

Bern University
of Applied Sciences

Software Engineering and Design

Requirements Specification

Mental Health Care Patient Management System (MHC-PMS)

Team White:

Dellsperger Jan
Ellenberger Roger
Sheppard David
Sidler Matthias
Spring Mathias
Thöni Stefan

April 5, 2016

Version 1.0

Contents

1 Vorwort	4
1.1 Über dieses Dokument	4
1.2 Zielgruppe	4
1.3 Änderungsnachweis	4
2 Einleitung	5
3 Glossar	6
4 User-Requirements Definition	7
4.1 User Requirements	7
4.2 Use-Case 1	8
4.3 Use-Case 3	9
5 System Architektur	11
6 System-Requirements Spezifikation	12
6.1 Functional Requirements	12
6.1.1 UC1: Dashboard	12
6.1.2 UC3: Reporting	12
6.1.3 Allgemein	13
6.2 Non-Functional Requirements	13
6.3 Domain Requirements	14
6.3.1 Branchenspezifisch	14
6.3.2 Rechtlich	14
7 System Models	15
8 System Evolution	16
9 Testing	17
10 Appendix	18
10.1 Verwendete Spezifikationen	18
10.1.1 FMEA	18
10.2 Quellen	19
11 Index	20

1 Vorwort

1.1 Über dieses Dokument

Dieses Dokument beschreibt den Requirements-Engineering-Prozess des Projekts *MHC-PMS*. Es spezifiziert die Erkenntnisse auf dem Design-Thinking-Prozess.

1.2 Zielgruppe

Das Dokument richtet sich an den Endkunden, die Projektleitung, die Personalplanung, die Entwickler, Test-Engineers und das zukünftige Betriebsteam.

1.3 Änderungsnachweis

Version	Beschreibung	Autor	Datum
0.1	Dokument aus Vorlage erstellt	Team White	April 1, 2016
1.0	Finale Version des Dokuments	Team White	April 5, 2016

2 Einleitung

Die Betreuung von Personen mit psychischen Störungen soll mit unserer Software vereinfacht werden. Die Zielkundschaft sind kleine bis grosse Einrichtungen für die Behandlung (ambulant und Hausbesuche) von Patienten mit psychischen Störungen.

Wir fokussieren dabei auf Funktionen für das Management. Wir möchten Personen mit Führungsfunktion bei Planungsarbeiten, administrativen Tätigkeiten und Strategie-Entscheidungen unterstützen. Die Verwaltung und Auswertung von Patientendaten (Behandlungshistorie und Verrechnung) steht dabei im Fokus. Zudem soll die Personalplanung mit der Patientenverwaltung verknüpft werden. Export von Berichten für Partnerorganisationen und Behörden soll möglichst unkompliziert gestaltet werden.

Unser Produkt ist rein für die Datenauswertung gedacht. Als Datenquellen nutzen wir eine bestehende Patientenverwaltung (mit Personalplanung und Verrechnung). Durch die gewonnenen Erkenntnisse soll das Management ihre Strategie mit Fakten fundiert steuern können und im Alltag weniger administrativen Aufwand vorfinden. In der Gesundheitsbranche sollen damit Kosteneinsparungen erreicht werden. Ineffiziente Behandlungsarten können schneller erkannt und Patienten mit schlechten Behandlungserfolg besser überwacht werden.

