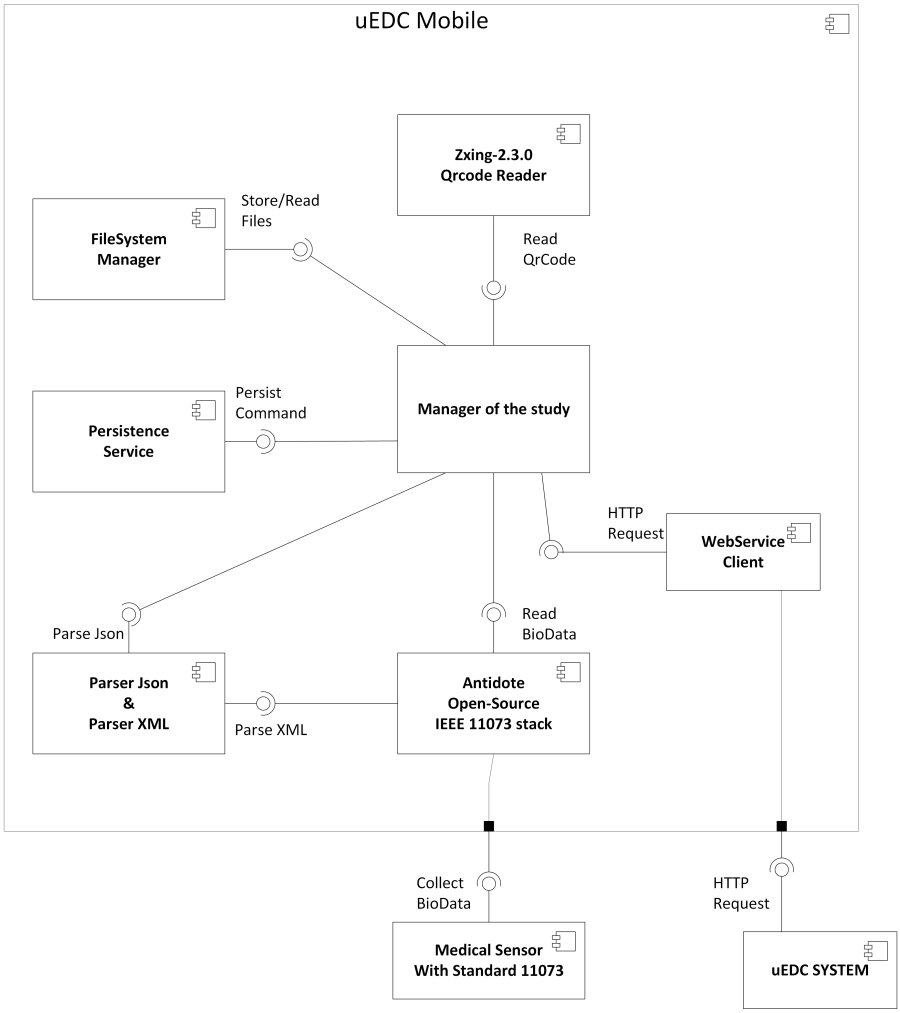


Figure 4.3: Component diagram of uEDC Mobile system

4.2.1 STRUCTURE OF UEDC MOBILE

The uEDC mobile is logically divided into several components as shown on figure 4.3.

The component zxing-2.3.0 QRcode Reader is a QRcode reader that is responsible for reading the QRcode that will be presented by the user. This QRcode has information about the study, e.g. the name of study, the responsible of the study or the logic location of the study, and some Information about the user (the id of the case for example). The component FileSystem manager is responsible for saving/reading all the files used on uEDC Mobile. The structure for the file storage is explained in chapter 5.

The component Persistence Service is responsible for database access on uEDC Mobile. The component parser JavaScript Object Notation (JSON) & parser XML has the responsibility of parsing the JSON [55] and the XML [56] into objects in Java.

The component Antidote Open-Source ISO/IEEE 11073 is a library that has the respon-

sibility to do the communication between the mobile device and the medical sensor. This library implements the standard ISO/IEEE 11073, where all the data retrieved is in XML. This library needs an XML parser to extract the information that is sent by the medical sensor. The component WebService Client is responsible to use the network, in order to do the requests that will be sent by other components. The component allows for requests, both synchronous and asynchronous.

The main component is the manager of the studies, with two main subcomponents:

Fetch/configure study: This component is responsible for fetching the configuration file from the uEDC system and for the configuration of the study.

Questions manager: This component is responsible for creating the views that will be shown to the user and for managing the answers. In order to do that, the component will use the files that were created by the component fetch/configure study.

4.2.2 WORKFLOW OF UEDC MOBILE

The workflow of the uEDC mobile is very simple to understand, as stated previously, the main components of the uEDC mobile are the fetch/configure study and manager of questions. The workflow of the fetch/configure study component is described in figure 4.4. In the beginning, the component will read the QRcode presented by the user. If the QRcode is invalid, the component informs the user that is not the correct QRcode, but if the QRcode is valid, the component WebService Client gets the files needed in uEDC System.

After getting the response, the component converts the response into objects in Java, parsing the JSON. Once the Java object is created, it is persisted (by serialization). After this process of fetching/configure, the study is ready for use.

The workflow of Questions Manager is described in figure 4.5 and is a little more complex than the fetch/configure study component. This component manages all the questions and all the save/send for the user answers.

In the beginning, the component checks if the study is configured or not. If not, the component redirects to the fetch/configure study component but, if it is configured, the process can advance on the workflow. After this first step, the component will use the FileSystem Manager to seek and read all the files necessary for the study that was chosen by the user. In these files, some of them have the configuration of the navigation inside the study, the menu of the study. Now the user can choose the group that he wants to answer or see the answers

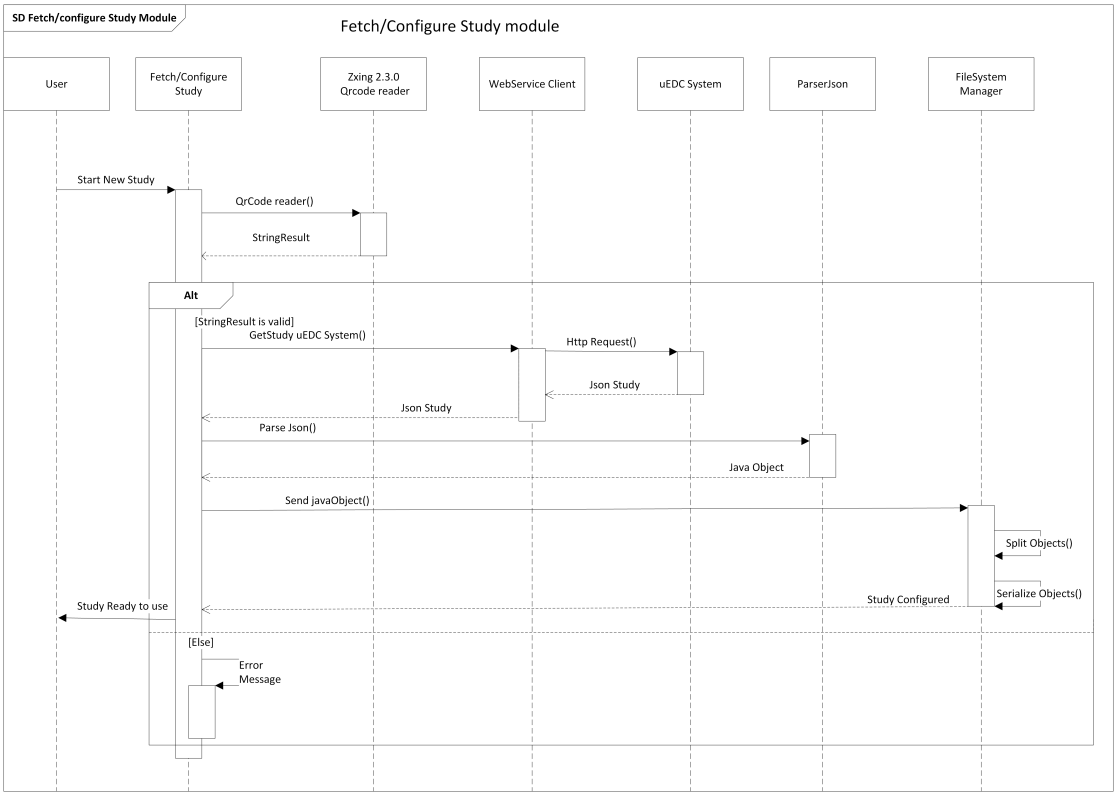


Figure 4.4: Workflow of the component fetch/configure study to configure one study

already made. This process can take some time because it is necessary to seek information and configure all the views before it can be used.

Every time, the user changes or makes a new answer the component saves locally the answer, but if the user changes a group or study, the component synchronizes the answers with the uEDC System. This is to ensure the synchronization with all the system because the user can answer the same study on mobile or on a web page. With this synchronization, the user can answer on mobile while s/he is travelling and on the web page when s/he is at home.

Every time the group is changed, the component creates new views with the answers made. After all the questions are correctly answered, the user can send the answers to uEDC System to end the participation on the study, but if the answers are incorrect, the component shows an error message to the user and the participation on the study will not terminate until all the answers are correct.

