



Konzeption und Realisierung einer mobilen Android Anwendung zur feedbackgestützten Kommunikation zwischen Schwangeren und betreuendem Geburtshelfer

Bachelorarbeit an der Universität Ulm

Vorgelegt von:

Jessica Tatjana Hohn
jessica.hohn@uni-ulm.de

Gutachter:

Prof. Dr. Manfred Reichert

Betreuer:

Dr. Rüdiger Pryss

2018

Fassung 4. Juli 2018

© 2018 Jessica Tatjana Hohn

This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 3.0 License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc-sa/3.0/de/> or send a letter to Creative Commons, 543 Howard Street, 5th Floor, San Francisco, California, 94105, USA.

Satz: PDF- \LaTeX 2 ϵ

Kurzfassung

Der Verlauf einer Schwangerschaft ist ein entscheidender Prädiktor für die Gesundheit und Entwicklung des Kindes, sowie für die spätere Mutter-Kind-Beziehung, sodass es für Geburtshelfer/Gynäkologen unerlässlich ist, potentielle Risikofaktoren rechtzeitig zu erkennen. Derartige Risikofaktoren lassen sich mittels evidenzbasierter Fragebögen, wie dem KINDEX, durch Geburtshelfer/Gynäkologen ermitteln. Deren Durchführung gestaltet sich jedoch im klinischen Alltag aus zeitlichen Gründen oft problematisch.

Aus diesem Grund wird in der vorliegenden Arbeit die Konzeption und Realisierung einer Android App vorgestellt, welche Schwangeren die Möglichkeit bietet die deutsche oder englische Version des KINDEX zu mehreren Zeitpunkten im Verlauf der Schwangerschaft mobil auszufüllen. Die somit erfassten Antworten werden ausgewertet und dem behandelnden Geburtshelfer/Gynäkologen über die gleiche Anwendung präsentiert. Somit erhält dieser einen Einblick in die aktuelle Situation seiner Patientin und kann entsprechende Vorkehrungen treffen, um eine individuelle Geburtsvorsorge zu gewährleisten. Es wird auf die Anforderungen und Architektur der entwickelten Anwendung eingegangen. Des Weiteren wird ein Einblick in Teile der Implementierung und das visuelle Resultat gegeben.

Inhaltsverzeichnis

| | | |
|----------|--|-----------|
| 1 | Einleitung | 1 |
| 1.1 | Problemstellung | 1 |
| 1.2 | Zielsetzung | 2 |
| 1.3 | Struktur der Arbeit | 2 |
| 2 | Verwandte Arbeiten | 5 |
| 2.1 | Mobiles Schwangerschaftsunterstützendes System | 5 |
| 2.2 | Die KINDEX-App | 6 |
| 3 | Anforderungsanalyse | 9 |
| 3.1 | Nicht-funktionale Anforderungen | 9 |
| 3.2 | Funktionale Anforderungen | 10 |
| 4 | Architektur der Anwendung | 15 |
| 4.1 | Ablaufbeschreibung | 15 |
| 4.2 | Architekturkonzept | 24 |
| 4.2.1 | Grundlegender Aufbau | 24 |
| 4.2.2 | Backend | 25 |
| 4.2.3 | Datenbank | 26 |
| 4.3 | Verwendete Bibliothek - Retrofit | 31 |
| 5 | Ausgewählte Implementierungsdetails | 33 |
| 5.1 | Fragebogenblock anzeigen | 33 |
| 5.2 | Konsistenzprüfung der Fragen | 35 |
| 5.3 | Fragebogen abschicken | 38 |
| 6 | Vorstellung der Anwendung | 43 |
| 6.1 | Login | 43 |
| 6.2 | Registrierung | 44 |
| 6.3 | Passwort vergessen | 45 |
| 6.4 | Hilfestellung für Patientinnen | 45 |

Inhaltsverzeichnis

| | | |
|----------|---|-----------|
| 6.5 | Rollen-unabhängige Funktionalitäten | 46 |
| 6.5.1 | Änderung des Passworts | 46 |
| 6.5.2 | Kontaktierung der Entwickler | 47 |
| 6.5.3 | Kontaktinformationen | 47 |
| 6.5.4 | Änderung der Spracheinstellung | 47 |
| 6.6 | Funktionalitäten für Patientinnen | 48 |
| 6.6.1 | Beginn der Behandlung | 48 |
| 6.6.2 | Startseite Patientin | 49 |
| 6.6.3 | Ausfüllen eines Fragebogens | 51 |
| 6.6.4 | Einsehen eines Feedbacks | 53 |
| 6.6.5 | Beenden der Behandlung | 53 |
| 6.7 | Funktionalitäten für Ärzte | 54 |
| 6.7.1 | Startseite Arzt | 54 |
| 6.7.2 | Detailansicht einer Patientin | 54 |
| 6.7.3 | Schicken eines Feedbacks | 55 |
| 6.8 | Menüs | 56 |
| 6.9 | Logout | 57 |
| 7 | Abgleich der Anforderungen | 59 |
| 7.1 | Nicht-funktionale Anforderungen | 59 |
| 7.2 | Funktionale Anforderungen | 60 |
| 8 | Fazit | 63 |
| 8.1 | Zusammenfassung | 63 |
| 8.2 | Ausblick | 64 |

1

Einleitung

1.1 Problemstellung

Zahlreiche psychosoziale Faktoren in der pränatalen Phase beeinflussen die Gesundheit und Entwicklung des Kindes, sowie die spätere Mutter-Kind-Bindung. Neben protektiven Faktoren, wie das Kohärenzgefühl zu Beginn einer Schwangerschaft [1], ist vor allem das Wissen um Risikofaktoren von großer Relevanz, um negative Entwicklungen rechtzeitig erkennen zu können.

Die Veränderungen des weiblichen Körpers während der Schwangerschaft gehen häufig mit physischen und psychischen Beeinträchtigungen [2], wie Depression, Angstzuständen oder einem erhöhten Cholesterinspiegel [3], einher. Diese vermindern nicht nur die Lebensqualität der werdenden Mutter, sondern wirken sich oft auch negativ auf das ungeborene Kind aus [4], sodass der frühzeitigen Erkennung von Belastungsfaktoren eine große Bedeutung zukommt. Auslöser für Stress können beispielsweise Konflikte in der Partnerschaft, die derzeitige Wohnsituation oder das Herkunftsland der werdenden Eltern sein [5]. Aber auch Missbrauch in der Kindheit kann eine große Beeinträchtigung für die mentale Gesundheit der werdenden Mutter, sowie die spätere Mutter-Kind-Beziehung darstellen [6]. Um eine, auf die Schwangere bestmöglich angepasste, prä- und postnatale Betreuung bieten zu können, ist es von großer Bedeutung alle möglichen Risikofaktoren zu kennen. Dies gilt sowohl für den Gynäkologen¹ als auch für die werdende Mutter.

Zur Erfassung dieser Faktoren wurde an der Universität Konstanz ein strukturierter Fra-

¹ Aus Gründen der besseren Lesbarkeit und Verständlichkeit wird in dieser Arbeit von einer geschlechts-spezifische Differenzierung abgesehen und stets die Formulierung Gynäkologe und Arzt verwendet. Diese Formulierung schließt Hebammen und Geburtshelfer allgemein nicht aus.

1 Einleitung

gebogen, der Konstanzer Index (KINDEX), entwickelt. Dieser enthält 23 Fragen, welche elf Belastungsfaktoren abdecken.² Der KINDEX wird als valides Erhebungsinstrument für psychosoziale Belastungsfaktoren während der Schwangerschaft gewertet [6], [7] und liegt zum aktuellen Zeitpunkt sowohl in Deutsch, Englisch, Spanisch als auch Griechisch vor.

1.2 Zielsetzung

Die zentrale Aufgabe dieser Arbeit bestand in der Konzeption und Realisierung einer mobilen Anwendung für die Plattform Android, welche die Kommunikation zwischen schwangeren Patientinnen und deren betreuendem Arzt erleichtern und somit eine individualisierte Geburtsvorsorge ermöglichen soll. Hierbei kann die Patientin Fragebögen ausfüllen, die nach dem KINDEX ausgewertet werden. Das daraus resultierende Feedback ist für beide Parteien einsehbar und ermöglicht so einen Informationsaustausch, ohne dass dafür ein Treffen stattfinden muss. Zusätzlich müssen Ärzte ihre geringen zeitlichen Ressourcen nicht zur Durchführung und Evaluation aufbringen, da die Auswertung voll automatisiert verläuft.

1.3 Struktur der Arbeit

Die zugrundeliegende Struktur dieser Arbeit beginnt mit einer Heranführung an die Thematik durch eine Vorstellung verwandter Arbeiten (Kapitel 2). Anschließend folgt eine Übersicht über die gestellten (nicht-)funktionalen Anforderungen an die Anwendung. Kapitel 4 bietet einen Überblick der Applikationsarchitektur, welche sich aus den internen Abläufen und dem konzeptionellen Aufbau der Implementierung zusammensetzt. Daraufhin werden einige Codeausschnitte erläutert, um einen direkten Einblick in die Programmierung zu erhalten. Nachdem der Hintergrund der Konzeption ausreichend beleuchtet wurde, wird in Kapitel 6 die Realisierung der Anwendung, gestützt von

² Der komplette Fragebogen ist in Interviewform unter folgendem Link abrufbar:
<http://www.babyforum-landkreis-konstanz.de/de/aktuelles/KINDEX-Interview.pdf>

1.3 Struktur der Arbeit

Screenshots, vorgestellt. Im Anschluss werden die im Kapitel 3 gestellten Anforderungen mit dem Endergebnis abgeglichen, um zu sehen, ob die Realisierung den Vorgaben entspricht. Das letzte Kapitel dient als Zusammenfassung der Arbeit.

2

Verwandte Arbeiten

Dieser Abschnitt soll als eine Heranführung an die Thematik dienen, indem Arbeiten vorgestellt werden, die ebenfalls im Zusammenhang mit Applikationen für Schwangere stehen.

2.1 Mobiles Schwangerschaftsunterstützendes System

Nach einer Studie des Sunway University College [8] wurden 15 Paare in einem privaten Krankenhaus und zwei Geburtskliniken zu ihrem Interesse an einem mobilen schwangerschaftsunterstützenden System befragt. Dafür füllten die Probanden einen Fragebogen aus, der einem Überblick zu der Bereitschaft der Nutzung einer solchen Anwendung, sowie über Erwartungen an die Unterstützungsformen diente.

Die Ergebnisse zeigten, dass alle Studienteilnehmer daran interessiert waren ein mobiles System als Quelle für Informationen zur Gesundheit in der Schwangerschaft zu nutzen.

Folgende Unterstützungsformen wurden dabei von den meisten Probanden erwarten:

- Ein Berechner für den Stichtag
- Informationen zu den Stadien der Schwangerschaft
- Erinnerungen an Kontrolluntersuchungen
- Auflistung nahegelegener Krankenhäuser

Aufbauend auf den Ergebnissen der Befragung wurde ein prototypisches System entwickelt. Dieses beinhaltete insgesamt sechs verschiedene Unterstützungsmodule.

2 Verwandte Arbeiten

Modul 1 bot generelle Informationen zur Gesundheitsvorsorge für Schwangere, damit diese einen Überblick erhalten, welche Medikamente eingenommen werden können oder welche Maßnahmen in bestimmten Situationen ergriffen werden sollten. Als weitere Unterstützung wurde Modul 2 eingerichtet, welches es ermöglichte Partner oder nahe Verwandte in Notfallsituationen, wie beispielsweise einem Sturz zu informieren, sowie relevante Kontaktinformationen für nahegelegene Krankenhäuser oder die Ambulanz schnell abzurufen. Um einen Überblick über das aktuelle Schwangerschaftsstadium zu erhalten, wurde in Modul 3 ein Countdown bis zum Stichtag eingerichtet, sowie Informationen zum Entwicklungsstadiums des Kindes und der werdenden Mutter bereitgestellt. Die Entwicklung wurde hierbei zusätzlich mit Grafiken ergänzt, die einen wachsenden Fötus in der Gebärmutter zeigen. Damit die werdenden Eltern nicht den Überblick über anstehende Kontrolltermine verlieren, wurde mittels Modul 4 ein Schwangerschaftskalender und -tagebuch in den Prototyp mit aufgenommen. Hier konnten Gewichtszunahme und andere schwangerschaftsbedingte Veränderungen notiert, sowie Ultraschallbilder eingebunden werden. Die Module 5 und 6 beinhalteten eine Übersicht von einfachen Sportübungen, die während der Schwangerschaft geeignet sind, sowie Hilfestellungen nach der Geburt.

2.2 Die KINDEX-App

Bei der in [7] präsentierten KINDEX-App wurde eine Tablet-Anwendung für den gleichnamigen Fragebogen entwickelt und anschließend, von 2012 bis 2014, von etwa 1000 Schwangeren getestet. Ziel dieser Studie war es, die Akzeptanz eines solchen Instruments von Seiten der behandelnden Gynäkologen und den Patientinnen zu ermitteln. Schwangere füllten den Fragebogen in den Praxen an einem Tablet aus, wobei Fragen übersprungen werden konnten. Im Anschluss hatte der Arzt die Möglichkeit die Antworten sowie die Auswertung eines Fragebogens einzusehen. Des Weiteren bestand die Möglichkeit zu dokumentieren, ob Ergebnisse bereits eingesehen oder besprochen wurden.

Auswertungen der KINDEX-App Studie ergaben, dass es sich um ein akzeptiertes und praktikables Instrument handelt. Damit ein Screening nicht nur für deutschsprachige

Schwangere möglich ist, wurde der Fragebogen ins Spanische und Griechische übersetzt [9].

Neben den Vorteilen der App, gibt es jedoch auch Nachteile. So nimmt die eben beschriebene prototypische Umsetzung der KINDEX-App keine Rücksicht auf die Tatsache, dass das Ausfüllen des Fragebogens emotional sehr belastend sein kann. Die Beantwortung der Fragen in einem privaten Umfeld und in der, für die schwangere Patientin geeigneten, Geschwindigkeit ist daher ein nicht außer Acht zu lassender Faktor. Des Weiteren wurde den behandelten Ärzten keine Vorbereitungszeit auf die Antworten eingeräumt, welche abhängig von dem Resultat der Auswertung durchaus erforderlich sein kann. Zusätzlich bietet die vorgestellte Umsetzung Patientinnen nicht die Möglichkeit die Auswertung der eigenen Angaben einzusehen.

Diese Aspekte wurden bei der Implementierung der in dieser Arbeit vorgestellten Anwendung berücksichtigt.

3

Anforderungsanalyse

In diesem Kapitel sollen die Ansprüche an die Anwendung beschrieben werden. Hierbei unterscheidet man zwischen zwei Anforderungstypen: Nicht-funktionale Anforderungen, wie beispielsweise das zugrunde liegende Betriebssystem, sind unabhängig von der Funktionalität eines Systems, während funktionalen Anforderungen ausschließlich den Funktionsumfang eines Systems umfassen, wie beispielsweise die Möglichkeit die Spracheinstellungen zu ändern.

3.1 Nicht-funktionale Anforderungen

| Nr. | Anforderung und Beschreibung |
|-----|---|
| 1 | Betriebssystem Die Anwendung soll für das Android Betriebssystem implementiert werden. |
| 2 | Android-Versionen Die Anwendung soll ab der Version 5.0, welche dem API-Level 21 entspricht, verwendbar sein. |
| 3 | Verständlichkeit Die Nutzung der Anwendung soll leicht verständlich und ohne ausführliche Erklärung möglich sein. |
| 4 | Bedienbarkeit Der Nutzer soll die Anwendung intuitiv bedienen können. |

Tabelle 3.1: Nicht-funktionale Anforderungen

3.2 Funktionale Anforderungen

| Nr. | Anforderung und Beschreibung |
|-----------------------------------|---|
| <i>Allgemein</i> | |
| 1 | Registrierung Zur Nutzung der App muss man ein Benutzerkonto anlegen. Hierbei soll die Registrierung einer Patientin direkt in der App möglich sein. Ärzte werden durch die Universität im System registriert. |
| 2 | Verifizierung der E-Mail-Adresse Eine Registrierung ist nur erfolgreich, wenn eine gültige E-Mail-Adresse angegeben wurde. Es wird eine Verifizierungsmail geschickt, welche bestätigt werden muss, um sich später einloggen zu können. |
| 3 | Passwort vergessen Handelt es sich um einen verifizierten Nutzer, so soll dieser sein Passwort auf dem Loginscreen zurücksetzen können. Es soll dann eine Mail mit Anweisungen an die angegebene E-Mail-Adresse gesendet werden. |
| 4 | Rollenkonzept Die Anwendung soll zwischen zwei Nutzer-Rollen unterscheiden: Arzt und Patientin. Diese sollen die gleiche Anwendung unterschiedlich nutzen können. Beim Login soll die Rolle für den weiteren Anwendungsverlauf ausgewählt werden können. |
| 5 | Nutzung nur mit Internetverbindung Die Anwendung soll nur mit bestehender Internetverbindung nutzbar sein. |
| <i>Allgemein - nach Einloggen</i> | |
| 6 | Passwortrücksetzung Es soll dem Nutzer möglich sein, sein Passwort zu ändern. Hierzu muss ein neues Passwort eingegeben und wiederholt werden. |

| | |
|------------------------|---|
| 7 | Entwickler kontaktieren Über ein Dialogfeld sollen die Entwickler (Uni Konstanz/Ulm) kontaktiert werden können. Es sollen Probleme oder Fragen eingegeben werden können, welche als E-Mail an die Uni Konstanz/Ulm versendet werden. |
| 8 | Kontaktinformationen In der Anwendung sollen Kontaktinformationen der beteiligten Universitäten und Entwicklern hinterlegt sein. |
| 9 | Änderung der Spracheinstellung Die Sprache der Anwendung soll selbstständig geändert werden können. Hierbei stehen dem Benutzer die serverseitig-unterstützten Sprachen zur Verfügung (derzeit Deutsch und Englisch). Standardmäßig soll die Anwendung in der Systemsprache (defaultmäßig in Englisch, wenn die Systemsprache nicht unterstützt wird) geschrieben sein. Die Sprache soll während der Laufzeit geändert werden und permanent gespeichert sein. |
| <i>Rolle Patientin</i> | |
| 10 | Hilfestellung zur Anwendung Die Patientin soll auf dem Loginscreen die Möglichkeit haben, sich schrittweise die Nutzung der Anwendung erklären zu lassen. |
| 11 | Login als Patientin Nachdem sich die Patientin verifiziert hat, soll sie sich mit den bei der Registrierung eingegebenen Daten einloggen können. Die nachfolgenden Anforderungen sollen nur möglich sein, wenn die Patientin eingeloggt ist. |
| 12 | Behandlungsbeginn: Kontoverknüpfung mit dem behandelnden Arzt Um einen Fragebogen ausfüllen zu können, muss das Konto der Patientin mit dem ihres behandelnden Arztes verbunden sein. Hierfür soll die Behandlungsstudie des Gynäkologen und das Passwort für die Teilnahme eingegeben werden. Diese Daten müssen davor ausgetauscht worden sein und der Arzt muss bereits registriert und verifiziert worden sein. |

13 **Beenden der Behandlung**

Die Patientin soll jederzeit die Behandlung beenden und somit die Verknüpfung ihres Kontos mit dem des Arztes aufheben können. Nach Behandlungsbeendigung kann der Gynäkologe die Fragebögen der Patientin nicht mehr einsehen.

