Katharina Maria Duregger, MSc

Mobile and Near Field Communication (NFC) Technology for Electronic Patient-Reported Outcomes in Paediatric Oncology

MASTER'S THESIS

to achieve the university degree of
Diplom-Ingenieurin

Master's degree programme: Biomedical Engineering

submitted to
Graz University of Technology

Supervisor
Univ.-Doz. Dipl.-Ing. Dr.techn. Günter Schreier, MSc

Institute for Knowledge Discovery

Graz, January 2016

This master's thesis has been conducted
in cooperation with:

AIT Austrian Institute of Technology GmbH
Digital Safety & Security Department

Supervisor
Dipl.-Ing. Dr. Dieter Hayn

Reininghausstraße 13/1
8020 Graz
Austria

EIDESSTAATLICHE ERKLÄRUNG

AFFIDAVIT

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt, und die den benutzten Quellen wörtlich und inhaltlich entnommenen Stellen als solche kenntlich gemacht habe. Das in TUGRAZonline hochgeladene Textdokument ist mit der vorliegenden Masterarbeit identisch.

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitly marked all material which has been quoted either literally or by content from the used sources. The text document uploaded to TUGRAZonline is identical to the present master thesis.

Datum / Date

Unterschrift / Signature

Introduction: Usually children suffering from cancer are registered and treated within clinical trials being part of medical research networks. Electronic Patient-Reported Outcome (ePRO) represents a valuable source of information when improving treatment and survival of children suffering from neuroblastoma or other rare diseases. Aim of this thesis was to develop a concept and to implement a prototype for introducing ePRO into the neuroblastoma research network by applying mobile technology and Near Field Communication (NFC).

Methods: A mobile application was developed for Android and iOS devices using the cross-platform tool Apache Cordova. This application was intended to be used by physicians for registering patients in the neuroblastoma research network by creating pseudonyms according to the patient identity management concept within the European Network for Cancer Research in Children and Adolescents (ENCCA) and for providing patients with an ID card and a PIN for authentication when submitting telemonitoring data to the Electronic Data Capture (EDC) system OpenClinica. The mobile application of the already existing telemonitoring system for capturing measurement values from NFC-enabled devices was extended by a Simple Object Access Protocol (SOAP) interface. The patient card ID could contactlessly be read via NFC or QR Code.

Results: The *EUPID Mobile* application for physicians offered three functionalities. New patients could be registered in defined contexts by entering their identity data and obtaining a context-specific pseudonym. After linking a patient ID card to a generated pseudonym and obtaining a PIN, patients could be equipped with these credentials and were then able to transmit nine different health parameters and toxicities to OpenClinica. On Android devices, generated pseudonyms could be stored on NFC tags in order to label biosamples or patient-related documents. The *EUPID Mobile* application was successfully presented to the Paediatric Oncology Community (POC) at the ENCCA Closing Conference in 2015.

Conclusion: ePRO represents an important data source in cancer research. In this work a concept for empowering neuroblastoma patients to transmit telemonitoring data to the neuroblastoma research network was developed and prototypically implemented. The next step will be the implementation of the back-end features. Moreover, the ID card provided to the patients might in future play an important role in patient follow-up.

Einleitung: An Krebs erkankte Kinder werden üblicherweise in medizinischen Forschungsnetzwerken registriert und im Rahmen klinischer Studien behandelt. Electronic Patient-Reported Outcome (ePRO) stellt eine wertvolle Informationsquelle bei der Verbesserung der Behandlung von Neuroblastomen und anderer seltenen Erkrankungen dar. Ziel dieser Arbeit war es, ein Konzept zur Miteinbeziehung von ePRO in die Neuroblastom-Forschung mithilfe von Mobiltechnologie und Near Field Communication (NFC) zu entwickeln und prototypisch zu implementieren.

Methoden: Mithilfe von Apache Cordova zur Entwicklung von betriebssystemunabhängigen Anwendungen wurde eine mobile Applikation für Android- und iOS-Geräte entwickelt, welche mitunter zur Registrierung von Patienten im Neuroblastom-Forschungsnetzwerk diente. Dabei wurden, gemäß dem im European Network for Cancer Research in Children and Adolescents (ENCCA) verwendeten Konzept zur Patientenidentifizierung, kontextspezifische Pseudonyme erstellt. Mithilfe dieser Applikation konnte den Patienten eine ID-Karte und ein PIN zur Authentifizierung bei der Übertragung von Telemonitoring-Daten an das Electronic Data Capture (EDC)-System OpenClinica zur Verfügung gestellt werden. Die mobile Applikation des bestehenden, auf NFC basierenden, Telemonitoring-Systems wurde um eine Simple Object Access Protocol (SOAP)-Schnittstelle erweitert. Das Auslesen der ID-Karte des Systems war kontaktlos über NFC oder QR Code möglich.

Ergebnisse: Die mobile Applikation *EUPID Mobile* stellte drei Funktionen zur Verfügung. Ärzte konnten damit neue Patienten in definierten Kontexten durch Eingabe der Identitätsdaten registrieren und Patienten eine ID-Karte samt zugehörigen PIN für die Übertragung von neun verschiedenen Gesundheitsparametern und Toxizitäten an OpenClinica zur Verfügung stellen. Auf Android-Geräten war es möglich, Pseudonyme direkt auf NFC Tags, zur Beschriftung von beispielsweise Blutproben oder Dokumente, zu speichern. Die Applikation *EUPID Mobile* wurde bei der ENCCA Closing Conference 2015 erfolgreich der Paediatric Oncology Community (POC) präsentiert.

Konklusion: ePRO stellt eine wichtige Informationsquelle in der Erforschung von Krebserkrankungen dar. Im Rahmen dieser Arbeit wurde ein Konzept entwickelt, welches es Neuroblastom-Patienten ermöglicht, Telemonitoring-Daten an das in der Neuroblastom-Forschung verwendete Electronic Data Capture (EDC)-System OpenClinica zu übertragen. Ebenso wurde das Konzept prototypisch umgesetzt. Weiterführende Arbeit beinhaltet die Implementierung der Backend-Services. Die ID-Karte, als zentrales Element, könnte in Zukunft eine wichtige Rolle im Follow-up von Patienten spielen.

"... Wir könnten auch nicht schlafen, wenn du nicht noch mal kämst. Und uns, bevor wir träumen, in deine Arme nähmst. ..."

Danke, Mama

Poem by Eva Rechlin

Contents

List of Figures

List of Tables

List of Listings

List of Abbreviations

ABCD-4-E	Advanced Biomedical Collaboration Domain for ENCCA
ADAL	Active Directory Authentication Library
AIT	AIT Austrian Institute of Technology
AJAX	Asynchronous JavaScript and XML
CCRI	Children's Cancer Research Institute
CDISC	Clinical Data Interchange Standards Consortium
CDM	Clinical Data Management
CRF	Case Report Form
CRP	C-Reactive Protein
CSS	Cascading Style Sheets
EDC	Electronic Data Capture
EHR	Electronic Health Record
ENCCA	European Network for Cancer Research in Children and Adolescents
ePRO	Electronic Patient-Reported Outcome
ESQH	European Society for Quality in Healthcare
EUPID	ENCCA Unified Patient Identifier
FDA	Food and Drug Administration
GUI	Graphical User Interface
HTML5	Hypertext Markup Language, revision 5
HTTPS	HyperText Transfer Protocol Secure
IDAT	Identity Data
IDATcr	Encrypted Identity Data

IDATph ..... Phonetically hashed Identity Data

IrDA ..... Infrared Data Association

JSON ..... JavaScript Object Notation

MAAD ..... Microsoft Azure Active Directory

NDEF ..... NFC Data Exchange Format

ODM ..... Operational Data Model

OMS/DES ..... Opsoclonus Myoclonus Syndrome/Dancing Eye Syndrome

PRO ..... Patient-Reported Outcome

PSN ..... Pseudonym

PW ..... Password

QR Code ..... Quick Response Code

REST ..... Representational State Transfer

RFID ..... Radio Frequency Identification

RTD ..... Record Type Definition

SaaS ..... Software as a Service

SIOPEN ..... Society of Paediatric Oncology Europe Neuroblastoma

SOAP ..... Simple Object Access Protocol

TAN ..... Touch Area Network

TTP ..... Trusted Third Party

UN ..... User name

URI ..... Uniform Resource Identifiers

VI ..... Virtual Institute

WAAD ..... Windows Azure Active Directory

WBC ..... White Blood Cell Count

XML ..... Extensible Markup Language

XSLT ..... Extensible Stylesheet Language Transformations

ÖK ..... Austrian Childhood Cancer Organisation

Chapter 1

Introduction

Chapter 1 gives a brief introduction into the neuroblastoma disease and a description of two important research networks in paediatric oncology. Furthermore, the terms 'horizontal integration' and 'vertical integration' of clinical data are defined. The importance of actively involving the patients in capturing outcome data within clinical trials is highlighted and the research question of this thesis is finally presented.

The patient identity management and dataset standardization concepts used within the European Network for Cancer Research in Children and Adolescents are explained in Chapter 2. Quick Response Codes and Near Field Communication are described as state-of-the-art technologies for identification purposes and available systems for capturing electronic research data from clinicians or patients are introduced.

In Chapter 3 approaches, technologies and resources used for implementation are described.

Chapter 4 contains a detailed description of the overall implementation concept. The user interfaces of the two resulting mobile applications *EUPID Mobile* and *MoKi* are introduced and the transmission of data to the back-end system is demonstrated for the health parameter 'wellbeing'.

Chapter 5 contains general considerations, describes problems occuring during the development process and delivers some argumentation on why the selected methods were used. Concluding words can be found in Chapter 6.

Appendix B and Appendix C contain additional information and an instruction on how to set up the environment for developing and debugging cross-platform applications.

1.1 Neuroblastoma

In Austria, approximately 180 children up to the age of 14 are diagnosed with cancer every year. Those represent 1% of all cancer patients. In most cases treatment is successful, approximately 80% of the children survive.[1]

Figure 1.1 shows the distribution of cancer types diagnosed between 2002 and 2011 in Austria. With an annual incidence rate of 1.8 per million the neuroblastoma is the most common embryonal cancer in Europe [2].

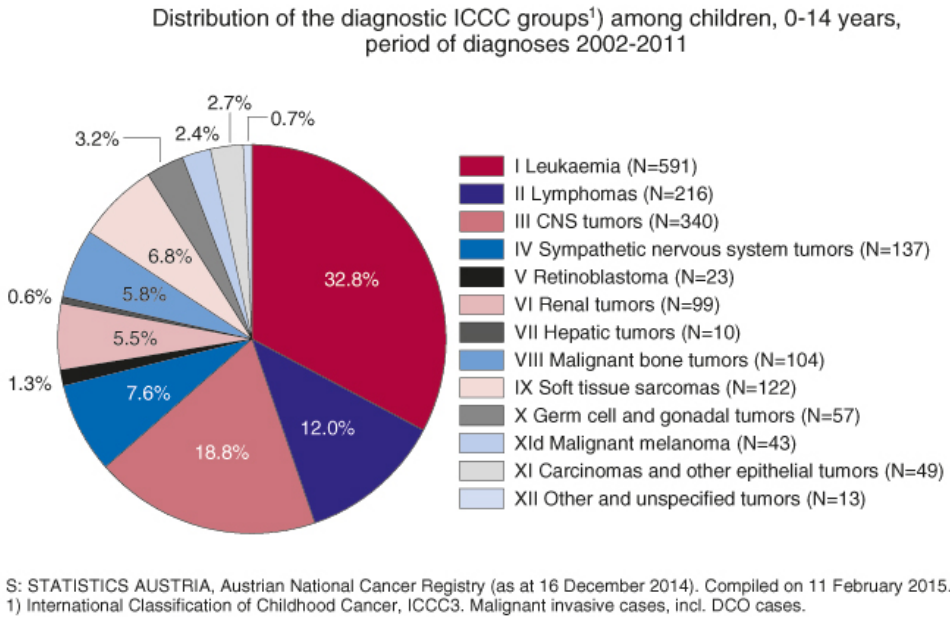

Figure 1.1: Distribution of cancer types concerning children in Austria [1].

Neuroblastomas develop from nerve cells – sometimes due to inherited gene alterations – and are often found near the spinal cord or the adrenal gland (see Figure 1.2). Among others, symptoms of neuroblastoma include flu symptoms, swollen or paralysed body parts or uncontrolled muscle movements. Whether treatment is successful depends on several parameters like the stage of the cancer, the child's age or the tumour histology.[3]

Depending on the treatment plan, the debilitating therapy involves chemotherapy, radiotherapy, surgery or immunotherapy. In addition, patients might suffer from home sickness since intensive treatment phases require multiple and long lasting hospitalizations.[5]

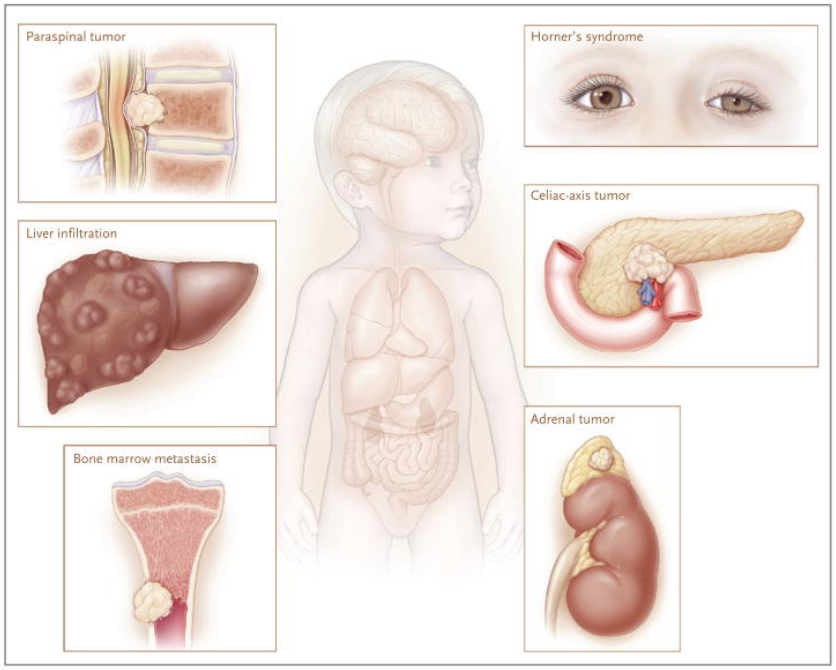

Figure 1.2: Neuroblastoma manifestation along the sympathetic nervous system [4].

Anti-cancer drugs administered during chemotherapy might cause severe toxicities like hair loss, nausea, skin alterations, increased risk of infection due to low blood cell counts, pain or irreversible organ damage [6].

Parents often have to decide within a short period of time after diagnosis whether their child should be enrolled in a clinical trial for treatment [7]. By August 2015, five studies targeting different risk stages of the disease or certain symptoms were conducted by the International Society of Paediatric Oncology Europe Neuroblastoma (SIOPEN) [8]:

- High Risk Neuroblastoma study [9]
- European Low and Intermediate Risk Neuroblastoma study [10]
- Opsoclonus Myoclonus Syndrome/Dancing Eye Syndrome (OMS/DES) in Children With and Without Neuroblastoma [11]
- Long Term Continuous Infusion study [12]
- Study on treatment of relapsed or refractory neuroblastoma [13]

Patient data gathered in clinical trials are important information sources for investigating rare diseases and therefore represent an essential element of medical research networks.

1.2 Clinical Research Networks in Rare Diseases

The neuroblastoma is one example for a rare disease[1]. Due to a low number of patients in single research facilities, multicentric clinical trials are required. Thus, standardized treatment as well as transnational collaboration and knowledge exchange whithin clinical research networks are ensured.[7]

1.2.1 SIOPEN-R-NET

The SIOPEN-R-NET is the research network used to coordinate multinational prospective studies aiming to standardize neuroblastoma treatment throughout Europe. Among other things, the Internet-based research environment is used for information sharing and Electronic Data Capture (EDC) in clinical trials.[15]

With the main objective to improve the treatment of patients suffering from neuroblastoma, the SIOPEN-R-NET enables a close collaboration of various disciplines involved during therapy (see Figure 1.3).

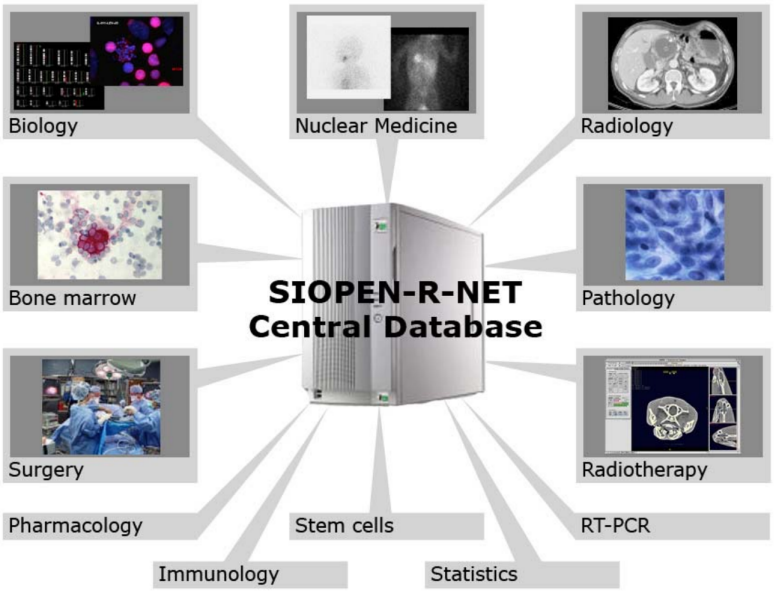

Figure 1.3: Connecting different disciplines in neuroblastoma treatment [16].

[1] According to the European Commission a rare disease is characterized by an incidence rate lower than 5 people per 10 000 [14].

1.2.2 European Network for Cancer Research in Children and Adolescents

The European Network for Cancer Research in Children and Adolescents (ENCCA) project[2] aims to connect clinical trial networks in paediatric and adolescent oncology all over Europe in order to improve the quality of life of patients in the long run [17].

Currently eleven European countries with a total of 34 research facilities are partners of ENCCA [18]. Austria is represented by the following institutions:

- Children's Cancer Research Institute (CCRI) [19]
- European Society for Quality in Healthcare (ESQH) [20]
- AIT Austrian Institute of Technology (AIT) [21]
- Austrian Childhood Cancer Organisation (Österreichische Kinder-Krebs-Hilfe, ÖK) [22]

One field of work of ENCCA is to establish a Virtual Institute (VI), which can be used by researchers to share expertise or to access tools for research purposes. Figure 1.4 shows the aims of ENCCA VI.

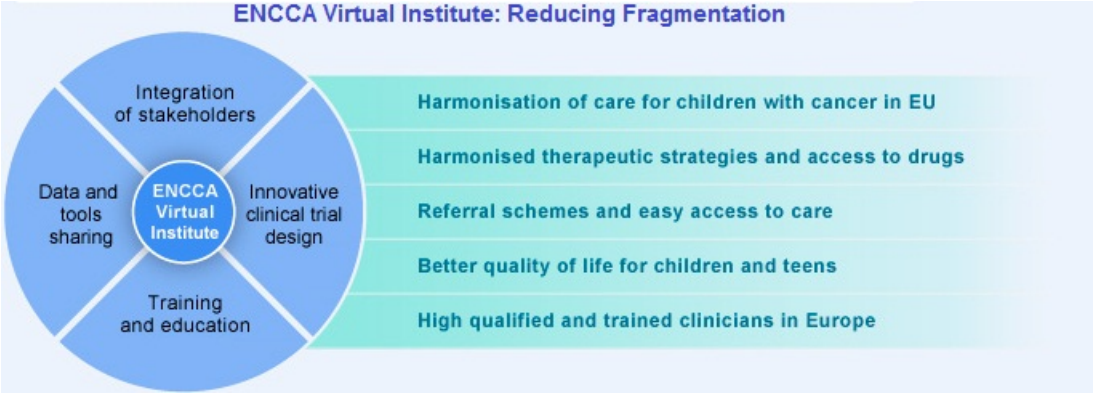

Figure 1.4: Priorities and aims of the virtual institute of the ENCCA project [17].

The concept of the VI is described in [23]. With the VI, data collected from different sources of various research communities shall be provided for cancer research [23].

[2]The ENCCA project is funded by the European Commission (FP7-HEALTH-F2-2011 Grant no. 261474).

