

# Patientenspezifisch konfigurierbare Pflegeprozesse basierend auf BPMN-Prozessbausteinen für die Prozessengine Camunda

Prof. Dr. Peter Edelmann

Technische Hochschule  
Mittelhessen  
Wilhelm-Leuschner-Straße 13  
61169 Friedberg  
Peter.Edelman@mnd.thm.de

Prof. Dr. Armin  
Wagenknecht

Technische Hochschule  
Mittelhessen  
Wiesenstraße 14  
35396 Gießent  
Armin.Wagen-  
knecht@mni.thm.de

M.Sc. Melvin Schmidt

Technische Hochschule  
Mittelhessen  
Wiesenstraße 14  
35396 Gießen  
melvin.schmidt92@gmail.com

## ABSTRACT

In der Praxis werden wegen des hohen Aufwands meist nur solche Prozesse mittels BPMN automatisiert, die oft wiederverwendet werden können wie etwa ein Warenbestellprozess. Im Pflegebereich benötigt man jedoch für jeden Patienten einen individualisierten Prozess, der oft nur genau einmal für diesen Patienten durchlaufen wird und für den nächsten Durchlauf eventuell schon wieder an die veränderte Gesundheit des Patienten angepasst werden muss.

Die vorliegende Arbeit untersucht durch Entwicklung eines Prototyps die Hypothese, dass die Kapselung einer Camunda-Prozessengine durch eine React-Applikation einer Pflegekraft ermöglicht, den für die Pflege eines Patienten notwendigen Prozess mittels Drag&Drop als geschachtelte ToDo-Liste auf einem Tablet zu spezifizieren. In diesem Prototyp werden komplexe, in BPMN modellierte Subprozesse zur Ausführung einzelner Pflegeaufgaben abstrahiert und so gekapselt, dass ein Lego-ähnliches System aus grobgranularen und patientenspezifisch konfigurierbaren Pflegebausteinen entsteht. Damit werden unwesentliche Prozessdetails und somit Komplexität vor der Pflegekraft verborgen.

## SCHLÜSSELWÖRTER

Prozessautomatisierung, Prozessindividualisierung, Pflegeprozesse, Medizintechnik, BPMN

## EINLEITUNG

Im Mai 2021 wurde das „Digitale-Versorgung-und-Pflege-Modernisierungs-Gesetz“ vom Bundesrat gebilligt. Es unterstützt die Verwendung digitaler Pflegeanwendungen in der Regelversorgung [Bun21]. Nach einer Studie der Bertelsmann-Stiftung führen digitale Anwendungen in der Pflege zu einer Fehlerreduzierung und Zeiteinsparung im Pflegealltag [LTBHB21].

Die vorliegende Arbeit untersucht, ob eine Camunda Prozessengine so durch eine React Applikation gekapselt werden kann, dass eine Pflegekraft in der Lage ist, selbstständig und ohne Kenntnis von BPMN den für die Pflege eines Patienten notwendigen Prozess zu spezifizieren.

## PROBLEME HERKÖMMLICHER

### PROZESSAUTOMATISIERUNGEN IN DER

### PFLEGE

In der Praxis werden aus Aufwands- und Kostengründen meist nur solche Prozesse automatisiert, die oft wiederverwendet werden können, wie etwa die Kreditabwicklung einer Bank oder ein Warenbestellprozess. Hier lohnt die Beauftragung eines Informatikers zur direkten Codierung oder Konfigurierung einer Prozessengine mittels

BPMN. Solche Prozesse sind ohne ein Hintergrundwissen zur Codierung oder zu BPMN meist nicht modifizierbar.

Im Pflegebereich benötigt man für jeden Patienten einen individualisierten Prozess, der oft nur genau einmal für diesen Patienten durchlaufen wird und für den nächsten Durchlauf ggf. schon wieder modifiziert werden muss, um ihn an die veränderte Gesundheit des Patienten oder eine veränderte Pflegesituation anzupassen. Design und Modifikation dieser patientenspezifischen Prozesse sollten daher von Personen ohne informationstechnischen Hintergrund, z.B. vom Pflegepersonal, durchgeführt werden können. Für einen Patienten müssen sowohl der Top-Level-Prozess in Form der grobgranularen Pflegeaufgaben (z.B. Diabetis-Kontrolle, Verband wechseln, Medikament verabreichen, ...) wie auch die zu den Pflegeaufgaben gehörenden Subprozesse (z.B. Zucker messen und schwellenwertabhängig Insulin spritzen im Rahmen der Diabetis-Kontrolle) passend konfiguriert werden.

## STAND DER FORSCHUNG

Die nachfolgend vorgestellten Arbeiten beschreiben Systeme, die eine eingeschränkte Konfiguration von Patientenprozessen durch medizinisches Fachpersonal ermöglichen.

### Behandlungspfade in der Medizin

Der Artikel [LGM09] beschreibt Vor- und Nachteile von Behandlungspfaden in der Medizin.

Behandlungspfade definieren einen Prozess, der den gesamten Behandlungsablauf abbildet: So wird z.B. für die Behandlung des akuten Schlaganfalls festgelegt, wie Patienten, die sich für eine Thrombolyse eignen, bereits vom Sanitäter oder dem Hausarzt angemeldet werden, welches Behandlungsteam involviert ist und wie eine standardisierte Bildgebung vor der Lysetherapie erfolgt. Es werden Kriterien festgelegt, wie lange ein Patient auf der Intensivstation oder der Stroke Unit überwacht werden muss, wie und durch wen die Weiterbetreuung auf der Abteilung erfolgt und wann der Patient in die Rehabilitation verlegt werden kann.

Durch dieses standardisierte Vorgehen können Qualitätskriterien wie die „door-to-needle“-Zeit oder die Hospitalisationsdauer gemessen werden. Diese Strukturierung komplexer Abklärungs- und Behandlungswege innerhalb eines Krankenhauses ermöglicht es, Prozesse zu optimieren, eine Kosten-, Ergebnis- und Leistungstransparenz zu ermöglichen und vorhandene Leistungs- und Kostensenkungspotentiale auszuschöpfen.

Behandlungspfade sind jedoch sehr komplexe Konstrukte, die einen hohen Aufwand zu ihrer Erstellung und Aufrechterhaltung benötigen. Behandlungspfade liegen daher meist nur in Papier-Form vor, nicht aber in digitaler oder sogar automatisierter Form.

### iManageMyHealth and iSupportMyPatients

Der Artikel „iManageMyHealth and iSupportMyPatients“ [SSBa18] aus dem Jahr 2018 beschreibt ein spezielles System im Bereich der Krebsbehandlung. Das System besteht aus den Applikationen „iSupportMyPatients“ für die Ärzte sowie „iManageMyHealth“ für die Patienten. In Abbildung 1 ist die Prozessverwaltung der „iSupportMyPatients“-Applikation abgebildet.

