

**TELEMEDIZIN ÜBER EINE MOBILE APPLIKATION IN
UNTERVERSORGTE ENDEMIEGEBIETEN**

Eine Masterarbeit von
DZANAN MLIVO, BSC

betreut von
Dr. Christof Pabinger, Medizinische Universität Graz
und
FH-Prof. Ing. Dipl.-Ing. Dr. techn. Harald Burgsteiner

und eingereicht am
Studiengang eHealth
der FH JOANNEUM Graz
Graz University of Applied Sciences

zur teilweisen Erfüllung der Anforderungen
zur Erlangung des akademischen Grades
Master of Science (MSc)

September 2014
(REMOVE THIS FROM THE FINAL DOCUMENT:
version from 2. September 2014)

"An dieser Stelle bedanke ich mich für die Erfahrungen, die ich während meines Studiums machen durfte"

Abstract

Hip dysplasia is the most common pediatric orthopedic deformity. The implied destruction of the hip joint leads to further consequences. Many operations (costs) and the decreasing quality of life are the biggest problems. The solution for this problem is hip sonography (screening). The screening is a simple process with which a simple treatment can be initiated. Bandages or abduction pants are used to fix the newborns hips for a few weeks of treatment. In Austria, Germany, Switzerland and the Czech Republic hip screenings led to a significant reduction in operations and disabilities (limping and pain). However, in other countries, no comprehensive screenings of neonatal hips are performed. This leads to an insufficient supply of expertise and thereby a large number of destroyed hips. Furthermore, there are endemic areas, regions where the incidence of hip dysplasia is increased. A mobile application (app) provides a good complement to the existing web portal xMEDx. Images taken with the app were validated in test series. The validation showed that the captured images are quite suitable for evaluation through experts (inclination up to 5°).

Zusammenfassung

Hüftdysplasie ist die häufigste kinderorthopädische Fehlstellung. Die dadurch implizierte Zerstörung des Hüftgelenks führt zu weiteren Konsequenzen. Viele Operationen (Kosten) und der Verlust an Lebensqualität sind die größten Probleme. Die Lösung für das Problem liegt in der Hüftsonographie (Screening). Das Screening ist ein einfaches Verfahren mit dem eine einfache Therapie eingeleitet werden kann. Dazu werden für einige Wochen Bandagen oder Spreizhosen am Neugeborenen angebracht. In Österreich, Deutschland, der Schweiz und Tschechien führte das Screening zu einer deutlichen Reduktion an Operationen und Behinderungen (Hinken und Schmerzen). Jedoch werden in anderen Ländern keine flächendeckenden Screenings der Neugeborenenhüfte durchgeführt. Dies führt zu einer Unterversorgung an Fachwissen und dadurch zu einer Vielzahl von deformierten Hüften. Des weiteren gibt es Endemiegebiete, Regionen in denen die Inzidenz für Hüftdysplasie erhöht ist. Eine mobile Applikation (App) bietet eine gute Ergänzung zu dem bestehenden Webportal von xMEDx. Die mit der App aufgenommenen Bilder wurden in Testserien validiert. Die Validierung zeigte, dass bei Neigungen von bis zu 5° die aufgenommenen Bilder durchaus zur Beurteilung durch Experten geeignet sind.

Inhaltsverzeichnis

Abstract	iii
Zusammenfassung	iv
Abbildungsverzeichnis	vii
Tabellenverzeichnis	viii
1 Einleitung	1
1.1 Informations- und Kommunikationstechnologie im Gesundheitswesen . . .	1
1.1.1 Telemedizin	1
1.1.2 eHealth	3
1.1.3 Die Elektronische Gesundheitsakte	4
1.1.4 mHealth	4
1.1.5 Rechtliche Rahmenbedingungen	5
1.2 Ziele dieser Masterarbeit	5
1.2.1 Inzidenzen und Länder	6
1.2.2 Mobile Applikation	6
1.2.3 Validierung	6
2 Material und Methoden	7
2.1 Hüftgelenksfehlstellungen	7
2.1.1 Epidemiologie der Hüftdysplasie	8
2.1.2 Krankheitsmechanismus und Risikofaktoren	9
2.1.3 Nicht-Operative Behandlung der Hüftdysplasie	9
2.2 Hüftsonografie und Diagnose	10
2.2.1 Anatomische Identifizierung	11
2.2.2 Klassifikation und Typeinteilung nach Graf	12
2.2.3 Detaillierte Typisierung	13
2.3 Kunden	16

3	Android, Apps und Smartphones	18
3.1	Betriebssysteme und Smartphones	18
3.1.1	Allgemeines über Betriebssysteme	18
3.1.2	Betriebssysteme für Smartphones	20
3.1.3	Überblick über Smartphones	23
3.1.4	Apps für Smartphones	25
3.2	Android	27
3.2.1	Android Software Stack	28
3.2.2	Grundlegende Komponenten	30
3.2.3	Aktivitäten	31
3.2.4	Sensoren	34
3.2.5	AsyncTask	35
4	Resultate	37
4.1	Hüftdysplasie: Inzidenzen und Länder	37
4.2	Prototyp der App	39
4.2.1	Screenshots und Beschreibung der App	39
4.3	Datenvalidierung	51
4.3.1	Das Test-Sonogramm	52
4.3.2	Aufnahme der Sonogramme	53
4.3.3	Lage Sonogramme	54
4.3.4	Entzerrte Sonogramme	56
4.4	Zusammenfassung	58
5	Diskussion	59
	Literaturverzeichnis	60
	Internetquellen	61
	Eidesstattliche Erklärung	64

Abbildungsverzeichnis

1.1	IKT im Gesundheitswesen	2
2.1	Einteilung der Befunde nach Graf	8
2.2	Ausreifungsbehandlungen von Hüftdysplasie	10
2.3	Anatomie der Säuglingshüfte	12
2.4	Vermessung der Neugeborenenhüfte	15
2.5	Sonometer nach Graf	16
3.1	Einordnung von Betriebssystemen	19
3.2	Prognose mobile Betriebssysteme 2018	21
3.3	Prognose für den Verkauf von Smarten Geräten	24
3.4	Anzahl der Apps in den jeweiligen Stores	26
3.5	Distribution aktueller Android Versionen	27
3.6	Android Software Stack im Überblick	29
3.7	Lebenszyklus einer Aktivität	32
3.8	Koordinatensystem der Android Sensoren	35
4.1	Hüftdysplasie: Inzidenzen und Länder	38
4.2	Prototyp Login	40
4.3	Prototyp Start	42
4.4	Prototyp Camera	44
4.5	Prototyp Check	45
4.6	Prototyp Draw 1	46
4.7	Prototyp Draw 2	47
4.8	Prototyp Expert	49
4.9	Prototyp Mail	50
4.10	Test-Sonogramm für die Auswertung	52
4.11	Schema des Smartphones und Neigungswinkel	53
4.12	Korrektur der Perspektive durch Act Scanner	56

Tabellenverzeichnis

2.1	Typeinteilung nach Formausprägung	14
4.1	Reale Winkel des Sonogramms	52
4.2	Winkel der 30° Sonogramme	54
4.3	Winkel der 20° Sonogramme	54
4.4	Winkel der 10° Sonogramme	55
4.5	Winkel der 5° Sonogramme	55
4.6	Winkel der 0° Sonogramme	55
4.7	Winkel der 30° entzerrten Sonogramme	57
4.8	Winkel der 20° entzerrten Sonogramme	57
4.9	Winkel der 10° entzerrten Sonogramme	58
4.10	Winkel der 5° entzerrten Sonogramme	58

Kapitel 1

Einleitung

1.1 Informations- und Kommunikationstechnologie im Gesundheitswesen

Der Begriff Telemedizin beschreibt die Verwendung von Informations- und Kommunikationstechnologien (IKT) um Leistungen im Gesundheitswesen zu unterstützen oder bereitzustellen [BMG, 2014b]. Des weiteren bedeutet Telemedizin, dass Patientinnen bzw. Patienten und Gesundheitsdienstleister räumlich sowie auch zeitlich getrennt sind. Unter Gesundheitsdienstleister (GDA), fallen unter anderem Ärzte, Apotheken, Krankenhäuser. Voraussetzung für die Nutzung von Telemedizin ist die gesicherte Übertragung von medizinischen Daten. Diese Daten lassen sich in unterschiedlichen Formaten transportieren. Medizinische Daten können als Text, Ton oder Bildmaterial versendet werden. Durch die Technik elektronischer Kommunikation können somit Barrieren überwunden werden, die zwischen Patienten und GDA's auf natürlichem Weg entstehen.

1.1.1 Telemedizin

Nach dem Österreichischen Bundesministerium für Gesundheit (BMG) kann Telemedizin in unterschiedliche Kategorien eingeteilt werden. Das BMG teilt Telemedizin in folgende Gebiete auf. Bei **Telemonitoring** (z.Bsp.: Überwachung von Neugeborenen) handelt es sich um die Überwachung des Gesundheitszustandes eines Patienten welcher über eine räumliche Distanz vom GDA getrennt ist. **Teletherapie** bedeutet das ein GDA auf

¹Günter Schreier, interaktive Vorlesung über "Mobile Anwendung und Telemedizin", FH Joanneum Gesellschaft mbH, Wintersemester 2012/13

Zusammenhang von Medizin und Informationstechnologie

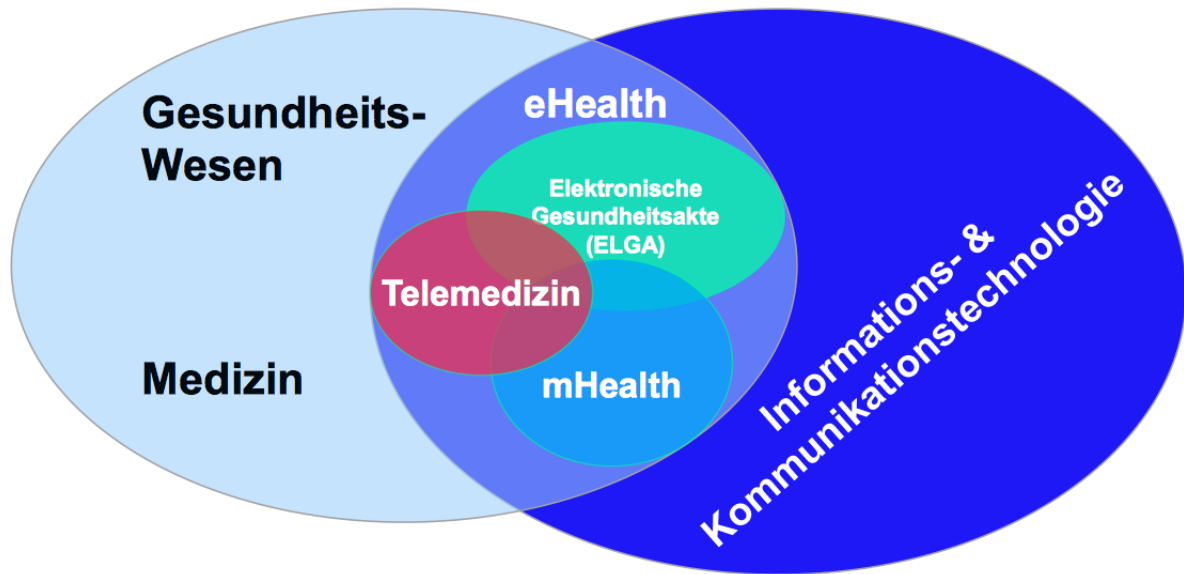


Abbildung 1.1: In dieser Abbildung werden die Gemeinsamkeiten der unterschiedlichen Bereich der IKT in Verbindung mit dem Gesundheitswesen dargestellt.¹

die Therapie eines entfernten Patienten aktiv Einfluss nehmen kann. **Telekonzile** dienen einem behandelnden GDA um die Meinung (Zweitmeinung) eines anderen Gesundheitsdiensteanbieter einzuholen z.Bsp.: Fernbefundung in der Radiologie. **Telekonferenzen** werden dazu verwendet das ein entfernter GDA in die laufende medizinische Behandlung einbezogen wird.

Unter **Store and forward** versteht man dass medizinische Daten (Bilder, Signale) aufgezeichnet werden und zu einem späteren Zeitpunkt an einen Spezialisten weitergeleitet werden. Der Spezialist (GDA) kann die erhaltenen medizinischen Daten dann Orts- und Zeitunabhängig begutachten. Dadurch gäbe es die Möglichkeit je nach Priorität des Falls die Patienten in einer gewichteten Warteschlange zu bearbeiten. **Remote Monitoring** wird für das Beobachten und Bewältigen von chronischen Erkrankungen verwendet. Herz-erkrankungen, Diabetes und andere chronische Erkrankungen fallen unter diesen Bereich des Telemonitoring. Ein Beispiel dafür ist Diabetes: Der Blutzucker wird vom Patienten gemessen. Das Zuckermessgerät sendet die Daten an eine Basisstation und diese überträgt die Werte an den GDA welcher die Daten begutachten kann. Unter Real-Time Interactive versteht man, das die Kommunikation zwischen GDA und Patienten direkt stattfindet. Dies kann durch Telefon, Videotelefonie, Hausbesuche sowie über das Internet erfolgen.

Im Rahmen eines Österreichischen Pilotprojekts (24 Stunden Telemedizin [ORF, 2014]) soll ab nächstem Jahr ein Telemedizinischer Service angeboten werden. Dieser Service soll Patienten und Patientinnen Auskunft geben und rund um die Uhr zur Verfügung stehen (Telefon und Internet). Bis jetzt haben sich drei Österreichische Bundesländer (Wien, Niederösterreich und Vorarlberg) dazu verpflichtet. Das Ziel ist es die Entlastung der Spitalsambulanzen. Es soll in erster Linie ein Krankenpfleger oder Krankenpflegerin den Patienten Auskunft geben und erste notwendige Daten nach einem standardisierten Vorgang erheben. Falls aber ärztliche Expertise gefragt sein sollte kann diese von Allgemeinmedizinern im Back-Office eingeholt werden. Ob dieser Pilotversuch Früchte trägt entscheidet aber letztendlich der Patient und die Patientin.

1.1.2 eHealth

Über die grundsätzliche Definition des Begriffs eHealth herrscht Einigkeit. Aber was der Begriff im Detail alles beinhaltet ist unzureichend geklärt, da eHealth sehr viele Teilbereiche der Gesundheitsversorgung und Technik anspricht. Beispiele für eHealth wären unter anderem die elektronische Krankenversicherungskarte, elektronische Gesundheitsakte, Medikamentenübersicht sowie die Telemedizin. Die sichere Übertragung medizinischer Daten, welche für die Prävention, Diagnose, Behandlung und weitere Betreuung der Patienten und Patientinnen, ist dabei Voraussetzung. Das Österreichische Bundesministerium für Gesundheit (BMG) beschreibt eHealth folgendermaßen [BMG, 2014a]:

"Der Begriff der elektronischen Gesundheitsdienste („e-Health“) bezeichnet den Einsatz von Informations- und Kommunikationstechnologien (IKT) in gesundheitsbezogenen Produkten, Dienstleistungen und Prozessen. Verbunden mit strukturellen Änderungen im Gesundheitssystem trägt e-Health zur Verbesserung der Gesundheit der Bürgerinnen und Bürger sowie der Effizienz bei der Erbringung von Gesundheitsdienstleistungen bei."

Des weiteren definiert die World Health Organization (WHO) eHealth indem sie auch eCommerce, eBusiness sowie Public Health in ihrer Definition anschneiden [WHO, 2014]:

"E-health is the transfer of health resources and health care by electronic means. It encompasses three main areas:

- *The delivery of health information, for health professionals and health consumers, through the Internet and telecommunications.*

- *Using the power of IT and e-commerce to improve public health services, e.g. through the education and training of health workers.*
- *The use of e-commerce and e-business practices in health systems management."*

1.1.3 Die Elektronische Gesundheitsakte

Die elektronische Gesundheitsakte (ELGA), soll Patienten und Patientinnen sowie allen Gesundheitsdienstleistern (GDA) z.Bsp.: Spitälern, niedergelassenen Ärzten und Ärztinnen, Apotheken oder auch Pflegeheimen den Orts- und Zeitunabhängigen Zugang auf Gesundheitsdaten ermöglichen. Unter anderem können Befunde aus Labor und Radiologie sowie Entlassungsbriefe und die aktuellen Medikamente von Patienten abgerufen werden. Dadurch soll eine verbesserte Grundlage (Informationsfluss) für die weitere Behandlung und Diagnostik entstehen [Gesundheit.gv.at, 2014]. Das interessante an ELGA ist das es keinen zentralen Datenspeicher gibt, in welchen Gesundheitsrelevante Daten gespeichert werden. Das bedeutet das alle Daten dezentral, also dort wo sie erstellt worden sind (z.Bsp.: Krankenhaus oder Arzt/Ärztin), gespeichert werden und ELGA lediglich auf diese schon erstellten Dokumente verweist (referenziert).

1.1.4 mHealth

mHealth ist ein Teilbereich des vorher besprochenen Bereiches eHealth. Die WHO schreibt auch das es zur Zeit keine standardisierte Definition des mHealth Begriffs gibt. Im Grunde kann mHealth oder auch mobile Health als Unterstützung der medizinischen Versorgung durch mobile Geräte (Mobiltelefone, Smartphones, Monitoring Systeme usw.) bezeichnet werden. Bei mHealth steht somit der Patient mit chronischen Erkrankung im Mittelpunkt. Durch die Freiheit der Zeit- und Ortsungebundenheit können Therapien für chronisch erkrankte Patienten die an z.Bsp.: Diabetes oder Herzerkrankungen leiden optimiert werden.

mHealth Kategorien nach Nutzern

Da mHealth ein Teilgebiet der Telemedizin ist deckt es ein breites Spektrum an Verwendungsmöglichkeiten ab. Unter der Voraussetzung das dem Endbenutzer ein mobiles Endgerät und dessen Funktionalitäten zur Verfügung stehen. Die Nutzer von mHealth

können in drei Schichten eingeteilt werden. Zunächst wären da die Gruppe der Smartphone Nutzer die meist in Industriestaaten leben. Sie haben fast immer Internetzugang und keine Probleme mit Energieknappheit. Dann wären da die Gruppe der Nutzer die zwar in Industriestaaten leben aber kein Smartphone besitzen. Darunter fallen unter anderem ältere Personen. Zuletzt wären dann die Gruppe die kein Smartphone besitzen sondern lediglich ein einfaches Mobiltelefon das Textnachrichten und Telefongespräche ermöglicht. Diese Gruppe findet sich meist in Ländern welche keinen Industriestaat Status genießen [Smartphonehc, 2014].

1.1.5 Rechtliche Rahmenbedingungen

Das Datenschutzgesetz 2000 (DSG 2000) [RIS, 2014a] ist das Bundesgesetz über den Schutz von personenbezogenen Daten. Dieses Gesetz ist die Grundlage der elektronischen Datenverarbeitung in Österreich. In §9 DSG 2000 sind die Bedingungen für die Verwendung von sensiblen Daten (z.Bsp.: Medizinische Diagnostik) geregelt und wie diese definiert sind. Des weiteren regelt das Gesundheitstelematikgesetz 2012 (GTelG 2012) [RIS, 2014b] die Maßnahmen zur Datensicherheit bei der Verwendung elektronischer Gesundheitsdaten. Das GTelG definiert jedenfalls welche Datenkategorien als Gesundheitsdaten anzusehen sind [BMG, 2014c]. Das GTelG sieht nur vor, das Daten vor Missbrauch von Dritten technisch und wirtschaftlich wirksam verhindert wird. Die Methoden welche vor Missbrauch schützen sollen, finden sich in der Gesundheitstelematikverordnung (GTelV) [BMG, 2014d].

1.2 Ziele dieser Masterarbeit

Diese Masterarbeit soll drei Fragestellungen beantworten. Im Grunde handelt es sich bei dieser Arbeit um die Themen Hüftdysplasie, mobile Applikationen und Sonogramme (Ultraschall der Neugeborenenhüfte). Die Hüftsonographie wurde wegen folgender Faktoren gewählt. Bilder sind klein (Datenvolumen), Hüftdysplasie betrifft weltweit viele Menschen, einfache Behandlung durch Bandage und die Akzeptanz der Telemedizin (z.Bsp.: Teleradiologie).