1.3 Connecting Clinical Data

Especially in rare diseases – such as paediatric cancer – reusing and pooling research data from single sources might be helpful in answering further research questions. Therefore, the VI of ENCCA, i.e. the Advanced Biomedical Collaboration Domain for ENCCA (ABCD-4-E), provides the basis for connecting patient-related data from different sources – called contexts[3] – using pseudonymization.[24]

When connecting data for research purposes, two ways of proceeding are possible (see Figure 1.5). Horizontal integration means that either patient data from different studies within a specific tumour domain are connected (e.g. studies in neuroblastoma research) or patient data from different tumour domains (e.g. neuroblastoma study and Wilms tumour[4] study) are provided. In case of a vertical integration study data is connected with other domains within cancer treatment (e.g. neuroblastoma study and biobank data).[27]

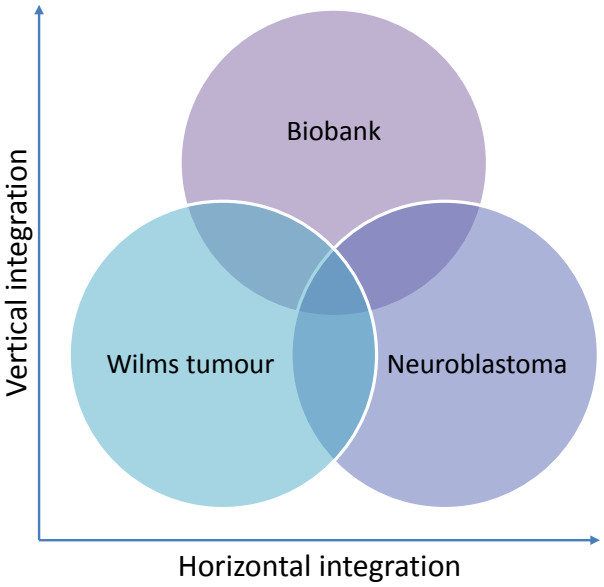

Figure 1.5: Horizontal and vertical integration of research data. Adapted from [27].

[3] A context can be described as an area, for which a patient signed an informed consent.

[4] The Wilms tumour is also called nephroblastoma and is – as neuroblastomas might be – located in the kidneys [25, 26]. Tumours of different types might occur as secondary malignancy after the treatment of the primary tumour. Also a wrong initial diagnosis might necessitate the change of a treatment plan and therefore enrolment in studies within other tumour domains.

1.4 Patient-Reported Outcome

> *"A PRO* (i.e. Patient-Reported Outcome) *is any report of the status of a patient's health condition that comes directly from the patient, without interpretation of the patient's response by a clinician or anyone else."* [28]

Subjective health parameters as well as parameters only assessable in the domestic environment of patients should be reported by the patients themselves [29] in order to obtain a more comprehensive and more realistic set of patient data and to consider the patient's individual perception of his or her condition [30]. Compared to symptom assessment performed by clinicians, patients detect health deteriorations earlier and stage the severity of adverse events with higher accuracy [29]. Further, gathering PRO helps to assess the patient's compliance and increases patient empowerment [31].

In the context of clinical trials PRO might support the proof of efficacy of new treatment methods [29, 30]. Therefore, the Food and Drug Administration (FDA) published a guideline for creating PRO questionnaires used in clinical trials in 2009 [28].

The steadily growing digitalization and interconnection helps to overcome barriers of introducing PRO in clinical research caused by questionnaires and similar paper-based survey tools. Electronic Patient-Reported Outcome (ePRO) tools use the Internet and modern communication technology for capturing vital signs and symptoms assessed by patients using mobile devices [29]. Among others, ePRO offers the following advantages compared to paper-based PRO [29, 30, 31]:

- decreased administrative burden
- data is immediately available
- automated data evaluation and pattern detection
- data is not biased due to bad handwriting
- easier tracking of health parameters over time
- automatic trigger of alarms
- enrichment of the patient's Electronic Health Record (EHR) [32]
- increased patient empowerment

1.5 Research Question

As described in Chapter 1.4, introducing ePRO might add a valuable source of information to cancer research, especially for rare diseases such as the neuroblastoma.

Therefore, the aim of this thesis was to design a concept and develop a prototype for integrating ePRO into the existing neuroblastoma research network for research purposes.

In order to increase usability and to reduce the amount of additional workload for the people involved to a minimum, mobile devices and Near Field Communication (NFC) technology should be used for authentication and data transmission whenever possible and adequate. Two major tasks evolve from this research question:

Telemonitoring System
Patients and their parents, respectively, should be provided with an appropriate point-of-care telemonitoring system for capturing and transmitting vital parameters and possibly occurring toxicities. The system should be attractive for children and easy to use. The ePRO should be transmitted to the currently used EDC system to also allow clinicians insight into their patients' reported data.

It was not the aim of this thesis to implement time schedules for patients for transmitting vital parameters, reminders or alerts in case of missing or deteriorating health parameters.

Patient Identification
A process for authorization and authentication of patients to report health parameters to the research network should be developed. The first priority of this thesis was to develop and to prototypically implement the concept on the front-end side.

The aim here was not to fully implement all necessary back-end features and systems for the realization of the whole workflow. Due to the complexity of the existing research network and the sensitivity of the processed data this is beyond the scope of this thesis.

Chapter 2

State of the Art

In this chapter recent developments within research networks in paediatric oncology, which are relevant for this thesis and as a basis for implementation, are introduced.

2.1 Standardization of Patient Datasets within ENCCA

To perform analyses in a structured way, standardizing the patient datasets from different sources, not only in terms of syntax but also semantics, is inevitable [33]. Therefore, a web application [24, 27] has previously been developed, which is available on the ABCD-4-E platform and can be used to process Operational Data Model (ODM)[1] datasets. As shown in Figure 2.1, the datasets are converted into a standardized core dataset using Extensible Stylesheet Language Transformations (XSLT). Further information can be found in Hochedlinger et al. [24].

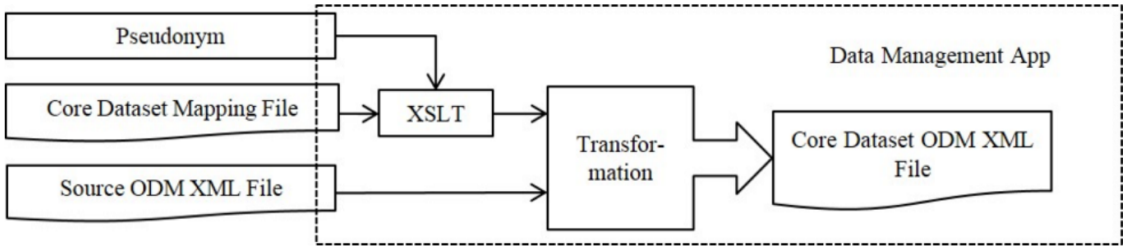

Figure 2.1: Concept of the data management web application [24].

[1]Further information is provided on [34].

2.2 Patient Identity Management within ENCCA

According to the Directive 95/46/EC of the European Parliament anonymization means that data are processed *"in such a way that the data subject is no longer identifiable"* and reidentification *"is no longer possible"* [35]. When connecting a patient's data from different contexts in medical research, the datasets must at least be pseudonymized [23]. Pseudonymization must not be equated to anonymization since pseudonymization only *"reduces the linkability of a dataset with the original identity of a data subject"* and is done by *"replacing one attribute (typically a unique attribute) in a record by another. The natural person is therefore still likely to be identified indirectly"* [36].

Wiesenauer et al. [33] summarized several requirements concerning patient identity and data protection when merging patient data for research purposes:

- informed consent of the patients on their rights to have insight into their data
- transparent and reasonable purposes for data merge
- an independent institution must be entrusted with anonymizing/pseudonymizing
- minimized possibility to indirectly identify patients within external contexts

Patients might directly benefit from new findings arising from cancer research based on merging their data, i.e. their current treatment might be improved. Therefore, anonymizing patients would not be expedient and proper pseudonymization should be used instead [33]. For patient identity management within ENCCA Nitzlnader et al. [37] developed a concept based on five main requirements outlined in Table 2.1.

Descriptions of the Main Requirements	
R1	Prevention of duplicate registration of one and the same patient.
R2	Preserve the possibility to re-identify subjects by a trusted third party. [...]
R3	Different pseudonyms should be used also for one and the same patient for different contexts. [...]
R4	Avoid creating a transparent universal patient ID that would impose re-identification threats by the potential availability of an increasing number of linked datasets.
R5	The patient identity management concept has to be feasible in a distributed computing environment.

Table 2.1: Main requirements for a patient identity management concept within ENCCA. Adapted from [37].

Figure 2.2 shows the patient identity management concept. The central element of the concept is the ENCCA Unified Patient Identifier (EUPID), which is unique for every patient and inaccessible. If patients are registered in a certain context using their Identity Data (IDAT), a context-specific Patient Identifier (PID) is created. By applying a one-way hashing algorithm, a pseudonym (PSN) is created from the PID. This PSN is then assigned to the patient's EUPID, which is generated when the patient is registered for the first time.[37]

In order to enable reidentification if necessary, the IDAT are encrypted using a public key (IDATcr) and are assigned to the EUPID. Only a trusted third party (TTP) holds the corresponding private key for decryption.[37]

To prevent the creation of more than one EUPID per patient as best as possible, the IDAT are additionally hashed (IDATph) using a hashing algorithm [38] delivering phonetic codes. The characteristic of this algorithm is that phonetically similar strings – which apply to spelling mistakes in the patient's IDAT – result in an identical hash.[37]

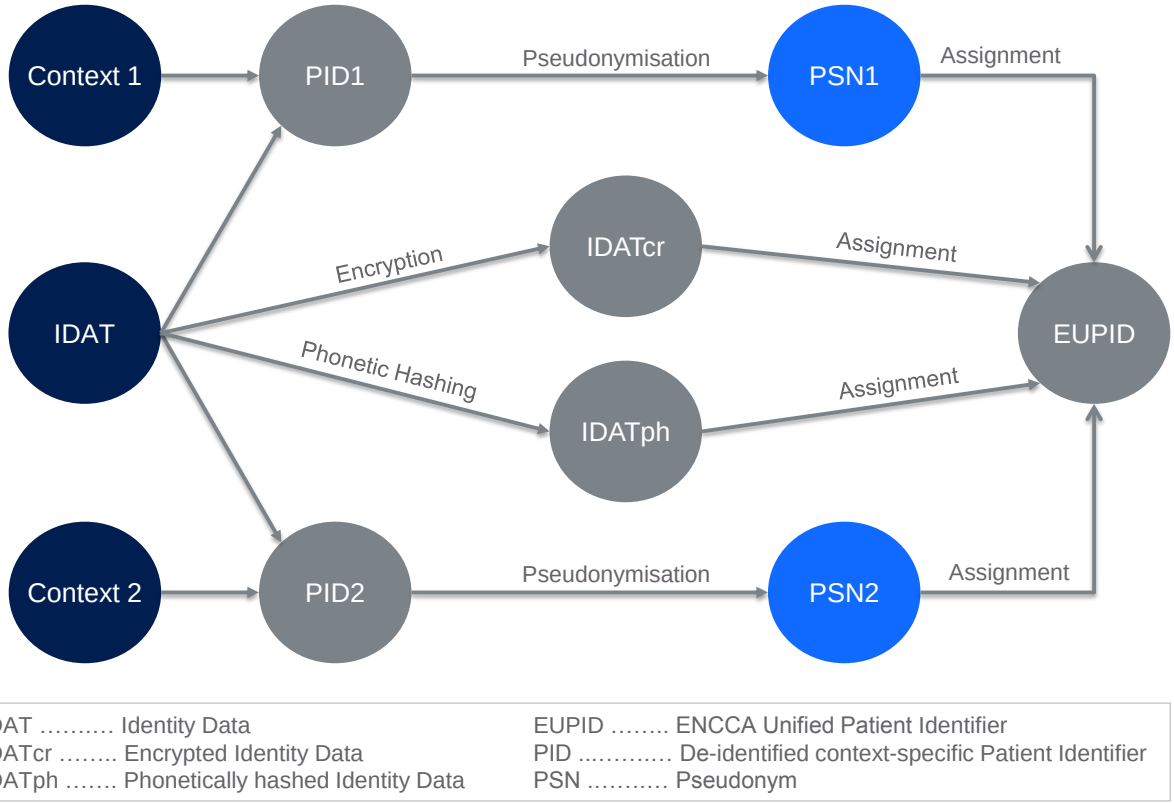

Figure 2.2: Patient identity management concept within ENCCA. Adapted from [37].

2.3 Quick Response Codes

In 1994, Quick Response (QR) Codes were introduced by Denso Wave Incorporated (Kariya, Japan) as an improvement of the ordinary barcode. QR Codes are two-dimensional codes and are nowadays used in many fields of application, e.g. for storing information or merchandising purposes, in production management or to quickly access websites by scanning the QR Code with a mobile phone's camera. Denso Wave Incorporated decided not to exercise their patent rights and in 2002 the QR Code was made an ISO standard (ISO/IEC 18004).[39, 40]

Figure 2.3 shows the types and structure of information stored in a QR Code. The pattern of this example contains a link to the main page of Wikipedia Free Encyclopedia (Wikimedia Foundation, San Francisco, California).

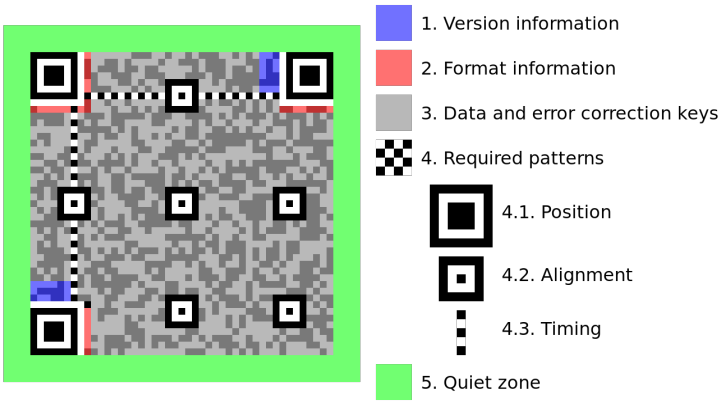

Figure 2.3: Types and structure of information stored in a QR Code [41].

Table 2.2 shows the basic model of the QR Code (Model 2) and several further developments. In contrast to the one dimensional barcode, a Model 2 QR Code can be used to store up to 4,296 alphanumeric characters depending on the symbol version and the error correction level. As shown in Figure 2.4 the symbol version is determined by the module configuration (the number of white and black squares) and ranges from 1 (21x21 squares) to 40 (177x177 squares) for Model 2 QR Codes.[42, 43]

In order to preserve readability to a certain extent in case the QR Code gets dirty or is damaged (especially in production settings), an error correction code is added to the actual data. Table 2.3 shows the four different error correction levels, which differ in the approximate amount of restorable data. The higher the restoring capability, the less amount of actual data can be stored in the QR pattern.[44]

Type	Example	Characteristics
QR Code Model 2		Improvement of the original Model 1 QR Code, the maximum version is 40
Micro QR Code		Only one orientation detecting pattern is required
iQR Code		Modules can either be square or rectangular, the maximum theoretical version is 61
SQRC		Secure QR Code, reading restricting function, used to store private data, no difference to the regular QR Code
Frame QR		Canvas area, which can be flexibly used

Table 2.2: QR Code types and characteristics. Adapted from [42].

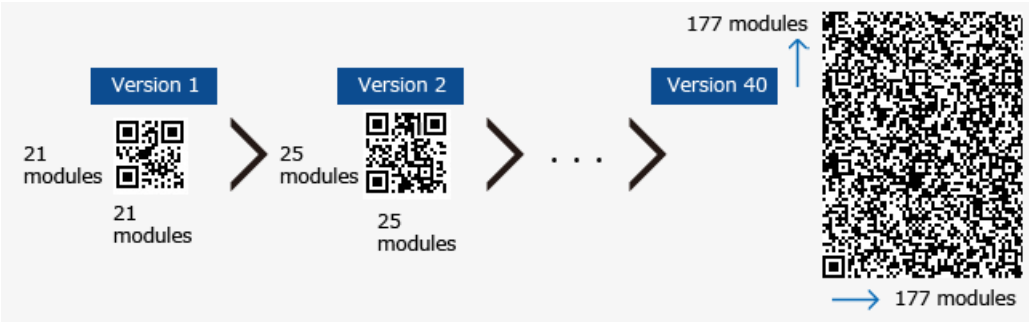

Figure 2.4: QR Code versions [43].

Level	Approximate Error Correction Capability
L	7%
M	15%
Q	25%
H	30%

Table 2.3: Error correction capability levels of QR Codes. Adapted from [44].

2.4 Near Field Communication

Near Field Communication (NFC) defines a contactless data exchange method based on Radio Frequency Identification (RFID) within short distances [45, 46]. The NFC standard was specified in ISO/IEC 18092. Table 2.4 lists several technical characteristics.

Frequency	13,56 MHz
Distance	<4 cm
Bit rates	106 kbps 212 kbps 424 kbps

Table 2.4: Technical data of Near Field Communication. Adapted from [47, 48, 49].

As shown in Figure 2.5, NFC supports three operation modes. The card emulation mode replaces smart cards and is used for ticketing or making payments. While the Peer-to-Peer mode allows active devices to exchange data, the Reader/Writer mode is used for reading the content of NFC tags or storing data.[49]

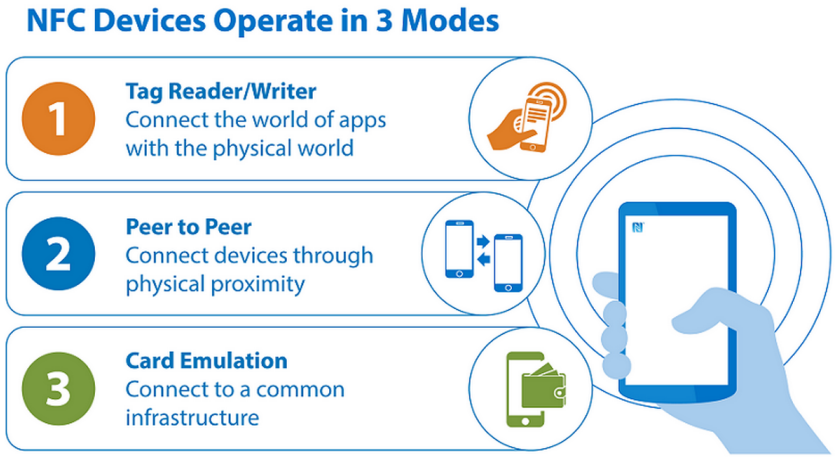

Figure 2.5: Operating modes of Near Field Communication [49].

Table 2.5 shows characteristics of the different types of NFC tags specified by the NFC Forum [50]. When exchanging data between devices and tags, the NFC Data Exchange Format (NDEF) specified by the NFC Forum is used. NDEF messages contain one or more NDEF records. Table 2.6 shows several Record Type Definitions (RTD).

NFC offers several benefits. Due to the limited range, it is suitable for communication within touch area networks (TAN). In combination with mobile phones supporting NFC

and smartphone applications it further reduces the neccessity to be able to read or write. In hospitals, NFC might improve patient management [53, 54, 55]. It has been used for creating notifications for medical appointments in the patient's mobile device [56] or for geolocalization within hospitals [57, 58].

NFC Tag	Underlying standard	Characteristics
NFC Forum Type 1	ISO/IEC 14443A	Read and re-write capability, can be configured to become read-only, memory: 96 bytes to 2 kbyte
NFC Forum Type 2	ISO/IEC 14443A	Read and re-write capability, can be configured to become read-only, memory: 48 bytes to 2 kbyte
NFC Forum Type 3	Japanese Industrial Standard X 6319-4 (also called FeliCa)	Pre-configured at manufacture to be either read and re-writeable or read-only
NFC Forum Type 4	ISO/IEC 14443	Pre-configured at manufacture to be either read and re-writeable or read-only, variable memory availability (up to 32 kbyte per service)

Table 2.5: Types and characteristics of NFC tags as defined by the NFC Forum. Adapted from [51].

NFC RTD	Usage and Examples
Text	Storing text strings in multiple languages
URI	Storing Uniform Resource Identifiers (URI)
Smart Poster	Storing URLs, SMSs or phone numbers on an NFC tag or transport them between devices, builds on Text RTD and URI RTD
Signature	Signing single or multiple NDEF records
MIME (RFC2046)	Storing text or images
Android Application	Specifying the Android application which should handle the NFC intent (open a certain application or the market if the application is not installed)

Table 2.6: NFC Record Type Definitions (NFC RTD). Adapted from [51, 52].

2.5 Comparison of QR Code and NFC

Razi et al. [55] developed particular NFC Tags for the identification of newly born babies within a hospital. Table 2.7 shows a comparison of several criteria between reading identity data via NFC and via scanning QR tags. These criteria should also be considered when developing wireless and mobile technology-based solutions for patient identification in paediatric oncology.

Criterion	QR Codes	NFC
Efficiency	Less efficient	Very efficient
Security	Less secure	Very secure, no visible data
Rewritable	No	Yes
Storage capacity	Less (3 Kb)	More (32 Kb)
Costs	Inexpensive	More expensive than QR Codes
Availability	Easily available/printable	Not easily available
Persistence	More prone to damages	Less prone to damages
Readability	Difficult to read at all angles	Easy to read at all angles
Integration	Less product integration	More product integration
Programmable	Yes	Yes

Table 2.7: Comparison of QR Codes and NFC for identification of newly born babies. Adapted from [55].

