

CS01 Task 04: Requirements Specification Mental Health Care Patient Management System (MHC-PMS)

Studiengang:

Luder Sandro
Wyss Patrick
Frutiger Joel
Argollo Andre Pitta
Polo Claudio
Kieliger Martin



Inhaltsverzeichnis

1	Vorwort	3
1.1	Änderungsnachweis	3
2	Einführung	3
3	Glossary	3
4	User requirements definition	4
4.1	User Requirements	4
4.2	Use Case 1	5
4.3	Use Case 2	6
5	System architecture	7
6	System requirements specification	8
6.1	Functional Requirements	8
6.2	Non-Functional Requirements	8
6.3	Domain Requirements	8
6.3.1	Allgemein	8
6.3.2	Rechtliches	8
7	System model	9
8	System evolution	10
9	Testing	10
10	Appendices	11
10.1	Specifications	11
11	Index	12



1 Vorwort

Dieses Dokument erläutert detailliert die gewonnenen Erkenntnisse aus dem Task03 (Design-Thinking-Prozess). Es zeigt die User Requirements zu unserer zukünftigen mobilen Applikation, welche im Weiteren nur noch als System bezeichnet wird. Die Patienten (BPS-Erkrankte) werden als User aufgeführt.

– Zielgruppe: Unsere Zielgruppe sind Patienten, welche unter der Borderline-Persönlichkeitsstörung leiden. Das Dokument richtet sich an spezifische Test-User, Interviewpartner aus dem Design-Thinking Prozess und ans Entwicklerteam.

1.1 Änderungsnachweis

Der Verlauf der Änderungen an der finalen Version des Dokuments werden hier aufgelistet:

Version	Beschreibung	Autor	Datum
1.0	Finale Version des Dokuments	Team Blue	23. März, 2017

2 Einführung

Recherchen im Internet haben uns gezeigt, dass bereits diverse Applikationen im Playstore zu dem Thema Borderline-Persönlichkeitsstörung existieren, aber keine, welche den User im Alltag unterstützen. Wir haben erkannt, dass diese Apps entweder Tests sind um festzustellen, ob jemand an BPS erkrankt ist, oder es handelt sich um Informations-Apps.

Aus Interviews mit Fachleuten aus dem Bereich Mental-Health-Care, einem BPS-Erkrankten und aus unseren Resultaten aus dem Design-Thinking-Prozess erfuhren wir, dass Borderline-Patienten sehr instabile Persönlichkeiten sind. Sie leiden oft unter Depressionen oder haben die Tendenz sich selber zu verletzen, in den schlimmsten Fällen kann dies sogar bis zu Suizid führen. Ihre Gefühle können im Minutentakt ändern, von glücklich bis zu extrem traurig, einsam und wütend.

Ziel unserer Software soll also sein, dem User in genau in solchen schwierigen Situationen zu helfen und zu unterstützen. Um dies zu erreichen sind folgende Funktionen vorgesehen, welche anschliessend noch detailliert erläutert werden:

Im Sinne eines Tagebuchs kann der User mit Angabe von Zeit und Datum, seine Gefühle niederschreiben. Er kann hinzufügen warum er sich so fühlt, was er erlebt hat und er kann erläutern was dagegen gemacht wurde. Dies hängt direkt mit einer weiteren System Funktion zusammen. In Form von Übungen und/oder beruhigender Musik soll das System dem User helfen, sich in stressigen Situationen zu beruhigen. Im äußersten Notfall kann der User einen Notfall Button betätigen, womit er anschließend telefonisch Hilfe erhält.