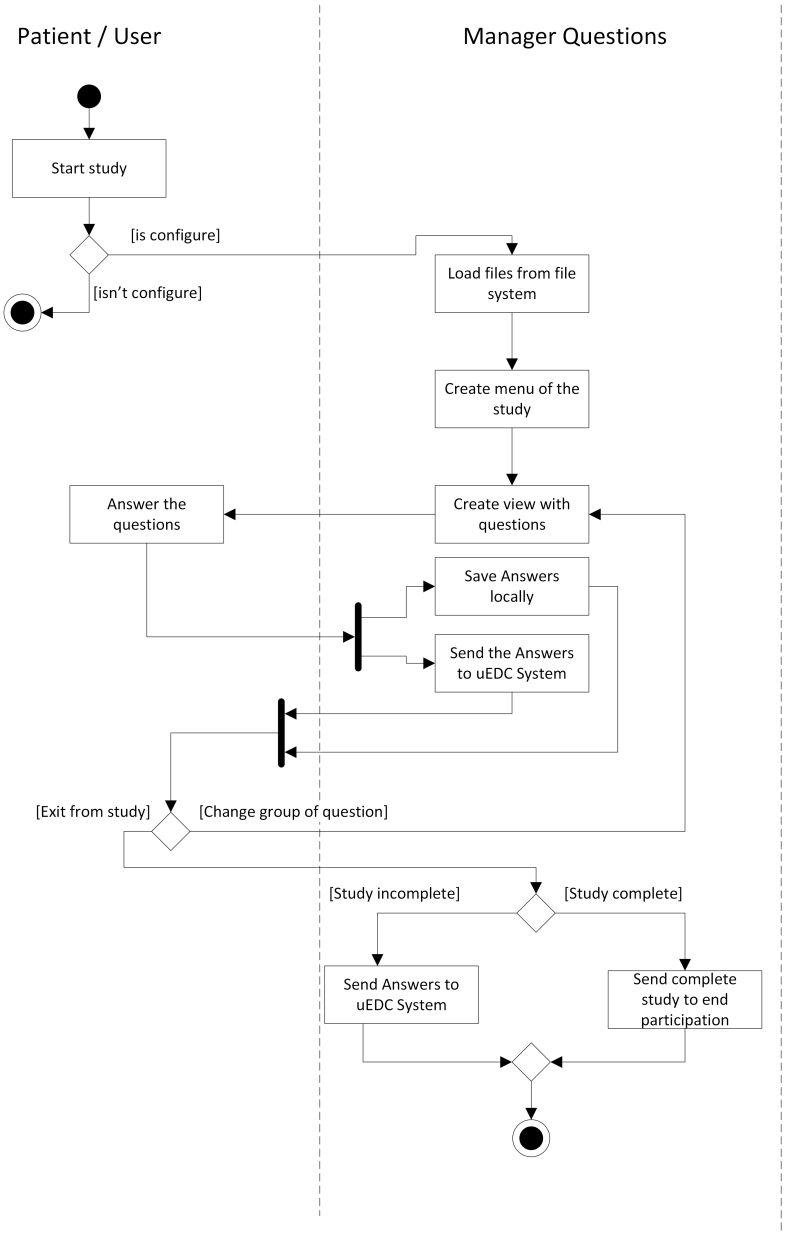


Figure 4.5: Workflow of the component questions manager to answer one study

CHAPTER 5

uEDC Mobile Implementation

This chapter presents the implementation of uEDC Mobile. We show the components that are inside of the uEDC Mobile and which solutions we use to solve the problems.

5.1 OVERVIEW

The objective of uEDC Mobile is to be an extension of the uEDC system that is already on market, to benefit from the use of mobile devices. uEDC Mobile was designed and implemented as a native Android OS application for version 4.+ because it uses technology that only exist on the version 4.+ Ice Cream Sandwich (ICS) (the HDP). One reason to adopt Android OS is that it has a larger market share. The other reason is that the uEDC Mobile uses the library Antidote: ISO/IEEE 11073-20601 stack. Antidote is an open-source library that implements ISO/IEEE 11073-20601 protocol stack. Health and fitness devices employ this standard, to retrieve data from device and fill-in ECRF. That is the big achievement of the uEDC Mobile.

The uEDC Mobile uses one database to save all the answers and errors that exist in each study. This database is implemented in SQLite [57] native on Android OS. The uEDC Mobile also uses the shared Preference of Android OS to help the navigation on each study. The other EDC Systems can use the uEDC Mobile, but each system has to provide the same aspects that the uEDC System use to communicate with uEDC Mobile. Each study is an

instance of uEDC System where the uEDC Mobile can iterate with multiple instances of the uEDC System at the same time.

A complementary Android application was developed to demonstrate the communication between the Android device and the medical device. This Android OS application (Blood Pressure Tracker) is an application to manage the blood pressure. The user can manually insert the data or use a medical sensor that uses the ISO/IEEE 11073. In this application it is possible to see more information of the data collected, for example, the date of the collecting or the sensor that sent the data among other things (see also 5.3). In both applications, Blood Pressure Tracker and uEDC Mobile, the minimum required version of Android OS is 4.x. uEDC Mobile has other requirement, as the existence of a camera is required to read the QRcode.

5.2 UEDC MOBILE

This section presents the technologies used and how the uEDC Mobile works. As described on chapter 4, the uEDC Mobile is divided in several components where each component on uEDC mobile is a set of Java classes.

5.2.1 OBTAINING CLINICAL TRIALS

The uEDC Mobile supports multiple clinical studies. Each patient has a number encoded using a QRcode. The Zxing 2.3.0 library is used to read the QRcode. To use this library, the uEDC Mobile calls an intent and after that, the Zxing library reads the QRcode and returns a String result. uEDC Mobile uses the Android Shared Preferences [58] capability to store the processed string.

With this process completed, the uEDC Mobile has all the information needed to get the study on uEDC System, addressing the REST API, to return the complete study. Each study has the following information:

- The description of the questions - with codes for each question;
- The answers that are saved on uEDC System;
- The restrictions of each question (validations that are to be fulfilled).

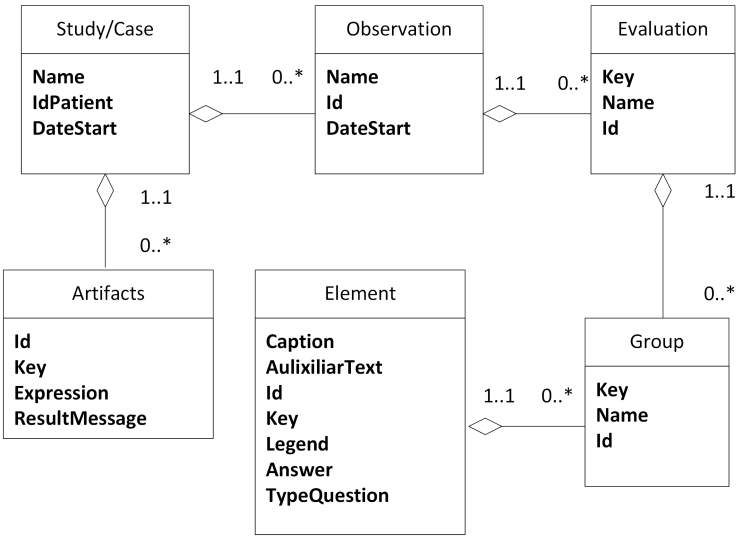


Figure 5.1: Structure of a study in JSON

FORMAT OF THE STUDY

The study that is sent has a well defined structure that is similar to the structure used on the uEDC System backend database, as described in figure 5.1. The study is divided in two big groups: cases and validations. The cases group contains the information about questions and the answers. The validations group contains information about the restrictions of each question.

The questions of each study are divided in three parts: the observations, the evaluations and the groups. Each Study can have multiple observations, the observations can have multiple evaluations and the evaluations can have multiple groups. The class group contains the information about the questions and the answers. Each validation is divided in an artifact, where inside of each artifact exists:

- The error messages;
- The target question or questions;
- The regular expressions that are need to be fulfill.

Each question can have multiple validations and there are validations that target more than one question. The uEDC Mobile receives the study and converts it to Java objects to be simpler and quicker, in terms of performance, to search and navigate inside of the study. The parsing of JSON to Java uses a library called Gson [59]. Google develops this library to help in the conversion of JSON to Java objects and vice-versa.

STRUCTURE USED ON THE FILE SYSTEM

After the conversion, the uEDC Mobile will split the object in other small objects and save them in the file system. In this process, the object is divided in observations, validations and menu objects. For example, if the study has three observations this process will split the objects in:

- Three observations objects;
- One object that has the structure of the observations;
- Three menu objects;
- One validation object;
- One object that has the information of the whole study.

When all objects are created, the uEDC Mobile calls the File System Manager, to do the serialization of each object and saves it in file system. The quantity of objects can increase linearly and raises the need for a structure on the file system. After this step, the patient can answer the clinical trial.

5.2.2 ANSWER THE CLINICAL TRIAL

Figure 5.2 illustrates the most important features implemented in uEDC Mobile. The numbers in red indicate the sequence present and the arrows indicate one possible navigation in uEDC Mobile. All the data used are fictitious and simulated for effects of demonstration. The study used was developed by iUZ to demonstrate the potential of uEDC System.