Taking into account the results of the analysis in Table 2.7, NFC should be preferred to QR Codes. With respect to usability, reading NFC tags with mobile phones is easier than reading QR Codes since NFC tags automatically start the suitable application and the reading process is faster.[55]

2.6 Alternatives to NFC

The main advantage of short range communication technologies used with mobile phones is the contactless transmission of data combined with flexibility. For a few years this combination has increasingly found its way into payment systems, which have high requirements regarding reliability and data security.[59]

Table 2.8 compares short range communication technologies currently used in mobile payment systems, which might also be used in health care settings for patient identi-

fication purposes. Table 2.9 shows application examples for several types of wireless technology.

Characteristic	**NFC**	**RFID**	**Bluetooth V2.1**	**IrDA**
Information Transmission	Coupling of magnetic field	Magnetic field	Electromagnetic radiation	Infrared light
Operating frequency	13.56 MHz	13.56 MHz	2.4 GHz	infrared
Transmission range	0.04 - 0.1 m	up to 1 m	10 - 100 m	0 - 2 m
Communication	Two way	One way	Two way	One way
Maximum data rate	424 kbps	128 kbps	2.1 Mbps	16 Mbps
Power consumption	Very low	Very low	Low	Very low
Security	Unsecured unless protected	Unsecured unless protected	Less secure	Very secure
Modes	Active - Active Active - Passiv	Reader mode Card-like mode	Active - Active	Active - Active

Table 2.8: Comparison of short range wireless communication technologies. (NFC = Near Field Communication, RFID = Radio Frequency Identification, IrDA = Infrared Data Association) Adapted from [59].

Wireless Technology	**Suitable Application Area**
RFID	Access control, inventory control, smart cards, mobile payment
NFC	Data exchange, contactless smart card, mobile payment
Bluetooth	Data exchange, electronic device remote control, payment system
RFID	Data exchange, device remote control, payment system

Table 2.9: Application examples of short range wireless communication technologies. Adapted from [59].

2.7 An ePRO System for Paediatric Cancer Patients

A front-end telemonitoring solution for gathering patient-reported outcomes of patients suffering from neuroblastomas has been developed in previous work [5, 60]. The mobile phone-based system was designed with respect to the requirements of paediatric oncology and consists of:

- smartphone LG-P700 (LG Electronics, Seoul, South Korea) with NFC antenna (AIT Austrian Institute of Technology, Vienna, Austria)
- Android application *MoKi* (AIT, Vienna, Austria)
- ID card for starting *MoKi* via NFC
- smart poster for *MoKi* (AIT, Vienna, Austria)
- blood pressure meter UA-767 Plus NFC (A & D Company, Limited, Tokyo, Japan)
- body weight scales UC-324 NFC (A & D Company, Limited, Tokyo, Japan)

The system can be used to manually capture nine different vital parameters and toxicities likely to occur during oncological treatment: wellbeing, blood pressure and heart rate, body temperature, body weight, C-Reactive Protein[2] (CRP), White Blood Cell Count (WBC), pain intensity, nausea and skin alterations (rashes or skin lesions at catheter sites). Measured values are stored in an SQLite database on the mobile phone's SD card. Figure 2.6 shows the main screen of the Android application and the screen for entering blood pressure and heart rate values.[5]

Assessment of wellbeing, nausea, pain, blood pressure and heart rate can be done by gathering measured values via NFC using the corresponding devices displayed in Figure 2.7. For the measurement of subjective parameters a specific smart poster can be used. The poster is equipped with RFID tags, each one representing a certain intensity level. The system also offers the option to take a photo using the mobile phone's built-in camera, for instance to take pictures of skin alterations. [5]

An alternative ePRO system would be OpenClinica Participate, which is described in the following section.

[2] An increased blood level of C-reactive protein indicates an inflammation somewhere in the body [61].

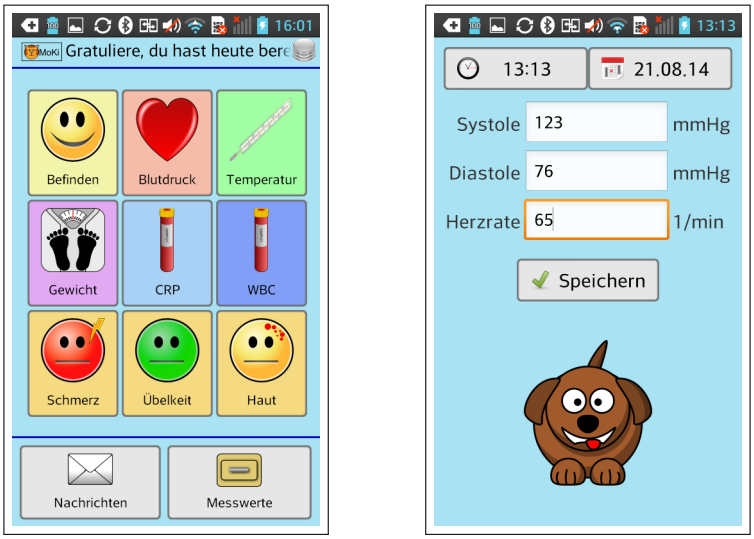

Figure 2.6: Main screen of MoKi for documentation of nine different health parameters (left) and screen for entering blood pressure and heart rate (right). Adapted from [5].

Figure 2.7: Smart poster (back), body weight scales (middle) and blood pressure meter (front) for the measurement of vital parameters and toxicities. Adapted from [5, 60].

2.8 OpenClinica for Electronic Data Capture

If properly implemented, EDC systems might have some advantages over paper-based patient data entry within clinical studies [62].

OpenClinica (OpenClinica, LLC, Waltham, Massachusetts) is an open source Clinical Data Management (CDM) system currently used in many clinical trials all over the world. The Community edition of OpenClinica is free to use, the Enterprise edition offers for instance certified software, user training or enhanced support. By November 2015 the latest version was 3.7.[63, 64]

Via a platform [64] information on implementation, technical documentation and supporting videos are provided. A free to use demo instance of the latest Enterprise edition is available on the OpenClinica website [65].

The entered data is stored in a PostgreSQL (PostgreSQL Global Development Group) [66] database. The Case Report Forms (CRF) are created using Microsoft Excel (Microsoft Corporation, Redmond, Washington) spreadsheets, which are subsequently uploaded and applied in the corresponding studies.[63, 64]

Table 2.10 shows the different user roles in OpenClinica and some corresponding permissions within a certain study. At the investigational sites the roles Investigator, Monitor, Clinical Research Coordinator and Data Entry Person are available. [67]
Compared to other EDC systems, the role concept of OpenClinica turned out to be more advantageous [68].

User Role	Permissions
Data Manager	Add/manage subjects, view/schedule/enter data into CRFs, import data, source data verification, create/edit/manage CRFs, manage rules, extract data, view/build study, assign users
Study Director	Same as Data Manager
Data Specialist	Add/manage subjects, view/schedule/enter data into CRFs, import data, sign subject data, extract data
Monitor	View subjects/events, source data verification, extract data
Data Entry Person	Add/manage subjects, view/schedule/enter data into CRFs, import data

Table 2.10: Roles and selected permissions within a certain study in OpenClinica. Adapted from [67].

As shown in Table 2.11, OpenClinica offers several Simple Object Access Protocol (SOAP) web services, e.g. for listing or creating study subjects or scheduling events. Certain functionalities like listing study metadata or validating rules are implemented as Representational State Transfer (REST) web services.[64]

Web Service Name	**Function**
StudySubject	Creating a study subject in a target study/site, Listing study subjects that are part of a certain study or site, Querying if a study subject is part of a certain study/site
Event	Scheduling an event for a study subject in a certain study
Data	Insert CRF item data in CDISC ODM XML format
Study	Listing the metadata of a certain study, Listing all studies/sites of the OpenClinica instance
StudyEventDefinition	Listing study event definitions and their properties for a certain study

Table 2.11: Web services of the OpenClinica SOAP API. Adapted from [69].

In various cancer studies e.g. [70, 71, 72] and also for fundamental research activities in the neuroblastoma disease [73] OpenClinica has already been used for data capture.

The graphical user interface (GUI) of OpenClinica could be improved in terms of usability [68] and is not optimized for the use on mobile devices. Therefore, the framework Bootstrap [74] has been applied to the currently used version within the neuroblastoma research network to introduce responsive design[3] and to increase user-friendliness.

Figure 2.8 shows the EDC system's main page of the OMS/DES clinical trial, which has been adapted using the Bootstrap framework.

Alternative EDC systems are:

- REDCap [76] (Vanderbilt University, Nashville, Tennessee)
- OpenMRS [77] (OpenMRS Community, Indiana)
- ClinCapture [78] (Clinovo, Sunnyvale, Califorina)
- Clirinx [79] (Dublin, Ireland)

[3]The web page content adjusts itself to different screen sizes [75].

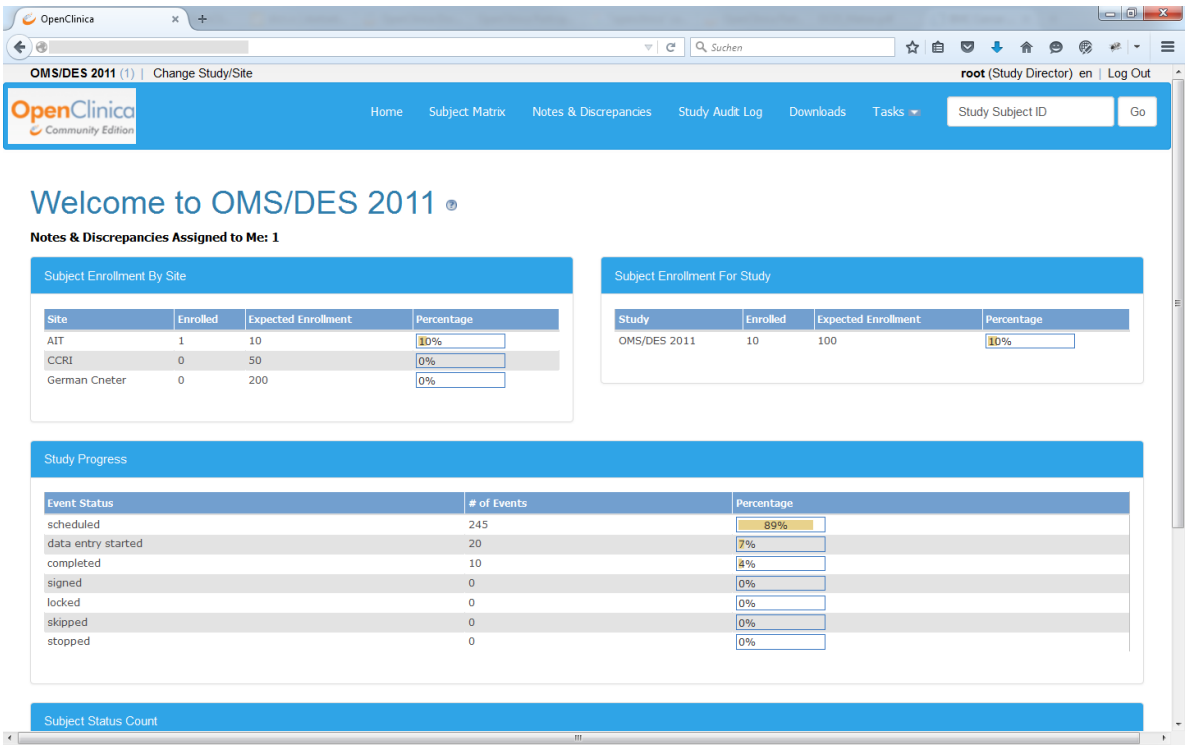

Figure 2.8: OpenClinica with Bootstrap used in the OMS/DES study.

2.8.1 OpenClinica Participate

The Enterprise Edition of OpenClinica offers a module for capturing PRO of study participants. Patients, for which 'OpenClinica Participate' has been enabled in OpenClinica, will receive a message via SMS or email containing a link to the CRFs to be completed. Submitted data is then available in OpenClinica. Figure 2.9 shows the icon for connecting participants in OpenClinica.[80]

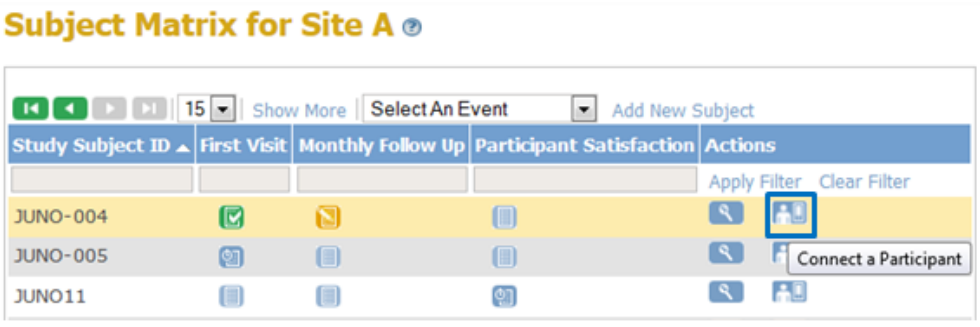

Figure 2.9: Connecting participants in OpenClinica [81].

OpenClinica provides Participate as a Software as a Service (SaaS) module. As shown in Figure 2.10, messages sent to participants contain a link to a study-specific subdomain and enable patients to enter data using unique access codes. [80, 82]

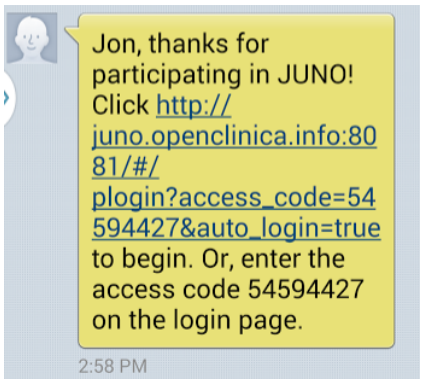

Figure 2.10: Message from OpenClinica sent to participants [81].

CRFs are presented to patients as shown in Figure 2.11. Forms are based on an Node.js-version of the open source Enketo Smart Paper (Enketo LLC, Denver, Colorado) API used for conducting surveys [83], which has been adapted to OpenClinica.[80, 82, 84]

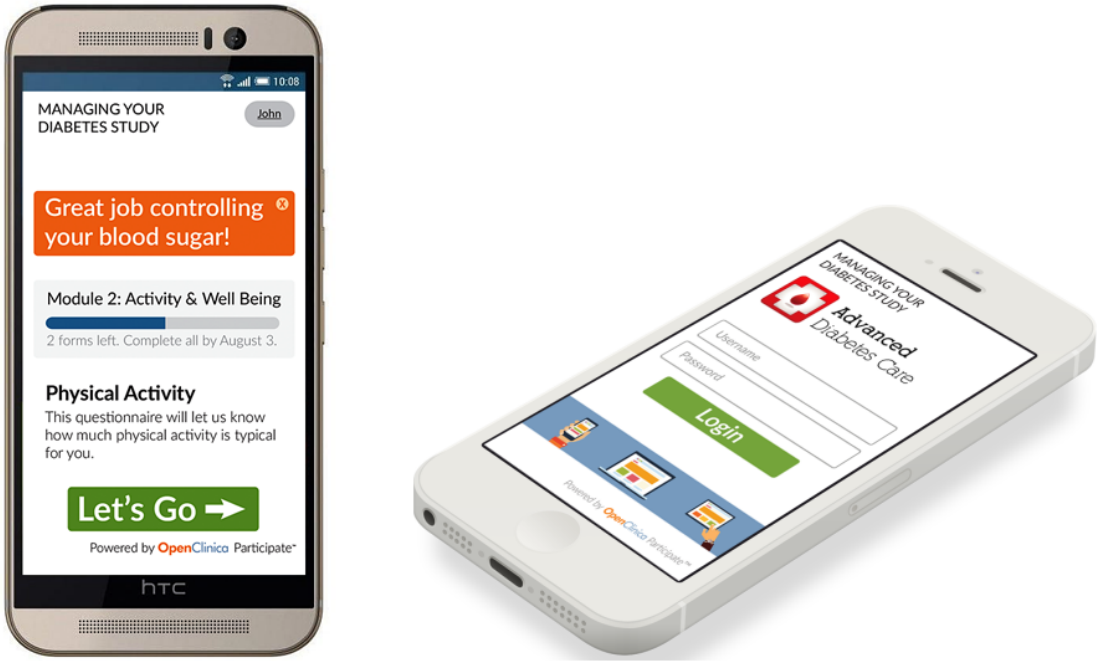

Figure 2.11: Completing forms with OpenClinica Participate. Adapted from [81, 82].

Chapter 3

Methods

This chapter describes the development environments, installation processes, programming languages, devices, technologies and general approaches used for developing the *EUPID Mobile* application for researchers and for modifying the *MoKi* application in order to transmit telemonitoring data to a back-end.

The implementation process consisted of two major tasks:

1. A user-friendly mobile application executable on different mobile operating systems, which allows physicians to easily register patients in medical research networks by creating pseudonyms, should be developed. The application should also offer the possibility to assign ID cards to patients and to save pseudonyms on NFC tags. The development environment should be selected with respect to the existing environment and should facilitate the possibility to reuse an existing source code. The functionality of the application should be evaluated on Android and iOS devices.

2. The telemonitoring solution described in Chapter 2.7 should be connected to an OpenClinica instance. Since up to now the ID being part of the system was only used for starting the *MoKi* application, interfaces to the back-end should be implemented and the integration of the ID card should be redesigned in order to enable its use for authentication purposes. An appropriate server technology for implementation and testing should be selected and set up on a local computer.

3.1 Developing EUPID Mobile with Apache Cordova

The resulting application should be made available to as many clinicians and researchers as possible. In order to account for the diversity of available operating systems for mobile devices and to avoid duplicated development effort, a framework for cross-platform application development was used.

Apache Cordova (Apache Software Foundation, Forest Hill, Maryland) is an open source framework, which allows development of multi-platform applications by using Hypertext Markup Language (HTML5), Cascading Style Sheets (CSS) and JavaScript instead of native programming languages.[85]

At the time of writing this thesis, the following platforms were primarily supported. The degree of support depended on the required functionality [86]:

- Amazon Fire OS (Amazon.com, Inc., Seattle, Washington)
- Android (Open Handset Alliance, Mountain View, California)
- Blackberry 10 (Blackberry, Waterloo, Canada)
- Firefox OS (Mozilla Corporation, Mountain View, California)
- iOS (Apple Inc., Cupertino, California)
- Ubuntu (Canonical Ltd. and Ubuntu community, London, United Kingdom)
- Windows Phone 8/8.1 (Microsoft Corporation, Redmond, Washington)
- Windows 8.0/8.1/10/ (Microsoft Corporation, Redmond, Washington)
- Tizen (Linux Foundation, San Francisco, California)

Since Android and iOS were the leading mobile operating systems in the second quarter of 2015 with a market share of 82,8% or 13,9% [87] respectively the development of the application for registering patients and assigning ID cards to patients was targeted to these two platforms.

Installing Apache Cordova was easily possible via the command line interface. For more complex applications and for proper debugging and versioning, Visual Studio 2015 Community (Microsoft Corporation, Redmond, Washington) offered tools for creating

Apache Cordova projects and running the application on virtual or attached devices. Installation guides are available on [88] and [89].

Apache Cordova applications are neither web applications nor native applications. For execution, the source code is wrapped according to the operating system of the device. In contrast to mere web applications, it is possible to easily access device-specific hardware through plugins. Plugins enable access to e.g. the mobile phone's camera, accelerometer or media.[85, 86]

Table 3.1 shows the plugins used for the *EUPID Mobile* application. A comprehensive list of existing plugins can be found under [90]. Plugins could be added in Visual Studio in the solution's *config.xml* file.

Name	**Plug-In-ID**	**Used for**
Whitelist	cordova-plugin-whitelist	App navigation
Active Directory Authentication Library (ADAL)	cordova-plugin-ms-adal	Microsoft Azure Active Directory (OAuth) client authentication
NFC	com.chariotsolutions.nfc.plugin	Reading and writing NFC tags on Android and determining whether NFC is enabled on the device
BarcodeScanner	phonegap-plugin-barcodescanner	Scanning barcodes on Android and iOS
Clipboard	com.verso.cordova.clipboard	Deleting clipboard content
Toast	cordova-plugin-x-toast	Giving user feedback via native text popups
Network Information	cordova-plugin-network-information	Determining whether the mobile phone has a network connection
Device	cordova-plugin-device	Determining whether the mobile phone's operating system is Android or iOS
Vibration	cordova-plugin-vibration	Giving haptic user feedback

Table 3.1: EUPID Mobile - Used Cordova plugins.

In order to reuse an already existing source code with reasonable effort and to be able to integrate some of the generated functionalities into other Cordova applications, a module-based approach was applied during implementation. Therefore, the open source

JavaScript framework AngularJS (Google Inc., Mountain View, California) described on [91] was used.

The device emulators provided by Visual Studio were not used for debugging. To simplify the development process, the application was initially debugged on an Android 5.0-powered Samsung Galaxy S5 smartphone (Samsung, Seoul, South Korea) using Visual Studio. After the implementation of the core functionalities, the application was also tested on an Apple iPhone 5c (Apple Inc., Cupertino, California).

Since debugging iOS devices was not directly possible with computers running on Windows, an Apple MacBook (Apple Inc., Cupertino, California) had to be used. Two possible approaches were identified. For each approach the neccessary steps for setting up the debugging environment are described in Appendix B.1. Moreover, some troubleshooting information is provided.