1.2.1 Inzidenzen und Länder

Das erste Ziel der Arbeit stellt die Recherche nach Inzidenzen für Hüftdysplasie dar. Forschungsfrage: *Welche Länder/Gebiete weisen erhöhte Inzidenzen bei Hüftdysplasie auf?*

1.2.2 Mobile Applikation

Das zweite Ziel ist die Programmierung einer mobilen Applikation die als Tool für ärztliche Zweitmeinung dient. Dieser Prototyp soll die folgende Forschungsfrage beantworten: *Ist eine mobile Applikation in der Lage Sonogramme korrekt aufzunehmen?*

1.2.3 Validierung

Der dritte Teil beschäftigt sich mit der Validierung der entstandenen Daten. Forschungsfrage: *Können mit dem Smartphone aufgenommene Sonogramme für eine ärztliche Zweitmeinung herangezogen werden?*

Kapitel 2

Material und Methoden

2.1 Hüftgelenksfehlstellungen

Man unterscheidet bei Hüftgelenksfehlstellungen die Hüftgelenksdysplasie (englisch: *DDH = developmental dysplasia of the hip*) und die Hüftgelenksluxation (englisch: *CDH = congenital dislocation of the hip*). Hüftdysplasie ist der Oberbegriff für die Fehlstellung und Störung des Hüftgelenks. Diese Fehlstellung entsteht bei der Verknöcherung des Hüftgelenks (Neugeborene) und tritt durch eine Folge von Faktoren auf. Beengte intrauterine Platzverhältnisse mit hohen Scherkräften des Oberschenkelkopfes führen zu einer Schädigung des knorpelig ausgeprägten Pfannendachs. Die Fehlstellung der Hüfte tritt meist alleine auf, aber kann auch zusammen mit anderen Fehlbildungen auftreten. Ohne Behandlung kann es zu sehr schweren Schäden des Hüftgelenks kommen. Um Hüftdysplasie zu diagnostizieren wird vor allem die Sonografie verwendet. In Abbildung 2.1 werden die verschiedenen Hüftgelenksstellungen schematisch dargestellt.

Definiert wird diese Erkrankung als ungenügende Ausbildung des Hüftgelenkes mit Störung der Verknöcherung des Pfannenerkers [Hefti, 2006]. Der Pfannenerker ist der Teil der Hüftpfanne welcher als Überdachung des Hüftgelenks dient. Ist dieses *Dach* abgeflacht, also dysplastisch, dezentriert sich der Hüftkopf (Femurkopf) aus der natürlichen Position. Dadurch entsteht Druck auf die falschen Stellen im Becken, welche die Ossifikation (Verknöcherung) und das Wachstum des Gelenks beeinträchtigen. Bei Fortschreiten dieser Fehlentwicklung kann es passieren das der Femurkopf weiter dezentriert und letztlich aus der Gelenkspfanne austritt. Diesen Austritt nennt man Hüftgelenkssubluxation oder Hüftgelenksluxation. Bei Hüftgelenksluxationen dezentriert der Hüftkopf also aus der Hüftgelenkspfanne.

Einteilung von Hüftgelenksfehlstellungen nach Graf

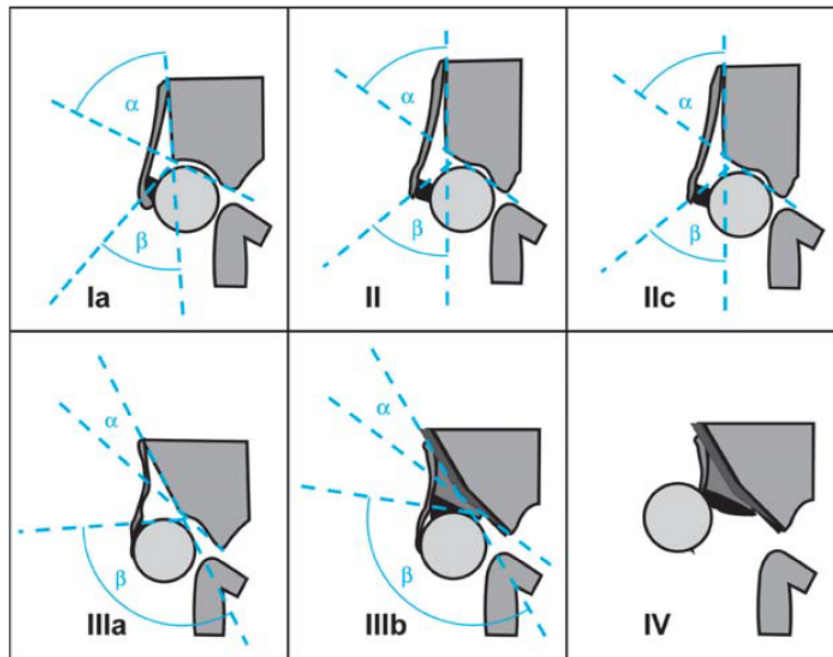


Abbildung 2.1: Einteilung der Ultraschallbefunde nach Graf. **I**: normal Hüfte; **IIa**: unreife Hüfte; **IIc**: instabile unreife Hüfte; **IIIa**: Hüftluxation, Knorpel nach kranial verlängert; **IIIb**: wie a, jedoch Knorpel verdichtet; **IV**: Hüftluxation. Abbildung aus [Hefti, 2006, Seite 182].

2.1.1 Epidemiologie der Hüftdysplasie

Die Anzahl der Diagnosen und Operationen hat sich in Österreich seit 1992 verringert. Damals wurde ein Hüftultraschallscreening eingeführt. Daher kommt es nur äußerst selten zu verspäteten Diagnosen die erst nach dem ersten, aber auch erst im zweiten Lebensjahr erkennbar werden. Länder mit Screening sind Österreich, Deutschland, Schweiz und Tschechien.

Die Häufigkeitsverteilung der Hüftdysplasie zeigt sehr starke Unterschiede je nach Ethnie und geografischer Lage. Die Verteilung ist aber auch von ihrer Definition und von den Methoden ihrer Diagnostik (Sonografie, Röntgen oder klinisch) abhängig.

Besonders hohe Inzidenzen wurden in slawischen Regionen (Tschechien und Kroatien) erfasst. Regionen in denen verstärkte Inzidenzen herrschen werden als Luxationsnester bezeichnet. Des weiteren finden sich sehr hohe Häufigkeiten der Hüftgelenkluxationen bei den Indianern und Inuit (Eskimos). Diese Naturvölker binden ihre Säuglinge mit gestreckten Hüft- und Kniegelenken auf ein Wickelbrett. Durch dieses Verfahren werden

Hüftluxationen stark begünstigt und führen zu einer Erkrankungswahrscheinlichkeit von 40%! (Hüftgelenksluxation) [Niethard, 1997]. Bei schwarzhäutigen Menschen findet sich Dysplasie äußerst selten. Dies könnte durch die Art und Weise wie die Kinder getragen werden erklärt werden: Entweder mit gespreizten Beinen oder zusammengewickelt in Beugung.

2.1.2 Krankheitsmechanismus und Risikofaktoren

Bei Neugeborenen die größer als der Durchschnitt sind und bei manchen Zwillings- bzw. Mehrlingslagen ist die Häufigkeit durch die intrauterine (innerhalb der Gebärmutter) Enge ein bisschen höher [Wülker, 2005].

Das Hüftgelenk kann als Kugelgelenk bezeichnet werden. Jedoch ähnelt es in seiner Form doch eher einem Nussgelenk. Die Pfanne des Gelenks gehört zu den Beckenknochen und der Gelenkkopf zu dem Oberschenkel. Das Hüftgelenk besteht beim Neugeborenen aus Knorpel im Gegensatz zum Kind, Jugendlichen und Erwachsenen. Der Knorpel verknöchert beim Säugling fortlaufend vom dritten bis neunten Lebensmonat.

Bleiben diese Deformationen (Hüftdysplasie) unbehandelt, kann es zu schweren bleibenden Schäden (Hinken, Störungen beim Gehen und Schmerzen) am Gelenk führen. Meist kommt es zu einer Hüftgelenksarthrose.

Hüftgelenksdysplasie tritt auch in Abhängigkeit des Geschlechts unterschiedlich oft auf. So haben Mädchen eine deutlich höhere Wahrscheinlichkeit [Hao et al., 2014] für das Auftreten der Fehlstellung. Somit ist die Hüftgelenksdysplasie das häufigste kinderorthopädische Krankheitsbild bei Neugeborenen [Orthopaedie, 2014].

2.1.3 Nicht-Operative Behandlung der Hüftdysplasie

Da das Hüftgelenk eines Neugeborenen zum Zeitpunkt nach der Geburt unreif ist, muss es stabilisiert werden. Durch die geringe Formsicherung (Knochen-Knorpel) und die gesteigerte Laxität (Schlaffheit) muss das Hüftgelenk postnatal (nach der Geburt) ausreifen. Dies ist nur möglich in der interuterinen Beugestellung (nach Salter auch *human position*) möglich. Das Beugen der Hüftgelenke hat den Vorteil, dass Säuglinge sich in der "human position" befinden und dadurch kann das Hüftgelenk in den ersten drei Lebensmonaten nachreifen und sich stabilisieren.

Ausreifungsbehandlung von Hüftdysplasie



(a) Human position



(b) Spreizhose



(c) Breites Wickeln

Abbildung 2.2: Ausreifungsbehandlung von Hüftdysplasie und -luxation bei Babys mit unterschiedlichen Hilfsmitteln. 2.2a zeigt eine Samburu Frau die ihr Kind in einem Tragetuch (human position) trägt [Tumblr, 2014]. In 2.2b erkennt man den Einsatz der sogenannten Tübinger Spreizhose [Wikipedia, 2014]. Abbildung 2.2c zeigt das breite Wickeln eines Neugeborenen aus [Huefte, 2014].

Wenn jedoch eine frühzeitige Streckung der Hüftgelenke vorgenommen wird können dadurch Luxationen begünstigt werden (Indianer und Inuit). Genauso wird die ständige Bauchlagerung von Kleinkindern als Luxationsfördernd angesehen. Somit ist das Wickeln der Hüfte in Beugestellung oder das Tragen der Babys im Tragetuch (Südostasien und Afrika) die beste Prävention. Eine Mehrwöchige Behandlung mit Bandagen (Pavlik Bandage, breit Wickeln oder Spreizhose) führt zum Ausreifen der Hüfte. Daher ist diese Erkrankung prädestiniert für Screenings. Durch eine einfache, schmerzlose und billige Untersuchung (Hüftsonographie) kann eine schmerzlose Behandlung (Bandage) eingeleitet werden und somit dem Patienten teure und schmerzhaft Operationen (OP nach Salter, Hüftprothesen usw.) vermieden werden. Der dadurch entstandene Gewinn an Lebensqualität ist immanent.

2.2 Hüftsonografie und Diagnose

Die Hüftsonographie wird vor der sechsten Lebenswoche des Säuglings durchgeführt und ist im Mutter-Kind-Pass verpflichtend vorgeschrieben. Sie basiert auf drei Standbeinen. Das wären Abtasttechnik, Anatomische Identifizierung und die Brauchbarkeitsprüfung.

2.2.1 Anatomische Identifizierung

Knochen-Knorpel-Grenze: Das koxale Ende des Femurs (Oberschenkelknochen) ist überwiegend hyalin (knorpelig) vorgebildet. Die Teilung der knöchernen von den hyalinen Anteilen wird als Knorpelknochengrenze (KKG) bezeichnet. Die hyalinen Teile sind der Hüftkopf, Trochanter major sowie der proximale Anteil des Schenkelhalses. Für die Identifizierung der anderen anatomischen Strukturen ist das Echo der KKG eine wichtige Beschaffenheitsstruktur. Die KKG kann dabei entweder bogenförmig, mit Schallpalisaden oder ohne medialen Anteil auftreten.

Hüftkopf: Der Hüftkopf entspricht bei einem Säugling einem Nussgelenk und ist eher Oval. Im knorpeligen Anteil finden sich Gefäße welche als Sinusoide bezeichnet werden. Diese werden im Sonogramm als kleine Wellen dargestellt. Diese befinden sich aber nur in der gefäßreicheren Zone (Zona zentralis). Die Oberfläche des Hüftkopfes ist jedoch größtenteils frei von Gefäßen (Zona anularis). Dieser Teil des Sonogramms ist im Grunde Echoarm.

Hüftkopfkern: Der Kern des Hüftkopfes ist nicht mit dem Zentrum des Hüftkopfes zu verwechseln. Der Hüftkopfkern darf nicht zur Beurteilung und Diagnose des Hüftkopf-Pfannen-Systems verwendet werden. Ist der Kern soweit verknöchert, dass es nicht mehr möglich ist den unteren Rand des Os Ilium zu erkennen (Schallschatten), darf das Sonogramm nicht zur Diagnose einer Hüftreifestörung verwendet werden.

Umschlagfalte: Die Umbiegestelle der Gelenkkapsel bezeichnet man in einem Sonogramm als Umschlagfalte. Diese Umbiegestelle liegt im Bereich in dem der Schenkelhals in den Trochanter major übergeht.

Gelenkkapsel: Der Kopf der Hüfte wird lateral (seitlich) von der Gelenkkapsel umfasst. Im Anschluss an die Umschlagfalte befindet sich nach kranial (zum Kopf hin) dem Hüftkopf entlang die Kapsel des Gelenks.

Labrum acetabulare: Von der Gelenkkapsel aus dem Hüftkopf folgend in Richtung kranial befindet sich das Labrum acetabulare. Es ist ein bogenförmiger aus Faserknorpel bestehender Vorsprung am Oberrand der Hüftpfanne (Acetabulum).

Definition des Umschlagpunkt: Der knöcherne Pfannenerker wird als Umschlagpunkt von Konkav auf Konvex definiert. Im Sonogramm wird dieser Punkt als Schallauslöschung wahrgenommen.

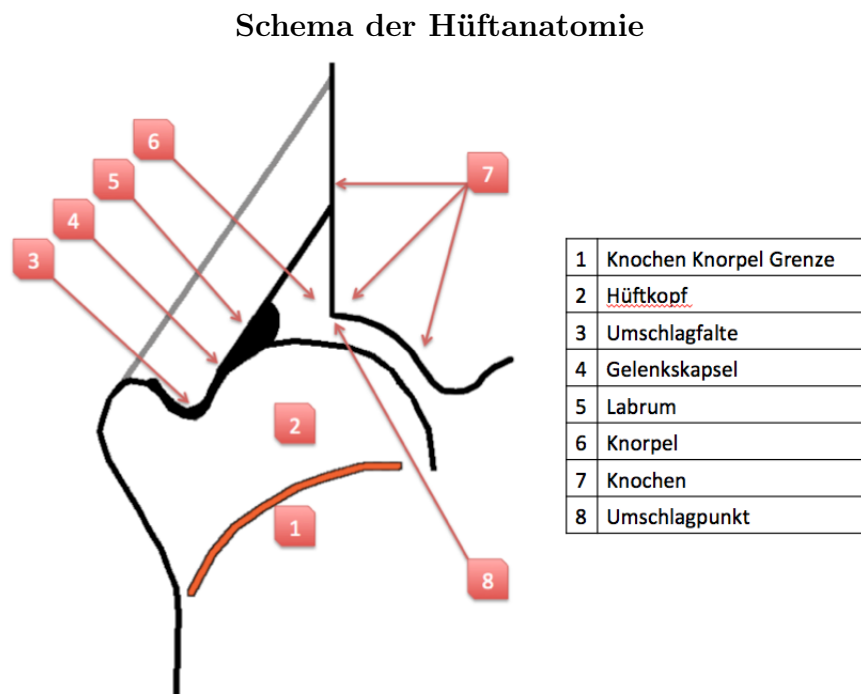


Abbildung 2.3: Diese Abbildung beschreibt die Anatomie der Säuglingshüfte aus dem Kurshandbuch für Hüftsonografie [Graf et al., 2013].

2.2.2 Klassifikation und Typeinteilung nach Graf

Die Einteilung (Typisierung) in der Hüftsonografie wird nach dem Grad der pathologischen Veränderungen im Hüftgelenk gerichtet. Es wird nicht das Ausmaß der Dislokation typisiert. Somit klassifiziert man in der Hüftsonografie die knöcherne und knorpelige Hüftpfanne in Korrelation zum Alter des Säuglings.

Typ I:

Hüftgelenke des Typ I werden als vollständig ausgereift angesehen. Der knöcherne Teil der Pfanne ist gut ausgebildet und der Bereich des Erkers (Pfannendach) ist eckig oder stumpf. Der knorpelige Teil des Pfannendachs überdeckt den Kopf der Hüfte.

Typ II:

Typ II der Klassifikation ist jedenfalls ein Gelenk mit zentriertem Hüftkopf. Die gesamte Überdachung (Knochen- und Knorpelteil) des Gelenks ist gut. Jedoch fällt das Verhältniss von Knorpel- und Knochenüberdachung zu Gunsten des knorpeligen Teil aus. Die Ausbildung des knöchernen Teils der Pfannenüberdachung kann von mangelhaft (Typ IIa-/IIb) bis ausreichend (Typ IIa+) reichen. Der knöcherne Erkerbereich erscheint eher rund, jedoch übergreift der knorpelige Teil der Hüfte das Gelenk ziemlich gut.

Typ III:

Beim einem Typ III handelt es sich um ein dezentriertes Gelenk. Der knöcherne Teil der Pfannenüberdachung ist in einem schlechten Zustand. Der Erker befindet sich in einem abgeflachten Zustand. Der knorpelige Teil der Überdachung wird in Richtung des Kopfes (kranial) verdrängt. Wegen des schlechten Zustands der Pfanne luxiert das Gelenk aus der Überdachung heraus. Auf diesem Weg der Luxation drängt der Hüftkopf den größten Teil der knorpeligen Pfanne nach kranial. Nur ein geringer Teil des knorpeligen Teils wird in die Richtung der ursprünglichen Hüfte geschoben.

Typ IV:

Der Typ IV entspricht genauso wie Typ III einem dezentrierten Gelenk. Der Unterschied besteht darin das der Hüftkopf die vollständige knorpelige Pfannenüberdachung in Richtung der Pfanne schiebt. Die Folge dieser Verschiebung bedeutet für den Hüftkopf das dieser von keinem Knorpel mehr überzogen ist.

2.2.3 Detaillierte Typisierung

Wenn jeder Typ den es bei Hüften gibt mit ihren Alpha- und Betawerten in einer Tabelle aufgelistet wird dann ist das Ergebnis ein Sonometer. In diesem Sonometer können alle Hüfttypen und deren Reifungsgrade mit ihren Winkeln in Relation zum Alter des Säuglings gesetzt werden.

Typeinteilung der Hüftdysplasie nach Graf

Typ	Knöcherne Formgebung	Knöcherne Erkerform	Knorpeliger Erker
I	gut	eckig/ stumpf	übergreifend
II	mangelhaft/ ausreichend	rund	übergreifend
III	schlecht	flach	nach oben verdrängt
IV	schlecht	flach	nach unten verdrängt
II mit Nachverknöcherung	mangelhaft/ ausreichend	eckig	übergreifend

Tabelle 2.1: Diese Tabelle zeigt zusammengefasst die unterschiedlichen Einteilungen der Typen in Verbindung zu der Ausprägung der Hüftpfanne. [Graf et al., 2013]

Durch das verwendete Winkelmesssystems können einige Vorteile genutzt werden. Die Vorteile liegen darin das es sich um Proportionen handelt. Die Proportionen des Pfannendachs sowie das Verhältnis des Knorpels zu dem Knorpeldach sind nicht von der Größe der Hüftgelenke (klein oder groß) abhängig. Die biomechanischen Verhältnisse bleiben auch unabhängig von der Größe dieselben [Graf et al., 2010].

Alphawert

Wenn man alle Alphawerte in einer Reihe auf einer Linie aufträgt, ergibt das einen guten Überblick über die verschiedenen Teilbereiche. Diese Linie wird durch ihre Eigenschaften in drei unterschiedliche Teilbereiche getrennt werden.

Abbildung 2.5 zeigt im oberen Teil die Auflistung der Alphawerte im Sonometer. Der **rechte Bereich** der Alphawerte betrifft Hüftgelenke mit Werten von 60° und größer. Diese sind als Typ I Gelenke Typisiert.

Im **mittlere Bereich** liegt Alpha zwischen 43° und 59° . Der mittlere Bereich umfasst die Typ II Gelenke. Bei Typ II Gelenken handelt es sich immer um zentrierte Gelenke. Dieser Typ kann noch weiter unterteilt werden.