Es soll möglich sein die Behandlung bei dem gleichen Arzt wieder aufzunehmen. Dann können Fragebögen wieder von beiden Parteien eingesehen werden.

14 **Übersicht der Fragebögen und des Feedbacks auf der Startseite**

Die Patientin soll auf ihrer Startseite eine Übersicht über alle abgeschickten und abgebrochenen Fragebögen, sowie die erhaltenen Feedbacks haben.

15 **Ausfüllen eines neuen Fragebogens**

Es sollen beliebig viele Fragebögen ausgefüllt werden können. Nach dem Ausfüllen ist eine Bestätigung der Patientin notwendig, dass die Daten an das Backend übermittelt werden und somit der Arzt den Fragebogen einsehen kann.

16 **Feste Abfolge der Fragen**

Die Abfolge der Fragen soll festgelegt sein. Der Patientin soll es nicht möglich sein, zwischen Fragen vor oder zurück zu navigieren. Es soll nur die aktuell angezeigte Frage beantwortet werden können. Eine bereits beantwortete Frage kann rückwirkend nicht geändert werden. Während des Ausfüllens eines Fragebogens soll auf keine andere Funktionalität der Anwendung zugegriffen werden können.

17 **Anzeige des Fragebogenfortschritts**

Beim Ausfüllen des Fragebogens soll jederzeit ersichtlich sein, wie viele Fragen bereits beantwortet wurden. Hierzu soll eine Progressbar den prozentualen Fortschritt anzeigen.

18 **Abbruch des Fragebogens**

Das Ausfüllen eines Fragebogens soll jederzeit abgebrochen werden können. Die bis dahin beantworteten Fragen sollen lokal gespeichert werden und die restlichen Fragen können zu einem späteren Zeitpunkt beantwortet werden.

19 **Verwerfen eines Fragebogens**

Nach dem vollständigen Ausfüllen eines Fragebogens soll die Patientin die Möglichkeit haben, den Fragebogen zu verwerfen, sodass dieser nicht an den Arzt weitergeleitet wird. Dieser Fragebogen soll verworfen/gelöscht und auch nicht mehr angezeigt werden.

20 **Anzeige des Feedbackstatus eines Fragebogens**

Wurde ein Fragebogen an den Arzt übermittelt soll der Patientin angezeigt werden, ob es ein Feedback des Arztes gibt, oder nicht.

21 **Einsehen eines Feedbacks**

Ist Feedback für einen Fragebogen vorhanden, soll dieses eingesehen werden können. Es sollen Informationen zum Fragebogen angezeigt werden, wie das Ausfülldatum und die Anzahl der kritisch beantworteten Fragen. Bei kritisch beantworteten Fragen sollen Behandlungsvorschläge angezeigt werden.

Rolle Arzt

22 **Login als Arzt**

Mit den bei der Registrierung vergebenen Daten (E-Mail-Adresse und Passwort) soll sich der Arzt einloggen können.

Die nachfolgenden Anforderungen sollen nur möglich sein, wenn der Arzt eingeloggt ist.

23 **Patientenübersicht**

Auf seiner Startseite soll der Gynäkologe eine Übersicht all seiner Patientinnen angezeigt bekommen. Wurde ein Fragebogen einer Patientin übermittelt, den der Arzt noch nicht zurückgeschickt hat, soll es einen entsprechenden Hinweis geben.

| | |
|----|---|
| 24 | Detailansicht einer Patientin Hat der Arzt eine Patientin auf der Startseite ausgewählt, so soll ihm der volle Name, die E-Mail-Adresse und eine Liste der ausgefüllten Fragebögen der Patientin angezeigt werden. |
| 25 | Detailansicht eines Fragebogens Hat der Arzt einen Fragebogen aus der Detailansicht einer Patientin ausgewählt, soll er eine Detailansicht dieses Fragebogens erhalten. Hier soll er das Ausfülldatum und die Anzahl der kritisch beantworteten Fragen angezeigt bekommen. Für jede kritisch beantwortete Frage soll der entsprechende Behandlungsvorschlag angezeigt werden. |
| 26 | Schicken eines Feedbacks Stimmt der Arzt den Behandlungsvorschlägen zu, soll er das Feedback an die Patientin schicken können. Erst ab diesem Zeitpunkt soll es für die Patientin einsehbar sein. Wurde ein Feedback zu einem Fragebogen verschickt, soll dieses nicht zurückgenommen oder geändert werden können. |

Tabelle 3.2: Funktionale Anforderungen

4

Architektur der Anwendung

In diesem Kapitel wird zunächst der generelle Ablauf (Abschnitt 4.1) innerhalb der Anwendung beleuchtet. Anschließend wird die Architektur in Abschnitt 4.2 vorgestellt. Hierbei wird zunächst der grundlegende Aufbau betrachtet und anschließend der Architekturbereich, der zur Kommunikation mit dem Backend benötigt wird, sowie die Struktur der lokalen Datenbank. Zum Abschluss wird in Abschnitt 4.3 die verwendete Bibliothek Retrofit vorgestellt.

4.1 Ablaufbeschreibung

Um einen genaueren Einblick in die Prozesse und Abläufe der KINDEX Applikation zu erhalten, werden diese anhand von BPMN¹ Prozessmodellen illustriert und erklärt. Da es sich hierbei um eine Anwendung handelt, die auf einem Rollenkonzept basiert, gibt es zwei verschiedene Hauptprozesse und einige Subprozesse, die nur für eine der beiden Rollen ausgeführt werden. Andere Prozesse haben für beide Rollen den gleichen Verlauf und werden daher auch nur einmal beschrieben.

Hinweis:

Weiße Briefumschläge symbolisieren eine GET Anfrage an die API.

Schwarze Briefumschläge symbolisieren eine POST Anfrage an die API.

Um der Chronologie der Anwendungsnutzung zu folgen, wird als erstes auf den Registrierungsprozess eingegangen. Der in Abbildung 4.1 veranschaulichte Prozess wird nur

¹ Business Process Model and Notation

4 Architektur der Anwendung

von einer neuen Patientin durchgeführt, um sich ein Nutzerkonto anzulegen. Hierbei ist es notwendig einige Daten anzugeben, welche an die API² übermittelt werden, um die neue Patientin zu registrieren. Waren alle Eingaben korrekt, so bekommt der Nutzer dies durch einen Dialog mitgeteilt und erhält eine E-Mail an die angegebene Adresse. Hierbei handelt es sich um eine Verifizierungsmail, die bestätigt werden muss, um die Anwendung nutzen zu können.

Ärzte werden manuell in das System eingespeist und sind somit automatisch verifiziert und können die Anwendung nutzen.

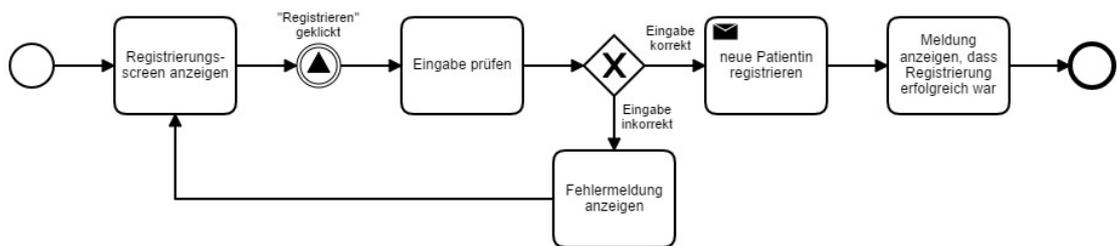


Abbildung 4.1: Registrierungsprozess

Um Zugriff auf die Funktionen der Anwendung zu haben, ist es notwendig sich einzuloggen. Für Arzt und Patientin verläuft dieser Vorgang, der in Abbildung 4.2 dargestellt wird, gleich. Beim Öffnen der Applikation wird der Loginscreen angezeigt, mit der Aufforderung seine Anmeldedaten, bestehend aus E-Mail-Adresse und Passwort, einzugeben, sowie die eigene Rolle auszuwählen. Nach Drücken des „Anmelden“ Buttons werden zunächst die Eingaben überprüft und gegebenenfalls eine Fehlermeldung angezeigt. Anschließend wird eine Anfrage an die API gesendet, die die Rolle des Nutzers bestimmt. Wurde beim Einloggen die falsche Rolle gewählt, so wird der Nutzer darauf hingewiesen und muss seine Angaben korrigieren. Bei korrekter Eingabe und Rolle wird der Nutzer angemeldet und der Hauptprozess wird eingeleitet. Dieser ist für eine Patientin in Abbildung 4.3 und für einen Arzt in Abbildung 4.8 beschrieben.

Hat sich eine Patientin eingeloggt, wird eine Anfrage für ihr Profil geschickt. Anschließend wird überprüft, ob die Nutzerin bereits Teil einer Studie ist, sich also in Behandlung eines Gynäkologen befindet. Ist dies nicht der Fall, so wird der Subprozess „Behandlung

² Application Programming Interface (Schnittstelle für Anwendungen)

4.1 Ablaufbeschreibung

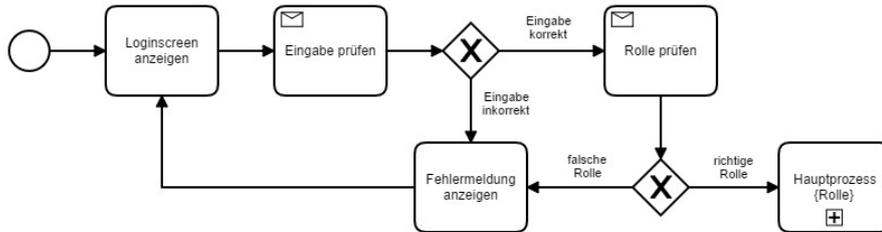


Abbildung 4.2: Loginprozess

beginnen", der in Abbildung 4.4 illustriert ist, eingeleitet. Ansonsten werden ihre Answersheets abgerufen und anschließend aus der Datenbank abgebrochenen Fragebögen ausgelesen. Um die Answersheets auch mit dem Abgabedatum anzeigen zu können, werden noch die Details der einzelnen Answersheets abgefragt und anschließend in der Startseite der Patientin angezeigt.

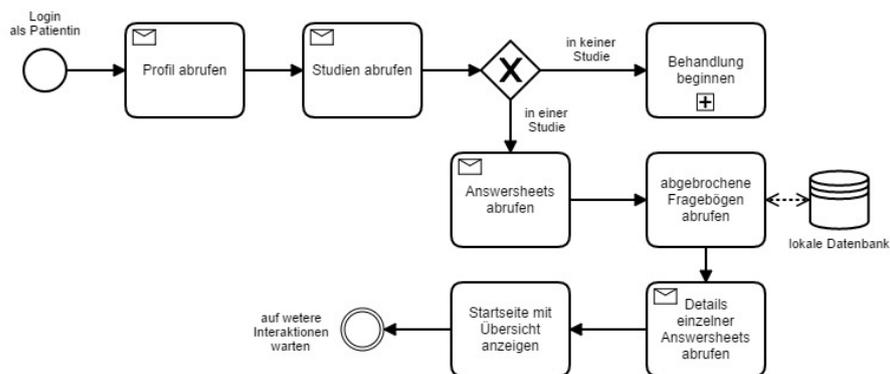


Abbildung 4.3: Hauptprozess Patientin

Damit eine Patientin die Hauptfunktionalitäten der KINDEX Applikation nutzen kann, muss sie Teil einer Studie sein. Um eine Behandlung zu beginnen (siehe Abbildung 4.4), und somit in eine Studie eingeschrieben zu werden, werden alle verfügbaren Studien abgerufen und der „Behandlung beitreten“ Screen wird angezeigt. Um eine Übersicht über die Studien zu erhalten, ist ein Klick auf das Feld „Wählen Sie eine Studie“ nötig. Nachdem die Patientin eine Auswahl getroffen hat, wird die Übersicht eingeklappt und es muss das Zugangspasswort eingegeben werden. Nach Drücken des „Beitreten“ Buttons, werden die Eingaben überprüft und bei korrekten Angaben wird die Patientin in die

4 Architektur der Anwendung

Studie eingeschrieben und der Hauptprozess wird eingeleitet.

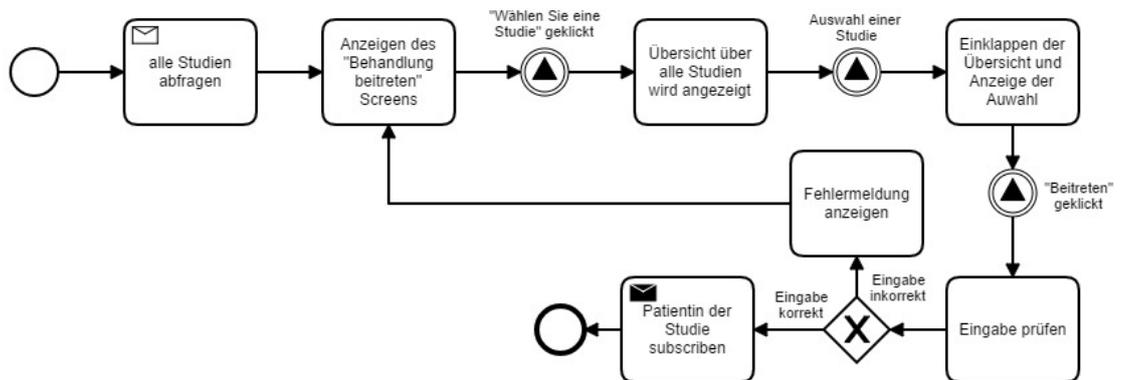


Abbildung 4.4: Prozess Beginn der Behandlung

Eine der Hauptfunktionalitäten der Applikation ist das Beantworten von Fragebögen. Den in Abbildung 4.5 beschriebenen Prozess kann die Patientin einleiten, indem sie einen bereits angefangenen Fragebogen wieder öffnet und diesen weiter ausfüllt oder durch Starten eines neuen Fragebogens. Für die erste Variante werden alle bereits gegebenen Antworten aus der lokalen Datenbank ausgelesen und anschließend gelöscht. Nach diesem Zwischenschritt ist der Ablauf für beide Varianten gleich. Es wird die Struktur des Fragebogens von der API angefragt. Eine solche Struktur wird dann in einzelne Frageblöcke unterteilt und anschließend der aktuelle Block angezeigt. Wurde auf „Weiter“ gedrückt, um den nächsten Fragenblock anzuzeigen, wird zunächst geprüft, ob alle benötigten Fragen beantwortet wurden. Ist dies der Fall, so werden die aktuell gegebenen Antworten in einer Liste gespeichert und der nächste Frageblock wird angezeigt. Fehlen jedoch Antworten, so wird die Patientin darauf hingewiesen und muss diese ergänzen, um fortfahren zu können. Sollte der Fragebogen abgebrochen werden wollen, wird zunächst eine Bestätigung gefordert, dass ein Abbruch erwünscht ist. Wird diese erteilt, wird die Liste, mit den zu diesem Zeitpunkt gegebenen Antworten, in der lokalen Datenbank abgelegt und die Patientin gelangt zurück auf ihre Startseite. Kam es jedoch nicht zu einer Abbruchbestätigung, kann das Beantworten der Fragen fortgesetzt werden. Nachdem alle Frageblöcke vollständig ausgefüllt wurden kann der Fragebogen an den behandelnden Arzt übermittelt oder verworfen werden. Hat sich die

Patientin für das Abschicken ihrer Antworten entschieden, wird hierfür eine Zustimmung zur Datenübermittlung gefordert. Wird diese erteilt, so wird das Anwersheet, bestehend aus allen gegebenen Antworten des Fragebogens, an die API übermittelt und die Patientin wird auf ihre Startseite weitergeleitet. Sonst wird der letzte Frageblock angezeigt. Sollte sich die Patientin für das Verwerfen des Fragebogens entscheiden, wird sie darauf hingewiesen, dass alle Antworten verworfen werden. Wird dem nicht zugestimmt, wird der letzte Frageblock angezeigt, sonst gelangt die Patientin zurück zu ihrer Startseite.