3 Glossar

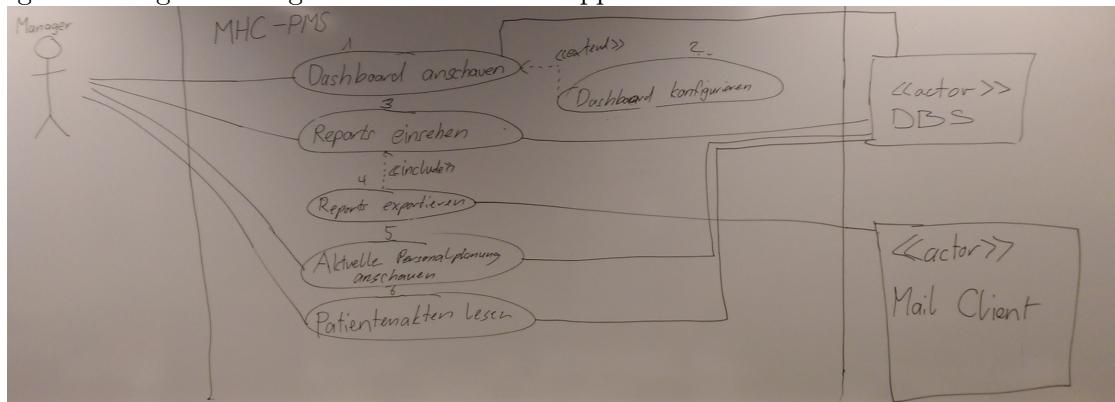
Begriff	Beschreibung
Dashboard	Übersichtseite zu einem Themenbereich
HIT	Health IT: Informatik-Sparte, die sich mit dem Gesundheitswesen auseinandersetzt
MHC-PMS	Mental Health Care Patient Management System
DBS	Database System
Report	Bericht / Auswertung

4 User-Requirements Definition

Nachfolgender Abschnitt beschreibt die User-Requirements der Applikation.

4.1 User Requirements

Folgendes Diagramm zeigt die Use-Cases der Applikation.



Es existieren sechs Use-Cases, wo der Benutzer mit dem System interagiert:

1. **Dashboard anschauen:** Daten werden aufbereitet und auf dem Dashboard angezeigt
2. **Dashboard konfigurieren:** Manager personalisiert sein persönliches Dashboard je nach Anforderungen individuell
3. **Reports einsehen:** Manager lässt Report zur Einsicht generieren
4. **Reports exportieren:** Generierte Reports werden zum digitalen Versand, Ausdruck oder lokalen Abspeichern exportiert.
5. **Aktuelle Personalplanung anschauen:** Manager sieht aktuelle Personalplanung an
6. **Patientenakten lesen:** Manager sieht Patientenakten ein

