

Software Requirements Specification

Version 1.0

Preface

Date	Version	Description	Author
29.03.2016	0.1	Document setup	Nicolas Schmid
02.04.2016	0.2	Added Use-Cases	Karol Ugorcak
02.04.2016	0.x	Added Use-Cases	Denis Shevchenko
03.04.2016	0.x	Added Use-Cases and System Architecture	Nicolas Schmid
03.04.2016	0.x	Added Use-Cases, translation	Michel Hosmann
02.04.2016	0.x	Added Use-Case Diagram	Nicolas Schmid
03.04.2016	0.x	Added Einführung / System Models / Funktionale Anforderungen	Christoph Sutter
04.04.2016	0.x	Added Calendar / My Appointments Use-Cases	Sergii Bilousov
04.04.2016	1.0	Translation, final checks	alle

Einführung

Diese Applikation soll sich unterstützend in den Alltag eines Health-Visitors einbringen. Sie soll die Tagesplanung und Checklisten zur täglichen Arbeit enthalten damit jeder Betreuer seine Individuellen Termine inkl. Materiallisten sofort im Blick hat. Des Weiteren soll die App auch eine Rapportfunktion enthalten welche vom Betreuer nach jedem Besuch ausgefüllt

werden kann mit einer allfälligen Option einer Unterschrift der zu betreuenden Person. Mit den Daten aus den Rapporten als Grundlage soll eine Auswertung gemacht werden um den Verlauf, der Betreuung, der Kunden/Patienten zu prüfen.

Glossar

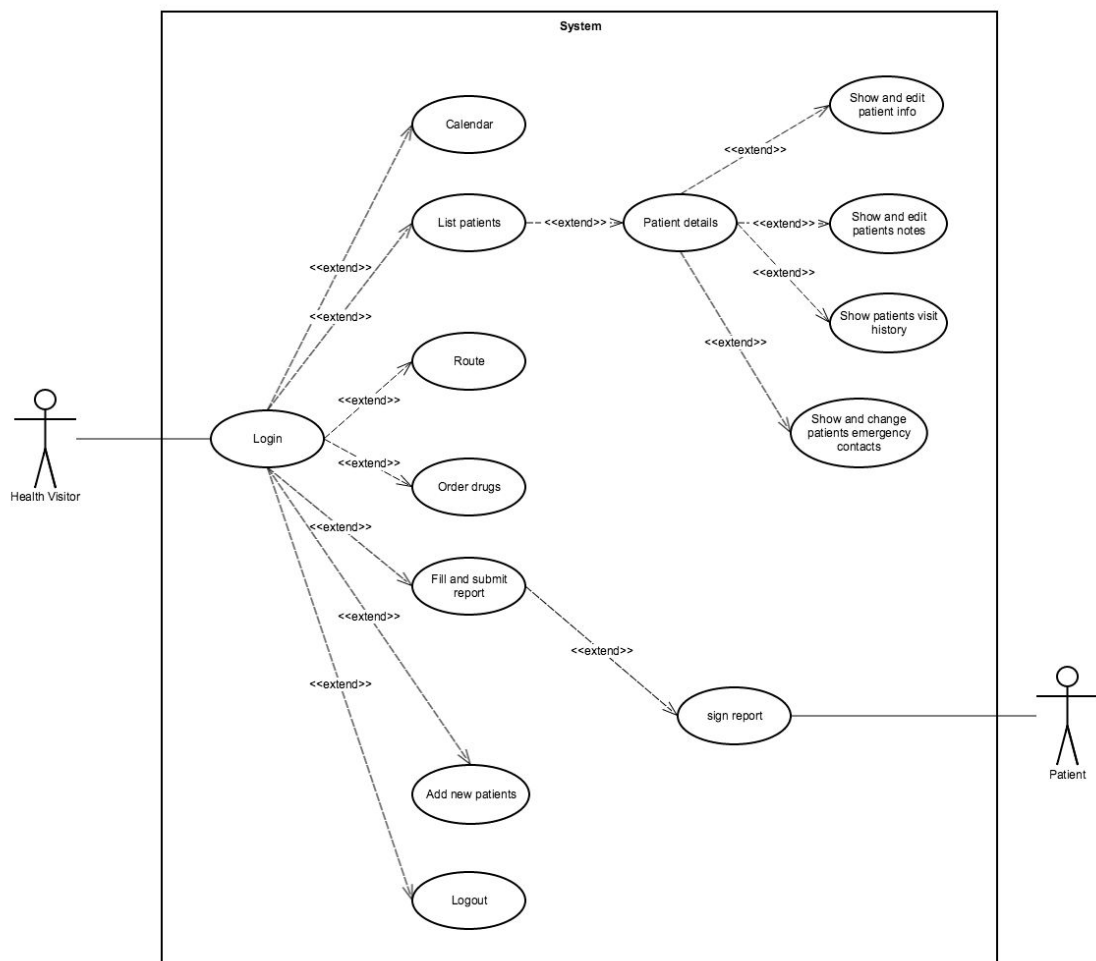
Health Visitor	Ein „Health Visitor“ ist eine Person im Public-Health-Bereich, welche Hausbesuche durchführen. Dazu gehören Hebammen und Krankenschwestern mit Zusatzausbildungen. Ihre Hauptaufgabe ist es, ihren Patienten ein hygienisches Verhalten beizubringen oder zu ermöglichen und ihr Wohlergehen sowie das ihrer Familie (insbesondere ihrer Kinder) sicherzustellen. Sie sind in ein grosses, interdisziplinäres System mit Ärzten, Krankenschwestern und ähnlichem eingebettet.