To answer the questions, the user has to see the list of the studies in progress as shown in figure 5.4. If the user only has one study or wants to go to the last study, he can resume using the button "resume study" on main page as shown in figure 5.3. If the user already started answering a study, the application will put one pin on the last study that was visited. After the user chooses one study, the menu will appear. This menu shows three possibilities to choose from:

Show more information about the study

On this option, the uEDC mobile shows information about the study, for example, who is responsible, the start date of the study, whether the study is completed and other information. On this option it is possible to submit the study to the uEDC System as shown on figure 5.5. This option requires Internet connection to complete the study.

Figure 5.2: StoryBoard of uEDC Mobile

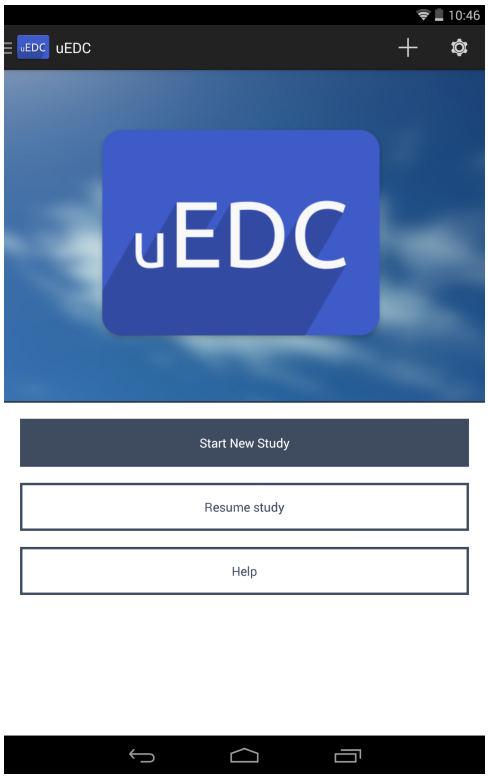


Figure 5.3: Main Page of uEDC Mobile

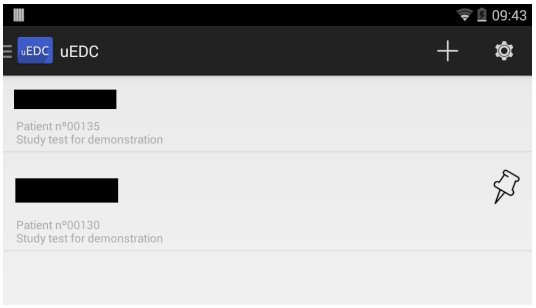


Figure 5.4: uEDC mobile List of studies

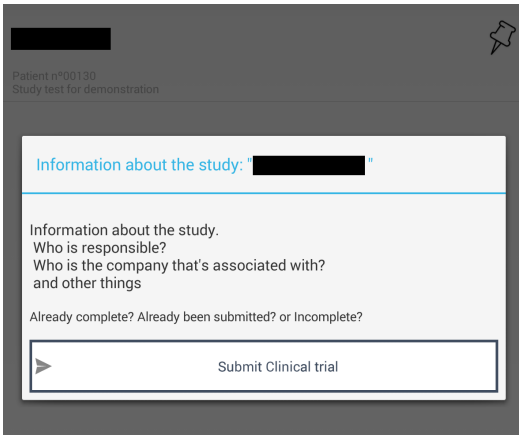


Figure 5.5: Information about the study

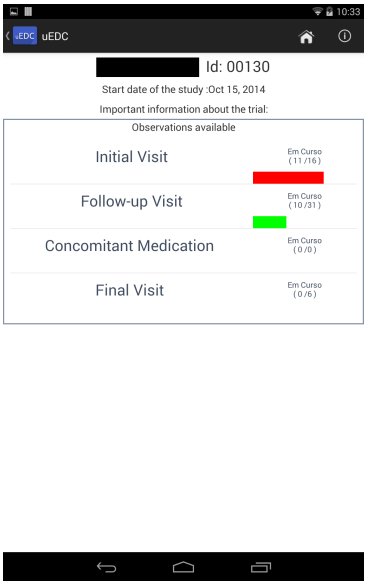


Figure 5.6: Information about the current state of the study

Resume study

This option allows the user to resume the study. The user will start in the last group where he has paused. The uEDC Mobile saves the last group visit in each observation to improve the usability of uEDC Mobile.

Delete study

This option allows the user to remove the study on uEDC mobile but the answers in the uEDC System will remain unaffected. To delete all the answers on the uEDC System, the user has to contact the responsible of the study.

Once the user resumes the study, the user accesses another window, as shown in figure 5.6, with all the observations available on the study, the number of questions that are missing (through a progress bar) and warnings (through a color scheme). After the user chooses an

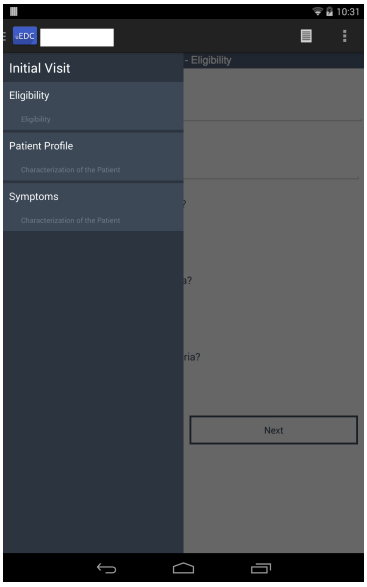


Figure 5.7: Menu that is inside of each Observation

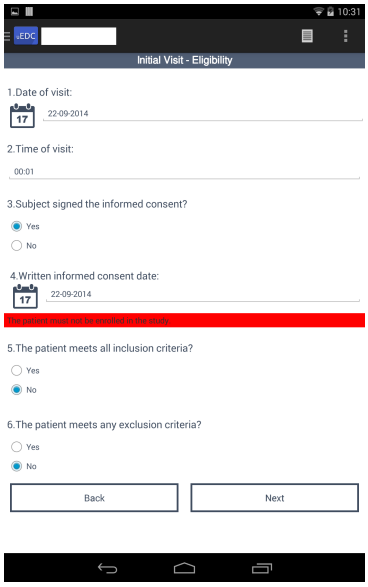


Figure 5.8: View example of one group of questions

observation available, the uEDC Mobile reads the files and creates the menu of navigation, as shown in figure 5.7, and the questions, as shown in figure 5.8. In this point, the user can answer all the questions that are in the observation. The figure 5.8 shows one example of a view with questions and an answer with warnings. This answer can compromise the participation in study and the uEDC Mobile will block the remaining groups until the responses that compromise the study are resolved.

MEDICAL DATA COLLECTION

In some studies where it is needed, at some point, to collect physiological data. For that, the user can put the data manually, as shown on figure 5.9. If the user manually inserts the physiological data, there is a chance to manipulate the data or maybe a possibility of an error on the data entry. uEDC Mobile will see if the data is between the normal values but the user can put the value s/he wants freely. To solve these problems we provide the possibility of retrieving physiological data using sensors that use the ISO/IEEE 11073 standard.

uEDC Mobile checks the type of data that it needs to collect, using the nomenclature defines on the standard, and sends that information to Antidote library. As shown in figure 5.11, the Antidote library retrieves the data from the medical device and presents it. The user can choose if he wants to save the data or if he wants to repeat the reading. Only the last data shown will be saved internally. Figure 5.10 shows the data already integrated with the study.