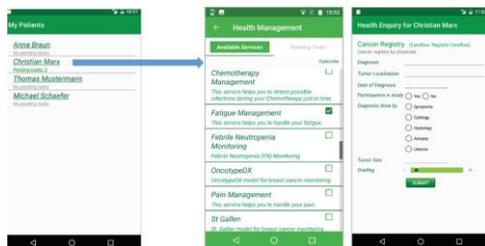


Abbildung 1: Benutzungsoberfläche „iSupportMyPatients“ [SSBa18]

Krebsspezialisten können mittels eines vereinfachten Editors für jeden Patienten individuell einen patientenspezifischen Prozess zur Krebsbehandlung entwerfen. Der Editor ähnelt einem BPMN-Editor und ermöglicht den Aufbau des Prozesses aus elementaren BPMN-Elementen wie User-Tasks, Service-Tasks und Gateways. Ein Generator bildet dabei die für medizinisches Fachpersonal angepassten Prozess-Elemente des Editors unter Beibehaltung der Struktur auf BPMN 2.0 Elemente ab. Der resultierende BPMN Prozess wird von der Process-Engine Activiti [Alf21] ausgeführt. Patienten und Ärzte erhalten von der Prozessengine ihre Aufgaben zur Bearbeitung. Die Entscheidungen der Gateways basieren

auf Decision Rules oder Predictive Models für Krebsbehandlungen. Prozesse können für Patienten selektiert und aktiviert werden.

### A Process-oriented Approach for Supporting Clinical Decisions for Infection Management

Der Artikel „A Process-oriented Approach for Supporting Clinical Decisions for Infection Management“ [BFB+17] beschreibt ein System, das die Erstellung von Prozessen für die Infektionsverwaltung basierend auf klinischen Richtlinien ermöglicht. Dabei werden die Prozesse zum Infektionsmanagement von Domänenexperten in einem BPMN-Editor erstellt. Das System zeigt den Status der einzelnen Aufgaben und sendet Erinnerungen. Zudem werden die medizinischen Tasks als farbliche Blöcke in einer Zeitleiste visualisiert. Die Blöcke repräsentieren die Zeitabschnitte der Tasks aus dem BPMN-Prozess und ermöglichen eine Überwachung des Fortschritts.

Die Prozesse werden nicht von einer gängigen Processengine ausgeführt, sondern von einer eigens entwickelten Lösung, welche Daten aus verschiedenen Krankenhausinformationssystemen bezieht.

In Abbildung 2 ist die grafische Oberfläche der Anwendung dargestellt.

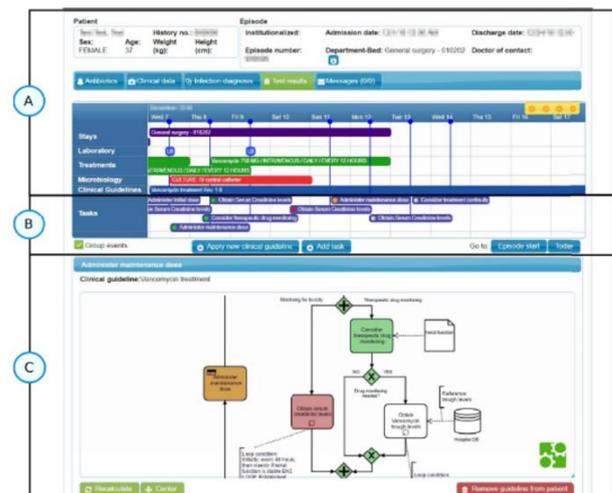


Abbildung 2: Screenshot der Anwendung

Oben: Übersicht Patientenprozess mit Aufgaben in einem Zeitstrahl;

Unten: Lokalisierung der Aufgabe innerhalb des BPMN-Prozess[BFB+17]

## APP FÜR DURCH PFLEGEKRÄFTE

### KONFIGURIERBARE

### PATIENTENSPEZIFISCHE PFLEGEPROZESS

Der vorliegende Ansatz löst die oben genannten Probleme von individualisierten Prozessen, indem er Subprozesse zur Ausführung einzelner Pflegeaufgaben so kapselt, dass ein Lego-ähnliches System aus grobgranularen Pflegebausteinen entsteht (siehe Abbildung 3).

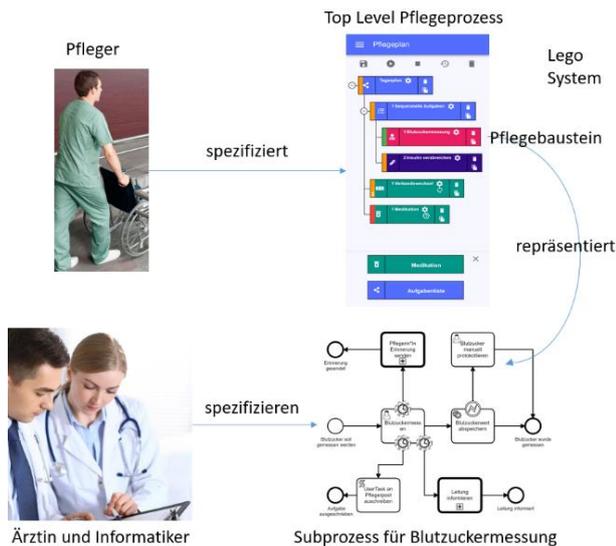


Abbildung 3: Visualisierung des Lego-ähnlichen Konzeptes

Quelle: Eigene Darstellung

Die Pflegebausteine repräsentieren komplette, in BPMN modellierte, wiederverwendbare und patientenspezifisch konfigurierbare Subprozesse. Durch die Kapselung als Pflegebausteine werden diese Subprozesse abstrahiert und damit unwesentliche Prozessdetails und somit Komplexität vor der Pflegekraft verborgen (Information Hiding).

Im Gegensatz zu den im Kapitel Stand der Forschung vorgestellten Ansätzen werden die patientenspezifischen Pflegeprozesse im vorliegenden Ansatz nicht mit einem BPMN-ähnlichen Editor aus elementaren Low-Level-Elementen wie Human Tasks, Service Tasks und Gateways als Aktivitätsfluss modelliert, sondern aus grobgranularen Pflegebausteinen als Repräsentanten für komplexe, abstrahierte Subprozesse baumartig strukturiert.