3.2 Structure of EUPID Mobile

The workflow of the application has been developed with respect to usability and consistency regarding the existing patient registration web application. The following libraries were used:

- AngularJS 1.4.7
- jQuery 1.9.1
- Bootstrap 3.3.5
- Angular UI Bootstrap 0.14.2

The *EUPID Mobile* application was included in the *EupidMobileApp* module and was divided into AngularJS controllers and views as shown in Figure 3.1 in order to facilitate expandability. The view *patReg.html* and the corresponding controller *patReg.js* for patient registration had been taken and adjusted from the EUPID web application.

index.html represented the overall container for the different views (i.e. screens). *app.js* provided routing information and contained login and logout functionalities.

For authentication at the services via OAuth2.0[1] the Active Directory Authentication Library (ADAL) , which was provided in the form of a Cordova plugin, was used. Methods for requesting access tokens, refreshing tokens, reading the token cache and deleting tokens were defined in *adal-cordova-angular.js*. The configuration for obtaining tokens from a Microsoft Azure Active Directory (MAAD) consisted of the variables *authority*, *redirectUri*, *resourceUri* and *clientId*, which were defined accordingly.

[1]For more information see `http://oauth.net/2/`.

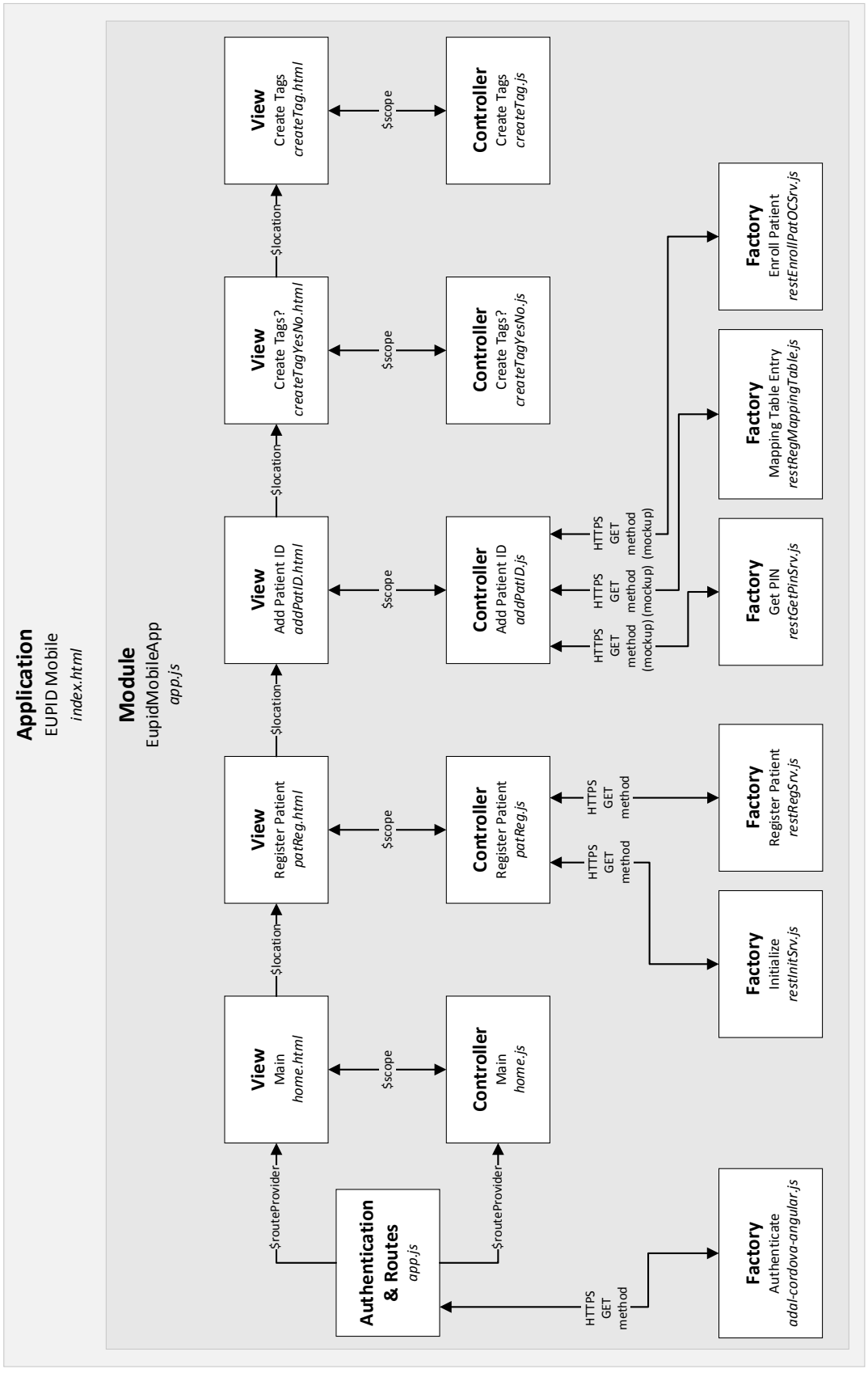

Figure 3.1: AngularJS modules and views of the EUPID Mobile application.

The different REST services used for implementation were called using Asynchronous JavaScript and XML (AJAX) requests and are described in Table 3.2. Within *EUPID Mobile* all data transfer was done via HyperText Transfer Protocol Secure (HTTPS). The REST interfaces for obtaining a PIN, for making an entry in the mapping table and for enrolling the patient in OpenClinica were implemented as mockups. Therefore, JavaScript Object Notation (JSON) objects stored in JavaScript files simulated the server's response. These objects were returned to the AJAX calls for testing purposes.

For debugging the REST interfaces and for analyzing the network traffic on the mobile phone the application *PacketCapture* (Developer Grey Shirts, Google Play Store) was used.

Nr.	Purpose	Method	Parameters	Response
1	Initialization	GET	Token (Bearer)	Response Code, list of contexts
2	Register Patient	POST	Token (Bearer), Context, hashed IDAT, encrypted IDAT, forced or not	Response Code, PSN
3	Register ID Card	GET	Pseudonym, Card ID	Response Code, PIN
4	Make Mapping Table Entry	GET	Pseudonym, Patient Card ID, PIN	Response Code
5	Enroll Patient in OpenClinica	GET	Pseudonym	Response Code

Table 3.2: Representational State Transfer (REST) interfaces used in EUPID Mobile. Requests 3 – 5 were implemented as mockups. (IDAT = Identity Data)

3.2.1 Content of the Patient ID Card

RFID and NFC provide intuitive ways of reading ID cards simply by touching them with a mobile phone. However, up to now iOS devices do not support this function. Therefore, in order to offer the possibility to contactlessly read the ID of a patient ID card on iOS devices, a QR Code reader was embedded as an alternative to NFC.

The patient ID card contained an NFC tag and a QR Code. The premanufactured ID card was conform to ISO/IEC 7810 ID-1 and carried the information shown in Table 3.3. For developmental purposes the patient ID on the ID card was set to "PATIDCARD1234567890". The application specified to handle the NFC intent was defined

as "at.ac.at.moKi". The QR Code was generated via [92]. For writing the NFC tag, an NFC plugin for Eclipse (Thomas Skjølberg, GitHub Repository) [93] in combination with the application *NFC Developer* (Developer Thomas Rorvik Skjølberg, Google Play Store) was used.

NFC Tag	
Type	NFC Forum Type 2
Total Memory Size	168 bytes
Stored Records	MIME type (RFC2046) record: containing the patient ID Android Application record: use *MoKi* for handling NFC intents
QR Code	
Type	QR Code Model 2
Version	1
Error Correction	M
Content	ID of patient card

Table 3.3: Characteristics of the patient ID card. (NFC = Near Field Communication)

3.3 Connecting MoKi to OpenClinica

The general approach was to locally set up a virtual Windows 7 machine using Oracle Virtualbox 5.0.12 (Oracle Corporation, Redwood City, California) and installing an Apache Tomcat server (Apache Software Foundation, Forest Hill, Maryland) which provided an OpenClinica instance as a back-end for *MoKi*. As shown in Figure 3.2, observations were transmitted to OpenClinica using the OpenClinica SOAP API.

Figure 3.2: Data transmission between patient and server. Partly taken from [63][94].

3.3.1 Setup of OpenClinica

Detailed descriptions on how to set up OpenClinica 3.5 and other versions on an Apache Tomcat Server can be found in the Wiki-section of the OpenClinica GitHub Repository under [95]. The following versions of required software were used:

- Java Development Kit (JDK) 7.51
- Apache Tomcat 7.0.52
- PostgreSQL 8.4.21
- OpenClinica 3.5
- OpenClinica Web service Package 3.5

In OpenClinica, for every vital parameter or toxicity respectively, a CRF had been created using Microsoft Excel and was uploaded to the demonstration study "NBLStudy". Structure and content of the CRFs are provided in Appendix C.2.

The mapping service and the service for enrolling patients using their PSN were not implemented for OpenClinica yet. Therefore, for demonstration purposes a data entry

person with user name "PATIDCARD1234567890" and PIN "12345678" was created, which was authorized to schedule events for a patient named "PATIDCARD1234567890" enrolled in the study "NBLStudy" in OpenClinica.

Regarding the documentation of skin alterations, where taking photos using the smartphone's camera was possible, a simple web application for transmitting files was programmed and deployed on the Apache Tomcat server.

3.3.2 Adjusting the MoKi Application

For developing, the Eclipse Java EE IDE for Web Developers 4.5.0 (Eclipse Foundation, Inc., Ottawa, Canada) was used. The application was debugged on an LG-P700 smartphone, which was part of the original telemonitoring system described in Chapter 2.7.

The *MoKi* application has been extended in order to allow communication with OpenClinica as a back-end. The supported health parameters and toxicities were not changed and the general workflow for gathering measurements from the measurement devices or the smart poster remained the same.

The routine for reading the patient ID card via NFC was adjusted and an input field for entering the PIN was added, which appeared when the ID card was approached.

XML files containing the SOAP requests were added and classes for hashing the entered PIN and connection management were implemented.

Sending SOAP Requests

Two versions of the source code have been implemented:

- Data transmission via Hypertext Transfer Protocol (HTTP) for developmental purposes.
- Data transmission via HTTPS. Switching from HTTP to HTTPS could easily be done by uncommenting the corresponding source code.

For scheduling an event and entering telemonitoring data into OpenClinica CRFs, the SOAP web services *Event* and *Data* described in Chapter 2.8 were used.

Listing 3.1 shows the SOAP message header containing user name (*#OCUSER*) and password (#OCPIN) for data entry authorization. The password was hashed using the Secure Hash Algorithm 1[2] (SHA1) before transmission. At runtime, both values were replaced with the patient's user credentials (ID, PIN) for demonstration purposes.

Listing 3.1: SOAP requests - header

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/
    envelope/" xmlns:v1="http://openclinica.org/ws/event/v1" xmlns:
    bean="http://openclinica.org/ws/beans">
  <soapenv:Header>
    <wsse:Security soapenv:mustUnderstand="1"
      xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
          wss-wssecurity-secext-1.0.xsd">
      <wsse:UsernameToken wsu:Id="UsernameToken-27777511"
      xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
          wss-wssecurity-utility-1.0.xsd">
        <wsse:Username>#OCUSER</wsse:Username>
        <wsse:Password type="http://docs.oasis-open.org/wss/2004/01/
            oasis-200401-wss-username-token-profile-1.0#PasswordText">
            #OCPIN</wsse:Password>
      </wsse:UsernameToken>
    </wsse:Security>
  </soapenv:Header>
  <soapenv:Body>...</soapenv:Body>
</soapenv:Envelope>
```

[2]For further information see [96].

The SOAP message body for scheduling a new event is shown in Listing 3.2. At runtime, *#OCPATIENTLABEL* was replaced by the ID stored on the patient's ID card. The study label (*#OCSTUDYLABEL*) was fixed to "NBLStudy". The event type (*#EVENTTYPE*) was replaced at runtime by the type of event to be created (e.g. wellbeing, pain or blood pressure). *#EVENTDATE* was replaced by the current timestamp.

Listing 3.2: SOAP request - scheduling an event

```
<soapenv:Body>
  <v1:scheduleRequest>
    <v1:event>
      <bean:studySubjectRef>
        <bean:label>#OCPATIENTLABEL</bean:label>
      </bean:studySubjectRef>
      <bean:studyRef>
        <bean:identifier>#OCSTUDYLABEL</bean:identifier>
      </bean:studyRef>
      <bean:eventDefinitionOID>#EVENTTYPE</bean:eventDefinitionOID>
      <bean:startDate>#EVENTDATE</bean:startDate>
    </v1:event>
  </v1:scheduleRequest>
</soapenv:Body>
```

Listing 3.3 shows the SOAP request body for entering data into an earlier created wellbeing event (see also Listing 3.2). *#OCSTUDYOID*, *#OCSTUDYSUBJECTOID* and *#STUDYEVENTORDINAL* were parsed from the response of the previous SOAP request for creating a new event. *#WELLBEINGDATE*, *#WELLBEINGTIME* and *#WELLBEINGVALUE* were replaced by the user input values for wellbeing, date and time at runtime.

Listing 3.3: SOAP request - entering data in a scheduled wellbeing event