Der Typ II Bereich wird in fünf Subtypen eingeteilt. Typ IIc, IIa, IIa+, IIa- und Typ IIb. **IIc** bedeutet ein α -Wert von 43° bis 49° . Dabei handelt es sich um Hüftgelenk das schon sehr nahe an der Dezentrierung liegt. Diese kommen in fast jedem Alter vor und

Grafische Darstellung der Vermessung der Neugeborenenhüfte

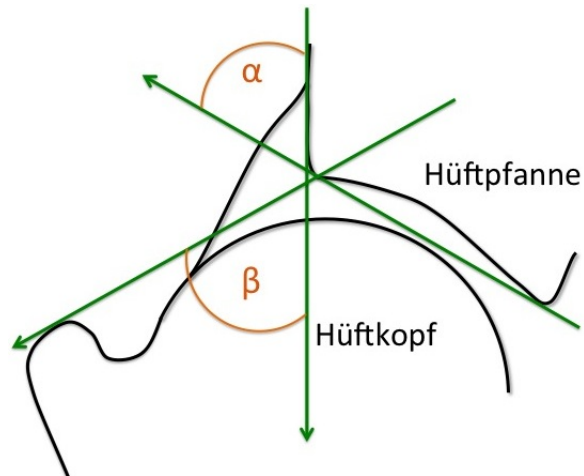


Abbildung 2.4: Von der Grundlinie aus, kann durch die Ausstelllinie und die Pfannendachlinie ein Maß für die Ausprägung der knöchernen Pfanne (α) und des knorpeligen Pfannendachs (β) gefunden werden. Aus [Graf et al., 2013].

müssen einer Stressuntersuchung unterzogen werden. Bei **IIa** liegt α zwischen 50° und 59° . Diese Gelenke sind nicht älter als 3 Monate. Der Typ IIa kann in zwei weitere Bereiche unterteilt werden. Das wären IIa+ und IIa-. Hüften mit **Typ IIa+** sind physiologisch unreif aber unter Einbezug des Alters gesund. Diesen Typ gibt es nur in den ersten 12 Lebenswochen. **Typ IIa-** bedeutet das das Hüftgelenk den Mindestreifungszustand (50°) nicht erreicht und eine Therapie eingeleitet werden sollte. Ein IIa- Gelenk hat eine Reifungsverzögerung (pathologisch) unter den ersten 12 Lebenswochen. **Typ IIb** ist auch im Alphabereich von 50° bis 59° . Das Hüftgelenk ist älter als 12 Wochen und wird als dysplastisches Gelenk bezeichnet.

Der **linke Bereich** beschreibt Gelenke des Typ III und Typ IV. Bei diesen Gelenken liegt Alpha unter 43° .

Betawert

Der Betawert unterliegt durch die sehr unterschiedlichen (individuellen) knorpeligen Pfannendachanteile einer größeren statistischen Schwankung als der vorher Besprochene Alphawert.

Bei ausgereiften Hüften kann das knöcherne Verhältnis gleich bleiben aber durch die Form des Knorpeligen Daches kann es zu Schwankungen kommen. Typ I Gelenke mit $\beta <$

Grafische Typeinteilung von α - und β -Winkel in einem Sonometer

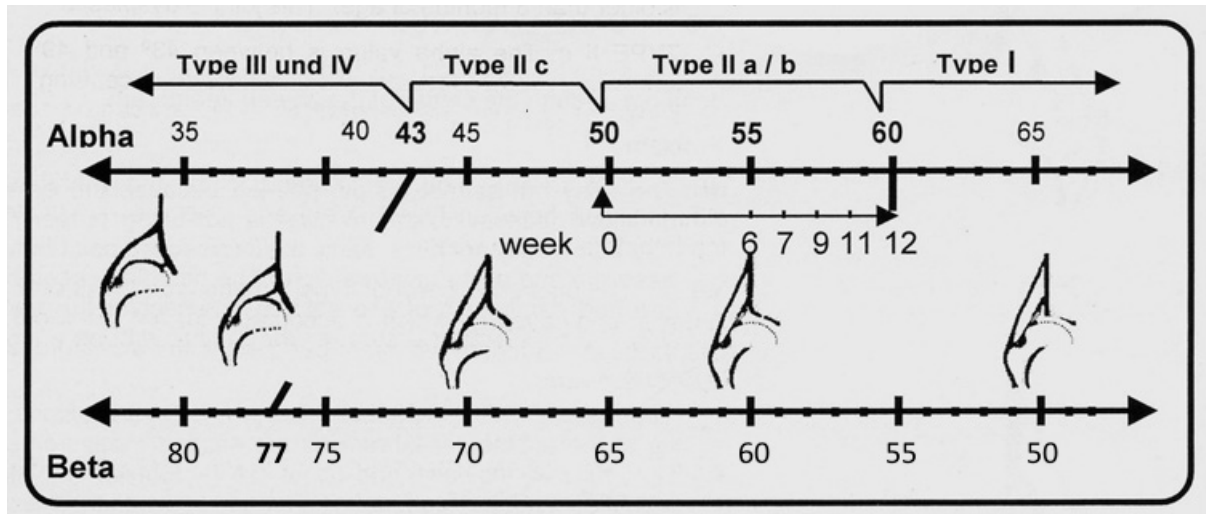


Abbildung 2.5: Detaillierte Typisierung der Säuglingshüfte mit Alpha und Beta Winkeln. Diese Art der grafischen Darstellung wird nach Graf Sonometer genannt. Abbildung aus [Toennis, 2014].

55° werden **Ia** genannt. Ist Beta > 55° wird das Gelenk **Ib** bezeichnet. Beide Gelenktypen sind gleich gut zu bewerten. Ist der Betawert im Bereich der IIc Gelenke > 77° wird dieses Gelenk als Typ D angesehen. **Typ D** ist der Anfang eines sich dezentrierenden Gelenks. Also kommt man zum Schluss das Typ IIc Gelenk mit Beta < 77° auch Typ IIc Gelenke bleiben. Jedoch ist ein Typ IIc mit Beta > 77° ein Typ D Gelenk.

2.3 Kunden

Kunden für xMEDx oder den App Prototyp sind Menschen in Ländern, mit erhöhtem Bedarf und geringem Angebot für Hüftsonographie. Die Kunden können mit der App und der Online Plattform einen erfahrenen Spezialisten ansprechen und ihm Befunde übermitteln. Der Prototyp, der in Kapitel 4 erklärt wird, zeigt wie ein Befund (Sonogramm) der Hüfte zu dem gewünschten Experten übertragen wird. Potenzielle Kunden wären Länder, welche hohe Geburtenzahlen sowie hohe Inzidenzen der Hüftdysplasie zeigen. In [Loder und Skopelja, 2011], einem Review Article vom ISRN Orthopedics (International Scholarly Research Network) wurden 422 Artikel untersucht, die einen Bezug zu DDH hatten. **Die Ergebnisse zeigten, das DDH meist links-seitig (64%) und einseitig auftritt (63,4%).** Die Inzidenzen pro 1000 Neugeborenen liegt in einem Bereich von 0,06 bei

Afrikanern in Afrika bis zu 76,1 bei den amerikanischen Ureinwohnern. Die Instabilität der Hüfte bei der Geburt liegt zwischen 0,4 bei Afrikanern und 61,7 bei polnischen Kaukasiern. Anzeichen für DDH sind Beckenendlage mit gestreckten Beinen, das Auftreten von DDH in der Familie und das Geschlecht (Weiblich).

Kapitel 3

Android, Apps und Smartphones

Im folgenden Kapitel wird auf die Technologie des App Prototyp und das Umfeld von Smartphones näher eingegangen. Es wird ein kurzer Überblick über Betriebssysteme allgemein und speziell für Smartphones gegeben. Es wurde der aktuelle Markt und die Verteilung der Smartphone Betriebssysteme untersucht. Der Unterschied zwischen drei Arten von mobilen Anwendungen wird erklärt und ein unkomplizierter Einblick in das Android Betriebssystem gewährt.

3.1 Betriebssysteme und Smartphones

Dieses Unterkapitel beschäftigt sich mit mobilen Betriebssystemen, Marktanteil und Anwendungen. Um mobile Anwendungen zu verstehen werde ich einen kleinen Überblick über die aktuelle Technik im Bereich der Smartphones geben. Es wird näher auf mobile Anwendungen (Apps) eingegangen und Unterschiede in diesem Bereich erklärt. Des weiteren werde ich auf Markt, App Stores und die jeweiligen Betriebssysteme eingehen.

3.1.1 Allgemeines über Betriebssysteme

Moderne Rechner bestehen aus einem oder mehreren Prozessoren (CPU), Arbeitsspeicher (RAM), Druckern Festplatten, Bildschirm, Netzwerkkarten und anderen unterschiedlichen Ein- und Ausgabegeräten. Das macht einen Rechner zu einem sehr komplexen System.

Um all diese Komponenten zu steuern und zu verwalten, sind komplexe Arbeitsschritte notwendig. Das Betriebssystem übernimmt diese Funktionen um dem Anwender eine besseres Verständnis für den Computer zu geben und um die Ressourcen zu verwalten.

Die unterste Ebene bei einem Betriebssystem bildet die Hardware. Unter anderem sind damit Chips, Platinen, Bildschirme und andere physische Komponenten gemeint. Auf der Hardware liegend befindet sich die Software. Der **Kernmodus** liegt direkt über der Hardware und bildet das grundsätzliche Betriebssystem. Im Kernmodus besteht kompletter Zugriff auf die gesamte Hardware und es können alle Befehle auf der Maschine ausgeführt werden. Im **Benutzermodus** können nur Teile der Maschinenbefehle verfügbar sein. Hier läuft die installierte Software des Anwenders.

Grundsätzlicher Aufbau eines Betriebssystems

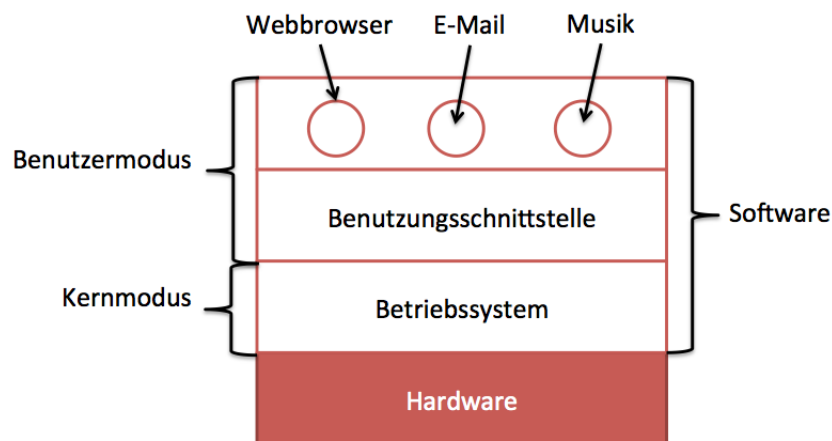


Abbildung 3.1: Einordnung von Betriebssystem nach [Tanenbaum, 2009, Seite 30]

Der Prozess ist das grundlegende in einem Betriebssystem. Im Grunde ist ein Prozess ein Programm in Ausführung. Jeder Prozess bekommt dafür seinen eigenen Adressraum zugeordnet. Dieser Adressraum wird benötigt damit ein Prozess lesen und schreiben kann. Die wichtigsten Aufrufe in einem System beschäftigen sich mit dem Erzeugen und Beenden eines Prozesses. Zusätzlich kann eine Prozess auch mehr Speicher anfordern oder unbenutzten wieder frei zu geben.

Deadlocks oder auch Verklemmungen entstehen, wenn zwei oder mehrere Prozesse zusammenarbeiten. Dabei kann es zu Situationen kommen, aus denen die Prozesse sich nicht

mehr lösen können. Nehmen wir an Prozess 1 wurde der Bildschirm zugeteilt und gleichzeitig wird der Drucker benötigt. Auf der anderen Seite ist aber der Drucker dem Prozess 2 zugeteilt und dieser benötigt noch den Bildschirm. Prozess 1 und 2 warten vergeblich das der andere die Ressource freigibt.

Die Speicherverwaltung dient der Verwaltung des Hauptspeichers in einem Computer. Einfache Betriebssysteme können nur ein ausführbares Programm zu einem Zeitpunkt im Speicher haben. Intelligenter Betriebssysteme können aber auch mehrere Programme gleichzeitig im Speicher haben.

Ein- und Ausgabegeräte sind Komponenten die jeder Computers besitzt. Das erleichtert uns die Kommunikation mit dem Computer. Der Computer liefert uns Ausgaben und der Anwender liefert ihm wiederum Eingaben. Das Betriebssystem verwaltet all diese Geräte.

Die Dateiverwaltung ist auch ein sehr essentieller Bestandteil von Betriebssystemen. Betriebssysteme haben die Aufgabe dem Anwender oder Programmierer ein verständliches abstraktes Bild eines Computers zu geben, ohne das man sich um Einzelheiten kümmern soll. Um Dateien zu lesen und zu schreiben verlassen sich Betriebssysteme auf das Konzept der Verzeichnisse. Das ist ein einfacher Weg um Dateien zu gruppieren und an einem Ort festzuhalten. Systemaufrufe können Verzeichnisse erzeugen und löschen sowie Dateien hinzufügen oder verschieben. Dabei entsteht eine Hierarchie oder Dateisystem. Jede Datei hat in dieser Hierarchie kann durch Pfadnamen benannt werden. Immer ausgehend von dem Wurzelverzeichnis bildet sich der individuelle Pfad.

3.1.2 Betriebssysteme für Smartphones

Smartphones unterscheiden sich nur sehr gering in ihrer Zusammensetzung der integrierten Hardware. Jedoch unterscheiden sich die mobilen Endgeräte im Bezug auf die verwendeten Betriebssysteme. Dadurch ergibt sich eine große Vielfalt in Bereichen wie Hersteller, Eigenschaften und Funktionen. Smartphones unterscheiden sich von klassischen Mobiltelefonen und PDAs. Sie sind nicht zum Telefonieren optimiert sondern sollen es ermöglichen eine Vielzahl von Anwendungen zu bedienen. Unter den Extras das ein Smartphone mit sich bringt befinden sich auch eine große Anzahl von Sensoren die sich je nach verwendetem

Gerät unterscheidet. Dazu zählen Bewegungs-, Magnetfeld-, Licht- und Näherungssensoren sowie GPS-Empfänger. Die aktuellen Betriebssysteme für Smartphones werden in diesem Abschnitt behandelt.

Aktueller Stand und Prognose der Marktanteile von mobilen Betriebssystemen

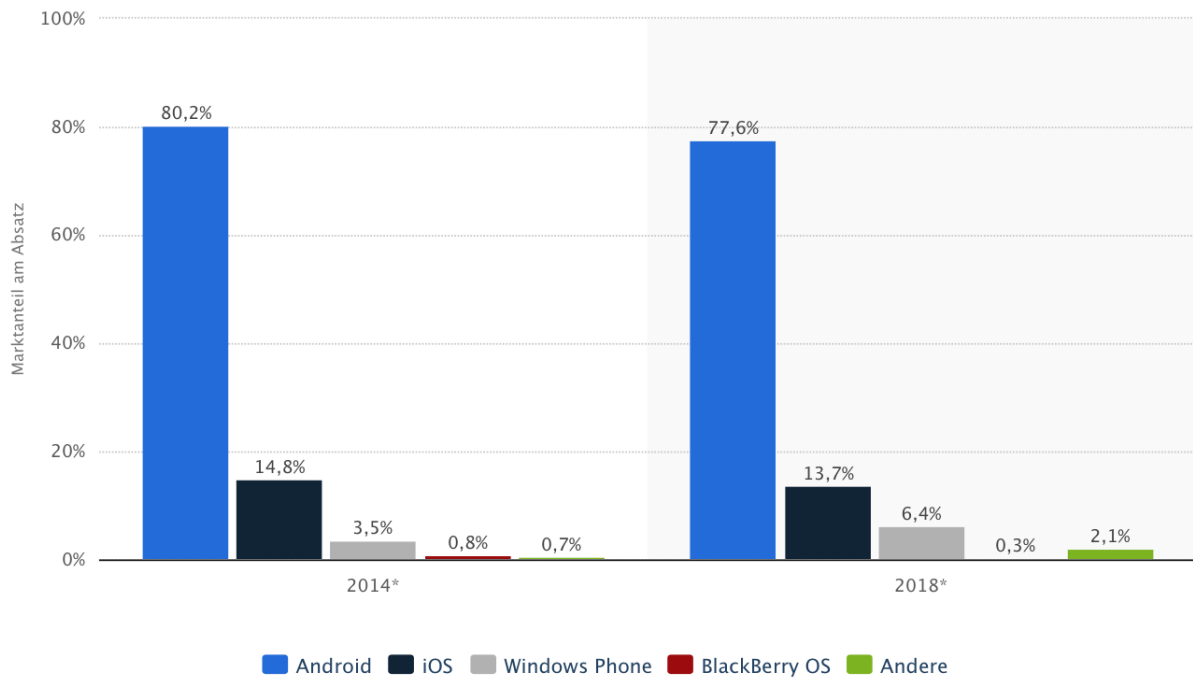


Abbildung 3.2: Prognose zu den Marktanteilen der Betriebssysteme am Absatz vom Smartphones weltweit in den Jahren 2014 und 2018 von [Statista, 2014a]

Android

Android ist ein Betriebssystem das von Google weiterentwickelt und finanziert wird. Es ist ein freies Betriebssystem, welches für mobile Endgeräte ständig weiterentwickelt und verbessert wird. Android ist hauptsächlich für Smartphones, Tablets und Netbooks als Betriebssystem gedacht. Unterschiedliche Hersteller von Hardware bieten Android als Betriebssystem an. Samsung, HTC, LG, Motorola und viele andere bedienen sich des Betriebssystems von Google. Apps für Android können über den zentralen Google App Store (Google Play) heruntergeladen werden. Android erfreut sich großer Beliebtheit und ist zum meist verwendeten Smartphone-Betriebssystem im Konsumentenbereich gewachsen.

Die Softwareentwicklung für Android Apps basiert unter anderem auf Standards wie Java. Die meisten integrierten Entwicklungsumgebungen (IDEs), wie Eclipse und NetBeans, unterstützen die Entwicklung von Android Applikationen. Durch die starke Nutzung von etablierten Standards können Android Apps sehr gut für die Entwicklung von Prototypen verwendet werden.

- Entwicklung: Google
- Programmiersprache: Java
- Lizenz: Apache 2.0, GPLv2 (Open Source)
- Marktanteil 2014: 80,2%

iOS

iOS ist das Betriebssystem, das Apple entwickelt hat, um seine mobilen Geräte auszustatten. iOS wird für Produkte wie iPad, iPhone und iPod touch verwendet. Da Apple die Hard- und Software für ihre Geräte herstellt sind ihre Produkte auch sehr gut miteinander kompatibel. Das ist ein großer Vorteil im Vergleich zu Android das von sehr vielen unterschiedlichen Herstellern genutzt wird.

- Entwickler: Apple Inc.
- Programmiersprache: Objective C/C++
- Lizenz: GPL, APSL, Apple-EULA u. a.
- Marktanteil 2014: 14,8%

Windows Phone

Windows Phone wurde von Microsoft entwickelt und ist ein Betriebssystem das hauptsächlich für Nokia Smartphones genutzt werden soll. Das Betriebssystem wird aber auch in kleinem Umfang an Hersteller wie Samsung, HTC oder Huawei vertrieben.

- Entwickler: Microsoft Cooperation
- Programmiersprache: C#/C++
- Lizenz: Microsoft-EULA
- Marktanteil 2014: 3,5%

Blackberry OS

Das Unternehmen Blackberry (vormals Research In Motion oder RIM) mit Sitz in Kanada entwickelt das Betriebssystem Blackberry OS ausschließlich für die eigenen Smartphones (Blackberry) [Sammer et al., 2014]. Dieses Betriebssystem ist sehr stark im Geschäftsbereich vertreten. Besonderes Augenmerk wird auf das Lesen und Schreiben von E-Mails sowie einen guten Sicherheitsstandard (Advanced Encryption Standard 256 Bit - Verschlüsselung des Datenverkehrs und des Gerätes selbst).