Beispielsweise kann eine Pflegekraft für einen Patienten einen Pflegebaustein „Blutzuckermessung“ direkt vor dem Pflegebaustein „Frühstück“, im Rahmen des Top-Level-Pflegeprozesses für einen Tag konfigurieren. Sie orientiert sich dabei ggf. an einem Behandlungspfad „ZuckerErkrankung“. Die einzelnen Pflegeaufgaben der von Ärzten definierten und in der Regel nicht durch eine Prozessengine automatisierten Behandlungspfade bilden damit die Basis für die Zusammenstellung des täglichen oder wöchentlichen durch eine Prozessengine automatisierten Pflegeprozesses für einen spezifischen Patienten.

Die Baumstruktur der Pflegebausteine ermöglicht einer Pflegekraft, den für die Pflege eines Patienten benötigten Top-Level-Prozess mittels Drag&Drop als geschachtelte ToDo-Listen zu spezifizieren.

Die Applikation ist zur besseren Integration in den Pflegealltag ergonomisch für Smartphones und Tablets gestaltet und unterstützt die Arbeit der Pflegekräfte durch Protokollierung und Erinnerungen.

Eine Evaluation hinsichtlich Praxistauglichkeit und Zuverlässigkeit erfolgte durch examinierte Pflegekräfte.

Anwendbar ist der vorliegende Ansatz in solchen Domänen, die wie im Pflegebereich Top-Level-Prozesse mit einem hohen Individualanteil und wiederverwendbare Subprozesse enthalten, wie etwa kundenspezifische Betreuungs- oder Entwicklungsprozesse.

Die im Rahmen des Prototyps modellierten Pflegeprozesse haben nicht den Anspruch inhaltlich vollständig und medizinisch korrekt zu sein. Sie dienen lediglich als Anschauungsbeispiele.

Basierend aus der Relevanz im Pflegealltag, werden die Pflegeaufgaben „Blutzuckermessung“, „Insulin verabreichen“, „Medikation“ und „Verbandswechsel“ als Beispiele ausgewählt.

### EIGENSCHAFTEN DES PROTOTYPS

Im Folgenden werden nur die für den vorliegenden Ansatz wesentlichen Eigenschaften anhand von Screenshots erläutert.

#### Funktionale Eigenschaften

##### Pflegeprozess erstellen/bearbeiten/starten

Die Pflegekraft kann einen individuellen baumartig strukturierten Top-Level-Pflegeprozess für einen Patienten erstellen und dazu Pflegebausteine in den Pflegeprozess per Drag&Drop einfügen sowie kopieren und löschen (siehe Abbildung Abbildung 4a (G) sowie 4b (H)).

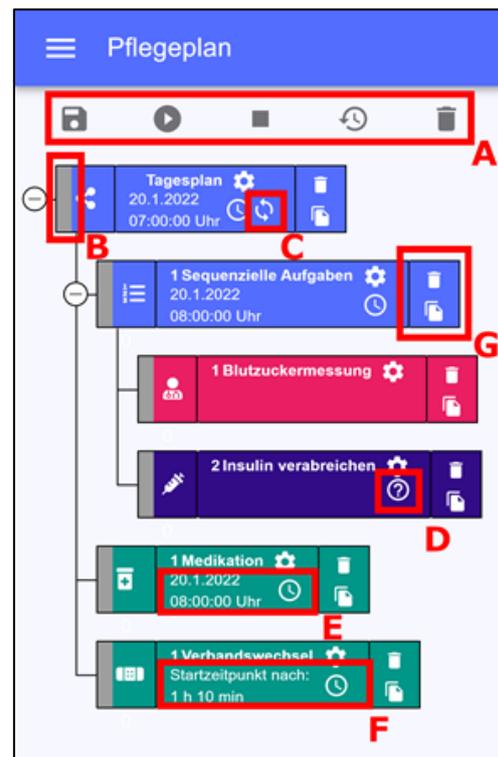


Abbildung 4a: Oberfläche des Prozesskonfigurators

Quelle: Eigene Darstellung



Abbildung 4b: Konfigurationsdialog für den Blutzuckermanagement-Baustein

Quelle: Eigene Darstellung

Die Pflegekraft kann z.B. im Konfigurationsdialog des Pflegebausteins "Blutzuckermanagement" den Startzeitpunkt, den Schwellwert für eine Verabreichung von Insulin oder eventuelle Wiederholungen spezifizieren (siehe Abbildung 4b).

Nach dem Start des erstellten Pflegeprozess durch die Pflegekraft (siehe Abbildung 4a (A)) startet die Applikation den zugehörigen Top-Level-BPMN-Camunda-Prozess und ggf. die durch die Pflegebausteine spezifizierten BPMN-Camunda-Subprozesse.

### Ausführen einer Human-Task

Die Pflegekraft wird von der Prozessengine Camunda mittels TaskList zur Ausführung einer innerhalb eines Subprozesses spezifizierten elementaren Human-Task, z.B. zur Verabreichung eines Medikamentes, aufgefordert und die Verabreichung wird nach der Quittierung der Human Task protokolliert (siehe Abbildung 5).

Abbildung 5: Human-Task "Medikament verabreichen"

Quelle: Eigene Darstellung

### Prozesshistorie und Pflegeprotokoll einsehen

Die Pflegekraft kann sich sowohl über die Historie und den Zustand des Prozesses informieren (z.B. grün=erledigt, gelb=laufend, rot=fehlerhaft, siehe Abbildung 6), als auch über Daten und Fehlermeldungen, die während der Ausführung von Pflegebausteinen entstanden sind (siehe Abbildung 7).

Der erfolgreiche Abschluss eines Pflegebausteins, die Anzahl der Wiederholungen und aufgetretene Fehler werden angezeigt.

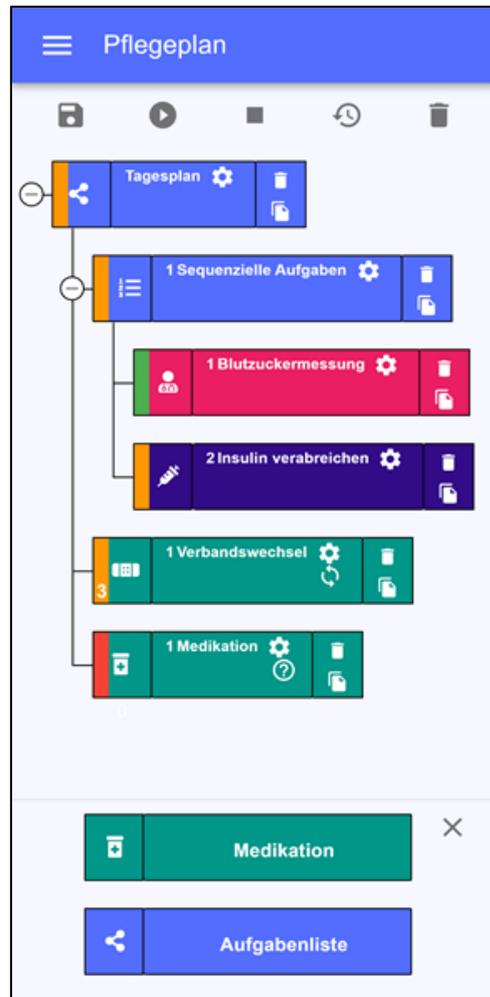


Abbildung 6: Darstellung des Prozesszustands

Quelle: Eigene Darstellung

### Qualitative Eigenschaften

Durch die Reduktion der Sicht von dem Subprozess auf den kapselnden Pflegebaustein mit zugehörigem Konfigurationsdialog wird die Komplexität des Subprozesses vor der Pflegekraft verborgen.

