

Patientenapp

Requirements Specifications

**Autoren**

Thomas Buchegger  
Stefan Tanner  
Markus Loosli  
Filip Hofer  
Lukas Knöpfel  
James Mayr

Inhalt

[Zweck des Dokument 4](#_Toc414983643)

[Versionsübersicht 4](#_Toc414983644)

[Einführung 5](#_Toc414983645)

[Systemfunktionen 5](#_Toc414983646)

[Einfache Navigation 5](#_Toc414983647)

[Medikation 6](#_Toc414983648)

[Terminverwaltung 7](#_Toc414983649)

[Behandlungssystem „Life up“ 8](#_Toc414983650)

[Notfall 10](#_Toc414983651)

[Out of Scope 11](#_Toc414983652)

[Kommunikation mit anderen Systemen 12](#_Toc414983653)

[Benutzeranforderungen 13](#_Toc414983654)

[Benutzeranforderungen Übersicht 13](#_Toc414983655)

[Use Case Diagramme 14](#_Toc414983656)

[Use Case 1: In „Life up“ einen geforderten Ort erreichen 15](#_Toc414983657)

[Use Case 2: In Life up ein gefordertes Objekt fotografieren 16](#_Toc414983658)

[Use Case 3: An Medikamenteinnahme erinnern 17](#_Toc414983659)

[Activity Diagramm 18](#_Toc414983660)

[Systemarchitektur / System Modelle 19](#_Toc414983661)

[Information 19](#_Toc414983662)

[Legende 19](#_Toc414983663)

[Diagramme 20](#_Toc414983664)

[1.0 Patientenapp 22](#_Toc414983665)

[1.0.1 Kontaktmodul 23](#_Toc414983666)

[1.0.2 Terminmodul 24](#_Toc414983667)

[1.0.3 Kommunikationsmodul 25](#_Toc414983668)

[1.0.4 Notfallmodul 26](#_Toc414983669)

[1.0.5 Medikationsmodul 27](#_Toc414983670)

[1.0.6 Life up 28](#_Toc414983671)

[Systemanforderungen 29](#_Toc414983672)

[Functional Requirements 29](#_Toc414983673)

[User Requirements 29](#_Toc414983674)

[System Requirements 29](#_Toc414983675)

[Non-Functional Requirements 29](#_Toc414983676)

[Product Requirements 29](#_Toc414983677)

[Organizational Requirements 29](#_Toc414983678)

[Domain Specific Requirements 29](#_Toc414983679)

[System evolution 30](#_Toc414983680)

[Version 0.1 - Homebereich, GUI 30](#_Toc414983681)

[Version 0.2 - Kalendermodul 30](#_Toc414983682)

[Version 0.3 - Erinnerungsfunktion 30](#_Toc414983683)

[Version 0.4 - Medikationsmodul 30](#_Toc414983684)

[Version 0.5 - Rückfall-Notfallmodul 30](#_Toc414983685)

[Version 0.6 - „life-up“-Modul, Grundsystem 30](#_Toc414983686)

[Version 0.7 - „life-up“-Modul, Level-Up 30](#_Toc414983687)

[Version 0.8 - Fehlerbehebungen 30](#_Toc414983688)

[Version 1 - Erstes Rollout 30](#_Toc414983689)

[Testkonzept 31](#_Toc414983690)

[Unittests 31](#_Toc414983691)

[Systemtests 31](#_Toc414983692)

[Benutzertests 32](#_Toc414983693)

[Appendices 33](#_Toc414983694)

[Hardwareanforderungen 33](#_Toc414983695)

[Datenbanklayout 34](#_Toc414983696)

[Glossar 35](#_Toc414983697)

[Quellenverzeichnis 35](#_Toc414983698)

[Abbildungsverzeichnis 36](#_Toc414983699)

[Tabellenverzeichnis 36](#_Toc414983700)

# Zweck des Dokument

Dieses Dokument zeigt und beschreibt

* die Systemgrenzen des Projekts “Health Care”;
* die Schnittstellen zu Fremdsystemen;
* die Systemweiterentwicklung / Erweiterbarkeit;
* die Systemarchitektur;
* wie das System zu testen ist.

## Versionsübersicht

Tabelle 1 Versionsübersicht des Dokuments

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Datum** | **Änderungen** | **Kürzel** |
| 1.0 | 17.03.2015 | Dokumentvorlage erstellt. | JM |
| 1.1 | 19.03.2015 | Kapitel „Preface“ und „Einführung“ hinzugefügt. | JM |
| 1.2 | 19.03.2015 | Kapitel „Einführung“ erweitert. | JM |
| 1.3 | 19.03.2015 | Kapitel „Kommunikation mit anderen System“ hinzugefügt. | ST |
| 1.4 | 21.03.2015 | Kapitel „Systemarchitektur“ hinzugefügt. | ST |
| 1.5 | 21.03.2015 | Kapitel “Functional Requirements” und “non Functional Requirements” hinzugefügt. | FH |
| 1.6 | 21.03.2015 | Kapitel “System evolution” hinzugefügt. | ML |
| 1.7 | 22.03.2015 | Kapitel „Out Of Scope“ hinzugefügt, Anforderungen überarbeitet. | JM |
| 1.8 | 23.03.2015 | Rechtschreibung korrigiert und Formatvorlage auf „CI“ angepasst. | ML |
| 1.9 | 23.03.2015 | Appendencies ergänzt | JM |

# Einführung

Das System soll Suchtpatienten nach einer stationären Behandlung unterstützen und ihnen dabei helfen zurück in ein geregeltes Leben zu finden.   
Damit die Patienten zu einem selbstständigen Leben zurück finden erinnert die Patientenapp an anstehende Termine. Zusätzlich liefert die App einen Überblick über die Medikation. Für Notfallsituationen beinhaltet die App Notfallfunktionen wie einen Notfallplan und eine Anruffunktion.

## Systemfunktionen

Das System besteht grob aus folgenden Teilbereichen:

* Terminplanung
* Medikation
* Behandlungssystem “Life up”
* Notfallfunktionen

### Einfache Navigation

Die App soll einfach und übersichtlich sein. Auf dem Hauptbildschirm ist markan, im oberen Teil des Bildschirms, die nächste Tätigkeit aufgeführt. Bei der Tätigkeit kann es sich um ein „Life up“-Event, einen anstehenden Termin oder an eine Erinnerung zur Medikament Einnahme handeln. Mit einem Knopfdruck erreicht der Patient alle Bereiche der App.



Abbildung 1 Die App ist übersichtlich und einfach strukturiert

### Medikation

Die App erinnert den Patienten an die Medikamenteneinnahme und bietet den Überblick über die Medikation.

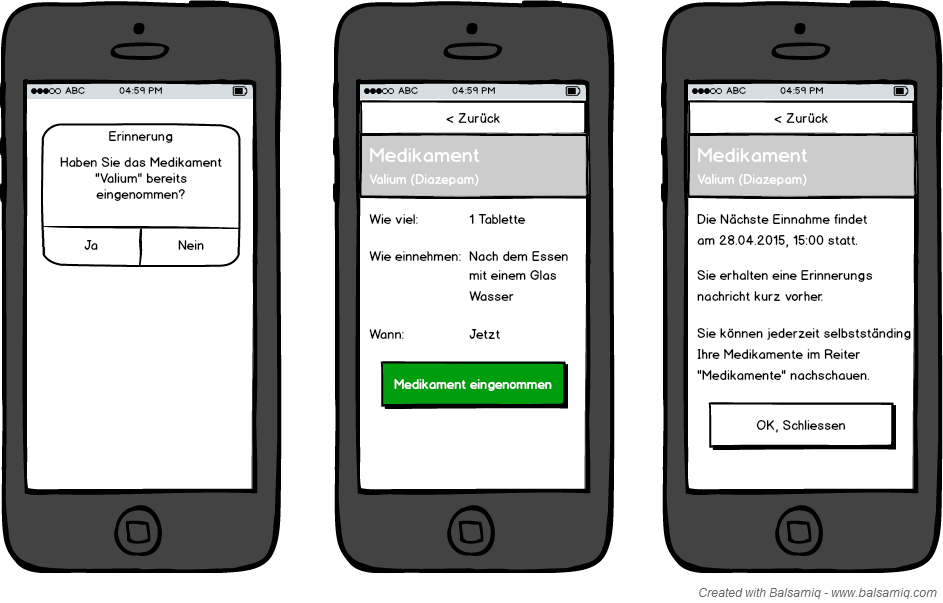


Abbildung 2 Der Patient wird an die Medikamenteneinnahme erinnert.

