Software Engineering and Design

Versionierung

|  |  |  |  |
| --- | --- | --- | --- |
| # | Beschreibung | Name | Datum |
| 0.1 | Initiales Dokument | Dario Furigo | 09.04.2018 |
| 0.2 | Einleitung und Management Summary erstellt | Dario Furigo | 11.04.2018 |
| 0.3 | User requirements definition & System Evolution erstellt | Beat Schärz | 11.04.2018 |
| 0.4 | System requirements specification |  |  |

Inhaltsverzeichnis

[Inhaltsverzeichnis 2](#_Toc511245864)

[1. Management Summary 3](#_Toc511245865)

[2. Einleitung 4](#_Toc511245866)

[Zielgruppe 4](#_Toc511245867)

[Projektmitarbeiter 4](#_Toc511245868)

[Funktionsübersicht 4](#_Toc511245869)

[Patienten-Tagebuch 4](#_Toc511245870)

[Angehörigen-Tagebuch 4](#_Toc511245871)

[Zielvereinbarungen 5](#_Toc511245872)

[Anbindung an Umsysteme 5](#_Toc511245873)

[3. Requirements Specification Document 6](#_Toc511245874)

[4. Anhang 8](#_Toc511245875)

[5. Glossar 8](#_Toc511245876)

[6. Index 8](#_Toc511245877)

# Management Summary

Dieses Dokument beinhalt eine Zusammenfassung aller Anforderungen des Kunden und eine Übersicht über die Hauptfunktionen. Zudem ist ersichtlich welche technischen Voraussetzungen vorhanden sein müssen, damit das Patientenmanagementsystem erfolgreich umgesetzt werden kann.

Die Sprache dieses Dokumentes ist bewusst nicht-technisch gehalten, um den Ansprüchen der Zielgruppe gerecht zu werden.

# Einleitung

## Zielgruppe

Dieses Dokument bedient primär folgende Zielgruppen:

* Medizinisches Personal (Ärzte/Therapeuten)
* Leitende Angestellte von Spitälern und Kliniken
* Zentraler Einkauf von Spitälern und Kliniken
* Angehörige von Patienten mit Sozialer Phobie
* Entwickler/Architekten des PMS

## Projektmitarbeiter

|  |  |  |
| --- | --- | --- |
| Name | Github Username | Rolle |
| Andreas Erb | Res13 | Student |
| Beat Schärz | Pidu2 | Student |
| Nik Arm | projektins | Student |
| Dario Furigo | VirtSyntaxError | Student |
| Ivo Kozina | kozinai | Student |
| Janick Lüdi | janexander | Student |
| Ohran Mujkic | mujko1 | Student |

## Funktionsübersicht

Dieser Teil der Software soll folgende 3 Hauptfunktionen vorweisen:

### Patienten-Tagebuch

Hierbei handelt es sich um ein Tagebuch zwischen Patienten und deren Ärzten oder Therapeuten. Angehörige können Einträge (mit Einverständnis des Patienten) auch lesen, aber keine eigenen Einträge und Anpassungen vornehmen.

Das Tagebuch beinhaltet sowohl Instruktionen als auch Medikamentenverschreibungen.

### Angehörigen-Tagebuch

Das Angehörigen-Tagebuch sorgt für einen einfachen Informationsfluss zwischen Angehörigen und Ärzten/Therapeuten. Angehörige tragen hier z.B. alltägliche Beobachtungen über den Patienten ein um dem Arzt/Therapeuten Informationen über Fortschritte und Veränderungen im Verhalten einfach zur Verfügung zu stellen.

Ärzte/Therapeuten haben hier die Möglichkeit Angehörigen Tipps zu geben und sie zu Unterstützen.

Der Patient hat auf diese Einträge keinen Zugriff.

### Zielvereinbarungen

Dieses Feature ermöglicht dem behandelnden Arzt oder Therapeuten, sowie auch den näheren Angehörigen die einfache Erfassung definierter Ziele für den Patienten.

Laufend kann der aktuelle Status des Ziels angepasst werden. Ebenfalls gibt es die Möglichkeit ein Ziel zu favorisieren und dem Patienten dazu eine Mitteilung zu senden.

Der Patient selbst darf keine Ziele eintragen, anpassen oder löschen. Er hat aber jederzeit lesenden Zugriff auf diese.