Die Zusammenstellung von ToDo-Listen mittels Drag&Drop ermöglicht eine für Pflegekräfte intuitive und einfache Bedienbarkeit.

Durch das Lego-ähnliche Bausteinprinzip werden Wartbarkeit, Wiederverwendbarkeit und Erweiterbarkeit des Systems gewährleistet.

Pflegeprotokoll	
BLUTZUCKER	VERBAND
<div style="display: flex; justify-content: space-between;"> <span>🔍</span> <span>📄</span> <span>🖨️</span> <span>☰</span> <span>☰</span> </div>	
Patient*in	Erika Mustermann
Pfleger*in	demo
Blutzucker	130
Blutzucker wurde gemessen?	ja
Notiz	Patient nicht nüchtern
Datum	20.1.2022, 11:56:08
Patient*in	Erika Mustermann
Pfleger*in	

Abbildung 7: Protokoll Blutzuckermessung

Quelle: Eigene Darstellung

## VERWENDETE FRAMEWORKS UND TECHNOLOGIEN

Als grafische Notation für die Modellierung und Automatisierung der Prozesse dient der ISO-zertifizierte und dementsprechend weltweit genormte Prozessmodellierungsstandard BPMN 2.0.

Das Lesen und Schreiben von BPMN-2.0-Diagrammdateien erfolgte mittels BPMN Moddle, ein Camunda Open-Source Framework für Javascript [Git22a].

Für die Erstellung und Ausführung von BPMN-2.0-Prozessen wurde Camunda BPM, ein quelloffenes und auf Java basierendes Framework verwendet [Cam19].

Die auf React basierende grafische Oberfläche nutzt Capacitor [Cap22b], eine Laufzeitumgebung, welche die Entwicklung von hybriden Applikationen erlaubt, die nativ auf iOS-, Android- und im Browser laufen [Ion22].

Der Prototyp soll lediglich die Machbarkeit des Konzeptes zeigen. Die verwendeten Technologien wurden daher nicht im Hinblick auf eine Optimierung für den Pflege-

bereich ausgewählt. Jedoch unterstützt React als Framework für die grafische Oberfläche unterschiedlichste von Pflegekräften verwendete Smartphone-Betriebssysteme. Die leichtgewichtige Prozessengine Camunda ist als Bibliothek einem Microservice Container hinzufügbare und über „external worker“ und Polling asynchron an beliebige Technologien für Pflegemanagement-Systeme anbindbar.

## ARCHITEKTURKONZEPTE

### Grobgranulare Pflegebausteine

Grobgranulare Pflegebausteine repräsentieren und abstrahieren komplexe, wiederverwendbare, in BPMN modellierte Subprozesse für die Durchführung einer Pflegeaufgabe. Diese Subprozesse beschreiben die im Rahmen einer Pflegeaufgabe auszuführenden Aktionen der beteiligten Personen oder Softwaresysteme.

Für die Aufgabe „Blutzuckermanagement“ beinhaltet dies etwa die Messung des Blutzuckers, die Protokollierung des gemessenen Wertes in einer Datenbank, die Prüfung des Schwellwertes für eine Verabreichung von Insulin, die Verabreichung von Insulin, sowie Sonderfälle wie das Eskalationsmanagement durch eine Push-Nachricht an die Pflegeleitung, die Absicherung der zu verabreichenden Insulindosis durch das 2-Augen-Prinzip, eine Wiederholung der Blutzuckermessung oder eine Fehlerbehandlung.

Durch die Kapselung als grobgranulare Pflegebausteine werden diese Subprozesse abstrahiert und damit unwesentliche Prozessdetails und somit Komplexität vor der Pflegekraft verborgen (Information Hiding), so z.B. die komplexen Prozessstrukturen, die sich durch ein gestuftes Eskalationsmanagement ergeben.

Die Bausteine sind patientenspezifisch konfigurierbar. Sie werden durch typisierte JavaScript-Objekte repräsentiert und die zugehörigen Konfigurationsdaten für den repräsentierten Subprozess durch typisierte Attribute.

### Lego-System aus Pflegebausteinen

Die Pflegebausteine können über Arrays zu einem Baum verlinkt werden. Der Baum repräsentiert aus der Sicht der Pflegekraft geschachtelte ToDo-Listen.

Die Pflegebausteine können durch eine Kapselung mit Loop-Bausteinen explizit wiederholt, mit If-Bausteinen bedingt ausgeführt und mit And-Bausteinen gruppiert werden. Die Gruppierung ermöglicht eine sequentielle oder parallele Ausführung. Prozessvariablen ermöglichen einen Informationstransport zwischen den Bausteinen.

Neue Bausteine erfordern im Wesentlichen die BPMN-Modellierung der elementaren Aktionen zur Durchführung der Pflegeaufgabe als Subprozess, die Codierung der den Subprozess repräsentierenden JavaScript-Klasse und der zugehörigen Datenbanktabellen zur Persistierung.

## KOMPONENTENSTRUKTUR DES PROTOTYPS

Ein Übersichtsdiagramm der Systemkomponenten ist in Abbildung 8 zu sehen.

Die Komponenten CareProcessManagement und BPMNProcessAndTaskManagement sind für das Verständnis des vorliegenden Ansatzes wesentlich und werden daher näher erläutert. Die übrigen Komponenten des Systems dienen der Verwaltung von Patienten, Nutzern, Zugriffsrechten und dem Reporting geloggtter medizinischer Messwerte.



Abbildung 8: Übersicht der System Komponenten

Quelle: Eigene Darstellung

### Komponente CareProcessManagement

Diese Komponente (siehe Abbildung 9) ist verantwortlich für die Erstellung, Verwaltung und Überwachung der von einer Pflegekraft spezifizierten patientenspezifischen Pflegeprozesse.