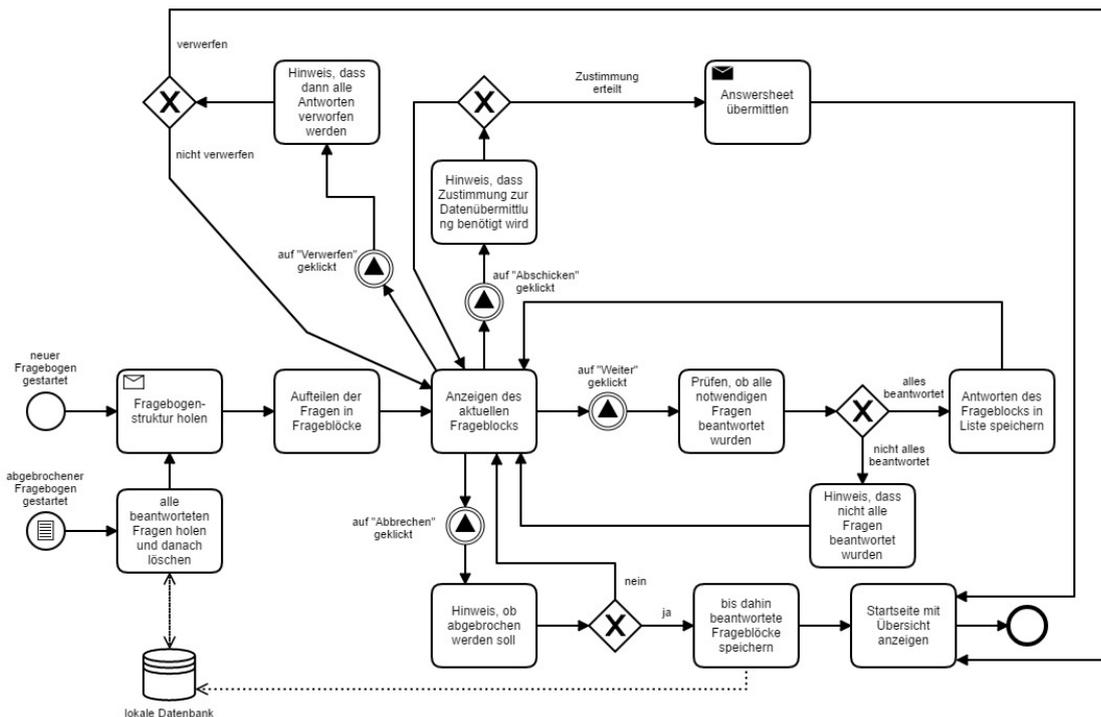


Abbildung 4.5: Prozess Ausfüllen eines Fragebogens

Eine weitere Hauptfunktion ist das Anzeigen des Feedbacks zu einem Fragebogen. In Abbildung 4.6 wird dieser Vorgang illustriert. Nachdem die Patientin auf ihrer Startseite ein Anwersheet ausgewählt hat, zu dem ein Feedback vorliegt, wird dieser von der API evaluiert. Anschließend wird das vorliegende Feedback, aufgeteilt in kritisch beantwortete Fragen, angezeigt.

Die Patientin hat zu jedem Zeitpunkt die Möglichkeit die Behandlung und somit die Teilnahme an der Studie zu beenden (siehe Abbildung 4.7). Hierzu wird zunächst eine

4 Architektur der Anwendung

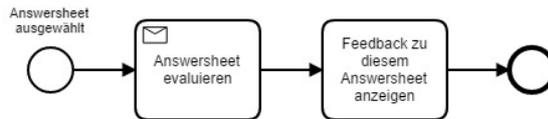


Abbildung 4.6: Prozess Einsehen eines Feedbacks

Bestätigung der Nutzerin gefordert. Sollte diese gegeben werden, wird die Patientin aus der Studie entfernt und der Startscreen der App, der den erneuten Beginn der Behandlung ermöglicht, wird angezeigt. Abgebrochene Fragebögen bleiben in der lokalen Datenbank bestehen und können somit bei einem erneuten Studienbeitritt wieder aufgegriffen werden.

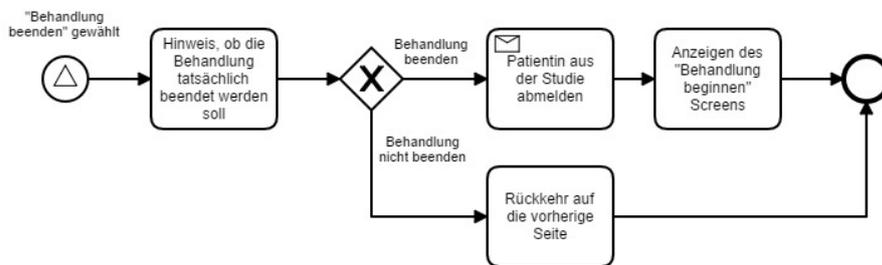


Abbildung 4.7: Prozess Beenden der Behandlung

Nach Erklärung aller essenziellen Abläufe der Patientenrolle, werden nun die Prozesse für die Rolle des Arztes betrachtet. Dazu wird zunächst der in Abbildung 4.8 gezeigte Hauptprozess eines Gynäkologen beschrieben.

Nach dem Login werden, ähnlich zum Hauptprozess der Patientin, zunächst die Profildaten und anschließend die Studien, in denen der Arzt beteiligt ist, abgerufen. Anschließend werden die in Behandlung stehenden Patientinnen dieser Studien angefordert. Nach Abrufen der neuen Answersheets aller Patientinnen, wird die Startseite mit einer Übersicht über alle Schwangeren angezeigt.

Die Detailansicht einer Patientin liefert dem Gynäkologen eine Übersicht über die Answersheets der Schwangeren. Um diese angezeigt zu bekommen, wird, wie in Abbildung 4.9 dargestellt, nach Auswahl der Patientin auf der Startseite, deren Answersheets und Details abgerufen.

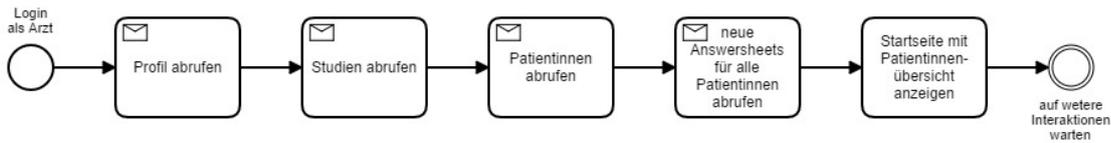


Abbildung 4.8: Hauptprozess Arzt



Abbildung 4.9: Prozess Detailansicht einer Patientin

Wie in Abbildung 4.10 erkennbar, lassen sich API-seitig evaluierte Anwersheets als Feedback anzeigen, nachdem der behandelnde Arzt einen solchen Anwersheet einer seiner Patientinnen ausgewählt hat. Anschließend wird überprüft, ob das Feedback bereits an die Patientin verschickt wurde. Ist dies nicht der Fall, hat der Arzt die Möglichkeit, durch einen Klick auf den „Feedback senden“ Button, den evaluierten Fragebogen an die Patientin weiterzuleiten. Das Anwersheet wird als gesehen und somit als versandt markiert, der Gynäkologe erhält eine Meldung, dass die Übermittlung erfolgreich war. Ab diesem Zeitpunkt kann die Patientin das Feedback einsehen.

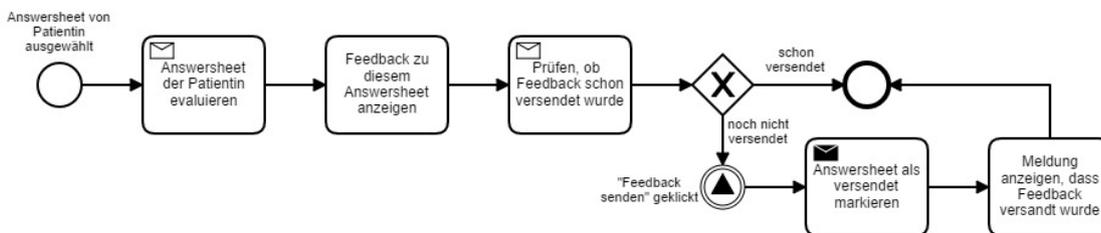


Abbildung 4.10: Prozess Einsehen und Schicken eines Feedbacks

Nachdem nun alle rollenspezifischen Prozesse erläutert wurden, folgen abschließend Abläufe, die für Arzt und Patientin identisch sind. Diese beinhalten den Prozess bei einem vergessenen Passwort, die Änderung des Passworts und der Spracheinstellung, sowie das Kontaktieren der Entwickler und den Logout.

4 Architektur der Anwendung

Hat ein Nutzer sein Passwort vergessen, so wird er vom Loginscreen zum „Passwort vergessen“ Screen weitergeleitet. Wie in Abbildung 4.11 zu sehen ist, wird nach einem Klick auf „Passwort anfordern“ geprüft, ob die eingegeben E-Mail-Adresse richtig, also im System vorhanden, ist. Trifft dies nicht zu, bekommt der Nutzer eine Fehlermeldung angezeigt. Ansonsten wird eine Anfrage an die API geschickt, die Instruktionen an die angegebene E-Mail-Adresse sendet. Anschließend wird angezeigt, dass die Passwortanforderung erfolgreich war.

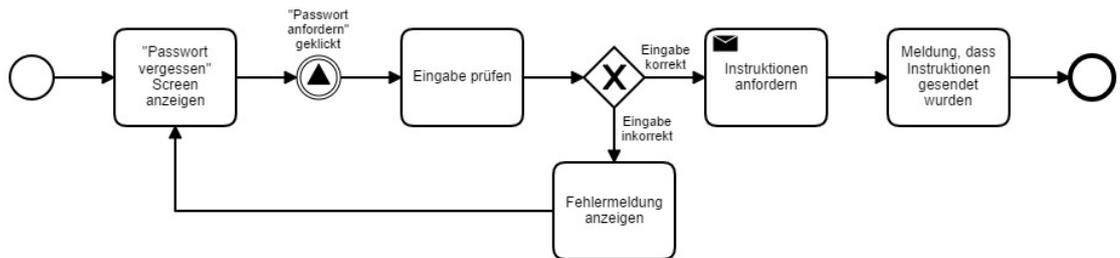


Abbildung 4.11: Prozess Passwort vergessen

Sollte der Nutzer sein Passwort ändern wollen, so kann er dies nach dem Login auf seinem Profil machen. Hierzu wird dann der in Abbildung 4.12 beschriebene Prozess ausgeführt. Das Profil wird geöffnet und angezeigt. Es ist erforderlich ein neues Passwort einzugeben und dieses zu wiederholen. Durch das Klicken auf den „Passwort ändern“ Button wird die Nutzereingabe geprüft, ist diese inkorrekt, wird eine Fehlermeldung angezeigt. Bei korrekter Eingabe, wird das Passwort geändert und ein Hinweis, dass die Änderungen übernommen wurden, erscheint.

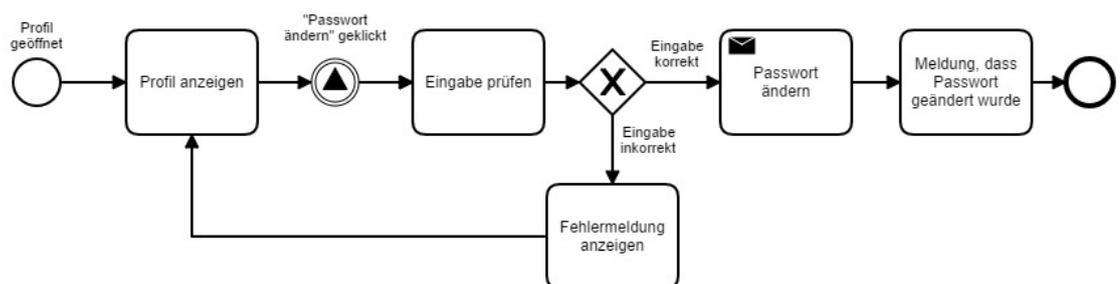


Abbildung 4.12: Prozess Änderung des Passworts

4.1 Ablaufbeschreibung

Die Abbildung 4.13 zeigt die Sprachänderung. Hierzu werden die Einstellungen angezeigt. Nach Änderung der Sprachauswahl und Klick auf den „Speichern“ Button, wird die neue Spracheinstellung an die API übermittelt und die Anzeige in der Anwendung angepasst. Anschließend bekommt der Nutzer eine Bestätigung, dass die Änderungen erfolgreich vorgenommen worden sind.

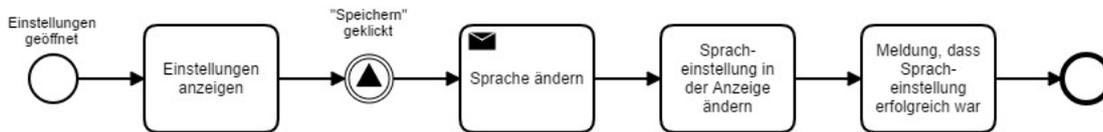


Abbildung 4.13: Prozess Änderung der Spracheinstellung

Es besteht die Möglichkeit die Entwickler zu kontaktieren, hierfür muss, wie in Abbildung 4.14 zu sehen, die entsprechende Menüoption ausgewählt werden. Anschließend wird der Kontaktierungsscreen angezeigt, in dem eine Nachricht verfasst werden kann. Durch einem Klick auf den „Entwickler kontaktieren“ Button erhält der Nutzer eine Auflistung aller auf dem Gerät verfügbaren E-Mail-Programme angezeigt. Nach der Auswahl des gewünschten Programms, wird die verfasste Nachricht dort als E-Mail geöffnet. An diesem Punkt ist der Prozess beendet und die E-Mail kann mit dem gewählten Programm versendet werden.

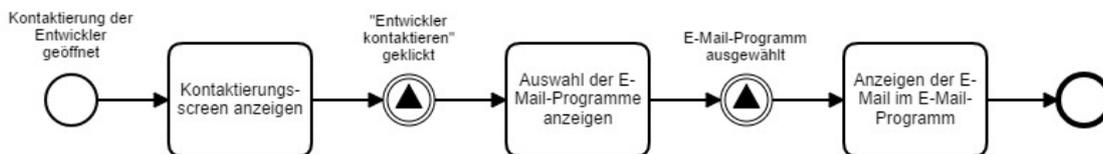


Abbildung 4.14: Prozess Kontaktierung der Entwickler

Der letzte Vorgang, der beschrieben werden soll, ist der Logoutprozess (siehe Abbildung 4.15). Hier muss im Menü die Option „Abmelden“ ausgewählt werden. Die Session des Nutzers wird daraufhin beendet und der Loginscreen wird angezeigt.

4 Architektur der Anwendung

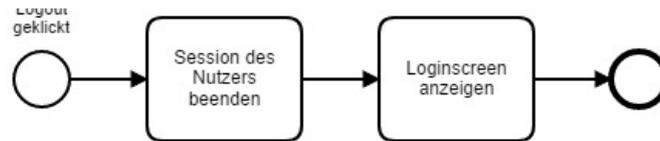


Abbildung 4.15: Logoutprozess

4.2 Architekturkonzept

Dieser Abschnitt bietet eine Übersicht über die Architektur der KINDEX Applikation. Zunächst wird hierbei der grundlegende Aufbau (siehe Abschnitt 4.2.1) beschrieben. Anschließend wird auf die Verbindung zum Backend (siehe Abschnitt 4.2.2) und die lokale Datenbank (siehe Abschnitt 4.2.3) eingegangen.

4.2.1 Grundlegender Aufbau

In Abbildung 4.16 werden alle Packages³, Activities⁴ und Fragments⁵ der KINDEX Anwendung aufgeführt. Hierbei zeigt sich, dass alle Activities von der `AppCompatActivity` erben. Zusammen mit den Fragments sind sie die Komponenten, die für die Präsentation der Benutzeroberfläche, die sogenannte View, zuständig sind. Die Klasse `KindexApplication` erzeugt eine lokale `User`- und `Studymember`-Instanz, deren Attribute, wie beispielsweise das Attribut `token`, häufig benötigt werden, sodass der Zugriff von überall möglich sein muss. Der Aufbau dieser Klassen ist in Abbildung 4.17 zu sehen. Das Package `adapter` beinhaltet die Adapter⁶ für in Activities und Fragments verwendete `RecyclerViews`⁷. Der Inhalt des `backend` Packages wird in Abschnitt 4.2.2 genauer beleuchtet. Die Erzeugung der lokalen Datenbank wird durch die Klassen des `database` Packages übernommen.