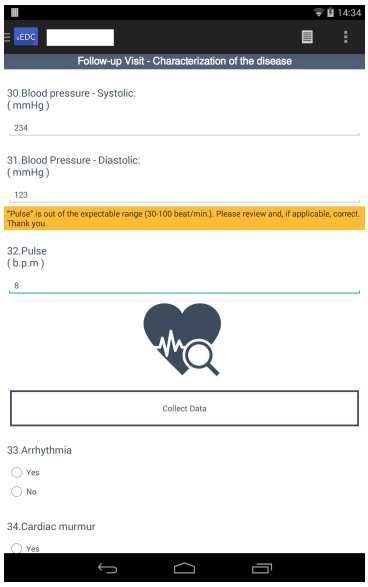


Figure 5.9: Collect medical data using a medical sensor or manually

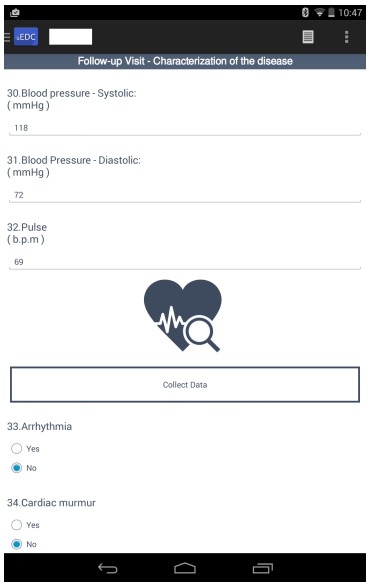


Figure 5.10: Data collected integrated with the Study

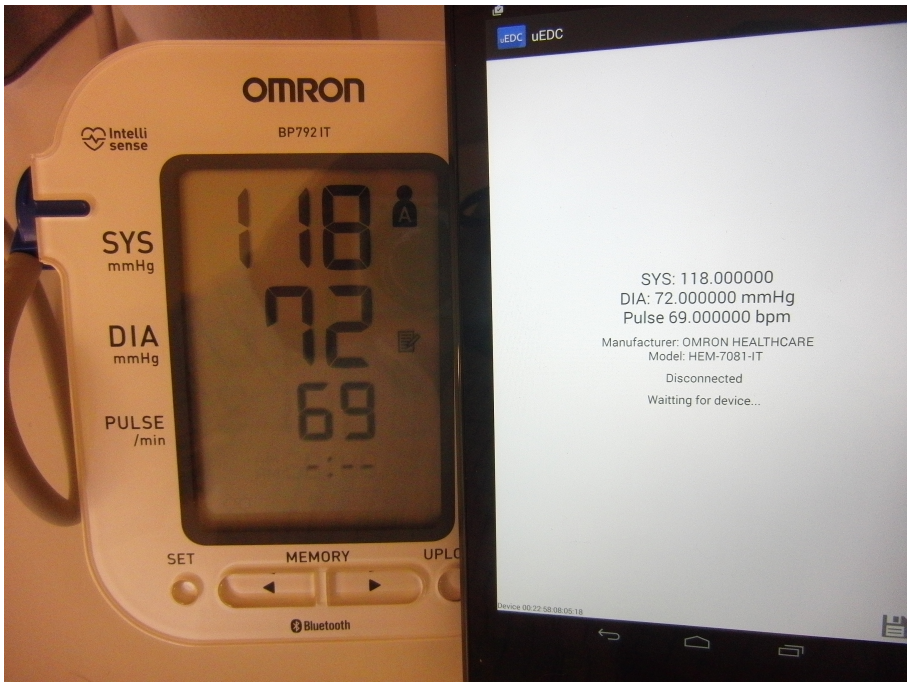


Figure 5.11: Retrieving medical data from sensor

SYNCHRONIZATION/SUBMISSION OF ANSWERS

Every time the user changes a group or leaves the study, the uEDC Mobile sends the data to the uEDC System if a connection is available. For this process, the uEDC Mobile uses the WebService Client component that relies on the Volley library [60] to send the request. The body of each request contains the answers in JSON. This library does all the management of the network and the requests will be asynchronous. With this possibility, the user can continue without waiting for synchronization to complete.

At the end of each observation, the user can submit the answers and close the observation. This will be possible if the observation does not have any errors and if exists Internet connectivity. This process will be synchronous and the user has to wait until all the answers are submitted. With submission of the answers the study can change, for example other observations can be changed from blocked to available. To refresh the state, the uEDC Mobile will do a new fetch/configuration of the study, which implies the creation of all files again.

5.2.3 SECURITY

In terms of security, uEDC Mobile has some solutions to protect the identity of the patient and the medical information. The information that identifies the patient is encrypted, e.g the identification number of the case and the identification number of patient is coded. All the decrypting and encrypting is responsibility of uEDC System to reduce the processing time on the mobile device.

The other security measure is when uEDC Mobile sends data to uEDC System. All the data is coded with numbers, omitting the semantic of values, i.e. the question "Subject signed the informed consent" is sent with the internal code of the question and the answer. Figure 5.12 shown an example of the file that is send to uEDC System, respecting the format of the study, the questions code and the answers made by the user.

The user can activate the use of a security pin, like the pin we have on credit card or on sim card, as shown in figure 5.13 and will be asked every time launch uEDC Mobile is launched. With this measure, the user can have an extra security layer to protect his medical data.

```
{"observations": [
        {"evaluations": [
                {"groups": [
                    {"answers": [
                          {"answerCurrentValue": "118",
                           "questionId": "8191",
                           "questionKey": "Q41",
                           "questionName": "8191",
                           "questionText": "30.Blood pressure -
                              Systolic: "},

                          {"answerCurrentValue": "72",
                           "questionId": "8192",
                           "questionKey": "Q42",
                           "questionName": "8192",
                           "questionText": "31.Blood Pressure -
                              Diastolic: "},

                          {"answerCurrentValue": "69",
                           "questionId": "8193",
                           "questionKey": "Q43",
                           "questionName": "8193",
                           "questionText": "32.Pulse "},

                          {"answerCurrentValue": "3",
                           "questionId": "8194",
                           "questionKey": "Q44",
                           "questionName": "8194",
                           "questionText": "33.Arrhythmia "},

                          {"answerCurrentValue": "3",
                           "questionId": "8195",
                           "questionKey": "Q45",
                           "questionName": "8195",
                           "questionText": "34.Cardiac murmur "}
                           ],
                           "id": 1205,
                           "status": 5
                    }]
              ,"id": 703}]
      ,"id": 402}]
}
```

Figure 5.12: Example of one answer sent to uEDC System

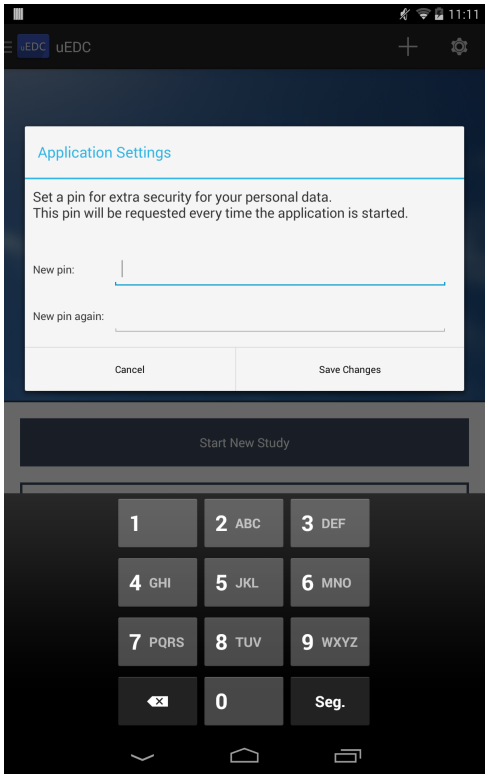


Figure 5.13: Define or remove the pin

5.3 BLOOD PRESSURE TRACKER

The blood pressure tracker is a proof of concept to show the communication between mobile device and medical device using the standard ISO/IEEE 11073. This application is a blood pressure tracker in which the user can save this own records of the blood pressure. If the user does not have a device that supports the ISO/IEEE 11073 or Continua health Alliance certification, the application gives the possibility to insert the records manually. The user can see trends of his records or export that information for his doctor. The application is not related in any way to the uEDC solution, only serves as a helper sand-box to test device interoperability.

With this application it is possible to test the communication between Android devices and medical devices more easily because the uEDC Mobile is property of iUZ and the is some data that needs to be protected. This application is on the Android application market and can be test by the community in general.

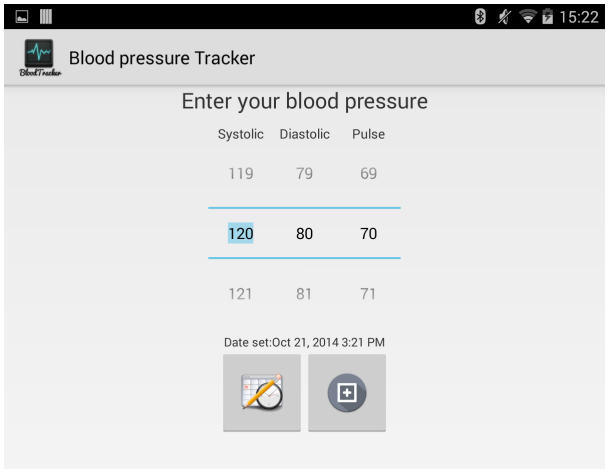


Figure 5.14: Insert the data manually

5.3.1 DATA COLLECTION

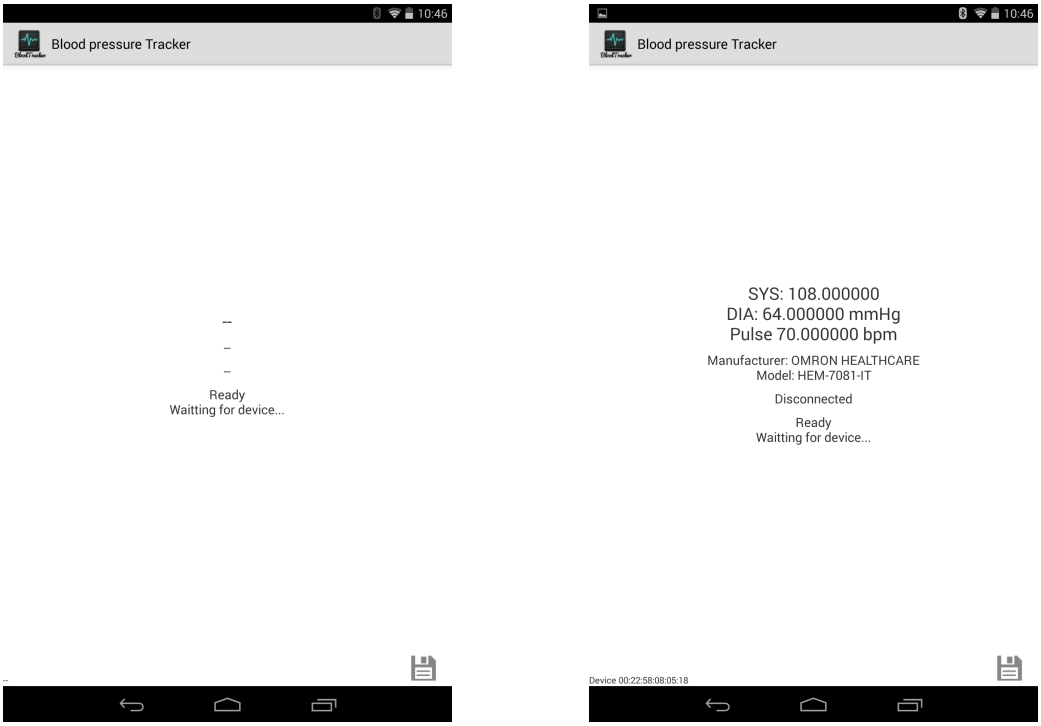
The blood pressure tracker is mainly for the collection of medical data using medical devices or manually. When the user wants to insert more records, the application shows an menu in which it is possible to see these options:

Manually

As shown in figure 5.14, the user can set the blood pressure data and the hour.

