

Requirements for Patient Management System

1. Preface

Expected Readership

Dieses Dokument richtet sich an die an dem Projekt PMS beteiligten Systemingenieure, Endbenutzer, Systemadministratoren und Manager auf Kundenseite.

Distribution Schema

#	Description	Initials	Date
1.0	Initial Document created	alle	21.03.2017
1.01	Formatierung	schms27	23.03.2017
1.02	Non Functional Requirements, System model, Appendices	thekm1/baiuj1	24.03.2017

2. Introduction

Initial	Name	Role
jodet1	Joder Tobias	Student
thekem1	Thekkekara Mathew	Student
ladel1	Läderach Lukas	Student
schms27	Schmid Simon	Student
baiuj1	Baiutti Joris	Student
vonac1	von Allmen Cedric	Student
vgj1	Prof. Dr. Jürgen Vogel	Dozent

Das vorliegende Dokument beschreibt die Anforderungen an das durch Teamgreen™ zu implementierende PMS **Borderline Organisation and Operations Business System** zur Verwaltung von Patienten mit psychischen Krankheiten mit besonderem Fokus auf Patienten mit der Borderline-Persönlichkeitsstörung.

Die momentane Situation im Bereich der Patientenverwaltung besteht aus vielen individuellen Teilsystemen, die einen Informationsaustausch nur sehr umständlich zulassen.

Dies führt dazu, dass viele Patientenverwaltungssysteme redundante Patientendaten enthalten. Dadurch ist der Datenverwaltungsaufwand bei Kliniken, Praxen und Beratungsstellen momentan noch sehr hoch.

Das Gesamtsystem soll als zentrale Informationsquelle für Ärzte, Klinikpersonal, Sozialarbeiter, Patienten und Verwandte dienen. In diesem Dokument wird der Fokus auf die Rolle des Health Visitors (Sozialarbeiter, Sozialberater) gelegt.

3. Glossary

Term	Definition
PMS	P atient M anagement S ystem
DMZ	D emilitarized Z one: Computernetz mit kontrollierten Zugriffsmöglichkeiten auf die angeschlossenen Server
RDBMS	R elational D atabase M anagement S ystem

4. User requirements definition

Functional User Requirements

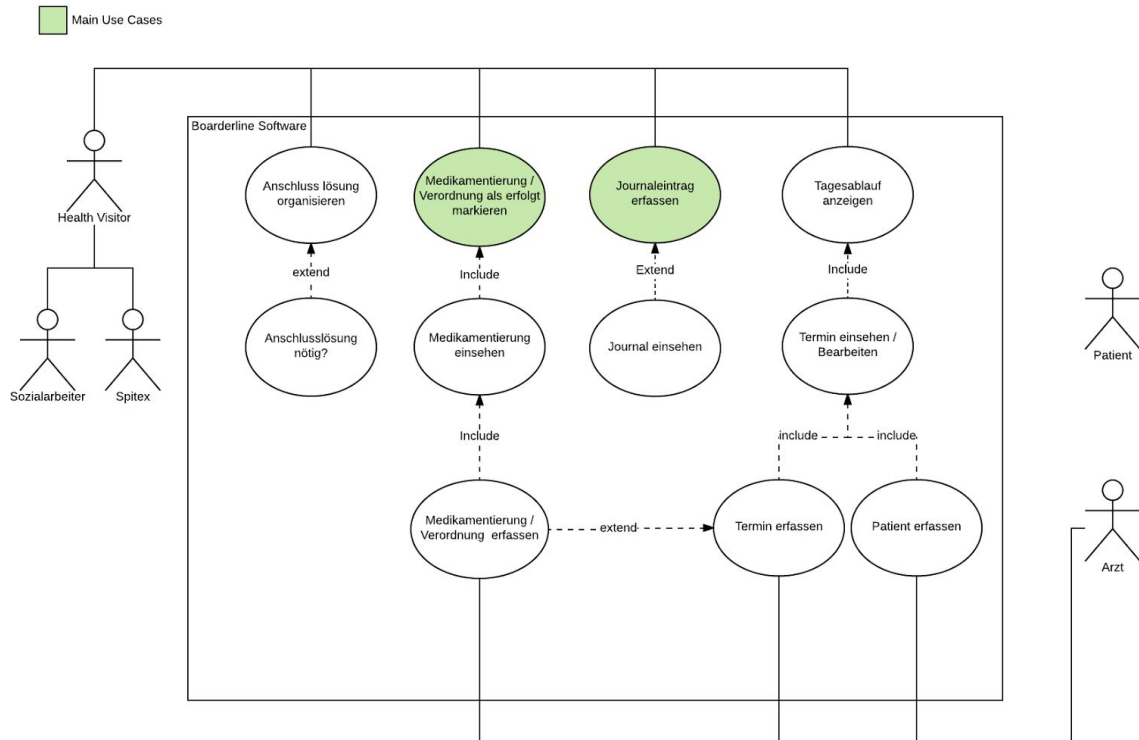
#	Requirement
1	Eine Wissensdatenbank mit Informationen über das Krankheitsbild
2	Spitex MitarbeiterIn hat Einsicht in die Medikamente und kann sie vor Ort verabreichen. Wenn die Medikamente verabreicht wurden, werden sie von der Person vor Ort abgehakt
3	Auf der App gibt es einen Alarmknopf, welcher den Ort direkt an die Notfallzentrale übermittelt.
4	Arzt kann Patient Medikamente inklusive Dosierung verordnen und im System registrieren
5	Patient muss zu 'Meine Patienten' hinzugefügt werden können
6	Sozialarbeiterin muss Patient im System suchen können
7	Allgemeine Informationen wie Wohnort, Telefonnr. müssen abgerufen werden können
8	Sozialarbeiterin kann Informationen zu Hilfsmitteln im Notfall(Notfallkit) für Borderline Patienten im System abrufen.
9	Sozialarbeiterin muss Zugriff auf ausgewählte Gesprächsprotokolle zwischen Arzt und Patient haben.
10	Sozialarbeiterin muss per Handykamera vor Ort gemachte Fotos zu Patientenakte hinzufügen können.
11	Die grafischen Dokumente sollen auch auf den mobilen Geräten zoombar sein
12	Es soll ein systemweiter Chat verfügbar sein, wo sich die Sozialarbeiterin bei Unklarheiten beispielsweise schriftlich beim behandelnden Arzt melden kann.
13	Sozialarbeiterin kann Gesprächsverlauf und Gesamteindruck über Bewertungsskala im System hinterlegen.