³ Werden genutzt, um zusammengehörige Klassen zu strukturieren. Für weitere Informationen siehe [10]

⁴ Container für Benutzeroberflächen, die spezielle Aktionen ausführen. Für weitere Informationen siehe [11]

⁵ Repräsentation eines Teils einer Benutzeroberfläche. Für weitere Informationen siehe [12]

⁶ Dient als Vorschrift zur Übertragung von Daten in eine Anzeige. Für weitere Informationen siehe [13]

⁷ Gruppierung von Views ähnlich zu einer Liste. Für weitere Informationen siehe [14]

4.2.2 Backend

Die Abbildung 4.17 zeigt die Struktur des `backend` Packages. Dieses wird benötigt, um mit der API kommunizieren zu können. Hierfür beinhaltet das `KINDEXService` Interface⁸ alle Anfragemethoden, die an die API gestellt werden müssen. Über den `ClientGenerator` kann ein Client für die serverseitigen Anfragen erzeugt werden. Er enthält die Grund-URL, an die die im `KINDEXService` definierten Pfade angehängt werden, sobald die zum Pfad gehörige Methode aufgerufen wird. Dieser Vorgang wird durch ein Retrofit-Objekt ausgeführt. Auf diese Bibliothek wird in Abschnitt 4.3 genauer eingegangen. Neben dem Interface und dem Generator befindet sich das `model` Package in `backend`, dessen beinhalteten Klassen modellieren die Struktur der Backenddaten. Dabei enthält es auch die in Abschnitt 4.2.1 bereits erwähnten Klassen `User` und `StudyMember`. Da das Backend sehr viele unterschiedliche Datenstrukturen aufweist, wurden die Klassen noch in die Kategorien Anfrage und Antwort untergliedert. Diese finden sich in die entsprechenden Packages `request` und `response` wieder, die in Abbildung 4.18 und 4.19 dargestellt sind.

Allgemein setzt sich hier eine sogenannte `POST`-Anfrage immer aus einer `Request<T>`- und `Data<T>`-Instanz zusammen. Das Generic `T`⁹ lässt sich durch beliebige Instanzen ersetzen und stellt dann das Attribut `attributes` der `Data<T>`-Instanz dar. Beim Einloggen eines Nutzers wird aus den eingegebenen Daten ein `LoginReq`-Objekt erzeugt, welches für das Generic `T` eingesetzt wird. Das so entstandene Konstrukt wird mittels GSON¹⁰ zu einem JSON-Objekt¹¹ konvertiert und als Retrofit-Anfrage an die API übermittelt. Dieser Ablauf ist für alle Klassen des `request`-Packages gleich. `Answersheet` dient zur Übermittlung von vollständig ausgefüllten Fragebögen, `Subscribe` zum Einschreiben einer Patientin in eine Studie, `UpdatePassword` und `ResetPassword` aktualisieren das Nutzerpasswort bzw. fordern Anweisung zu Änderung an.

Nachdem die Verwendung des `request`-Packages genauer beleuchtet wurde, soll nun auf das `response`-Package eingegangen werden, das in Abbildung 4.19 dargestellt

⁸ Gruppierung verwandter Methoden. Für weitere Informationen siehe [15]

⁹ Platzhalter für einen nicht festgelegten Datentyp. Für weitere Informationen siehe [16]

¹⁰ Java Bibliothek zur Konvertierung von Java-Objekten in JSON-Objekte und umgekehrt. Für weitere Informationen siehe [17]

¹¹ JavaScript Object Notation (Datenformat). Für weitere Informationen siehe [18]

4 Architektur der Anwendung

ist. Aus den beinhalteten Klassen werden `GET`-Anfragen an das Backend erstellt. Die Antwort, welche die API liefert, wird dann mittels GSON von einem JSON-Objekt zu einem Konstrukt von Klassen aus dem `response`-Package. Es gibt zwei verschiedene Formate, in denen die Antwort vorliegen kann: Als einfache `Response<T>` oder als `ListResponse<T>`. Der Unterschied liegt im `data`-Attribut, was entweder ein `Data<T>`-Objekt ist, oder eine Liste von diesen. Wie zuvor für die `request`-Klassen beschrieben wurde, wird auch hier das Generic `T` durch beliebige Instanzen ersetzt. Nach der weiter oben beschriebenen Loginanfrage würde die API eine Antwort der Form `Response<LoginRes>` liefern.

4.2.3 Datenbank

Die lokale Datenbank der KINDEX Anwendung ist eine SQLite¹² Datenbank, bestehend aus zwei Tabellen, wie in Abbildung 4.20 dargestellt. Diese wird benötigt, um abgebrochene Fragebögen eines Nutzers zu speichern. Dabei wird bei Abbruch ein neuer Eintrag mit der ID der Patientin in die Tabelle `information` abgelegt, um auch bei mehreren Benutzern an einem Gerät einen Antwortbogen richtig zuordnen zu können. Die anderen Spalten dienen zur Präsentation der Abbruchinformationen auf der Startseite der Patientin. Anschließend werden die beantworteten Fragen in der Tabelle `answersheets` gespeichert. Hierbei wird für die Spalte `answersheetID` die autoinkrementelle ID des zuvor angelegten `information`-Eintrags verwendet, um eine Fremdschlüsselverknüpfung zwischen den beiden Tabellen herzustellen.

¹² Bibliothek für Datenbanksysteme. Für weitere Informationen siehe [19]

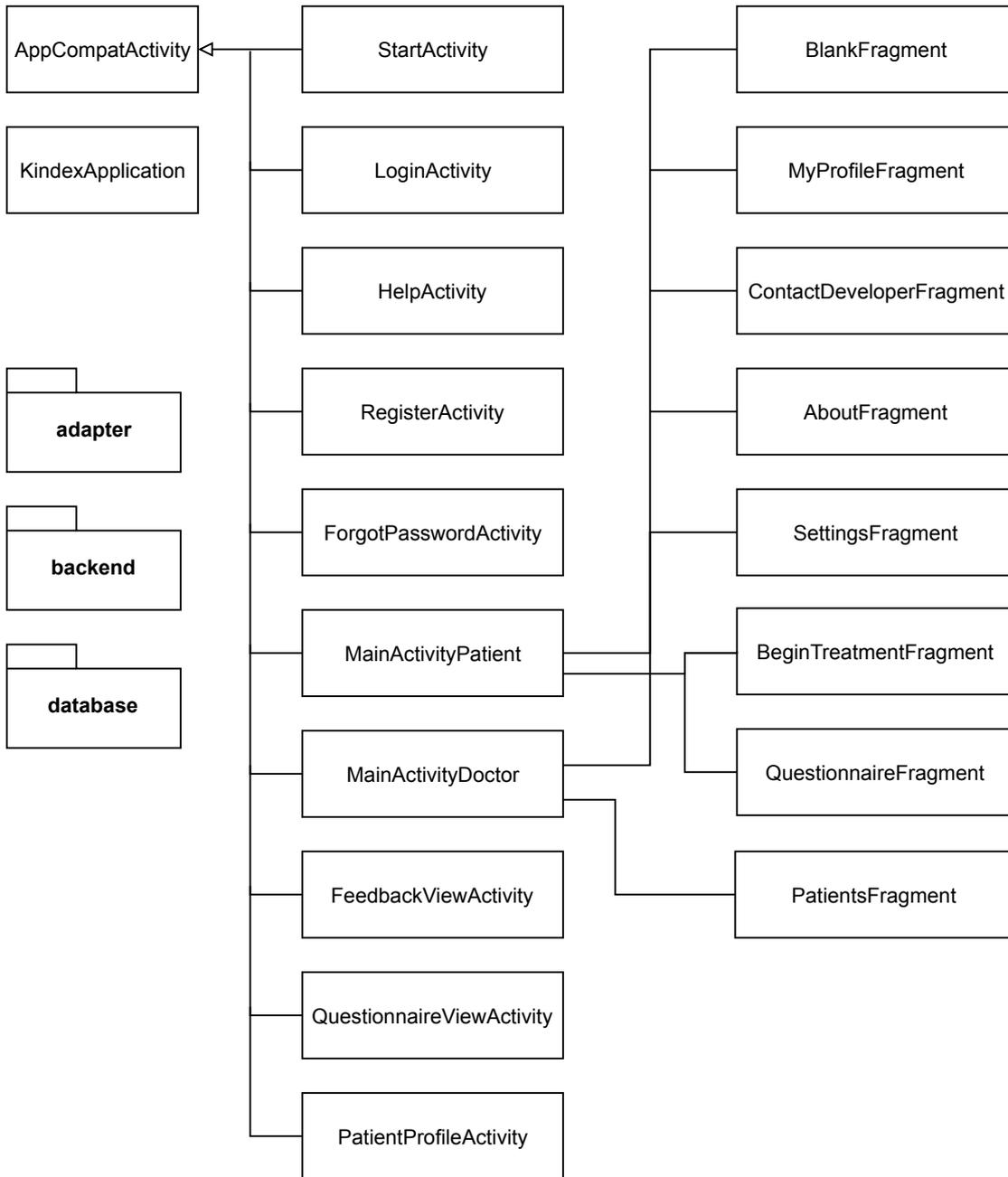


Abbildung 4.16: Klassendiagramm

4 Architektur der Anwendung

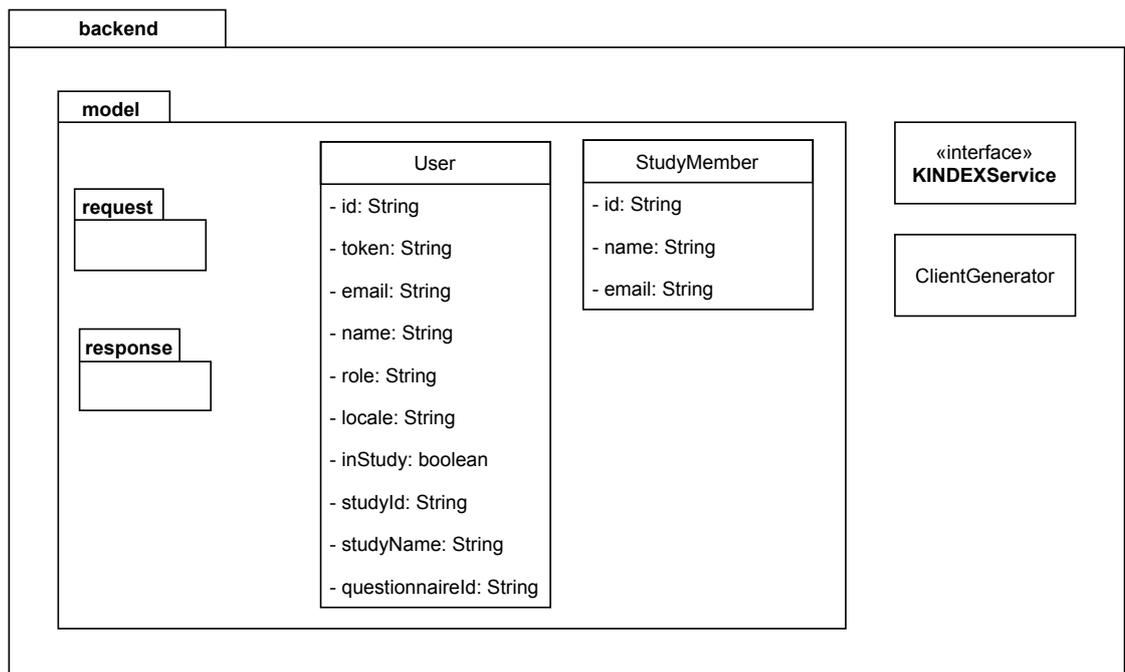


Abbildung 4.17: Backend Package

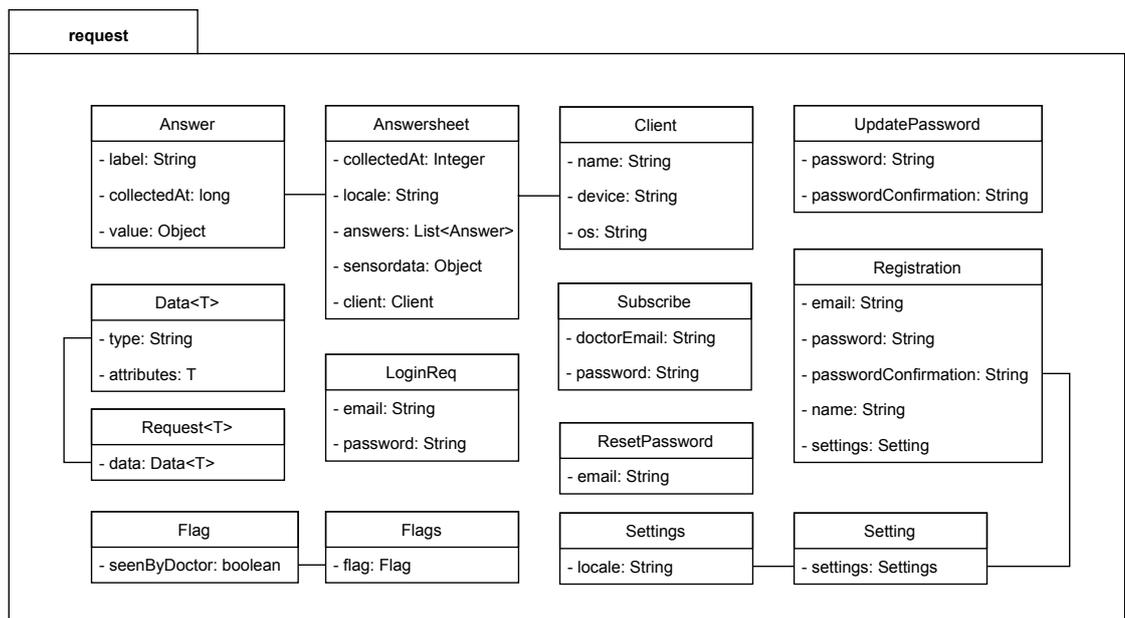


Abbildung 4.18: Request Package

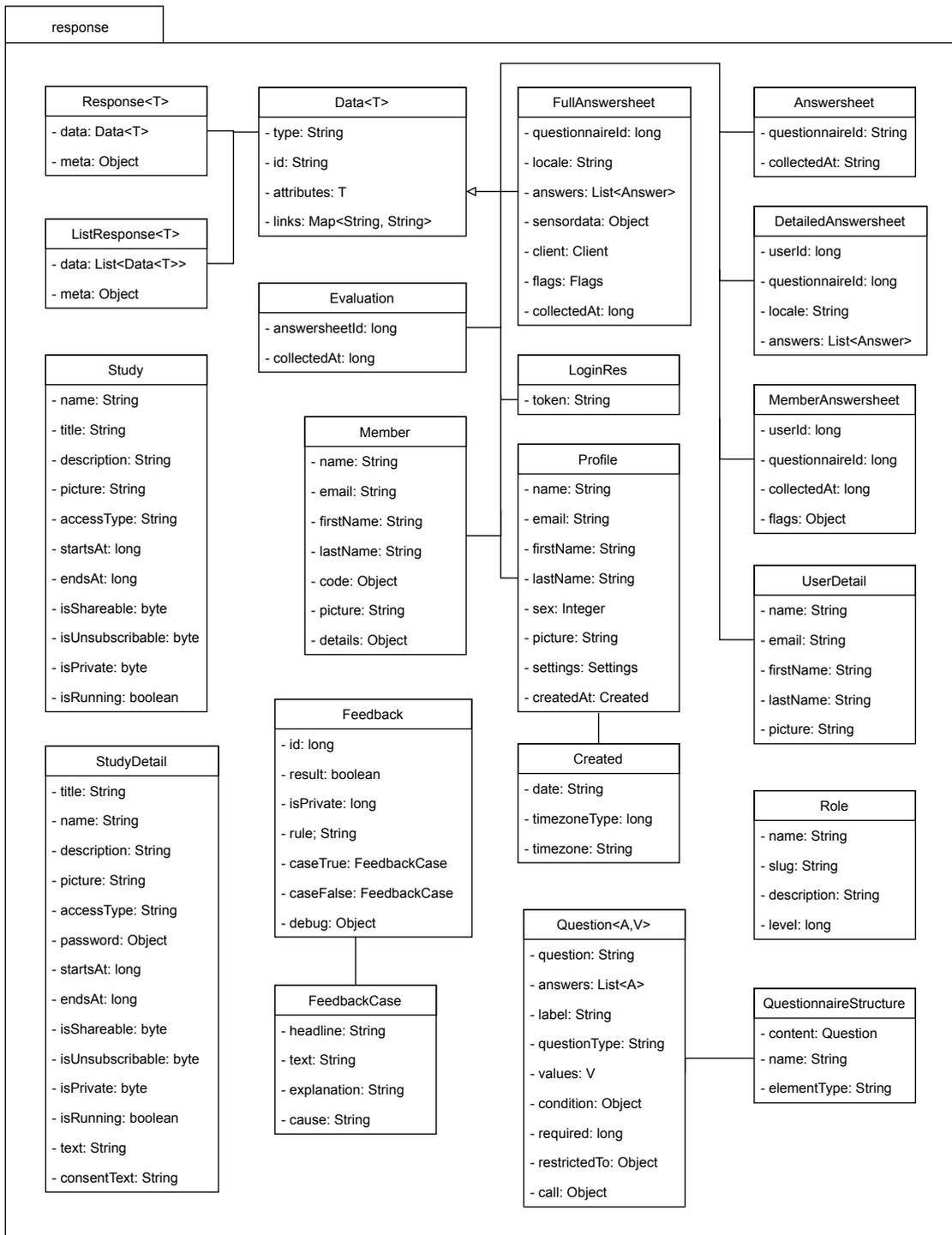


Abbildung 4.19: Response Package

4 Architektur der Anwendung

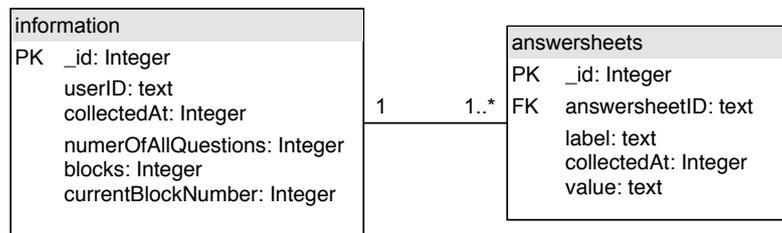


Abbildung 4.20: Lokale Datenbank

4.3 Verwendete Bibliothek - Retrofit

In diesem Abschnitt wird die Bibliothek beleuchtet, die bei der Implementierung der KINDEX Applikation verwendet wurde, um die Programmierung zu unterstützen.

