**PMS – Health Visitor Team Black**

****

**Organizer**

**Requirements Specification**

Gruppenmitglieder:

Bühler Ueli

Jahn Tim Allan

Kandiah Rajilatha

Klembowski Raphael

Ristic Nikola

Sutter Christoph

Inhaltsverzeichnis

[1. Preface 3](#_Toc5634892)

[1.1 Versionsverlauf: 3](#_Toc5634893)

[2. Introduction 3](#_Toc5634894)

[3. Glossary 4](#_Toc5634895)

[4. User requirements definition 4](#_Toc5634896)

[4.1 Use Case - Planung und Route 5](#_Toc5634897)

[4.1.1 Tages/Wochenübersicht Anzeigen U1.1 5](#_Toc5634898)

[4.1.2 Nächster Patient U1.2 6](#_Toc5634899)

[4.1.3 Navigation zum Patienten U1.3 6](#_Toc5634900)

[4.2 Use Case - Verlaufsbericht erstellen 7](#_Toc5634901)

[4.2.1 Tages/Wochenübersicht Anzeigen (Analog U1.1) 7](#_Toc5634902)

[4.2.2 Patienten für Verlaufsbericht auswählen 8](#_Toc5634903)

[4.2.3 Verlaufsbericht ausfüllen 8](#_Toc5634904)

[5. System Architecture 9](#_Toc5634905)

[5.1 Database 9](#_Toc5634906)

[5.2 WebApp 9](#_Toc5634907)

[5.3 Client 9](#_Toc5634908)

[6. System Requirements Specifications 9](#_Toc5634909)

[6.1 Functional Requirements 9](#_Toc5634910)

[6.2 Non-Functional Requirements 10](#_Toc5634911)

[6.3 Interfaces 10](#_Toc5634912)

[7. System Models 10](#_Toc5634913)

[8. System Evolution 11](#_Toc5634914)

[8.1 Web User 11](#_Toc5634915)

[8.2 App-User 11](#_Toc5634916)

[8.3 System Generell 11](#_Toc5634917)

[9. Testing 11](#_Toc5634918)

[9.1 Unit Tests 11](#_Toc5634919)

[9.2 Integration Tests 11](#_Toc5634920)

[9.3 Application Tests 11](#_Toc5634921)

[9.4 Story Tests 12](#_Toc5634922)

[10. Appendices 12](#_Toc5634923)

# Preface

Dieses Dokument richtet sich an alle involvierten Teammitglieder des Projektes "Spitex Organizer". Es inkludiert neben dem Team Black auch die Seite des Anwenders welche durch unsere Dozenten repräsentiert wird. Das Dokument soll jedoch auch für Laien verständlich und interpretierbar sein.

# Versionsverlauf:

|  |  |  |
| --- | --- | --- |
| **Version** | **Changelog** | **Date** |
| 0.1 | Initialdokument, Inhaltspunkte, Introduction | 01.04.2019 |
| 0.2 | Versionsverlauf-styling update | 01.04.2019 |
| 0.3 | Systemarchitektur hinzugefügt | 05.04.2019 |
| 0.4 | System Models und System Evolution hinzugefügt | 06.04.2019 |
| 0.5 | Diverse Korrekturen, Updates und Final Touch | 08.04.2019 |

# Introduction

Die Software "Spitex Organizer" des Teams Black soll die tägliche Arbeit von Spitexmitarbeitern bestmöglichst vereinfachen und somit zur Verbesserung der Arbeitsbedingungen wie auch zur Wirtschaftlichkeit beitragen.

Insbesondere liegt der Schwerpunkt der Software, soweit möglich, auf dem Krankheitsbild von Zwangsstörungen der "Obsessive Compulsive Disorder" oder kurz "OCD". Die Kernfunktionen umfassen insbesondere die Darstellung der Tages- und Wochenplanungen der Spitexmitarbeiter sowie deren, beim Patienten, zu erledigenden Tasks. Zusätzlich legt die Applikation besonderen Fokus auf das Rapportieren und das periodische Evaluieren der Patienten.

Der "Spitex Organizer" ist überwiegend eine anwenderspezifische Software mit möglichst einfacher und intuitiver Bedienung für die benutzende Person und soll sowohl in einem Browser wie auch auf einem Tablett genutzt werden können.

