

Software Engineering and Design - Team Red

Software Requirements Specification Document

Task 2: Requirements

Thomas Baumann
Ismael Riedo

Frédéric Lehmann
Fridolin Zurlinden
Tobias Weissert

Severin Thalmann
Roland Roccaro

April 15, 2018

Contents

1	Preface	3
1.1	Leserschaft	3
1.2	Version	3
1.3	Versionsbegründungen	3
1.4	Versionsänderungen	3
2	Introduction	4
2.1	Kurzbeschreibung der Funktionen des Systems	4
2.2	Wie wird es mit anderen Systemen zusammen arbeiten	4
3	Glossary	5
4	User requirements definition	6
4.1	Use Cases Diagramm	6
4.2	Use Cases	7
4.2.1	Use Case 1	7
4.3	Ablauf	7
4.3.1	Use Case 2	7
5	System architecture	9
5.1	Übersicht der System Architektur	9
6	System requirements specification	10
6.1	Data Flow	10
6.2	Schnittstellen	10
7	System models	11
7.1	Modell einer Datenbank	11
7.2	Modell des Data-Flow	12
8	System evolution	13
9	Testing	14
9.1	Was wird getestet?	14
9.2	Wie wird es getestet?	14
10	Appendices	15
10.1	Hardware requirements	15
10.2	Database requirements	15
11	Index	16

1 Preface

1.1 Leserschaft

Dieses Dokument richtet sich an die Kunden des Team Red, Kurs Software Engineering and Design Frühlingssemester 2018. Es dient als Grundbeschreibung der Applikation welche durch das Team entwickelt wird.

1.2 Version

Table 1.1: Version

Versionsnummer	Autor	Beschreibung
1.0	Roccaro	Grundfassung, Bassierend auf Design Thinking

1.3 Versionsbegründungen

Table 1.2: Versionsbegründungen

Versionsnummer	Autor	Beschreibung
1.0	Roccaro	Grundfassung des Spezifikationsdokuments, Fassung zur Ansicht der Spezifikation durch Kunden.

1.4 Versionsänderungen

Table 1.3: Versionsänderungen

Versionsnummer	Autor	Beschreibung
1.0	Roccaro	Grundfassung des Spezifikationsdokuments, Fassung zur Ansicht der Spezifikation durch Kunden.

2 Introduction

Das System wird benötigt im Umgang mit Angststörungen bzw. Sozialen . Aktuell existieren keine Lösungen für Patienten in ambulanter Behandlung die an Angststörungen leiden. Unsere Applikation wird den Patienten bei der Genesung unterstützen und im mehr Lebensqualität geben. Auch soll es dem Patienten ermöglichen durch weniger Angst wieder ein produktives Mitglied der Gesellschaft zu werden.

2.1 Kurzbeschreibung der Funktionen des Systems

Unsere Applikation bietet über einen Anreiz sich jeden Tag schrittweise an ein Ziel anzunähern. Die Applikation bietet die Möglichkeit, das der Patient zusammen mit dem Therapeuten erstellt. Diese sind Aufgaben welche der Patient anschliessend gestellt bekommt. Nach der Ausführung einer Challenge wird der Patient nach einer Beschreibung des Erlebnisses gefragt. Diese Bechreibungen sind für den Therapeuten zugänglich, später während den Therapiesitzungen können sie besprochen werden.

2.2 Wie wird es mit anderen Systemen zusammen arbeiten

Die Applikation wird selbst laufen bzw. keinen Zugang zu anderen Systemen benötigen. Es müssen keine Schnittstellen zu anderen Systemen für den Datenabgleich vorgesehen werden.

Die Applikation komuniziert nicht mit anderen Systemen. Die Verbindungen welche projektiert werden, sind Emails oder Push Nachrichten an den Therapeuten, welche von unserer Applikation ausgelöst werden.

3 Glossary

The latex typesetting markup language is specially suitable for documentation.