Bei `Retrofit` [20] handelt es sich um einen HTTP-Client für Java-Programmierung in Android. Hierbei wird eine bestehende API in ein Java `Interface` umgewandelt, indem die HTTP-Anfragen, ein sogenannter `Call`, als Methoden hinterlegt werden. Der Aufbau einer solchen Methode wird anhand des nachfolgenden Beispiels erläutert.

```
1 @GET("users/{user}/list")
2 Call<List<User>> listUsers(@Path("user") String user, @Query("token") String
   token);
```

Listing 4.1: GET-Anfrage mit Retrofit

Die Annotation in Zeile 1 legt zunächst den Typ der HTTP-Anfrage fest, hier handelt es sich um eine GET-Anfrage, sowie die relative URL, die eine Route der API beschreibt. Bei den geschweiften Klammern handelt es sich um einen Block in der URL, der dynamisch ersetzt werden kann. Diese Ersetzung wird in Zeile 2 durch `@Path` vorgenommen. In `Call` ist der erwartete Rückgabedatentyp enthalten und anschließend folgt der Methodename, mit dem man die HTTP-Anfrage an die API aufrufen kann. Die Parameter dieser Methode können mittels `@Query` zu der Anfrage hinzugefügt werden. Bei einer POST-Anfrage müsste als Parameter ein Objekt, mittels `@Body`, an den Anfragenkörper angehängt werden.

5

Ausgewählte Implementierungsdetails

Um einen Einblick in die Implementierung der Anwendung zu geben, werden in diesem Kapitel einige Codeausschnitte vorgestellt und genauer erläutert. Da es sich bei dem Ausfüllen eines Fragebogens durch die Patientin um eine der Hauptfunktionalitäten der Applikation handelt, wurde ein Augenmerk auf die Repräsentation einzelner Frageblöcke gelegt (siehe Abschnitt 5.1). Des Weiteren wird die Überprüfung auf erforderliche Fragen innerhalb eines Frageblocks in Abschnitt 5.2 und das Abschicken eines Fragebogens in Abschnitt 5.3 schematisch dargestellt.

5.1 Fragebogenblock anzeigen

Dieser Abschnitt dient der groben Beschreibung der Darstellung eines Fragebogenblocks in der Anwendung. Um den Fragebogen anzuzeigen, muss zunächst eine Anfrage bezüglich dessen Struktur erfolgen. Wurde die Fragebogenstruktur von der API angefragt, so wird eine Liste von `Data<QuestionnaireStructure>` zurückgegeben. Wie in Abbildung 4.17 zu sehen war, hat ein `QuestionnaireStructure`-Element einen `elementType`. Hier unterscheidet man zwischen Fragen oder `PAGE`-Elementen, diese dienen zur Abtrennung zweier Frageböcken. Da immer nur ein Frageblock auf einmal sichtbar sein soll, dient die Methode `prepareSplit` zur Vorbereitung einer Unterteilung der erhaltenen Liste in einen Frageblock. Hierzu wird zunächst in Zeile 2 von Listing 5.1 ein Zähler initialisiert. Dieser repräsentiert die Position, an der später eine Unterteilung der Strukturliste (`structureList`) vorgenommen werden soll. Anschließend wird in Zeile 3-7 über die Liste iteriert und geprüft, ob es sich um ein Frage handelt. Ist dies der Fall, so wird der Zähler erhöht, da diese Frage noch Teil des darzustellenden Blocks ist.

5 Ausgewählte Implementierungsdetails

Ansonsten wird die Iteration beendet, weil nun das Ende des aktuellen Blocks erreicht wurde. Anschließend wird sofort die Methode `splitForQuestionBlocks`, die in Listing 5.2 aufgeführt ist, aufgerufen.

```
1 public void prepareSplit() {
2     Integer i = 0;
3     for(Data<QuestionnaireStructure> structure : structureList) {
4         if(!structure.getAttributes().getName().equals("PAGE")) {
5             i++;
6         } else break;
7     }
8     splitForQuestionBlocks(structureList, i);
9 }
```

Listing 5.1: Methode `prepareSplit`

Diese Methode unterteilt die übergebene Liste `splitStructureList` in zwei andere Listen (Zeile 2-3): Die erste wird `questionBlock` genannt und enthält nur die Elemente für den aktuellen Frageblock, den wir anzeigen wollen. In der zweiten, die als `structureList` bezeichnet wurde, wird der Rest von `splitStructureList` gespeichert. Somit wird durch jeden Aufruf der `splitForQuestionBlocks`-Methode die `structureList` kleiner. Hat sie keine Elemente mehr, entspricht die Größe also Null, wie es in Zeile 4 überprüft wird, befinden wir uns im letzten Frageblock. Nun soll der Button, der zum Durchnavieren gedient hat eine neue Beschriftung erhalten und ein zusätzlicher Button, zum Verwerfen des Fragebogens, angezeigt werden (Zeile 5-6). Anschließend werden die Fragen und der Fragetype des Frageblocks getrennt in den Listen `questionsOnly` und `questionType` gespeichert (Zeile 10-12). Um den Block nun anzuzeigen, wird in Zeile 14 ein neuer Adapter vom Typ `AdapterQuestions` erzeugt, indem beschrieben wird, wie die Elemente einer `RecyclerView` dargestellt werden sollen. Dieser wird in Zeile 15 auf die `RecyclerView` (`questionContent`) angewandt, die zur Repräsentation der einzelnen Fragen dient. Nun sind alle Vorbereitungen getroffen und der aktuelle Frageblock wird angezeigt.

```

1 public void splitForQuestionBlocks (List<Data<QuestionnaireStructure>>
    splitStructureList, Integer position) {
2     questionBlock = splitStructureList.subList(0, position);
3     structureList = splitStructureList.subList(position + 1,
        splitStructureList.size());
4     if(structureList.size() == 0) {
5         nextBtn.setText(R.string.action_send_questionnaire);
6         rejectBtn.setVisibility(View.VISIBLE);
7     }
8     List<String> questionsOnly = new ArrayList<>();
9     List<String> questionType = new ArrayList<>();
10    for(Data<QuestionnaireStructure> structure : questionBlock) {
11        questionsOnly.add(structure.getAttributes().getContent().getQuestion());
12        questionType.add(structure.getAttributes().getContent()
            .getQuestionType());
13    }
14    questionsAdapter = new AdapterQuestions(getApplicationContext(),
        questionsOnly, questionType, questionBlock);
15    questionContent.setAdapter(questionsAdapter);
16 }

```

Listing 5.2: Methode splitForQuestionBlocks

5.2 Konsistenzprüfung der Fragen

Bevor es einer Patientin möglich ist, den nächsten Frageblock zu beantworten, ist es notwendig zu prüfen, ob alle erforderlichen Fragen beantwortet wurden. Da diese Überprüfung für die unterschiedlichen Fragetypen ähnlich verläuft, wurden nicht alle Fälle ausführlich aufgeführt. Diese sind im Code entsprechend markiert.

Die Methode `checkForRequired` dient zur Konsistenzprüfung und gibt einen `boolean` zurück, abhängig davon, ob alle erforderlichen Fragen beantwortet wurden oder nicht. Hierzu wird über alle Fragen eines Blocks iteriert, wie in Zeile 2 zu sehen ist. In den Zeilen 3-6 werden alle benötigten Informationen (`required`: Erforderlichkeit, `label`: Bezeichnung, `type`: Fragetyp) zu einer einzelnen Frage zunächst gespeichert. Anschlie-

5 Ausgewählte Implementierungsdetails

End wird geprüft, ob eine Eingabe durch die Patientin erfolgt ist. Dies wird mittels einem `switch` Statement¹ über die Fragetypen umgesetzt, der in Zeile 7 eingeleitet wird. Liegt eine Eingabe vor, so wird, wie beispielsweise in Zeile 11, die Methode `prepareAnswer` aufgerufen. Ansonsten wird, wie in Zeile 13, überprüft, ob es sich um eine erforderliche Frage handelt. Ist dies der Fall, so gibt die Methode `false` zurück. Sonst handelt es sich um eine nicht beantwortete, aber auch nicht zwingend zu beantwortende Frage, weshalb hier keine Aktion notwendig ist. Nachdem der grobe Ablauf der Überprüfung anhand von Zahleneingaben (`Freetext-NumberOnly`) erläutert wurde, soll nun das Vorgehen der Methode für die übrigen Fragetypen explizit erklärt werden.

Der Fall `Freetext` wurde bewusst nicht aufgeführt, da er ähnlich verläuft wie `Freetext-NumberOnly`.

Bei Fragen vom Typ `Slider` erfolgt die Eingabe durch Ziehen eines Punktes (`Thumb`) entlang einer Eingabeleiste (`SeekBar`² oder `Slider`) zum gewünschten Wert. Zur Überprüfung wird hierbei der Alphawert des `Thumb`s der `SeekBar` betrachtet (Zeile 20), da dieser defaultmäßig auf Null gesetzt wurde, damit er beim Anzeigen der Frage nicht sichtbar ist. Bei einer Eingabe durch die Patientin wird auch der Alphawert angepasst, er wird erhöht und der `Thumb` wird sichtbar. Ist der Wert Null, so wurde keine Eingabe vorgenommen, also muss geprüft werden, ob die Antwort erforderlich ist (Zeile 21). Ansonsten wird der Antwortwert vom Fortschritt des `Sliders` abgeleitet und damit in Zeile 26 `prepareAnswer` aufgerufen.

Bei dem Fragetyp `SingleChoice` wird zu Beginn, in Zeile 30, der Antwortwert auf „-“ gesetzt und anschließend überprüft, welcher der beiden `RadioButton` [23] gedrückt wurde (Zeile 31-34). Abschließend wird der Wert entsprechend angepasst. Dieser ist entweder „YES“ oder „NO“, da es sich hierbei um Ja-Nein-Fragen handelt. Wurde kein Button ausgewählt, so muss wieder die Erforderlichkeit der Frage geprüft werden (Zeile 36).

Der Fall `MultipleChoice` verhält sich ähnlich zu `SingleChoice`, nur dass eine größere Auswahl an Antwortmöglichkeiten besteht und somit mehrere `RadioButton` geprüft werden müssen. Für eine bessere Übersichtlichkeit wurde hier der Code gekürzt.

¹ Statement mit mehreren möglichen Ausführungspfaden. Für weiter Informationen siehe [21]

² Eingabeinstrument von Android. Für weitere Informationen siehe [22]

5.2 Konsistenzprüfung der Fragen

```
1 public boolean checkForRequired() {
2   for(int i = 0; i < questionContent.getChildCount(); i++) {
3     View view = questionContent.findViewByIdForAdapterPosition(i).itemView;
4     boolean required = (view.findViewById(R.id.required).getVisibility() ==
5       View.VISIBLE);
6     String label = ((TextView)
7       view.findViewById(R.id.label)).getText().toString();
8     String type =
9       questionBlock.get(i).getAttributes().getContent().getQuestionType();
10    switch(type) {
11      case "Freetext-NumberOnly":
12        String valueFTN = ((EditText)
13          view.findViewById(R.id.value_freeTextNumber)).getText().toString();
14        if(!valueFTN.isEmpty()) {
15          prepareAnswer(label, valueFTN);
16        } else {
17          if(required) {
18            return false;
19          }
20        }
21        break;
22      case "Freetext": /* auf Details verzichtet */ break;
23      case "Slider":
24        if(((SeekBar)
25          view.findViewById(R.id.value_slider)).getThumb().getAlpha() == 0) {
26          if(required) {
27            return false;
28          }
29        } else {
30          long valueS = ((SeekBar)
31            view.findViewById(R.id.value_slider)).getProgress();
32          prepareAnswer(label, valueS);
33        }
34        break;
35      case "SingleChoice":
36        String valueSC = "-";
37        if(((RadioButton) view.findViewById(R.id.value1_single)).isChecked()) {
38          valueSC = "YES";
39        }
40    }
41  }
42}
```

5 Ausgewählte Implementierungsdetails

```
33     } else if(((RadioButton)
34         view.findViewById(R.id.value2_single)).isChecked()) {
35     } else {
36         if(required) {
37             return false;
38         }
39     }
40     prepareAnswer(label, valueSC);
41     break;
42     case "MultipleChoice": /* auf Details verzichtet */ break;
43 }
44 }
45 return true;
46 }
```

Listing 5.3: Methode checkForRequired

5.3 Fragebogen abschicken

Nachdem die Aspekte zum Anzeigen und Ausfüllen des Fragebogens betrachtet wurden, soll nun auf die Versand eingegangen werden. Beim Abschicken eines Fragebogens muss eine Liste der gegebenen Antworten in die API Anfrage angebunden werden. Diese Liste wird mit der Methode `prepareAnswer` vorbereitet, die in Listing 5.4 dargestellt ist. Hierzu wird beim Aufrufen zunächst die aktuelle Zeit abgefragt (Zeile 2) und mit dieser Angabe sowie den übergebenen Parametern, `label` und `value`, ein `Answer`-Objekt erzeugt (Zeile 3). Außerhalb der Methode gibt es die globale Liste `answerList`, zu der in Zeile 4 das neue Objekt hinzugefügt wird. Da diese Methode in Listing 5.3 für jede gegebene Antwort aufgerufen wird, wächst die Liste nach jedem Frageblock an, bis der Fragebogen vollständig ausgefüllt wurde.

```

1 public void prepareAnswer(String label, Object value) {
2     long collectedAt = System.currentTimeMillis();
3     Answer answer = new Answer(label, collectedAt, value);
4     answerList.add(answer);
5 }

```

Listing 5.4: Methode prepareAnswer

Nachdem alle Antworten gespeichert wurden, können diese nun an die API übermittelt werden. Die Methode `createSubmitAnswersRequest`, die in Listing 5.5 aufgeführt ist, erfüllt diese Aufgabe. Zunächst muss ein `Answersheet` Objekt erzeugt werden, welches folgende Parameter benötigt:

CollectedAt Die aktuelle Uhrzeit, um zu dokumentieren, wann der Fragebogen abgeschickt wurde. Hierzu wird in Zeile 2 die Systemzeit des Gerätes ausgelesen und in einen `Integer` umgewandelt.

Locale Die Sprache, in welcher der Fragebogen ausgefüllt wurde. Diese befindet sich in dem lokalen `User` Objekt und wird in Zeile 3 abgefragt.

AnswerList Eine Liste mit allen Antworten, diese wurde mit `prepareAnswer` erzeugt.

SensorData Dieser Parameter wurde hier auf `null` gesetzt, da bei diesem Fragebogen keine Sensordaten aufgezeichnet werden.

Client Hier wird das Gerät des Nutzers angegeben, dazu wird in Zeile 4 ein `Client` Objekt angelegt.

Anschließend wird in Zeile 6 ein `Data<Answersheet>` Objekt erzeugt, welches zur Konstruktion des `Request` Objekts in Zeile 7 benötigt wird. Damit wurden alle Vorbereitungen für eine API-Anfrage getroffen, die in Zeile 8 eingeleitet wird. Es wird die Methode `submitAnswers` des `KINDEXService` Interfaces als gewünschte HTTP-Anfrage festgelegt, welche in Listing 5.6 zu sehen ist und anschließend erläutert wird. Durch `enqueue` in Zeile 9, wird diese asynchron ausgeführt. `Retrofit` fordert das Überschreiben der Methoden `onResponse`, welche die Ereignisse nach einer erfolgreichen Anfrage beschreibt, und `onFailure`, welche das Verhalten bei Verbindungsproblemen beinhaltet. Wurden die Antworten an die API übermittelt, soll die Patientin auf ihre

5 Ausgewählte Implementierungsdetails

Startseite zurückgeleitet werden. Hierfür wird in der `onResponse`-Methode ein `Intent` [24] erzeugt, das den aktuellen Anwendungskontext und die Klasse der Zielactivity (hier `MainActivityPatient`) enthält (Zeile 13). Anschließend wird in Zeile 14 mit `startActivity` dieses `Intent` zum Öffnen der angegebenen `Activity` verwendet. Sollte es zu Problemen bei der Übermittlung der Antworten an die API kommen, wird die Anfrage in der `onFailure`-Methode abgebrochen (Zeile 19).

```
1 public void createSubmitAnswersRequest() {
2     Integer collectedAt = (int) (System.currentTimeMillis()/1000L);
3     String locale = KindexApplication.user.getLocale();
4     Client client = new Client("postman", "lenovo", "android");
5     Answersheet answersheet = new Answersheet(collectedAt, locale, answerList,
6         null, client);
7     Data<Answersheet> data = new Data<>("answersheets", answersheet);
8     Request request = new Request(data);
9     Call<Void> call = service.submitAnswers(KindexApplication.user.getQId(),
10        KindexApplication.user.getToken(), request);
11    call.enqueue(new Callback<Void>() {
12        @Override
13        public void onResponse(Call<Void> call, Response<Void> response) {
14            if(response.isSuccessful()){
15                Intent intent = new Intent(getApplicationContext(),
16                    MainActivityPatient.class);
17                startActivity(intent);
18            }
19        }
20    });
21    @Override
22    public void onFailure(Call<Void> call, Throwable t) {
23        call.cancel();
24    }
25 }
```

Listing 5.5: Methode `createSubmitAnswersRequest`

Da die Methode `submitAnswers` des `KINDEXService` Interfaces in Listing 5.5 verwendet wird, soll diese anhand von Listing 5.6 genauer betrachtet werden.