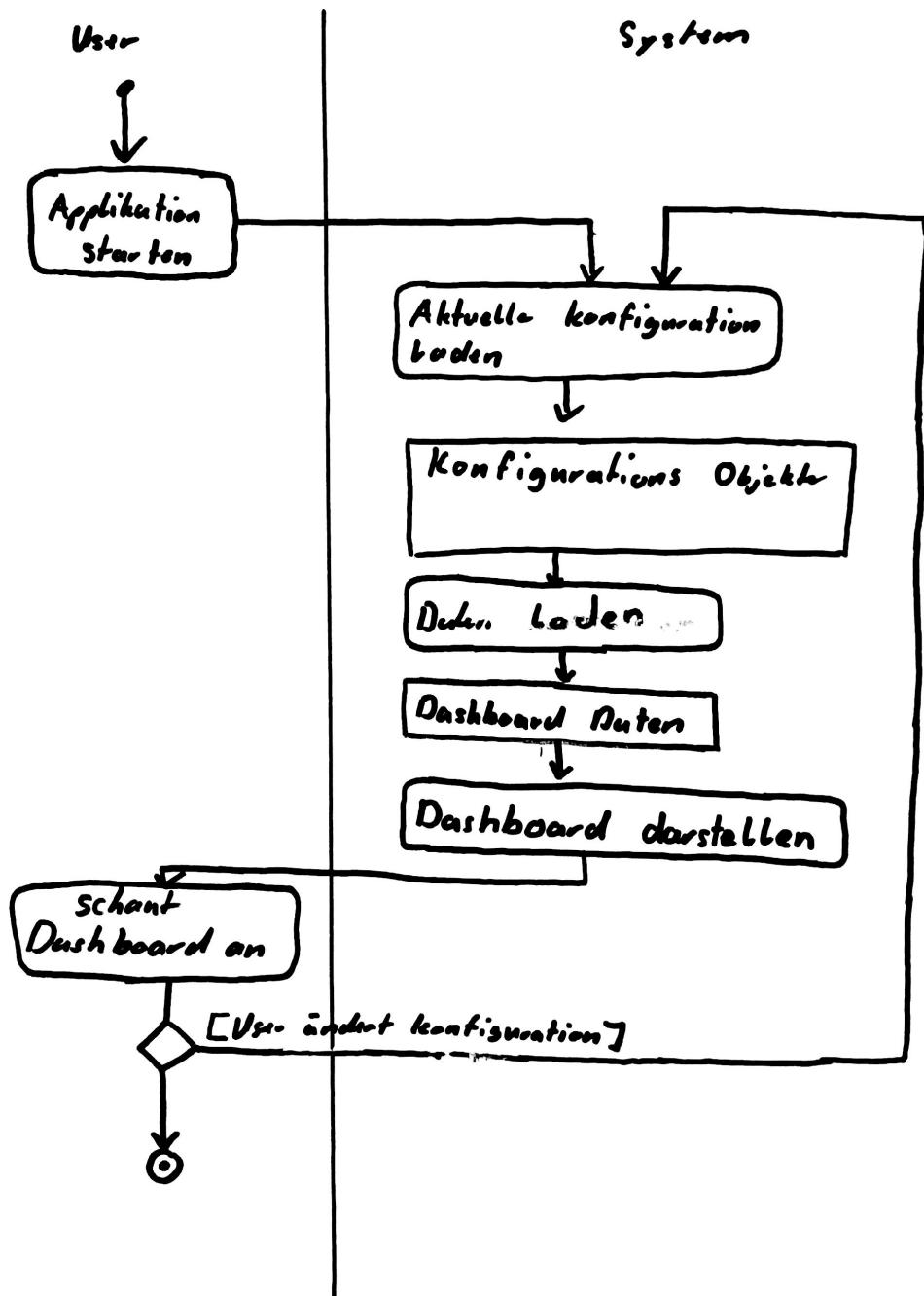
Es existieren zudem zwei weitere Akteure. Zum einen der zum Versenden der Reports benutzte Mail-Client, der auf den Endgerät vorausgesetzt wird, und ein Datenbanksystem "DBS" (oder allenfalls mehrere Datenbanksysteme) als Datenquelle.

Nachfolgend sind Use-Case eins und drei noch genauer beschrieben.

4.2 Use-Case 1

Aktivitätsdiagramm

Aktivitäts-Diagramm: Dashboard anschauen



Szenario

Name:	Dashboard Anschauen
Nummer:	1
Kurzbeschreibung:	Daten werden aufbereitet und auf dem Dashboard angezeigt
Beteiligte Akteure:	Manager, System, DBS
Auslöser/Vorbedingung:	-
Ergebnis/Nachbedingung:	Daten sind angezeigt

Ablauf:

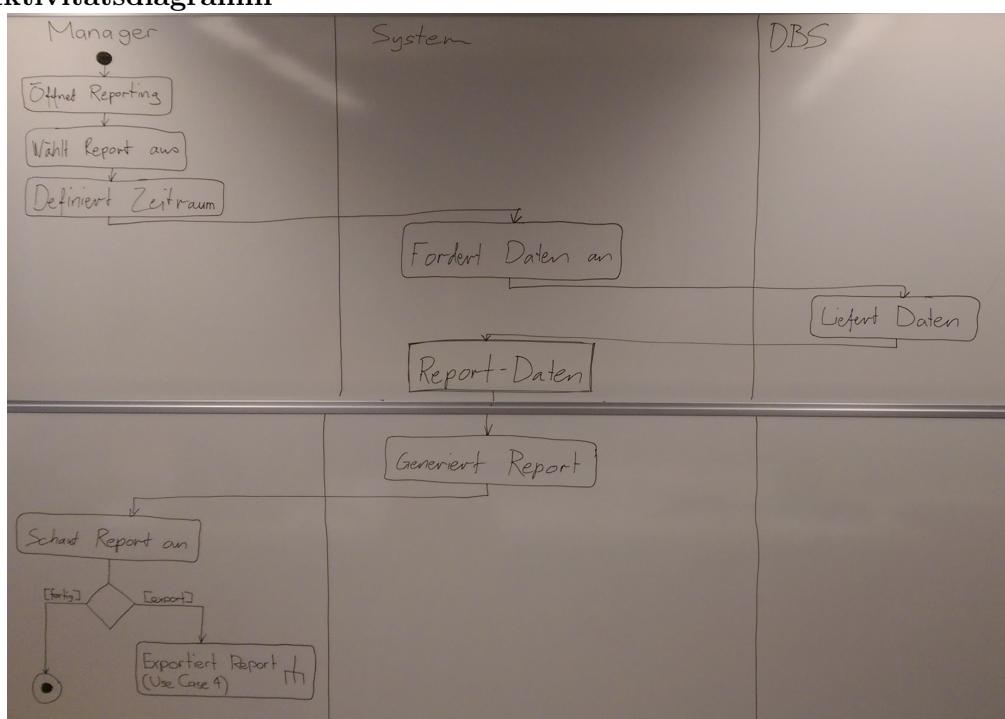
Nr.	Wer	Was
1.0	Manager	startet die Applikation
1.1	System	lädt die aktuelle Konfiguration
1.2	System	fordert Daten an
1.3	DBS	liefert Daten
1.4	System	verarbeitet Daten
1.5	System	stellt das Dashboard dar
1.6	Manager	schaut das Dashboard an