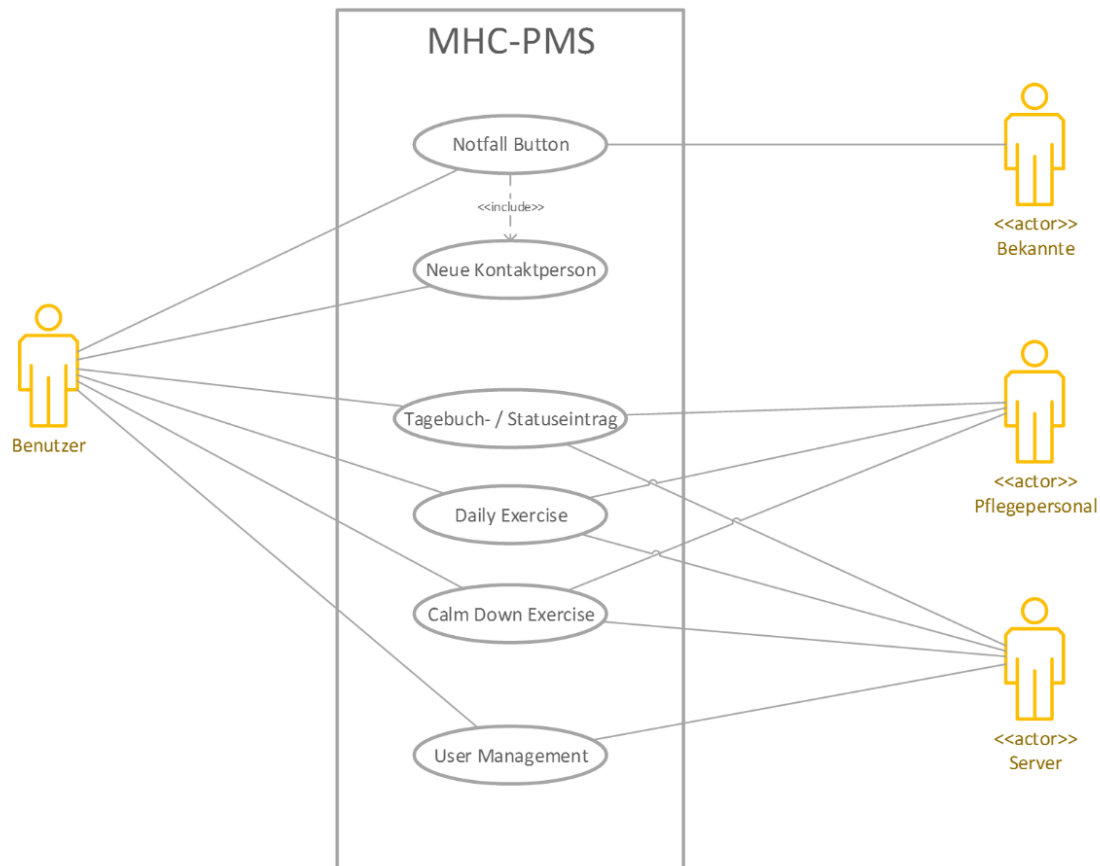
Da das System sich direkt nur an BPS-Erkrankte richtet, sollte es von keinem anderem System abhängig sein. Die Daten der User werden auf einem eigenständigen Webserver gespeichert, wobei der Datenschutz beachtet werden muss.

3 Glossary

GUI	Das graphical user interface ist die graphische Oberfläche zum Bedienen einer Applikation
HTTPS	Das HyperText Transfer Protocol Secure ist ein verschlüsseltes Kommunikationsprotokoll
SHA-512	Secure Hash Algorithm ist eine standardisierte kryptographische Hashfunktion
SSL/TLS	Secure Sockets Layer / Transport Layer Security sind Netzwerkprotokolle zur sicheren Übertragung von Daten

4 User requirements definition

4.1 User Requirements



User Management (Login/Logout/Neuer Account etc.)

Der Benutzer muss in der Lage sein sich in die Applikation einzuloggen und Berechtigungen zu erteilen.

Neue Kontaktperson erfassen

Der Benutzer soll Kontaktpersonen erfassen können, welche z.B. beim betätigen des Notfall Buttons kontaktiert werden.

Notfall Button betätigen

Beim betätigen des Notfall Buttons soll eine vordefinierte Person (je nach Gemütszustand) kontaktiert werden.

Tagebuch- / Statuseintrag erfassen

Ein Tagebuch in welches der Patient seinen aktuellen Gemütszustand eintragen kann, sowie dessen Ursprung.

Daily Exercise auswählen und bestätigen

Der Patient kann tägliche Übungen auswählen und ausführen, welche ihm helfen körperlich sowie Emotional fit zu bleiben.

Calm Down Exercise

Übungen welche dem Patienten helfen seinen aktuellen Gemütszustand wieder zu verbessern und seinen Stresslevel zu senken, diese auswählen und eine Liste erstellen.

4.2 Use Case 1

Nr. and Name:	1. Tagebuch- / Statuseintrag erfassen
Short Description:	Der User erfasst einen Tagebuch und Statuseintrag auf der Webapplikation.
Actors:	User, System
Starting Event and Preconditions:	Der User entscheidet sich aufgrund eines Reminders oder Selbstinitiative einen Tagebucheintrag zu erfassen.
Result and Postconditions:	Tagebuch und Statuseintrag wurde gespeichert und wird in der Übersicht angezeigt.