In Zeile 1 ist erkennbar, dass eine `POST`-Anfrage vorliegt, die in der Route `questionnaires/{qId}/answersheets` der API definiert ist. Bei `{qId}` handelt es sich um einen Platzhalter, der die Fragebogen ID fordert, welche dynamisch ergänzt werden soll. Die Methode hat keinen Rückgabewert, da sie lediglich zur Datenübermittlung dient. Die Route wird in Zeile 2 programmatisch manipuliert, indem der Platzhalter durch den geforderten String ersetzt wird und die `Query` um einen String erweitert wird, der das `Token`³ des Nutzers enthält. Dieses dient zur Authentifizierung des Nutzers. Da eine `POST`-Anfrage gegeben ist, muss noch ein Objekt, hier vom Typ `Request<Answersheet>`, an den Anfragekörper angehängt werden.

```
1 @POST("questionnaires/{qId}/answersheets")
2 Call<Void> submitAnswers(@Path("qId") String qId, @Query("token") String
    token, @Body Request<Answersheet> data);
```

Listing 5.6: Methode `submitAnswers`

³ Mittel zur Authentifizierung. Für weitere Informationen siehe [25]

6

Vorstellung der Anwendung

Um die bereits beschriebenen Anforderungen und die Nutzung der KINDEX Anwendung nachvollziehen zu können, soll im folgenden Abschnitt die App vorgestellt werden. Hierbei wurde versucht, den Ablauf der Bedienung chronologisch darzustellen. Da es sich hierbei jedoch um eine Anwendung mit Rollenkonzept handelt, konnte dies nicht immer gewährleistet werden.

6.1 Login

Beim Öffnen der Applikation wird nach dem Willkommensscreen zunächst der Loginscreen angezeigt (siehe Abbildung 6.2). Hier muss der Nutzer seine Rolle auswählen, die er zur Anmeldung haben möchte. Nach der Eingabe seiner E-Mail-Adresse und seines Passwortes kann er sich mit einem Klick auf den „Anmelden“ Button einloggen. Hierbei wird neben korrekten Zugangsdaten auch die Rolle des Nutzers überprüft und gegebenenfalls ein Hinweis angezeigt, dass die falsche Rolle ausgewählt wurde. Handelt es sich um eine Patientin, die neu bei KINDEX ist, so kann sie von diesem Screen aus zum Registrierungsscreen gelangen, der in Abbildung 6.3 dargestellt ist. Hat ein Nutzer sein Passwort vergessen, so kann er auch zu dem entsprechenden Screen weitergeleitet werden. Sollte man

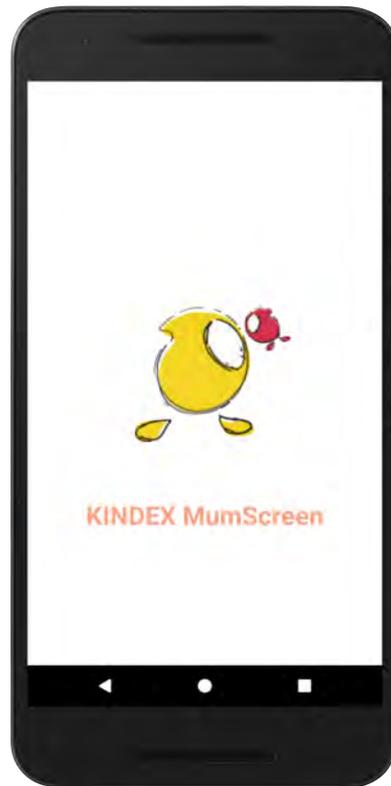


Abbildung 6.1: Willkommensscreen

6 Vorstellung der Anwendung

noch nicht mit der Nutzungsweise der Applikation vertraut sein, erhält der Nutzer eine Hilfestellung (siehe Abbildung 6.5), wenn er den Button mit dem Fragezeichen in der oberen rechten Ecke drückt.



Abbildung 6.2: Login

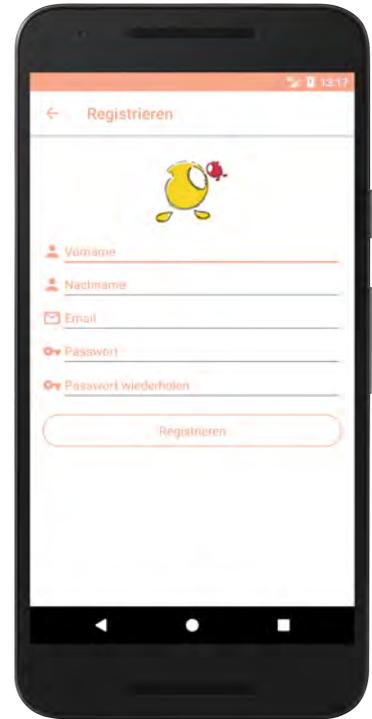


Abbildung 6.3: Registrierung

6.2 Registrierung

Um sich als neue Patientin bei der KINDEX Applikation zu registrieren, müssen Vor-, Nachname, E-Mail-Adresse sowie das gewünschte Passwort mit Wiederholung angegeben werden. Durch einen Klick auf den „Registrieren“ Button wird zunächst geprüft, ob die E-Mail-Adresse bereits vergeben ist. Sind alle Angaben vollständig und korrekt wird eine Verifizierungsmail an die angegebene E-Mail-Adresse versandt, die von der Patientin bestätigt werden muss. Erst dann kann die Nutzerin sich auf dem Loginscreen anmelden.

6.3 Passwort vergessen

Sollte der Nutzer sein Passwort vergessen haben, so kann er auf dem in Abbildung 6.4 abgebildeten Screen durch Angabe seiner E-Mail-Adresse, die mit seinem Konto bei KINDEX verknüpft sein muss, ein neues Passwort anfordern. Durch einen Klick auf den „Passwort anfordern“ Button wird eine E-Mail versandt, die Anweisungen zum weiteren Vorgehen enthält.

6.4 Hilfestellung für Patientinnen

Um sich mit der Nutzungsweise der KINDEX Applikation vertraut zu machen, ist eine Hilfestellung auf dem Loginscreen integriert. Hier werden zunächst die fünf Hauptfunktionen aufgelistet, die eine Patientin nacheinander durchführen kann. Anschließend wird zur Verdeutlichung, wo welche Funktionen möglich sind, zu jedem dieser fünf Schritte ein Screenshot der Anwendung gezeigt. Die neue Patientin hat somit einen Einblick in den Aufbau und das Aussehen der Anwendung erhalten. Sollten dennoch zu einem späteren Zeitpunkt Unklarheiten auftreten, kann jederzeit auf die Hilfestellung zurückgegriffen werden. Abbildung 6.5 stellt hierbei einen Ausschnitt der Schritt-für-Schritt-Anleitung dar.

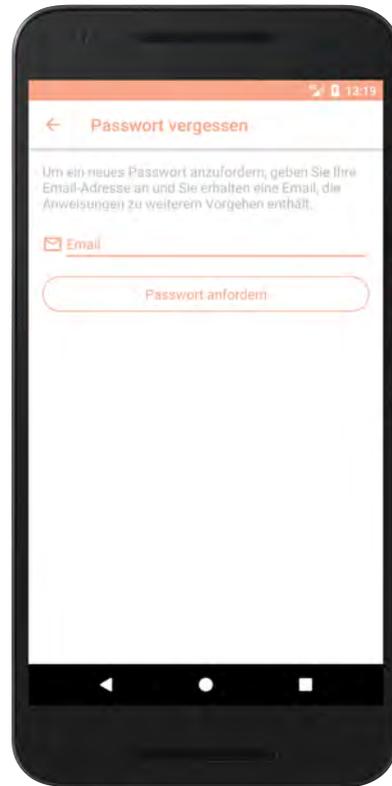


Abbildung 6.4: Passwort vergessen

6 Vorstellung der Anwendung

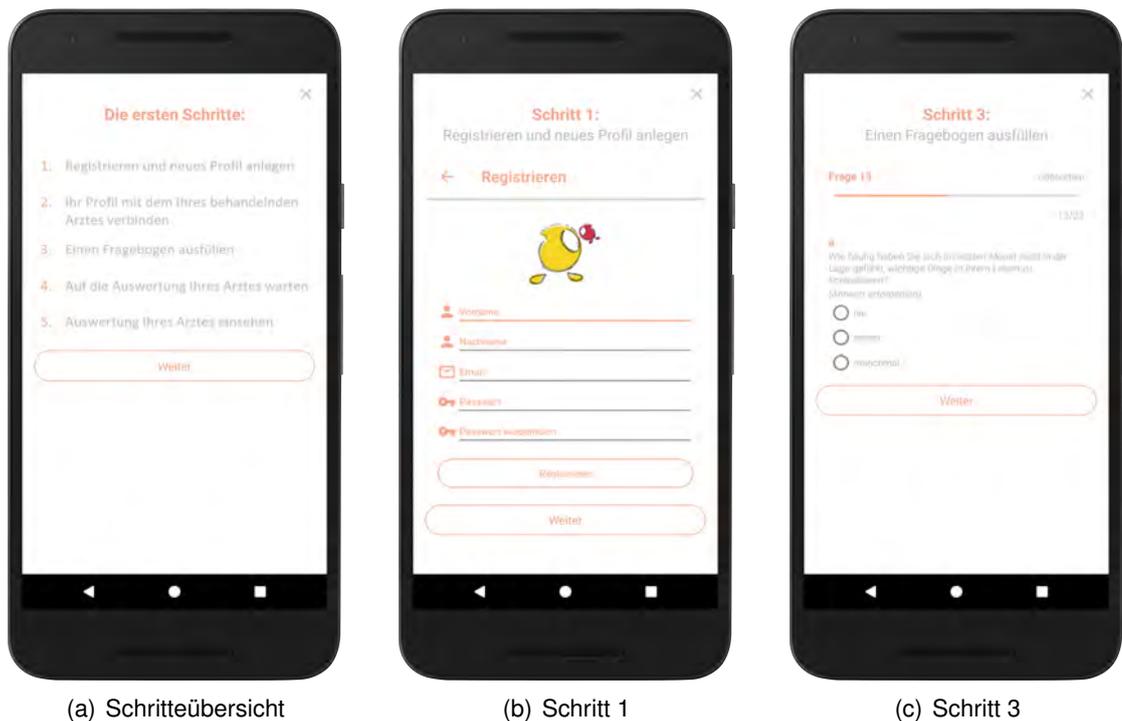


Abbildung 6.5: Hilfestellung für neue Patientinnen

6.5 Rollen-unabhängige Funktionalitäten

Die nachfolgenden Funktionalitäten sind für Nutzer beider Rollen identisch und werden daher zusammen vorgestellt.

6.5.1 Änderung des Passworts

Auf seinem Profil (Abbildung 6.6 (a)) hat der Nutzer die Möglichkeit sein aktuelles Passwort zu ändern. Hierzu muss lediglich ein neues Passwort eingegeben und wiederholt werden. Durch einen Klick auf den „Passwort ändern“ Button wird bei vollständigen und korrekten Angaben das Passwort geändert und dem Nutzer eine entsprechende Nachricht angezeigt. Des Weiteren hat er eine Übersicht über seine Angaben und seine Behandlungsstudie.

6.5.2 Kontaktierung der Entwickler

Sollte es zu Problemen mit oder Fragen zu der Anwendung kommen, bietet diese dem Nutzer die Möglichkeit, die Entwickler zu kontaktieren. Hierfür muss das Textfeld in Abbildung 6.6 (b) für Fragen/Probleme ausgefüllt werden. Nach einem Klick auf den „Entwickler kontaktieren“ Button muss der Nutzer sein E-Mail-Programm auswählen. Anschließend wird dieses geöffnet und die E-Mail wird bereits ausgefüllt mit Empfänger, Betreff und eingegebener Nachricht angezeigt. Am Ende der Nachricht wird noch der Hinweis angehängt, dass diese E-Mail aus der KINDEX Android Applikation von Patientin X (X: Platzhalter für den Namen der Patientin) gesendet wurde, um eine leichtere Verarbeitung zu ermöglichen. Diese E-Mail kann durch den Nutzer nun einfach abgesendet werden.

6.5.3 Kontaktinformationen

Die Kontaktinformationen der Applikation enthalten den Implementierungszweck, sowie die an dem Projekt beteiligten Institute und Universitäten.

6.5.4 Änderung der Spracheinstellung

Sollte der Nutzer die Sprache der Anwendung ändern wollen, so ist dies auf dem Einstellungsscreen, welcher in Abbildung 6.6 (c) gezeigt ist, möglich. Die derzeit eingestellte Sprache ist zunächst ausgewählt. Ändert der Nutzer die Angabe und drückt den „Einstellungen speichern“ Button, bekommt der Nutzer einen Hinweis für die erfolgreiche Umstellung der Einstellungen. Die Anwendung wird neu geladen, um die Sprachänderungen zu übernehmen und der Nutzer gelangt zurück zu seinem Startscreen.

6 Vorstellung der Anwendung

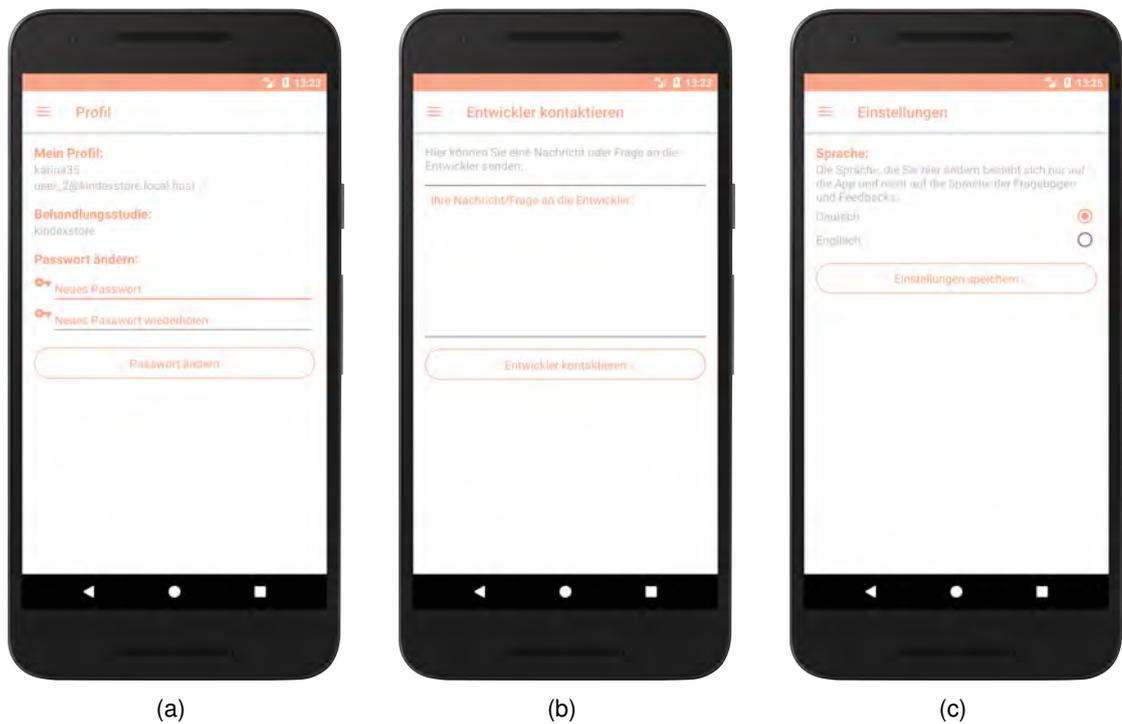


Abbildung 6.6: Rollen-unabhängige Funktionalitäten

6.6 Funktionalitäten für Patientinnen

Die nachfolgenden Funktionalitäten stehen nur Nutzern der Rolle Patientin zur Verfügung.