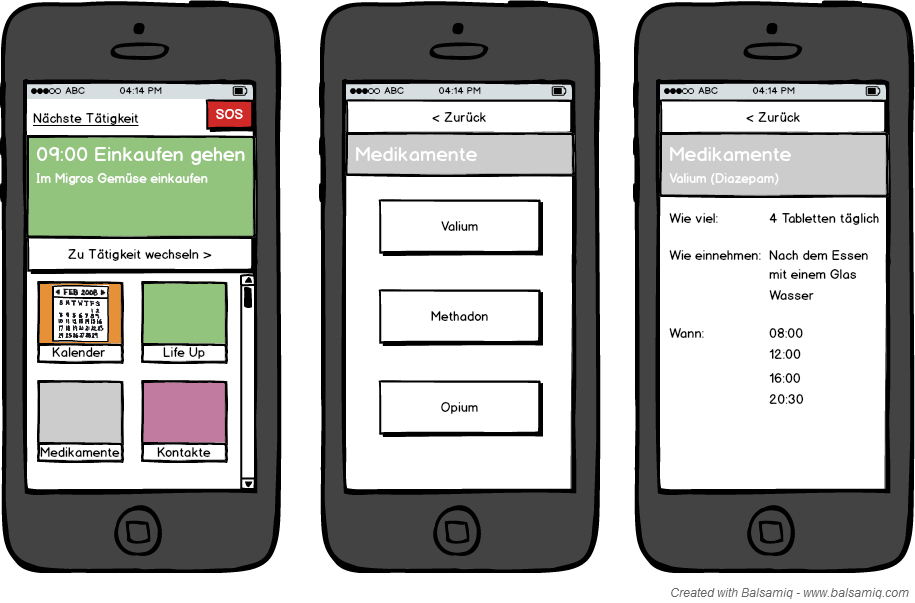


Abbildung 3 Der Patient behällt den Überblick über die Medikation

### Terminverwaltung

Damit der Patient alle anstehenden Termine wahrnehmen kann, listet die alle Termine im Terminmodul auf. Zusätzlich erinnert die App den Patienten frühzeitig an die anstehenden Termine.

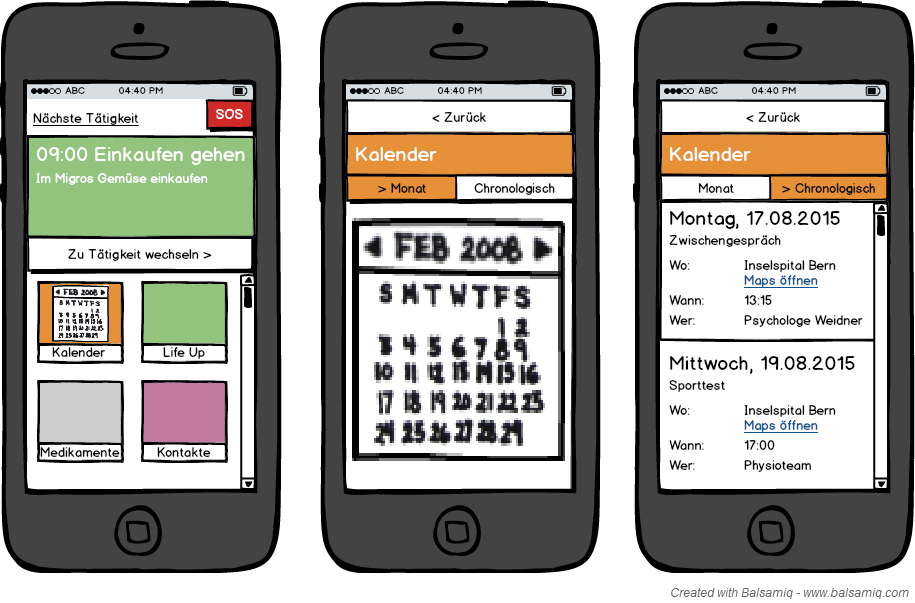


Abbildung 4 Mit der App hat der Patient seine Termine immer im Überblick

### Behandlungssystem „Life up“

Das zentrale Element der Patientenapp ist das Behandlungssystem „Life up“. Ziel dieses Systems ist es, den Patienten zu einem strukturierten Tagesablauf zu motivieren. Erreicht werden soll dies durch spielerisch Aufgaben wie z. B. Briefkasten leeren, im Migros einkaufen. Sobald der Patient diese Aufgaben erfüllt erhält er dafür Erfahrungspunkte.

#### Der grobe Ablauf von „Life up“

* Der Patient erhält täglich einige Aufgaben (z. B. ins Migros einkaufen gehen);
* für das Erledigen der Aufgabe hat er ein begrenztes Zeitfenster;
* er schliesst seine Aufgaben erfolgreich ab indem er z. B. einen Code oder ein Objekt fotografiert oder seinen Standort mittels GPS-Empfänger aufzeichnet.
* Für jede erfolgreich abgeschlossene Aufgabe erhält er Erfahrungspunkte;
* Sobald er genügend Erfahrungspunkte gesammelt, wird er mit einem Level up belohnt;
* Um eine zusätzlich Motivation zu bieten kann er mit z. B. mit Gutscheinen belohnt werden.

#### Vorteile für Patient und Einrichtung

Die Erfolge sind für den Patienten sowie für die Einrichtung/Klinik messbar. Der Patient wird spielerisch zum Alltag zurück geführt.



Abbildung 5 Der Patient wird spielerisch zum Alltag zurück geführt.

### Notfall

Leider läuft nicht immer alles Rund und der Patienten droht Rückfällg zu werden. Üblicherweise wird der Behandlung mit dem Patienten ein Notfallplan ausgearbeitet.

Die Patientenapp bietet die Möglichkeit diesen Notfallplan anzuzeigen. Der Patient kann den Plan Schritt für Schritt abarbeiten. Ein solcher Schritt kann zum Beispiel ein Anruf bei Angehörigen oder Freunden sein. Falls der Schritt nicht wirkt, kann der Patient direkt aus der App die Notfallstation seiner Klinik anrufen.

Die Notfallfunktion ist direkt über das Hauptmenü der App erreichbar.

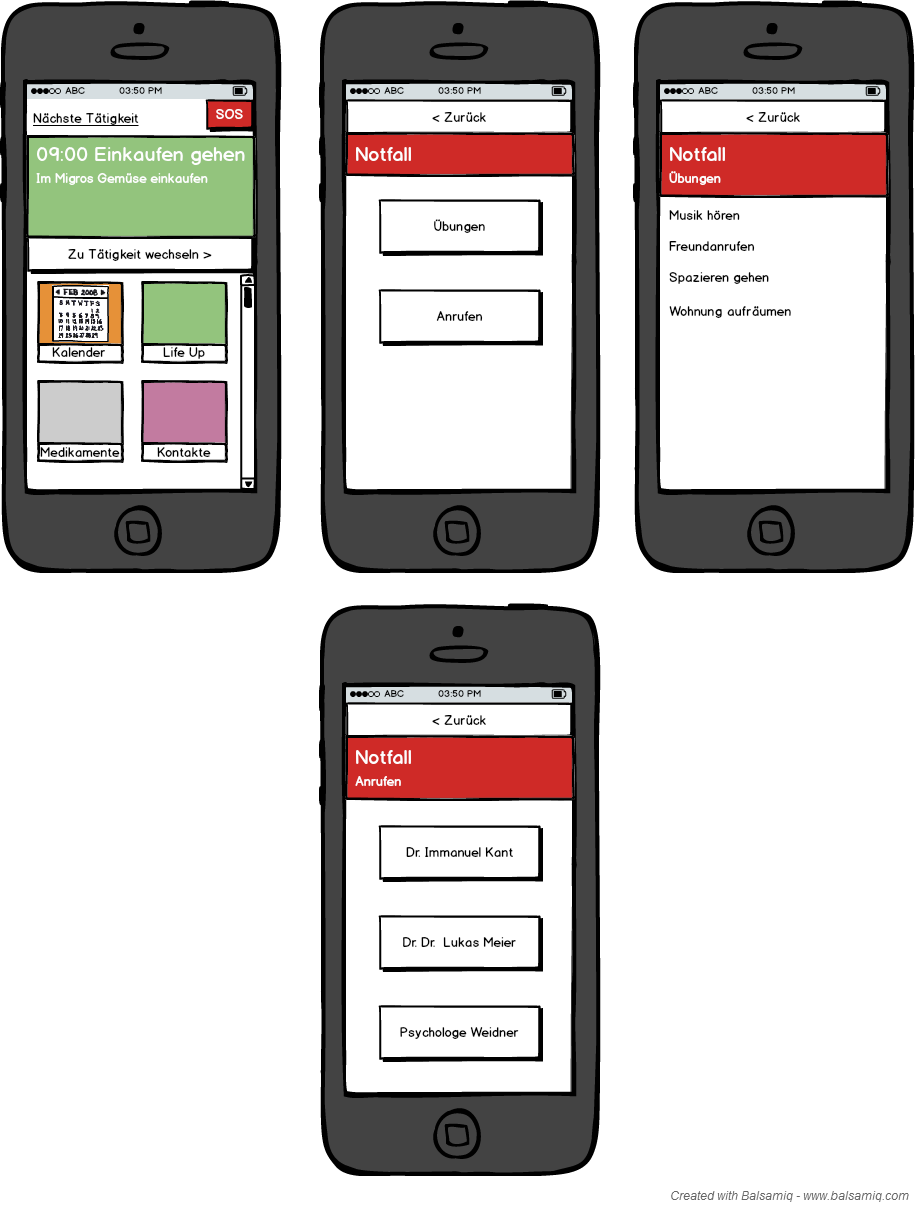


Abbildung 6 Die App unterstützt den Patienten wenn er droht Rückfällig zu werden

## Out of Scope

Out of Scope sind alle Funktionen die der Klinik erlauben die App zu konfigurieren sowie die Daten der App zu aktualisieren wie zum Beispiel:

* Erfassen der Termine / Medikamente / Medikation;
* Neue Tagesaufgaben erfassen;
* Auswerten der Patientendaten;
* Erfassung der Notfallnummer.
* Datenbereitstellung der Klinik

## Kommunikation mit anderen Systemen

Das System kommuniziert mit folgenden Fremdsystemen:

* GPS-Sensor des Smartphones zur Standortbestimmung;
* Telefon zum Anrufen;
* Kamera des Smartphones zur Erfassung von Objekten;
* QR Code App des Smartphones;
* Verwaltungssystem der jeweiligen Klinik.