Steps:

Nr.	Actor	Description
1.1.1	User	Öffnet die Webapplikation und wählt den Menüpunkt Tagebuch/Status aus.
1.1.2	System	Die Webapplikation stellt eine Übersicht der bereits erfassten Einträgen in einer Liste dar und zeigt einen Button zum Erfassen eines neuen Eintrags.
1.1.3	System	Ausnahme: Es sind noch keine Einträge vorhanden.
1.1.4	User	Wählt den Erfassen Button aus.
1.1.5	System	Applikation zeigt ein Texteingabefeld sowie Radiobuttons zur Auswahl des Status. Der neutrale Status ist bereits ausgewählt. Ein Button zur Bestätigung Eingabe wird angezeigt.
1.1.6	User	Der User schreibt den Tagebuch Eintrag, wählt den Status aus und bestätigt die Eingabe.
1.1.7	User	Ausnahme: Der User wählt keinen Status aus und schreibt keinen Tagebuch Text.
1.1.8	System	Übersicht der Tagebucheinträge/Status wird angezeigt, der neu hinzugefügte Tagebucheintrag/Status ist in der Liste ersichtlich.

Exceptions, Variants:

Nr.	Actor	Step
1.2.1		Es sind noch keine Einträge vorhanden
1.2.2	System	Folgender Dialog wird angezeigt: "Es ist noch kein Eintrag vorhanden erstelle deinen Ersten Eintrag:", dazu wird ein grosser Button zum Erstellen eines Tagebuch-/Status-Eintrags angezeigt
1.2.3		Der User schreibt wählt keinen Status aus und schreibt keinen Tagebuch Text
1.2.4	System	Ein Status ohne Tagebucheintrag ist möglich. Es wird ein neuer Eintrag erstellt mit neutralem Status

4.3 Use Case 2

Nr. and Name:	2. Calm Down Exercise
Short Description:	Der User kann anhand seines momentanen Zustands einen geeigneten "Skill" auswählen um sein Stresspegel zu senken.
Actors:	User, System
Starting Event and Preconditions:	Der User merkt, dass sein Stresslevel erhöht ist und will diesen senken.
Result and Postconditions:	Dem User werden verschiedene passende "Skills" angezeigt welche zu seinem momentanen Zustand passt um seinen Stress in den Griff zu bekommen.

Steps:

Nr.	Actor	Description
2.1.1	User	Öffnet die Webapplikation und wählt den Menüpunkt "Calm Down Exercises" aus
2.1.2	System	Das System ermöglicht dem User die Angaben bezüglich seinem aktuellen Stresslevels
2.1.3	User	Der User gibt seinen momentanen Zustand im System an
2.1.4	System	Das System gibt dem Benutzer mögliche "Skills" zur Auswahl Hier kann der User zwischen den gewohnten und neuen Exercises auswählen um seine Skill-Liste aktuell zu halten.
2.1.5	System	Ausnahme: Noch keine Skill-Liste vorhanden
2.1.6	System	Das System merkt sich wie Oft welcher Skill ausgewählt wurde, um möglichst effiziente Vorschläge anzubieten
2.1.7	User	Der Benutzer sieht die verschiedenen Skills