- Entwickler: Blackberry
- Programmiersprache: C++
- Lizenz: Proprietär
- Marktanteil 2014: 0,8%

3.1.3 Überblick über Smartphones

Smartphones sind Mobiltelefone mit zahlreichen Zusatzfunktion. Anwender können können im Internet surfen und Medien betrachten (Videos und Podcasts). Sie besitzen einen vollständigen E-Mail Client. Es können Termine erstellt werden und mit dem Server eines Firmennetzes synchronisiert werden. Zu den Schnittstellen zählen unter anderem WLAN und Bluetooth. Im Auslieferungszustand besitzt fast jedes Smartphone auch vorinstallierte Clients für soziale Netzwerke.

Ein Smartphone kann im Unterschied zu Mobiltelefonen seine Funktionen erweitern. Wie auch bei einem Computer können Anwendungen installiert werden. Diese Anwendungen oder Programme nennt man bei Smartphones Apps. Diese lassen sich bequem herunterladen und sind einfach zu installieren.

Im Unterschied zu normalen Computern wird ein Smartphone meist nur mit einer Hand bedient. Die Tastatur und der Bildschirm fallen dabei beim Smartphone natürlich kleiner aus als bei Computern. Die Rechenleistung der Smartphones ist im Vergleich zu normalen Computern ebenso verringert.

Als Beispiel betrachten wir das Samsung Galaxy S III (GT-I9300). Dieses Smartphone wurde zum Testen des App-Prototyp verwendet. Das Smartphone läuft mit dem Android Betriebssystem Jelly Bean Version 4.1.2. Es besitzt zwei Kameras, eine auf der Rückseite (8 Megapixel) und eine auf der Vorderseite (1.9 Megapixel) des Smartphones. Der

Prozessor (1.400 MHz) stammt von Samsung selbst und wird als Quad-Core betrieben. Des weiteren besitzt es einen Arbeitsspeicher von 1 GigaByte. Das Display ist mit einer Auflösung von 1280x720 Pixel sehr gut. Es fähig über NFC (Near Field Communication), WLAN sowie Bluetooth zu kommunizieren. Neben diesen technischen Daten hat das S3 auch eine Sprachsteuerung (S Voice),

Es ist beträchtlich wie sich die Computertechnik in den letzten Jahren in den Bereich der Mobiltelefonie gemischt hat. Die Grenzen verschwimmen und die Smartphones werden immer leistungsfähiger.

Nicht nur das Smartphones immer leistungsfähiger werden, sie werden auch immer öfter gekauft. Eine Studie der IDC (International Data Corporation) zeigt Verkaufszahlen von Smartphones, Tablets, Portablen PCs und Desktop PCs im Jahr 2012 und prognostizierte die Verkaufszahlen für 2017.

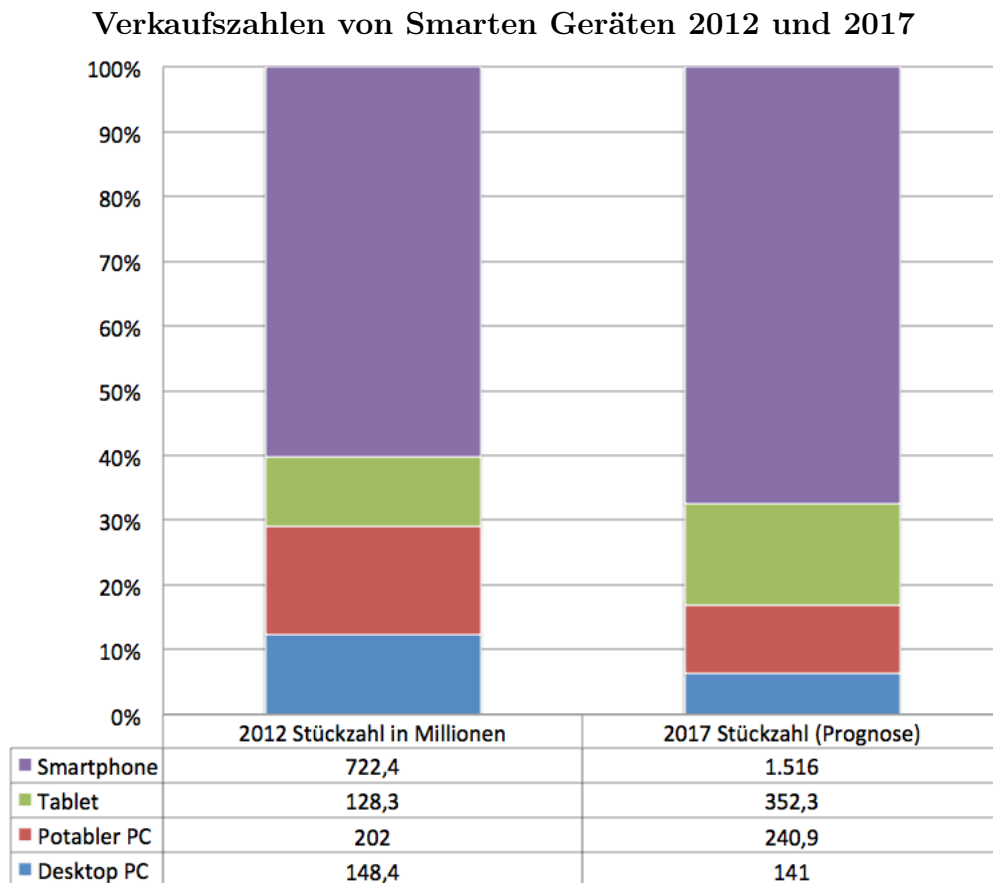


Abbildung 3.3: Prognose für den Verkauf von Smarten Geräten aus [IDC, 2013]. Aus der Grafik lässt sich erkennen das Smartphones und Tablets den Großteil der Verkauften Geräte ausmachen.

3.1.4 Apps für Smartphones

Als mobile App wird eine Anwendung bezeichnet die speziell für mobile Endgeräte entwickelt wurde und auf Smartphones oder Tablets ausgeführt wird.

Native Apps

Als Native App versteht man eine Applikation, die für ein bestimmtes Betriebssystem programmiert wurde. Da Native Apps für ein Betriebssystem optimiert sind, können damit komplexere und ressourcenintensive Berechnungen bewerkstelligt werden. Durch die gute Hardware die in aktuellen Smartphones verbaut ist lassen sich auch interaktive Apps wie Spiele deutlich besser realisieren. Auch die Nutzung der vom Gerät bereitgestellten Hardware (Kamera, Sensoren usw.) obliegt im allgemeinen auch nur dieser Kategorie von Apps. Diese Kategorie von Apps wird in der Programmiersprache der genutzten Plattform geschrieben und werden im allgemeinen als Download bereitgestellt. Native Apps können über einen der App Stores bereitgestellt und bewertet werden.

Web Apps

Bei Web Apps handelt es sich um Webseiten für mobile Endgeräte. Web Apps funktionieren auf allen Betriebssystemen/Smartphones und können dadurch mehr Nutzer ansprechen. Web Apps müssen auch keinen Zulassungsprozess durchlaufen. Damit können sich neue Versionen der Web App schnell verbreiten. Im Grunde ist es eine speziell in HTML5 und CSS3 programmierte App die auf allen gängigen Browsern genutzt werden kann. Die App funktioniert wie ein Browser der eine spezielle Website aufruft. Diese Art von Apps kann aber nicht auf die Hardware (z.Bsp.: Kamera) eines Smartphones zugreifen.

Hybrid Apps

Das Verpacken von Web Apps in Native Apps (Hybrid Apps) kann Vorteile beider Kategorien kombinieren. Das ermöglicht es Hybrid Apps z.Bsp.: im iPhone App Store und/oder im Google Play Store zu nutzen. Technisch gesehen wird dem Anwender vorgespielt das es sich um eine Native App handelt. In Wirklichkeit wird die Hybrid App nur als Webview genutzt. Das bedeutet das der User seine App öffnet und der Inhalt

aber einer Website entspricht. Dadurch wird auch eine Art der Plattformunabhängigkeit erreicht.

Wissenswertes über Apps und App Stores

Auch ältere mobile Endgeräte unterstützten schon Apps. Dazu zählten unter anderem Apps für Kalender, Taschenrechner oder auch Spiele. Diese waren für das jeweilige Betriebssystem geschrieben, vorinstalliert und im Vergleich zu heutigen Apps sehr einfach aufgebaut. Später konnte man durch WAP Anwendungen des eigenen Mobilfunkbetreibers installieren.

Die stabilere Internetverbindung und die moderneren Smartphones erhöhten den Bedarf an Anwendungen. Durch Apple und Google gelang dann der Durchbruch bei den Konsumenten. Durch das iPhone und Android Geräte konnten Nutzer ihre Apps direkt und von sich selbst aus beziehen. Die Entwickler können ihre Apps kostenlos und/oder kostenpflichtig zum Download bereitstellen.

Anzahl der angebotenen Apps in den unterschiedlichen App Stores

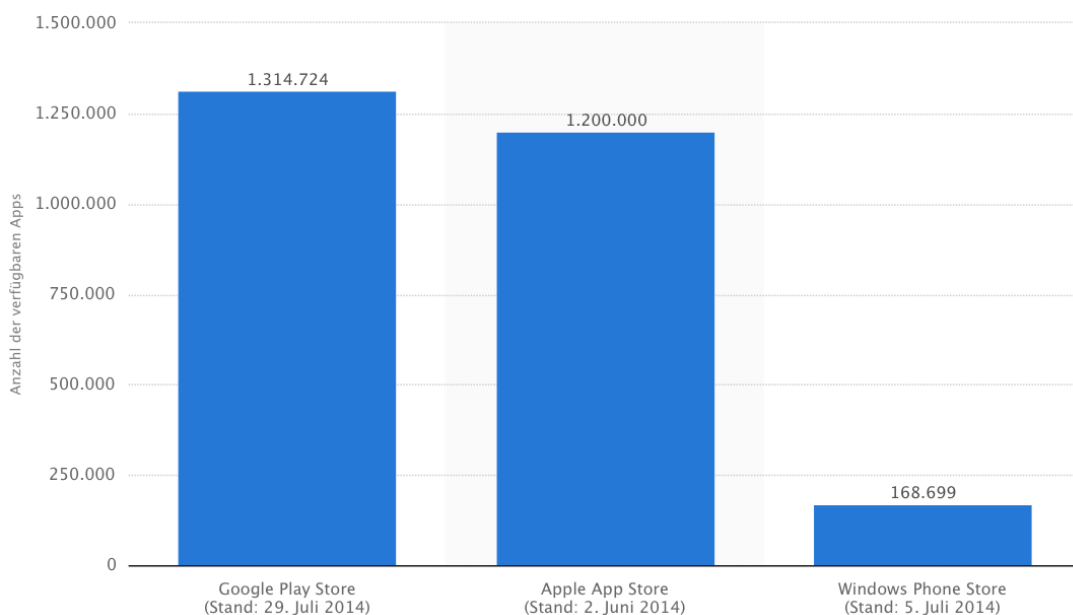


Abbildung 3.4: Anzahl der Apps welche in 3 unterschiedlichen App Stores zu Verfügung stehen. Diagramm aus [Statista, 2014b]

Die zwei größten App Stores sind die von Google und Apple. Der Google Play Store und der Apple App Store. Diese zwei Unternehmen teilen sich auch den Großteil des Marktes.

Nicht zu vergessen hat Microsoft auch seinen Windows Store in dem Apps für Windows Betriebssysteme heruntergeladen werden können.

3.2 Android

Wie die Prognosegrafik 3.2 zeigt ist der Anteil von Android Betriebssystemen am Smartphone Markt mit 80,2% deutlich der Spitzenreiter. Gefolgt von iOS (14,8%), Windows Phone (3,5%) und Blackberry OS (0,8%). Daher haben wir entschieden eine Android Applikation als Prototypen für xMEDx zu entwickeln.

Verteilung der aktuellen Android Versionen

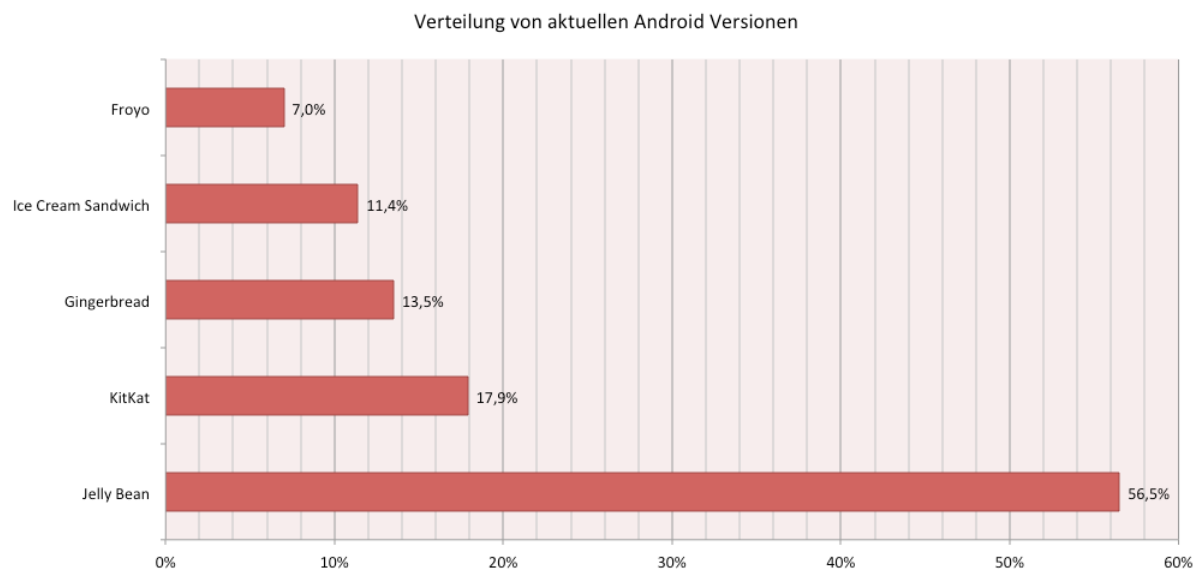


Abbildung 3.5: Verteilung des Google Betriebssystems Android nach Versionen. Die Daten wurden im Laufe einer sieben Tage andauernden Periode gesammelt welche am 07.07.2014 endete. [DeveloperAndroid, 2014d]

Android besitzt wie jedes andere Betriebssystem unterschiedliche Versionen. Je nach Erscheinungsdatum, Weiterentwicklung (Software und Hardware) gibt es am Markt unterschiedliche Versionen mit bestimmten Codenamen. Die folgende Grafik listet die aktuellen Versionen nach deren Verteilung. Die Versionen sind aufsteigend geordnet nach Marktanteil der Version. Beginnend bei Version 2.2 (Froyo), Version 2.3.3 - 2.3.7 (Gingerbread), Version 4.0.3 - 4.0.4 (Ice Cream Sandwich), die neueste Version 4.4 namens KitKat.Version und 4.1.x - 4.3 (Jelly Bean) welche den aktuell größten Anteil am Markt besitzt.

Ältere Versionen werden hier nicht berücksichtigt weil diese die aktuelle Google Play Store App nicht unterstützen. Die Daten aus Grafik 3.5 zeigen die Android Versionen welche aktiv den Google Play Store besuchen und nutzen.

Des weiteren findet man auf [DeveloperAndroid, 2014d] auch die Verteilung der Bildschirmgrößen sowie auch die Dichte der Displays.

3.2.1 Android Software Stack

Android ist eine umfangreiche Plattform die auf einem Linux Betriebssystem basiert. Auf diesem Basis Betriebssystem aufbauend stapeln sich mehrerer Softwaremodule (Software Stack) welche unterschiedliche Aufgaben bewerkstelligen. Die Verwaltung des Gerätes, des Speichers und der Prozesse erfolgt durch das Betriebssystem. Das bereitstellen von z.Bsp.: Telefonie, Video und Gestaltung des User Interface erfolgt durch Android Java Bibliotheken.

Google gibt Java-Entwicklern die Möglichkeit mit dem Android Framework zu Arbeiten und für dieses zu entwickeln. Das Android Framework wird als Standard Development Kit (SDK) oder Android SDK bezeichnet. Durch das SDK können viele der Bibliotheken verwendet werden die man auch von handelsüblichen Desktop-Rechnern kennt [Komatineni und MacLean, 2012].

Da Apps auf Android in Java programmiert werden wird auch eine Java Virtual Machine (JVM) benötigt. Die JVM ist verantwortlich um den Laufzeit Java Byte Code zu interpretieren. Die JVM verfügt um die notwendigen Optimierungen um den Java Code leistungsfähiger zu gestalten. Um die JVM auf mobilen Geräten zu nutzen, muss ein Augenmerk auf die beschränkten Ressourcen wie Speicher, Rechenzeit und Stromverbrauch gelegt werden. Für Android gibt es deswegen eine eigene virtuelle Maschine, die Dalvik Virtual Machine (DVM).

Den Kern der Android Plattform bildet ein **Linux Kernel**. Dieser Kernel verwaltet die grundsätzlichen Funktionen (Bildschirm, Tastatur, Kamera usw.) des mobilen Gerätes. Obwohl der Kern auf Linux aufgebaut ist, sind fast alle Applikationen auf einem Android Gerät in Java geschrieben.

Direkt darauf, also auf dem Kernel, sitzen bestimmte **Kernbibliotheken (Native)** auf. Diese Bibliotheken basieren auf der Programmiersprache C/C++. Da wären unter anderem OpenGL, WebKit, Secure Sockets Layer (SSL) und SQLite. Die **Dalvik VM** ist dazu optimiert mehrere Instanzen von virtuellen Maschinen zu verwalten. Wenn also

eine Android Applikation auf die Kernbibliotheken zugreift bekommt sie ihre eigene VM-Instanz.

Die **Hauptbibliotheken (SDK)** der Android Java API (Programmierschnittstelle) beinhaltet mehrere Teile. Unter anderem findet sich darin Bibliotheken für das Telefonieren, Ressourcen, Benutzeroberfläche, Content Providers und Packet Manager. Entwickler programmieren Applikationen für die Endnutzer auf dieser obersten Ebene der Android Java API. Als Beispiele wären da Apps wie Kontakte, Telefon sowie der Internetbrowser. In Grafik 3.6 als Anwendungsrahmen dargestellt.

In Abbildung 3.6 ist ein detaillierter Überblick des oben beschriebenen Android Software Stacks zu finden.