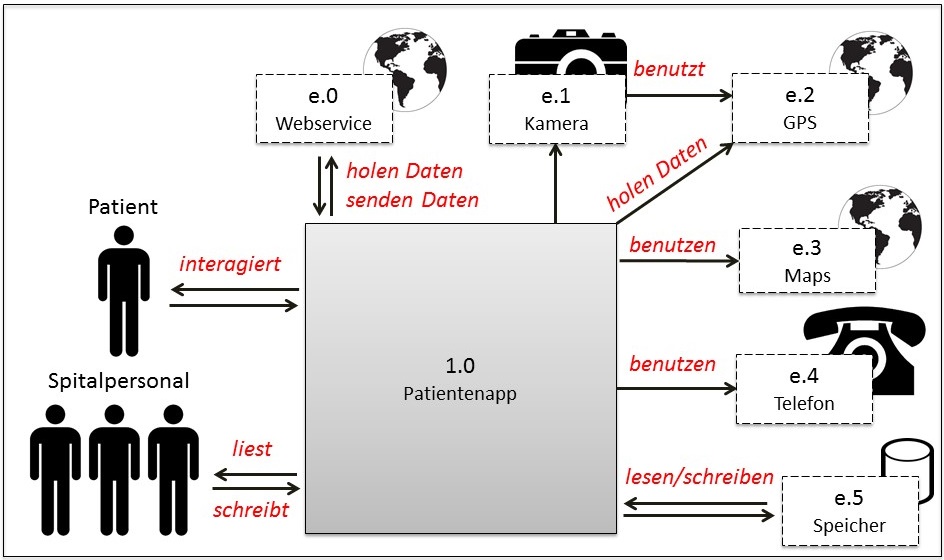


Abbildung 7 Überblick der Kommunikation mit Fremdsystemen

# Benutzeranforderungen

In diesem Kapitel sind die Benutzeranforderungen aufgelistet und die Use Cases definiert respektive modelliert. Zusätzlich ist in diesem Kapitel ein Activity Diagramm, das den Ablauf des Life Up Behandlungssystem besser beschreibt.

## Benutzeranforderungen Übersicht

Folgende Anforderungen werden an das System gestellt:

**Allgemein**

* Authentifizierung / Autorisierung
* Hauptmenü anzeigen
* Kurzinfo anzeigen (nächste Tätigkeit)

**Behandlungssystem „Life up“**

* Tagesablauf anzeigen
* Tagesablauf erinnern
* Fortschritt anzeigen
* *Einen geforderten Ort erreichten*
* *Einen geforderten Objekt fotografieren*
* Erfolgreiche Aufgaben aufzeichnen

**Medikation**

* *Medikamenterinnerung*
* Medikamentübersicht anzeigen
* Detailliertes Medikament anzeigen

**Termine**

* Terminübersicht anzeigen
* Details zu Termin anzeigen
* Terminerinnerung

**Notfall**

* Kontaktliste anzeigen
* Kontaktperson anrufen
* Notfallstelle anrufen
* Notfallplan anzeigen

## Use Case Diagramme

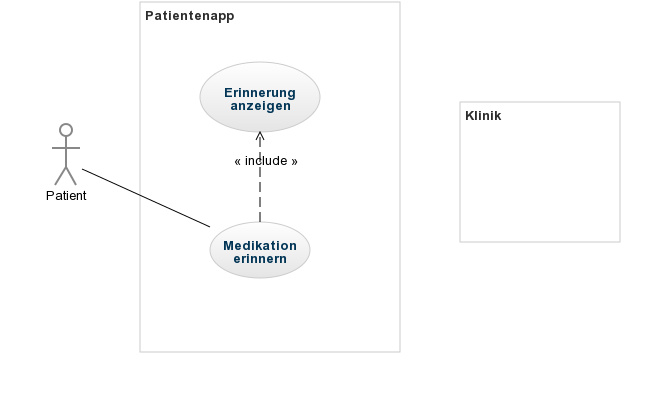


Abbildung 8 Use Case Diagramm Medikation

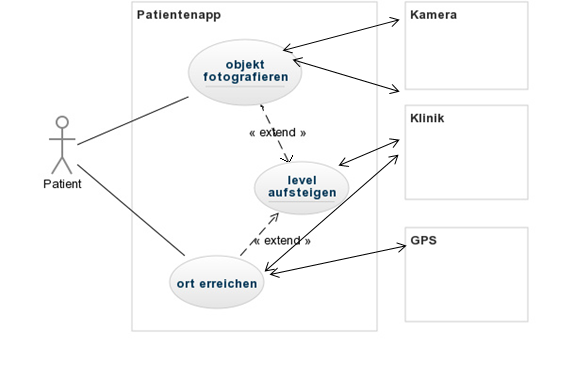


Abbildung 9 Use Case Diagramm “Life up”

## Use Case 1: In „Life up“ einen geforderten Ort erreichen

Tabelle 2 Beschreibung Use Case 1

|  |  |  |  |
| --- | --- | --- | --- |
| **Nr und Name** | | 1 in Life up einen geforderten Ort erreichen | |
| **Szenario** | | Der Patient erreicht einen von der täglichen Aufgabe geforderten Ort. | |
| **Kurzbeschreibung** | | Der Patient erhält Erfahrungspunkte für das erfolgreiche erreichen eines von der täglichen Aufgabe geforderten Ortes. | |
| **Beteiligte Akteure** | | Patient, System, Fremdsystem: Datenbank der Klinik | |
| **Auslöser/Vorbedingungen** | | * Smartphones des Benutzers ist eingeschalten * Der Patient hat den geforderten Ort erreicht. * Maximalfrist für das Erfüllen der täglichen Aufgabe ist noch nicht abgelaufen. | |
| **Ergebnisse/Nachbedingungen** | | * Der Patient erhält Erfahrungspunkte für den Erfolg. * Use Case „Level aufstieg“ wenn genügend Erfahrungspunkte gesammelt. | |
| **Ablauf** | | | |
| **Nr.** | **Wer** | | **Was** |
| 1. | Patient | | Erreicht geforderten Ort bevor die zeit abgelaufen ist. |
| 2. | System | | System überprüft alle Bedingungen (hat der Patienten den Ort erreicht, ist die Zeit nicht überschritten worden) |
| 3. | Fremdsystem: Datenbank | | Daten werden im Fremdsystem gespeichert |
| 4. | System | | Zeigt dem Patienten an, dass die Aufgabe erfolgreich abgeschlossen ist. |
| 5. | System | | Das System vergütet dem Patienten Erfahrungspunkte für das erfüllen der Aufgabe. Die Erfahrungspunkteanzeige wird angepasst. |
| **Ausnahmen, Varianten** | | | |
| **Nr.** | **Wer** | | **Was** |
| 1.1 | Patient | | Erreicht geforderten Ort nachdem die Zeit abgelaufen ist |
| 2.1 | System | | Überprüft alle Bedingungen |
| 3.1 | Fremdsystem: Datenbank | | Daten werden im Fremdsystem gespeichert |
| 4.1 | System | | Zeigt dem Patienten, dass die Aufgabe teilweise erfolgreich abgeschlossen ist. |
| 5.1 | System | | Das System vergütet dem Patienten ein reduzierte Anzahl Erfahrungspunkte für das teilweise erfüllen der Aufgabe. Die Erfahrungspunkteanzeige wird angepasst. |
| 2.2 | System | | System bemerkt, dass der Patient eine neue Stufe erreicht hat. |
| 3.2 | Fremdsystem: Datenbank | | Daten werden im Fremdsystem gespeichert |
| 4.2 | System | | Zeigt dem Patienten an, dass die Aufgabe erfolgreich abgeschlossen ist. |
| 5.2 | System | | Das System vergütet dem Patienten Erfahrungspunkte für das erfüllen der Aufgabe. |
| 6.2 | System | | Use Case „ein Level in Life up aufsteigen“ wird ausgeführt. |