User requirements definition

Overview

- UC-1011 Log in
- UC-1012 Log out
- UC-1021 List patients
- UC-1022 Show and edit patient notes
- UC-1023 Show patients visits history
- UC-1024 Show and change patients emergency contacts
- UC-1025 Calendar
- UC-1026 Optimized path
- UC-1031 Fill and submit report
- UC-1032 Sign report
- UC-1033 Order drugs
- UC-1040 Add new patients

Use-Case Diagram



Use Cases

1. Patientendaten einsehen

ID	UC-1021
Name	Patientendaten einsehen
Szenario	Der Health Visitor möchte die Patientendaten einsehen.
Beschreibung	Aus der Patientenliste wird ein gewollter Patient ausgewählt. Mit dem Klick darauf werden die Patientendetails angezeigt.

Akteure	Health Visitor, System
Bedingung	Der Health Visitor muss mit seinen Credentials eingeloggt sein.
Resultat	Die Patientendaten werden angezeigt.

Ablauf

#	Akteur	Tätigkeit
1.1	Health Visitor	Klickt auf den Patientennamen.
1.2	System	Zeigt Patienteninformationen. Das Sytem ermöglicht weitere Auswahl zu Visits History, Notes, Contacts.

Alternativer Ablauf

#	Akteur	Tätigkeit
1.1	Health Visitor	Select "Notes" (UC-1022)
1.2	Health Visitor	Select "Visits History" (UC-1023)
1.3	Health Visitor	Select "Contacts" (UC-1024)

2. Notes einsehen/eintragen

ID	UC-1022
Name	Notes (Anmerkungen zum Besuch) einsehen/eintragen.
Szenario	Der Health Visitor möchte die Notes einsehen/eintragen.
Beschreibung	Der Health Visitor kann die bereits eingetragenen Notes anschauen und neue zum aktuellen Besuch hinzufügen.
Akteure	Health Visitor, System
Bedingung	Der Health Visitor muss mit seinen Credentials eingeloggt sein.
Resultat	Die Notes werden angezeigt.

Ablauf

#	Akteur	Tätigkeit
1.1	Health Visitor	Klickt auf das "Notes"-Icon.
1.2	System	Das System listet alle bisherigen Notes auf.

Alternativer Ablauf

#	Akteur	Tätigkeit
1.1	Health Visitor	Klickt auf "Neuer Eintrag"-Icon.
1.2	System	Zeigt mit einem Pop-Up Fenster ein Textfeld auf.
1.3	Health Visitor	Schreibt neuen "Notes"-Eintrag ein.

3. Visits History (Besuche) einsehen

ID	UC-1023
Name	Visits History einsehen.
Szenario	Der Health Visitor möchte die Visits History einsehen.
Beschreibung	Der Health Visitor kann die bereits eingetragenen Besuche anschauen.
Akteure	Health Visitor, System
Bedingung	Der Health Visitor muss mit seinen Credentials eingeloggt sein.
Resultat	Die Visits History wird angezeigt.