From Device

In this option similar to the one used on uEDC mobile, as shown on figure 5.15, which shows the interoperability of Antidote library and how simple it is to integrate the standard ISO/IEEE 11073 in the mobile application.



((a)) Waiting for a communication from a sensor ((b)) Retrieving medical information from device

Figure 5.15: Collection of medical data using standard ISO/IEEE 11073

5.3.2 VISUALIZATION OF DATA

The other feature is the visualization of data, in single samples or on one graph, where the user can see his trends. Figure 5.16 shows examples of the visualization on isolated samples in which the application shows if the data collected is inside of the normal values or if it is too high or too low. As shown in figure 5.16, the second record has the diastolic too high to be considerate a normal value.

If the user selects a record, the application will show another view, as shown in figure 5.17. Here it is possible to see more information about the selected record.

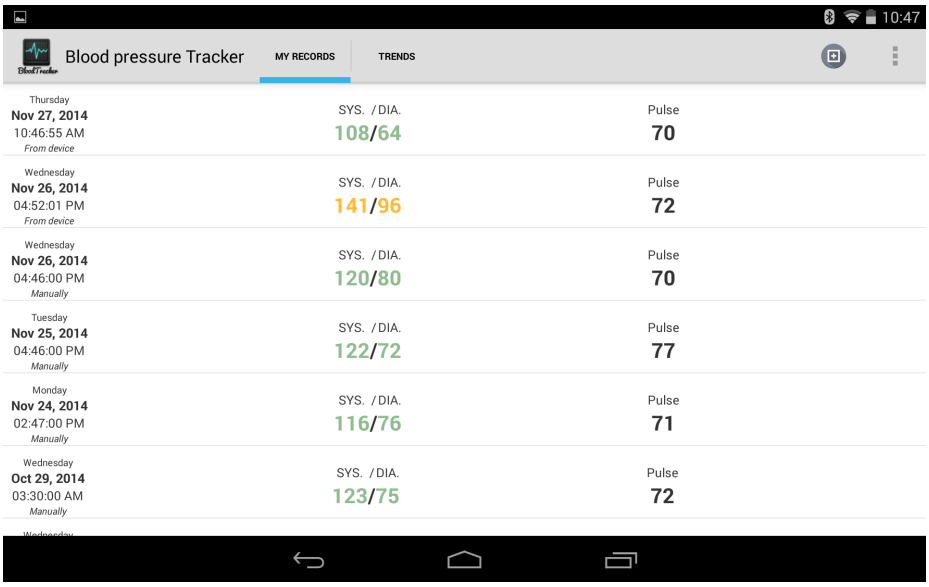


Figure 5.16: List of records

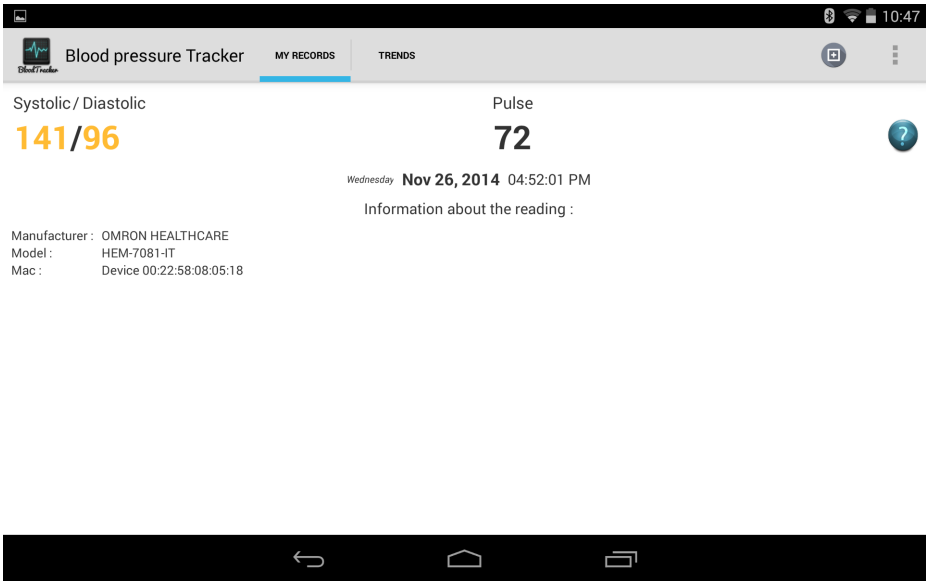


Figure 5.17: More information about the record saved

EXPORT MEDICAL DATA

The user can export data into Comma-Separated Values (CSV) or XML files. These two formats will be saved on the file system, e.g. the user can copy these files and share them with their doctor or import to Excel. However, the Blood Pressure Tracker provides two other ways to export the data: share and synchronize with cloud.

For the process of synchronization with cloud, we created a web service on Google AppEngine [61] and the Blood Pressure Tracker sends the data associated with Google account. This way, the user can synchronize the medical data across several mobile devices.

VISUALIZATION OF TRENDS

The other way to see the records is using one graph, as shown in figure 5.18, where the user can see the evolution of his health values.

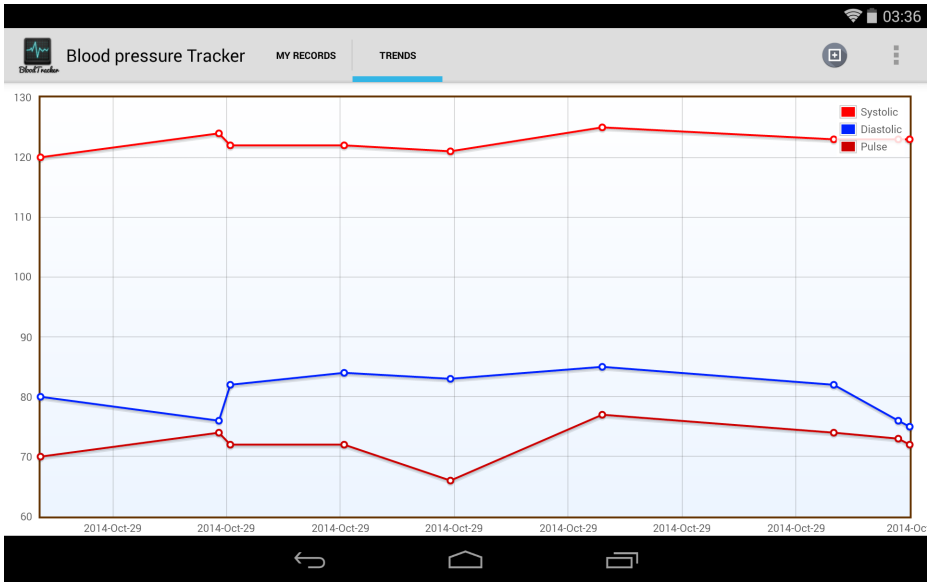


Figure 5.18: Graph with records

CHAPTER 6

Results and system validation

This chapter presents the tests performed and the results that we obtained. These tests are at the level of usability and interoperability between medical devices and mobile devices. For the usability tests we used a simple questionnaire in which the users would input their opinion about each task and about the application. Figure 6.1 shows the setup used for this tests.