## Use Case 2: In Life up ein gefordertes Objekt fotografieren

Tabelle 3 Beschreibung Use Case 2

|  |  |  |  |
| --- | --- | --- | --- |
| **Nr und Name** | | 2 in Life up einen geforderten Objekt fotografieren | |
| **Szenario** | | Der Patient fotografiert einen von der täglichen Aufgabe geforderten Objekt. | |
| **Kurzbeschreibung** | | Der Patient erhält Erfahrungspunkte für das erfolgreiche fotografieren eines von der täglichen Aufgabe geforderten Objekts. | |
| **Beteiligte Akteure** | | Patient, System, Fremdsystem: Datenbank der Klinik, Kamerasystem | |
| **Auslöser/Vorbedingungen** | | * Smartphones des Benutzers ist eingeschalten * Kamerasystem des Benutzer ist funktionstüchtig. * Der Patient hat das geforderte Objekt fotografiert. * Maximalfrist für das Erfüllen der täglichen Aufgabe ist noch nicht abgelaufen. | |
| **Ergebnisse/Nachbedingungen** | | * Der Patient erhält Erfahrungspunkte für den Erfolg. * Use Case „Level aufstieg“ wenn genügend Erfahrungspunkte gesammelt. | |
| **Ablauf** | | | |
| **Nr.** | **Wer** | | **Was** |
| 1. | Patient | | Fotografiert den geforderten Objekt bevor die Zeit abgelaufen ist. |
| 2. | System | | System überprüft alle Bedingungen (hat der Patienten den geforderten Objekt fotografiert, ist die Zeit nicht überschritten worden) |
| 3. | Fremdsystem: Datenbank | | Daten werden im Fremdsystem gespeichert |
| 4. | System | | Zeigt dem Patienten an, dass die Aufgabe erfolgreich abgeschlossen ist. |
| **Ausnahmen, Varianten** | | | |
| 5. | System | | Das System vergütet dem Patienten Erfahrungspunkte für das erfüllen der Aufgabe. Die Erfahrungspunkteanzeige wird angepasst. |
| **Nr.** | **Wer** | | **Was** |
| 1.1 | Patient | | Fotografiert geforderten Objekt Ort nachdem die Zeit abgelaufen ist |
| 2.1 | System | | Überprüft alle Bedingungen |
| 3.1 | Fremdsystem: Datenbank | | Daten werden im Fremdsystem gespeichert |
| 4.1 | System | | Zeigt dem Patienten, dass die Aufgabe teilweise erfolgreich abgeschlossen ist. |
| 5.1 | System | | Das System vergütet dem Patienten ein reduzierte Anzahl Erfahrungspunkte für das teilweise erfüllen der Aufgabe. Die Erfahrungspunkteanzeige wird angepasst. |
| 2.2 | System | | System bemerkt, dass der Patient eine neue Stufe erreicht hat. |
| 3.2 | Fremdsystem: Datenbank | | Daten werden im Fremdsystem gespeichert |
| 4.2 | System | | Zeigt dem Patienten an, dass die Aufgabe erfolgreich abgeschlossen ist. |

|  |  |  |
| --- | --- | --- |
| 5.2 | System | Das System vergütet dem Patienten Erfahrungspunkte für das erfüllen der Aufgabe. |
| 6.2 | System | Use Case „15 ein Level in Life up aufsteigen“ wird ausgeführt. |

## Use Case 3: An Medikamenteinnahme erinnern

Tabelle 4 Beschreibung Use Case 3

|  |  |  |  |
| --- | --- | --- | --- |
| **Nr und Name** | | 3 Medikation erinnern | |
| **Szenario** | | Ist genügend Zeit seit der letzten Medikamenteneinnahme vergangen, so wird der Benutzer an eine neue Einnahme erinnert | |
| **Kurzbeschreibung** | | Es erscheint eine Meldung auf dem Display welche den Benutzer an eine erneute Einnahme der Medikamente erinnert | |
| **Beteiligte Akteure** | | System, Patient, Fremdsystem: Datenbank der Klinik | |
| **Auslöser/Vorbedingungen** | | Die Zeit für eine neue Einnahme ist um | |
| **Ergebnisse/Nachbedingungen** | | Der Patient hat seine Medikamente eingenommen | |
| **Ablauf** | | | |
| **Nr.** | **Wer** | | **Was** |
| 1. | System | | Holt aus der Datenbank den Zeitpunkt der letzten Einnahme |
| 2. | System | | Timer läuft ab |
| 3. | System | | Auf dem Bildschirm erscheint eine Meldung die den Patienten daran erinnert, das Medikament wieder einzunehmen |
| 4. | Patient | | Nimmt das Medikament |
| 5. | System | | Anzeige von Ja/Nein Button |
| 6. | Datenbank | | Schreibt den aktuellen Zeitpunkt in die Datenbank |
| 7. | System | | Timer wird bei der Einnahme des Medikaments neu gestartet |
| **Ausnahmen, Varianten** | | | |
| **Nr.** | **Wer** | | **Was** |
| 5.1 | System | | Wählt der Benutzer „Ja“, so wird ein neuer Timer für die nächste Einnahme gestartet |
| 5.2 | System | | Wählt der Benutzer „Nein“, so wird ein Timer für die nächste Erinnerung gestartet |

## Activity Diagramm

Folgendes Diagramm zeigt den Ablauf, wenn der Patient ein gefordertes Objekt einer Aufgabe im „LIfe up“-Behandlungssystems anklickt.

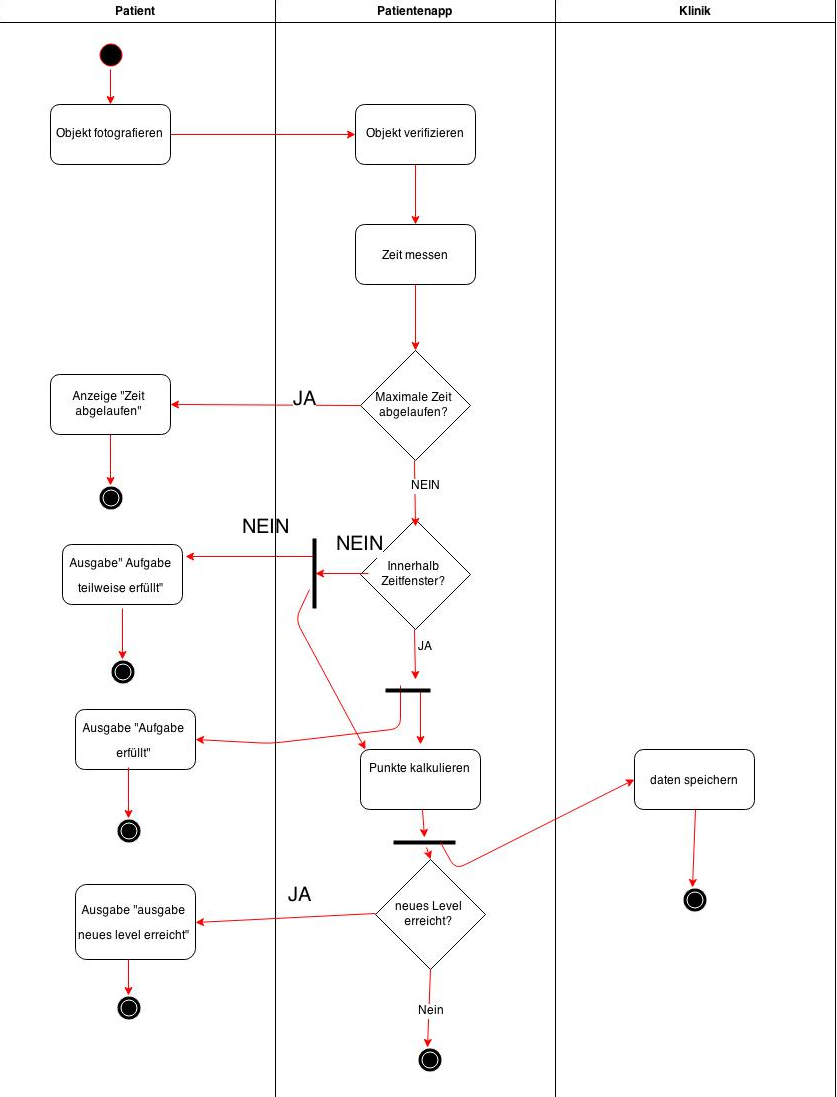


Abbildung 10 Activity Diagramm „in Life up ein gefordertes Objekt fotografieren“

# Systemarchitektur / System Modelle

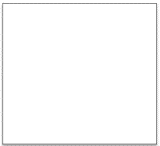
## Information

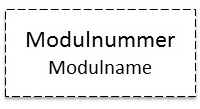
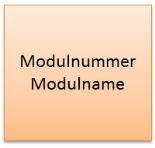
Das System ist in einzelne Untersysteme, sogenannte Module, unterteilt. Dadurch wird das System übersichtlicher und verständlicher. Die Diagramme wurden nicht nach einem spezifischen Standard erstellt. Ziel ist es, die Hauptbestandteile des Systems aufzuzeigen und zu veranschaulichen. Weiter sollen die Diagramme in der Systemarchitektur als Idee/Vorlage für eine technische Umsetzung dienen.

## Legende

Grundlegende Komponenten des Diagramms:

**Modul**   
Ein Modul ist ein grau-hinterlegtes Viereck. Die Modulbezeichnung besteht aus einer Nummer und einem Namen. Der Name befindet sich unterhalb der Modulnummer.

**Rahmen**Ein leeres Viereck fasst Komponenten zusammen. Sie zum jeweiligen Modul.

**Externes Modul**  
Externe Module sind Module, die nicht teil der Patientenapp sind. Der Nummer von externen Modulen ist das Präfix „e“ vorangestellt.

**Spezielles Modul**Spezielle Module weichen farblich ab.

**Beziehung**Die Abhängigkeiten und Beziehungen zwischen Modulen werden mit einem beschrifteten Pfeil erläutert, dargestellt. Die Abhängigkeit ist in Pfeilrichtung.