latex Is a mark up language specially suited for scientific documents. 5

4 User requirements definition

4.1 Use Cases Diagramm

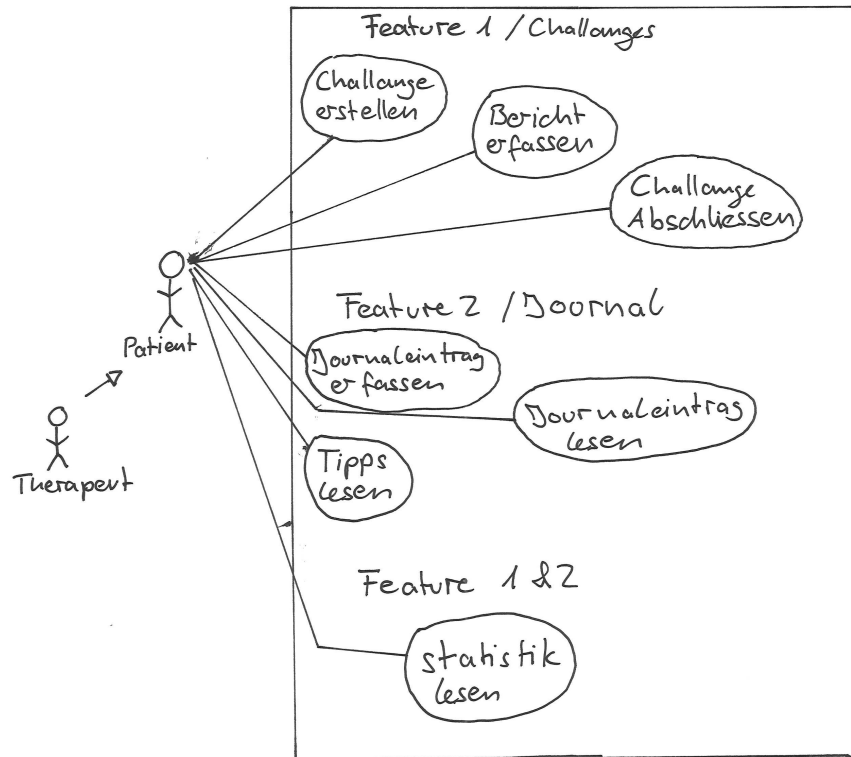


Figure 4.1: Use Case Diagramm Feature 1

4.2 Use Cases

4.2.1 Use Case 1

Table 4.1: Use Case 1

Bezeichnung	Beschreibung
Name	Challange erstellen
Nummer	1
Kurzbeschreibung	Der Patient erfasst eine neue Challange.
Beteiligte Akteure	Benutzer
Auslöser / Vorbedingung	keine
Ergebnisse / Nachbedingung	Die Challange erscheint im richtigen Level mit dem richtigen Status. Sie wird für die Statistik berücksichtigt.

4.3 Ablauf

Table 4.2: Version

Nr.	Akteur	Aktion
1.0	Benutzer	Der Patient betätigt den Hinzufügen Buttton.
1.1	Benutzer	Über die Erstellungsmaske werden die Details für die Challange abgefüllt. Danach wird die Challange gespeichert.
1.2	System	Überprüfung der eingegebenen Daten.
1.3	System	Die Challange wird dem dazugehörigen Level angefügt und auf dem UI angezeigt.

4.3.1 Use Case 2

Table 4.3: Use Case 2

Bezeichnung	Beschreibung
Name	Neuer Journaleintrag
Nummer	2
Kurzbeschreibung	Der Patient erfasst einen neuen Journaleintrag.
Beteiligte Akteure	Benutzer
Auslöser / Vorbedingung	keine
Ergebnisse / Nachbedingung	Der Journal eintrag wird erstellt und ist in der Übersicht der Journaleinträge ersichtlich. Der neue Journaleintrag wird für die Statistik berücksichtigt.

Ablauf

Table 4.4: Version

Nr.	Akteur	Aktion
1.0	Benutzer	Der Benutzer wählt den Hinzufügen Button.
1.1	Benutzer	Über die Erstellungsmaske werden die Details für den Journaleintrag eingefügt. Danach wird er gespeichert.
1.2	System	Überprüfung der eingegebenen Daten.
1.3	System	Der Eintrag wird den bestehenden Einträgen angefügt und im UI dargestellt.

5 System architecture

Bei der Applikation handelt es sich um eine Webapplikation, sie soll also funktionieren. Einzige Systemanforderung an den Client ist ein moderner Browser. Durch den steigenden Trend an mobilen Geräten soll die Applikation sowohl auf Smartphones wie auch Tablets funktionieren, dies geschieht durch ein und einer Entwicklung. Die Business Logik der Applikation soll in Java geschrieben werden, da sich das Team bereits ein breites Wissen in dieser Technologie erarbeitet hat. Genauere Spezifikationen an das Backend und die Datenbank werden im Kapitel Appendices spezifiziert.

5.1 Übersicht der System Architektur

Die Webapplikation wird mit umgesetzt. Dabei wird ein hoher Wert auf die Simplität auf Clientseite gesetzt. Sämtliche Daten werden auf dem Webserver abgelegt bzw. dessen verbundenen Datenbank. Auf der Clientsite werden keine Installationen durchgeführt.

Alle Daten werden ausschliesslich von der Applikation verwendet und nicht Drittanbietern zur Verfügung gestellt. Die Applikation geht mit den Personendaten voll um.