Detaillierter Aufbau des Android Software Stacks

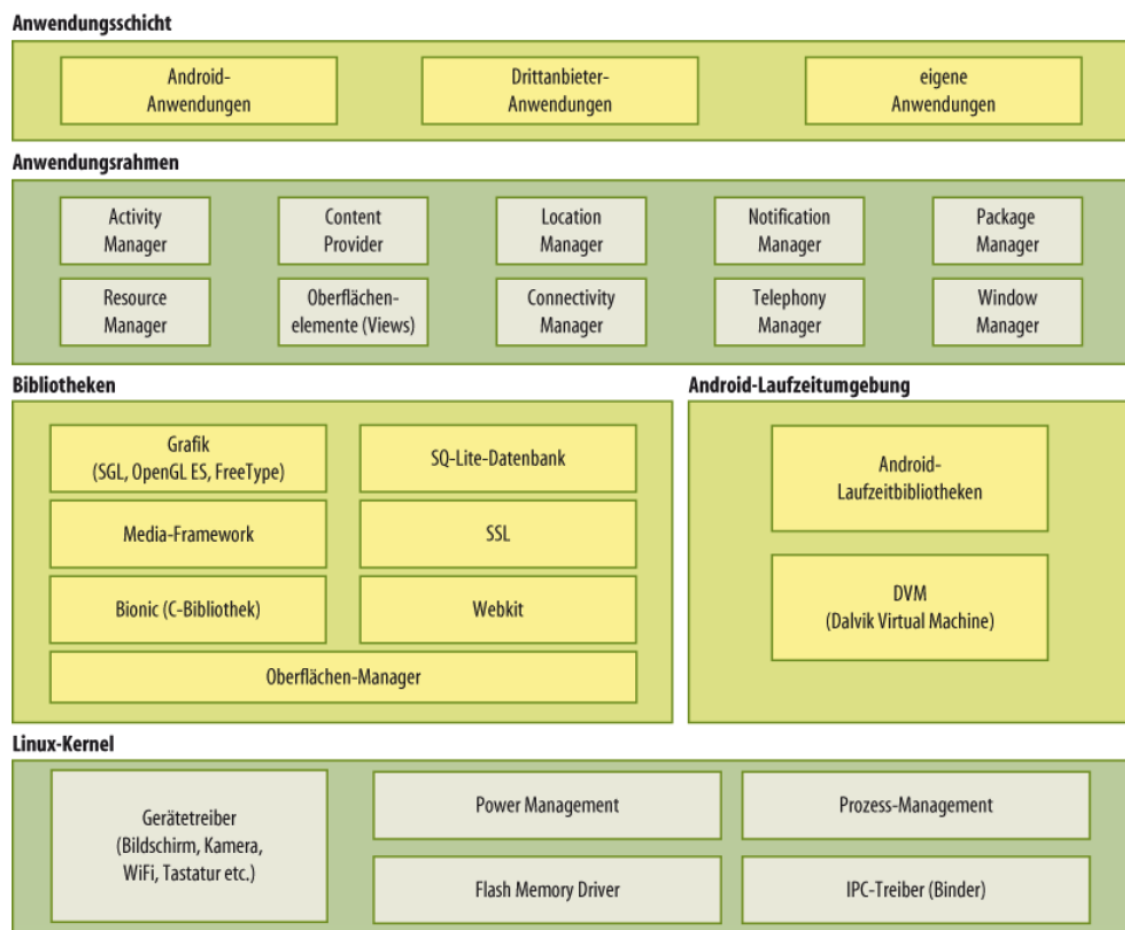


Abbildung 3.6: Detaillierter Überblick des Software Stacks von Android aus Sicht eines Entwicklers aus [Becker, 2011]

3.2.2 Grundlegende Komponenten

Das Android Framework nutzt mehrere Konzepte. Eines dieser Konzepte ist die Verwendung von Intents. Ein Intent ist ein Mechanismus der es uns erlaubt Komponenten in Android aufzurufen. Als Komponente wird ein Code bezeichnet, der einem bestimmten Lebenszyklus folgt. Es gibt vier Arten von App Komponenten, die auch Blöcke genannt werden.

Der folgende Code Abschnitt zeigt, wie Intents verwendet werden, um Komponenten in Android aufzurufen. Von der OneActivity ausgehend löst der der Intent eine andere Aktivität aus, die TwoActivity. Mit Intents ist es auch möglich, zusätzliche Daten zu übertragen. Das geschieht entweder mit dem Befehl `putExtra` oder `setData`.

Beispiel eines Intents:

```
Intent intent = new Intent(OneActivity.this, TwoActivity.class);
intent.putExtra("key", value);
OneActivity.this.startActivity(intent);
```

Android bietet Entwicklern auch den sehr einfachen Zugriff auf Ressourcen. Ressourcen sind unter anderem Strings, Bitmaps und XML Dateien welche für die Benutzeroberfläche verwendet werden (ähnlich HTML). Der folgende XML Code Abschnitt beschreibt ein Lineares Layout das einen TextView beinhaltet. Der Text wird aus der Ressourcen-Datei geholt und dargestellt.

Beispiel eines XML Layouts:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout>
<TextView android:id="@+id/textViewId"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="@string/hello"
/>
</LinearLayout>
```

Für die genutzten Elemente in XML Dateien generiert Android automatisch ID's. Wenn ein bestimmtes XML Element genutzt werden soll wird die ID dieses Elements verwendet um es

anzusprechen. Diese ID's befinden sich in der `R.java` Datei. In einer Aktivität z.Bsp.: könnte man diesen `TextView` in dem XML Beispiel folgendermaßen ansprechen: `TextView tv = (TextView) this.findViewById(R.id.textViewId);`

Ein besonderer Augenmerk sollte man auf das App Manifest richten. Besser gesagt auf die `AndroidManifest.xml` Datei. Diese Datei muss sich bei jeder Anwendung in dessen Hauptverzeichnis befinden. Das Manifest enthält alle wesentlichen Informationen der App. Nur durch diese Information, kann das System eine App ausführen. Name des Java Pakets, Komponenten (Aktivitäten und Dienste), die Berechtigungen und die Mindestversion der App.

3.2.3 Aktivitäten

Eine Aktivität repräsentiert ein Fenster in einer App. Als Beispiel können wir eine E-Mail App betrachten. Eine Aktivität dient dazu eine List von ihren neuen E-Mails anzuzeigen. Eine andere Aktivität lässt uns neue E-Mails verfassen und versenden. Jede dieser Aktivitäten arbeitet zusammen, um eine vollständige App zu bilden. Dabei sind Aktivitäten nicht von einander abhängig. Je nachdem was einer Aktivität erlaubt ist, können auch andere Apps oder Aktivitäten aufgerufen werden.

Eine App besteht meist aus mehreren Aktivitäten, welche lose miteinander gekoppelt sind. Üblicherweise ist eine Aktivität der App dazu bestimmt, die "main"-Aktivität zu sein. Das bedeutet das diese main beim Start der App als erstes Fenster angezeigt wird. Aus dieser main können dann weitere Aktivitäten gestartet werden. Jedes mal wenn eine neue Aktivität gestartet wird die vorhergehende gestoppt. Jedoch behält das System diese in einem Stapel (Back-Stack). Wenn der Nutzer die Aktivität beendet und den Zurück-Button drückt wird diese Aktivität vom Stapel genommen und zerstört (destroyed).

Eine gestoppte Aktivität ändert ihren Status in Stop. Aktivitäten können sich in diesem Status einiger Rückruf-Methoden (Callback Methods), welche im Lebenszyklus der Aktivität beschrieben sind, bedienen. Es gibt bei Aktivitäten unterschiedliche Status. Unter anderem gibt es Status für das Erstellen, Stoppen, Fortsetzen und Zerstören. Jede dieser Methoden kann spezielle Arbeit in der Aktivität leisten. Die Statusveränderungen sind Teil des Lebenszyklus einer Aktivität.

Um Aktivitäten zu erstellen, ist es notwendig, eine Unterklasse der Aktivität zu erstellen. In dieser Klasse müssen die vorher besprochenen Callback Methoden implementiert werden um den Status der Aktivität zu beschreiben. Die zwei wichtigsten Methoden sind dabei `onCreate()` und `onPause()`.

`onCreate()` ist eine Methode die implementiert werden muss. Das ist die Methode die aufgerufen wird, wenn eine Aktivität erstellt wird. In dieser Methode sollten alle verwendeten Komponenten initialisiert werden die für diese Aktivität notwendig sind. Darin befindet sich auch die Methode `setContentView()` mit welcher die Aktivität eine Benutzeroberfläche (Layout) bekommt.

Lebenszyklus einer Aktivität

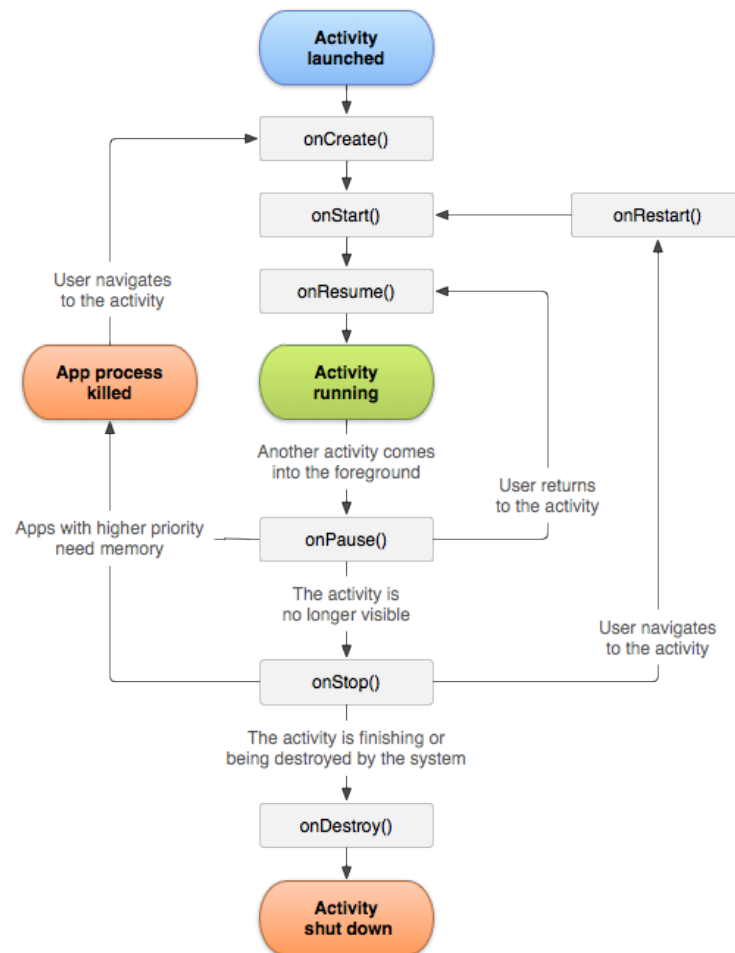


Abbildung 3.7: Ablauf und Lebenszyklus einer Android Aktivität. In diesem Ablauf lassen uns die unterschiedlichen Status und Callback-Methoden erkennen. Der Verlauf des Lebenszyklus wird durch die Navigation des Users bestimmt. [DeveloperAndroid, 2014a]

Benutzeroberflächen für Aktivitäten, werden als Views bezeichnet. Jeder View kontrolliert einen bestimmten Bereich in der Oberfläche der Aktivität und kann auf Eingaben reagieren. Ein Beispiel kann ein Button sein, welcher wenn er gedrückt wird eine Aktion auslöst. Android verfügt über vorgefertigte Views, die genutzt und verändert werden können. So genannte Widgets sind unter anderem Elemente wie Buttons, TextFields, CheckBoxen oder Bilder. Layouts sind auch Views (ViewGroup) die eine bestimmte Darstellung der kleineren Elemente ermöglicht. Relative Layout, Grid Layout und Linear Layout wären als Auswahl möglich. Es können aber durch Kombination von View Group und Views eigene Layout für eine Aktivität erstellt werden.

Im Manifest müssen alle Aktivitäten deklariert werden um aufgerufen werden zu können. Um Aktivitäten zu deklarieren müssen diese in ein neues `<activity>` Kindelement im `<application>` Element des Android Manifest hinzugefügt werden. Es können noch eine Vielzahl von anderen Attributen in diesen Elementen verwendet werden. Das einzig notwendige Attribut dabei ist `android:name`. In diesem Attribut muss der Name der Klasse der Aktivität stehen.

Ein `<activity>` Element können auch verschiedene Intent Filter nutzen, um zu definieren wie andere App Komponenten diese Aktivität aufrufen können. Das `<action>` Element spezifiziert das diese Aktivität der Haupteinstiegspunkt ist. Das Element `<category>` bedeutet das die Aktivität in den Launcher geschrieben wird und vom Nutzer gestartet werden kann. Falls kein Zugriff von anderen Apps auf diese Aktivität besteht, benötigt man auch keinen zusätzlichen Intent Filter mehr.

Beispiel eines Manifest:

```
<manifest ... >
<activity android:name=".ExampleActivity"
android:icon="@drawable/app_icon">
    <intent-filter>
        <action android:name="android.intent.action.MAIN" />
        <category android:name="android.intent.category.LAUNCHER" />
    </intent-filter>
</activity>
</manifest >
```

3.2.4 Sensoren

Fast alle Android Smartphones besitzen in ihrer Hardware integrierte Sensoren. Diese können unterschiedliche Gegebenheiten messen. Darunter fallen z.Bsp.: Bewegungen, Lagen und Lagenänderungen sowie andere Umwelteinflüsse. Android unterstützt Sensoren in den folgenden drei Kategorien.

Bewegungssensoren messen Beschleunigungs- und Rotationsbewegungen in drei Achsen. Unter diese Kategorie fallen Beschleunigungsmesser, Schwerkraft Sensoren und Rotations Sensoren. **Umweltsensoren** können unterschiedliche Umweltparameter messen. Umgebungstemperatur, Luftdruck, Beleuchtung und Luftfeuchtigkeit. Barometer, Lichtmesser und Thermometer fallen unter diese Kategorie. **Lagesensoren** messen die Lage/Orientierung des Android Gerätes. Darunter fallen Orientierungssensoren und Magnetometer.

Das Android Framework bietet sehr viele Sensoren an die genutzt werden können. Einige dieser Sensoren basieren auf Hardware andere auf Software. Auf Hardware basierende Sensoren sind Komponenten des Smartphones und sind in dieses integriert. Diese Physikalischen Sensoren (Hardware) messen direkte Einflüsse auf den Sensor (Beschleunigung und Magnetfeld). Software basierte Sensoren sind keine direkt integrierten Komponenten des Smartphones. Die Software Sensoren bedienen sich aber der Messungen von einem oder mehr Hardware Sensoren. Deshalb werden sie auch virtuelle oder synthetische Sensoren genannt. Als Beispiel eines Software Sensors kann der Schwerkraftsensor genannt werden. Die Sensoren in Android sind Teil des `android.hardware` Pakets und beinhaltet die spezifischen Klassen um Sensoren nutzen zu können. `SensorManager` wird genutzt um einen `SensorService` zu instantziieren.. Mit dem `Sensor Manager` können Sensoren im Allgemeinen verwaltet werden. Mit `Sensor` können spezifische Sensoren instantziiert werden und auf wichtige Methoden zugegriffen werden. `SensorEvent` wird verwendet um Events, also Informationen des Sensors, bereitzustellen. Der `SensorEventListener` beinhaltet zwei Methoden, welche sich mit der Veränderung der Sensorwerte oder der Genauigkeit des Sensors beschäftigt.

Koordinatensystem der Sensoren

Android benutzt ein Standard 3-Achsen Koordinatensystem um Daten zu liefern. Für die meisten Sensoren ist das Koordinatensystem relativ zur Standardposition (Portrait oder Landscape) definiert. Befindet sich das Smartphone in der Standardposition liegt die X-Achse Horizontal und zeigt nach rechts. Die Y-Achse liegt Vertikal und zeigt hinauf. Die Z-Achse geht direkt durch das Smartphone und tritt beim Bildschirm aus.

Dieses Koordinatensystem wird unter anderem von Beschleunigungs-, Schwerkrafts- und Magnetfeldsensoren verwendet. Wichtig dabei ist das dieses Koordinatensystem sich nicht verän-

Koordinatensystem der Android Sensoren

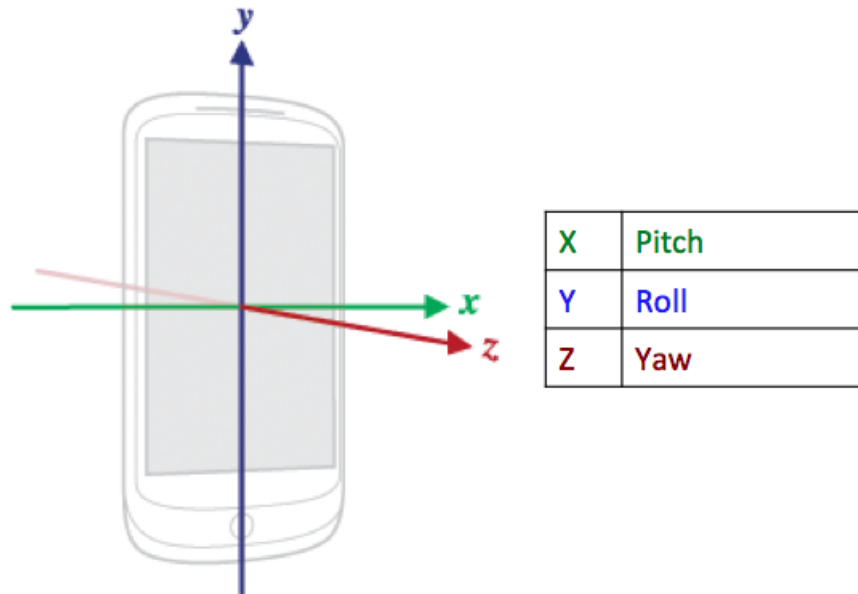


Abbildung 3.8: Grafische Darstellung des Koordinatensystems von Android Smartphones. In 3-Achsen können die Lage und Bewegungen des Smartphones gemessen werden.[DeveloperAndroid, 2014e]

dert, wenn sich das Android Gerät in der der Lage ändert. Das bedeutet das Bewegungen und Lagenänderungen keine Auswirkung auf das Koordinatensystem der Sensoren haben. Für Tablets ist zu beachten das die Standardposition meist dem Landscape Modus entspricht. Das Koordinatensystem ist immer an die Standardposition angepasst.

3.2.5 AsyncTask

Der AsyncTask ist eine weitere Wichtige Eigenschaft die für Android Geräte genutzt werden kann. Mit dem AsyncTask kann Code im Hintergrund ausgeführt werden. AsyncTasks sollten für kurze Operationen verwendet werden. Ein asynchroner Task ist definiert als eine Berechnung die im Hintergrund der Benutzeroberfläche läuft. Ein AsyncTask wird durch 3 generische Parameter (Params (Typ der an den Task übergebenen Paramter), Progress (Typ der im Hintergrund berechneten Einheiten) und Result (Typ des Resultates des Hintergrund-Tasks)) definiert. Die Methoden `onPreExecute`, `doInBackground`, `onProgressUpdate` und `onPostExecute` werden im AsyncTask verwendet.

Beispiel eines AsyncTask aus [DeveloperAndroid, 2014c]:

```
private class DownloadTask extends AsyncTask<URL, Integer, Long> {
```

```
protected Long doInBackground(URL... urls) {
    int count = urls.length;
    long totalSize = 0;
    for (int i = 0; i < count; i++) {
        totalSize += Downloader.downloadFile(urls[i]);
        publishProgress((int) ((i / (float) count) * 100));
        if (isCancelled()) break;
    }
    return totalSize;
}

protected void onProgressUpdate(Integer... progress) {
    setProgressPercent(progress[0]);
}

protected void onPostExecute(Long result) {
    showDialog("Downloaded " + result + " bytes");
}
}
```

Kapitel 4

Resultate

Diese Kapitel soll die Ergebnisse dieser Masterarbeit präsentieren. Der Android App Prototyp wird hier näher beschrieben und durch Screenshots und Code-Snippets näher erläutert. Des weiteren werden in diesem Kapitel auch die Bilder (Daten) die durch den Prototyp entstanden sind überprüft. Wie diese Kontrolle der Bilder erfolgte wird im Kapitel Datenvalidierung besprochen.

4.1 Hüftdysplasie: Inzidenzen und Länder

Die Länder mit den höchsten DDH Inzidenzen pro 1000 Neugeborenen sind: Kroatien (63,2), Tschechien (52,4), Deutschland (32,2), Spanien (26,8), Slowakei (26,5), Ungarn (26,1), Italien (Sizilianer) (25,5), Schottland (24,4) und Frankreich (22,6). Länder, die eine Inzidenz bis zu 20 DDH Fällen pro 1000 Neugeborene haben, sind Türkei, Israel, Griechenland (Kreter) die Niederlande und Polen. In Tabelle 4.1 sind die "wichtigeren" Länder (hohe Inzidenz) detaillierter zusammengefasst. Das Interval in der Tabelle gibt den Bereich an, in denen die Inzidenzen der zahlreichen Studien für die bestimmten Länder liegen. Wird Hüftdysplasie nicht behandelt kann es dazu kommen, dass Patienten vor dem 30. Lebensjahr ihre erste Hüftgelenksprothese operativ eingesetzt bekommen. Da auch Hüftprothesen eine gewisse Lebensdauer besitzen, kann es für den "jungen" Patienten bis zu drei weitere Revisionsoperationen geben. Im schlimmsten Fall wird die Nutzung eines Rollstuhls unumgänglich. Behandelt man die Hüfte jedoch früh genug mit einer Bandage, können hohe Kosten und der Verlust von Lebensqualität vermieden werden.

Berechnung der Inzidenz und der jährlich Neugeborenen mit DDH:

*Inzidenz = DDH Fälle in Studienpopulation / Studienpopulation * 1000.*

*Jährlich Neugeborene mit DDH = Population * Geburtenziffer * Inzidenz.*

Länder und berechnete Inzidenzen für Hüftdysplasie (DDH)

Länder	Inzidenz pro Tsd. Patienten ¹	Anzahl von DDH (Population)	Jährlich Neugeborene mit DDH ² 2014	Population 2014, in Tsd. ²	Population 2050, in Tsd. ²	Population Veränderung 2014-2050, % ²	CBR pro 1 000, 2010- 2015 ²	Studien population ¹	Fälle von DDH in Studienpopulation ¹	Interval ¹
Kroatien	63,2	269 491	2 425	4 266	3 412	-20,0	9	37 071	2 342	5,2 - 84,0
Tschechien	52,4	561 134	5 611	10 702	10 250	-4,2	10	100 545	5 272	18,5 - 113,2
Deutschland	32,2	2 655 413	21 243	82 544	69 007	-16,4	8	5 098	164	32,2
Spanien	26,8	1 256 511	12 565	46 958	45 357	-3,4	10	79 378	2 124	8,8 - 43,4
Slowakei	26,5	144 198	1 442	5 442	4 579	-15,9	10	16 718	443	12,6 - 44,8
Ungarn	26,1	258 311	2 325	9 910	8 263	-16,6	9	144 631	3 770	5,44 - 28,7
Italien (Sizilianer)	25,5	1 553 561	13 982	60 924	60 973	0,1	9	2 000	51	25,5
Schottland	24,4	1 548 306	18 580	63 485	72 697	14,5	12	229 001	5 585	3,1 - 50,0
Frankreich	22,6	1 463 137	17 558	64 632	72 931	12,8	12	3 048	69	12,2 - 36,2
Polen	18,7	713 305	7 133	38 135	45 003	18,0	10	17 108	320	11 - 61,7
Türkei	17,2	1 311 761	23 612	76 071	103 388	35,9	18	13 512	233	1,7 - 47
Niederlande	15,3	256 863	2 825	16 792	16 559	-1,4	11	2 092	32	15,3
Griechenland (Kreter)	10,6	117 614	1 059	11 110	10 132	-8,8	9	6 140	65	10,6
Israel	10,4	81 555	1 631	7 820	12 938	65,4	20	76 039	793	2,7 - 46
Norwegen	8,1	41 257	495	5 091	6 530	28,3	12	1508 249	12 222	1,7 - 22,4
Irland	7,6	484 907	5 819	63 485	72 697	14,5	12	10 081	77	7,6
Australien	7,6	257 771	3 351	33 871	48 544	43,3	13	278 698	2 121	6,6 - 19,4
Finnland	6,8	37 076	408	5 442	5 639	3,6	11	151 924	1 035	6,8
Dänemark	6,1	34 432	379	5 637	6 259	11,0	11	13 589	83	6,1
Kanada	3,4	122 420	1 347	35 496	43 649	23,0	11	172 522	595	1,2 - 9,9
Schweden	3,0	29 327	352	9 626	11 721	21,8	12	926 538	2 823	0,99 - 28,0
USA	3,0	957 160	13 400	323 316	409 286	26,6	14	196 592	582	0,2 - 15,3
England	2,6	166 370	1 996	63 485	72 697	14,5	12	697 544	1 828	0,66 - 19,3
Nord Irland	2,0	124 452	1 493	63 485	72 697	14,5	12	173 440	340	1,8 - 2,8

Abbildung 4.1: Zu ¹: [Loder und Skopelja, 2011]
Zu ²: [UnitedNations, 2014]

4.2 Prototyp der App

Dieses Unterkapitel wird den App Prototyp näher beschreiben, welcher entwickelt wurde. Mit Screenshots und und kleinen Code Snippets wird die Funktion besser erläutert. Die App wurde für den Landscape Modus optimiert. Die Notification- sowie die ActionBar wurden entfernt, um die gesamte Oberfläche des Smartphones zu nutzen. Zum Navigieren dienen Buttons, Intents und Aktivitäten. Auf Zurück-Buttons wurde bei der Programmierung der App verzichtet. Der Standard Back Button mit dem jedes Android Gerät hardwarespezifisch ausgerüstet ist, wird genutzt. Die Android App wurde in Android Studio [DeveloperAndroid, 2014b] entwickelt welches auf der IntelliJ IDEA [Jetbrains, 2014] basiert.

Der App Prototyp arbeitet mit Hilfe von bestimmten Sensoren des Smartphones. Durch die Sensoren ist es möglich die aktuelle Position des Smartphones zu bestimmen. Die Nutzung der Kamera und das Aufnehmen von Bildern mit dieser sind auch Funktionalitäten der App. Aufgenommene Bilder können bearbeitet werden um sensible Daten (Name, Geburtsdatum usw.) daraus zu entfernen. Des weiteren können die Bilder über eMail und SSL (Secure Socket Layer) an xMEDx versendet werden. Eine Loginfunktion sowie Uploadfunktion sind auch Teil der App. Diese wurden durch SOAP (Simple Object Access Protocol) realisiert.

4.2.1 Screenshots und Beschreibung der App

Wird die App gestartet wird der User zu dem Login geleitet. Der Username und das Passwort werden zu einem Server übertragen. Der Server überprüft die Logindaten und kann diese bestätigen oder nicht. Wird der User nicht bestätigt kann er die App nicht nutzen. Um die App nutzen zu können muss sich der User registrieren. Die Registrierung und dessen Verlauf werden zu einem späteren Zeitpunkt implementiert, da es sich noch um einen Prototyp handelt und das Backend mit Datenbankankbindung noch nicht komplett implementiert wurde.

Login des Prototyp

Die Bestätigung des Logins erfolgt auf Codeebene folgendermaßen. Der Username und das Passwort werden ausgelesen und an einen `AsyncTask` übergeben. Dieser Task verpackt die notwendigen Daten und sendet sie an den Server. Der Server überprüft die Userdaten und reagiert entweder mit einer Bestätigung oder einer Verweigerung (`loginStatus`). Das versenden der Daten (Request) zum Server erfolgt über `KSOAP2`. Der Username und das Passwort werden in sogenannte `PropertyInfo` geschrieben und zusammen in das `request` Objekt geschrieben.

Login-Screen im Prototyp

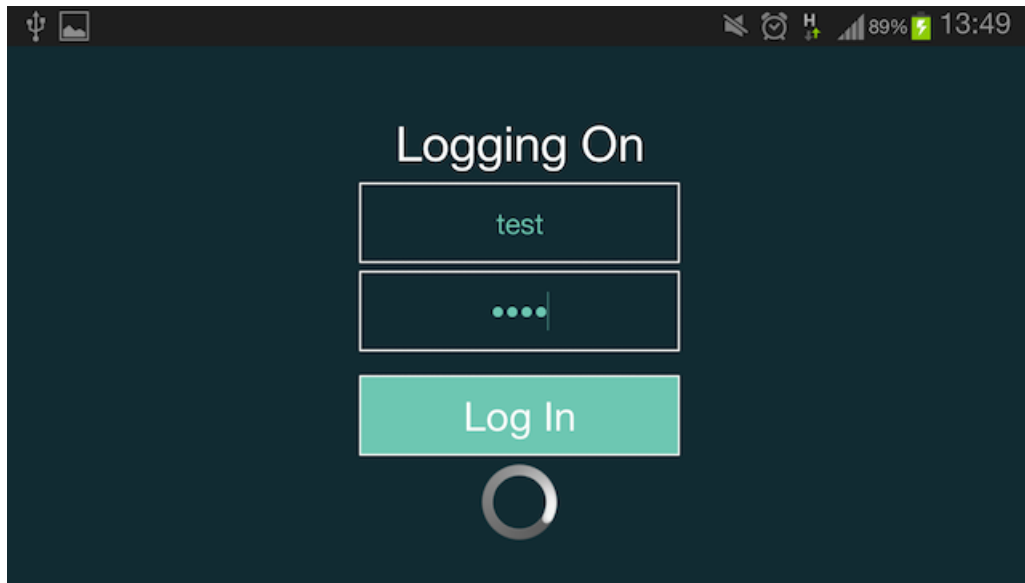


Abbildung 4.2: Der Login soll zunächst einem User den Zutritt zu der App gewähren. Username und Paswort wird erfordert.

Codesnippet des Logins aus [AndroidGuru, 2014]:

```
public static boolean invokeLoginWS(String userName, String
    passWord, String webMethName) {
    boolean loginStatus = false;

    SoapObject request = new SoapObject(NAMESPACE, webMethName);
    PropertyInfo unamePI = new PropertyInfo();
    PropertyInfo passPI = new PropertyInfo();

    unamePI.setName("username");
    unamePI.setValue(userName);
    unamePI.setType(String.class);
    request.addProperty(unamePI);

    passPI.setName("password");
    passPI.setValue(passWord);
    passPI.setType(String.class);
    request.addProperty(passPI);

    SoapSerializationEnvelope envelope = new
        SoapSerializationEnvelope(
            SoapEnvelope.VER11);

    envelope.setOutputSoapObject(request);

    //SSL HTTP Request
    //HttpTransportSE androidHttpTransport = new
        KeepAliveHttpsTransportSE("10.0.0.3",
            8443, "/WebService/services/WebService?wsdl", 1000);

    HttpTransportSE androidHttpTransport = new
        HttpTransportSE(URL);

    try {
        androidHttpTransport.call(SOAP_ACTION + webMethName,
            envelope);
        SoapPrimitive response = (SoapPrimitive)
            envelope.getResponse();
    }
```

```
loginStatus = Boolean.parseBoolean(response.toString());

} catch (Exception e) {
    CheckLoginActivity.errorred = true;
    e.printStackTrace();
}
return loginStatus;
}
```

In Android gibt es folgende Möglichkeiten Daten persistent abzulegen. SQLite Datenbanken, SharedPreferences und am Android Gerät selbst (interner und externer Speicher (SD-Karte)) [DeveloperAndroid, 2014d]. Der loginStatus wird im Prototyp mit SharedPreferences realisiert. Ist der Login erfolgreich wird der Status in die SharedPreferences geschrieben. Der Login entfällt daraufhin bis sich der User wieder aus der App ausloggt.

Start-Screen des Prototyp

Startbildschirm im Prototyp

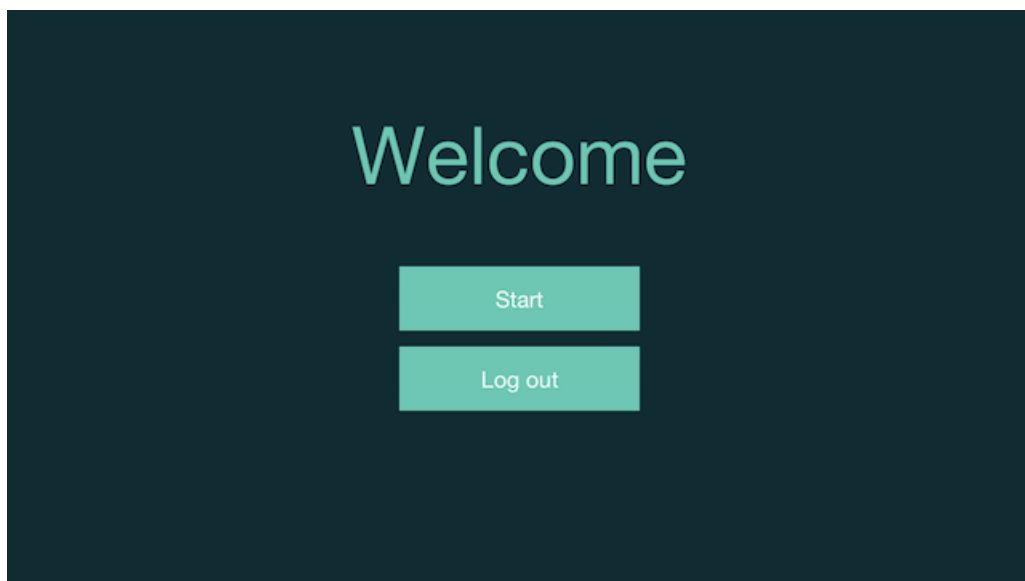


Abbildung 4.3: Der Startbildschirm erfolgt nach dem Log In und bietet die Möglichkeit sich auszuloggen oder zu bleiben und den Hauptprozess der App zu starten.

Der Startbildschirm der App erscheint nach dem Login. Er besteht im Prototyp aus zwei wichtigen Buttons. Zunächst der Startbutton der den Hauptprozess startet (Kamera) und den Logout Button. Die Navigation durch die App erfolgt dabei durch Intents.

Codesnippet des Startbildschirms:

```
@Override
    public void onClick(View v) {

        if (v == btn_start) {
            Intent camIntent = new Intent(MainActivity.this,
                CameraActivity.class);
            MainActivity.this.startActivity(camIntent);
        } else if (v == btn_logout) {
            SaveSharedPreference.clearLoggedInEmailAddress(mContext);
            Intent logIntent = new Intent(MainActivity.this,
                CheckLoginActivity.class);
            MainActivity.this.startActivity(logIntent);
            finish();
        }
    }
}
```

Kamera-Screen des Prototyp

Kameraansicht im Prototyp

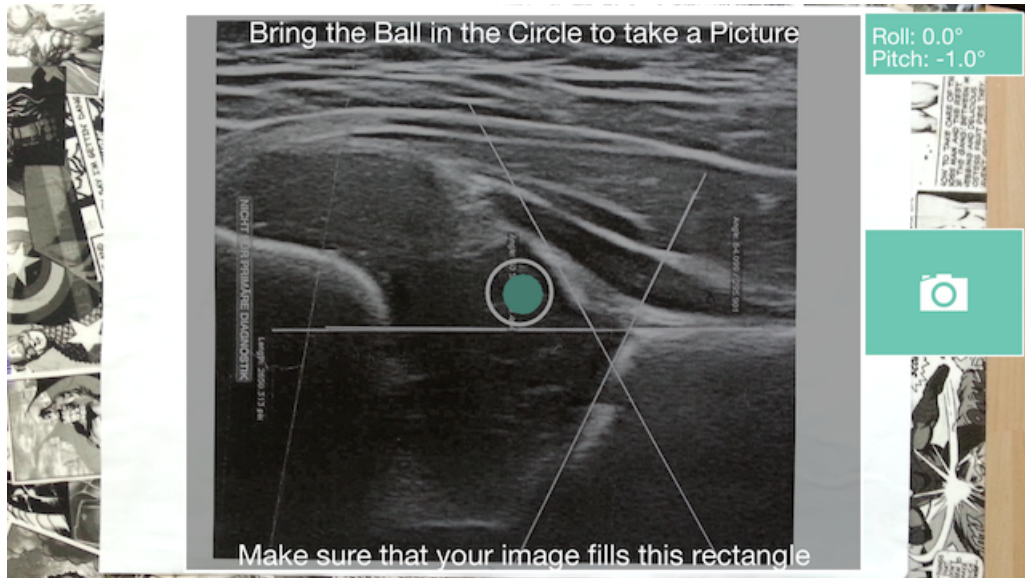


Abbildung 4.4: Die Kamerafunktion bietet durch den Einsatz von Sensoren (Lage) die Möglichkeit Bilder in unterschiedlichen Lagen des Smartphones aufzunehmen.

Der Kamera-Screen des Prototyp besteht im wesentlichen aus drei Hauptteilen. Teil eins besteht aus der Kamera des Smartphones welche eine CameraPreview zur Verfügung stellt. Der zweite Teil betrifft die Lage des Smartphones während der Aufnahme von Bildern. Die Lage des Smartphones wird in 4.4 rechts oben dargestellt. Sie besteht aus zwei Achsen (X und Y) des Koordinatensystems jedes Android Gerätes. In der Mitte der Abbildung erkennt man auch einen farbigen Punkt in einem Kreis. Der Punkt stellt den Beschleunigungssensor dar, welcher für die Berechnung des Pitch und des Roll notwendig ist. Dieser Punkt dient auch der Vereinfachung der Ausrichtung des Smartphones für den User. Teil drei besteht aus dem Layout, also der Benutzeroberfläche selbst. Alle drei Teile ergeben den Screenshot den man in 4.4 erkennen kann.

Codesnippet zur Ausgabe der Lage:

```
@Override
public void onSensorChanged(SensorEvent event) {
    if (event.accuracy == SensorManager.SENSOR_STATUS_UNRELIABLE) {
        return;
    }
}
```

```
float roll = Math.round(event.values[2]);
float pitch = Math.round(event.values[1]);

positionText.setText("Roll: " + Float.toString(roll) +
    "Degree" + "\n" + "Pitch: " + Float.toString(pitch) +
    "Degree");
positionValues = "_X_Roll_" +
    Float.toString(Math.round(event.values[2])) + "_Y_Pitch_" +
    Float.toString(Math.round(event.values[1]));

if (((roll >= -2.5) && (roll <= 2.5)) && ((pitch >= -2.5) &&
    (pitch <= 2.5))) {
    buttonTakePicture.setEnabled(true);
} else {
    buttonTakePicture.setEnabled(false);
}
}
```

Check-Screen des Prototyp

Überprüfung der Aufnahme des Prototyp

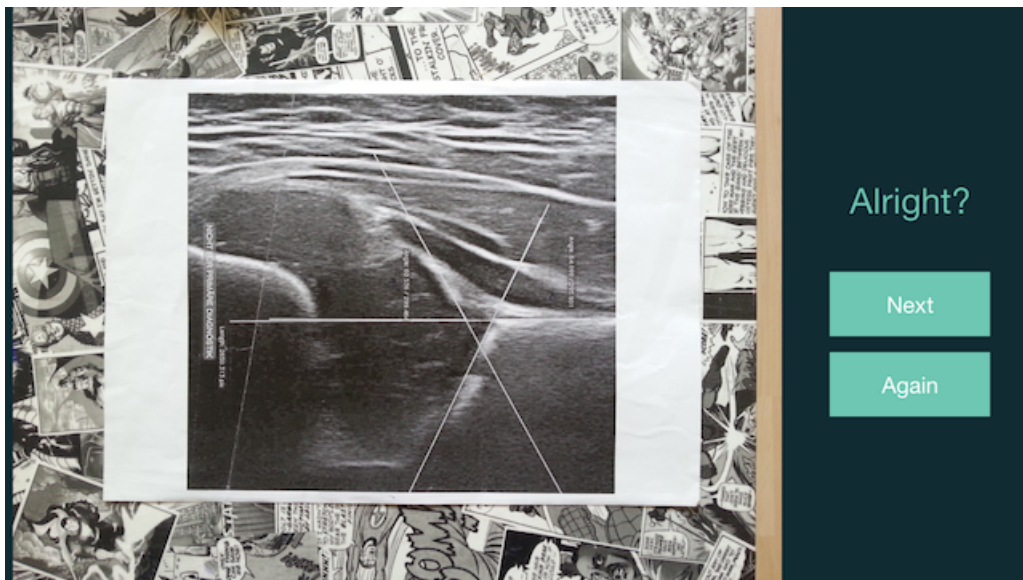


Abbildung 4.5: Die Check Aktivität lässt den User überprüfen ob die Aufnahme weiterverwendet werden kann, oder ob sie wiederholt werden muss.

Dieses Fenster dient der Kontrolle der Aufnahme. Falls die Aufnahme unbrauchbar ist, kann man über Knopfdruck die Kamerafunktion erneut starten. Das alte Bild wird verworfen und nicht gespeichert. Falls die Aufnahme korrekt ist, kann fortgefahren werden.

Draw-Screen des Prototyp

Pseudonymisierung im Prototyp

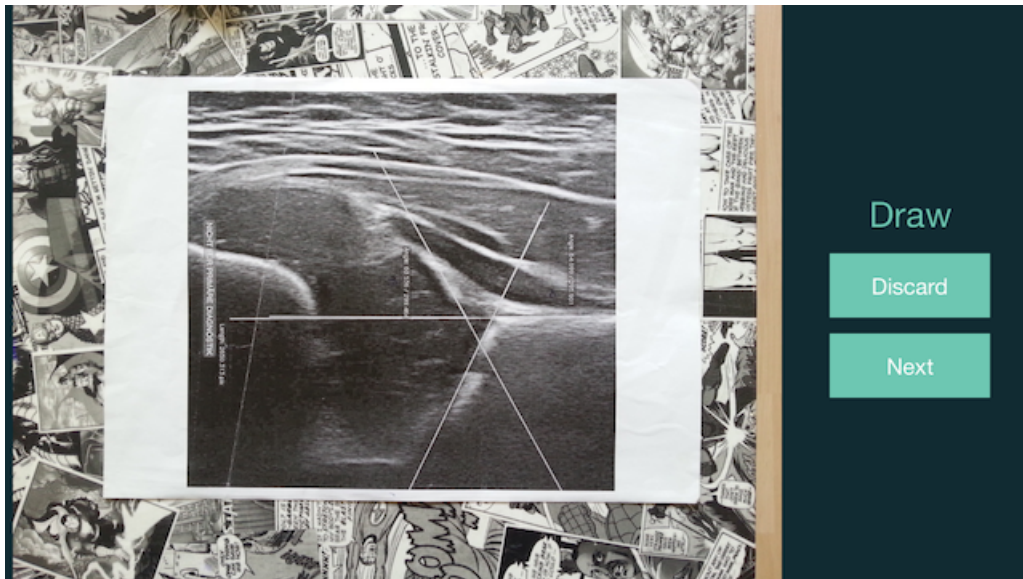


Abbildung 4.6: Das Menü im rechten Teil des Screenshots bietet: Discard: erneutes Laden der Aufnahme. Next: Weiteres Vorgehen und Bestätigung der Schwärzung.

Pseudonymisierung im Prototyp 2

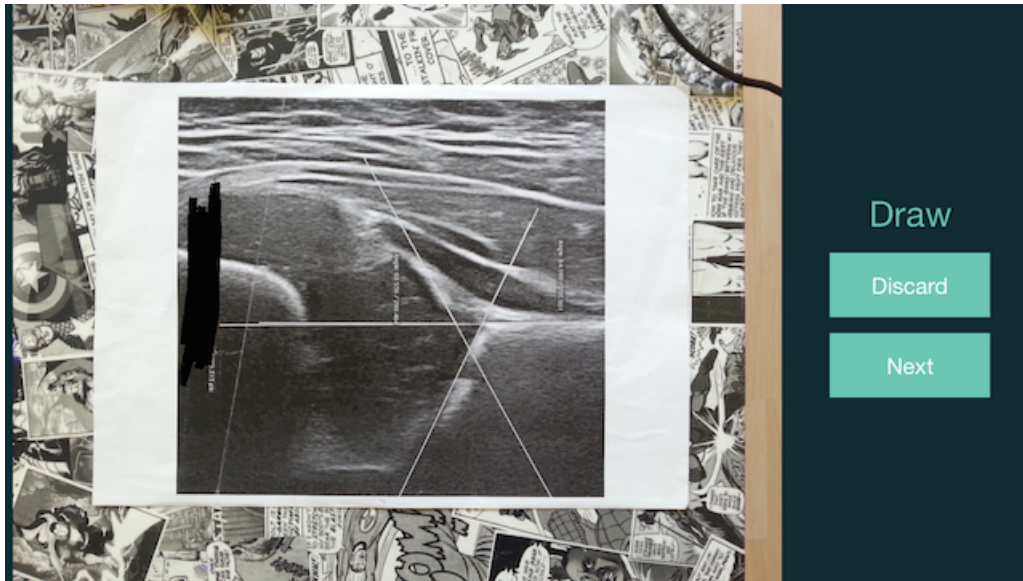


Abbildung 4.7: Die Draw Funktion lässt den User Bilder mit dem Finger minimal bearbeiten (schwärzen) um sensible Daten aus der Aufnahme zu entfernen.

In diesem Teil der App kann ein aufgenommenes Bild bearbeitet werden. Die Bearbeitung der Aufnahme erfolgt mit dem Finger. Streicht man mit dem Finger über die Aufnahme, können bestimmte Teile des Bildes geschwärzt werden. Das Paket `android.graphics` beinhaltet einige Werkzeuge die in dieser Aktivität verwendet werden. Die Canvas Klasse beinhaltet die "Draw" Befehle. Für Canvas werden 4 Komponenten benötigt. Eine Bitmap welche die Pixel enthält. Ein Canvas um in die Bitmap zu zeichnen. Eine Zeichenart, z.Bsp.: in 4.7 wäre das `drawline` und zum Abschluss noch eine Farbe (`paint`).

Codesnippet zum Bearbeiten der Aufnahme:

```
public boolean onTouch(View v, MotionEvent event) {
    int action = event.getAction();
    switch (action) {
        case MotionEvent.ACTION_DOWN:
            downx = event.getX() / scale_ratio +
                canvas.getClipBounds().left;
            downy = event.getY() / scale_ratio +
                canvas.getClipBounds().top;
            choosenImageView.invalidate();
            break;
        case MotionEvent.ACTION_MOVE:
```

```
        upx = event.getX() / scale_ratio +
            canvas.getClipBounds().left;
        upy = event.getY() / scale_ratio +
            canvas.getClipBounds().top;
        canvas.drawLine(downx, downy, upx, upy, paint);
        choosenImageView.invalidate();
        downx = upx;
        downy = upy;
        break;
    case MotionEvent.ACTION_UP:
        upx = event.getX() / scale_ratio +
            canvas.getClipBounds().left;
        upy = event.getY() / scale_ratio +
            canvas.getClipBounds().top;
        canvas.drawLine(downx, downy, upx, upy, paint);
        choosenImageView.invalidate();
        break;
    case MotionEvent.ACTION_CANCEL:
        break;
    default:
        break;
}
return true;
}
```

Expert-Screen des Prototyp

Experten im Prototyp

Choose your Expert

☒ Prof.h.c.Prof. Reinhard Graf
Inventor of Infant Hip Sonography

Payment

☒ Credit Card
*****3461

Experts

Next

Abbildung 4.8: Checkboxes zur Auswahl eines bestimmten Experten, sowie eine Bezahl-information.

Die Expertenauswahl ermöglicht es aus einer Liste von Experten zu Wählen. Je nach Themen-gebiet soll aus einer Liste von Experten gewählt werden können. Im weiteren soll es möglich sein auch eine Bezahlungsfunktion einzubauen und dadurch die Zweitmeinung zu bezahlen. Diese Informationen sollen während der Registrierung aufgenommen und zum Backend übertragen werden.

Mail-Screen des Prototyp

Versenden im Prototyp

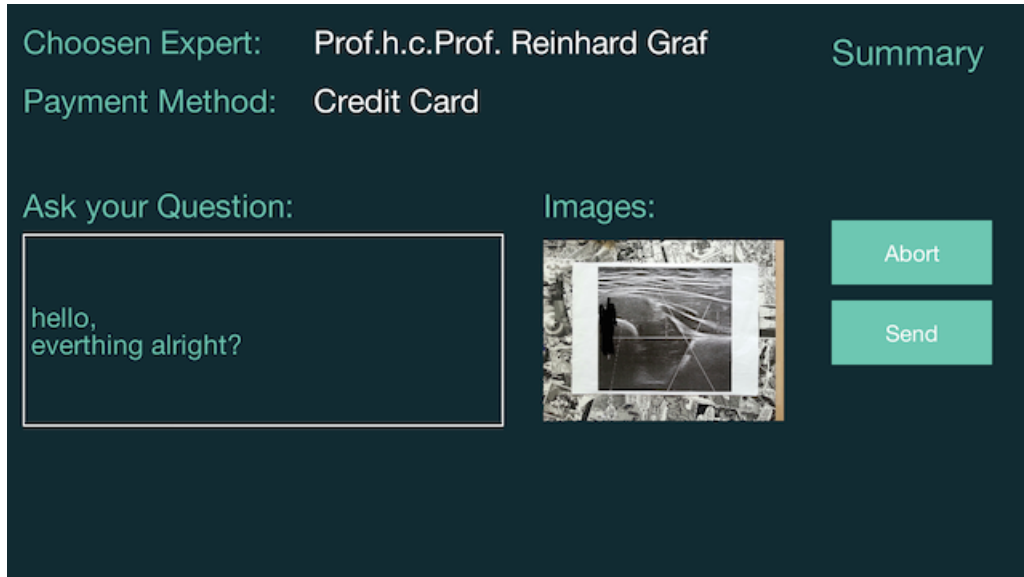


Abbildung 4.9: Die Bilder (Sonogramme) werden per eMail übermittelt, die im Hintergrund versendet wird. Der User hat hier alle seine eingegebenen Daten noch einmal zusammengefasst und kann diese kontrollieren.

Codesnippet zur Bearbeiten der Aufnahme:

```
private class SendAsyncTask extends AsyncTask<Void, Void, Void> {
    protected Void doInBackground(Void... params) {

        final Mail m = new Mail("username", "passwort");

        String[] toArr = {"recipient1", "recipient2"};
        m.setTo(toArr);
        m.setSubject("Request for:
            "+subjectTV.getText().toString());
        m.setBody("Question: "+messageET.getText().toString());

        try {
            String a = selectedImageUri.getPath();
            m.addAttachment(a);
            m.send();
        } catch (Exception e) {
```

```
        Toast.makeText(MailApp.this, "There was a problem  
        sending the email.", Toast.LENGTH_LONG).show();  
        Log.e("MailApp", "Could not send email", e);  
    }  
    return null;  
}  
  