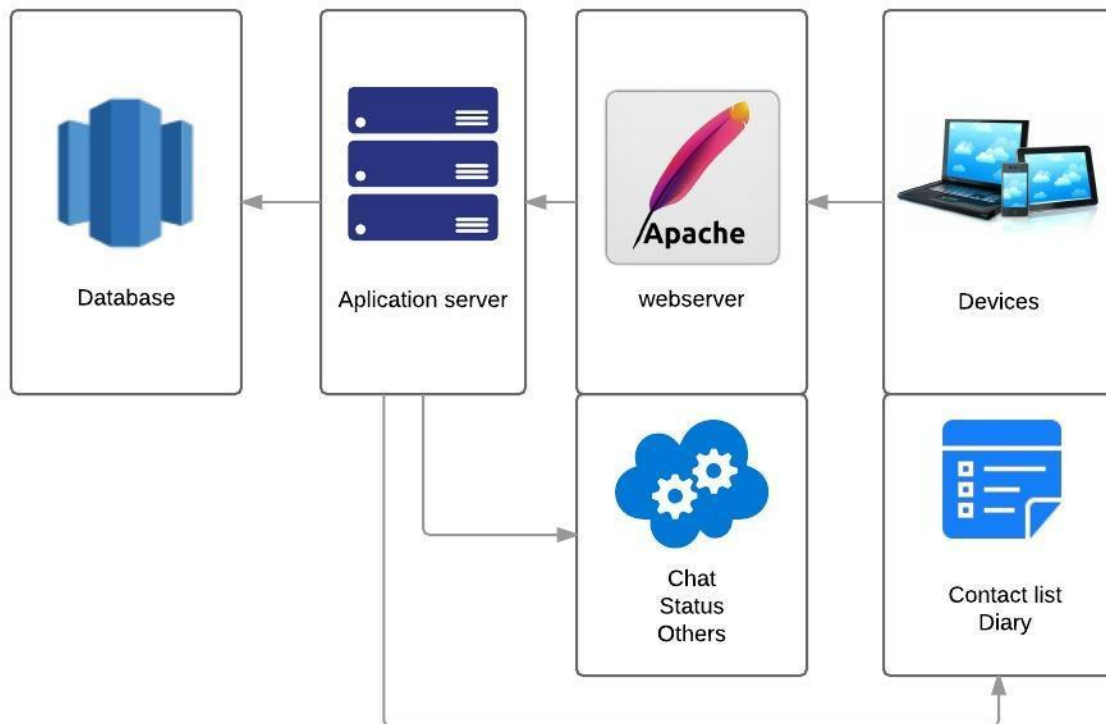
Exceptions, Variants:

Nr.	Actor	Step
2.2.1		Noch keine Skill-Liste vorhanden
2.2.2	System	Das System gibt dem Benutzer mögliche "Skills" anhand seines Zustandes zur Auswahl
2.2.3	User	Der Benutzer kann seine gewünschten Skills auswählen.
2.2.4	System	Das System merkt welche Skills der Benutzer ausgewählt hat und speichert diese in seiner persönlichen Skill-Liste.

5 System architecture

Die Systemarchitektur, besteht im Wesentlichen aus den in der Abbildung ersichtlichen Einzelkomponenten.

In diesem Kapitel, es wird eine kurze Erklärung über ihre Eigenschaften gezeigt.



Database

Die Datenbank wird verwendet zur Speicherung der Informationen von unseren Usern, welche unsere Applikation benutzen. Die Informationen auf der Datenbank stehen unter hohen Sicherheitsanforderungen, da sie zur Kategorie der besonders schützenswerten Personendaten gehören.

Applicationserver

Der Applicationserver bietet eine Schicht zwischen der Datenbank und dem Web-Server. Seine Hauptfunktionen ist die Kontrolle über die Systemdienstleistungen wie Login-Authentifizierung und Zugriff auf verschiedene Informationen in der Datenbank.

Webserver

Der Webserver ist die graphische Verbindung zum User, welche direkt auf die Inputs der Applikation reagiert und unser GUI darstellt.

Devices

Der Benutzer greift auf den Service durch einen Web-Browser oder Client zu. Die benutzten Geräte sind in der Regel Laptops, Smartphones oder Tablets. Wenn der Benutzer mit einem Smartphone auf die App zugreift, werden für einige Dienste (wie den Chat) Internetzugang benötigt. Andere Funktionen wie der Help-Button werden ohne Internet-Verbindung verfügbar sein und die SMS oder den Anruf direkt über das Handynetzt tätigen.



Security

Jede Kommunikation zwischen Client und Server wird zur Sicherheit mit HTTPS (SSL/TLS) verschlüsselt. Auch die sensiblen Daten auf der Datenbank werden mit SHA-512 verschlüsselt. Die Serverinfrastruktur muss vor unerlaubten Zugriffen geschützt sein. Die Applikation kann auf Informationen des Mobilgeräts zugreifen (Telefonnummern, Kontaktangaben) sofern der Benutzer dies erlaubt.

6 System requirements specification

6.1 Functional Requirements

Responsive Design

Die Oberfläche der Applikation sollte möglichst einfach und selbsterklärend gestaltet sein. Am Rand befinden sich die verschiedenen Menüpunkte und Zentral auf dem Display die Dazugehörigen Informationen.