```
<soapenv:Body>
  <v1:importRequest>
    <ODM>
      <ClinicalData StudyOID="#OCSTUDYOID" MetaDataVersionOID="v1.0.0">
        <SubjectData SubjectKey="#OCSTUDYSUBJECTOID">
          <StudyEventData StudyEventOID="SE_WELLBEING"
              StudyEventRepeatKey="#STUDYEVENTORDINAL">
            <FormData FormOID="F_CRF_WELLBEIN_V1">
              <ItemGroupData ItemGroupOID="
                  IG_CRF_W_LISTOFMEASUREMENTS_2214" ItemGroupRepeatKey
                  ="1" TransactionType="Insert">
                <ItemData ItemOID="I_CRF_W_WELLBDATE"
                Value="#WELLBEINGDATE"/>
                <ItemData ItemOID="I_CRF_W_WELLBTIME"
                Value="#WELLBEINGTIME"/>
                <ItemData ItemOID="I_CRF_W_WELLBV"
                Value="#WELLBEINGVALUE"/>
              </ItemGroupData>
            </FormData>
          </StudyEventData>
        </SubjectData>
      </ClinicalData>
    </ODM>
  </v1:importRequest>
</soapenv:Body>
```

The SOAP requests for the other vital parameters and toxicities can be found in Appendix C. The item *#SKINIMAGE* of the SOAP request for entering observed skin alterations contained the path to the image on the server site.

Chapter 4

Results

Figure 4.2 shows the approach and overall workflow for integrating ePRO into the existing neuroblastoma research network. The patient was equipped with a telemonitoring system and used the ID card and PIN – both provided by the physician in charge – for transmitting vital parameters and occurring toxicities to the EDC system OpenClinica described in Chapter 2.8. In OpenClinica, the data were then not only available for treatment purposes but also for cancer research.

Figure 4.1: Gathering electronic patient-reported outcome for research purposes.

Furthermore, Figure 4.2 depicts the two working packages, which have successfully been implemented at the front-end site:

- A mobile application has been developed for physicians enabling them to register patients in a certain context (i.e., creating PSNs), to provide patients with an ID card and a PIN and to create NFC tags.
- The *MoKi* application being part of the telemonitoring system described in Chapter 2.7 has been modified to enable user authentication via ID card and PIN and to transmit telemonitoring data to OpenClinica.

4.1 Implementation Concept

The workflow shown in Figure 4.2 is described below. After starting the *EUPID Mobile* application, physicians were forwarded to the MAAD login page. After entering user name and password a token was returned (1), which was used for accessing the EUPID portal services. After entering the patient's IDAT and selecting an available context, the patient was registered (2). Three scenarios were possible (3):

- The patient had not yet been registered within the selected context. Then a PSN was created and returned to the physician.
- A similar patient had already been registered within the selected context. Physicians could force the patient registration and a PSN was returned.
- The patient had already been registered within the selected context. In this case the already existing PSN for this patient was returned.

In addition to the patient's IDAT, the hashed IDAT and encrypted IDAT were sent to the portal services for the reasons described in Chapter 2.2.

After obtaining a PSN, a premanufactured patient ID card could be linked to the patient. Therefore, the card ID was read via NFC or QR Code reader (4). A PIN was returned from the PIN service (5) and delivered to the physician (6). PSN, ID and PIN were stored in a mapping table, which was later used by OpenClinica for patient identification (7). The patient was also registered in OpenClinica with his or her PSN (8). The mapping service and PIN service were not implemented during working on this thesis. The REST interface for registering patients in OpenClinica via *EUPID Mobile* was concurrently being developed by the AIT. As the final step of the patient registration procedure, physicians could use an Android device for creating NFC tags containing PSNs (9).

Patients then received the ID card, which was put into a plush toy (10), and the PIN (11). After they had been provided with the telemonitoring system described in Chapter 2.7, the plush toy was used for launching the *MoKi* application via NFC (12). The authorization PIN had to be entered (13) and vital parameters as well as toxicities could be captured (14). The telemonitoring data were transmitted to OpenClinica using the ID and PIN as user credentials (15). Each patient represented a data entry person in OpenClinica. OpenClinica used the mapping service for identification of the corresponding PSN (16) and the new observations was entered for the correct patient (17).

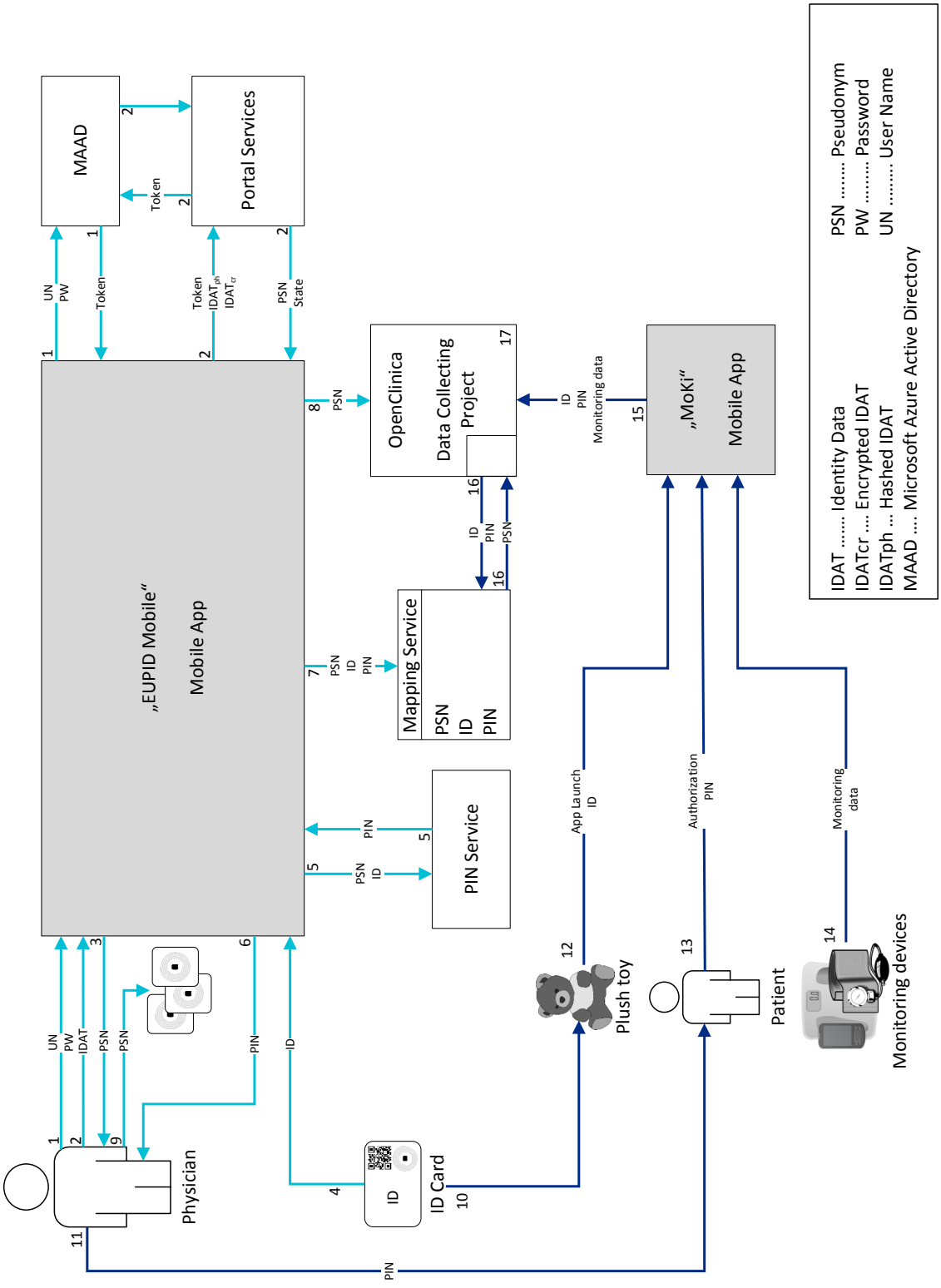

Figure 4.2: Concept for integrating electronic Patient-Reported Outcome into the existing neuroblastoma research network.

4.2 EUPID Mobile Application

Installing Visual Studio 2015 (Community Edition) on Microsoft Windows 7 (Microsoft Corporation, Redmond, Washington) and debugging on the Android device was not accompanied by significant problems.

The user interface of the *EUPID Mobile* application was designed with respect to the existing patient registration web application. Figure 4.3 shows the flowchart of the *EUPID Mobile* application, which offered the three main functionalities 'Register Patient', 'Link Patient ID Card' and 'Create NFC Tags'. The paths involving NFC were only available for Android devices.

The *EUPID Mobile* application was successfully presented to the ENCCA community at the ENCCA Closing Conference in December 2015 in Brussels.

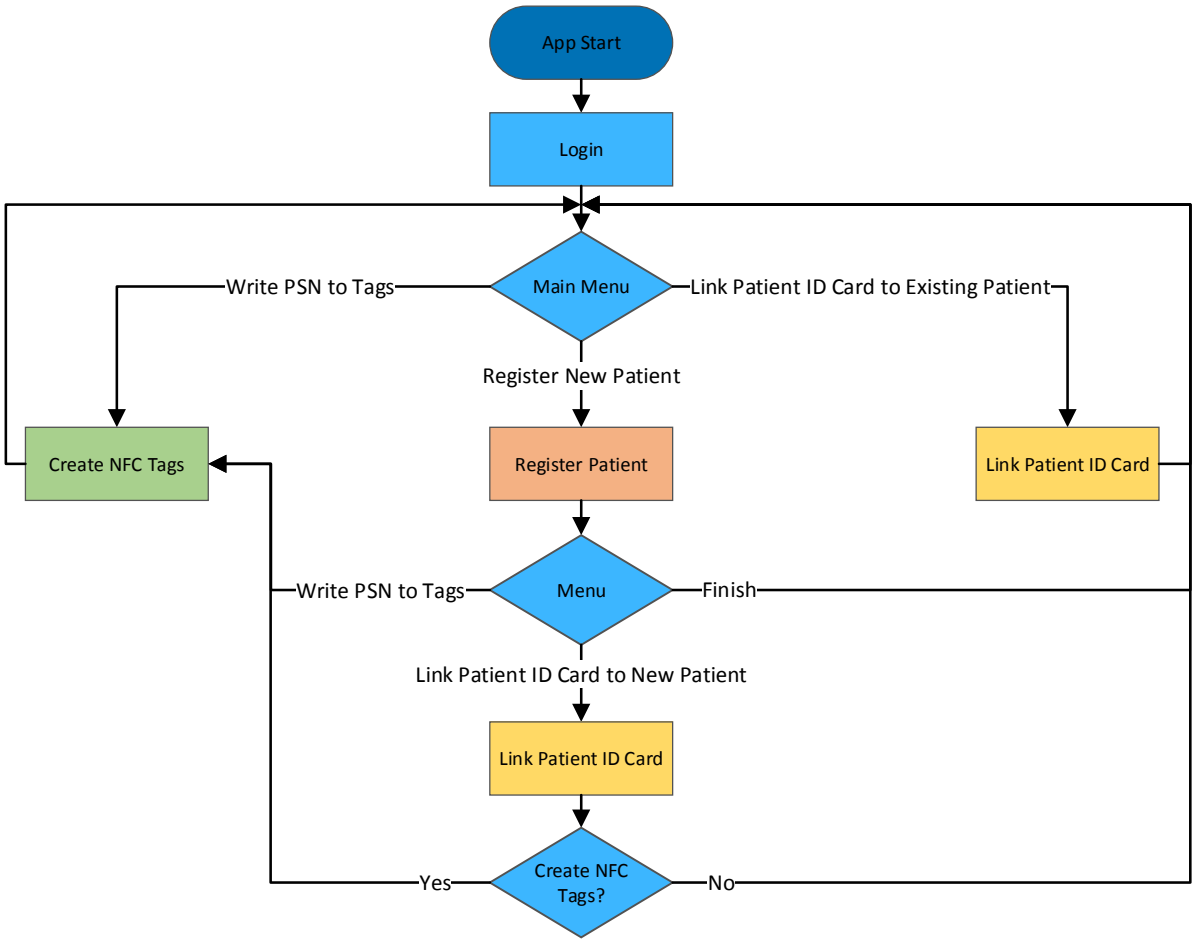

Figure 4.3: Flowchart of EUPID Mobile. (PSN = Pseudonym)

4.2.1 User Interface on iOS Devices

Figure 4.4 shows the icon, the main screen and the login page of *EUPID Mobile* on iOS devices. As shown in Figure 4.5, selecting a context and entering the patient's IDAT when registering a patient was supported by a drop-down list and a datepicker.

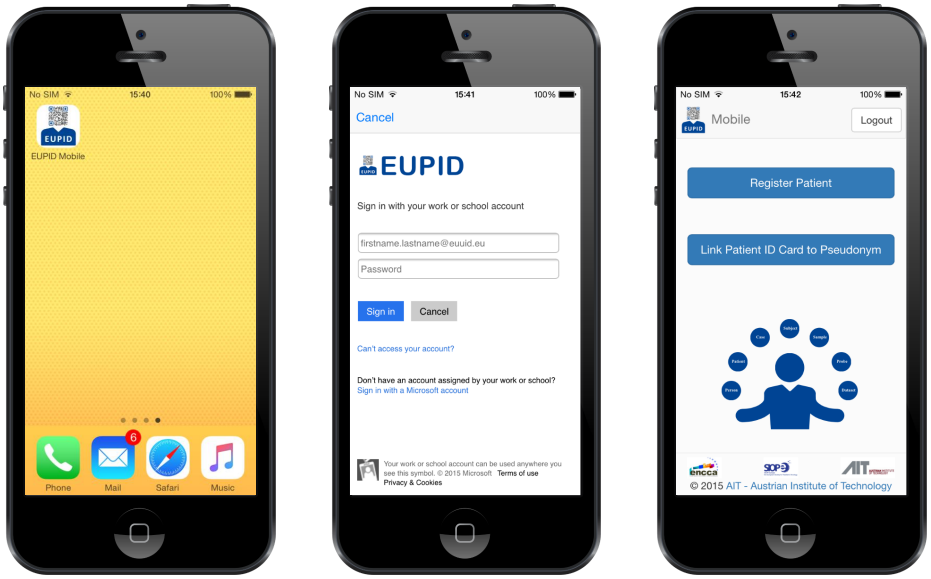

Figure 4.4: Icon (left), login page (middle) and main screen (right) of EUPID Mobile for iOS Devices.

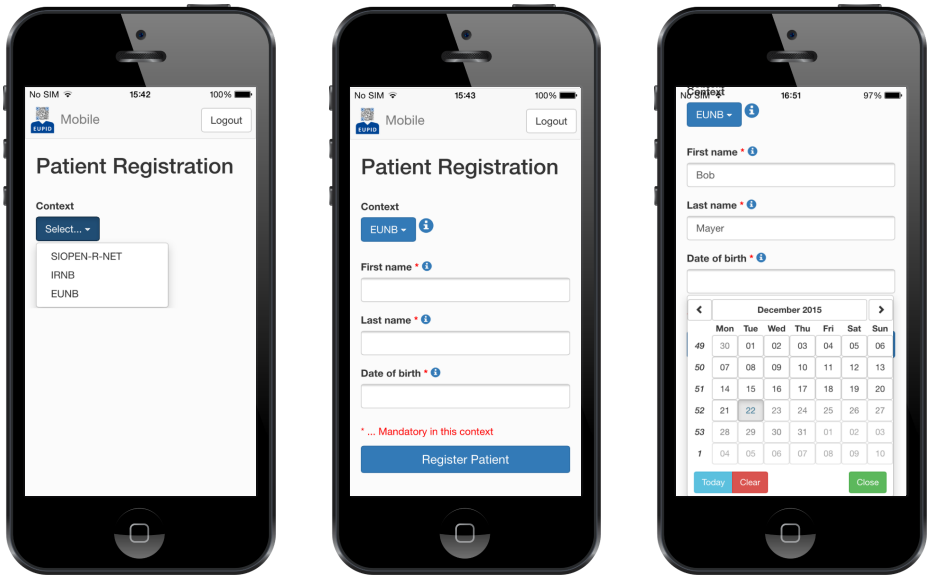

Figure 4.5: Selecting the context and entering the patient's identity data.

When creating a PSN, the three states shown in Figure 4.6 were possible. If a similar patient had already been registered in a certain context, patient registration could be forced by checking the checkbox **Force Patient Registration**.

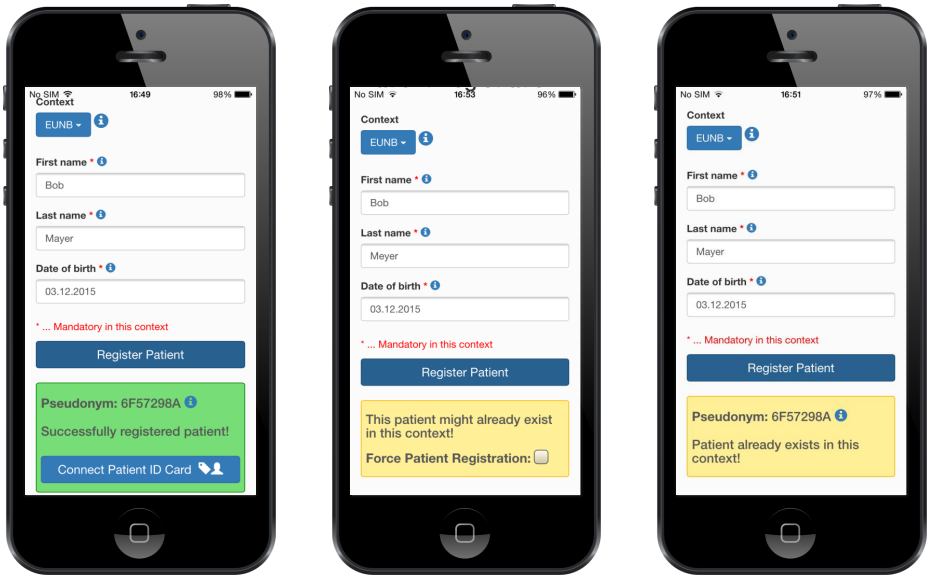

Figure 4.6: Possible states when registering a patient.

After clicking on Connect Patient ID Card, an ID card could be read via QR Code. A PIN was returned when clicking on Register ID.

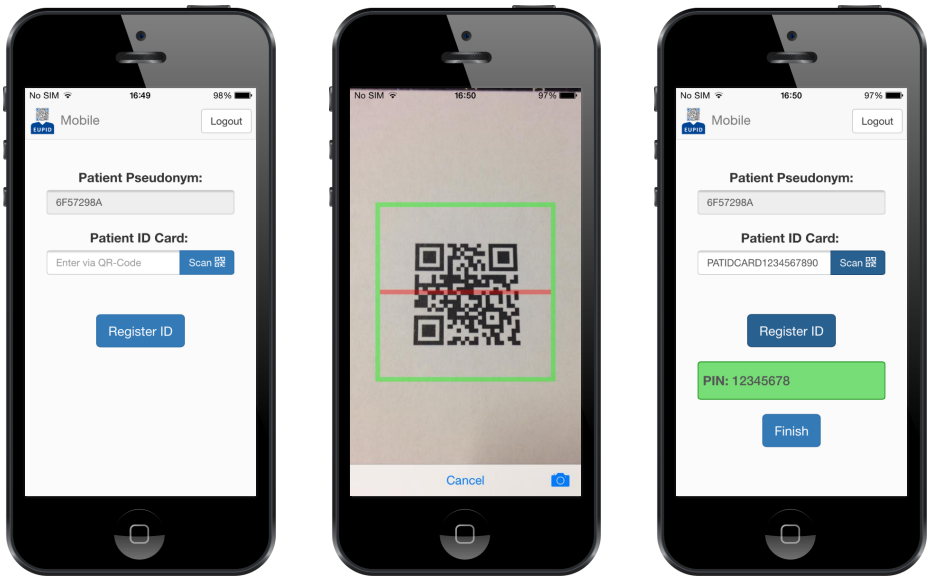

Figure 4.7: Linking a patient ID card to a pseudonym and retrieve PIN.

4.2.2 User Interface on Android Devices

On Android devices NFC can be used for customized purposes. Figure 4.8 shows the extended menus and the possibility to read patient card IDs via NFC. It was easily possible to write the PSN to NFC tags by approaching them to the mobile phone. Figure 4.9 shows the messages, which guided the user through the writing process.

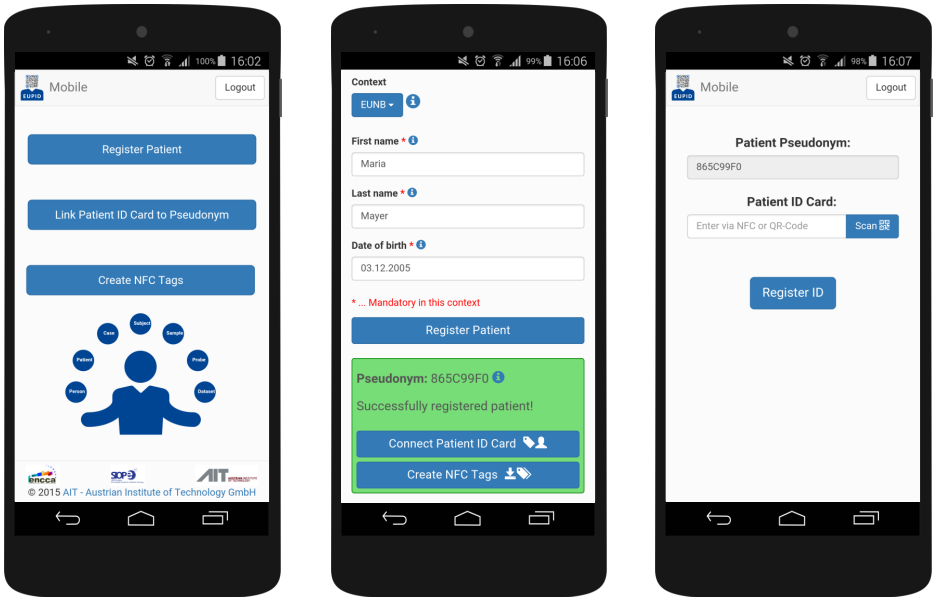

Figure 4.8: Extended menus and functionality on Android devices.

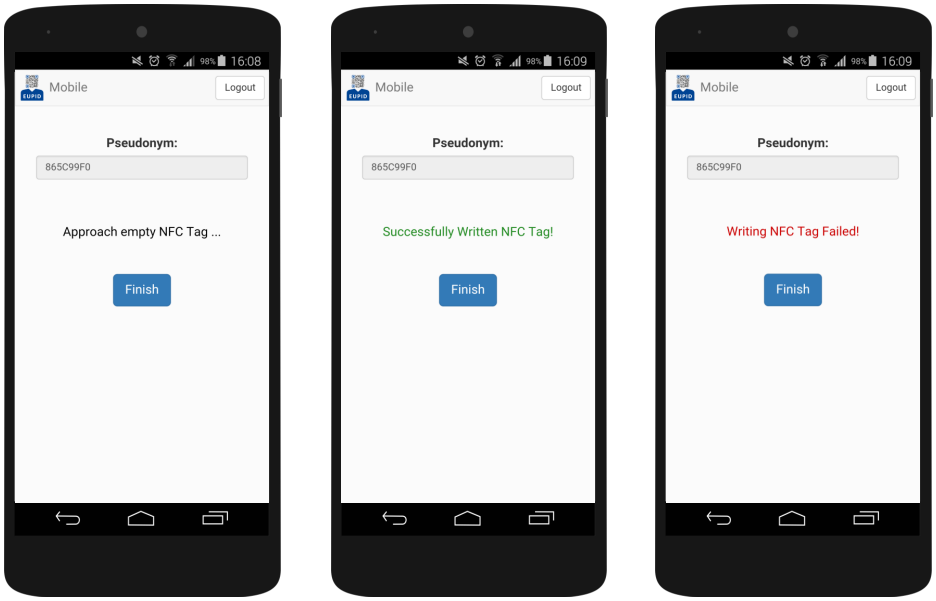

Figure 4.9: Messages guiding the creation of NFC tags on Android devices.

4.2.3 User Feedback

Besides spinners and toasts informing the user that work is in progress or he or she had been logged in or out respectively, some messages concering troubleshooting (no internet connection, NFC is disabled) had been implemented as shown in Figure 4.10. The Microsoft login page directly displayed an error message if the user could not be logged in due to invalid credentials or because the user was already logged in on an other device.

If further information or hints regarding e.g. the available contexts or the IDAT fields were available, this was indicated by the symbol .

Since only Android devices have got a hard key-back button and to increase usability of the single workflows, the back button had been disabled for all screens. A toast indicated that this function was not available and that going back to the main screen was possible by clicking on the EUPID icon in the menu bar.

On Android devices the mobile phone vibrated in case an NFC tag was approached in order to provide some haptic feedback to the user.

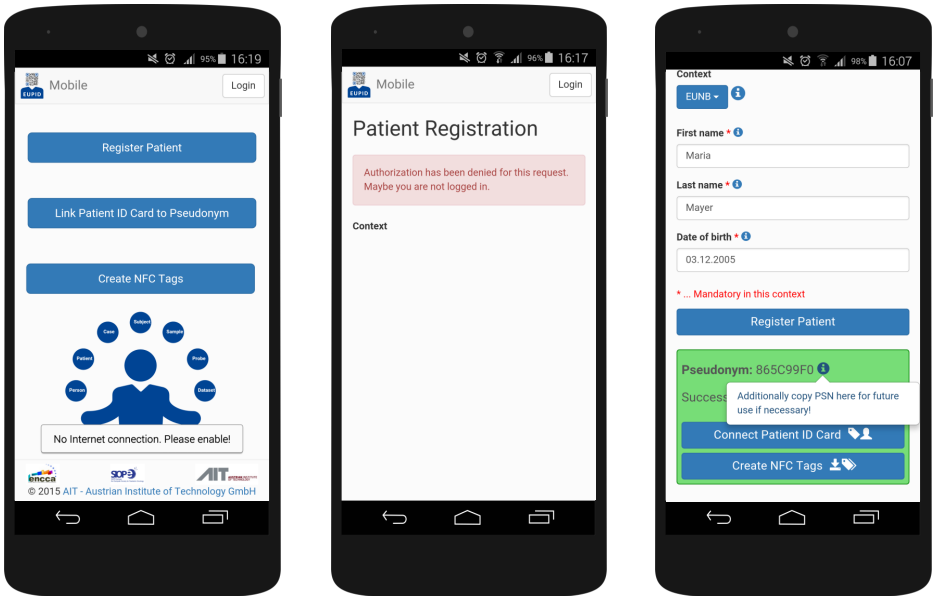

Figure 4.10: User feedback and help texts implemented in EUPID Mobile.

4.3 MoKi Application

Figure 4.11 shows the home screen of *MoKi* and prompts added to the user interface in order to capture the patient card ID via NFC and the PIN.

After starting the application, patient card ID and PIN could be provided by touching the ID card and entering the PIN. In case patient card ID and PIN had not already been entered after a measurement was performed, the user was prompted to enter the credentials before the measurement could be saved. Figure 4.12 shows the patient ID card used for demonstration purposes.

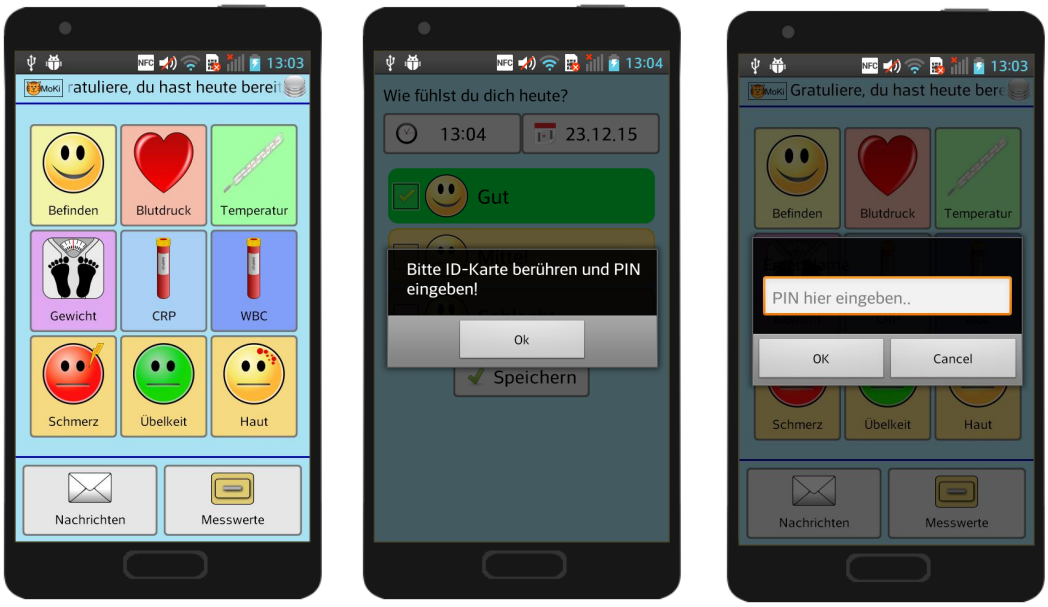

Figure 4.11: Main screen of MoKi (left) and prompts to touch patient card ID (middle) and to enter the PIN (right).

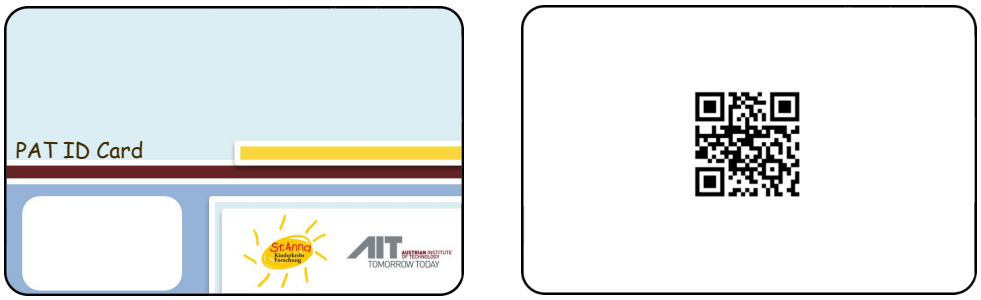

Figure 4.12: Patient ID card. Left: front, right: back.

4.4 Submitting Monitoring Data to OpenClinica

Transmitting telemonitoring data to OpenClinica is demonstrated for the case of capturing the health parameter "wellbeing" in the following. After starting the application via the patient ID card and entering the PIN, the value representing the current wellbeing was selected as shown in Figure 4.13. A toast indicated that the measurement value had been stored on the mobile phone and was successfully transmitted to OpenClinica.

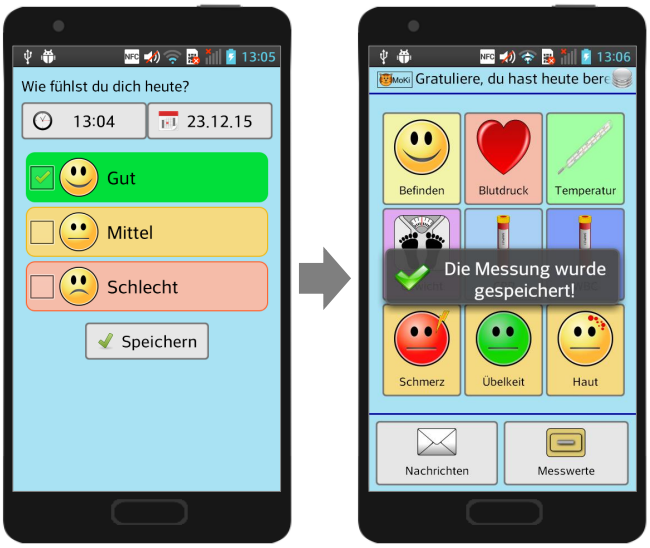

Figure 4.13: Transmitting a wellbeing observation to OpenClinica. After performing the measurement (left), the observation was saved on the mobile phone and transmitted to OpenClinica (right).

Figure 4.14 shows that a new event had successfully been scheduled for the specific patient and wellbeing data were transmitted.

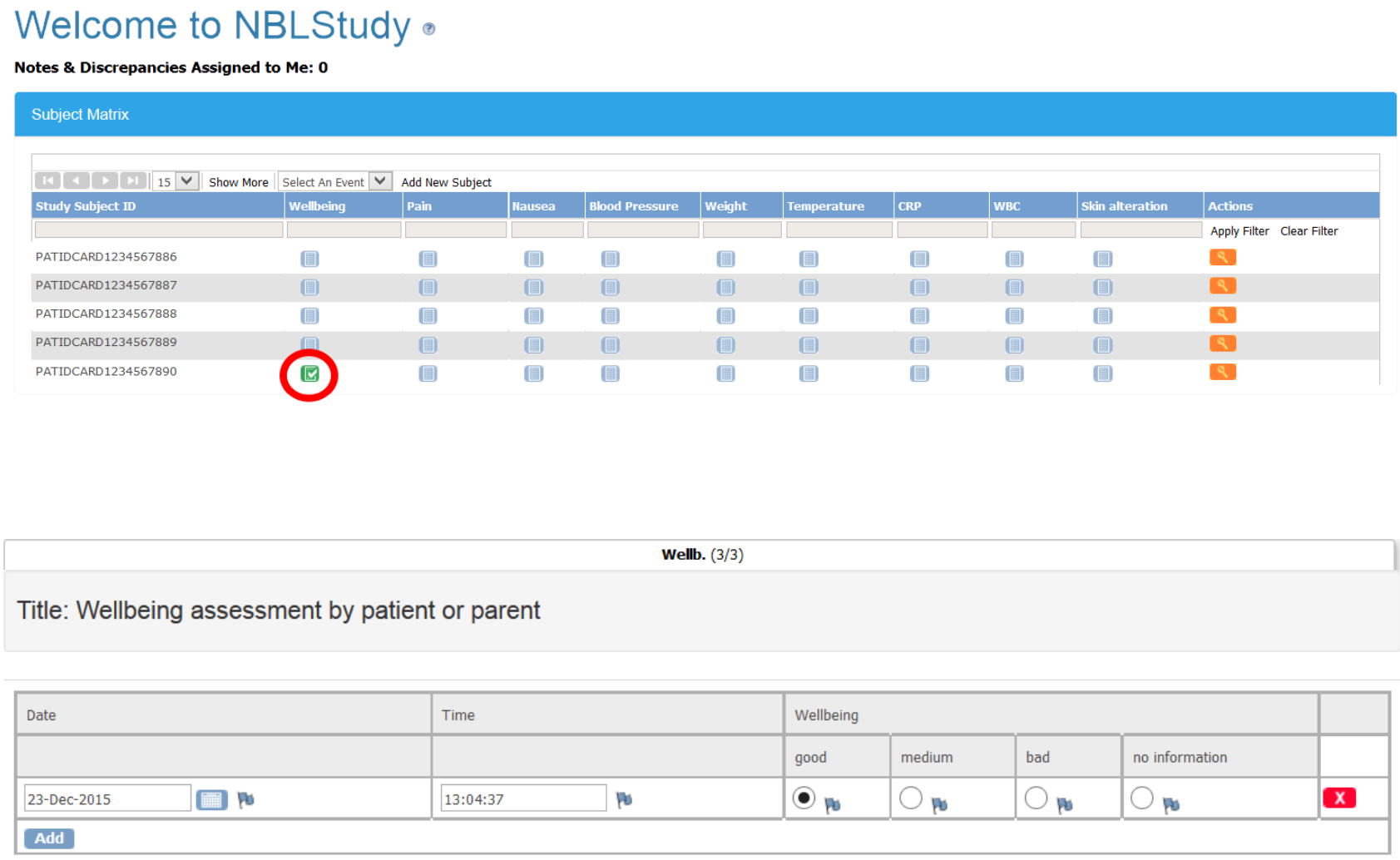

Figure 4.14: Event scheduled in OpenClinica (above) and transmitted wellbeing observation (below).

Chapter 5

Discussion

Cancer research by connecting data from different sources is important for generating a deeper understanding of rare diseases, improving their treatment and finding new treatment options. Therefore, ePRO is described as a valuable source of information in Chapter 1.4. The *EUPID Mobile* application, which was developed in the course of this thesis, provides an efficient way of enabling patients to submit health parameters and observed toxicities to the neuroblastoma research network by using an improved and extended version of *MoKi*.

In 2015, OpenClinica Participate was introduced. OpenClinica Participate is an alternative to *MoKi* and might be sufficient to record data from simple ePRO forms. In contrast to the *EUPID Mobile* application developed in this thesis, OpenClinica Participate did not support NFC. Furthermore, OpenClinica Participate was not supported by the Community Edition of OpenClinica at the end of writing this thesis. In order to activate OpenClinica Participate for a certain study in OpenClinica Enterprise, a request for activating a study-specific subdomain had to be sent to the OpenClinica team as described under [97]. Nevertheless, selected features of OpenClinica Participate could be used to further improve capturing ePRO by inviting patients to complete questionnaries or standard CRFs.

The premanufactured patient ID card or the plush toy containing it, respectively, represented the link between the two developed applications. In order to contactlessly read the patient card ID with *EUPID Mobile* on iOS devices, the ID was additionally stored in the form of a QR Code. Future improvements might necessitate storing a larger amount of information on the patient's ID card. Thus, a QR Code was favoured over a normal

barcode. However, the embedded reader of *EUPID Mobile* also supported barcodes as shown in Figure 5.1.

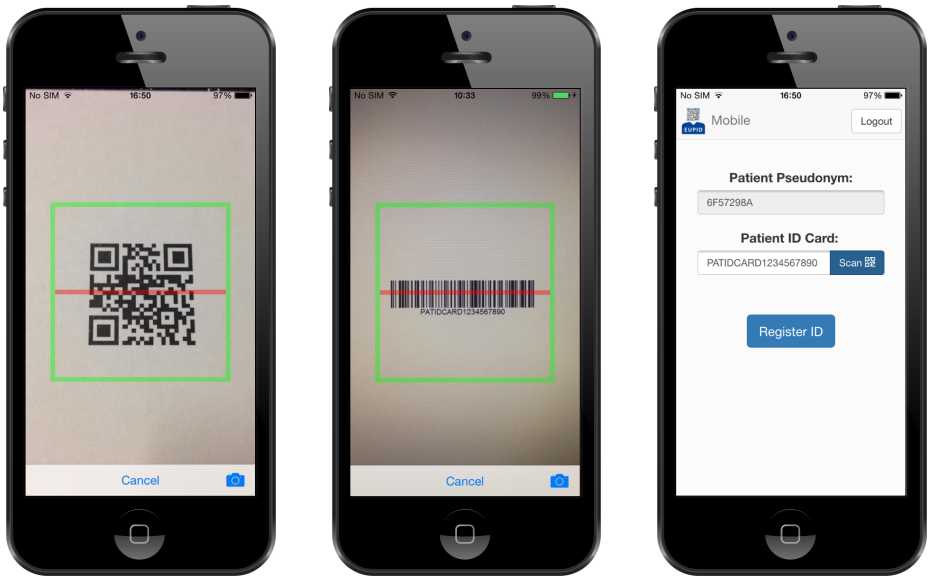

Figure 5.1: Possibility to use the embedded reader of EUPID Mobile for capturing the patient Card ID via QR Code (left) or alternatively via barcode (middle).

The patient ID card might also play an important role in patient follow-up as described in work package 13 [98] of ENCCA. One essential step of this work package is to provide a 'survivorship passport' to patients, which contains the cancer treatment history and helps to cure secondary diseases [99]. The patient ID card might be used for accessing these important data whenever needed in future medical interventions.

When connecting data from different contexts, proper identification of items containing patient information is necessary. Therefore, the workflow of *EUPID Mobile* for creating NFC tags could also be useful for labelling biosamples (e.g. blood probes) or patient related documents as shown in Figure 5.2. As an alternative to attaching loose NFC tags to items, specific and premanufactured documents and test tubes with embedded NFC tags might be more user-friendly. The main advantage of NFC tags for labelling purposes is that they make patient identification more difficult for unauthorized persons. In this concept, only a PSN was stored on the tag, which was not directly readable on the surface.

In general, physicians are still sceptical about the EUPID concept and the idea of assigning PSNs to patients. They prefer using names or at least initials for patient iden-

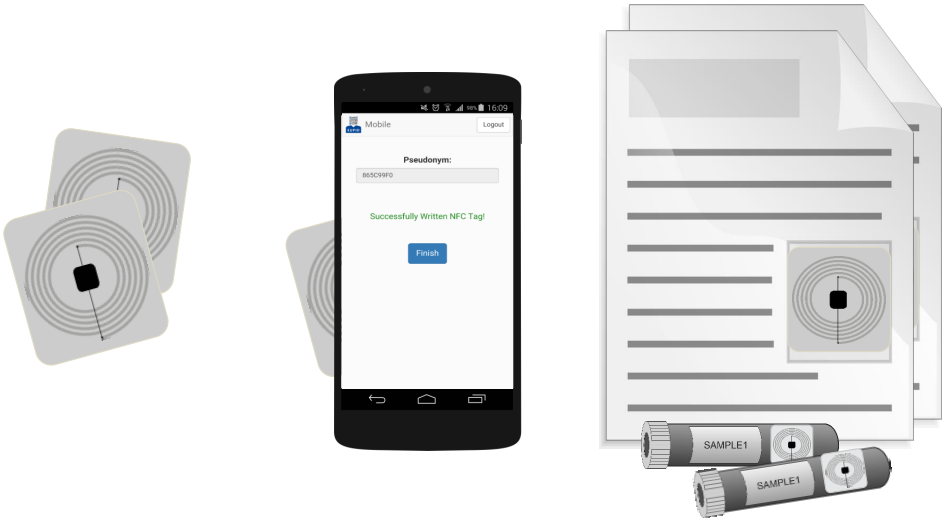

Figure 5.2: NFC tags for labelling patient related documents and biosamples.

tification. However, since the amount of available data steadily increases and data is progressively connected, such solutions will – also or especially in medical research – be inevitable in the future if patient privacy shall be preserved.

Patients have the right to insist on protection of their data. However, total anonymization would make it impossible for them to benefit from new findings and to receive the best possible treatment. Pseudonymization can be seen as a compromise if reidentification of patients is only done if necessary and by authorized people.

Initially, it was discussed that physicians should also be able to start *EUPID Mobile* and authenticate themselves via ID card and NFC. This process was not implemented because:

- A WAAD was used for handling the physicians' access to the different portal services. Therefore, storing the physicians' user name on an ID card was considered to be an unnecessary security risk (e.g. physicians might lose the ID card).
- iOS devices did not support NFC for starting applications and reading ID cards. Starting an application by reading a QR Code – which necessitated to manually start a QR Code reader application first – was considered to be needlessly complicated and inconvenient compared to starting *EUPID Mobile* manually.

For registering patients in research networks, the already existing web application for patient registration could be used as an alternative to *EUPID Mobile*. Eventually, it is

purely a matter of taste whether physicians and researchers use the mobile application or the web application for this purpose. However, *EUPID Mobile* offers functionalities which can hardly be implemented in a pure web application, especially reading and writing NFC tags. Although there are possibilities for embedding QR Code readers via JavaScript, for PCs additional hardware such as a camera or QR Code scanners is always needed. Nowadays, modern smartphones provide the neccessary hardware compactly stored within the device, which can easily be accessed and made available on different operating systems with Apache Cordova.

Several tools for developing cross-platform applications are available. Apache Cordova was selected for developing *EUPID Mobile* for the following reasons:

- As shown in Figure 5.3, Apache Cordova was the leading tool for cross-platform development in 2015.
- Parts of the *EUPID Mobile* application were taken from the web application for patient registration programmed with AngularJS. The source code could be added to the project without translation into another programming language and with only minor adjustments.
- Apache Cordova projects were supported by Microsoft Visual Studio.

When debugging Apache Cordova projects attention must especially be payed to the *deviceready* function. In order to fix certain bugs, the source code for adding listeners (e.g. NFC) or declarations had to be moved inside this function. Not doing so did not cause error messages but resulted in unreleasable listeners or undefined behaviour of the application.

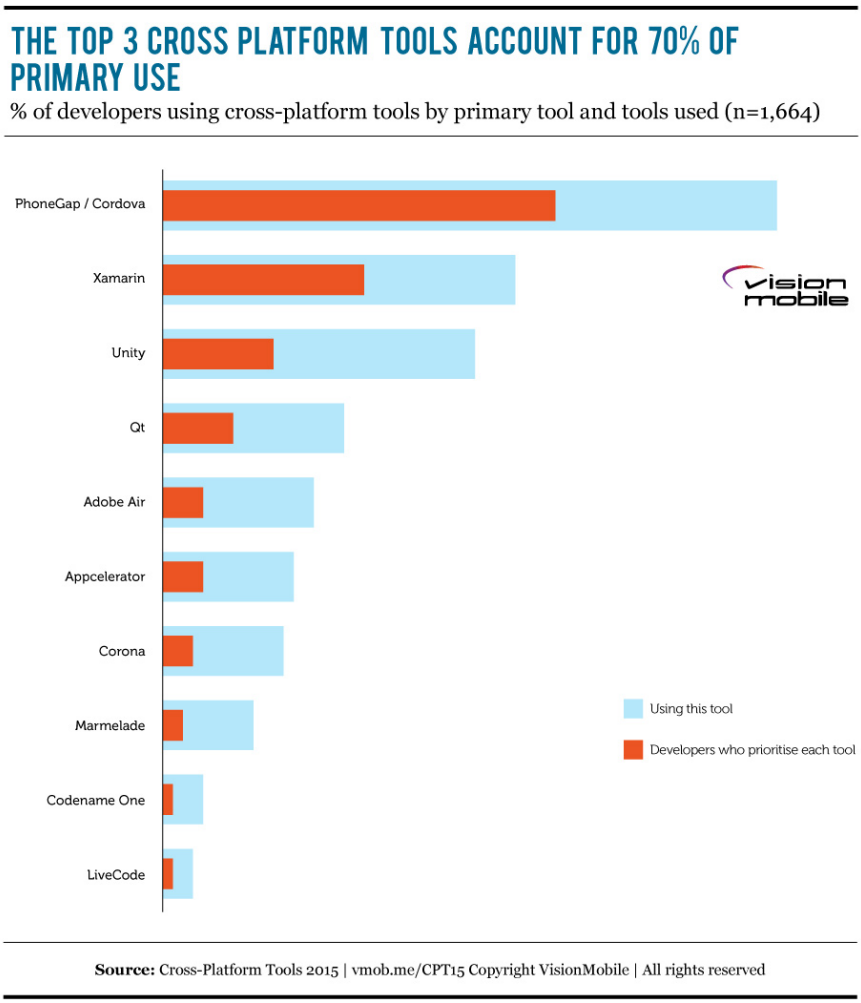

Figure 5.3: Leading tools for cross-platform development in 2015 [100].

5.1 Considerations and Future Fields of Work

During this thesis, two important milestones in introducing ePRO into the neuroblastoma research network could be reached. The existing *MoKi* telemonitoring system was connected to a back-end by introducing SOAP interfaces and adjusting the patient ID card. The *EUPID Mobile* application for patient registration was designed and fully implemented. However, future work is necessary.

The back-end service for creating PINs will have to be designed in detail and tested in subsequent developments.

The mapping table required by OpenClinica for patient identification will have to be implemented in future work and OpenClinica will have to be configured for communication with the mapping service.

Also the service for enrolling a patient in OpenClinica using a newly generated PSN will have to be developed.

For developmental purposes, every patient represented a separate data entry person in OpenClinica. However, the intended users of EDC systems are researchers and clinicians. One solution would be to establish an interface to the PostgreSQL database of OpenClinica and to directly enter the telemonitoring data via a web service.

In order to grant patients insight into their submitted health observations, functionalities for displaying and correcting their telemonitoring data could be added to the *MoKi* application.

Patients might lose their ID card or forget the PIN. For these cases, efficient workflows will have to be developed and implemented. Ideally, patients are empowered to autonomously request a new PIN or patient ID card using *MoKi*.

As described in Chapter 2.2 patients need to consent to merging their telemonitoring data with data from other sources. This could be handled by adding a particular paragraph to the informed consent covering their treatment.

By the end of this thesis, the *EUPID Mobile* application and the *MoKi* application were not available in the Apple App Store or the Google Play Store since proper beta tests will be necessary before releasing first versions.

Chapter 6

Conclusion

The present work represents the next important step for involving paediatric cancer patients and their parents, respectively, in clinical research by considering ePRO as a valuable source of information and including it in cancer research. This thesis outlines the development and prototypical implementation of a concept for integrating ePRO into the existing neuroblastoma research network by applying mobile technology and NFC. The *EUPID Mobile* application for patient registration was presented to the participants of the ENCCA Closing Conference. The user interface was considered to be simple and user-friendly and provided intuitive ways for reading the patient card ID via QR Code or NFC. Extending or adapting the functionalities was easily possible because of the application's modular design.

The mobile application *MoKi* of the existing telemonitoring system for paediatric cancer patients was successfully connected to the EDC system OpenClinica, which is currently used in neuroblastoma research. Future work will have to focus on fully implementing the back-end features. Allowing patients to be an active part in investigating their own disease increases patient empowerment, leads to a higher quality of research data and finally is expected to help with improving the treatment and survival of children suffering from cancer.