protected void onProgressUpdate(Void... progress) {  
}  
protected void onPostExecute(Void result) {  
}  
}
```

Die letzte Stufe der App ist das Versenden der Aufnahme und eine abschließende Begutachtung. Die persönliche Fragestellung die der User dem Experten stellt, soll ebenfalls eingetragen werden. Die Zusammenfassung zeigt das der gewählte Experte, die Bezahlinformation, die Aufnahme und die Fragestellung noch einmal zusammengefasst werden. Zum versenden wird der Send-Button gedrückt. Die im Hintergrund erstellte eMail nutzt das `javax.mail` Paket. Mit Choose kann ein andere Aufnahme gewählt werden. Mit dem Home-Button wird wieder zum Startbildschirm navigiert.

4.3 Datenvalidierung

Um zu bestätigen, dass die Aufnahmen in Verbindung mit den Android Sensoren verwertbar sind, wurde eine Validierung der Daten durchgeführt. Es wurden 17 Aufnahmen ein und desselben Sonogramms gemacht. Das Smartphone hatte dabei 17 unterschiedliche Lagen, die durch unterschiedliche Neigung in X und Y Achse definiert sind. Testlagen: $\pm 30^\circ$, 20° , 10° , 5° und 0° für Pitch und Roll. Die Vermessung der Aufnahmen erfolgte manuell am Computer durch das Programm ImageJ [ImageJ, 2014].

Die aufgenommenen Test-Bilder wurden wie oben beschrieben vermessen. Die Neigungen bei der Aufnahme des Sonogramms, sind in den Dateinamen der aufgenommenen Test-Bilder hinterlegt. Die folgenden Tabellen zeigen die Ergebnisse, also die Winkel, im Vergleich. Tabelle 4.1 zeigt die realen Winkel des untersuchten Test-Sonogramms. Die realen Winkel betragen für $\alpha = 64,10^\circ$ und $\beta = 60,54^\circ$. Diese beiden Winkel werden für die Diagnose einer Hüftgelenksfehlstellung herangezogen.

4.3.1 Das Test-Sonogramm

Das Sonogramm in der folgenden Abbildung, wurde für die Auswertung und Vergleich der Winkel genutzt. Das Test-Sonogramm stellt die Basis für die Vermessung der Winkel dar. Es wurde von Dr. Christof Pabinger für die Studie zur Verfügung bereitgestellt.

Das Test-Sonogramm

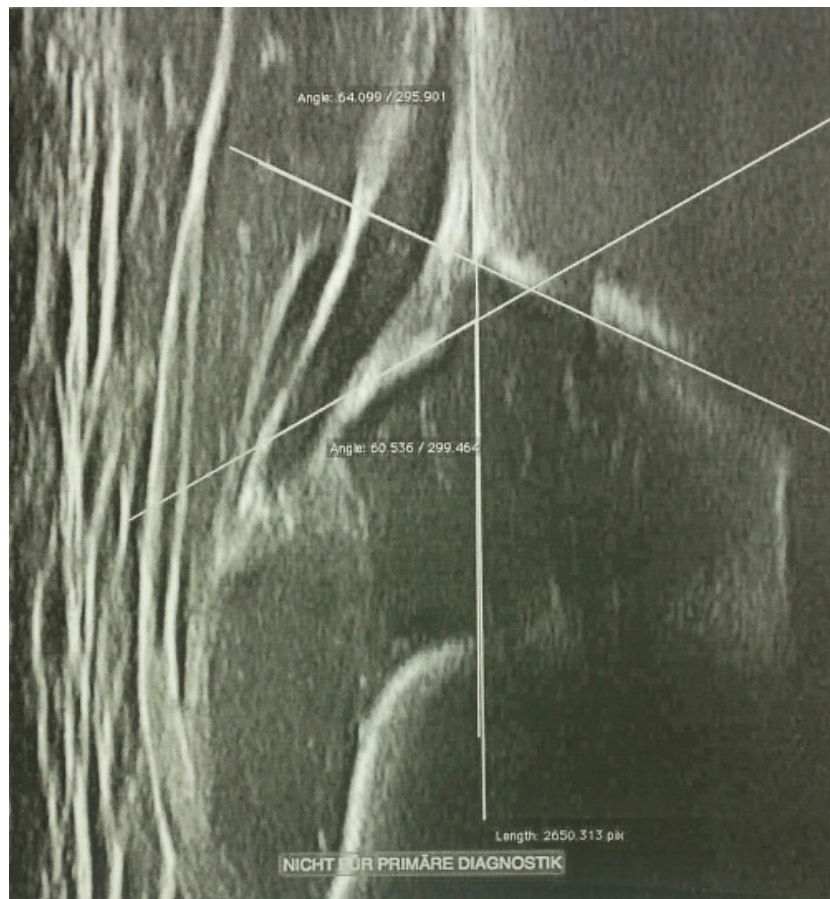


Abbildung 4.10: Das für die Validierung der Bilder verwendete Test-Sonogramm.

Die realen Winkel α und β des Test-Sonogramms

Echter Winkel α	Echter Winkel β
64,10°	60,54°

Tabelle 4.1: Die realen Winkel des verwendeten Test-Sonogramms

4.3.2 Aufnahme der Sonogramme

Das Test-Sonogramm wurde Ausgedruckt und auf eine Ebene Fläche gelegt. Durch das Smartphone wurden dann Test-Bilder des Sonogramms aufgenommen. Die Aufnahmen, die durch die Kamera erfolgten, wurden dann manuell auf einem Computer ausgemessen. Die Aufnahmen wurden in unterschiedlichen Perspektiven aufgenommen. Ziel war es, die Grenze zu finden, ab wann die Winkel des aufgenommenen Sonogramms nicht mehr verwertbar sind.

Es wurde der Orientierungssensor und das Koordinatensystem des Android Gerätes verwendet (s. Kapitel 3), um das Smartphone in eine waagerechte Position zu bringen. Durch die waagerechte Position des Smartphones während der Aufnahme sollten die Winkel dem Original entsprechen. Für die Einhaltung der waagerechten Position sind in der Kamerafunktion der App die aktuellen Lagekoordinaten angezeigt.

Es wurden Aufnahmen in verschiedenen Winkeln aufgenommen. Es wurden für die App nur zwei Achsen der drei Verfügbaren (Pitch, Roll und Azimuth) verwendet. Die X-Achse oder Pitch genannt und die Y-Achse als Roll bekannt. Diese zwei Achsen zeigen die Position des Smartphones im Koordinatensystem an. Wenn diese zwei Werte sich 0° annähern liegt das Smartphone in einer waagerechten Position.

Das Smartphone und die Neigungswinkel für die Aufnahmen

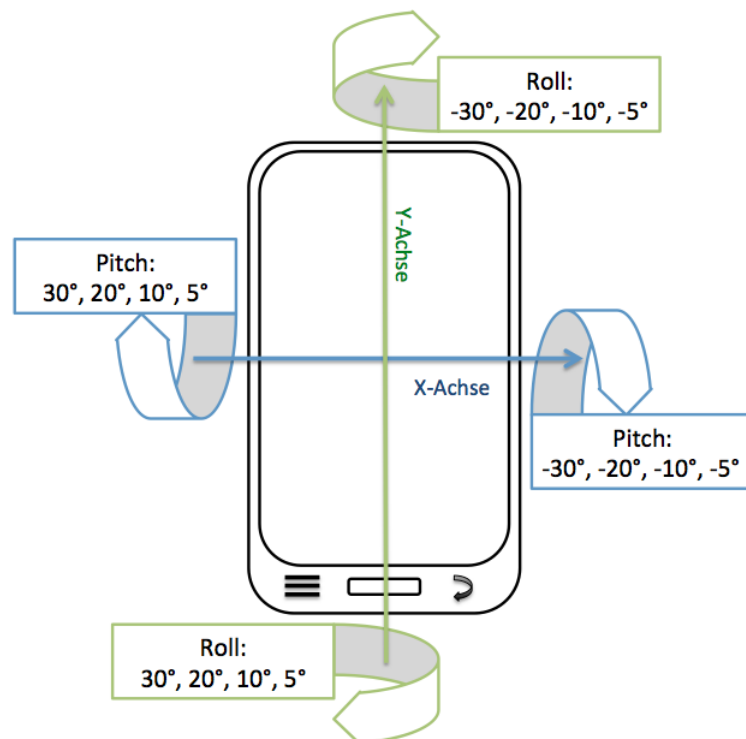


Abbildung 4.11: Die Neigungswinkel des Smartphones bei der Aufnahme des ilder

Die Neigungswinkel des Smartphones bei der Aufnahme waren 30° , 20° , 10° , 5° und 0° . Bei einer Neigung von 0° sollte der reale Winkel des Sonogramms denen aus der Aufnahme (Test-Bilder) entsprechen.

4.3.3 Lage Sonogramme

Die folgenden Tabellen repräsentieren die gemessenen Winkel für die Sonogramme, die in unterschiedlichen Neigungen (+/- Pitch und +/- Roll) des Smartphones aufgenommen worden sind.

$Fehler = gemessenerWinkel - echterWinkel$

Vermessung der 30° geneigten Sonogramme

Smartphone-Lage	α	β	Fehler α	Fehler β
Pitch 30°	$66,44^\circ$	$63,64^\circ$	$2,34^\circ$	$3,11^\circ$
Pitch -30°	$67,32^\circ$	$66,13^\circ$	$3,22^\circ$	$5,59^\circ$
Roll 30°	$58,78^\circ$	$57,02^\circ$	$-5,32^\circ$	$-3,52^\circ$
Roll -30°	$65,70^\circ$	$54,58^\circ$	$1,60^\circ$	$-5,96^\circ$

Tabelle 4.2: Aus den in Schräglage (30°) aufgenommenen Sonogrammen wurden die Winkel ausgemessen. Die Differenz zwischen tatsächlichem und gemessenem Winkel wurde berechnet.

Vermessung der 20° geneigten Sonogramme

Smartphone-Lage	α	β	Fehler α	Fehler β
Pitch 20°	$65,67^\circ$	$61,61^\circ$	$1,58^\circ$	$1,07^\circ$
Pitch -20°	$66,10^\circ$	$64,06^\circ$	$2,01^\circ$	$3,53^\circ$
Roll 20°	$59,17^\circ$	$61,54^\circ$	$-4,93^\circ$	$1,01^\circ$
Roll -20°	$65,11^\circ$	$57,70^\circ$	$1,01^\circ$	$-2,83^\circ$

Tabelle 4.3: Aus den in Schräglage (20°) aufgenommenen Sonogrammen wurden die Winkel ausgemessen. Die Differenz zwischen tatsächlichem und gemessenem Winkel wurde berechnet.

Vermessung der 10° geneigten Sonogramme

Smartphone-Lage	α	β	Fehler α	Fehler β
Pitch 10°	64,72°	61,25°	0,62°	0,71°
Pitch -10°	64,99°	61,99°	0,89°	1,46°
Roll 10°	63,16°	61,24°	-0,94°	0,70°
Roll -10°	64,62°	60,63°	0,52°	0,1°

Tabelle 4.4: Aus den in Schräglage (10°) aufgenommenen Sonogrammen wurden die Winkel ausgemessen. Die Differenz zwischen tatsächlichem und gemessenem Winkel wurde berechnet.

Vermessung der 5° geneigten Sonogramme

Smartphone-Lage	α	β	Fehler α	Fehler β
Pitch 5°	64,16°	60,77°	0,06°	0,23°
Pitch -5°	64,62°	61,29°	0,52°	0,75°
Roll 5°	64,31°	61,43°	0,21°	0,89°
Roll -5°	64,72°	60,66°	0,62°	0,13°

Tabelle 4.5: Aus den in Schräglage (5°) aufgenommenen Sonogrammen wurden die Winkel ausgemessen. Die Differenz zwischen tatsächlichem und gemessenem Winkel wurde berechnet.

Vermessung des 0° Pitch und 0° Roll (ungeneigt) Sonogramms

Smartphone-Lage	α	β	Fehler α	Fehler β
Pitch 0° und Roll 0°	64,16°	60,84°	0,06°	0,31°

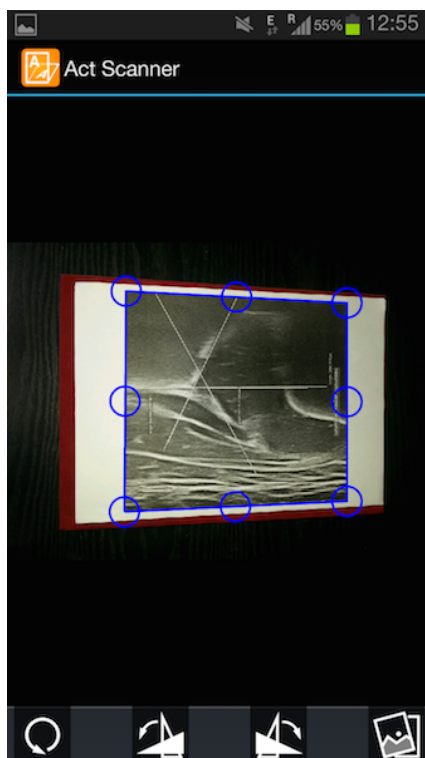
Tabelle 4.6: Aus dem in 0° Pitch und 0° Roll aufgenommenen Sonogramm wurden die Winkel ausgemessen. Die Differenz zwischen tatsächlichem und gemessenem Winkel wurde berechnet.

Die oben angeführten Tabellen beinhalten die Ergebnisse des Ersten Teils der Datenauswertung. Die erste Spalte gibt die Lage des Smartphones beim Aufnehmen der Bilder an. Die Spalte zwei und drei beinhaltet die Ergebnisse der Winkel α und β der in Neigung aufgenommen Bilder. Die Differenz (Winkel in Neigung - Winkel Real) zwischen den Neigungsbildern und dem Original wurden in den Spalten vier und fünf berechnet. Die Ergebnisse in den Tabellen zeigen uns, dass es möglich ist, die beiden Winkel in einem mit dem Smartphone aufgenommenen Bild zu messen. Die Größe des Fehlers (Winkeldifferenz) nimmt mit stärkerer Neigung des Smartphones zu.

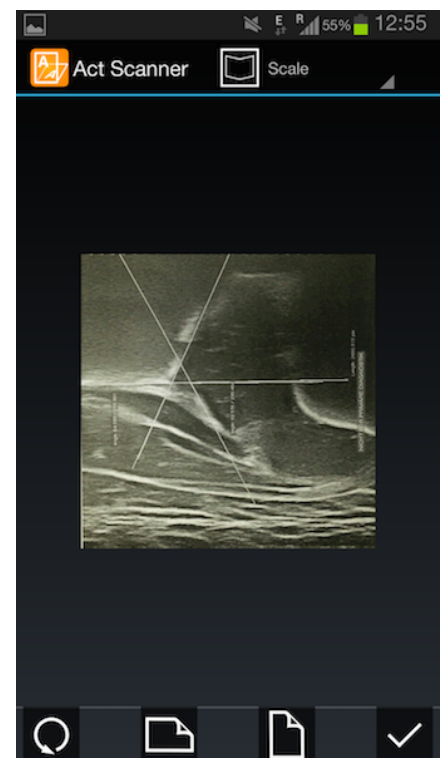
Wie in Tabelle 4.2 zu sehen ergibt sich durch die größte Neigung des Smartphones auch die größte Differenz in den Winkeln. Die Winkel der Neigungsbilder sinken demnach, je geringer die Neigung des Smartphones wird. Ab 5° Neigung verringert sich der Fehler unter 1° (Tabelle 4.5). Bei der angenommenen optimalen Lage des Smartphones (Pitch 0 und Roll 0) ergibt sich wie in Tabelle 4.6 ein Fehler von unter $0,5^\circ$.

4.3.4 Entzerzte Sonogramme

Die im vorherigen Kapitel besprochenen Test-Bilder wurden für einen weiteren Test verwendet. Der zweite Teil der Datenauswertung soll klären, ob die Test-Bilder nach einer automatischen Korrektur der Perspektive eine verbessern oder verschlechtern der Winkeldifferenz eintritt. Dazu wurden die vorher aufgenommenen Lage-Sonogramme durch eine kostenlose App aus dem Google Play Store in ihrer Perspektive korrigiert.



(a) Aufgenommenes Sonogramm vor der Korrektur



(b) Sonogramm nach der Korrektur durch Act Scanner

Abbildung 4.12: Diese Abbildung erläutert, wie die Sonogramme entzerrt (Korrektur der Perspektive) wurden. Act Scanner erkennt die Kanten in den Test-Bildern. Die Kanten mit den blauen Kreisen in 4.12a können manuell angepasst werden, falls die Kanten nicht ordnungsgemäß erkannt wurden.

Für die Perspektivenkorrektur wurde eine App mit dem Namen Act Scanner [GooglePlayStore, 2014] auf dem Android Gerät installiert. Die App ermöglicht es, Bilder zu entzerren bzw. die Perspektive zu korrigieren. Dieser Teil beschäftigt sich mit den korrigierten Bildern und vergleicht die daraus gemessenen Winkeln mit den Winkeln aus dem Test-Sonogramm.

Die folgenden Tabellen zeigen die Ergebnisse der entzerzten Sonogramme. Die korrigierten Sonogramme sind auf dieselbe Art und Weise ausgemessen worden wie die Test-Bilder. Die Ergebnisse werden in derselben Form präsentiert wie es bei den geneigten Sonogrammen der Fall war.

Es zeigt sich, dass die Act Scanner App nicht Winkelgetreu arbeitet und keine Verbesserung auftritt. Bilder bis 5° Neigung bleiben valide. Bilder mit stärkerer Neigung, können nicht in der Perspektive korrigiert werden.

Vermessung der 30° entzerzten Sonogramme

Smartphone-Lage	α	β	Fehler α	Fehler β
Pitch 30°	$67,84^\circ$	$68,97^\circ$	$3,74^\circ$	$8,44^\circ$
Pitch -30°	$65,43^\circ$	$59,21^\circ$	$1,33^\circ$	$-1,33^\circ$
Roll 30°	$66,01^\circ$	$62,49^\circ$	$1,91^\circ$	$1,95^\circ$
Roll -30°	$57,02^\circ$	$52,86^\circ$	$-7,08^\circ$	$-7,68^\circ$

Tabelle 4.7: Die entzerzten (30°) Sonogramme wurden in ihrer Perspektive korrigiert und die Winkel ausgemessen. Die Differenz zwischen tatsächlichem und gemessenem (korrigierten) Winkel wurde berechnet.

Vermessung der 20° entzerzten Sonogramme

Smartphone-Lage	α	β	Fehler α	Fehler β
Pitch 20°	$66,25^\circ$	$65,79^\circ$	$2,15^\circ$	$5,26^\circ$
Pitch -20°	$65,08^\circ$	$59,91^\circ$	$0,98^\circ$	$-0,62^\circ$
Roll 20°	$64,64^\circ$	$62,47^\circ$	$0,54^\circ$	$1,93^\circ$
Roll -20°	$60,38^\circ$	$56,61^\circ$	$-3,72^\circ$	$-3,93^\circ$

Tabelle 4.8: Die entzerzten (20°) Sonogramme wurden in ihrer Perspektive korrigiert, und die Winkel ausgemessen. Die Differenz zwischen tatsächlichem und gemessenem (korrigierten) Winkel wurde berechnet.

Vermessung der 10° entzerrten Sonogramme

Smartphone-Lage	α	β	Fehler α	Fehler β
Pitch 10°	64,94°	63,34°	0,84°	2,80°
Pitch -10°	64,06°	59,72°	-0,04°	-0,82°
Roll 10°	64,67°	62,27°	0,57°	1,73°
Roll -10°	63,27°	59,57°	-0,83°	-0,96°

Tabelle 4.9: Die entzerrten (10°) Sonogramme wurden in ihrer Perspektive korrigiert, und die Winkel ausgemessen. Die Differenz zwischen tatsächlichem und gemessenem (korrigierten) Winkel wurde berechnet.

Vermessung der 5° entzerrten Sonogramme

Smartphone-Lage	α	β	Fehler α	Fehler β
Pitch 5°	64,17°	61,50°	0,07°	0,96°
Pitch -5°	63,72°	61,00°	-0,38°	0,46°
Roll 5°	64,14°	61,41°	0,04°	0,87°
Roll -5°	63,58°	60,44°	-0,52°	-0,10°

Tabelle 4.10: Die entzerrten (5°) Sonogramme wurden in ihrer Perspektive korrigiert, und die Winkel ausgemessen. Die Differenz zwischen tatsächlichem und gemessenem (korrigierten) Winkel wurde berechnet.

4.4 Zusammenfassung

Das Ziel war es, die maximale Neigung (Kippung) des Smartphones zu bestimmen bei der die aufgenommenen Sonogramme einen Fehler unter 1° besitzen. Die Ergebnisse zeigen, dass das bei einer Lage von +/-5° (Pitch und Roll) der Fall ist. Die App Act Scanner führte bei diesem Test zu keiner weiteren Verbesserung.

Kapitel 5

Diskussion

Hüftdysplasie (DDH) ist die häufigste kinderorthopädische Krankheit. DDH führt unbehandelt zur Zerstörung des für die Bewegung essentiellen Hüftgelenks. Patienten mit unbehandelter DDH müssen mehrfach Operiert (Revision) werden. Mehr Operationen führen zu mehr Kosten die ein Gesundheitssystem tragen muss. Des weiteren verschlechtert sich auch die Lebensqualität eines Patienten nach jeder Operation. Es gibt Endemiegebiete in denen Hüftdysplasie erhöhte Inzidenzen aufweist. Naturvölker (Inuit, [Niethard, 1997]) z.Bsp.: tragen ihre Kinder mit gestreckten Beinen, dies führt zu erhöhten Inzidenzen. Ein Screening ist eine effektive Methode die eine einfache Behandlung erlaubt (Bandage) und ein vielfaches an Leid und Schmerz erspart. Afrikanische Naturvölker z.Bsp.: tragen ihre Kinder in Abduktion und Flexion (ähnlich Bandage), das wiederum reduziert die Inzidenz. Ultraschallscreenings sind zwar mit Kosten verbunden aber dennoch kosten effektiv [Thaler et al., 2011]. Durch erhöhte Geburtenraten in nicht EU Ländern wird die DDH Inzidenz weltweit zunehmen. Dies erfordert eine Lösung. Eine Android Applikation deckt ca. 80% des Smartphone Marktes ab [IDC, 2013]. Diese starken Zahlen kann man sich wegen des günstigeren Anschaffungspreises und leichten Anpassung erklären. Diese Vorteile sprechen Entwickler, Studenten und Jugendliche, die eher wenig Geld zur Verfügung haben eher an, als das iPhone das deutlich teurer bei der Anschaffung ist. Das Problem der Winkeltreue und der Neigung des Smartphones wurde durch den Einsatz von Sensoren gelöst (s. Kapitel 4.3.2). Die Neigungen des Smartphones bei einem Pitch und Roll von $\pm 5^\circ$, sind die gemessenen Winkel im Sonogramm akzeptabel (Fehler unter 1°) (s. Kapitel 4.3.3). Eine Entzerrung der in Neigung aufgenommenen Bilder brachte keine Verbesserung bei der Auswertung (s. Kapitel 4.3.4). Eine telemedizinische App für Android Smartphones wäre demnach eine günstige Lösung für Endemiegebiete, welche an Unterversorgung (Hüftsonographie und Fachwissen) leiden.

Literaturverzeichnis

- Graf, R., Lercher, K., Baumgartner, F., und Spieß, T. (2013). *Kurshandbuch für die Ausbildung in der Hüftsonographie*. LKH Stolzalpe.
- Graf, R., Lercher, K., Tschauner, C., Baumgartner, F., und Plattner, F. (2010). *Sonographie der Säuglingshüfte und therapeutische Konsequenzen: ein Kompendium*. Georg Thieme Verlag.
- Hao, Z., Dai, J., Shi, D., Xu, Z., Chen, D., Zhao, B., Teng, H., und Jiang, Q. (2014). Association of a single nucleotide polymorphism in HOXB9 with developmental dysplasia of the hip: a case-control study. *J Orthop Res*, 32(2):179–82.
- Hefti, F. (2006). *Kinderorthopädie in der Praxis*. Springer Medizin Verlag.
- Komatineni, S. und MacLean, D. (2012). *Pro Android 4*. Apress.
- Loder, R. T. und Skopelja, E. N. (2011). Review Article The Epidemiology and Demographics of Hip Dysplasia. *ISRN Orthopedics*, 2011:1–46.
- Niethard, F. U. (1997). *Kinderorthopädie*. Georg Thieme Verlag.
- Sammer, T., Back, A., und Walter, T. (2014). *Mobile Business: Management von mobiler IT in Unternehmen*. Verlag buch und netz.
- Tanenbaum, A. S. (2009). *Moderne Betriebssysteme*. Pearson Studium.
- Thaler, M., Biedermann, R., Lair, J., Krismer, M., und Landauer, F. (2011). Cost-effectiveness of universal ultrasound screening compared with clinical examination alone in the diagnosis and treatment of neonatal hip dysplasia in Austria. *J Bone Joint Surg [Br]*, 93-B:1126–1130.
- Wülker, N. (2005). *Taschenlehrbuch - Orthopädie und Unfallchirurgie*. Georg Thieme Verlag.

Internetquellen

- AndroidGuru (2014). Android Java Web Service Login. URL: <http://programmerguru.com/android-tutorial/android-java-web-service-login/>. abgerufen am 02.09.2014.
- Becker, A. (2011). Inside Android. URL: <http://www.heise.de/ct/artikel/Innenansichten-1176816.html>. abgerufen am 07.08.2014.
- BMG (2014a). BMG:BMG - E-Health/Elga. URL: http://www.bmg.gv.at/home/Schwerpunkte/E_Health_Elga/. abgerufen am 09.07.2014.
- BMG (2014b). Bundeministerium für Gesundheit: BMG - Telemedizin. URL: http://www.bmg.gv.at/home/Schwerpunkte/E_Health_Elga/Telemedizin/#top. abgerufen am 02.06.2014.
- BMG (2014c). Gesundheitstelematikgesetz 2012. URL: http://www.bmg.gv.at/home/Schwerpunkte/E_Health_Elga/Rechtsrahmen/Gesundheitstelematikgesetz_Stammfassung. abgerufen am 02.09.2014.
- BMG (2014d). Gesundheitstelematikverordnung 2013. URL: <https://www.ris.bka.gv.at/GeltendeFassung/Bundesnormen/20008732/GTelV%C2%A02013%2c%20Fassung%20vom%2002.09.2014.pdf>. abgerufen am 02.09.2014.
- DeveloperAndroid (2014a). Activities. URL: <http://developer.android.com/guide/components/activities.html>. abgerufen am 07.08.2014.
- DeveloperAndroid (2014b). Android Studio. URL: <https://developer.android.com/sdk/installing/studio.html>. abgerufen am 24.08.2014.
- DeveloperAndroid (2014c). AsyncTask. URL: <http://developer.android.com/reference/android/os/AsyncTask.html>. abgerufen am 02.09.2014.
- DeveloperAndroid (2014d). Dashboards. URL: <http://developer.android.com/about/dashboards/index.html>. abgerufen am 30.07.2014.
- DeveloperAndroid (2014e). Sensor Overview. URL: http://developer.android.com/guide/topics/sensors/sensors_overview.html. abgerufen am 10.08.2014.

- Gesundheit.gv.at (2014). Die Elektronische Gesundheitsakte – ELGA. URL: <https://www.gesundheit.gv.at/Portal.Node/ghp/public/content/ELGA/elga-elektronische-gesundheitsakte.html>. abgerufen am 09.07.2014.
- GooglePlayStore (2014). Act Scanner - Image correction. URL: <https://play.google.com/store/apps/details?id=com.liuxqsmile.imagecorrect&hl=de>. abgerufen am 10.08.2014.
- Huefte (2014). Breit Wickeln. URL: http://www.huefte.ch/html/eltern/e_wickeln_1.html. abgerufen am 10.06.2014.
- IDC (2013). Worldwide Smart Connected Device Market Crossed 1 Billion Shipments in 2012, Apple Pulls Near Samsung in Fourth Quarter, According to IDC . URL: <http://www.idc.com/getdoc.jsp?containerId=prUS24037713>. abgerufen am 06.08.2014.
- ImageJ (2014). Image Processing and Analysis in Java. URL: <http://imagej.nih.gov/ij/index.html>. abgerufen am 07.08.2014.
- Jetbrains (2014). IntelliJ IDEA. URL: <http://www.jetbrains.com/idea/>. abgerufen am 24.08.2014.
- ORF (2014). 24-Stunden-Telemedizin: Skepsis und Zuversicht. URL: <http://oe1.orf.at/artikel/380729>. abgerufen am 08.07.2014.
- Orthopaedie (2014). Orthopädie Maschen. URL: <http://www.orthopaedie-maschen.de/dysplasie.htm>. abgerufen am 11.06.2014.
- RIS (2014a). Datenschutzgesetz 2000. URL: <https://www.ris.bka.gv.at/GeltendeFassung/Bundesnormen/10001597/DSG%202000%2c%20Fassung%20vom%2002.09.2014.pdf>. abgerufen am 02.09.2014.
- RIS (2014b). Gesundheitstelematikgesetz 2012. URL: <https://www.ris.bka.gv.at/GeltendeFassung/Bundesnormen/20008120/GTelG%C2%A02012%2c%20Fassung%20vom%2002.09.2014.pdf>. abgerufen am 02.09.2014.
- Smartphonehc (2014). mHealth Categories - Smart Phone Health Care. URL: <http://www.smartphonehc.com/tag/mhealth-categories/>. abgerufen am 10.07.2014.
- Statista (2014a). Prognose zu Marktanteilen von Smartphone-Betriebssystemen 2018. URL: <http://de.statista.com/statistik/daten/studie/182363/umfrage/prognostizierte-marktanteile-bei-smartphone-betriebssystemen/>. abgerufen am 22.07.2014.

- Statista (2014b). Worldwide Smart Connected Device Market Crossed 1 Billion Shipments in 2012, Apple Pulls Near Samsung in Fourth Quarter, According to IDC . URL: <http://de.statista.com/statistik/daten/studie/208599/umfrage/anzahl-der-apps-in-den-top-app-stores/>. abgerufen am 07.08.2014.
- Toennis (2014). Diagnosis and Treatment of Developmental Hip Dysplasia – Our Experience. URL: <http://www.dr-toennis.de/pages/vortraege/diagnosis-and-treatment-of-developmental-hip-dysplasia---our-experience.php>. abgerufen am 17.07.2014.
- Tumblr (2014). Human Position. URL: <http://weareallafricans.tumblr.com/post/24094216773/samburu-mother-carrying-her-baby-on-her-back>. abgerufen am 10.06.2014.
- UnitedNations (2014). World Population Prospects: The 2012 Revision. URL: http://esa.un.org/unpd/wpp/unpp/panel_indicators.htm. abgerufen am 01.09.2014.
- WHO (2014). E-Health. URL: <http://www.who.int/trade/glossary/story021/en/>. abgerufen am 08.07.2014.
- Wikipedia (2014). Wikipedia - Spreizhose. URL: <http://www.de.wikipedia.org/wiki/Spreizhose>. abgerufen am 10.06.2014.

Eidesstattliche Erklärung

zur Arbeit mit dem Titel

TELEMEDIZIN ÜBER EINE MOBILE APPLIKATION IN UN- TERVERSORGTEN ENDEMIEGEBIETEN

“Ich erkläre hiermit ehrenwörtlich, daß ich die vorliegende Arbeit selbständig angefertigt habe. Alle aus fremden Quellen direkt oder indirekt übernommenen Gedanken sind als solche kenntlich gemacht und in der Bibliographie angeführt. Die Arbeit wurde bisher keiner anderen Prüfungsbehörde vorgelegt und noch nirgends veröffentlicht.”

zur Arbeit mit dem Titel

Graz, 2. September 2014

(DZANAN MLIVO, BSC)