14	Sozialarbeiterin kann Gefährdungsmeldung über System an zuständige Behörde senden.
15	Als Gesundheitsexperte will ich, eine Übersicht meiner Patientenbesuche mit Zeit und Ort einsehen können
16	Als Gesundheitsexperte will ich nach einer Visite meine Notizen im Journal des Patienten dokumentieren
17	Als Gesundheitsexperte will ich, dass das System vorhandene Notschlafstellen mit Telefonnummern anzeigt. Zur Unterstützung beim Organisieren der Anschlusslösung.
18	Als Gesundheitsexperte will ich nach einer Visite im System eine Anschlusslösung für den Patienten dokumentieren können
19	Als Gesundheitsexperte will ich im Journal des Patienten nachvollziehen können welche Anschlusslösung gewählt wurde.
20	Als Gesundheitsexperte will ich ein Terminversäumnis im Journal dokumentieren / nachvollziehen können.
21	Als Gesundheitsexperte will ich das Datum eines bestehenden Termins verändern können.
22	Als Gesundheitsexperte will ich den Änderungsverlauf eines Termins einsehen könne.
23	Als Gesundheitsexperte will ich einen Journaleintrag des Patienten als "wichtig" markieren können.

Non Functional User Requirements

#	Requirement
1	Datenschutz, Geheime Patientendaten
2	Immer Verfügbar
3	Zuverlässig, sonst kann der Alarmknopf nicht genutzt werden
4	Performant, Kurze ladezeiten
5	Benutzerfreundlich, Intuitive Benutzerführung

Use Case Diagram



Auf dem Diagramm sind die ausgewählten 4 Use Cases mit ihren Subtasks ersichtlich. Die Subtasks werden als “extends” vererbt wenn der Main Task den Subtask optional verwendet. Die include dependency wird verwendet um zu verdeutlichen dass der subtask immer bei der benutzung des Main Tasks zwingend ausgeführt werden muss.

Die Aktoren und deren zusammenhang mit den Tasks sind an den verbindungslinien zu sehen.

Die Main Use cases welche grün gefärbt sind werden auf den nächsten Seiten detailliert.