**Benutzer / Benutzergruppen**Benutzer und Benutzergruppen werden mit einem Piktogramm dargestellt. Sie können auch in Beziehungen zu Modulen stehen.

**Weiteres**Zusätzliche Bilder bei Modulen dienen lediglich zur schnelleren Auffassung.

## Diagramme

1 Systemkontext

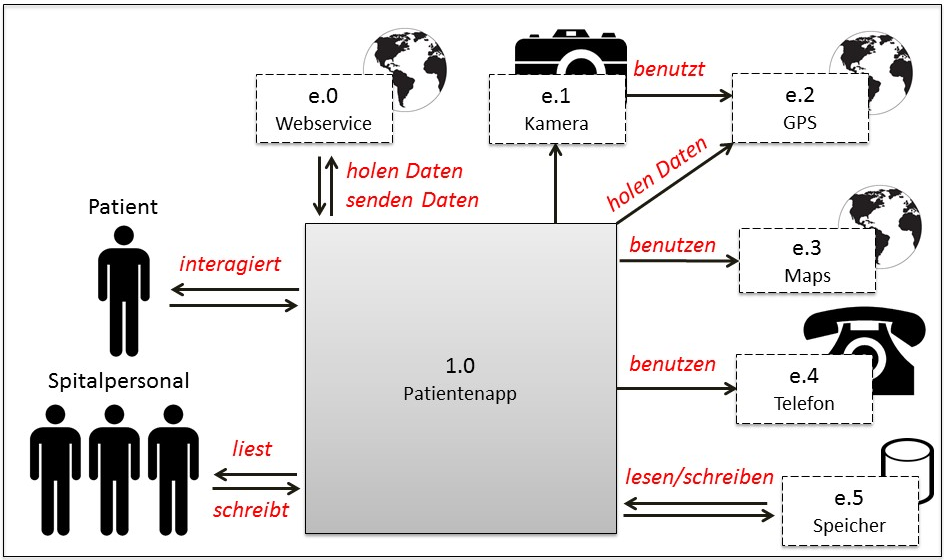


Abbildung 11 Systemarchitketur Systemkontext

**Erläuterung**

Der Systemkontext besteht im Wesentlichen aus drei Teilen. Den Benutzern, den systemfremden Modulen und der Patientenapp selbst.

Tabelle 5 Erläuterung Systemarchitektur

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ModulNr.** | **Modul / Benutzer** | **Aktivität** | **Beteiligungsmodul** | **Zusätzliches** |
| - | Patient | interagiert mit | Patientenapp | Der Patient benutzt die Patientenapp, die ihn in seiner Behandlung unterstützt. |
| - | Spitalpersonal | liest von und schreibt auf | Patientenapp | Ärzte, Krankenschwestern, etc. pflegen die Daten vom Patientenapp. |
| 1.0 | Patientenapp | sendet / empfängt Daten | e.0 (Webservice) | Daten werden via Webservice synchronisiert. |
| 1.0 | Patientenapp | verwendet | e.1 (Kamera) | Für die Standortprüfung im „Life up“ mithilfe von QR-Codes oder schiessen von Fotos. |
| 1.0 | Patientenapp | verwendet | e.2 (GPS) | Wird zur Standortprüfung im „Life Up“-System verwendet. |
| 1.0 | Patientenapp | verwendet / öffnet | e.3 (Maps) | Der Patient kann prüfen wo sich ein Termin befindet (Wo ist das Spital, etc.). |
| 1.0 | Patientenapp | verwendet | e.4 (Telefon) | Wird für Anrufe verwendet. |
| 1.0 | Patientenapp | verwendet | e.5 (Speicher) | Der Empfang von Daten aus den Fremdsystemen kann nicht zu jederzeit sichergestellt werden. Aus diesem Grund werden Daten in der Patientenapp zwischengespeichert. |
| e.1 | Kamera | verwendet | e.2 | Mit der eingebauten Kamera wird ein Foto geschossen. Mit den im Bild hinterlegten Informationen kann auch auf die Koordinaten rückgeschlossen werden. |

### 1.0 Patientenapp

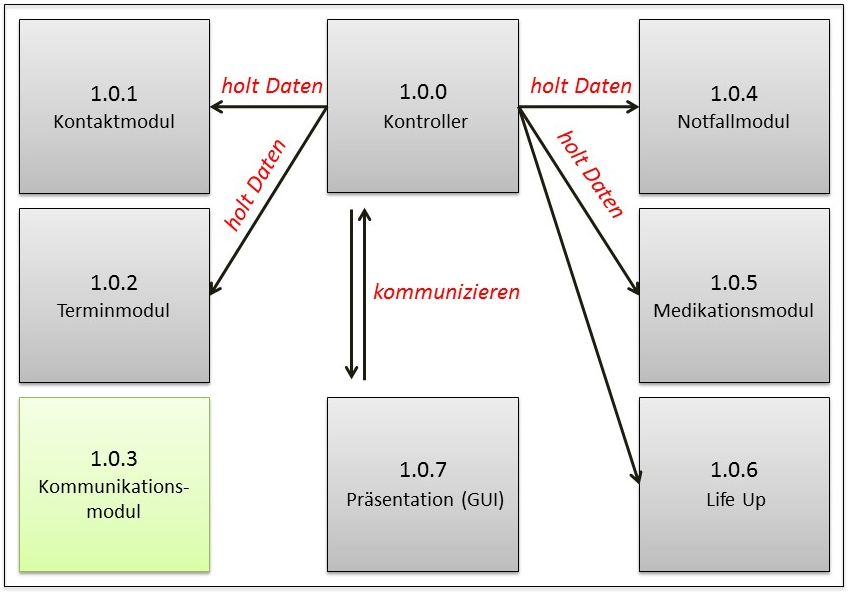


Abbildung 12 Die Patientenapp im Überblick

**Erläuterung**

Die Patientenapp besteht aus mehreren Funktionsgruppen, die sich zu Modulen zusammensetzen lassen. Das Modul 1.0.3 (Kommunikationsmodul) wird ist ein wesentlicher Teil der App und wird von beinahe jedem anderen Modul verwendet. Die wichtigsten Module zur Systemausführung sind der 1.0.0 (Kontroller) und 1.0.7 (Präsentation). Der Kontroller steuert die Applikation. Das 1.0.7 zeigt die Daten im GUI an.

Tabelle 6 Patientenapp Modulbeschreibungen

|  |  |  |  |
| --- | --- | --- | --- |
| **ModulNr.** | **Modul** | **Aktivität** | **Beteiligungsmodul** |
| 1.0.0 | Kontroller | holt Daten | 1.0.1 (Kontaktmodul) |
| 1.0.0 | Kontroller | holt Daten | 1.0.2 (Terminmodul) |
| 1.0.0 | Kontroller | holt Daten | 1.0.4 (Notfallmodul) |
| 1.0.0 | Kontroller | holt Daten | 1.0.5 (Medikationsmodul) |
| 1.0.0 | Kontroller | holt Daten | 1.0.6 (Life up) |
| 1.0.0 | Kontroller | kommuniziert | 1.0.7 (Präsentation) |
| 1.0.3 | Kommunikationsmodul | stellt Schnittstellen bereit |  |

### 1.0.1 Kontaktmodul

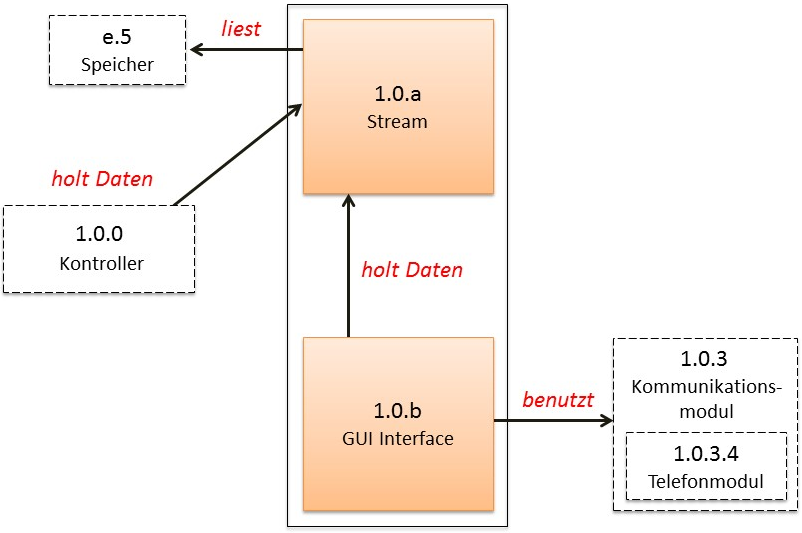


Abbildung 13 Kontaktmodul Übersicht

**Erläuterung**

Das Kontaktmodul behandelt die in der Patientenapp verfügbaren Kontaktpersonen. Um Anrufe tätigen zu können wird vom Kommunikationsmodul die Telefonschnittstelle benutzt.

Die beiden Spezialmodule 1.0.a (Stream) und 1.0.b (GUI Interface) werden generell in der gesamten Applikation verwendet. 1.0.a für das Lesen oder Schreiben auf e.5 (Speicher), 1.0.b für die Datenbefüllung vom 1.0.7 (Präsentation).