Appendix A

Bibliography

[1] STATISTIK AUSTRIA. Childhood and adolescent cancer. URL: http://www.statistik.at/web_en/statistics/PeopleSociety/health/cancer_incidence/childhood_and_adolescent_cancer/index.html, 2015. Accessed on 06.08.2015.

[2] Gatta, G. et al. Embryonal cancers in Europe. *European Journal of Cancer*, 48:1425–1433, 2012.

[3] National Cancer Institute. Neuroblastoma Treatment. URL: http://www.cancer.gov/types/neuroblastoma/patient/neuroblastoma-treatment-pdq, 2015. Accessed on 06.08.2015.

[4] Maris, J. M. Recent Advances in Neuroblastoma. *New England Journal of Medicine*, 362(23):2202–2211, 2010.

[5] Duregger, K. Telehealth in Paediatric Oncology. Master's thesis, FH JOANNEUM GmbH, Graz, Austria, 2015.

[6] American Cancer Society. Chemotherapy for neuroblastoma. URL: http://www.cancer.org/cancer/neuroblastoma/detailedguide/neuroblastoma-treating-chemotherapy, 2015. Accessed on 12.08.2015.

[7] Bond, M.C. and Pritchard, S. Understanding clinical trials in childhood cancer. *Paediatrics & Child Health*, 11(3):148–150, 2006.

[8] SIOPEN. What is a clinical trial?. URL: `http://www.siopen.org/siopen-studies/intro/what-is-a-clinical-trial`, 2015. Accessed on 06.08.2015.

[9] ClinicalTrials.gov. High Risk Neuroblastoma Study 1.7 of SIOP-Europe (SIOPEN). URL: `https://clinicaltrials.gov/ct2/show/NCT01704716`, 2015. Accessed on 06.08.2015.

[10] ClinicalTrials.gov. European Low and Intermediate Risk Neuroblastoma Protocol. URL: `https://clinicaltrials.gov/ct2/show/NCT01728155`, 2015. Accessed on 06.08.2015.

[11] ClinicalTrials.gov. Opsoclonus Myoclonus Syndrome/Dancing Eye Syndrome (OMS/DES) in Children With and Without Neuroblastoma (NBpos and NBneg). URL: `https://clinicaltrials.gov/ct2/show/NCT01868269`, 2015. Accessed on 06.08.2015.

[12] ClinicalTrials.gov. Long Term Continuous Infusion ch14.18/CHO Plus s.c. Aldesleukin (IL-2) (LTI). URL: `https://clinicaltrials.gov/ct2/show/NCT01701479`, 2015. Accessed on 06.08.2015.

[13] ClinicalTrials.gov. Bevacizumab, Irinotecan and Temozolomide for Relapsed or Refractory Neuroblastoma. URL: `https://clinicaltrials.gov/ct2/show/NCT01114555`, 2015. Accessed on 06.08.2015.

[14] European Commission. Rare diseases - sharing expertise to tackle rare disease. URL: `http://ec.europa.eu/health/rare_diseases/docs/2014_rare_diseases_informationsheet_en.pdf`, 2014.

[15] AIT Austrian Institute of Technology. SIOPEN-R-NET – Project overview. URL: `https://www.siopen-r-net.org/project_overview`, 2015. Accessed on 12.08.2015.

[16] Schreier, G. et al. A Web-based platform for interdisciplinary biomedical research. *Frontiers in Bioscience*, 14:2738–2746, 2009.

[17] European Network for Cancer Research in Children and Adolescents. Mission. URL: `http://www.encca.eu/project/mission`, 2015. Accessed on 13.08.2015.

[18] European Network for Cancer Research in Children and Adolescents. Scope. URL: http://www.encca.eu/project/scope, 2015. Accessed on 13.08.2015.

[19] Children's Cancer Research Institute. Mission. URL: http://science.ccri.at/the-institute/mission/, 2015. Accessed on 13.08.2015.

[20] European Society for Quality in Healthcare. About Us. URL: http://www.esqh.net/node/2, 2016. Accessed on 17.01.2016.

[21] AIT Austrian Institute of Technology. About Us. URL: http://www.ait.ac.at/about-us/?L=1, 2015. Accessed on 13.08.2015.

[22] Austrian Childhood Cancer Organisation. About Us. URL: http://www.kinderkrebshilfe.at/en/about-us, 2015. Accessed on 13.08.2015.

[23] Nitzlnader, M. Konzept eines IHE-basierten Medizinischen Forschungsnetzwerkes in der Cloud. Master's thesis, UMIT – Private University of Health Sciences, Medical Informatics and Technology, Austria, 2014.

[24] Hochedlinger, N. et al. Standardized Data Sharing in a Paediatric Oncology Research Network – A Proof-of-Concept Study. In *eHealth 2015 - Health Informatics Meets eHealth*, volume 212 of *Studies in Health Technology and Informatics*, pages 27–34. IOS Press, 2015.

[25] American Cancer Society. What is Wilms tumor?. URL: http://www.cancer.org/cancer/wilmstumor/detailedguide/wilms-tumor-what-is-wilms-tumor, 2015. Accessed on 24.08.2015.

[26] Institut national de la santé et de la recherche médicale. Nephroblastoma. URL: http://www.orpha.net/consor/cgi-bin/OC_Exp.php?Lng=DE&Expert=654, 2015. Accessed on 24.08.2015.

[27] Hochedlinger, N. Standardisierter Datenaustausch in medizinischen Forschungsnetzwerken mit Anwendung in der pädiatrischen Onkologie. Master's thesis, Graz University of Technology, 2015.

[28] Food and Drug Administration. Patient-Reported Outcome Measures: Use in Medical Product Development to Support Labeling Claims., 2009.

[29] Bennett, A. V. et al. Electronic Patient-Reported Outcome Systems in Oncology Clinical Practice. *CA: A Cancer Journal for Clinicians*, 62(5):337–347, 2012.

[30] Coons, S. J. et al. Recommendations on Evidence Needed to Support Measurement Equivalence between Electronic and Paper-Based Patient-Reported Outcome (PRO) Measures: ISPOR ePRO Good Research Practices Task Force Report. *Value in Health*, 12(4):419–424, 2009.

[31] Deshpande, P. R. et al. Patient-reported outcomes: A new era in clinical research. *Perspectives in Clinical Research*, 2(4):137–144, 2011.

[32] Wu, A. W. et al. Advances in the Use of Patient Reported Outcome Measures in Electronic Health Records, 2013.

[33] Wiesenauer, M. et al. Secondary Use of Clinical Data in Healthcare Providers – an Overview on Research, Regulatory and Ethical Requirements. In *Quality of Life through Quality of Information*, volume 180 of *Studies in Health Technology and Informatics*, pages 614–618. IOS Press, 2012.

[34] Clinical Data Interchange Standards Consortium. Operational Data Model. URL: `http://www.cdisc.org/odm`, 2015. Accessed on 31.12.2015.

[35] European Parliament and the Council of the European Union. Directive 95/46/EC, 1995.

[36] The Working Party on the Protection of Individuals with Regard to the Processing of Personal Data (Set up under Article 29 of Directive 95/46/EC). Opinion 05/2014 on Anonymisation Techniques, 2014.

[37] Nitzlnader, M. and Schreier, G. Patient Identity Management for Secondary Use of Biomedical Research Data in a Distributed Computing Environment. In *eHealth 2014 - Health Informatics Meets eHealth*, volume 198 of *Studies in Health Technology and Informatics*, pages 211–218. IOS Press, 2014.

[38] Karakasidis, A. and Verykios, V. S. Privacy Preserving Record Linkage Using Phonetic Codes. In *Proceedings of the 2009 Fourth Balkan Conference in Informatics*, pages 101–106, Thessaloniki, Greece, 2009.

[39] DENSO WAVE INCORPORATED. History of QR Code. URL: `http://www.qrcode.com/en/history/`, 2015. Accessed on 23.11.2015.

[40] DENSO WAVE INCORPORATED. QR Code Standardization. URL: `http://www.qrcode.com/en/about/standards.html`, 2015. Accessed on 23.11.2015.

[41] Wikipedia Community. Wikipedia - The Free Encyclopedia - QR code. URL: https://en.wikipedia.org/wiki/QR_code, 2015. Accessed on 24.11.2015.

[42] DENSO WAVE INCORPORATED. Types of QR Code. URL: http://www.qrcode.com/en/codes/, 2015. Accessed on 23.11.2015.

[43] DENSO WAVE INCORPORATED. Information capacity and versions of the QR Code. URL: http://www.qrcode.com/en/about/version.html, 2015. Accessed on 23.11.2015.

[44] DENSO WAVE INCORPORATED. Error Correction Feature. URL: http://www.qrcode.com/en/about/error_correction.html, 2015. Accessed on 24.11.2015.

[45] NearFieldCommunication.org. About Near Field Communication. URL: http://www.nearfieldcommunication.org/about-nfc.html, 2015. Accessed on 23.07.2015.

[46] NearFieldCommunication.org. How NFC Works. URL: http://www.nearfieldcommunication.org/how-it-works.html, 2015. Accessed on 23.07.2015.

[47] ISO. Information technology — Telecommunications and information exchange between systems — Near Field Communication — Interface and Protocol (NFCIP-1). ISO/IEC 18092:2013, International Organization for Standardization, Geneva, Switzerland, 2013.

[48] NearFieldCommunication.org. Near Field Communication Technology Standards. URL: http://www.nearfieldcommunication.org/technology.html, 2015. Accessed on 23.07.2015.

[49] NFC Forum. What It Does. URL: http://nfc-forum.org/what-is-nfc/what-it-does/, 2015. Accessed on 23.07.2015.

[50] NFC Forum. About Us. URL: http://nfc-forum.org/about-us/, 2015. Accessed on 02.12.2015.

[51] NFC Forum. NFC Forum Technical Specifications. URL: http://members.nfc-forum.org/specs/spec_list/, 2015. Accessed on 25.11.2015.

[52] Android Open Source Project. Developers - NdefRecord. URL: `http://developer.android.com/reference/android/nfc/NdefRecord.html`, 2015. Accessed on 26.11.2015.

[53] Krishna, A. V. G. et al. NFC-based Hospital Real-time Patient Management System. *International Journal of Engineering Trends and Technology (IJETT)*, 4(4):626–629, 2013.

[54] Bhadoriya, K. et al. NFC Based Healthcare System. *International Journal of Computer Science and Information Technologies*, 6(2):1364–1366, 2015.

[55] Razi, I. et al. Custom Made NFC Tags for Newly Born Babies at Hospital. *Science International*, 27(1):1463–1468, 2015.

[56] Quratulain, M. et al. NFC Tags-Based Notification System for Medical Appointments. *International Journal of Automation and Smart Technology*, 4(4):191–194, 2014.

[57] Meng, P. et al. Framework for Near-Field-Communication-Based Geo-Localization and Personalization for Android-Based Smartphones—Application in Hospital Environments. In *eHealth 2014 - Health Informatics Meets eHealth*, volume 198 of *Studies in Health Technology and Informatics*, pages 9–16. IOS Press, 2014.

[58] Meng, P. et al. Mobile Display of Information about Aggregated Antibiotic Resistance in the Hospital Setting Supported by Near Field Communication. In *pHealth 2014*, volume 200 of *Studies in Health Technology and Informatics*, pages 88–92. IOS Press, 2014.

[59] Ali, A. H. et al. Assessment and Comparison of Commonly used Wireless Technologies from Mobile payment Systems Perspective. *International Journal of Software Engineering and Its Applications*, 8(2):255–266, 2014.

[60] Duregger, K. et al. An mHealth System for Toxicity Monitoring of Paediatric Oncological Patients using Near Field Communication Technology. In *Proceedings of the 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 6848–6851, Milano, Italy, 2015.

[61] U.S. National Library of Medicine. C-reactive protein. URL: `http://www.nlm.nih.gov/medlineplus/ency/article/003356.htm`, 2015. Accessed on 23.07.2015.

[62] Walther, B. et al. Comparison of Electronic Data Capture (EDC) with the Standard Data Capture Method for Clinical Trial Data. *PLoS ONE*, 6(9), 2011.

[63] OpenClinica, LLC. Welcome to the world's fastest growing clinical trial software. URL: `https://openclinica.com/about`, 2015. Accessed on 06.11.2015.

[64] OpenClinica, LLC. OpenClinica Documentation. URL: `https://community.openclinica.com/documentation`, 2015. Accessed on 06.11.2015.

[65] OpenClinica, LLC. Self-guided OpenClinica Demo Instance. URL: `https://www.openclinica.com/self-guided-demo`, 2015. Accessed on 06.11.2015.

[66] The PostgreSQL Global Development Group. About. URL: `http://www.postgresql.org/about/`, 2015. Accessed on 06.11.2015.

[67] OpenClinica, LLC. Permissions for User Roles and Types. URL: `https://docs.openclinica.com/3.1/openclinica-user-guide/overview-user-types-and-roles/actionsbuttons-allowed-user-role`, 2015. Accessed on 06.11.2015.

[68] Franklin, J. et al. A partnership approach for Electronic Data Capture in small-scale clinical trials. *Journal of Biomedical Informatics*, 44(1):103–108, 2011.

[69] OpenClinica, LLC. SOAP Web Services - Introduction. URL: `https://docs.openclinica.com/3.1/technical-documents/openclinica-web-services-guide`, 2015. Accessed on 30.11.2015.

[70] Jarden, M. et al. Patient Activation through Counseling and Exercise – Acute Leukemia (PACE-AL) – a randomized controlled trial. *BMC Cancer*, 13:446, 2013.

[71] Lencioni, A. et al. An adverse event capture and management system for cancer studies. In *Proceedings of the 12th Annual MCBIOS Conference*, volume 16(13), Little Rock, Arkansas, USA, 2015.

[72] Quist, M. et al. "EXHALE": exercise as a strategy for rehabilitation in advanced stage lung cancer patients: a randomized clinical trial comparing the effects of 12 weeks supervised exercise intervention versus usual care for advanced stage lung cancer patients. *BMC Cancer*, 13, 2013.

[73] Stewart, E. et al. Development and characterization of a human orthotopic neuroblastoma xenograft. *Developmental Biology*, 2015.

[74] Otto, M. and Thornton, J. Bootstrap. URL: `http://getbootstrap.com/`, 2015. Accessed on 06.11.2015.

[75] Refsnes Data. HTML Responsive Web Design. URL: `http://www.w3schools.com/html/html_responsive.asp`, 2015. Accessed on 30.11.2015.

[76] Harris, P. A. et al. Research electronic data capture (REDCap) - A metadata-driven methodology and workflow process for providing translational research informatics support. *Journal of Biomedical Informatics*, 42(2):377–381, 2009.

[77] OpenMRS Inc. About OpenMRS. URL: `http://openmrs.org/about/`, 2015. Accessed on 30.11.2015.

[78] Clinovo. ClinCapture. URL: `http://www.clinovo.com/products/clincapture`, 2015. Accessed on 30.11.2015.

[79] CLIRINX Ltd. Academic Medical Research IT. URL: `http://www.clirinx.com/`, 2015. Accessed on 30.11.2015.

[80] OpenClinica, LLC. OpenClinica Participate. URL: `https://docs.openclinica.com/participate`, 2015. Accessed on 01.12.2015.

[81] OpenClinica, LLC. Connect Participants. URL: `https://docs.openclinica.com/participate/connect-participants`, 2015. Accessed on 01.12.2015.

[82] ccollins0601. OpenClinica Blog - Introducing OpenClinica Participate. URL: `http://blog.openclinica.com/2015/02/25/introducing-openclinica-participate/`, 2015. Accessed on 01.12.2015.

[83] van de Rijdt, M. GitHub Repository - enketo-express-oc. URL: `https://github.com/OpenClinica/enketo-express-oc`, 2015. Accessed on 01.12.2015.

[84] Enketo, LLC. ENKETO SMART PAPER. URL: `https://enketo.org/`, 2015. Accessed on 01.12.2015.

[85] Apache Software Foundation. Overview. URL: `https://cordova.apache.org/docs/en/latest/guide/overview/`, 2015. Accessed on 02.12.2015.

[86] Apache Software Foundation. Platform Support. URL: `https://cordova.apache.org/docs/en/latest/guide/support/index.html`, 2015. Accessed on 02.12.2015.

[87] IDC Research, Inc. Smartphone OS Market Share, 2015 Q2. URL: http://www.idc.com/prodserv/smartphone-os-market-share.jsp, 2015. Accessed on 02.12.2015.

[88] Apache Software Foundation. Get Started Fast. URL: https://cordova.apache.org/#getstarted, 2015. Accessed on 02.12.2015.

[89] Microsoft Corporation. Visual Studio Tools for Apache Cordova. URL: https://www.visualstudio.com/en-us/features/cordova-vs.aspx, 2015. Accessed on 02.12.2015.

[90] Apache Software Foundation. Cordova Plugins. URL: https://cordova.apache.org/plugins/, 2015. Accessed on 02.12.2015.

[91] Google Inc. ANGULARJS - HTML enhanced for web apps!. URL: https://angularjs.org/, 2015. Accessed on 09.12.2015.

[92] RACO Industries, LLC. QR Code Barcode Generator. URL: http://www.racoindustries.com/barcodegenerator/2d/qr-code.aspx, 2015. Accessed on 21.12.2015.

[93] Skjølberg, T. GitHub Repository - nfc-eclipse-plugin. URL: https://github.com/skjolber/nfc-eclipse-plugin, 2015. Accessed on 21.12.2015.

[94] Apache Software Foundation. Apache Tomcat. URL: http://tomcat.apache.org/, 2015. Accessed on 23.12.2015.

[95] OpenClinica, LLC. GitHub Repository - Wiki - Install OpenClinica. URL: https://github.com/OpenClinica/OpenClinica/wiki, 2015. Accessed on 23.12.2015.

[96] National Institute of Standards and Technology. Secure Hashing. URL: http://csrc.nist.gov/groups/ST/toolkit/secure_hashing.html, 2016. Accessed on 05.01.2016.

[97] OpenClinica, LLC. Activate and Build OpenClinica Participate for your Study. URL: https://docs.openclinica.com/participate/activate-openclinica-participate-your-study, 2015. Accessed on 07.01.2016.

[98] European Network for Cancer Research in Children and Adolescents. Work Packages. URL: `http://www.encca.eu/project/work-packages`, 2016. Accessed on 07.01.2016.

[99] Beishon, M. Passport to the future: Improving life for survivors of childhood cancer. *CancerWorld*, March-April 2014:46–49, 2014.

[100] VisionMobile Ltd. Cross-Platform Tools 2015 - What 8,000 developers and 185,000 apps reveal about the future of the market. URL: `http://www.visionmobile.com/product/cross-platform-tools-2015/`, 2016. Accessed on 07.01.2016.

[101] Apple Inc. Create Your Apple ID. URL: `https://appleid.apple.com/uk/#!&page=create`, 2015. Accessed on 02.01.2016.

[102] Microsoft Corporation. Remote Build. URL: `http://taco.tools/docs/remote-build.html`, 2016. Accessed on 05.01.2016.

[103] Microsoft Corporation. Install Visual Studio Tools for Apache Cordova. URL: `http://taco.visualstudio.com/en-us/docs/install-vs-tools-apache-cordova/`, 2015. Accessed on 10.12.2015.

Appendix B

Development Environment

B.1 Debugging Apache Cordova Applications on iOS Devices

For debugging iOS devices, an Apple MacBook (Apple Inc., Cupertino, California) had to be used. Below, the neccessary steps for setting up the debugging environment and troubleshooting information is provided for both approaches.

The MacBook Pro used had the following characteristics:

- OS X Yosemite v10.10.5
- 2,5GHz Intel Core i5
- 4GB 1600MHz DDR3

The iPhone 5c used had the following configuration:

- iOS 8.4
- 12,6 GB storage

B.1.1 Setup for Debugging with Apple Xcode

Apple Xcode (Apple Inc., Cupertino, California) is a development environment, which is among others used for programming iOS applications and debugging on attached iOS devices. It is available from the Mac App Store (Apple Inc., Cupertino, California). An active Apple developer account is not neccessary unless the application should be provided in the Apple App Store. For installing Xcode, an Apple ID is necessary which can be created on [101].

The following steps were neccessary to set up the MacBook for creating a Cordova project and debugging using `shell commands`:

1. Create Apple ID
2. Install Xcode 7.1.1 via Mac App Store
3. Install Node.js 5.0.0 from `https://nodejs.org/en/`
4. Install Cordova:
   `sudo npm install -g cordova`
5. Create a Cordova project named *HelloCordova*:
   `cordova create HelloCordova`
6. Adding platforms to the project:
   `cd HelloCordova`

   `cordova platform add ios`
7. Adding plugins to the project using GitHub-Repositories (e.g. NFC):
   `cordova plugin add https://github.com/chariotsolutions/phonegap-nfc.git`
8. If an existing Visual Studio project should be run in Xcode: Copy project files from the Visual Studio solution's *www*-Folder to the Xcode project's *www*-Folder
9. Rebuild Cordova project:
   `cd HelloCordova`

   `cordova build ios`
10. Start Xcode

11. Select *File* → *Open...*
12. Select *HelloCordova/platforms/ios/HelloCordova.xcodeproj*
13. Connect iPhone to MacBook
14. Select project *HelloCordova* and target *Generic iOS Device*. Enter Apple ID if requested. Click on the arrow to run the project:

In case the error messages `No non-expired provisioning profile` or `No matching provisioning profiles found` appear, as shown in Figure B.1, a valid *Bundle Identifier* needs to be entered. Then click on *Fix Issue*.

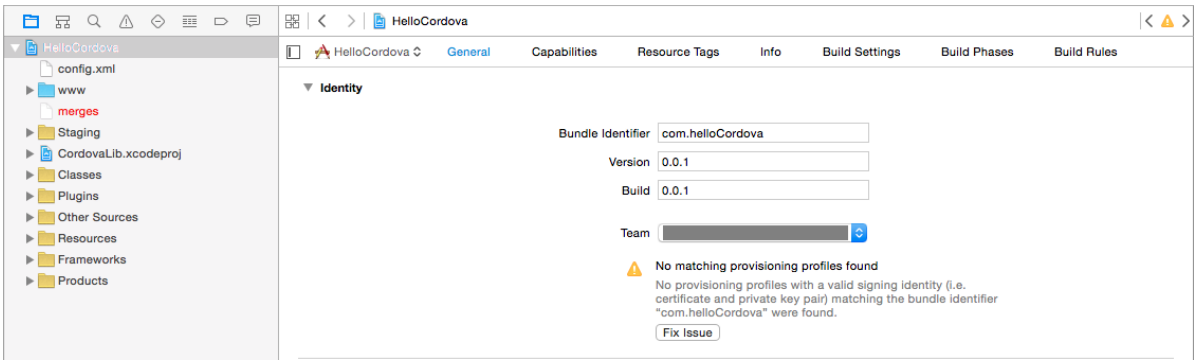

Figure B.1: Apple Xcode project error - No matching provisioning profiles found.

This first approach worked, but was not used for debugging the *EUPID Mobile* application since debugging in two different environments and working on two identical projects simultaneously was too cumbersome. The approach described below was used instead.

B.1.2 Setup for Debugging with Microsoft Visual Studio 2015

Visual Studio offers the possibility to run and debug solutions on iOS devices via a remote connection to a MacBook available within the same network.

Therefore first the Remote-Agent *remotebuild* had to be installed on the MacBook. Available commands for *remotebuild* are described under [102].

A more comprehensive description of the setup process and the configuration of Visual Studio can be found under [103].

Installing and running the Remote-Agent on the MacBook using `shell commands`:

1. Create Apple ID
2. Install Xcode 7.1.1 via Mac App Store
3. Install Node.js 5.0.0 from `https://nodejs.org/en/`
4. `xcode-select --install`
5. `sudo npm install -g remotebuild`
6. At the end of the installation process, the configuration details for Visual Studio are displayed as shown in the example in Figure B.2.