Die App soll die Spitexmitarbeiter bei der täglichen Arbeit unterstützen und jederzeit Einblicke in die zu erledigenden Arbeiten wie auch Patientendaten ermöglichen. Ebenso soll sie die Mitarbeiter bei der täglichen Rapportierung unterstützen. Auf diesem Weg soll die Planung vereinfacht, die Effizienz und somit die Wirtschaftlichkeit gesteigert und im gleichen Zug die Arbeitsbedingungen verbessert werden.

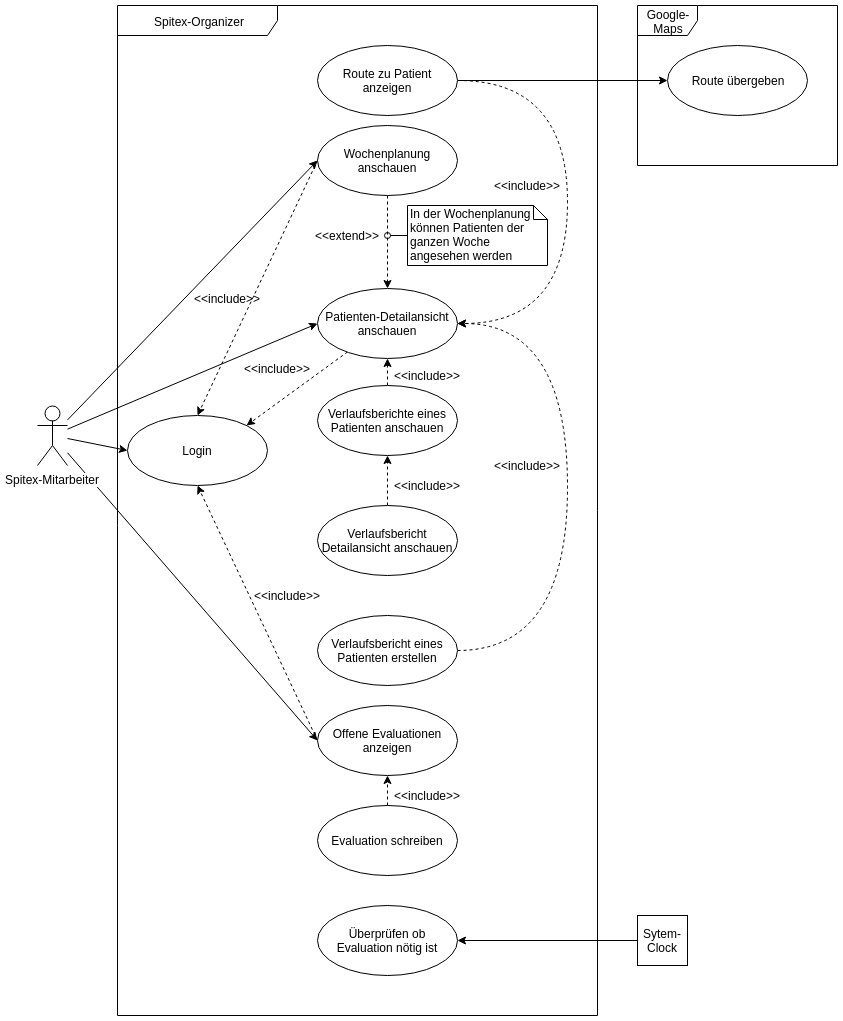
In einem weiterführenden Projekt kann die Software mit einer zusätzlichen Schnittstelle ausgestattet werden. Somit könnte der "Spitex Organizer" auch in bereits bestehende Systeme integriert werden.

# Glossary

|  |  |
| --- | --- |
| Abkürzung | Bedeutung |
| Client | Wird vom Benutzer verwendet um das System zu bedienen |
| PMS | Patient Management System |
| Spitexmitarbeiter | Pfleger/In der Spitex Organisation |
| Web App | Eine Applikation auf die man via Browser zugreifen kann, analog einer Website |

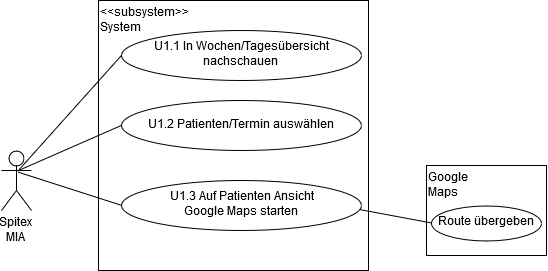
# User requirements definition

**Overview:**



* U1 Gruppe Planung und Route
  + U1.1 Wochenübersicht
  + U1.2 Aufruf der Patienten Ansicht
  + U1.3 Adresse an Google Maps übergeben
* U2 Gruppe Verlaufsberichte
  + U2.1 Wochen/Tagesansicht
  + U2.2 Aufruf der Patienten Ansicht
  + U2.3 Bearbeiten des Verlaufsberichts