So ermöglicht die CareProcessManagementGUI etwa die Erstellung und Konfiguration eines Pflegeprozesses als Baum CareProcessTree aus Pflegebausteinen CareProcessElement. Der Service CareProcessManagementService verwaltet und persistiert diesen Pflegeprozess mittels des CareProcessManagementDAOService, transformiert ihn mittels des BPMNCreatorService in einen BPMNProcess und deployed die zugehörige BPMN-XML-Datei an die Komponente BPMNProcessAndTaskManagement. Darüber hinaus erfragt der Service CareProcessManagementService nach dem Start des Pflegeprozesses den Status des zugehörigen BPMN-Prozesses bei der Komponente BPMNProcessAndTaskManagement für ein Update der Statusanzeige in der CareProcessManagementGUI.

Der CareProcessTree der Komponente CareProcessManagement repräsentiert den zugehörigen Top-Level-BPMNProcess der Komponente BPMNProcessAndTaskManagement für die patientenspezifische Pflege und das CareProcessElement den zugehörigen BPMNSubProcess für eine einzelne Pflegeaufgabe.

### Komponente BPMNProcessAndTaskManagement

Die Komponente wird im Wesentlichen durch das Framework Camunda realisiert (graue Komponenten, siehe Abbildung 10). Ausführbare BPMN-Prozesse und Tasks (hellblaue Komponenten) sind patientenspezifisch angepasst. Der ausführbare Top-Level BPMNProcess wird von der Komponente CareProcessManagement nach der Transformation des von der Pflegekraft als Baum spezifizierten Pflegeprozesses in eine BPMN-XML-Datei an die Komponente BPMNProcessAndTa-

skManagement übergeben und mittels ProcessManagementService verwaltet sowie mittels BPMNProcessAndDAOService persistiert.

Der ausführbare BPMNSubProcess zur Realisierung einer Pflegeaufgabe wird z.B. von einer Ärztin mittels ProcessModelerGUI passend zu dem medizinischen Pflegebaustein modelliert und vom BPMNProcessManagementService verwaltet. Er wird mittels von der Pflegekraft spezifizierten Prozessvariablen des Top-Level BPMNProcess patientenspezifisch konfiguriert.

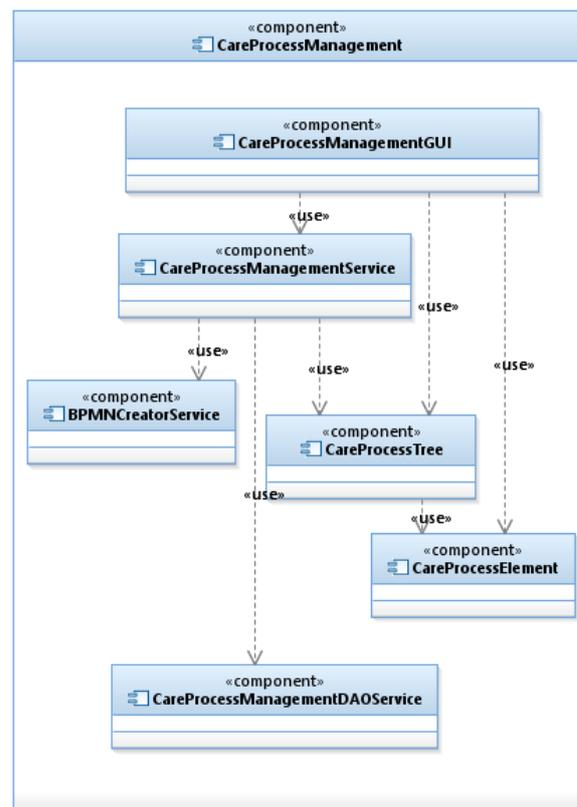


Abbildung 9: Übersicht der Komponente CareProcessManagement

Quelle: Eigene Darstellung

Die Prozessengine wird von der CareProcessManagement Komponente mit dem Start des Top-Level-BPMN-Prozesses beauftragt. Sie übernimmt die Ausführung des Top-Level-Prozesses durch Aufrufen der Subprozesse sowie Instantiieren von HumanTasks und ExternalTasks. Zur Bearbeitung anstehende HumanTasks werden von der TaskListGUI mittels HTML-Formulare angezeigt und von der Pflegekraft ausgeführt. Z.B. misst die Pflegekraft den Blutzucker und trägt diesen in das Formular ein. Anstehende ExternalTasks werden in der ExternalTaskList verwaltet und vom external Worker gepollt. Z.B. erfragt ein external Worker der Komponente ReportingManagement einen Log-Eintrag für den Blutzucker und persistiert ihn. Andere external Worker versenden z.B. im Rahmen des Eskalationsmanagements per E-Mail Erinnerungen an Pflegekräfte oder deren Vorgesetzte.

## Komponente BPMNCreatorService

Die Komponente BPMNCreatorService (siehe Abbildung 9) transformiert den von der Pflegekraft als Baum CareProcessTree spezifizierten Pflegeprozess in eine von der Prozessengine ausführbare BPMN-XML-Datei. Dazu verwendet sie BPMN Moddle, ein Camunda Open-Source Framework für Java-Script zum Lesen und Schreiben von BPMN2.0-Diagrammdateien [Git22a]. Der Baum aus CareProcessElement JavaScript-Objekten wird innerhalb des BPMNCreatorServices mittels Visitor-Pattern traversiert. Dabei wird für einen Pflegebaustein wie etwa ein BloodSugarElement ein Objekt der Camunda Bibliothek BPMN-Moddle erzeugt, welches eine BPMN CallActivity repräsentiert. Diese CallActivity ruft per Link im Rahmen ihrer Ausführung in Camunda z.B. den zuvor von der Ärztin modellierten BPMN Subprozess für die Pflegeaufgabe Blutzucker

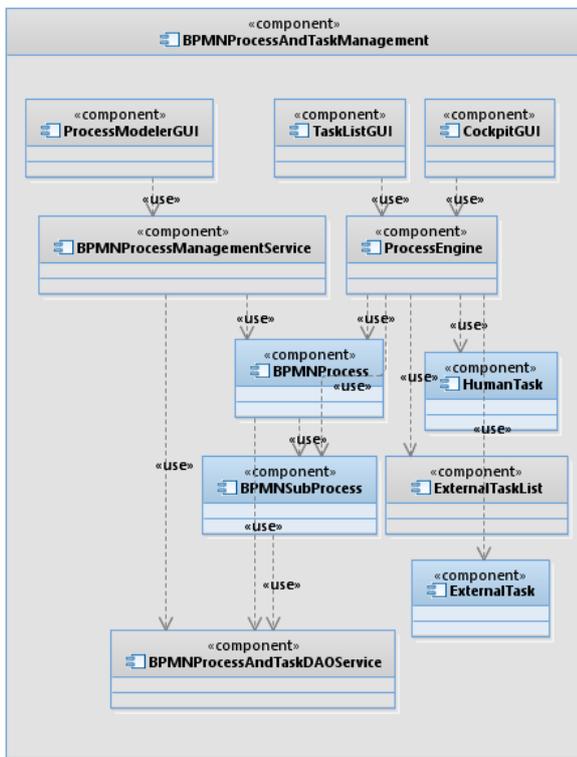


Abbildung 10: Übersicht der Komponente BPMNProcessAndTaskManagement

Quelle: Eigene Darstellung

Messung auf (Zuordnung über ID des SubProzesses). Der CallActivity werden Prozessvariablen zur patientenspezifischen Konfigurierung des BPMN Subprozesses übergeben, die zuvor von der Pflegekraft mittels Konfigurationsdialog spezifiziert wurden. Mittels BPMN-Moddle wird schließlich aus den miteinander verbundenen Moddle Objekten eine BPMN-XML-Datei für den patientenspezifischen Top-Level Pflegeprozess erzeugt.