Ausnahmen/Varianten:

Nr.	Wer	Was
1.6	Manager	ändert die Konfiguration
1.6.1	System	Vorgang ab Nr. 1.1 wiederholen

4.3 Use-Case 3

Aktivitätsdiagramm



Szenario

Name:	Sieht Report ein
Nummer:	3
Kurzbeschreibung:	Manager lässt Report zur Einsicht generieren
Beteiligte Akteure:	Manager
Auslöser/Vorbedingung:	Öffnet Reporting
Ergebnis/Nachbedingung:	Manager erhält Report

Ablauf:

Nr.	Wer	Was
3.0	Manager	Öffnet Reporting
3.1	Manager	Wählt Report aus
3.2	Manager	Definiert Zeitraum
3.3	System	Fordert Daten an
3.4	DBS	Liefert Daten
3.5	System	Generiert Report
3.6	Manager	Schaut report an
3.7	Manager	Variante: Wünscht Export

Ausnahmen/Varianten:

Nr.	Wer	Was
3.7	Manager	Variante: Wünscht Export
4	Manager	Exportiert Report

5 System Architektur

Bei der Applikation handelt es sich um eine Webanwendung, welche für mobile Endgeräte optimiert ist. Ein gutes Benutzererlebnis soll aber auch auf anderen Endgeräten wie PCs oder Notebooks gewährleistet sein. Die Applikation befindet sich in einer isolierten Applikationslandschaft mit lediglich einem DBS als Umsystem. Als Rahmenbedingung wird angenommen, dass dieses DBS alle Daten liefern kann, welche von Requirements verlangt werden.

6 System-Requirements Spezifikation

6.1 Functional Requirements

6.1.1 UC1: Dashboard

[1.1] Darstellung Dashboard Das System soll beim Starten oder alternativ per Klick auf ein Menu-Punkt ein Dashboard anzeigen. Dieses Dashboard soll, in Kacheln dargestellt, konfigurierte Kennzahlen und Statistiken darstellen. (Siehe Requirement [1.6]) Desweiteren soll per Klick auf einen solchen Kachel der entsprechende Report angezeigt werden.

[1.2] Dashboard Konfiguration Das System soll einen Menu-Punkt zur Verfügung stellen, der die Konfiguration des Dashboards erlaubt. Dies beinhaltet:

- Wieviele Kacheln dargestellt werden (Bis zu einem Maximum von 6).
- Welche Kennzahlen/Statistiken in Kacheln dargestellt werden.
- In welcher Reihenfolge diese angezeigt werden

6.1.2 UC3: Reporting

[1.3] Reporting anzeigen Das System soll via Menu-Punkt eine Reporting Funktionalität anbieten. Dazu wird eine Art Wizard gestartet, der in folgenden Schritten zum Report führt:

1. **Reporttyp wählen:** Gemäss den Kennzahlen kann jeweils ein Report erstellt werden (Siehe Requirement [1.6])
2. **Zeitraum wählen:** Der Zeitraum kann frei gewählt werden. Insbesondere soll es möglich sein, individuelle Perioden (beispielsweise Quartal oder halbjährlich) zu definieren.
3. (Optional) **Weiteren Zeitraum hinzufügen:** Falls Zeiträume verglichen werden sollen, besteht hier noch die Option, eine weitere Periode hinzuzufügen.