# Use Case - Planung und Route



# Tages/Wochenübersicht Anzeigen U1.1

|  |  |
| --- | --- |
| Use Case ID | U1.1 |
| Name: | Tages/Wochenübersicht |
| Szenario: | Der Spitexmitarbeiter sucht seinen nächsten Termin in der Tages/Wochenansicht |
| Kurzbeschreibung: | Der Spitexmitarbeiter muss nach einem erledigten Termin zum nächsten |
| Beteiligt Akteure: | Spitexmitarbeiter |
| Auslöser /  Vorbedingung: | Anwendung läuft  User erfolgreich Authentifiziert |
| Ergebnisse /  Nachbedingung: | Der Spitexmitarbeiter findet schnell seinen nächsten Termin |

Ablauf:

|  |  |  |
| --- | --- | --- |
| Nr. | Wer | Was |
| 1.1.1 | Spitexmitarbeiter | Der Spitexmitarbeiter öffnet Wochen/Tagesplanung |
| 1.1.2 | Spitexmitarbeiter | Der Spitexmitarbeiter kann nun seinen Tages/Wochenplan betrachten |
|  |  |  |

# Nächster Patient U1.2

|  |  |
| --- | --- |
| Use Case ID | U1.2 |
| Name: | Nächster Patient |
| Szenario: | Der Spitexmitarbeiter hat den nächsten Termin in der Wochen/Tagesansicht gefunden |
| Kurzbeschreibung: | Der Spitexmitarbeiter muss nach einem erledigten Termin zum nächsten |
| Beteiligt Akteure: | Spitexmitarbeiter |
| Auslöser /  Vorbedingung: | Anwendung läuft  User erfolgreich Authentifiziert |
| Ergebnisse /  Nachbedingung: | Der Spitexmitarbeiter findet schnell seinen nächsten Patienten |

Ablauf:

|  |  |  |
| --- | --- | --- |
| Nr. | Wer | Was |
| 1.2.1 | Spitexmitarbeiter | Der Spitexmitarbeiter öffnet die Patienten Ansicht via Wochen/Tagesplanung |
| 1.2.2 | Spitexmitarbeiter | Auf der Patienten Ansicht findet er die Adresse seines Patienten |
|  |  |  |

# Navigation zum Patienten U1.3

|  |  |
| --- | --- |
| Use Case ID | U1.3 |
| Name: | Navigation zu Patienten |
| Szenario: | Der Spitexmitarbeiter will zu einem Patienten fahren, kennt jedoch den Weg nicht. |
| Kurzbeschreibung: | Da der Spitexmitarbeiter den Weg zum Patienten nicht kennt, muss er das System konsultieren um den Weg zu finden. |
| Beteiligt Akteure: | Spitexmitarbeiter, Google Maps |
| Auslöser /  Vorbedingung: | Anwendung läuft  User erfolgreich Authentifiziert  Spitexmitarbeiter hat bereits Patienten Ansicht geöffnet |
| Ergebnisse /  Nachbedingung: | Der Spitexmitarbeiter wird durch Google Maps an sein Ziel geführt. |

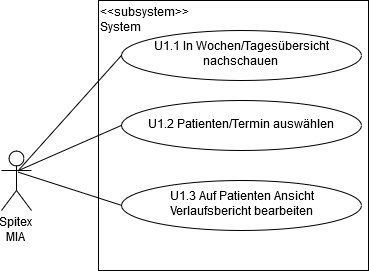
Ablauf:

|  |  |  |
| --- | --- | --- |
| Nr. | Wer | Was |
| 1.3.1 | Spitexmitarbeiter | Auf der Patienten Ansicht kann der Spitexmitarbeiter die Routenplanung starten |
| 1.3.2 | System | Das System übergibt Google Maps die Adresse für die Routenführung |
|  |  |  |

Ausnahmen, Varianten:

|  |  |  |
| --- | --- | --- |
| Nr. | Wer | Was |
| 1.3.1.1 | Spitexmitarbeiter | Der Spitexmitarbeiter kennt die Route bereits und braucht das Navigationssystem nicht |
|  |  |  |