Use Case Scenarios

Scenario 1:

Nr. and Name:	01 - Verordnungen einsehen und erledigen
Scenario:	Medikament wird dem Patienten durch Health Visitor nach Hause geliefert.
Short Description:	Health Visitor ist in der Lage die verordneten Medikament zu einem Patienten abzurufen, eine "Rüstliste" zu erstellen und die entsprechenden Medikament dem Patienten auszuhändigen oder zu verabreichen. Danach kann im System das Medikament als "Ausgegeben" markiert werden und dadurch eine lückenlose Rückverfolgung der Medikamentierung gewährleistet werden.
Actors:	Arzt, Health Visitor, Patient
Starting Event and Preconditions:	Patient benötigt Medikamentierung Arzt hat Medikament verschrieben
Result and Postconditions:	Patient erhält das Medikament ausgehändigt und / oder verabreicht

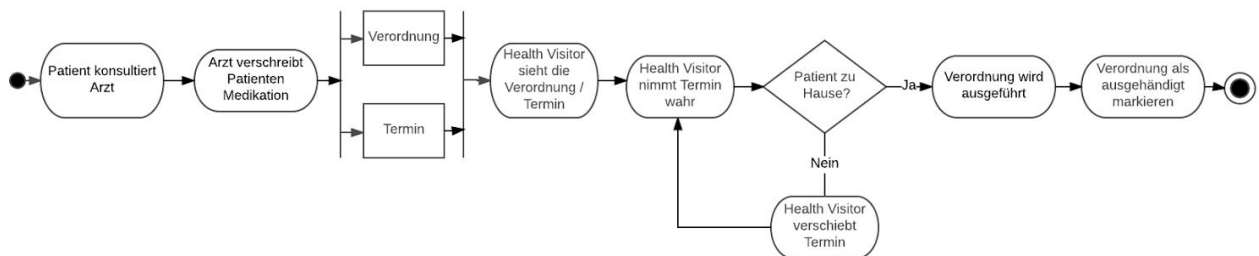
Steps:

Nr.	Actor	Description
1.0	Patient	Patient konsultiert Arzt
2.0	Arzt	Arzt verschreibt dem Patienten Medikation
2.1	Arzt	Er trägt die verschriebenen Menge, Einnahmezyklus und Dauer der Medikamentierung ins PMS ein.
3.0	Health Visitor	Health Visitor sieht auf dem Patientendossier die verschriebene Medikamentierung
3.1	Health Visitor	Rüstet entsprechende Medikamente für den heutigen Tagesablauf.
4.0	Health Visitor, Patient	Händigt die Medikamente dem Patienten aus
5.0	Health Visitor	Markiert das Medikament als "Ausgehändigt"

Exceptions, Variants:

Nr.	Actor	Step
1.0	Patient	Patient ist nicht zu Hause
1.1	Health Visitor	Versucht den Patient per Telefon zu kontaktieren
1.2	Health Visitor	Falls der Patient nicht erreicht werden kann wird dies entsprechend im 02 - Journal vermerkt

Activity Diagramm zu Use Case Scenario 01: Verordnungen einsehen und erledigen



Scenario 2:

Nr. and Name:	02 - Journal
Scenario:	Neuer Eintrag in das Journal
Short Description:	Health Visitor erstellt einen neuen Journal Eintrag. Dieser Eintrag wird anschliessend im Journal gespeichert.
Actors:	Health Visitor, System
Starting Event and Preconditions:	Patient muss im System erfasst sein
Result and Postconditions:	Journal Eintrag im Journal Verlauf

Steps:

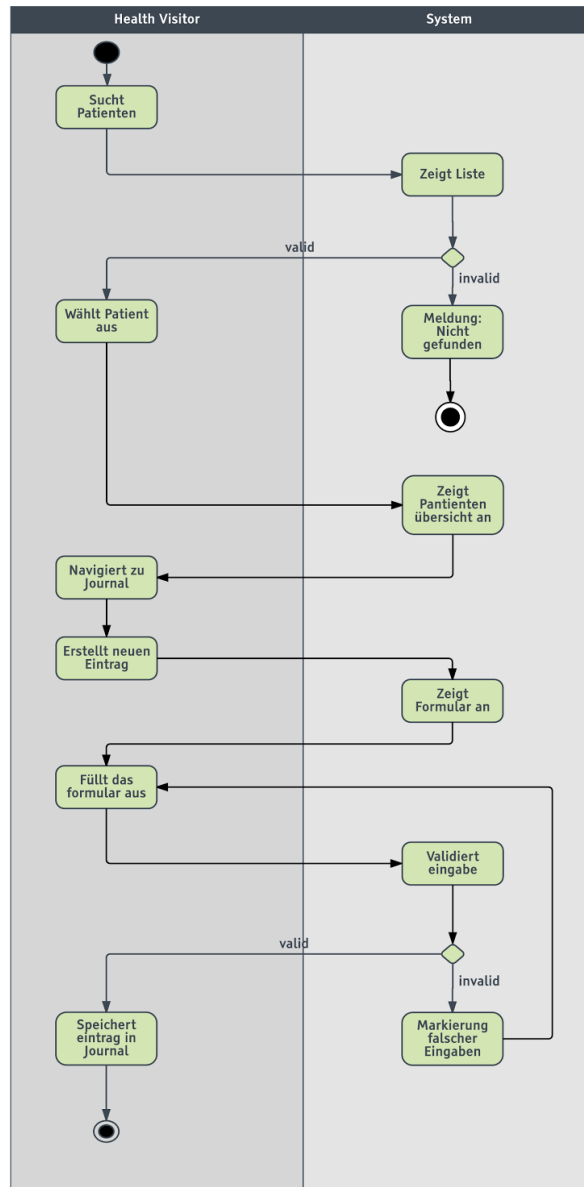
Nr.	Actor	Description
1.0	Health Visitor	sucht nach Patient im System
2.0	System	Zeigt eine Liste von Patienten
3.0	Health Visitor	wählt gewünschten Patienten
4.0	System	Zeigt die Patientenübersicht
5.0	Health Visitor	navigiert zu der Rubrik Journal und erstellt einen neuen Eintrag
6.0	Health Visitor	füllt das Formular aus
7.0	System	Validiert Eingabe
8.0	System	Speichert Eintrag im Journal

Exceptions, Variants:

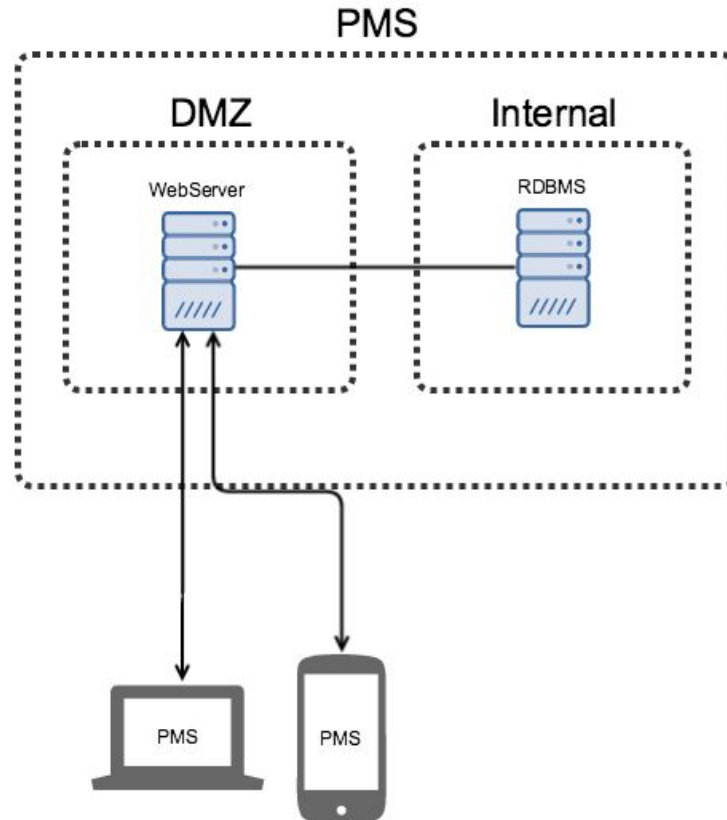
Nr.	Actor	Step
2.0	System	Patient kann nicht gefunden werden
2.1	System	Zeigt an -> Patient nicht vorhanden
7.0	System	Validierung schlägt fehl

7.1	System	Falsche Eingaben werden markiert
7.2	Health Visitor	Korrigiert Eingabe
7.3	System	Validiert Eingabe

Activity Diagramm zu Use Case Scenario 02: Journal: Neuer Eintrag in das Journal



5. System architecture



Der Webserver vom PMS wird über Https öffentlich verfügbar gemacht. Die Clients kommunizieren über das standardisierte Protokoll https mit dem Webserver. Die Datenbank ist nicht direkt von aussen erreichbar, nur der Webserver kommuniziert direkt mit der Datenbank.