Ablauf

#	Akteur	Tätigkeit
1.1	Health Visitor	Klickt auf das "Visits-History"-Icon.
1.2	System	Das System listet alle bisherigen Besuche auf.

4. Contacts einsehen / editieren

ID	UC-1024
Name	Contacts einsehen/ editieren.
Szenario	Der Health Visitor möchte die Contacts einsehen/ editieren.
Beschreibung	Der Health Visitor kann die bereits eingetragenen Kontakte des Patienten anschauen und entsprechend editieren.
Akteure	Health Visitor, System
Bedingung	Der Health Visitor muss mit seinen Credentials eingeloggt sein.
Resultat	Die Contacts werden angezeigt.

Ablauf

#	Akteur	Tätigkeit
1.1	Health Visitor	Klickt auf das "Contacts"-Icon.
1.2	System	Das System listet alle Contacts des Patienten auf.

Alternativer Ablauf

#	Akteur	Tätigkeit
1.1	Health Visitor	Klickt auf das "Editieren"-Icon.
1.2	System	Zeigt die editierbare "Contacts"-Form an.
1.3	Health Visitor	Ändert die "Contacts"-Angaben und speichert diese mit dem Klick auf das "Save"-Button.

5. Calendar / my Appointments

ID	UC-1025
Name	Calendar / my Appointments
Szenario	Der Health Visitor möchte die Termine hinzufügen / aktualisieren / anzeigen
Beschreibung	The health visitor can create / edit / show / delete appointments in the calendar
Akteure	Health Visitor, System
Bedingung	Der Health Visitor muss mit seinen Credentials eingeloggt sein.
Resultat	The appointments list will be shown

Ablauf

#	Akteur	Tätigkeit
1.1	Health Visitor	Klickt das "Calendar"-Icon.
1.2	System	Zeigt eine Liste der Termine in einer Daily / Weekly / Monthly Ansicht
2.1	Health Visitor	Erstellt einen neuen Termin
2.2	System	Akzeptiert die Eingabe und speichert die Daten in der DB
3.1	Health Visitor	Bearbeitet / löscht einen Termin
3.2	System	Kontrolliert und akzeptiert die Eingabe und speichert die Daten in der DB. Wechselt zur Terminansicht
4.1	Health Visitor	Wählt einen Termin
4.2	System	Zeigt Termindetails

Ausnahmen

#	Akteur	Tätigkeit
2.2, 3.1	System	Input check fails: wrong date/time

6. Show optimized path

ID	UC-1026
Name	Build a path in the maps for today's Appointments
Szenario	Der Health Visitor möchte eine "Karte" der heutigen Termine anzeigen
Beschreibung	Der Health Visitor kann eine Karte anzeigen lassen, auf der die Termine des Tages so angezeigt werden, dass der Weg zwischen den verschiedenen Kunden möglichst kurz ist.
Akteure	Health Visitor, System
Bedingung	Der Health Visitor muss mit seinen Credentials eingeloggt sein.
Resultat	"Shortest path" für die Termine des Tages

Ablauf

#	Akteur	Tätigkeit
1.1	Health Visitor	Klick auf das "Build path"-Icon.
1.2	System	Das System berechnet den kürzesten Weg und zeigt die Kartenansicht

Ausnahmen

#	Akteur	Tätigkeit
1.1	System	No appointments in selected day
1.2	System	Google Maps Service ist nicht verfügbar (kein Internet / andere Verbindungsfehler)

6. Fill and submit report

ID	UC-1031
Name	Fill and submit report
Szenario	Rapport über geleistete Arbeit ausfüllen und abschicken.
Beschreibung	Ein Rapport über die geleistete Arbeit beim Patient kann nach der Behandlung / der Betreuung erfasst und abgeschlossen werden.
Akteure	Health Visitor, System
Bedingung	Der Health Visitor muss mit seinen Credentials eingeloggt sein.
Resultat	Der Rapport wurde erfolgreich eingereicht.

Ablauf

#	Akteur	Tätigkeit
1.1	Health Visitor	Auswahl der Rapport-Funktion im Menu.
1.2	System	Daten für den aktuellen Besuch werden geladen.
1.3	Health Visitor	Zeit und Tätigkeiten werden erfasst.