Tabelle 7 Komponentenbeschreibung Kontaktmodul

|  |  |  |  |
| --- | --- | --- | --- |
| **ModulNr.** | **Modul** | **Aktivität** | **Beteiligungsmodul** |
| 1.0.0 | Kontroller | holt Daten | 1.0.a (Stream) |
| 1.0.a | Stream | liest | e.5 (Speicher) |
| 1.0.b | GUI Interface | holt Daten | 1.0.a (Stream) |
| 1.0.b | GUI Inteface | benutzt | 1.0.3.4 (Telefonmodul) |

### 1.0.2 Terminmodul

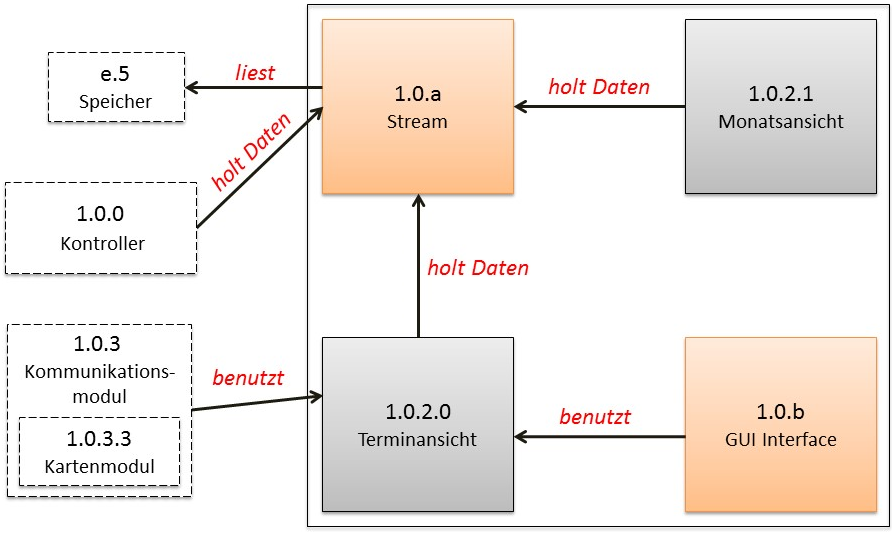


Abbildung 14 Terminmodul Übersicht

**Erläuterung**

Das Terminmodul verwaltet die Termine und stellt diese dar. Um dem Benutzer eine Hilfestellung zum Standort zu geben, kann eine externe Karte angezeigt werden.

Tabelle 8 Terminmodul Komponenten

|  |  |  |  |
| --- | --- | --- | --- |
| **ModulNr.** | **Modul** | **Aktivität** | **Beteiligungsmodul** |
| 1.0.0 | Kontroller | holt Daten | 1.0.a (Stream) |
| 1.0.a | Stream | liest | e.5 (Speicher) |
| 1.0.2.1 | Monatsansicht | holt Daten | 1.0.a (Stream) |
| 1.0.2.0 | Terminansicht | holt Daten | 1.0.a (Stream |
| 1.0.2.0 | Terminansicht | benutzt | 1.0.3.3 (Kartenmodul) |
| 1.0.b | GUI Interface | benutzt | 1.0.2.0 (Terminansicht) |

### 1.0.3 Kommunikationsmodul

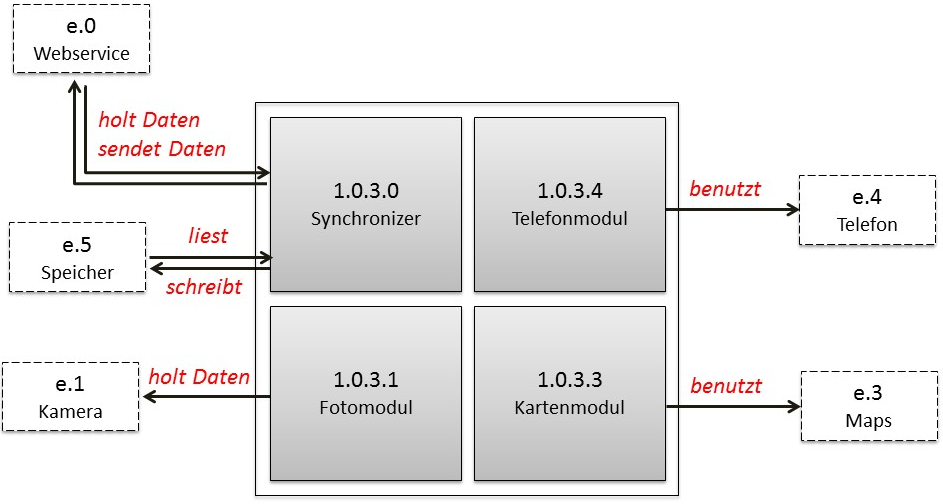


Abbildung 15 Kommunikationsmodul Übersicht

**Erläuterung**

Das Kommunikationsmodul ist die Verbindungskomponente zwischen den internen und externen Funktionen und Daten des Systems. Damit ist der Zugang zentralisiert und einfacher/praktischer zu handhaben.

Tabelle 9 Kommunikationsmodul Komponenten

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ModulNr.** | **Modul** | **Aktivität** | **Beteiligungsmodul** | **Zusätzliches** |
| 1.0.3.0 | Synchronizer | sendet und empfängt Daten | e.0 (Webserice) | Der Synchronzier synchronisiert die Daten mit dem Webservice bei verfügbarer Internetverbindung. |
| 1.0.3.0 | Synchronizer | liest und schreibt | e.5 (Speicher) |
| 1.0.3.1 | Fotomodul | holt Daten | e.1 (Kamera) | Schiesst Fotos |
| 1.0.3.3 | Kartenmodul | benutzt | e.3 (Maps) | Beliebige Kartenapp. Ist noch offen. |
| 1.0.3.4 | Telefonmodul | benutzt | e.4 (Telefon) | - |

### 1.0.4 Notfallmodul

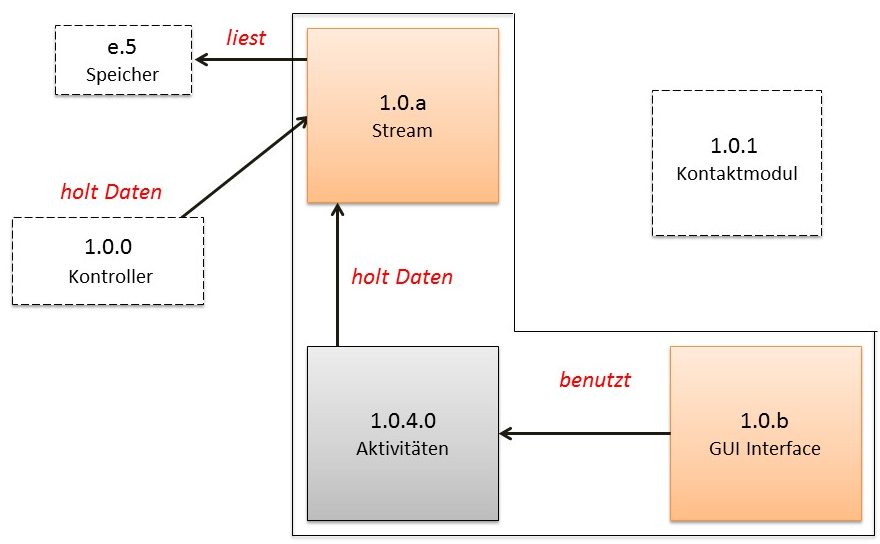


Abbildung 16 Notfallmodul Übersicht

**Erläuterung**

Das Notfallmodul besteht aus zwei Modulen. Dem Notfallmodul und dem Kontaktmodul.

Tabelle 10 Notfallmodul Komponentenbeschreibung

|  |  |  |  |
| --- | --- | --- | --- |
| **ModulNr.** | **Modul** | **Aktivität** | **Beteiligungsmodul** |
| 1.0.0 | Kontroller | holt Daten | 1.0.a (Stream) |
| 1.0.a | Stream | liest | e.5 (Speicher) |
| 1.0.4.0 | Aktivitäten | holt Daten | 1.0.a (Stream) |
| 1.0.b | GUI Interface | benutzt | 1.0.4.0 (Aktivitäten) |
| 1.0.4 | Notfallmodul | verwendet | 1.0.1 (Kontaktmodul) |

### 1.0.5 Medikationsmodul

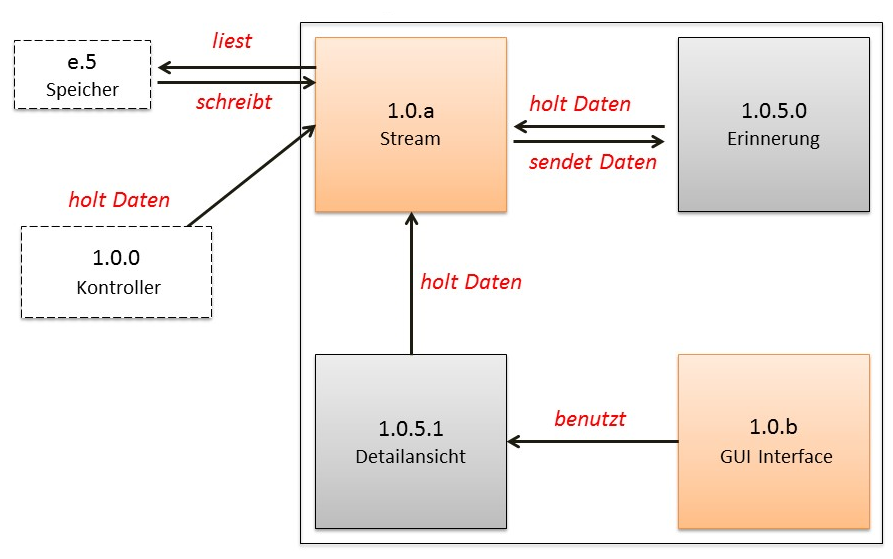


Abbildung 17 Medikationsmodul Übersicht

**Erläuterung**

Das Medikationsmodul löst die Erinnerungsmeldungen aus und kann Details zu verschrieben Medikamenten anzeigen.