Danach wird die Grafik angezeigt und in einer zweiten View die Primärdaten tabelarisch aufgelistet. Zwischen diesen Views kann zum Beispiel mit einer Swipe Geste oder Menu Buttons gewechselt werden. Falls einen weiteren Zeitraum ausgewählt wurde, werden die Grafiken und Daten einander gegenübergestellt.

Desweiteren soll beim generierten Report die Möglichkeiten bestehen, einen Shortcut für das Dashboard zu erstellen (Siehe Requirement [1.2]) oder den Report zu exportieren (Siehe Requirement [1.4]).

[1.4] Report Export Alle Reports sollen zum digitalen Versand, Ausdruck oder lokalen Abspeichern exportiert werden können. Dies beinhaltet folgende Formate

- **PDF:** Report als PDF
- **CSV:** Report als CSV
- **Mail:** Report kann mit vorausgesetztem Mail-Client direkt als PDF versandt werden

Zusätzlich kann gewählt werden, ob der Export Grafik, Primärdaten oder beides enthalten soll.

6.1.3 Allgemein

[1.5] Schnittstelle DBS Die Applikation soll in der Lage sein, Daten von einem dezentralen DBS zu laden und anzuzeigen.

[1.6] Kennzahlen Alle Kennzahlen können jeweils auf frei wählbare Zeitspannen berechnet werden, vorausgesetzt, die Daten sind im DBS hinterlegt. Als Beispiel soll es möglich sein, die Personalausfälle von heute sowie auch Personalausfälle vom März letzten Jahres einsehen zu können.

Folgende Kennzahlen werden von der Applikation unterstützt:

- Personelles: verfügbares Personal, nicht verfügbares Personal (Ferien etc.), unvorhergesehene Personalausfälle (Krankheit, Unfall o.ä.)
- Medizinisches: Besondere Vorkommnisse (Medikamentenverweigerungen, Gewalttaten, Suizidversuche, Suizide), Eintritte, Austritte, Aktuelle Patientenzahl)
- Finanzielles: Aufwand, Ertrag, Cash Flow

6.2 Non-Functional Requirements

[2.1] Performance: Starten der Applikation Das System soll nicht länger als 4 Sekunden benötigen um nach Start der Applikation das Dashboard anzuzeigen. (4 Sekunden sollten genügen um die relevanten Daten zu laden und anzuzeigen. Außerdem ist es genügend kurz um den Benutzer nicht zu belasten)¹

[2.1] Stabilität: Ausführung der Applikation Die Applikation soll im Betrieb keine Fehler aufweisen, welche die Kategorie 4 der FMEA übersteigen. Falls doch solche Fehler auftreten, müssen diese gemäss Requirement 2.2 behoben werden.²

¹ Anderson, Shaun (2016). www.hobo-web.co.uk.

² Ackermann (2015). S.208.

[2.2] Wartung: Fehlerbehebung Der Rollout und Delivery-Prozess muss ermöglichen, schwerwiegende Fehler (>FMEA Kat. 4) innerhalb nützlicher Frist (2 Arbeitstage) zu beheben.

[2.3] Datensicherheit Die Datensicherheit muss vom DBS gewährleistet werden. Die Applikation ist nicht verantwortlich für die Integrität und Sicherheit jeglicher Daten.

[2.4] Security Die Applikation soll die sensitiven Daten mittels HTTPS abhörsicher übertragen.

6.3 Domain Requirements

6.3.1 Branchenspezifisch

[3.1] Leistungsabrechnung Implementieren der Standards für Leistungserfassung gemäss TARMED, damit die verrechneten Leistungen überwacht werden können.