## EXEMPLARISCHE BAUSTEINE

### Struktur-Bausteine

#### Generischer Timer-If-Loop-Baustein zur Kapselung funktionaler Bausteine mit Strukturelementen

Timer, Wiederholungs- und Wenn-Bausteine basieren auf dem in Abbildung 11 dargestellten BPMN-Diagramm. Der Prozessfluss des Subprozesses wird im ersten Schritt durch ein Timer-Event (T) blockiert. Die Konfiguration erlaubt ein absolutes Datum oder einen relativen Zeitabstand zum vorhergehenden Baustein. Das OR-Gateway (IF1) überprüft zunächst, ob die Prozessvariablen aus der Wenn-Bedingung bereits im Prozess vorhanden sind. Falls nicht, wird eine External-Task aufgerufen, die eine Fehlermeldung an den ReportingService weitergibt. Anschließend wird im OR-Gateway (IF2) geprüft, ob die konfigurierte Wenn-Bedingung (z.B. BloodSugar > 120) erfüllt ist. Ist sie erfüllt, läuft der Prozessfluss weiter in den Kind-Baustein (z.B. Insulin verabreichen), ansonsten wird der Prozessfluss beendet (IF2).

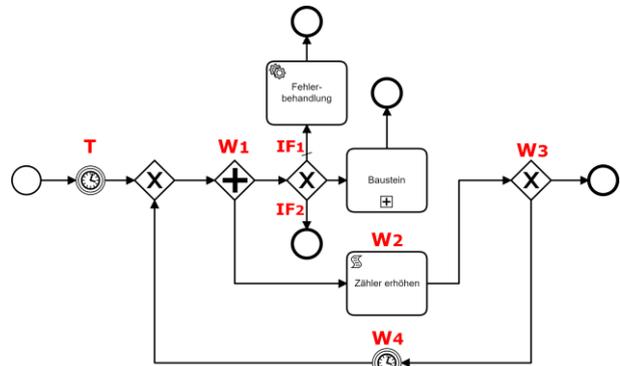


Abbildung 11: BPMN-Diagramm des „Timer-If-Loop“-Subprozess

Quelle: Eigene Darstellung

Der Prozessfluss wird nach dem Timer-Event mit einem AND-Gateway (W1) geteilt. Der so entstehende zweite Prozessfluss dient zur Realisierung einer Wiederholung des ummantelten Pflegebausteins ohne Blockade durch einen vorangegangenen ggf. hängenden Durchlauf. In diesem zweiten Prozessfluss wird durch eine Skript-Task (W2) ein Zähler erhöht. Im OR-Gateway (W3) wird geprüft, ob der Zähler den Wert in der Konfigurationsvariablen LOOP COUNT überschreitet oder ob das Datum END DATETIME erreicht ist und ggf. die Schleife beendet. Das TIMER-Gateway (W4) repräsentiert die Zeitspanne LOOP DURATION für die Wartezeit bis zu einem erneuten Durchlauf der Schleife.

Um die vielfältigen Konfigurationsmöglichkeiten in allen Pflegebausteinen verwenden zu können, wird jeder Baustein (außer Timer, Wenn und Wiederholung) mit diesem Timer-If-Loop Baustein ummantelt. Somit kann beispielsweise auch am Pflegebaustein Blutzuckermessung direkt eine Wiederholung definiert werden ohne explizite Anordnung unter einem isolierten Wiederholungsbaustein.

Der Aufbau des Timer-If-Loop Bausteins ermöglicht auch, strukturelle Prozesskonfigurationen in einzelne Bausteine zu kapseln. Beispielweise ist im isolierten Wenn-Baustein If der Ausdruck „true“ für das OR-Gateway (W3) hinterlegt, da dieser Baustein nicht wiederholt werden soll. Im Falle des isolierten Wiederholungsbausteins Loop ist im OR-Gateway (IF) der Wert „true“ gesetzt für eine nicht bedingte Ausführung des Kind-Bausteins. Der separate Timer-Baustein Timer bestimmt lediglich den Zeitpunkt (absolut oder relativ zum vorigen Baustein) der Ausführung.

Zur Konfiguration werden der Pflegekraft alle Bausteinparameter aus dem bis dahin modellierten Pflegeprozess in einem Drop-Down-Feld angezeigt. So kann etwa ein Schwellwert für den zuvor modellierten Blutzuckermessungs-Pflegebaustein als Bedingung für die Verabreichung von Insulin formuliert werden. Mehrere Bedingungen werden mit einem logischen UND-Operator verknüpft.

Der Timer-If-Loop Baustein enthält kein Eskalationsmanagement, da die Eskalationsstrategie von der semantischen Bedeutung des ummantelten Pflegebausteins abhängt. Beispielsweise kann die verspätete Ausführung des Bausteins „Insulin verabreichen“ lebensbedrohliche Folgen haben im Gegensatz zu einer „Zahnpflege“.

### Parallele-Liste Baustein

Die Pflegebausteine innerhalb dieser Liste werden unabhängig voneinander parallel ausgeführt.

### Sequenzielle-Liste Baustein

Pflegebausteine innerhalb dieser Liste werden hintereinander ausgeführt. Jeder Baustein innerhalb der Liste erhält eine Nummerierung basierend auf der Reihenfolge.

### Domänenspezifische Bausteine für die Pflege

#### Blutzuckermessung-Baustein

Der Pflegebaustein basiert auf dem in Abbildung 12 dargestellten BPMN-Diagramm.