Tabelle 11 Medikationsmodul Komponentenbeschreibung

|  |  |  |  |
| --- | --- | --- | --- |
| **ModulNr.** | **Modul** | **Aktivität** | **Beteiligungsmodul** |
| 1.0.0 | Kontroller | holt Daten | 1.0.a (Stream) |
| 1.0.a | Stream | liest | e.5 (Speicher) |
| 1.0.5.1 | Detailansicht | holt Daten | 1.0.a (Stream) |
| 1.0.b | GUI Interface | benutzt | 1.0.5.1 (Detailansicht) |
| 1.0.5.0 | Erinnerung | sendet und empfängt Daten | 1.0.a (Stream) |

### 1.0.6 Life up

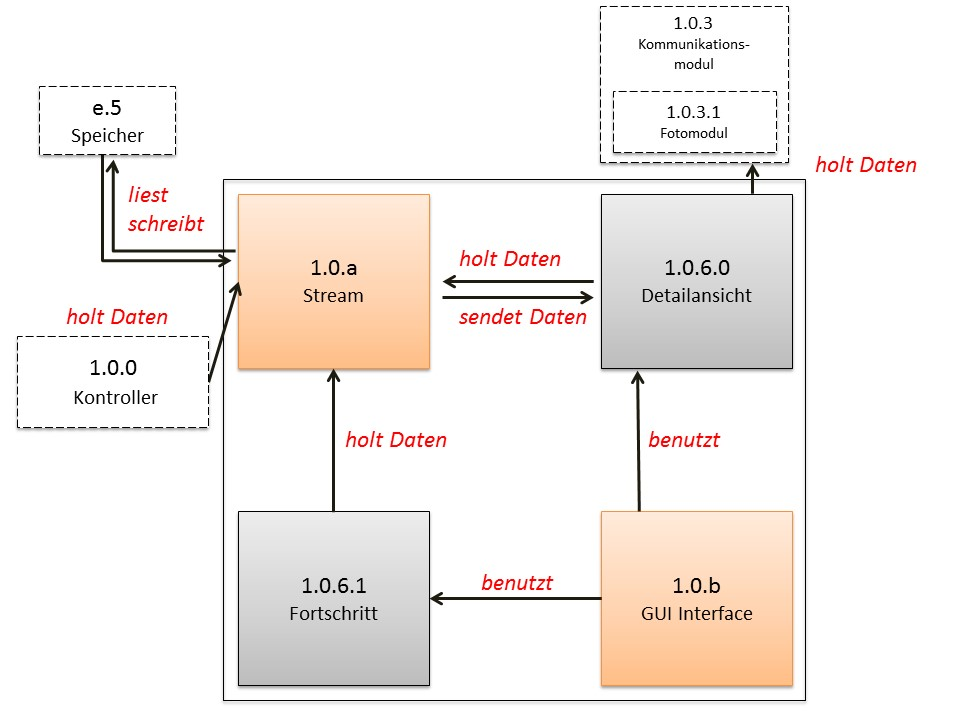


Abbildung 18 Life up Übersicht

**Erläuterung**

Das „Life up“ Modul soll den Patient animieren selbstständiger zu werden. Um gewisse Leistungen des Patienten nachweisen zu können, wird mit Fotomodul ein Foto von einem Geschäft geschossen. Mithilfe des GeoTags des Fotos kann der Therapeut die Aktivitäten des Patienten kontrollieren.

Tabelle 12 Life up Komponentenbeschreibung

|  |  |  |  |
| --- | --- | --- | --- |
| **ModulNr.** | **Modul** | **Aktivität** | **Beteiligungsmodul** |
| 1.0.0 | Kontroller | holt Daten | 1.0.a (Stream) |
| 1.0.a | Stream | liest | e.5 (Speicher) |
| 1.0.6.1 | Fortschritt | holt Daten | 1.0.a (Stream) |
| 1.0.b | GUI Interface | benutzt | 1.0.6.1 (Detailansicht) |
| 1.0.6.0 | Detailansicht | sendet und empfängt Daten | 1.0.a (Stream) |
| 1.0.6.0 | Detailansicht | holt Daten | 1.0.3.1 (Fotomodul) |

# Systemanforderungen

## Functional Requirements

### User Requirements

* Korrekte Abfrage der Medizineinnahme
* Intuitive Handhabung der App
* Schnelle Anzeige von wichtigen Kontaktdaten
* Schnelle Anzeige der benötigten Medikamenten
* Einfache Anzeige des Tagesablaufes

### System Requirements

* Präzise Messung der GPS-Daten
* Zuverlässige Auswertung der GSP Daten / fotografierten Objekte
* Regelmässige Aktualisierung der Daten (Medikamente, Termine, Tagesablauf, Kontakte)

## Non-Functional Requirements

### Product Requirements

* Benutzerhandbuch
* Schulung Patienten und Personal
* Einheitliches UI Design

### Organizational Requirements

* Umsetzung des objektorientierten Konzepts
* Versionsverwaltung mit git
  + Keine „Force Pushs“
  + Sinnvolle Kommentierung
  + Versionen gemäss System Evolution werden getaggt.
* Entwicklungsumgebung: Eclipse

### Domain Specific Requirements

* Ausfallsicherheit
* Datenschutz

## System evolution

Version 0.1 - Homebereich, GUIDas App startet mit dem Homebereich. Im oberen Teil der App befindet sich ein Platzhalter für die Schnellinformation zur nächsten / aktuellen Aufgabe. Der untere Teil der Oberfläche kann mehrere Modulbuttons aufnehmen. Falls zuwenig Platz für die Modulbuttons zur Verfügung steht, lässt sich der untere Bereich scrollen ohne das die Schnellinformation verschwindet.

### Version 0.2 - Kalendermodul

Über den Homebereich lässt sich das Kalendermodul öffnen. Das Kalendermodul zeigt die vergangenen und die zukünftigen Termine an. Die Termine kommen von einer externen Datenquelle. Mit dem öffnen des Termins werden zusätzliche und detailliertere Informationen zu diesem angezeigt.

### Version 0.3 - Erinnerungsfunktion

Die App erinnert an bevorstehende, markierte Termine. In der Schnellinformation im Homebereich der App wird das jeweilige Objekt angezeigt. Per Klick erscheinen detaillierte Informationen zum Objekt.

### Version 0.4 - Medikationsmodul

In den Schnellinformationen wird an die Einnahme der Medikamente erinnert. Per Klick werden details zur Medikation angezeigt.

### Version 0.5 - Rückfall-Notfallmodul

Das Rückfall-Notfallmodul ist eingebaut. Im Homebereich in der Schnellinformation wird ein markanter Button eingebaut der direkt auf das Modul verweist.

Version 0.6 - „life-up“-Modul, Grundsystem

Im Homebereich erscheinen „life-up“-Tätigkeiten. Ein Klick führt direkt ins Modul und zeigt die Details der entsprechenden Tätigkeit.

### Version 0.7 - „life-up“-Modul, Level-Up

Tätigkeiten lassen sich bestätigen. Durch Bestätigen von Tätigkeiten lassen sich Punkte verdienen. Genügend Punkte geben ein „Level-Up“.

### Version 0.8 - Fehlerbehebungen

Durch intensives Testen werden Fehler behoben und die Anwendung wird für ein Beta-Stadium vorbereitet (Version 1)

### Version 1 - Erstes Rollout

## Testkonzept

Getestet wird mit 3 Testverfahren:

* Unittests
* Praxis Test (in Test-Umgebung)
* Benutzertests

### Unittests

Mit Unittests werden während der Entwicklung Testfunktionen geschrieben. Diese Testfunktionen prüfen, ob die eingebundenen Systeme und Systemfunktionen ihre Aufgaben richtig erfüllen und das erwartete Resultat zurückliefern. Die Unit-Testfälle basieren auf den Use-Cases, die in diesem Dokument erfasst sind.

Sollte während der Entwicklung ein Fehler auftauchen, wird er dem zuständigen Programmierer gemeldet und dieser versucht den Fehler zu beheben.

### Systemtests

Für den Systemtest wird eine homogene Umgebung betrieben, auf der die Applikationen installiert ist. Die Applikation ist auf dem Testsystem komplett einsatzfähig. Auf dem Testsystem werden die Funktionen der Applikation vor einem Release getestet.