Der Server selbst wird in der Cloud positioniert, die Backups und Absicherung gegen Angriffe werden durch den Hoster des Servers durchgeführt.

6 System requirements specification

6.1 Data Flow

Da es sich um eine Webapplikation handelt werden keine Daten auf dem Gerät gespeichert, sondern werden jeweils direkt vom Server gelesen und geschrieben. Dadurch haben alle Geräte jeweils den aktuellen Stand der Daten. Beim speichern von neuen Daten werden Diese direkt auf Ihre Korrektheit überprüft und der Benutzer im Fehlerfall informiert.

6.2 Schnittstellen

Aktuell sind keine Schnittstellen in Fremdsysteme vorgesehen, die Applikation wird sowohl von Patienten wie auch vom Arzt standalone eingesetzt. Es kann aber nicht ausgeschlossen werden, dass eine solche Integration in Zukunft von einem Arzt gewünscht wird. Sodass er direkt in seinem Klinik Informationssystem den Status aller Challanges seiner Patienten prüfen kann.

7 System models

7.1 Modell einer Datenbank

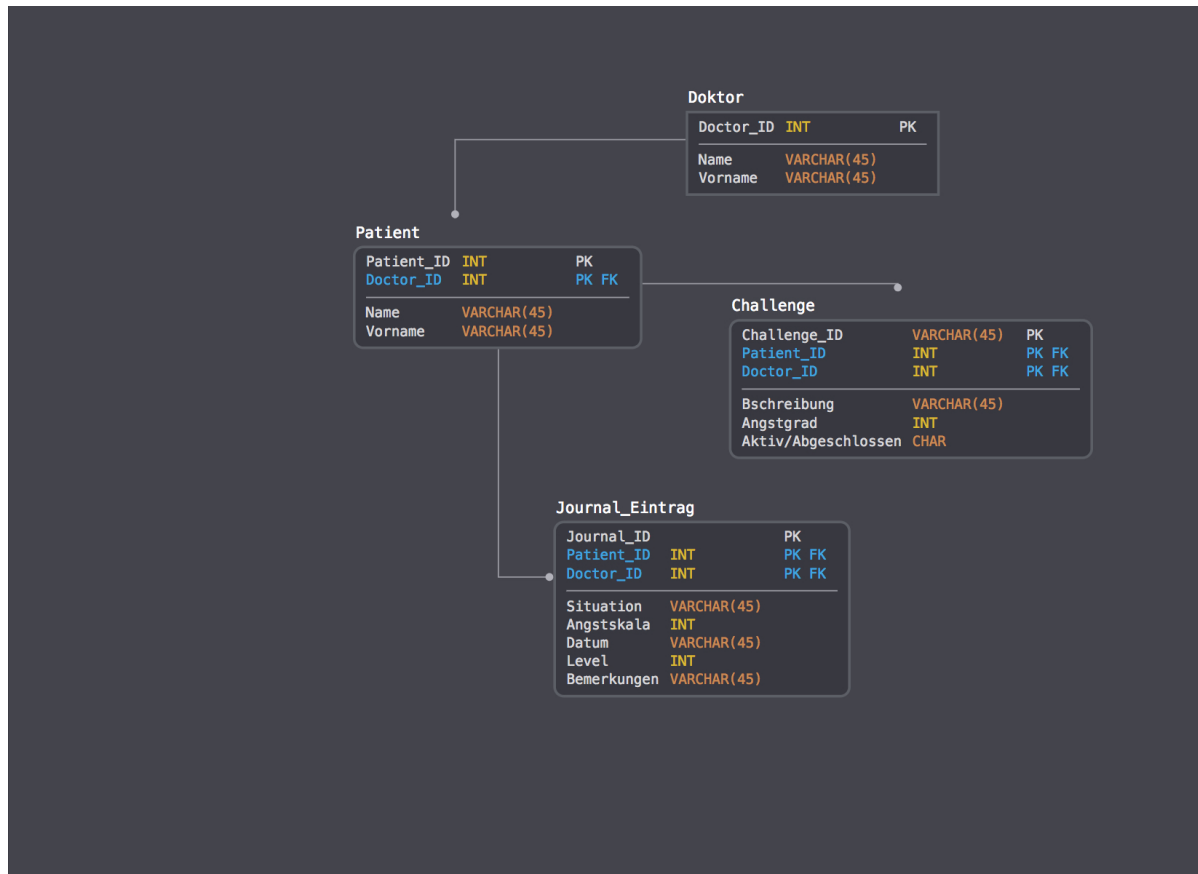


Figure 7.1: Database Schema

7.2 Modell des Data-Flow

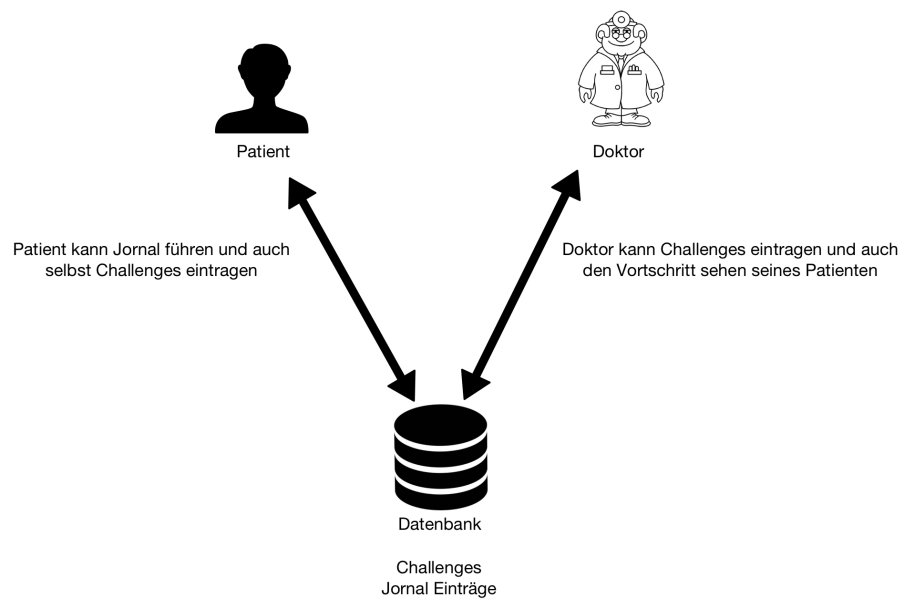


Figure 7.2: Data-Flow

8 System evolution

Software Evolution in Zusammenhang mit diesen acht Gesetzen von Lehman's.

9 Testing

Um sicherstellen zu können, dass unsere Applikation einwandfrei funktioniert werden wir wie folgt testen:

9.1 Was wird getestet?

Unser verschiedenen Futures werden zuerst einem umfangreichen funktionalen Test unterzogen. Dabei wird zuerst ein Unit testing und ein gesamter System Test durchgeführt. Weiter werden wir Sicherheitstests und zum Abschluss Interoperabilitätstests durchführen.

In einem zweiten Schritt werden Nicht-funktionalen Tests durchgeführt. Dabei wird die Benutzbarkeit, Effizienz sowie Zuverlässigkeit des Systems getestet. Gerade für die Usability-Tests werden wir externe Testpersonen brauchen. Dazu würden sich unsere Interviewpartner gut eignen.