## Anbindung an Umsysteme

Eine Anbindung an Umsysteme ist nicht vorgesehen, einzig die anderen Teile dieser Software, die in eigenen Projekten realisiert werden, sind als Schnittstellen vorhanden.

# Requirements Specification Document

## User requirements definition

Insgesamt gibt es drei grosse Services, welche wir mit dem System unseren Usern zur Verfügung stellen.

### 1. Patienten-Tagebuch

Das Tagebuch richtet sich sowohl an Ärzte, Patienten als auch an die Angehörigen.

Änderungen im Tagebuch dürfen allerdings nur Patiente und Ärzte vornehmen, Angehörige dürfen mit Einverständnis des Patienten die Einträge aber lesen.

Inhalt des Tagebuchs sind einerseits Instruktionen welche der Arzt dem Patienten gibt und andererseits eventuell anfällige Medikamente und deren Dosierung.

Dem Patienten (und Angehörigen) stehen so jederzeit die gegebenen Instruktionen und die Medikamenteninformationen zur Verfügung. Der Arzt kann sich mit dem Tagebuch auf die Behandlung vorbereiten und den Fortschritt des Patienten erkennen.

### 2. Angehörigen-Tagebuch

Dieses Tagebuch richtet sich einzig an die Angehörigen und Ärzte. Patienten haben hier keinen Zugriff darauf.

Das Angehörigen-Tagebuch dient vorallem dazu einen einfachen Informationsfluss zwischen den Angehörigen und Ärzten zu gewährleisten. Beispiele für Einträge wären hier erkannte Verhaltensänderungen des Patienten, Fortschritte oder einfach alltägliche Beobachtungen.

Auch Ärzte können Einträge vornehmen um die Angehörigen im Alltag mit Ratschlag zur Seite zu stehen.

### 3. Zielvereinbarungen

Mit dem Zielvereinbarungssystem wird den Usern, in diesem Fall den Ärzten, Patienten und Angehörigen, eine einfache Möglichkeit zur Verfügung gestellt, Ziele mit dem Patienten zu Erfassen und deren Status aktuell zu halten.

So kann z.B. der Status eines Ziels laufend angepasst oder auch einzelne Ziele priorisiert und dem Patienten sogar eine dazugehörige Mitteilung gesendet werden.

Der Patient hat hier keine Möglichkeit Änderungen vorzunehmen, kann aber jederzeit lesend aufs System zugreifen.

Ziele setzen dürfen sowohl Angehörige als auch Ärzte. Dies wird aber klar getrennt, da die Meinung des Arztes natürlich ausschlaggebender ist.

### Non-functional requirements

System

Beim System wird es sich um eine Web-App handeln.

Verfügbarkeit

Diese soll jederzeit, bis auf eventuell anfällige Wartungsarbeiten, zur Verfügung stehen.

Usability

Das System soll einfach zu bedienen sein, damit keine Schulung für die Benutzer notwendig ist.

Space

Da es sich bei den Informationen im System im Prinzip nur um Text handeln wird, wird es nicht sehr viel Platz in Anspruch nehmen. Bei vielen Usern kann sich über längere Zeit natürlich trotzdem eine grosse Menge an Informationen sammeln, weswegen in diesem Fall über eine Bereinigung älterer Einträge oder archivierung derer auf ein externes System nachgedacht werden kann.

Law

Der Datenschutz soll sich am Schweizerischen Datenschutzgesetztes über vertrauliche persönliche Informationen orientieren.

Weiter soll das System auch den Grundlagen des Schweizerischen Gesundheitsgesetzes entsprechen.

Development

Als Programmiersprache soll Java verwendet werden, da jedes Projektmitglied bereits Erfahrungen mit dieser Programmiersprache gemacht hat.

## System architecture

## System requirements specification

### Functional requirements

Development

Die Software wird mit Java Version 8, welche unter einer GNU Lizenz steht entwickelt. Als Framework wird Vaadin, das ein sehr mächtiges Tool für „rich internet applications“ ist. Die Lizenz für Vaadin ist Apache License 2.0.

Data

Die Applikation soll in der Lage sein mit einem Datenbanksystem zu kommunizieren. Hierbei wird auf eine SQL-Datenbank gesetzt. Die Daten sollen im System erstellt, bearbeitet, abgerufen sowie gelöscht werden können.