Ausnahmen

#	Akteur	Tätigkeit
1.1	System	Keine Internetverbindung - Rapport kann nicht abgeschickt werden.

7. Sign report

ID	UC-1032
Name	Sign report
Szenario	Den Rapport unterschreiben.

Beschreibung	Nachdem ein Rapport erfasst wurde, muss dieser vom Patient noch unterschrieben werden.
Akteure	Patient, System
Bedingung	<ul style="list-style-type: none"> - Der Health Visitor muss mit seinen Credentials eingeloggt sein. - Der Rapport muss vom Health Visitor erfasst sein.
Resultat	Der Rapport wurde erfolgreich vom Patienten unterschrieben.

Ablauf

#	Akteur	Tätigkeit
1.1	Health Visitor	Rapport wird Ausgefüllt (UC-1031).
1.2	System	Zusammenfassung des Rapports wird angezeigt.
1.3	Patient	Patient visiert / signiert den Rapport.

Ausnahmen

#	Akteur	Tätigkeit
1.1	System	Keine Internetverbindung - Unterschrift/Visum muss zwischengespeichert werden und später gesendet werden.

8. Order drugs

ID	1033
Name	Order drugs
Szenario	Medikamente für den Patienten online bestellen.
Beschreibung	Medikamente können automatisch bestellt werden.
Akteure	Health Visitor, System
Bedingung	<ul style="list-style-type: none"> - Der Health Visitor muss mit seinen Credentials eingeloggt sein. - Die Medikamente und Menge müssen vorgängig erfasst sein.
Resultat	Die Medikamente wurden dem Patienten nach Hause geliefert.

Ablauf

#	Akteur	Tätigkeit
1.1	Health Visitor	Öffnet Menüpunkt Medikamentenbestellung.
1.2	System	Überblick über Medikamente wird angezeigt.
1.3	Health Visitor	Medikamente auswählen und Bestellung auslösen.
1.4	System	Email mit Bestellung an Apotheke senden.

Ausnahmen

#	Akteur	Tätigkeit
1.1	System	Keine Internetverbindung - Bestellung temporär speichern und zu einem späteren Zeitpunkt erneut übermitteln.

Add new patients

ID	1040
Name	Add new patients
Szenario	Neue Patienten im System erfassen.
Beschreibung	Neue Patienten sollen im System erfasst und einem anderen Health Visitor auch zugewiesen werden können.
Akteure	Health Visitor, System
Bedingung	- Der Health Visitor muss mit seinen Credentials eingeloggt sein.
Resultat	Ein neuer Patient wurde im Sytem erfasst.

Ablauf

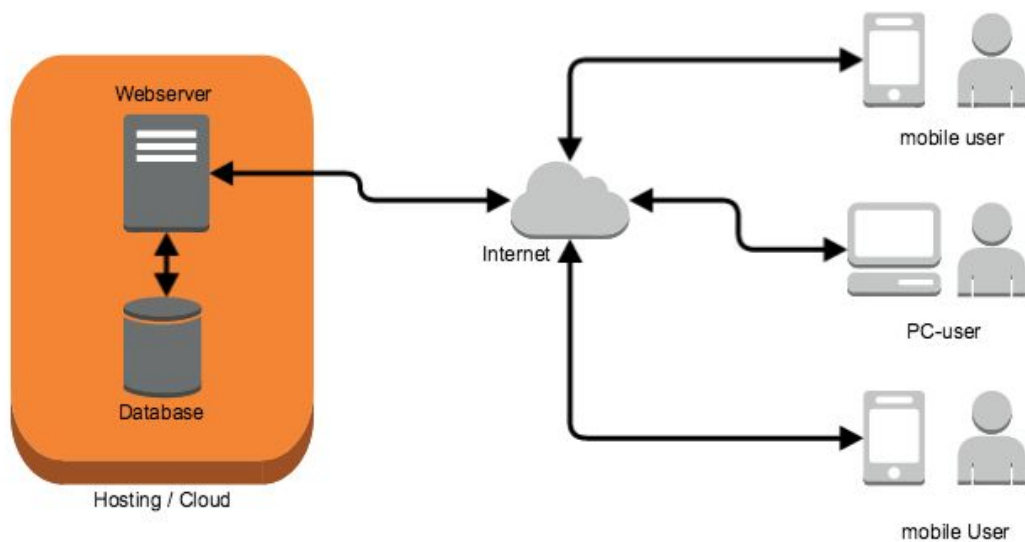
#	Akteur	Tätigkeit
1.1	Health Visitor	Öffnet Menüpunkt "neuer Patient erfassen".