6. System requirements specification

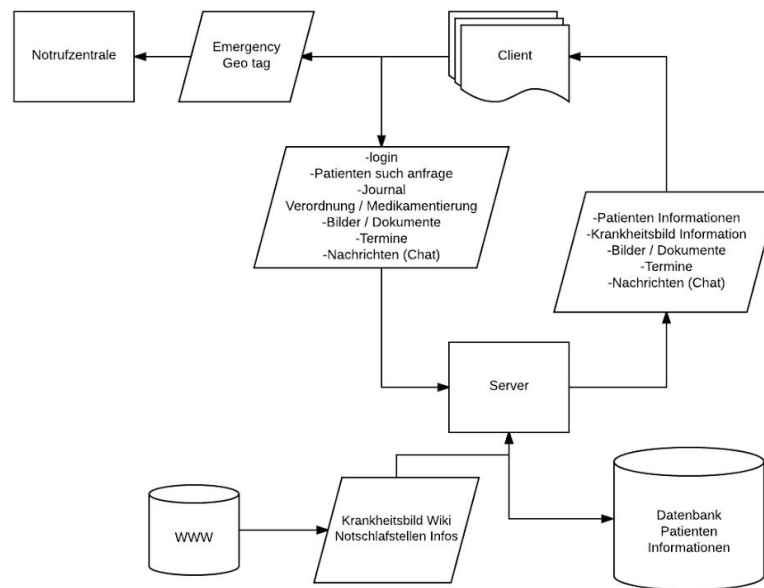
Functional System Requirements

#	Requirement
1	Daten sollten zusätzlich separat gesichert werden (Backup)
2	Daten müssen in einer gewissen Zeit wieder hergestellt werden können
3	Datenbank für das speichern der Daten
4	WebServer für den Zugriff über einen Browser
5	Sicher Datenverbindung zwischen Client und Server
6	Sichere Datenverbindung zwischen Datenbank und WebServer
7	Sichere Authentifizierung am System
8	Export und Import von Daten

Non Functional System Requirements

#	Requirement
1	System muss gegen Angriffe von außen geschützt sein
2	System sollte zu jeder Zeit verfügbar sein
3	Skalierbar Infrastruktur
4	Erfassung auch offline möglich, aktuelle Daten sind offline verfügbar
5	Fehlertoleranz
6	Wartbarkeit, Flexibel bei Änderungen

7. System models



8. System evolution

Das System soll vorerst primär auf die Bedürfnisse der Endusergruppe Healthvisitors ausgelegt werden. Allerdings kann die zukünftige Ausweitung auf weitere Benutzergruppen nicht ausgeschlossen werden. Die Implementierung dieser Benutzergruppe ist eine gute Grundlage um in Zukunft ein umfängliches System aufzubauen, das dem Bedürfnis eines erleichterten Datenaustauschs zwischen den verschiedenen Dienstleistern nach kommt. Auch soll das System im Bezug auf die abgedeckten psychischen Krankheiten jederzeit erweiterbar bleiben. Ein agiler Entwicklungsprozess hilft ebenfalls dabei, auf ändernde Anforderungen von Benutzerseite einzugehen.

9. Testing

Komponenten Tests

Um eine gute Softwarequalität zu erreichen, wird der Code einerseits durch sinnvolle Unittests (o.ä.) durch den Entwickler getestet. Andererseits soll durch Codereviews ein hoher Standard bezüglich des Codes erreicht werden.

Integrations Tests

Die "Sprints" sollen jeweils mit einem vollständig funktionierenden Programm abgeschlossen werden. Um dies zu erreichen wird am Schluss eines jeden Sprints durch Integrationstest geprüft, ob alle Komponenten gemäss den geplanten Abläufen zusammen funktionieren und die korrekten Ergebnisse liefern.

System Tests

Ebenfalls am Ende der Sprints werden die Ergebnisse gegen die Spezifikation geprüft. Somit wird klargestellt, dass die Requirements umgesetzt wurden und die Funktionalität gegeben ist. Zusätzlich werden Load Tests durchgeführt, um sicher zu sein, dass bei voller Auslastung sich das System korrekt verhält.

Abnahme Test

Zum Abschluss wird zusammen mit dem Kunden ein Abnahmetest durchgeführt.

10. Appendices

Systemvoraussetzungen:

User:

- Mobiles Gerät (Smartphone, Tablet) mit einer Internetverbindung. Die aktuellen, auf dem Markt erhältlichen Geräte bieten die notwendige Leistung.
- Chrome, Safari oder Firefox auf dem Gerät installiert
- Mobile Datenverbindung

Server:

- Stabile Datenverbindung (1GB/s Synchron)
- Firewall für die DMZ

- Virtualisierung damit es skalierbar ist
- Genügend Speicher für die Datenmasse
- Genügend Arbeitsspeicher und Rechenleistung um die Anzahl Zugriffe zu verarbeiten

Database:

- To be defined