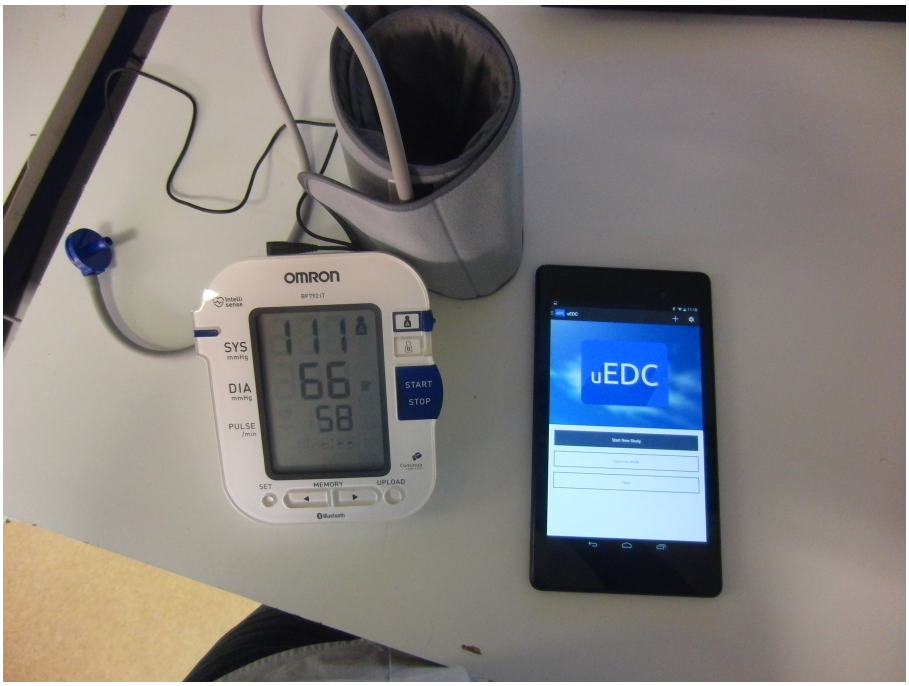


Figure 6.1: Setup used to test uEDC Mobile and Blood Pressure Tracker

	Nexus 7	Samsung Galaxy Fame	Sony Xperia Tablet Z LTE
Android OS Version	4.4.4	4.1.2	4.4.4
Screen Size	7.02"	3.5"	10.1"
Screen Resolution	1920x1200 HD	480x320	1920x1200
Camera	Yes	Yes	Yes
Bluetooth	Yes, version 4.0	Yes, version 4.0	Yes, version 4.0

Table 6.1: Mobile devices used

The tests made on uEDC Mobile were focused on usability, while the ones on Blood Pressure Tracker were focused on interoperability. The mobile devices used in developing and testing the uEDC Mobile and Blood Pressure Tracker are represented on table 6.1.

6.1 UEDC MOBILE

The usability tests made on the uEDC Mobile consisted of the filling of a questionnaire after the user has performed some tasks designed to test the usability of uEDC Mobile. This kind of test measures (in statistic terms) the usability of an application and helps the improvement of some tasks that take more time [62]. Each task can be evaluated by a numerical ranking going from 1 (easy) to 5 (hard). After the all the tasks are completed, the user can evaluate and leave some suggestions about the application.

The tasks chosen for this questionnaire were the principal functionalities of the uEDC Mobile:

- Start new Study (Task 1);
- Select one ongoing observation (Task 2);
- Answer the questions of the selected observation and end it (Task 3);
- See the list of studies ongoing (Task 4).

The evaluation is used as a post questionnaire method to measure the usability on the system, where we aim to evaluate:

- Intuitiveness (Evaluation 1 and the ranking was from 1 (easy) to 5 (hard));
- Necessity of learning in order to use the system (Evaluation 2 and the ranking was Yes or No);
- Whether the user would use the system again (Evaluation 3 and the ranking was Yes or No).

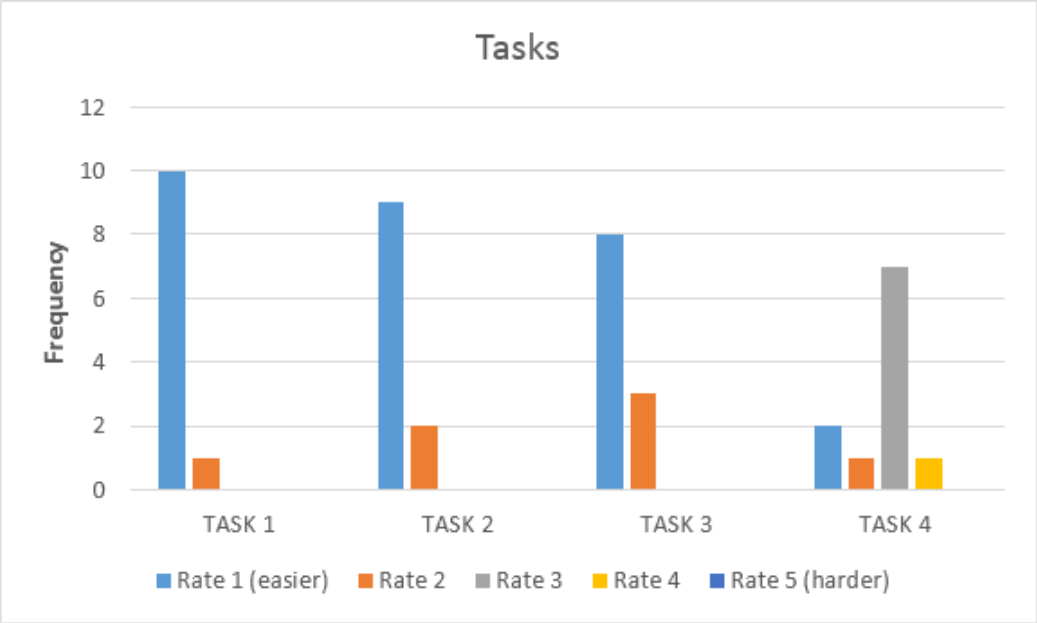


Figure 6.2: Result of the tasks

All the users receive two papers and a tablet. One paper has the questionnaire, the other paper has a QRcode and a quick help about how to use the application, and the tablet contains the uEDC Mobile installed. The study used on this test was developed by iUZ, in which there is a study example to demonstrate the potential of the system.

In this test, we count with the help of 11 participants, with ages between 21 and 24. All these users have experience with smartphones or android tablets. Figure 6.2 shows the results of the usability test, in terms of percentage and tasks, where the most difficult task was "see the list of studies ongoing". The list of studies is on the menu but most of participants (on suggestions) tell that it is not to intuitive to use the menu. This aspect was solved by adding one more topic about this in the menu help but could be solved to, by adding a new button on main page.

Figure 6.3 shows the results of the first two evaluations and the figure 6.4 shows the results of the other two evaluations. Analysing the result we see that 70% of the participants feel the necessity to attain some know-how before using the system. This happens because most of the participants never enrolled on a clinical trial. This data indicates that the new users or patients need to have some kind of explanation about how the clinical trials and the uEDC work. On uEDC Mobile, this kind of explanation can be resolved by creating a well-done help, e.g. using videos. On the other side, 90% of the participants tell that, if the application goes to the market they may use the application again. This is a positive indicator to advance with this solution.

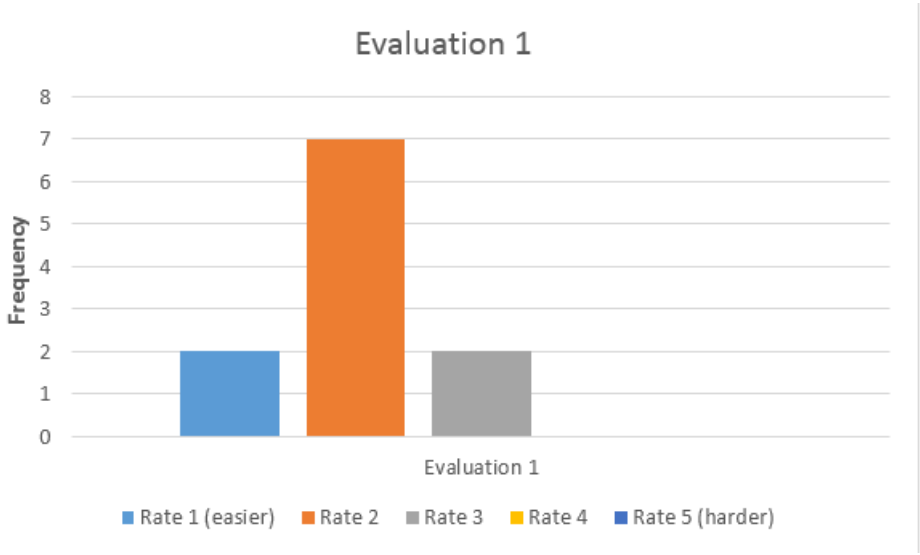


Figure 6.3: Result of the evaluation 1

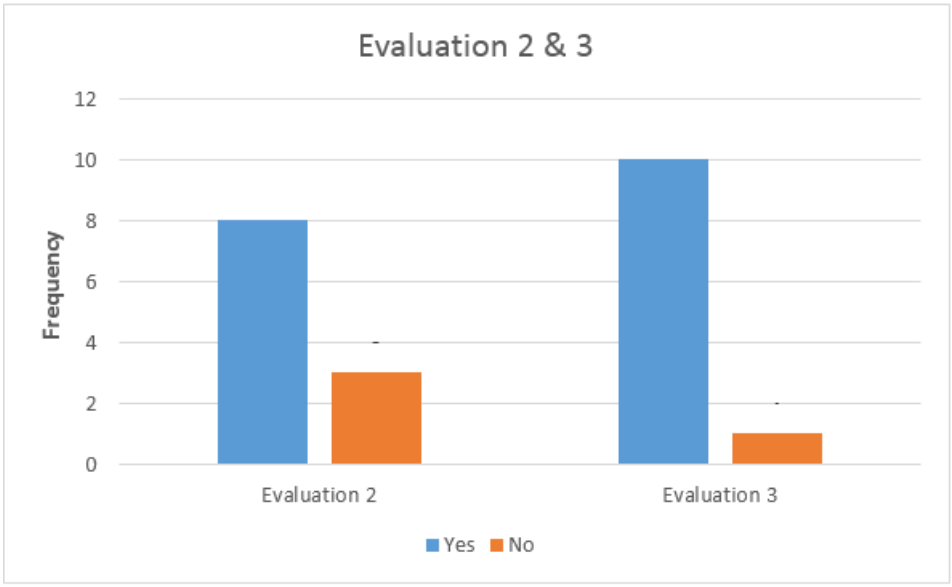


Figure 6.4: Result of the evaluation 2 and 3

The integration of the uEDC Mobile and production platform are currently made. Some types of questions do not exist due lack of support on REST API. In these cases, the uEDC Mobile notifies the user that s/he needs to check the web site. In term of validations, the uEDC Mobile only supports simple validations, e.g. if a date is bigger than another or if the number inserted is inside of a range. For the future, we think that it is better to do these validations in the server side. This solution removes some complexity on mobile device and, consequently reduces processing time.