9.2 Wie wird es getestet?

Zuerst werden aufgrund des Anforderungskatalogs verschiedene Testfälle entworfen. Verschiedene Testuser (interne und externe) bekommen ein Testprotokoll mit diesen Testfällen und den zu erwartenden Resultaten. Für die internen Test werden wir ein White-Box Testverfahren einsetzen. Für die externen werden wir ein Black-Box Verfahren vorbereiten.

10 Appendices

10.1 Hardware requirements

Server: Die wichtigste Anforderung an die Hardware ist die Erweiterbarkeit. Sie muss zu jederzeit einfach und schnell ausgebaut werden kann. Am Anfang, bei wenig Traffic auf dem Webserver wird die Hardware nur eine nebensächliche Rolle spielen. Bei steigenden Anfragen, muss die Hardware aber dementsprechend erweitert werden.

Client: Grundsätzlich kann auf unsere Applikation von jedem internetfähigen Gerät zugegriffen werden. Es ist aber empfohlen nicht mit veralteten Browsern die Applikation zu nutzen. Dabei kann nicht garantiert werden, dass alle Funktionen einwandfrei funktionieren.

10.2 Database requirements

Memory * Minimum: 1 GB Recommended: 4 GB

Processor Speed * Minimum: x64 Processor: 1.4 GHz Recommended: 2.0 GHz or faster
Processor Type x64 Processor: AMD Opteron, AMD Athlon 64, Intel Xeon with Intel EM64T support, Intel Pentium IV with EM64T support

Hard Disk * requires a minimum of 6 GB of available hard-disk space.

Drive * A DVD drive, as appropriate, is required for installation from disc.

Monitor * requires Super-VGA (800x600) or higher resolution monitor.

Internet * Internet functionality requires Internet access (fees may apply).

(Source: <https://docs.microsoft.com/en-us/sql/sql-server/install/hardware-and-software-req>)

11 Index