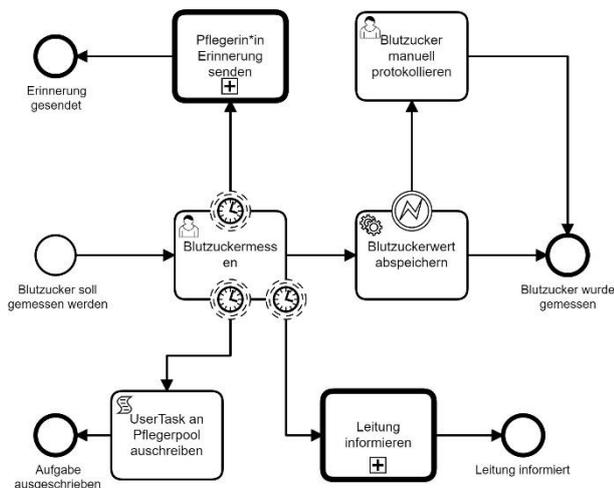


Abbildung 12: BPMN-Diagramm Blutzuckermessung

Quelle: Eigene Darstellung

Im Normalfall trägt die Pflegekraft den gemessenen Blutzuckerwert rechtzeitig im Rahmen der Human-Task „Blutzuckermessen“ in ein Webformular ein. Der Wert wird in die Prozessvariable bloodSugar gemappt. Der erfasste Blutzuckerwert wird über eine External-Task „Blutzuckerwert abspeichern“ durch den ReportingService in die Datenbank persistiert. Tritt im Rahmen der Persistierung eine Exception auf, wird die Pflegekraft mittels Human-Task „Blutzucker manuell protokollieren“ aufgefordert, die zuvor gemessenen Daten manuell zu notieren. Bei nicht rechtzeitiger Ausführung der Blutzuckermessung benachrichtigt das Timer-Boundary-Event in konfigurierten Zeitabständen die Pflegekraft über eine unerledigte Aufgabe in der Taskliste (CallActivity: „Pflegekraft Erinnerung senden“). Nach einer weiteren Zeitspanne wird die Aufgabe mittels Script-Task „UserTask an Pflegerpool ausschreiben“ zusätzlich an einen Pfleger-Pool vergeben. Mitglieder dieses Pools können diese Aufgabe nun ebenfalls sehen und erledigen. Wird die Aufgabe auch dann nicht erledigt, wird die Leitung per E-Mail (CallActivity: „Leitung informieren“) über die unerledigte Aufgabe informiert.

#### Insulin-verabreichen-Baustein

Die Pflegekraft kann im Rahmen dieses den Pflegebaustein Blutzuckermessung ergänzenden Bausteins die verabreichte Dosierung Insulin eintragen.

#### Medikament-verabreichen-Baustein

Mithilfe des Bausteins wird die Verabreichung von Medikamenten im Vier-Augen-Prinzip ermöglicht. Die erste Pflegekraft kann den Namen des Medikaments und die vorgesehene Dosierung angeben. Die zweite Pflegekraft überprüft und bestätigt die Angaben. Die erste Pflegekraft protokolliert die verabreichte Dosierung.

#### Kombinationsbaustein

#### BlutzuckerManagement-Baustein

Der Baustein „Blutzuckermanagement“ beinhaltet die Messung des Blutzuckers sowie die Verabreichung von Insulin in Abhängigkeit von einem Schwellwert. Die Abbildung 13 zeigt das BPMN-Diagramm des Subprozesses.

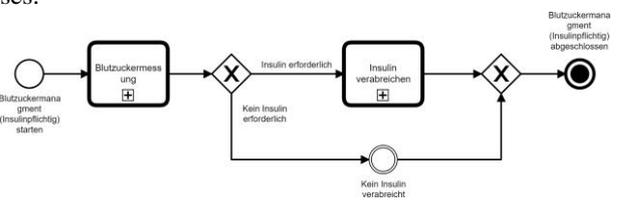


Abbildung 13: BPMN-Diagramm Blutzuckermanagement

Quelle: Eigene Darstellung

Nachdem der Blutzuckermessung-Subprozess beendet wurde, wird in einem OR-Gateway der Ausdruck „bloodSugar > bloodSugarThreshold“ ausgewertet. Die Variable bloodSugarThreshold wird über den Konfigurationsdialog an die CallActivity übergeben.

Der Kombinationsbaustein verbirgt die Struktur der Bedingung und die Abhängigkeit der Pflegebausteine „Blutzuckermessung“ und „Insulin verabreichen“ über die Prozessvariable bloodSugar vor der Pflegekraft. Die Konfiguration wird auf das Setzen des Schwellwertes reduziert.

## BAUSTEINÜBERGREIFENDER INFORMATIONSTRANSPORT

Der Camunda Scope regelt, welche Variablen für welchen Prozess sichtbar sind. Sobald der Prozessfluss in einen Subprozess eintritt, wird ein neuer Subprozess-Scope erzeugt. Alle Aktivitäten innerhalb des Subprozess-Scopes (z.B. Human Task) können die zugehörigen Variablen sehen, jedoch nicht die Variablen, die außerhalb des Subprozess-Scopes liegen. Für einen Subprozess kann über ein Prozessvariablen-Mapping definiert werden, welche Prozessvariablen bei Aufruf an den Subprozess weitergegeben werden und bei Abschluss des Subprozesses an den aufrufenden Prozess zurück propagiert werden.

Abbildung 14 zeigt geschachtelte Scopes am Beispiel „Blutzuckermessung“ und „Insulin verabreichen“. Aus Darstellungsgründen sind die „Timer-If-Loop“-Ummantelungen (B'),(I'),(M') der Pflegebausteine nicht vollständig mit ihren Elementen abgebildet.

Innerhalb der Human Task in (B) gibt die Pflegekraft den Wert der Blutzuckermessung (bspw. 130) ein. Somit wird die Prozessvariable bloodSugar=130 innerhalb von Scope (B) gesetzt. Wird nun der Subprozess B beendet, wird der bloodSugar-Wert zunächst in den Scope (B') propagiert und nach Beendigung von (B') in den Scope (S). Nun läuft der Prozessfluss weiter in die „Insulin-verabreichen“-Ummantelung (I'). Durch das Input Mapping ist nun die Prozessvariable bloodSugar im Scope (I') enthalten. Der „Timer-If-Loop“-Mantel kann nun den logischen Ausdruck (bloodSugar > 120) in (I') auswerten.

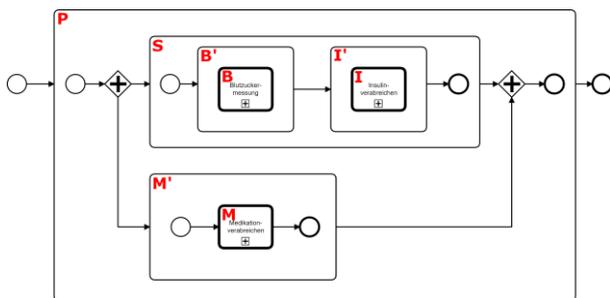


Abbildung 14: Weitergabe Prozessvariablen an Subprozess

Quelle: Eigene Darstellung

## TEST UND EVALUATION

Die Korrektheit und Zuverlässigkeit der Anwendung wurde am Beispiel von unterschiedlich konfigurierten Pflegeprozessen geprüft. Die Testfälle umfassten neben der zeitgerechten Durchführung von spezifizierten Pflegeaufgaben auch die Prüfung der vorgesehenen Eskalati-

onsstufen (Erinnerung, Zuweisung an Pflegepool, Nachricht an Pflegeleitung) bei nicht zeitgerechter Durchführung sowie eine Fehlkonfiguration bedingt durch die Vertauschung voneinander abhängiger Pflegebausteine. Die Evaluation der App hinsichtlich ihrer Anwendbarkeit und Nutzungsfreundlichkeit im Pflegeumfeld erfolgte am Beispiel eines kleinen exemplarischen Pflegeprozesses basierend auf den Bausteinen des Prototyps durch examinierte Pflegekräfte.