6.6.1 Beginn der Behandlung

Um die KINDEX Applikation nach Registrierung und Verifizierung verwenden zu können, muss eine Patientin ihr Konto mit dem ihres behandelnden Arztes verknüpfen. Nach dem Login wird zunächst der „Studie beitreten“ Screen angezeigt (Abbildung 6.7 (a)). Um die Behandlung zu beginnen sollte es zu einem Austausch der benötigten Daten, bestehend aus dem Namen der Behandlungsstudie des Arztes und einem Zugangspasswort, gekommen sein. Ohne diese Daten ist die Patientin keinem Arzt zugeordnet, weshalb keine Fragebögen ausgefüllt werden können und somit die Hauptfunktionalität der Applikation nicht genutzt werden kann. Nach einem Klick auf das Feld „Wählen Sie eine

Studie" bekommt die Patientin eine Übersicht über alle verfügbaren Behandlungsstudien (Abbildung 6.7 (b)). Hieraus kann sie die Behandlungsstudie ihres betreuenden Arztes auswählen. Nach anschließender Eingabe des Zugangspasswortes (Abbildung 6.7 (c)) und Drücken des „Beitreten“ Buttons werden die eingegebenen Daten zur Verknüpfung überprüft. Wurde alles vollständig und korrekt angegeben, wird die Patientin in die ausgewählte Behandlungsstudie eingeschrieben, auf ihre neue Startseite weitergeleitet und bei ihrem behandelnden Gynäkologen als Patientin angezeigt.

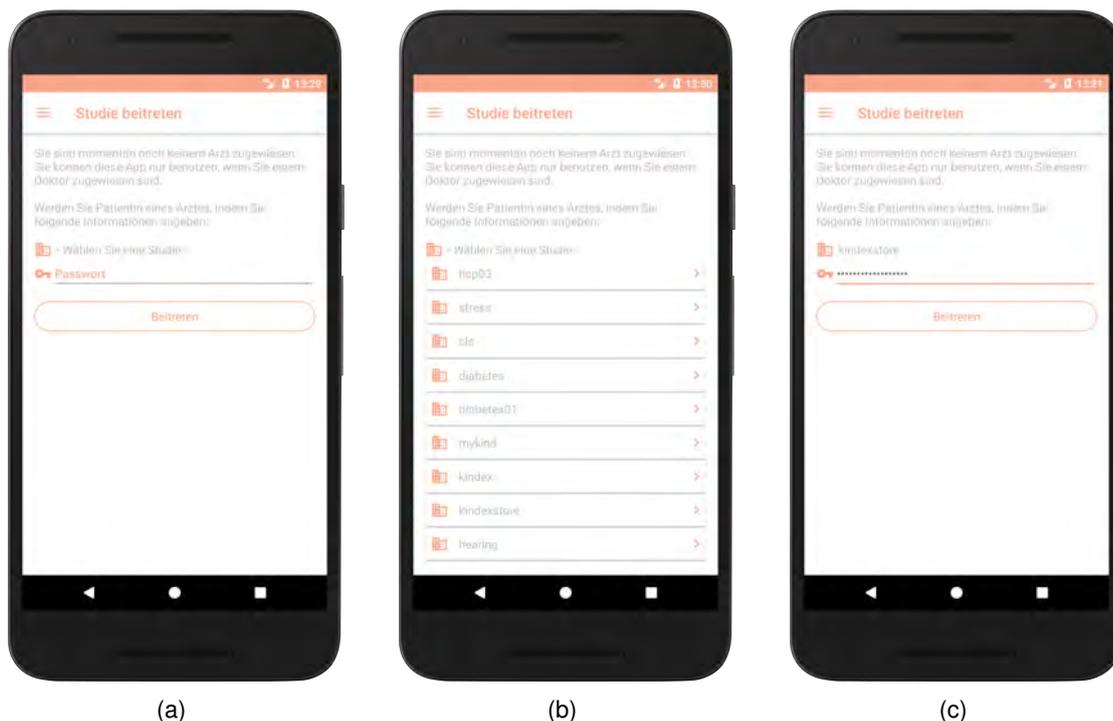


Abbildung 6.7: Behandlung beginnen

6.6.2 Startseite Patientin

Auf ihrer Startseite hat die Patientin eine Übersicht über abgebrochene und abgeschickte Fragebögen, sowie Fragebögen, zu denen sie ein Feedback erhalten hat. Zu Beginn einer Studie sind in keiner Kategorie Fragebögen vorhanden. Wie in Abbildung 6.8 zu sehen, enthalten diese Auflistungen im weiteren Anwendungsverlauf das Datum der Fertigstellung und bei abgebrochenen Fragebögen den derzeitigen Fortschritt.

6 Vorstellung der Anwendung

Wird ein Fragebogen ausgewählt, für den ein Feedback vorhanden ist, so kann dieses eingesehen werden (siehe Abschnitt 6.6.4). Bei abgeschickten Fragebögen handelt es sich um vollständig ausgefüllte und an den Arzt versandte Fragebögen. Bei Elementen dieser Kategorie ist keine weitere Interaktion möglich, bis ein Feedback vorhanden ist. Dieses muss zunächst durch den behandelnden Arzt eingesehen und zurück an die Patientin geschickt werden. Unterbrochene Fragebögen können jederzeit weiter ausgefüllt werden. Neben der Übersicht hat die Patientin auch die Möglichkeit einen neuen Fragebogen auszufüllen (siehe Abschnitt 6.6.3), indem sie auf den „Neuen Fragebogen beginnen“ Button drückt.

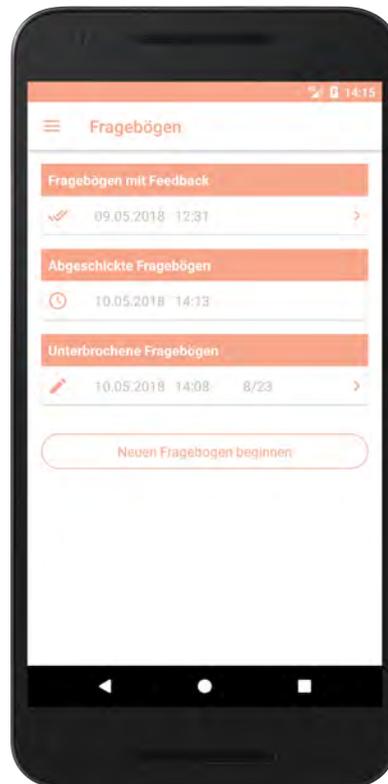


Abbildung 6.8: Startseite Patientin

6.6.3 Ausfüllen eines Fragebogens

Ein Fragebogen kann über die Startseite neu begonnen oder als Fortsetzung nach einem Abbruch gestartet werden. Der KINDEX Fragebogen besteht aus mehreren Frageblöcken, die der Reihe nach ausgefüllt werden müssen. Solche Blöcke können eine bis mehrere Fragen enthalten, welche sich zusätzlich in verpflichtende und nicht-verpflichtende Antworten unterteilen lassen. Am Ende jedes Blocks befindet sich ein „Weiter“ Button, der zum nächsten Block führt. Es ist der Patientin jedoch erst nach der Beantwortung aller verpflichtenden Fragen möglich mit dem nächsten Frageblock fortzufahren. Wird zu einem früheren Zeitpunkt versucht den nächsten Frageblock zu beginnen, erhält die Patientin einen Hinweis, dass noch Antworten benötigt werden. Zur Unterstützung der Nutzerin erhalten diese Fragen eine farblich Markierung, wie sie im Hintergrund des Hinweis-Dialogs aus Abbildung 6.9 (a) zu sehen ist. Beim Ausfüllen des Fragebogens wird der Patientin zu jedem Zeitpunkt angezeigt, bei welchem Frageblock sie sich befindet, wie groß der prozentuale Fortschritt ist, sowie die Anzahl der noch ausstehenden Frageblöcke.

Die Fragen des KINDEX Fragebogens lassen sich in unterschiedliche Fragetypen kategorisieren:

Zahleneingabe

Hier wird meist nach der Anzahl eines gewissen Sachverhalts oder zeitlichen Gegebenheiten gefragt.

Ja-Nein-Auswahl

Mit dieser Kategorie wird abgefragt, ob gewisse Sachverhalte zutreffend sind oder nicht.

Texteingabe

Diese Eingaben dienen oftmals als Erweiterung zu einer Ja-Nein-Eingabe, um diese in bestimmten Fällen genauer ausführen zu können.

Slider-Eingabe

Über diesen Eingabetyp soll es der Patientin ermöglicht werden, anzugeben wie sehr ein Sachverhalt zutrifft.

Einfache Auswahl

Ähnlich zur Slider-Eingabe dient auch die einfache Auswahl zur Angabe der Zutrefflichkeit eines Sachverhalts. Hier hat man vier vorgegebene Abstufungen.

Ein Beispiel eines Frageblockausschnitts mit einfacher Auswahl ist in Abbildung 6.9 (b) zu sehen. Es besteht die Möglichkeit einen Fragebogen abubrechen, indem man auf den „Abbrechen“ Button drückt. Dann werden die bereits beantworteten Frageblöcke lokal gespeichert und es wird ein neuer Eintrag in der Liste „Abgebrochene Fragebogen“ angezeigt. Die Patientin kann zu einem anderen Zeitpunkt mit der Beantwortung der Fragen fortfahren.

Nachdem alle Fragen beantwortet wurden wird der in Abbildung 6.9 (c) zu sehende Screen angezeigt und der Fragebogen kann an den behandelnden Arzt übermittelt oder aber auch vollständig verworfen werden. Durch das Versenden wird ein neuer Eintrag in der Liste „Abgeschickte Fragebögen“ angezeigt. Nach dem Verwerfen kann auf diesen Fragebogen nicht mehr zugegriffen werden, da er vollständig gelöscht wird.



Abbildung 6.9: Fragebogen

6.6.4 Einsehen eines Feedbacks

Gibt es zu einem Fragebogen ein Feedback, so kann die Patientin dieses einsehen. Dies ist die zweite Hauptfunktionalität der KINDEX Anwendung. Das Feedback enthält den Zeitpunkt der Fragebogenabgabe, sowie die Anzahl der, nach dem KINDEX berechneten, kritisch beantworteten Fragen. Sollte es solche Fragen geben, wie es im Beispiel aus Abbildung 6.10 der Fall ist, werden der Patientin eine Liste dieser Fragen angezeigt. Jedes Element aus der Liste ist ausklappbar und enthält dann das Feedback, bestehend aus einer Erklärung des Belastungsfaktors und Behandlungsmöglichkeiten. Manche der Maßnahmen enthalten Links zu Websites, die als zusätzliche Informationsquelle dienen sollen. Diese können von der Anwendung aus im Browser des Geräts geöffnet werden. Das Feedback ist jederzeit einsehbar und verschwindet nicht aus der Liste.



Abbildung 6.10: Feedback

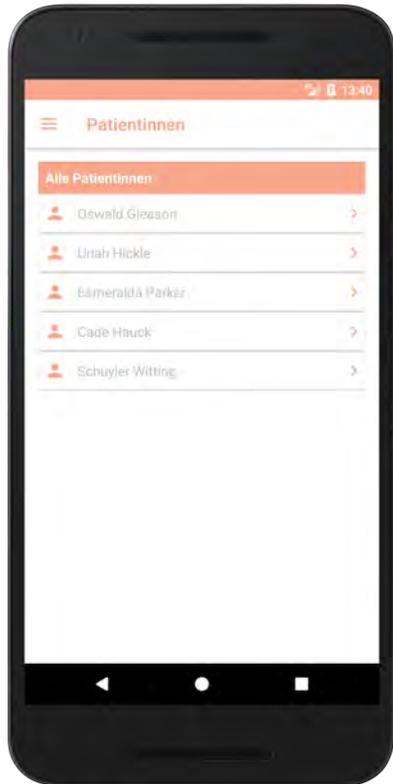
6.6.5 Beenden der Behandlung

Unterschiedliche Gründe, wie beispielsweise das Ende der Schwangerschaft oder ein Wechsel des behandelnden Arztes, können dazu führen, dass eine Behandlung von Seiten der Patientin beendet werden soll. Dies, sowie das erneute Beitreten in eine Studie, ist jederzeit möglich. Wurde die Anwendung vor Beginn der neuen Behandlung nicht deinstalliert, hat die Patientin anschließend erneut Zugriff auf lokal gespeicherte, abgebrochene Fragebögen.

6.7 Funktionalitäten für Ärzte

Die nachfolgenden Funktionalitäten stehen nur Nutzern der Rolle Arzt zur Verfügung.

6.7.1 Startseite Arzt



Die Startseite des Gynäkologen, die in Abbildung 6.11 zu sehen ist, enthält eine Übersicht über alle Patientinnen, die sich bei ihm in Behandlung befinden. Gibt es neue, noch nicht eingesehene Fragebögen zu einer Patientin, so zeigt ein Badge¹ neben dem Patientinnennamen an, um wie viele Fragebögen es sich handelt. Dies dient zum schnellen Überblick über ausstehende Feedbacks. Zum Einsehen dieser Fragebögen muss zunächst die gewünschte Patientin ausgewählt werden, um zur Detailansicht dieser Patientin zu gelangen (siehe Abschnitt 6.7.2).

Abbildung 6.11: Startseite Arzt

6.7.2 Detailansicht einer Patientin

In der Detailansicht einer Patientin (Abbildung 6.12 (a)) erhält der Arzt eine Übersicht über die übermittelten Fragebögen dieser Patientin. Diese kann selbstverständlich auch leer sein, wenn es noch keine Fragebögen gibt. Bei den verfügbaren Fragebögen einer

¹ Viewelement, welches als Hinweis für neuer Benachrichtigungen verwendet wird.

Schwangeren, werden neue Fragebögen mit einem Badge mit dem Inhalt „Neu“ gekennzeichnet. Wird ein solcher Fragebogen ausgewählt, kann das ausgewertete Feedback eingesehen und an die Patientin übermittelt werden (siehe Abschnitt 6.7.3). Fragebögen, für die bereits ein Feedback verschickt wurde, können immer noch eingesehen werden, jedoch ist das erneute Versenden des Feedbacks nicht möglich.

6.7.3 Schicken eines Feedbacks

Hat der Arzt einen neuen Fragebogen aus der Detailansicht einer Patientin ausgewählt, so kann er das ausgewertete Feedback einsehen und dieses an die Patientin übermitteln, indem er den „Feedback an Patientin schicken“ Button drückt (Abbildung 6.12 (b)). Er erhält eine Bestätigung, wie in Abbildung 6.12 (c) sichtbar, dass das Feedback versandt wurde. Der Button verschwindet, da ein Feedback für jeden Fragebogen nur einmal versendet werden kann. Um zu der Detailansicht der Patientin zurückzukehren, kann der Pfeil in der oberen linken Ecke verwendet werden.

6 Vorstellung der Anwendung



Abbildung 6.12: Detailansicht und Feedback einer Patientin

6.8 Menüs

Bei dem Menü der Anwendung handelt es sich um ein `DrawerLayout`², welches sich unterscheidet, je nachdem welche Rolle der Nutzer inne hat. Bei Patientinnen variiert das Menü auch abhängig davon, ob sie Teil einer Studie sind. Dabei ist der mittlere Teil des Menüs jeweils gleich, da es sich hierbei um Rollen-unabhängige Funktionalitäten handelt. Eine Übersicht zu den Menüs ist in Abbildung 6.13 dargestellt.

² View, die seitlich in den Screen gezogen werden kann. Für weitere Informationen siehe [26]

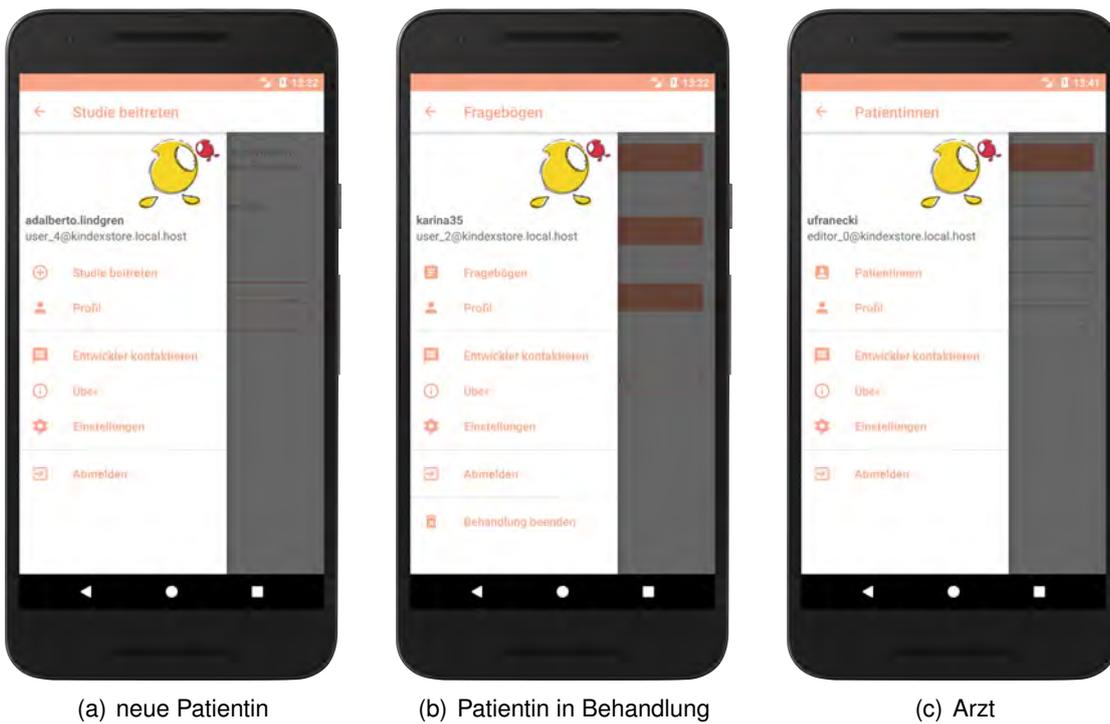


Abbildung 6.13: Menüs

6.9 Logout

Wenn der Nutzer die Anwendung beenden möchte, kann er sich im Menü mit der Auswahl „Abmelden“ von seinem Account ausloggen. Er wird dann von dem System abgemeldet und zurück auf den Loginscreen geleitet.