1.2	System	Maske um neuen Patient zu erfassen wird angezeigt.
1.3	Health Visitor	Patient erfassen.
1.4	System	Überprüft, ob Patient bereits im System erfasst ist.
1.5	Health Visitor	Patient einem Health Visitor zuweisen.

Ausnahmen

#	Akteur	Tätigkeit
1.1	System	Patient ist bereits im System erfasst.

System Architektur



Die Benutzer greifen über ein Web-Interface (Rest-API) auf die Applikation zu. Die Webapplikation läuft auf einem Webserver. Die Infrastruktur umfasst neben dem Webserver auch eine Datenbank.

System requirements specification

Funktionale Anforderungen

- Jeder Health Visitor erhält ein persönliches Login (kein Login, kein Zugriff)
- Die Shortcut-Icons steuern den Weg des Users

- Der Rapport muss nach jedem Besuch bei einem Kunden gesendet werden
- Die App soll intuitiv zu bedienen sein, so dass der User den Weg innerhalb der App mit möglichst wenig Aufwand findet.

Nicht-funktionale Anforderungen

- Service availability: das System sollte 24/7 verfügbar sein - mit Ausnahme von kurzen Wartungsfenstern, welche im Voraus angekündigt werden.
- Kapazität / Scaling: das System sollte mit einer grossen Menge an Usern klarkommen
- Security: die Benutzerdaten sollten nur für berechtigte Personen zugänglich sein
- Das System sollte auf allen Smartphones und Tablets laufen, die nach 2012 produziert wurden
- Das System sollte möglichst wenig Daten mit den Clients austauschen, damit der Health Visitor auch mit einer schlechten Internetverbindung arbeiten kann
- Das System soll den Health Visitors das Leben vereinfachen (nicht komplizierter...)

System models

Datenbank → Input → **Schnittstelle (DB)** → Input → **Datenaufbereitung (SRV)** → Input → **Darstellung (GUI)** → Output → **Health-Visitor**

System Evolution

Da die Software langfristig in Betrieb sein soll, wird die Software und das System so modular aufgebaut, dass Änderungen und Erweiterungen in Zukunft leicht möglich umzusetzen sind. In einer ersten Phase wird die Software nur auf einem System installiert, aber es werden bereits zu Beginn Vorkehrungen getroffen, die ein späteres Skalieren leicht umsetzbar machen. Auch müssen Versionsupdates der Datenbank und des Webserver sowie aber auch Hardware Veränderungen jederzeit möglich sein.

Testing

- **Functional testing:** testen der spezifischen Codefunktionen. Es soll folgende Frage beantwortet werden: "Kann der Benutzer eine bestimmte Funktion des Programms verwenden?"
- **Usability testing:** versteht der Benutzer die Benutzeroberfläche und ist diese für den Benutzer einfach zu bedienen?
- **Security testing:** testet, ob das System gegen unberechtigten Zugriff geschützt ist. Das System soll verhindern, dass unberechtigte Benutzer Daten lesen und/oder modifizieren.

Anhang

Unsere Applikation ist als eine Web-Anwendung realisiert und besteht aus den folgenden Teilen:

- Data Storage (Datenbank)
- Server-Side layer
- Client-Side layer

1. Data Storage

Die Applikation speichert die Daten in einer SQL-Datenbank ab. Die Datenbank befindet sich auf der AzureCloud und ist 32MB gross (optimal für unser Projekt).

2. Server-Side layer

Diese Ebene regelt, wie die Daten erzeugt, editiert, abgespeichert und gelöscht werden können.

Die Business-Logik wird anhand eines MVC-Frameworks implementiert und beinhaltet:

- Data access layer
- REST API methods

Anschliessend wird die Applikation auf der AzureCloud in einem Tomcat-Container publiziert

3. Client-Side layer

Diese Ebene implementiert die graphische Darstellung auf dem Client (web-browser).

Bei der Entwicklung werden folgende Technologien verwendet:

- HTML5 & Bootstrap
- AngularJS

Index