In terms of adaptation of the uEDC Mobile with a new backend, the changes required are:

New instance of uEDC System: The information about the location (web url) is inside of the QRcode.

New REST API: In this case it is necessary to change the methods that call the REST API.

New format for the study: In this case it is necessary to change all the objects used, in which implies a deep change on uEDC mobile.

6.2 BLOOD PRESSURE TRACKER

The tests made on this application aimed at verifying the interoperability between the medical sensors and mobile devices. The medical sensor used was Bluetooth Blood Pressure Monitor BP792IT Omron. The Antidote has an agent that emulates a medical sensor but to guarantee the true interoperability we chose the physical sensor as the main medical sensor to use in tests.

As shown in figure 6.5, all the devices have the Blood Pressure Tracker installed and it is possible retrieve the data from medical sensor without any complications. The process used to test the interoperability was the following:

- Open the Blood Pressure Tracker;
- Add a new record using the medical device;
- Pair the two devices via Bluetooth, returning to the application;
- Press a button on medical device to send the data;
- After a few seconds, the data appears on mobile device;
- Save the data on mobile device.

This test was made ten times on each mobile device, in which the result was a 100% success rate, but for example if the user skips the pairing between the two devices, the Blood

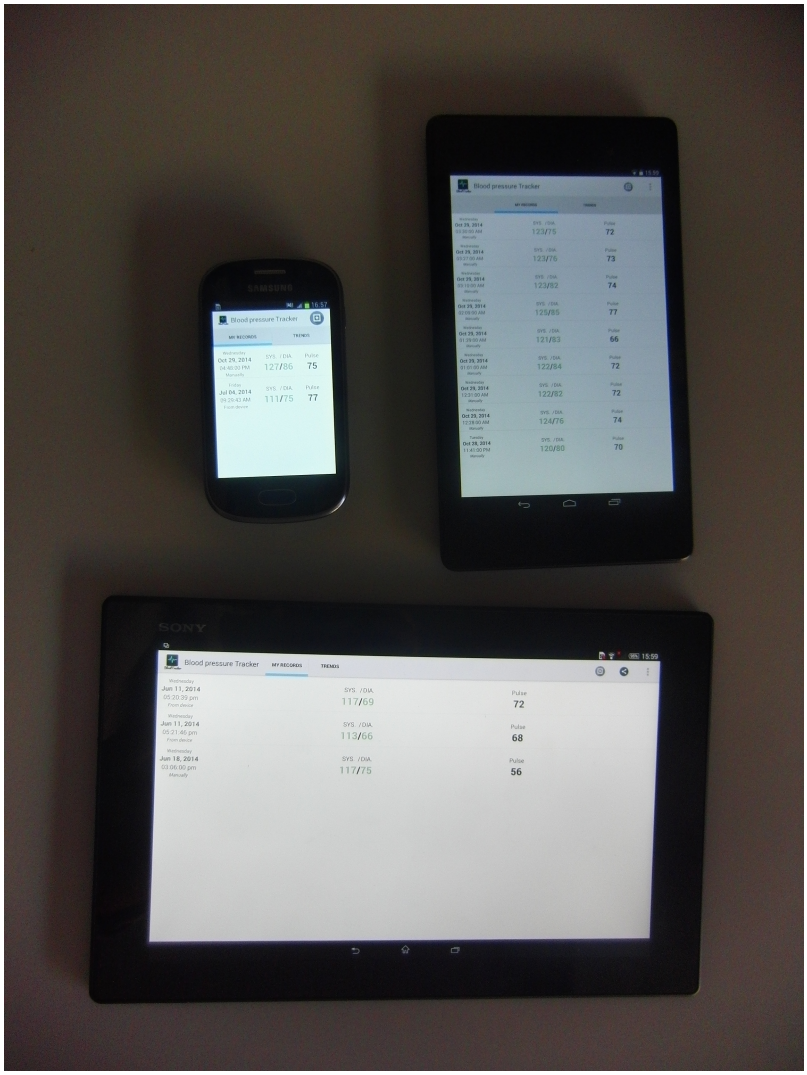


Figure 6.5: Smartphones and tablets used on tests

Pressure Tracker only receives medical data. The information about the sensor is only sent on first time. This behaviour is typical of the standard ISO/IEEE 11073 [22].

CHAPTER 7

Conclusion and future work

In this work we developed the uEDC Mobile system, which supports the integration of wearable sensors/medical devices with Android for clinical studies applications. This solution relies on the ISO/IEEE 11073 standard to communicate with compliant medical devices.

The uEDC Mobile can retrieve data from medical devices using Bluetooth and save to a clinical studies backend (in this case, the uEDC System by iUZ Technologies), avoiding problems with manual data entry. This solution can contribute to ambulatory participation in studies, avoiding the inconvenience of visiting a clinic to collect physiologic metrics.

The uEDC Mobile is a native Android OS application that was tested in different models and against several medical devices. The medical devices used to test the interoperability were the agent available on Antidote library and the Bluetooth Blood Pressure Monitor BP792IT Omron. The Antidote agent sends data associated to a simulated oximeter monitor. The Omron blood pressure monitor was used as the principal sensor on tests since it provides physical sensors, closer to real scenarios.

As proof of concept, we created a complimentary application, the Blood Pressure Tracker, that demonstrates the use of ISO/IEEE 11073 to link medical sensors and smartphones for blood pressure tracking. This application is published on the Android applications market and can be tested by the community.

As future work, the uEDC Mobile could support more subtypes of the ISO/IEEE 11073 standard since now it is configured only for Blood Pressure Monitors and Oximeter Monitors.

Another improvement that can be made is the use of native notifications, to inform the user about some changes on the study, to recall the time to retrieve a new sample, or inform that a new study where the patient can participate is available. Concerning usability tests, some of the participants suggested the use of demonstrative videos to help the user.

We successfully integrated ISO/IEEE 11073 in clinical studies workflows, leveraging common smartphone devices to extend an existing clinical studies digital platform.

References

[1] *Clinical Trials*, eng. [Online]. Available: http://www.nlm.nih.gov/medlineplus/clinicaltrials.html (visited on 10/06/2014).

[2] R. Kunz and A. D. Oxman, "The unpredictability paradox: review of empirical comparisons of randomised and non-randomised clinical trials", *BMJ*, vol. 317, no. 7167, pp. 1185–1190, Oct. 1998, ISSN: 0959-8138. DOI: 10.1136/bmj.317.7167.1185.

[3] E. L. Eisenstein, R. Collins, B. S. Cracknell, O. Podesta, E. D. Reid, P. Sandercock, Y. Shakhov, M. L. Terrin, M. A. Sellers, R. M. Califf, C. B. Granger, and R. Diaz, "Sensible approaches for reducing clinical trial costs.", *Clinical trials (London, England)*, vol. 5, no. 1, pp. 75–84, Jan. 2008, ISSN: 1740-7745. DOI: 10.1177/1740774507087551.

[4] C. López-Carrero, E. Arriaza, E. Bolaños, A. Ciudad, M. Municio, J. Ramos, and W. Hesen, "Internet in clinical research based on a pilot experience.", *Contemporary clinical trials*, vol. 26, no. 2, pp. 234–43, Apr. 2005, ISSN: 1551-7144. DOI: 10.1016/j.cct.2004.11.017.

[5] *The year for mobile-enabled clinical trials: 2013 or 2030? | mobihealthnews*. [Online]. Available: http://mobihealthnews.com/19158/the-year-for-mobile-enabled-clinical-trials-2013-or-2030/ (visited on 10/06/2014).

[6] "WHO | Welcome to the WHO ICTRP", [Online]. Available: http://www.who.int/ictrp/en/ (visited on 10/24/2014).

[7] I. Korkontzelos, T. Mu, A. Restificar, and S. Ananiadou, "Text mining for efficient search and assisted creation of clinical trials", New York, New York, USA: ACM Press, Oct. 2011, p. 43, ISBN: 9781450309608. DOI: 10.1145/2064696.2064706.

[8] *ClinicalTrials.gov*. [Online]. Available: http://clinicaltrials.gov/ (visited on 11/14/2014).

[9] T. Wang, X. Xue, and B. Huang, "Clinical trial knowledge management system using ontology", *2012 24th Chinese Control and Decision Conference (CCDC)*, pp. 3202–3207, May 2012. DOI: 10.1109/CCDC.2012.6244506.

[10] D. Kennedy, "Crf Designer", in *Crf Designer*, Canary Publications, ISBN: 0-9531174-7-2.

[11] J. Shah, D. Rajgor, S. Pradhan, M. McCready, A. Zaveri, and R. Pietrobon, "Electronic data capture for registries and clinical trials in orthopaedic surgery: open source versus commercial systems.", *Clinical orthopaedics and related research*, vol. 468, no. 10, pp. 2664–71, Oct. 2010, ISSN: 1528-1132. DOI: 10.1007/s11999-010-1469-3.