[3.2] Meldepflicht Ermöglichen von Exports der Behandlungshistorie aus den Patientendaten. Dies wird oft von Krankenkassen verlangt, wenn ihnen die vorhandenen Angaben nicht ausreichen.

[3.3] Arztgeheimnis Gewährleisten des Arztgeheimnis durch Implementation von Berechtigungsstrukturen auf Benutzerebene. Ein Eintrag in der Patientenakte, welche ein Patient nur seinem Arzt vertraulich mitgeteilt hat, muss vor unerlaubten Zugriffen geschützt sein.

6.3.2 Rechtlich

[3.4] Datenschutz Einhalten der schweizerischen Datenschutzrichtlinien für den Umgang mit den Patientendaten.

[3.5] Aufbewahrungspflicht Einhalten der 10-Jährigen Datenaufbewahrungspflicht (je nach Gesellschaftsform nach Schweizer Recht vorgeschrieben ³).

³RA Rigert, Christian und MLaw Seger, Alice (2013). www.dieadvokatur.ch.

7 System Models

8 System Evolution

In zukünftigen Versionen könnten folgende erweiternde oder neue Funktionen implementiert werden:

- Weitere Reports
Aufgrund von Rückmeldungen und Wünschen aus dem Daily Business können weitere Reports definiert werden.
- Eigene Reports definieren
Der User kann direkt in der Applikation auf den vorhandenen Daten eigene Reports konfigurieren und abspeichern. Hierzu steht dem Benutzer ein Report-Designer zur Verfügung.
- Big Data Funktionalitäten
Auswertungen werden um Big Data Funktionalitäten erweitert. So erhält der Benutzer noch bessere und tiefere Einblicke in seine Daten und deren Zusammenhänge.

9 Testing

10 Appendix

10.1 Verwendete Spezifikationen

10.1.1 FMEA

Failure Mode and Effects Analysis für die Software wird bewertet anhand untenstehender Tabelle.¹

	Severity of Effect	Occurrence	Detection
1	Cosmetic error: No loss in product functionality (incorrect doc., etc.).	1 / 100 years	Requirements
2	Cosmetic error: No loss in product functionality (incorrect doc., etc.).	1 / 10 years	Requirements
3	Product performance reduction, problem will "go away" after a period of time.	1 / year	Coding
4	Product performance reduction, problem will "go away" after a period of time.	1 / month	Coding
5	Functional impairment/loss: problem will not resolve itself, but "work around" can bypass problem without losing operation.	1 / week	Introduction phase
6	Functional impairment/loss: problem will not resolve itself, but "work around" can bypass problem without losing operation.	1 / day	Introduction phase
7	Functional impairment/loss: problem will not resolve itself, no "work around", functionality impaired or lost, but product can still be used to some extent.	1 / shift	Installation / start-up
8	Functional impairment/loss: problem will not resolve itself, no "work around", functionality impaired or lost, but product can still be used to some extent.	1 / hour	Installation / start-up
9	Product halts/process taken down/reboot required: product is hung up, all functionality lost and system reboot required.	1 / minute	Installation / start-up
10	Product halts/process taken down/reboot required: product is hung up, all functionality lost and system reboot required.	1+ / minute	In operation

¹Ackermann (2015). S.208.

10.2 Quellen

Referenz Vollnachweis

- 1 Anderson, Shaun (2016). How Fast Should A Website Load.
<http://www.hobo-web.co.uk/your-website-design-should-load-in-4-seconds/> (01.04.2016).
- 2, 4 Ackermann, Urs. 2015. Betriebswirtschaftslehre 2. S.208.
- 3 RA Rigert, Christian und MLaw Seger, Alice (2013). Rechtlichen Aspekte der elektronischen Datenaufbewahrung. http://www.dieadvokatur.ch/fileadmin/user_upload/Publikationen/Fachartikel/2013/Die_rechtlichen_Aspekte_der_elektronischen_Archivierung.pdf (01.04.2016).

11 Index