# Use Case - Verlaufsbericht erstellen



# Tages/Wochenübersicht Anzeigen (Analog U1.1)

|  |  |
| --- | --- |
| Use Case ID | U2.1 |
| Name: | Tages/Wochenübersicht |
| Szenario: | Der Spitexmitarbeiter sucht nach einem besuchten Patienten |
| Kurzbeschreibung: | Der Spitexmitarbeiter sucht nach einem bereits vergangenen Termin |
| Beteiligt Akteure: | Spitexmitarbeiter |
| Auslöser /  Vorbedingung: | Anwendung läuft  User erfolgreich Authentifiziert |
| Ergebnisse /  Nachbedingung: | Der Spitexmitarbeiter findet in der Tages/Wochenansicht seine besuchten Patienten |

Ablauf:

|  |  |  |
| --- | --- | --- |
| Nr. | Wer | Was |
| 2.1.1 | Spitexmitarbeiter | Der Spitexmitarbeiter öffnet Wochen/Tagesplanung |
| 2.1.2 | Spitexmitarbeiter | Der Spitexmitarbeiter kann nun seinen Tages/Wochenplan betrachten und seine besuchten Patienten suchen |
|  |  |  |

Ausnahmen, Varianten:

|  |  |  |
| --- | --- | --- |
| Nr. | Wer | Was |
|  |  |  |

# Patienten für Verlaufsbericht auswählen

|  |  |
| --- | --- |
| Use Case ID | U2.2 |
| Name: | Patienten für Verlaufsbericht auswählen |
| Szenario: | Der Spitexmitarbeiter hat den gesuchten Patienten gefunden |
| Kurzbeschreibung: | Der Spitexmitarbeiter wählt einen Patienten aus bei welchem noch ein Verlaufsbericht ausstehend ist |
| Beteiligt Akteure: | Spitexmitarbeiter |
| Auslöser /  Vorbedingung: | Anwendung läuft  User erfolgreich Authentifiziert  Wochen/Tagesansicht wurde ausgewählt |
| Ergebnisse /  Nachbedingung: | Der Spitexmitarbeiter sucht bei seinen besuchten Patienten nach einem bestimmten Termin und wählt diesen aus |

Ablauf:

|  |  |  |
| --- | --- | --- |
| Nr. | Wer | Was |
| 2.1.1 | Spitexmitarbeiter | Auf der Patienten Ansicht kann der Spitexmitarbeiter den Verlaufsbericht aufrufen |
| 2.1.2 | System | Zeigt alle bisherigen Verlaufsberichte an |
|  |  |  |

# Verlaufsbericht ausfüllen

|  |  |
| --- | --- |
| Use Case ID | U2.3 |
| Name: | Verlaufsbericht ausfüllen |
| Szenario: | Der Spitexmitarbeiter will einen Verlaufsbericht verfassen |
| Kurzbeschreibung: | Der Spitexmitarbeiter muss nach jedem erledigten Termin einen Verlaufsbericht erstellen |
| Beteiligt Akteure: | Spitexmitarbeiter |
| Auslöser /  Vorbedingung: | Anwendung läuft  User erfolgreich Authentifiziert  Patienten Ansicht wurde ausgewählt |
| Ergebnisse /  Nachbedingung: | Der Spitexmitarbeiter kann einen neuen Verlaufsbericht hinzufügen |

Ablauf:

|  |  |  |
| --- | --- | --- |
| Nr. | Wer | Was |
| 2.3.1 | Spitexmitarbeiter | Wählt den leeren Verlaufsbericht aus und füllt diesen entsprechend aus |
| 2.3.2 | Spitexmitarbeiter | Der Verlaufsbericht wird abgespeichert |
|  |  |  |

Ausnahmen, Varianten:

|  |  |  |
| --- | --- | --- |
| Nr. | Wer | Was |
| 2.3.1.1 | System | Das System entdeckt eine ungültige Eingabe und gibt eine Fehlermeldung aus |
|  |  |  |

# System Architecture



# Database

Die Datenbank liefert alle benötigten persistente Informationen über:

* Patienten
* Termine
* Verlaufsberichte
* Evaluationen
* etc.

# WebApp

Die Applikation, welche die Business Logik ausführt und mit der Datenbank kommuniziert.  
Sie antwortet auf Requests vom Client.

# Client

Der Benutzer muss irgendwie auf die Applikation zugreifen. Dies macht er durch einen Client, welcher in den meisten Fällen ein Web-Browser sein wird.