Datenbank

Die Applikation muss in der Lage sein mit dem Datenbanksystem zu kommunizieren. Informationen zu speichern und abzurufen, sowie im vorgegebenen GUI darzustellen.

Report/Log

Das System sollte die Unterstützung bieten das der Arzt Einsicht auf die Tagebucheinträge und/oder Exercises des Patienten haben kann, sofern dies vom Patienten erwünscht ist.

Einträge wie zum Beispiel, wann welche Übung durchgeführt wurde oder in welchem Gemütszustand sich der Patient zu einem bestimmten Zeitpunkt befand.

Report Konfiguration

Die Reports/Logs welche der Arzt einsehen kann, sollten konfigurierbar sein, so dass der Patient bestimmen kann, was eingesehen werden darf

6.2 Non-Functional Requirements

Performance

Die Applikation sollte nicht zu lange(3s) benötigen um zu starten. Genug um alle notwendigen Daten zu laden und dass es sich für den Benutzer angenehm anfühlt.

Sicherheit

Die Applikation darf unter keinen Umständen Sicherheitslücken betreffen Personendaten enthalten, damit Informationen und Identität des Patienten zu jederzeit geschützt sind.

6.3 Domain Requirements

6.3.1 Allgemein

Arztgeheimnis/Berechtigungshierarchie

Es müssen Berechtigungsstufen verwendet werden, damit die Vertraulichen Informationen des Patienten nur von den vorgesehenen Personen/Ärzten angesehen werden kann.