Responsive Design

Um eine gute UX zu gewährleisten soll die App responsive ausgerichtet sein. Das bedeutet in erster Linie, dass auf allen Gerätgrössen eine optimale Nutzung der App ermöglicht werden soll.

Roles

Es soll ein Berechtigungssystem geben, welches von bestimmten Benutzer angepasst werden. Ausserdem kürzt oder erweitert dieses Berechtigungskonzept die funktionalen Möglichkeiten des Benutzer. Ein nicht-bevormundeter Erkrankter (Soziale Phobie) soll die Möglichkeit erhalten die Berechtigungen der Angehörigen einschränken zu können.

Patienten-Tagebuch

Die Software ermöglicht einen einfachen Kommunikationsaustausch zwischen dem Klienten und dem Health Professional. Der Klient kann jeden Tag max. einen Tagebuch-Eintrag erfassen und mit seinem Betreuer sowie den Angehörigen teilen. Falls der Erkrankte nicht bevormundet ist, kann er die Leserecht der Angehörigen entziehen. Durch eine Kommentar-Funktion soll ähnlich wie bei einem Forum-Thread die Möglichkeit bestehen eine Diskussion zu führen.

Die Einträge sind, dank dem Berechtigungskonzept, nur vom Health Professional und Erkrankten bearbeitbar. Die Speicherung der Einträge erfolgt über die Datenbank.

Damit jeden Tag ein Eintrag erfasst wird, werden die Angehörigen per Mail informiert.

Angehörigen-Tagebuch

Die Software ermöglicht einen einfachen Kommunikationsaustausch zwischen dem Angehörigen und dem Health Professional. Der Angehörige kann jeden Tag max. einen Tagebuch-Eintrag erfassen und mit dem Health Professional teilen. Durch eine Kommentar-Funktion soll ähnlich wie bei einem Forum-Thread die Möglichkeit bestehen eine Diskussion zu führen.

Die Einträge sind, dank dem Berechtigungskonzept, nur vom Health Professional und Angehörigen bearbeitbar. Die Speicherung der Einträge erfolgt über die Datenbank.

Damit jeden Tag ein Eintrag erfasst wird, werden die Angehörigen per Mail informiert.

Zielsetzung

Das Berechtigungskonzept ermöglicht, dass Angehörige sowie Health Professionals in der Lage sind Ziele für den Erkrankten zu setzen. Die Erkrankten haben auf dieser Maskenansicht nur Leserechte.

Die Ziele sind auf der Datenbank gespeichert und werden von dort aus auch aufgerufen.

Damit der Erkrankte das Ziel nicht vergisst, wird er anhand eines Reminder (Mail) auf das aktuelle Ziel aufmerksam gemacht.

### Non-Functional requirements

Performance

Die Applikation sollte eine schnelle Reaktionszeit haben. Dabei sollten die Daten auch möglichst schnell geladen werden, so das für den Benutzer kaum Wartezeiten entstehen.

Privacy

Sicherheitslücken, besonders betreffend Personendaten, dürfen in der Applikation nicht vorhanden sein. Die Software sollte nach bestem Gewissen dem Datenschutzgesetz entsprechen.

Design

Für alle Benutzergruppen sollte das Layout der Applikation ansprechend sein. Die Verwendung einer solchen Software sollte selbsterklärend sowie einfach sein.

## System models

## System evolution

Für die System-Evolution kann man sich an den 8 Lehmans Gesetzen orientieren.

Die beiden wichtigsten Teile welche man diesen entnehmen kann sind, dass das System kontinuierlich an Neuerungen angepasst werden muss, da es ansonsten an Effektivität verliert und dass verhindert werden sollte, dass Änderungen der Software zu Unübersichtlichkeit und extremer Komplexität führen.

Die Software soll von Anfang an so aufgebaut werden, dass das Hinzufügen neuer Funktionalitäten auf Grund von Userwünschen kein Problem darstellt.

## Testing

# Anhang

# Glossar

|  |  |
| --- | --- |
| Begriff | Definition |
| PMS | Patient Management System / Patientenmanagementsystem |
| MHC | Mental Health Care / Psychische Gesundheitsfürsorge |
| RBAC | Role Based Access Control |

# Index