[12] I. Forte Research Systems, "An Introduction to Electronic Data Capture Software Learn the basics to find the right", [Online]. Available: http://info.forteresearch.com/an-introduction-to-electronic-data-capture-software (visited on 10/23/2014).

[13] P. Daponte, L. De Vito, F. Picariello, and M. Riccio, "State of the art and future developments of measurement applications on smartphones", *Measurement*, vol. 46, no. 9, pp. 3291–3307, Nov. 2013, ISSN: 02632241. DOI: 10.1016/j.measurement.2013.05.006.

[14] T. J. Betjeman, S. E. Soghoian, and M. P. Foran, *MHealth in sub-Saharan Africa*, 2013.

[15] © World Health Organization, *mHealth*, 2011. [Online]. Available: http://www.who.int/goe/publications/goe_mhealth_web.pdf (visited on 10/23/2014).

[16] K. P. M. Ms, "Mobile Health Technology Evaluation The mHealth Evidence Workshop", [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.361.6460 (visited on 10/24/2014).

[17] J.-Y. Park, G. Lee, S.-Y. Shin, J. H. Kim, H.-W. Han, T.-W. Kwon, W. S. Kim, and J. H. Lee, "Lessons learned from the development of health applications in a tertiary hospital.", *Telemedicine journal and e-health : the official journal of the American Telemedicine Association*, vol. 20, no. 3, pp. 215–22, Mar. 2014, ISSN: 1556-3669. DOI: 10.1089/tmj.2013.0192.

[18] *Gartner Says Worldwide Mobile Phone Sales Declined 1.7 Percent in 2012*. [Online]. Available: http://www.gartner.com/newsroom/id/2335616 (visited on 10/31/2014).

[19] *Gartner Says Smartphone Sales Grew 46.5 Percent in Second Quarter of 2013 and Exceeded Feature Phone Sales for First Time*. [Online]. Available: http://www.gartner.com/newsroom/id/2573415 (visited on 10/31/2014).

[20] A. Fioravanti, G. Fico, M. T. Arredondo, D. Salvi, and J. L. Villalar, "Integration of heterogeneous biomedical sensors into an ISO/IEEE 11073 compliant application.", *Conference proceedings : ... Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Annual Conference*, vol. 2010, pp. 1049–52, Jan. 2010, ISSN: 1557-170X. DOI: 10.1109/IEMBS.2010.5628086.

[21] B. R. Heydon, "Bluetooth Healthcare at Medica 2008 What is Bluetooth Low Energy Technology ?", no. 4, 2008.

[22] Signove, "Antidote : Program Guide", 2012. [Online]. Available: http://oss.signove.com/index.php/File:AntidoteProgramGuide.pdf (visited on 10/30/2014).

[23] *About Continua | Continua*. [Online]. Available: http://www.continuaalliance.org/about-continua (visited on 10/24/2014).

[24] C. T., "ISO/IEEE 11073 Standards for Medical Device Communication", *2nd Joint EMBS-BMES Conference, 2002; Houston, TX*, 2002.

[25] *ISO/IEEE 11073-10101: Nomenclature*. [Online]. Available: http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=37890 (visited on 10/29/2014).

[26] *Dhaca home*. [Online]. Available: http://www.dhaca.org.uk/ (visited on 10/24/2014).

[27] *ALIPinterop - Interoperability - Development & Test tools*. [Online]. Available: http://alipinterop.wikispaces.com/Interoperability+-+Development+&+Test+tools (visited on 10/24/2014).

[28] *TeamForge : stepstone project*. [Online]. Available: https://www.projects.openhealthtools.org/sf/projects/stepstone/ (visited on 10/24/2014).

[29] *S3 Group - Connected Consumer Technology company*. [Online]. Available: http://www.s3group.com/ (visited on 10/24/2014).

[30] *Openhealth Manager - ISO/IEEE 11073-20601 Manager implementation*. [Online]. Available: http://git.libresoft.es/OpenhealthManager/ (visited on 10/24/2014).

[31] *Antidote for Android - Wikidb*. [Online]. Available: http://oss.signove.com/index.php/Antidote_port_to_Android_ICS (visited on 10/24/2014).

[32] *Antidote in github*. [Online]. Available: https://github.com/signove/antidote (visited on 10/24/2014).

[33] I. Lamprey Networks, "OXPlib", 2009. [Online]. Available: http://www.lampreynetworks.com/assets/documents/OXPlib_whitepaper.pdf (visited on 10/24/2014).

[34] *Antidote: open-source IEEE 11073 stack - Google Groups*. [Online]. Available: https://groups.google.com/forum/#!forum/antidote-lib (visited on 10/24/2014).

[35] *Antidote port to old versions of Android*. [Online]. Available: http://oss.signove.com/index.php/Antidote_port_to_Android (visited on 10/24/2014).

[36] *Android NDK*. [Online]. Available: https://developer.android.com/tools/sdk/ndk/index.htmll (visited on 10/31/2014).

[37] *Java Oracle*. [Online]. Available: https://www.java.com/ (visited on 10/29/2014).

[38] *Android - Intents and Intent Filters*. [Online]. Available: http://developer.android.com/guide/components/intents-filters.html (visited on 10/31/2014).

[39] *ClinCapture - The Only Free Validated EDC*. [Online]. Available: http://www.clinovo.com/clincapture-features (visited on 10/27/2014).

[40] *Medrio - eClinical solutions including Cloud EDC.™*. [Online]. Available: http://m-core.medrio.com/ (visited on 10/27/2014).

[41] *Bioclinica- Global clinical trial solutions. Real-world Results*. [Online]. Available: http://www.bioclinica.com/eclinical/express-edc (visited on 10/27/2014).

[42] *RedCap - Research Electronic Data Capture*. [Online]. Available: http://project-redcap.org/ (visited on 10/27/2014).

[43] *HealthVault - Take control of your health*. [Online]. Available: https://www.healthvault.com/pt/en (visited on 10/27/2014).

[44] *Qualcomm Life*. [Online]. Available: http://www.qualcommlife.com/ (visited on 10/27/2014).

[45] *HealthVault - Take control of your health - Apps & Devices supported*. [Online]. Available: https://account.healthvault.co.uk/pt/en-US/Directory?target=Devices%5C&SearchType=ViewAll (visited on 10/27/2014).

[46] *A new health wristwatch may resuscitate Microsoft's HealthVault platform*. [Online]. Available: http://venturebeat.com/2014/06/21/a-new-health-wristwatch-may-resuscitate-microsofts-healthvault-platform/ (visited on 10/27/2014).

[47] l. M. Fowler, *UML Distilled: A Brief Guide to the Standard Object Modeling Language*. Addison-Wesley Longman Publishing Co., Inc., 2003, ISBN: 0321193687.

[48] U. Food and D. Administration, "Screening tests prior to study enrollment - information sheet guidance for institutional review boards and clinical investigators", [Online]. Available: http://www.fda.gov/RegulatoryInformation/Guidances/ucm126430.htm (visited on 10/25/2014).

[49] *QRCode.com, DENSO WAVE, the inventor of QRCode*. [Online]. Available: http://www.qrcode.com/en/index.html (visited on 10/28/2014).

[50] *iUZ Technologies*. [Online]. Available: http://www.iuz.pt/ (visited on 10/28/2014).

[51] *SimCrest - What is 3 tier Architecture: and Why do You need it*. [Online]. Available: http://blog.simcrest.com/what-is-3-tier-architecture-and-why-do-you-need-it/ (visited on 10/30/2014).

[52] Jioskea.net, "Networking - 3-Tier Client / Server Architecture", no. June, 2014. [Online]. Available: http://en.kioskea.net/contents/151-networking-3-tier-client-server-architecture (visited on 10/28/2014).

[53] *A multi-tier architecture for building RESTful Web services*, 2009. [Online]. Available: http://www.ibm.com/developerworks/library/wa-aj-multitier/ (visited on 10/29/2014).

[54] R. T. Fielding and R. N. Taylor, "Principled design of the modern Web architecture", *ACM Transactions on Internet Technology*, vol. 2, no. 2, pp. 115–150, May 2002, ISSN: 15335399. DOI: 10.1145/514183.514185.

[55] *Json*. [Online]. Available: http://www.json.org/ (visited on 10/29/2014).

[56] *XML Technology*. [Online]. Available: http://www.w3.org/standards/xml/ (visited on 10/29/2014).

[57] *SQLite*. [Online]. Available: http://www.sqlite.org/ (visited on 10/29/2014).

[58] *Android - Data Storage*. [Online]. Available: http://developer.android.com/training/basics/data-storage/shared-preferences.html (visited on 10/31/2014).

[59] *Google-Gson*. [Online]. Available: `https://code.google.com/p/google-gson/` (visited on 10/28/2014).

[60] *Transmitting Network Data*. [Online]. Available: `http://developer.android.com/training/volley/index.html` (visited on 10/28/2014).

[61] *Google App Engine: Platform as a Service*. [Online]. Available: `https://cloud.google.com/appengine/docs` (visited on 10/28/2014).

[62] *Measuring Usability with the USE Questionnaire*. [Online]. Available: `http://www.stcsig.org/usability/newsletter/0110_measuring_with_use.html` (visited on 10/29/2014).