7

Abgleich der Anforderungen

In Kapitel 3 wurden sowohl die funktionalen als auch die nicht-funktionalen Anforderungen an die Applikation beschrieben. Nach Vorstellung der Anwendung soll nun geprüft werden, ob diese Anforderungen erfüllt wurden.

7.1 Nicht-funktionale Anforderungen

| Nr. | Anforderung und Beschreibung |
|-----|---|
| 1 | Betriebssystem <i>Anforderung erfüllt</i> Die KINDEX Applikation ist eine Android Anwendung. |
| 2 | Android-Versionen <i>Anforderung erfüllt</i> Das Mindest-Android-SDK Level ist 21 und entspricht somit der Android Version 5.0 |
| 3 | Verständlichkeit <i>Anforderung erfüllt</i> Die Anwendung hat nur die zur Nutzung notwendigen Elemente, damit der Fokus auf der Funktionalität liegt. Es gibt eine Hilfestellung für neue Patientinnen, um sich einen ersten Eindruck über die Verwendung der Applikation machen zu können. |
| 4 | Bedienbarkeit <i>Anforderung erfüllt</i> Durch klares und schlichtes Design ist die Struktur der Anwendung sehr verständlich und diese somit leicht bedienbar. |

Tabelle 7.1: Abgleich nicht-funktionale Anforderungen

7.2 Funktionale Anforderungen

| Nr. | Anforderung und Beschreibung |
|-----------------------------------|---|
| <i>Allgemein</i> | |
| 1 | Registrierung <i>Anforderung erfüllt</i> Es ist einer neuen Patientin möglich, sich direkt über die Anwendung zu registrieren und somit ein Benutzerkonto anzulegen, Kapitel 6.2. |
| 2 | Verifizierung der E-Mail-Adresse <i>Anforderung teilweise erfüllt</i> Es wird eine E-Mail an die angegebene E-Mailadresse gesendet. |
| 3 | Passwort vergessen <i>Anforderung erfüllt</i> siehe Kapitel 6.3 |
| 4 | Rollenkonzept <i>Anforderung erfüllt</i> Das Rollenkonzept wurde umgesetzt, wie in Kapitel 6.6 und 6.7 beschrieben wurde. |
| 5 | Nutzung nur mit Internetverbindung <i>Anforderung erfüllt</i> Die Anwendung ist nur mit bestehender Internetverbindung nutzbar. |
| <i>Allgemein - nach Einloggen</i> | |
| 6 | Passworrücksetzung <i>Anforderung erfüllt</i> Ein Nutzer kann sein Passwort auf seinem Profil ändern, Kapitel 6.5.1. |
| 7 | Entwickler kontaktieren <i>Anforderung erfüllt</i> Nach Eingabe des gewünschten Textes, wird ein E-Mailprogramm des Geräts geöffnet, Kapitel 6.5.2. |
| 8 | Kontaktinformationen <i>Anforderung erfüllt</i> Kapitel 6.5.2 |

| | |
|------------------------|---|
| 9 | Änderung der Spracheinstellung <i>Anforderung erfüllt</i> Der Nutzer hat die Möglichkeit die Sprache der Anwendung auf Deutsch oder Englisch einzustellen, wie in Kapitel 6.5.4 beschrieben. |
| <i>Rolle Patientin</i> | |
| 10 | Hilfestellung zur Anwendung <i>Anforderung erfüllt</i> Auf dem Loginscreen gibt es eine Schritt-für-Schritt-Anleitung für die Anwendung durch eine Patientin, Kapitel 6.4. |
| 11 | Login als Patientin <i>Anforderung erfüllt</i> Das Einloggen als Patientin ist nach einer Registrierung und Bestätigung der Verifizierungsmail möglich. |
| 12 | Behandlungsbeginn: Kontoverknüpfung mit dem behandelnden Arzt <i>Anforderung erfüllt</i> Die Kontoverknüpfung wird durch Auswahl und Passwordeingabe zu einer Studie durchgeführt, Kapitel 6.6.1. |
| 13 | Beenden der Behandlung <i>Anforderung erfüllt</i> Die Patientin kann jederzeit die Behandlung beenden und somit aus der Studie austreten, Kapitel 6.6.5. |
| 14 | Übersicht der Fragebögen und des Feedbacks auf der Startseite <i>Anforderung erfüllt</i> Nach dem Login werden der Patientin alle bearbeiteten Fragebögen angezeigt, Kapitel 6.6.2. |
| 15 | Ausfüllen eines neuen Fragebogens <i>Anforderung erfüllt</i> Das Ausfüllen eines Fragebogens ist zu jeder Zeit möglich, Kapitel 6.6.3. |
| 16 | Feste Abfolge der Fragen <i>Anforderung erfüllt</i> Es kann immer nur der aktuelle Frageblock beantwortet und somit eingesehen werden, Kapitel 6.6.3. |
| 17 | Anzeige des Fragebogenfortschritts <i>Anforderung erfüllt</i> Es gibt zwei Anzeigen, um den Fragebogenfortschritt darzustellen, Kapitel 6.6.3. |

7 Abgleich der Anforderungen

| | |
|-------------------|---|
| 18 | Abbruch des Fragebogens <i>Anforderung erfüllt</i> Das Abbrechen eines Fragebogens ist zu jedem Zeitpunkt möglich, Kapitel 6.6.3. |
| 19 | Verwerfen eines Fragebogens <i>Anforderung erfüllt</i> Es besteht die Möglichkeit einen Fragebogen zu verwerfen, nachdem alle Fragen beantwortet wurden, Kapitel 6.6.3. |
| 20 | Anzeige des Feedbackstatus eines Fragebogens <i>Anforderung erfüllt</i> Der Feedbackstatus eines Fragebogens wird durch eine Aufteilung in die Listen „Abgeschickte Fragebögen“ und „Fragebögen mit Feedback“ übersichtlich dargestellt, Kapitel 6.6.2. |
| 21 | Einsehen eines Feedbacks <i>Anforderung erfüllt</i> Die Patientin kann das Feedback zu Fragebögen der Liste „Fragebögen mit Feedback“ einsehen, Kapitel 6.6.4. |
| <i>Rolle Arzt</i> | |
| 22 | Login als Arzt <i>Anforderung erfüllt</i> Ein Arzt kann sich einloggen, nachdem ihm ein Nutzerkonto von den zuständigen Universitäten angelegt wurde. |
| 23 | Patientenübersicht <i>Anforderung erfüllt</i> Direkt auf der Startseite hat der Arzt eine Übersicht über die Patientinnen, die sich in seiner Behandlung befinden, Kapitel 6.7.1. |
| 24 | Detailansicht einer Patientin <i>Anforderung erfüllt</i> Zu jeder Patientin kann eine Liste der vorhandenen Fragebögen angezeigt werden, Kapitel 6.7.2. |
| 25 | Detailansicht eines Fragebogens <i>Anforderung erfüllt</i> Nach Auswahl eines Fragebogens einer Patientin kann das zugehörige Feedback eingesehen werden, Kapitel 6.7.2. |
| 26 | Schicken eines Feedback <i>Anforderung erfüllt</i> Das Feedback zu jedem Fragebogen kann einmal an die Patientin geschickt werden, Kapitel 6.7.3. |

Tabelle 7.2: Abgleich funktionale Anforderungen

8

Fazit

8.1 Zusammenfassung

Diese Arbeit hat zum Ziel eine mobile Android Anwendung zu konzipieren und zu realisieren, welche, basierend auf dem KINDEX Fragebogen, eine feedbackgestützte Kommunikation zwischen behandelndem Arzt und seiner schwangeren Patientin ermöglicht.

Hierzu wurden zu Beginn die (nicht-) funktionalen Anforderungen an die Anwendung aufgelistet. Mit dieser Grundlage folgte die Konzeption der Anwendungsarchitektur, der Abläufe innerhalb der Applikation, sowie die Auflistung verwendeter Bibliotheken. Daraufhin wurden Details der Implementierung einer Hauptfunktionalität der Anwendung genauer beleuchtet. Im Anschluss erfolgte eine Vorstellung der Applikation, die, zur Orientierungserleichterung, durch Screenshots gestützt wurde. Als nächstes wurden die Anforderungen auf ihre Erfüllung überprüft, wobei es zu keinen Abweichungen zwischen dem Soll- und Ist-Zustand kam.

Insgesamt entstand bei diesem Projekt eine vollständig lauffähige Android Applikation, welche die vorgegebenen Ansprüche in Gänze erfüllt. Somit kann die Testphase der Applikation eingeleitet werden.

8.2 Ausblick

Da das Projekt zukünftig von Gynäkologen und deren Patientinnen verwendet werden soll, wäre als nächster Schritt die testweise Nutzung im Rahmen einer Pilotstudie sinnvoll. Daraus könnten Schlüsse für weitere Anforderungen oder Anpassungen der Anwendung gezogen werden und mögliche Fehlerfälle aufgezeigt werden.

Nach eingehender Reflexion über das Projekt wären folgende Verbesserungen möglich:

- Bei der Anzeige des Feedbacks wäre es zudem hilfreich für die schwangere Nutzerin eine Beurteilung der Relevanz einer Antwort angezeigt zu bekommen.
- Derzeit liegt der Fragebogen sowie das Feedback auf dem Backend nur in Deutsch und Englisch vor. Um eine größere Nutzergruppe unterstützen zu können, wäre eine Erweiterung der Sprachoptionen von Vorteil.
- Eine zusätzliche Erweiterungsmöglichkeit wäre ein Offline-Modus, damit die Anwendung auch ohne Internetzugriff genutzt werden kann.

Anschließend könnte, nach Einbezug von Änderungen, die Applikation im Google Play-Store veröffentlicht werden, um die Nutzung deutschlandweit zugänglich zu machen.

Literaturverzeichnis

- [1] Weidner, K., Bittner, A., Pirling, S., Galle, M., Junge-Hoffmeister, J., Einsle, F., Stöbel-Richter, Y.: Was hält schwangere gesund? protektive faktoren für postpartale depression. Zeitschrift für Psychosomatische Medizin und Psychotherapie **59** (2013) 391–407
- [2] Malich, L.: Die Gefühle der Schwangeren: Eine Geschichte somatischer Emotionalität (1780-2010). Volume 99. transcript Verlag (2017)
- [3] Weisman, C.S., Hillemeier, M.M., Chase, G.A., Dyer, A.M., Baker, S.A., Feinberg, M., Downs, D.S., Parrott, R.L., Cecil, H.K., Botti, J.J., et al.: Preconceptional health: risks of adverse pregnancy outcomes by reproductive life stage in the central pennsylvania women's health study (cepawhs). Women's Health Issues **16** (2006) 216–224
- [4] Krens, I., Krens, H.: Grundlagen einer vorgeburtlichen Psychologie. Vandenhoeck & Ruprecht (2005)
- [5] Wimmer-Puchinger, B.: Schwangerschaft als Krise: Psychosoziale Bedingungen von Schwangerschaftskomplikationen. Springer-Verlag (2013)
- [6] Spyridou, A., Schauer, M., Ruf-Leuschner, M.: Obstetric care providers are able to assess psychosocial risks, identify and refer high-risk pregnant women: validation of a short assessment tool—the kindex greek version. BMC pregnancy and childbirth **15** (2015) 41
- [7] Ruf-Leuschner, M., Brunnemann, N., Schauer, M., Pryss, R., Barnewitz, E., Liebrecht, M., Kratzer, W., Reichert, M., Elbert, T.: Die kindex-app-ein instrument zur erfassung und unmittelbaren auswertung von psychosozialen belastungen bei schwangeren in der täglichen praxis bei gynäkologinnen, hebammen und in frauenkliniken. Verhaltenstherapie **26** (2016) 171–181
- [8] Maniam, J., Chin, C.K., Chenapiah, K.: Mobile phone based pregnancy support system. (2007)

Literaturverzeichnis

- [9] Spyridou, A.: Healthy Mothers, Happy Children: Prenatal Assessment For Psychosocial Risk Factors Using The KINDEX In Public Health Settings In Spain, Greece And Peru. PhD thesis (2013)
- [10] Oracle: Packages. (<https://docs.oracle.com/javase/7/docs/api/java/lang/Package.html>) Eingesehen am 12.06.2018.
- [11] Google Inc.: Activity. (<https://developer.android.com/reference/android/app/Activity>) Eingesehen am 12.06.2018.
- [12] Google Inc.: Fragment. (<https://developer.android.com/guide/components/fragments>) Eingesehen am 12.06.2018.
- [13] Google Inc.: Adapter. (<https://developer.android.com/reference/android/widget/Adapter>) Eingesehen am 12.06.2018.
- [14] Google Inc.: Recyclerview. (<https://developer.android.com/reference/android/support/v7/widget/RecyclerView>) Eingesehen am 12.06.2018.
- [15] Oracle: Interface. (<https://docs.oracle.com/javase/tutorial/java/IandI/createinterface.html>) Eingesehen am 12.06.2018.
- [16] Oracle: Generics. (<https://docs.oracle.com/javase/8/docs/technotes/guides/language/generics.html>) Eingesehen am 12.06.2018.
- [17] GitHub Inc.: Gson. (<https://google.github.io/gson/apidocs/com/google/gson/Gson.html>) Eingesehen am 12.06.2018.
- [18] : Json. (<https://www.json.org/json-de.html>) Eingesehen am 12.06.2018.
- [19] : Sqlite. (<https://www.sqlite.org/index.html>) Eingesehen am 12.06.2018.
- [20] GitHub Inc.: Retrofit. (<http://square.github.io/retrofit>) Eingesehen am 05.06.2018.
- [21] Oracle: Switch statement. (<https://docs.oracle.com/javase/tutorial/java/nutsandbolts/switch.html>) Eingesehen am 12.06.2018.

- [22] Google Inc.: Seekbar. (<https://developer.android.com/reference/android/widget/SeekBar>) Eingesehen am 12.06.2018.
- [23] Google Inc.: Radiobutton. (<https://developer.android.com/reference/android/widget/RadioButton>) Eingesehen am 12.06.2018.
- [24] Google Inc.: Intent. (<https://developer.android.com/reference/android/content/Intent>) Eingesehen am 12.06.2018.
- [25] Oracle: token. (<https://docs.oracle.com/middleware/1213/adf/api-reference-model/oracle/jbo/script/debug/client/expr/Token.html>) Eingesehen am 13.06.2018.
- [26] Google Inc.: Drawerlayout. (<https://developer.android.com/reference/android/support/v4/widget/DrawerLayout>) Eingesehen am 12.06.2018.

Abbildungsverzeichnis

| | | |
|------|---|----|
| 4.1 | Registrierungsprozess | 16 |
| 4.2 | Loginprozess | 17 |
| 4.3 | Hauptprozess Patientin | 17 |
| 4.4 | Prozess Beginn der Behandlung | 18 |
| 4.5 | Prozess Ausfüllen eines Fragebogens | 19 |
| 4.6 | Prozess Einsehen eines Feedbacks | 20 |
| 4.7 | Prozess Beenden der Behandlung | 20 |
| 4.8 | Hauptprozess Arzt | 21 |
| 4.9 | Prozess Detailansicht einer Patientin | 21 |
| 4.10 | Prozess Einsehen und Schicken eines Feedbacks | 21 |
| 4.11 | Prozess Passwort vergessen | 22 |
| 4.12 | Prozess Änderung des Passworts | 22 |
| 4.13 | Prozess Änderung der Spracheinstellung | 23 |
| 4.14 | Prozess Kontaktierung der Entwickler | 23 |
| 4.15 | Logoutprozess | 24 |
| 4.16 | Klassendiagramm | 27 |
| 4.17 | Backend Package | 28 |
| 4.18 | Request Package | 28 |
| 4.19 | Response Package | 29 |
| 4.20 | Lokale Datenbank | 30 |
| 6.1 | Willkommensscreen | 43 |
| 6.2 | Login | 44 |
| 6.3 | Registrierung | 44 |
| 6.4 | Passwort vergessen | 45 |
| 6.5 | Hilfestellung für neue Patientinnen | 46 |
| 6.6 | Rollen-unabhängige Funktionalitäten | 48 |
| 6.7 | Behandlung beginnen | 49 |

Abbildungsverzeichnis

| | | |
|------|--|----|
| 6.8 | Startseite Patientin | 50 |
| 6.9 | Fragebogen | 52 |
| 6.10 | Feedback | 53 |
| 6.11 | Startseite Arzt | 54 |
| 6.12 | Detailansicht und Feedback einer Patientin | 56 |
| 6.13 | Menüs | 57 |

Tabellenverzeichnis

| | | |
|-----|--|----|
| 3.1 | Nicht-funktionale Anforderungen | 9 |
| 3.2 | Funktionale Anforderungen | 14 |
| 7.1 | Abgleich nicht-funktionale Anforderungen | 59 |
| 7.2 | Abgleich funktionale Anforderungen | 62 |

Name: Jessica Tatjana Hohn

Matrikelnummer: 868454

Erklärung

Ich erkläre, dass ich die Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet habe.

Ulm, den

Jessica Tatjana Hohn