Mögliche Tests sind:

* Kann der Benutzer (nur) seine Daten einsehen?
* Werden die GPS-Daten zuverlässig gemessen?
* Funktioniert das „Life up“ System (QR-Code lesen)
* Werden die Daten korrekt mit dem Server synchronisiert?

Die Testszenarien werden während der Entwicklung definiert und die erwarteten Ergebnisse schriftlich festgehalten.

Der Ablauf sieht vor, dass zuerst die gesamte Funktionspalette der App ohne ausserordentliche Inputs getestet wird. Funktionieren alle Module wie erwartet, beginnen die Entwickler, mittels falschen Inputs die Sicherheit und Fehleranfälligkeit der App zu testen.

### Benutzertests

Um ein besseres Bild der Benutzerfreundlichkeit zu erhalten wird die App vor dem Produktiven Einsatz bei Probanden installiert.   
Die Probanden geben nach einem festgelegten Zeitraum ein Feedback über die Benutzung, Menüführung und Systemstabilität.

* Kann der Benutzer die App möglichst intuitiv bedienen?
* Hat er Fragen oder Unklarheiten?
* Hat er allfällige Fehler festgestellt?
* Wo sieht er Verbesserungspotential?

Die Applikation wird von unterschiedlichen Patienten getestet um ein besseres Bild zu erhalten.

Die Patienten können laufend Feedback geben. Nach einem festgelegten Zeitraum erhalten die Probanden einen Fragebogen. Der Fragebogen stellt Fragen zur Bedienungsfreundlichkeit, der allgemeinen Zufriedenheit mit der App und Fragen zum Konzept.

Die Feedbacks werden anschliessend ausgewertet und nach sorgfältigem Abwägen in die Entwicklung einfliessen.

Falls wesentliche Bestandteile der App geändert haben, wird das Verfahren wiederholt.

# Appendices

## Hardwareanforderungen

Minimale Hardwareanforderungen an das Smartphone.

* GPS Empfänger
* Kamera
* Minimalsystemanforderung für Android 4.4  
  (https://static.googleusercontent.com/media/source.android.com/en//compatibility/4.4/android-4.4-cdd.pdf)

**Beispiel: Samsung Galaxy S4 Mini**

* Preis: ca. 200 CHF
* Sehr gutes Preisleistungsverhältnis



Abbildung 19 Smartphone mit sehr gutem Preisleistungsverhältnis

**Beispiel: CAT® S50 Smartphone:**

* Preis: ca. 500CHF
* Sehr robust



Abbildung 20 Sehr robustes Smartphone für die Patientenapp

## Datenbanklayout

Die Daten sind in einer Datenbank zu speichern. Das Layout der lokalen Datenbank sieht wie folgt aus:

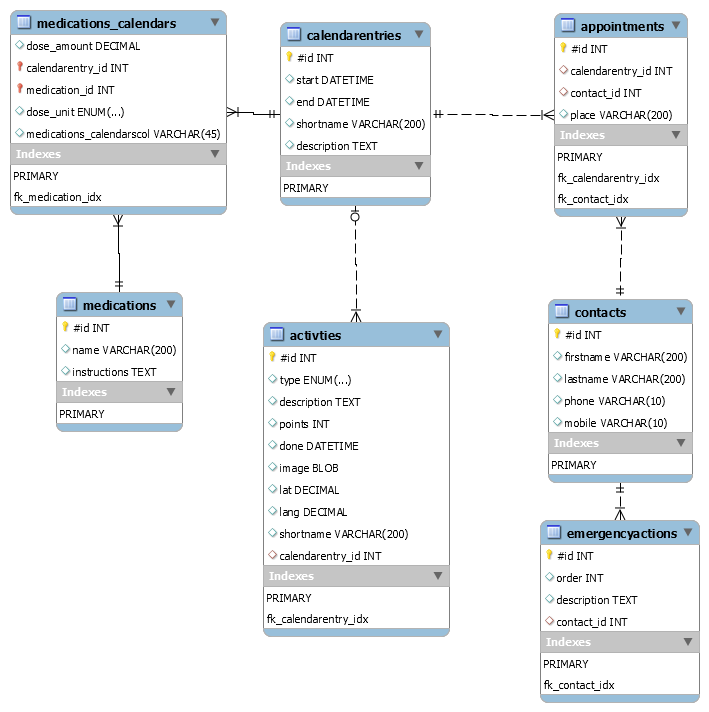


Abbildung 21 Layout der lokalen Datenbank

# Glossar

Tabelle 13 Glossar Begriffserklärung

|  |  |
| --- | --- |
| **Begriff** | **Bedeutung** |
| Life up | Behandlungssystem, welches den Patienten spielerisch motiviert seinen Alltag zu strukturieren. |
| QR-Code | Zweidimensionaler Code, der von Maschinen (z.B. einer Kamera) gelesen werden kann. Der Code beinhaltet Informationen. Das Verfahren ist sehr Fehlerunanfällig. |

# Quellenverzeichnis

Tabelle 14 Quellenverzeichnis

|  |  |  |
| --- | --- | --- |
| **Referenz** | **Quelle** | **Abgerufen am** |
| Logo BFH | http://www.gesundheit.bfh.ch/fileadmin/wgs\_upload/gesundheit/bilder\_und\_logos/BFH\_Logo\_A\_de\_100\_RGB.bmp | 23.03.2015 |
| Abbildung 19 Smartphone mit sehr gutem Preisleistungsverhältnis | http://www.chip.de/ii/1/7/4/4/8/3/9/8/Xperia\_U\_FrontV\_Yellow\_Gold\_600-70984e0eac0cc28f.jpg | 23.03.2015 |
| Abbildung 20 Sehr robustes Smartphone für die Patientenapp | http://www.catphones.com/media/images/360\_spins/b15/img-0.png | 23.03.2015 |

# Abbildungsverzeichnis

[Abbildung 1 Die App ist übersichtlich und einfach strukturiert 5](#_Toc414983622)

[Abbildung 2 Der Patient wird an die Medikamenteneinnahme erinnert. 6](#_Toc414983623)

[Abbildung 3 Der Patient behällt den Überblick über die Medikation 6](#_Toc414983624)

[Abbildung 4 Mit der App hat der Patient seine Termine immer im Überblick 7](#_Toc414983625)

[Abbildung 5 Der Patient wird spielerisch zum Alltag zurück geführt. 9](#_Toc414983626)

[Abbildung 6 Die App unterstützt den Patienten wenn er droht Rückfällig zu werden 10](#_Toc414983627)

[Abbildung 7 Überblick der Kommunikation mit Fremdsystemen 12](#_Toc414983628)

[Abbildung 8 Use Case Diagramm Medikation 14](file:///C:\Users\James\Dropbox\SED%20Gruppe%20rot\System%20requirements.docx#_Toc414983629)

[Abbildung 9 Use Case Diagramm “Life up” 14](file:///C:\Users\James\Dropbox\SED%20Gruppe%20rot\System%20requirements.docx#_Toc414983630)

[Abbildung 10 Activity Diagramm „in Life up ein gefordertes Objekt fotografieren“ 18](#_Toc414983631)

[Abbildung 11 Systemarchitketur Systemkontext 20](#_Toc414983632)

[Abbildung 12 Die Patientenapp im Überblick 22](#_Toc414983633)

[Abbildung 13 Kontaktmodul Übersicht 23](#_Toc414983634)

[Abbildung 14 Terminmodul Übersicht 24](#_Toc414983635)

[Abbildung 15 Kommunikationsmodul Übersicht 25](#_Toc414983636)

[Abbildung 16 Notfallmodul Übersicht 26](#_Toc414983637)

[Abbildung 17 Medikationsmodul Übersicht 27](#_Toc414983638)

[Abbildung 18 Life up Übersicht 28](#_Toc414983639)

[Abbildung 19 Smartphone mit sehr gutem Preisleistungsverhältnis 33](#_Toc414983640)

[Abbildung 20 Sehr robustes Smartphone für die Patientenapp 33](#_Toc414983641)

[Abbildung 21 Layout der lokalen Datenbank 34](#_Toc414983642)

# Tabellenverzeichnis

[Tabelle 1 Versionsübersicht des Dokuments 4](#_Toc414983608)

[Tabelle 2 Beschreibung Use Case 1 15](#_Toc414983609)

[Tabelle 3 Beschreibung Use Case 2 16](#_Toc414983610)

[Tabelle 4 Beschreibung Use Case 3 17](#_Toc414983611)

[Tabelle 5 Erläuterung Systemarchitektur 20](#_Toc414983612)

[Tabelle 6 Patientenapp Modulbeschreibungen 22](#_Toc414983613)

[Tabelle 7 Komponentenbeschreibung Kontaktmodul 23](#_Toc414983614)

[Tabelle 8 Terminmodul Komponenten 24](#_Toc414983615)

[Tabelle 9 Kommunikationsmodul Komponenten 25](#_Toc414983616)

[Tabelle 10 Notfallmodul Komponentenbeschreibung 26](#_Toc414983617)

[Tabelle 11 Medikationsmodul Komponentenbeschreibung 27](#_Toc414983618)

[Tabelle 12 Life up Komponentenbeschreibung 28](#_Toc414983619)

[Tabelle 13 Glossar Begriffserklärung 35](#_Toc414983620)

[Tabelle 14 Quellenverzeichnis 35](#_Toc414983621)