6.3.2 Rechtliches

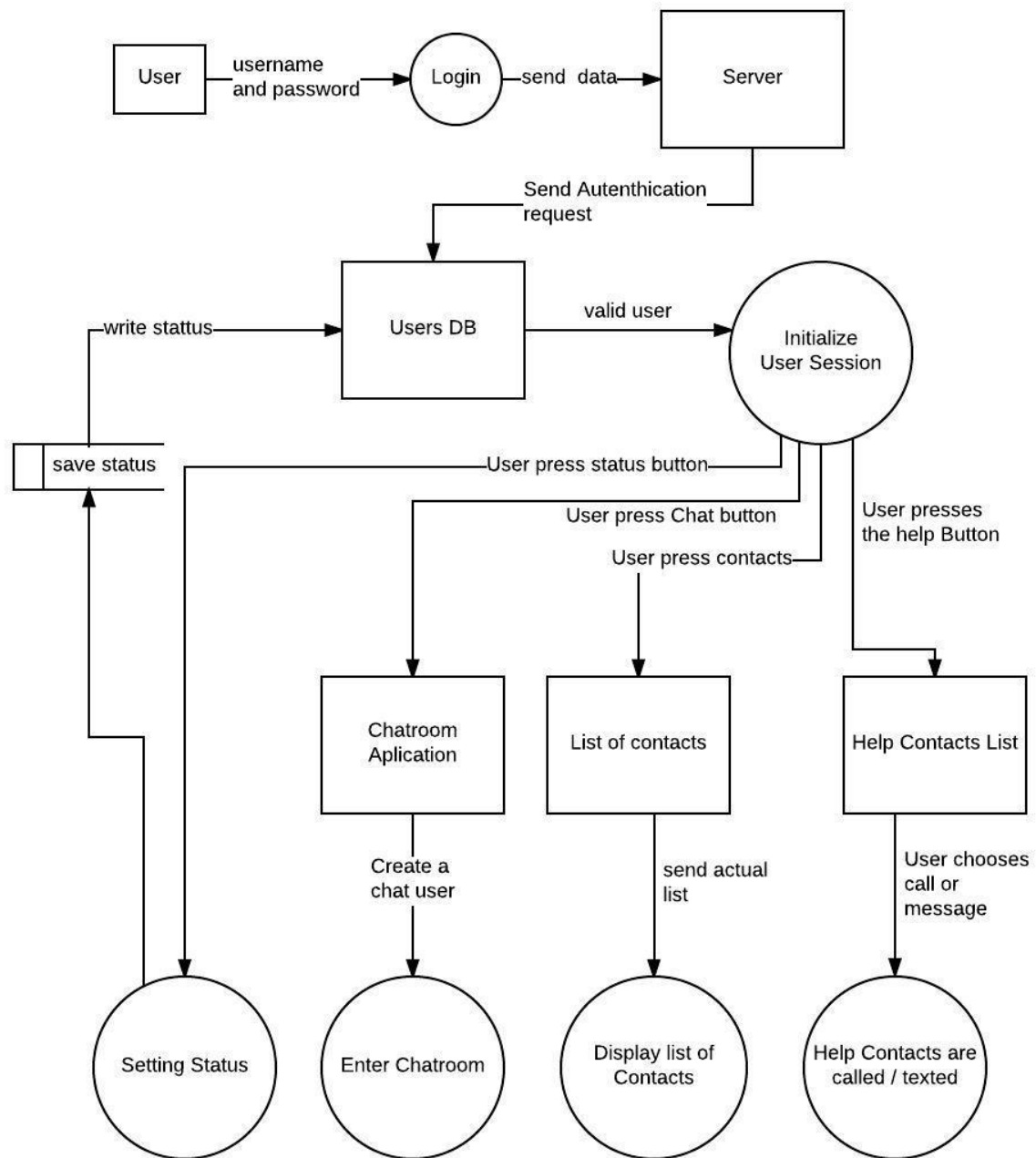
Datenschutz

Einhaltung der Datenschutzrichtlinien der Schweiz mit dem Umgang von Patientendaten.

Aufbewahrungspflicht

Je nach Art der Daten müssen diese mindestens 10–15 Jahre erhalten bleiben.

7 System model





8 System evolution

Neue Funktionen oder Erweiterungen für zukünftige Versionen:

- Gemüts Stadien für den Notfall Button.
Anhand von Feedback wäre es evtl. Sinnvoll verschiedene Stadien für den Notfallbutton zu implementieren. Welche je nach dem Zu einem Angehörigen, Arzt oder direkt den Notfall kontaktieren.
- Remotezugriff für Arzt oder Therapeut
Sofern dies vom Patienten erwünscht ist, wäre ein Remote Access auf das Diary möglich für die von ihm bestimmten Personen. Dies würde gewisse Sicherheitsvorkehrungen, betreffend dem Datenschutz des Patienten erfordern.
- Custom exercises
Gibt dem Patienten die Möglichkeit eigene Übungen welche ihm helfen ins App einzufügen und zu bearbeiten.

9 Testing

Die Software wird mittels Black-Box Konzept getestet. Dabei kennen die Tester weder den Software-Codes, noch die verwendete Infrastruktur. Die Beta-Tester sind nicht am Projekt beteiligt und haben daher keine Kenntnis über die App.

Nach der Einführung schreiben die Beta-Tester ein Protokoll, um die Fehler, die sie gefunden haben und ihre Verbesserungsvorschläge aufzuzeigen.

Das Protokoll sollte folgende Punkte beinhalten:

- Fehler bei den Funktionen
- Dateninkonsistenz
- Performanceverbesserungen
- Abstürze, Einfrieren oder Fehler beim Laden

10 Appendices

10.1 Specifications

Folgende Anforderungen müssen erfüllt sein:

- Frontend: Responsive Web Design (Mobile)
- Backend: Authentifizierung mittels Login
- Internetzugang erforderlich
- Web Server mit Datenbank
- Datensicherung und Verschlüsselung

Smartphone Anforderungen

Die Mobile App läuft auf allen gängigen Smartphones mit Android oder iOS. Keine speziellen Hardwareanforderungen.

Web Server Anforderungen beim Start des Projektes

	Minimal	Optimal
Prozessor	Single Core CPU	Quad-Core CPU
Arbeitsspeicher	4 GB	16 GB
Festplatte	50 GB	500 GB



11 Index

A

Allgemein	8
Änderungsnachweis	3
Appendices	11
Applicationserver	7
Arztgeheimnis/Berechtigungshierarchie	8
Aufbewahrungspflicht	8

D

Database	7
Datenbank	8
Datenschutz	8
Devices	7
Domain Requirements	8

E

Einführung	3
------------	---

F

Functional Requirements	8
-------------------------	---

G

Glossary	3
----------	---

I

Inhaltsverzeichnis	2
--------------------	---

N

Non-Functional Requirements	8
-----------------------------	---

P

Performance	8
--------------------	---

R

Rechtliches	8
Report Konfiguration	8
Report/Log	8
Responsive Design	8

S

Sicherheit	8
Specifications	11
System architecture	7
System evolution	10
System model	9
System requirements specification	8

T

Testing	10
---------	----

U

Use Case	5, 6
User Requirements	4
User requirements definition	4

V

Vorwort	3
---------	---

W

Webserver	7
------------------	---