```
Use the following information in Visual Studio under Tools, Options,
Tools for Apache Cordova, Remote Agent Configuration to use this agent:

Enable remote iOS processing: True
Host: Dans-Mac-mini.local
Port: 1234
Secure mode: True
Security PIN: 687149
```

Figure B.2: Remote-agent configuration for remote debugging. Adapted from [103].

7. The displayed PIN is required in Visual Studio for debugging in safety mode.

   Run *remotebuild* in safety mode:
   `remotebuild`

   Run *remotebuild* without safety mode (no PIN is required in Visual Studio):
   `remotebuild --secure false`

8. How to stop the Remote-Agent:
   `<Ctrl> + <C>`

9. How to generate a new PIN for safety mode:
   `remotebuild certificates generate`

Configuring Visual Studio for remote debugging (run *remotebuild* first):

1. In Visual Studio go to *Extras* → *Options* → *Tools for Apache Cordova* → *Remote Agent Configuration*
2. Configure the Remote Agent according to the settings previously obtained during the setup of *remotebuild*. An example configuration is shown in Figure B.3.

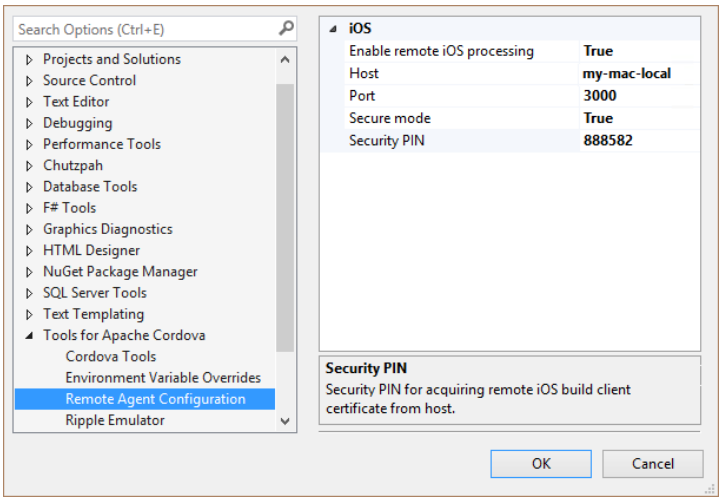

Figure B.3: Configuration of Visual Studio for remote debugging [103].

Configuring the iPhone 5c for remote debugging:

1. Connect the iPhone to the MacBook
2. Open the Apple Safari Browser on the MacBook and go to *Safari* → *Preferences...* → *Advanced*. Check *Show Develop menu in menu bar*.
3. On the iPhone go to *Settings* → *Safari* → *Advanced* and enable the *Web Inspector*
4. Disconnect the iPhone and reconnect it again

Debugging a Visual Studio Cordova project on an iPhone 5c:

1. Switch on the iPhone and connect it to the MacBook
2. Start *remotebuild* on the MacBook

3. Select *iOS* and *Remote Device* as target in Visual Studio according to Figure B.4.

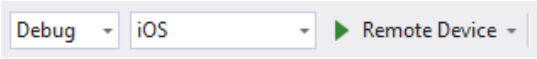

Figure B.4: Set target device in Visual Studio.

4. The home screen must be displayed on the iPhone. Click on *Remote Device* in Visual Studio to start the building process, which might take some time.

The debugger is attached to the build process only after *index.html* has been loaded. It might therefore be the case that errors and warnings occurring before are not detected by the debugger. In order to catch these, the command shown in Figure B.5 should be executed in the JavaScript console of Visual Studio while debugging is still in progress to restart the application.

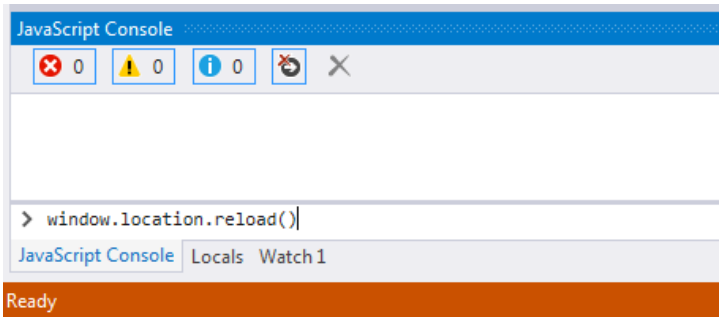

Figure B.5: Command to be executed in Visual Studio while debugging in order to find initialization errors.

If the error `Cannot read property 'localeCompare' of undefined` occurs on the MacBook terminal after starting the remote build, the following commands should be excuted on the MacBook:

1. `sudo npm cache clear`
2. `npm install -g npm@v3.3.7`

In case `Unable to use Cordova < 5.4.0 with Node >= 5.0.0.` is displayed in Visual Studio, install an older version of Node.js (e.g. version 0.12.7).

It is possible that the error `no non-expired provisioning profile` appears in Visual Studio. In this case the provisioning profile has to be adjusted manually:

1. Open *config.xml*
2. Go to *Common* → *Package Name*
3. Enter the provisioning profile defined for the project (see also Chapter B.1.1)

The bug causing the error `The file "ADALiOS.entitlements" couldn't be opened because there is no such file.` should be fixed by now in Visual Studio. However, if the error occurs:

1. Go to *Plugins* → *cordova-plugin-ms-adal* → *scripts*
2. Open *configureEntitlementsIos.js*
3. Replace:

```
var entitlementsFile = path.join("\"", projName, "Resources/
    ADALiOS.entitlements\"");
```

   by:

```
var entitlementsFile = path.join(projName, "Resources/ADALiOS.
    entitlements");
entitlementsFile = "\"" + entitlementsFile + "\"";
```

Sometimes changes in the source code are not applied in the following building process for some reason. In this case:

1. Delete the application from the iPhone
2. Terminate the remote session via `<Ctrl> + <C>`. This command deletes the folder containing the build files.
3. Restart *remotebuild*
4. Building the project in Visual Studio

Appendix C

Transmitting Data to OpenClinica

C.1 SOAP Requests

Listing C.1: SOAP request - entering data in a scheduled C-reactive protein event.

```
<soapenv:Body>
<v1:importRequest>
<ODM>
<ClinicalData StudyOID="#OCSTUDYOID" MetaDataVersionOID="v1.0.0">
<SubjectData SubjectKey="#OCSTUDYSUBJECTOID">
<StudyEventData StudyEventOID="SE_CRP" StudyEventRepeatKey="#
    STUDYEVENTORDINAL">
<FormData FormOID="F_CRF_CRP_V1">
<ItemGroupData ItemGroupOID="IG_CRF_C_LISTOFMEASUREMENTS"
    ItemGroupRepeatKey="1" TransactionType="Insert">
<ItemData ItemOID="I_CRF_C_CRPDATE"
Value="#CRPDATE"/>
<ItemData ItemOID="I_CRF_C_CRPTIME"
Value="#CRPTIME"/>
<ItemData ItemOID="I_CRF_C_CRP"
Value="#CRPVALUE"/>
</ItemGroupData>
</FormData>
</StudyEventData>
</SubjectData>
</ClinicalData>
</ODM>
</v1:importRequest>
</soapenv:Body>
```

Listing C.2: SOAP request - entering data in a scheduled blood pressure event.

```
<soapenv:Body>
  <v1:importRequest>
    <ODM>
      <ClinicalData StudyOID="#OCSTUDYOID" MetaDataVersionOID="v1.0.0">
        <SubjectData SubjectKey="#OCSTUDYSUBJECTOID">
          <StudyEventData StudyEventOID="SE_BLOODPRESSURE"
              StudyEventRepeatKey="#STUDYEVENTORDINAL">
            <FormData FormOID="F_CRF_BLPR_V1">
              <ItemGroupData ItemGroupOID="IG_CRF_B_LISTOFMEASUREMENTS"
                  ItemGroupRepeatKey="1" TransactionType="Insert">
                <ItemData ItemOID="I_CRF_B_BPDATE"
                Value="#BLOODPRESSUREDATE"/>
                <ItemData ItemOID="I_CRF_B_BPTIME"
                Value="#BLOODPRESSURETIME"/>
                <ItemData ItemOID="I_CRF_B_BPSYSTOLE"
                Value="#SYSTOLEVALUE"/>
                <ItemData ItemOID="I_CRF_B_BPDIASTOLE"
                Value="#DIASTOLEVALUE"/>
                <ItemData ItemOID="I_CRF_B_BPPULSE"
                Value="#PULSEVALUE"/>
              </ItemGroupData>
            </FormData>
          </StudyEventData>
        </SubjectData>
      </ClinicalData>
    </ODM>
  </v1:importRequest>
</soapenv:Body>
```

Listing C.3: SOAP request - entering data in a scheduled nausea event.

```
<soapenv:Body>
  <v1:importRequest>
    <ODM>
      <ClinicalData StudyOID="#OCSTUDYOID" MetaDataVersionOID="v1.0.0">
        <SubjectData SubjectKey="#OCSTUDYSUBJECTOID">
          <StudyEventData StudyEventOID="SE_NAUSEA" StudyEventRepeatKey
              ="#STUDYEVENTORDINAL">
            <FormData FormOID="F_CRF_NAUSEA_V1">
              <ItemGroupData ItemGroupOID="IG_CRF_N_LISTOFMEASUREMENTS"
                   ItemGroupRepeatKey="1" TransactionType="Insert">
                <ItemData ItemOID="I_CRF_N_NAUSEADATE"
                Value="#NAUSEADATE"/>
                <ItemData ItemOID="I_CRF_N_NAUSEATIME"
                Value="#NAUSEATIME"/>
                <ItemData ItemOID="I_CRF_N_NAUSEAVALUE"
                Value="#NAUSEAVALUE"/>
              </ItemGroupData>
            </FormData>
          </StudyEventData>
        </SubjectData>
      </ClinicalData>
    </ODM>
  </v1:importRequest>
</soapenv:Body>
```

Listing C.4: SOAP request - entering data in a scheduled pain event.

```
<soapenv:Body>
  <v1:importRequest>
    <ODM>
      <ClinicalData StudyOID="#OCSTUDYOID" MetaDataVersionOID="v1.0.0">
        <SubjectData SubjectKey="#OCSTUDYSUBJECTOID">
          <StudyEventData StudyEventOID="SE_PAIN" StudyEventRepeatKey="
              #STUDYEVENTORDINAL">
            <FormData FormOID="F_CRF_PAIN_V1">
              <ItemGroupData ItemGroupOID="IG_PROPA_LISTOFMEASUREMENTS"
                   ItemGroupRepeatKey="1" TransactionType="Insert">
                <ItemData ItemOID="I_CRF_P_PAINDATE"
                Value="#PAINDATE"/>
                <ItemData ItemOID="I_CRF_P_PAINTIME"
                Value="#PAINTIME"/>
                <ItemData ItemOID="I_CRF_P_PAINVALUE"
                Value="#PAINVALUE"/>
              </ItemGroupData>
            </FormData>
          </StudyEventData>
        </SubjectData>
      </ClinicalData>
    </ODM>
  </v1:importRequest>
</soapenv:Body>
```

Listing C.5: SOAP request - entering data in a scheduled skin event.

```
<soapenv:Body>
  <v1:importRequest>
    <ODM>
      <ClinicalData StudyOID="#OCSTUDYOID" MetaDataVersionOID="v1.0.0">
        <SubjectData SubjectKey="#OCSTUDYSUBJECTOID">
          <StudyEventData StudyEventOID="SE_SKINALTERATION"
              StudyEventRepeatKey="#STUDYEVENTORDINAL">
            <FormData FormOID="F_CRF_SKINFULL_V1">
              <ItemGroupData ItemGroupOID="IG_CRF_S_LISTOFMEASUREMENTS"
                  ItemGroupRepeatKey="1" TransactionType="Insert">
                <ItemData ItemOID="I_CRF_S_SKINDATE"
                Value="#SKINDATE"/>
                <ItemData ItemOID="I_CRF_S_SKINTIME"
                Value="#SKINTIME"/>
                <ItemData ItemOID="I_CRF_S_SKINIMAGE"
                Value="#SKINIMAGE"/>
                <ItemData ItemOID="I_CRF_S_SKINTYPE"
                Value="#SKINTYPE"/>
                <ItemData ItemOID="I_CRF_S_SKINLOC"
                Value="#SKINLOC"/>
                <ItemData ItemOID="I_CRF_S_SKINCARDRED"
                Value="#SKINRED"/>
                <ItemData ItemOID="I_CRF_S_SKINCARDSWE"
                Value="#SKINSWE"/>
                <ItemData ItemOID="I_CRF_S_SKINCARDWAR"
                Value="#SKINWAR"/>
                <ItemData ItemOID="I_CRF_S_SKINCARDPAI"
                Value="#SKINPAI"/>
                <ItemData ItemOID="I_CRF_S_SKINCOMMENT"
                Value="#SKINCOM"/>
              </ItemGroupData>
            </FormData>
          </StudyEventData>
        </SubjectData>
      </ClinicalData>
    </ODM>
  </v1:importRequest>
</soapenv:Body>
```

Listing C.6: SOAP request - entering data in a scheduled temperature event.

```
<soapenv:Body>
  <v1:importRequest>
    <ODM>
      <ClinicalData StudyOID="#OCSTUDYOID" MetaDataVersionOID="v1.0.0">
        <SubjectData SubjectKey="#OCSTUDYSUBJECTOID">
          <StudyEventData StudyEventOID="SE_TEMPERATURE"
              StudyEventRepeatKey="#STUDYEVENTORDINAL">
            <FormData FormOID="F_CRF_TEMP_V1">
              <ItemGroupData ItemGroupOID="IG_CRF_T_LISTOFMEASUREMENTS"
                  ItemGroupRepeatKey="1" TransactionType="Insert">
                <ItemData ItemOID="I_CRF_T_TEMPDATE"
                Value="#TEMPDATE"/>
                <ItemData ItemOID="I_CRF_T_TEMPTIME"
                Value="#TEMPTIME"/>
                <ItemData ItemOID="I_CRF_T_TEMP"
                Value="#TEMPVALUE"/>
              </ItemGroupData>
            </FormData>
          </StudyEventData>
        </SubjectData>
      </ClinicalData>
    </ODM>
  </v1:importRequest>
</soapenv:Body>
```

Listing C.7: SOAP request - entering data in a scheduled white blood cell event.

```
<soapenv:Body>
  <v1:importRequest>
    <ODM>
      <ClinicalData StudyOID="#OCSTUDYOID" MetaDataVersionOID="v1.0.0">
        <SubjectData SubjectKey="#OCSTUDYSUBJECTOID">
          <StudyEventData StudyEventOID="SE_WBC" StudyEventRepeatKey="#
              STUDYEVENTORDINAL">
            <FormData FormOID="F_CRF_WBC_V1">
              <ItemGroupData ItemGroupOID="IG_CRF_W_LISTOFMEASUREMENTS"
                  ItemGroupRepeatKey="1" TransactionType="Insert">
                <ItemData ItemOID="I_CRF_W_WBCDATE"
                Value="#WBCDATE"/>
                <ItemData ItemOID="I_CRF_W_WBCTIME"
                Value="#WBCTIME"/>
                <ItemData ItemOID="I_CRF_W_WBC"
                Value="#WBCVALUE"/>
              </ItemGroupData>
            </FormData>
          </StudyEventData>
        </SubjectData>
      </ClinicalData>
    </ODM>
  </v1:importRequest>
</soapenv:Body>
```

Listing C.8: SOAP request - entering data in a scheduled weight event.

```
<soapenv:Body>
  <v1:importRequest>
    <ODM>
      <ClinicalData StudyOID="#OCSTUDYOID" MetaDataVersionOID="v1.0.0">
        <SubjectData SubjectKey="#OCSTUDYSUBJECTOID">
          <StudyEventData StudyEventOID="SE_WEIGHT" StudyEventRepeatKey
              ="#STUDYEVENTORDINAL">
            <FormData FormOID="F_CRF_WEIGHT1_V1">
              <ItemGroupData ItemGroupOID="
                  IG_CRF_W_LISTOFMEASUREMENTS_3478" ItemGroupRepeatKey
                  ="1" TransactionType="Insert">
                <ItemData ItemOID="I_CRF_W_WEIGHTDATE"
                Value="#WEIGHTDATE"/>
                <ItemData ItemOID="I_CRF_W_WEIGHTTIME"
                Value="#WEIGHTTIME"/>
                <ItemData ItemOID="I_CRF_W_WEIGHT"
                Value="#WEIGHTVALUE"/>
              </ItemGroupData>
            </FormData>
          </StudyEventData>
        </SubjectData>
      </ClinicalData>
    </ODM>
  </v1:importRequest>
</soapenv:Body>
```

C.2 Case Report Forms

Blood Pressure			
Item	Unit	Response Type	Data Type
BPDate		text	Date
BPTime		text	Character string
BPSystole	mmHg	text	Integer
BPDiastole	mmHg	text	Integer
BPPulse	1/min	text	Integer

C-Reactive Protein			
Item	Unit	Response Type	Data Type
CRPDate		text	Date
CRPTime		text	Character string
CRP	mg/L	text	Real

Nausea			
Item	Unit	Response Type	Data Type
NauseaDate		text	Date
NauseaTime		text	Character string
NauseaValue		radio (None, Little, Little More, Even More, Whole Lot, Vomiting)	Integer

Pain			
Item	Unit	Response Type	Data Type
PainDate		text	Date
PainTime		text	Character string
PainValue		radio (None, Little, Little More, Even More, Whole Lot, Maximum)	Integer

Skin			
Item	Unit	Response Type	Data Type
SkinDate		text	Date
SkinTime		text	Character string
SkinType		single select (Rash, Catheter side)	Character string
SkinLoc		single select (Not specified, Head, Chest, Left leg, Right leg, Left arm, Right arm, Back)	Integer
SkinCardRed		radio (Not selected, Yes, No, Don't know)	Integer
SkinCardSwe		radio (Not selected, Yes, No, Don't know)	Integer
SkinCardWar		radio (Not selected, Yes, No, Don't know)	Integer
SkinCardPai		radio (Not selected, Yes, No, Don't know)	Integer
SkinComment		textarea	Character string
SkinImage		file	File
Temperature			
Item	Unit	Response Type	Data Type
TempDate		text	Date
TempTime		text	Character string
Temp	°C	text	Real
White Blood Cell Count			
Item	Unit	Response Type	Data Type
WBCDate		text	Date
WBCTime		text	Character string
WBC	$1/\mu$L	text	Real
Weight			
Item	Unit	Response Type	Data Type
WeightDate		text	Date
WeightTime		text	Character string
Weight	kg	text	Real
Wellbeing			
Item	Unit	Response Type	Data Type
WellbDate		text	Date
WellbTime		text	Character string
WellbV		radio (good,medium,bad)	Integer

Table C.1: Properties of the used Case Report Forms.