# System Requirements Specifications

# Functional Requirements

**Das System soll:**

* periodisch überprüfen ob für einen Patienten eine Evaluation fällig ist.
* Daten in eine Datenbank speichern können.
* Patientendaten anzeigen können.
* Verlaufsberichte von Patienten anzeigen können.
* Verlaufsberichte erstellen können.
* Google Maps öffnen können und die Route zum Patienten anzeigen.
* die Wochenplanung vom Spitexmitarbeiter anzeigen.
* die Eingabefelder vor dem Verarbeiten validieren.

# Non-Functional Requirements

**Das System soll:**

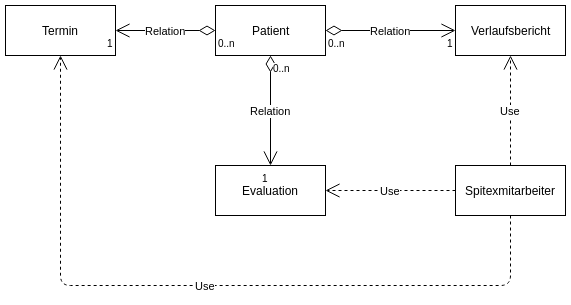
* via Website und Smartphone oder Tablet verfügbar sein.
* eine benutzerfreundliche Oberfläche bieten.
* dem Benutzer Fehlermeldungen liefern, wenn die Validierung einer Eingabe negativ ist.
* intuitiv zu bedienen sein.
* ist responsive.

# Interfaces

**Das System soll:**

* in eine bestehende IT-Infrastruktur eingebettet werden können.
* mit bestehenden Systemen interagieren können.

# System Models



* Patient
  + Name, Vorname, Wohnadresse, Alter, Telefon, Krankheit, Medikamente, Termine
* Spitexmitarbeiter
  + Name, Vorname, Benutzername, Passwort, Telefon
* Evaluation
  + Zeitstempel
* Verlauf
  + Patient, Verlauf, Zeitstempel, Spitexmitarbeiter

# System Evolution

# Web User

In der ersten Version soll das System lediglich die Möglichkeit bieten via Webseite auf die App zuzugreifen. Dieser Zugriff kann von einem Computer wie auch von einem Tablet oder Handy erfolgen.

# App-User

Bezüglich der Anforderung der Routenplanung wird nach einer erfolgreichen stabilen Version auf Basis der Website eine App eingeführt. Dies soll bereits während der Entwicklung berücksichtigt werden damit die App, komplett unabhängig von der Website, verwendet werden kann.

# System Generell

Für weiterführende Funktionen gibt es viele Möglichkeiten, wie beispielsweise:

* Im System einen Arbeitsplan (Kalenderform) für jeden Spitexmitarbeiter erstellen
* Automatisierte Erstellung von Routen zu Patienten um Weg und Zeit zu optimieren
* Automatisierte Evaluations Vergleiche um den Fortschritt der Behandlung zu analysieren
* Anhand der Rapporte können die erbrachten Leistungen direkt bei den richtigen Stellen in Rechnung gestellt werden

# Testing

# Unit Tests (White Box)

Diese Tests haben das Ziel einzelne Komponenten, Klassen oder Methoden auf ihre Funktionalität und Korrektheit zu prüfen. Diese werden nach jedem Build und Check-In automatisch ausgeführt. Eine Erfolgsquote von 100% ist dabei Voraussetzung für die Fortsetzung.

# Integration Tests (White Box)

Diese Tests haben das Ziel die Funktionalität der verschiedenen Komponenten untereinander zu verifizieren. Dies geschieht ebenfalls automatisch und im Falle eines Fehlers muss dieser anschliessend erst eliminiert werden.

# Application Tests (Black Box)

Diese Tests haben das Ziel das gesamte System zu prüfen. Hauptsächlich soll hier automatisch getestet werden ob Applikation und Datenbank korrekte Antworten auf spezifische Anfragen liefern.

# Story Tests

Die abgeschlossenen Use Cases sollen von einem Tester manuell geprüft werden sobald diese umgesetzt wurden. Hier geschieht alles manuell.

# Appendices

Serverseitig wird die Software inklusive der Datenbank in ein bestehendes IT-System integriert. Im Falle einer ungenügenden Performance muss die bestehende Infrastruktur erweitert werden können. Auf der Seite des Clients wird darauf geachtet, dass möglichst jede gängige Hardware (Tablett, Mobiltelefon oder PC/Mac) unterstützt wird.