## FAZIT UND AUSBLICK

Der vorliegende Ansatz ermöglicht die Konfiguration von patientenspezifischen Pflegeprozessen durch Pflegekräfte ohne informationstechnischen Hintergrund, indem er Camunda-BPMN-Subprozesse für einzelne Pflegeaufgaben so durch eine React-Applikation kapselt, dass ein Lego-ähnliches System aus konfigurierbaren Pflegebausteinen entsteht. Diese Pflegebausteine repräsentieren und abstrahieren komplexe, in BPMN modellierte, wiederverwendbare Subprozesse für die Durchführung einer Pflegeaufgabe.

Im Gegensatz zu den im Kapitel Stand der Forschung vorgestellten Ansätzen werden die Pflegeprozesse nicht als Aktivitätsflüsse mit einem BPMN-Editor aus elementaren Low-Level-Elementen modelliert, sondern aus grobgranularen Pflegebausteinen baumartig strukturiert. Die durchgeführte Evaluation lässt erwarten, dass Pflegekräfte ohne Kenntnisse von BPMN bereits nach kurzer Einlernphase komplexe Pflegeprozesse als geschachtelte ToDo-Listen mittels Drag&Drop auf einem Tablet erstellen können.

Für das betrachtete Verfahren ergeben sich Verbesserungsmöglichkeiten bezogen auf die nachfolgenden Aspekte.

Für eine belastbare Evaluation in einer typischen Pflegeeinrichtung sollten die Pflegebausteine des Prototyps um weitere üblicherweise benötigte Bausteine ergänzt werden.

Hinzugefügt werden sollte auch ein Rechtssystem mit Rollen, um den im medizinischen Bereich benötigten Zugriffs- und Datenschutz zu gewährleisten. Mit einem solchen System könnte auch der Freiheitsgrad beim Design der Pflegeprozesse an die Fähigkeiten des Wissensarbeiters angepasst werden. Z.B. könnte das Recht, den Schwellwert für die Insulinverabreichung zu spezifizieren oder mittels isolierter Loop- und If-Bausteine beliebig komplex strukturierte Pflegeprozesse zu erstellen, auf die Pflegeleitung beschränkt werden.

Im oft hektischen Alltag der Pflegekräfte wäre eine Integration der Camunda-Taskliste in die App hilfreich, ebenso wie eine Erinnerung an ausstehende Tasks über Push-Nachrichten basierend auf einem Dienst wie Firebase Cloud.

Hilfreich wäre auch die Möglichkeit, zur Laufzeit Änderungen am Prozess vornehmen zu können.

Camunda erlaubt im Prinzip, Prozesse dynamisch zu ändern. Eine Übertragung des Konzeptes auf die App steht noch aus.

Interessant wäre eine Untersuchung, wie sehr durch generische Komponenten, kombiniert mit einem Plugin-Konzept, der Aufwand zur Ergänzung von Pflegebausteinen bzw. zum Umstieg auf eine andere Domäne reduziert werden kann.

Ebenfalls von Interesse wäre eine Untersuchung, inwieweit die Einbringung z.B. eines vollständig automatisierten Behandlungspfades „ZuckerErkrankung“ sinnvoll und möglich ist. Ggf. müssten zu diesem Zweck Behandlungspfade im Hinblick auf eine patientenspezifische Anpassung in elementare durch Drag&Drop zusammenstellbare Pflegebausteine zerlegt und mit Pflegebausteinen anderer Behandlungspfade synchronisiert werden (z.B. Zuckermessung eine halbe Stunde vor dem Frühstück).

In einem weiteren Prototyp könnte man prüfen, ob höhere Petrinetze ggf. besser für die angedachte Modellierung und Automatisierung der Pflegeprozesse geeignet sind.

Eine Evaluation des Konzeptes durch Anwendung auf andere Domänen, die wie im Pflegebereich Top-Level-Prozesse mit einem hohen Individualanteil und wiederverwendbare Subprozesse enthalten, wie etwa kundenspezifische Betreuungs- oder Entwicklungsprozesse, steht noch aus.

## LITERATURVERZEICHNIS

[Alf21] ALFRESCO SOFTWARE, Inc: Open Source Business Automation — Activiti. <https://www.activiti.org/>. Version: 21.07.2021

[BFB+17] B. CAÑOVASS-SEGURA, B.; ZERBATO, F.; OLIBONI, B.; COMBI, C.; CAMPOS M.; MORALES, A.; JUAREZ, J. M.; MARIN R.; PALACIOS, F.: A Process-Oriented Approach for Supporting Clinical Decisions for Infection Management. Version: 2017. <http://dx.doi.org/10.1109/ICHL.2017.73>. In: IEEE (Hrsg.): IEEE International Conference 2017. 2017. S. 91–100

[Bun21] BUNDESREGIERUNG: Mehr Digitalisierung in der Pflege. <https://www.bundesregierung.de/bregde/suche/digitalisierung-pflege-1841204>. Version: 27.04.2021

[Cam19]CAMUNDA: User Guide — camunda BPM docs. <https://docs.camunda.org/manual/7.3/guides/user-guide/>. Version: 28.05.2019

[Cap22b] CAPACITOR: CROSS-PLATFORM NATIVE RUNTIME for WEB APPS: Cordova and PhoneGap - Capacitor. <https://capacitorjs.com/docs/cordova>. Version: 04.02.2022

[Git22a] GITHUB: bpmn-io/bpmn-moddle: Read and write BPMN 2.0 XML from JavaScript. <https://github.com/bpmn-io/bpmn-moddle>. Version: 04.02.2022

[Ion22] IONIC: Capacitor Web View for iOS and Android Apps - Ionic Framework. <https://ionicframework.com/docs/core-concepts/webview>. Version: 04.02.2022

[LGM09] LAIFER, G.; GAUDENZ, R.; MEIER, C. M.: Behandlungspfade in der Medizin; Klinik für Innere Medizin, Stadtspital Triemli, Zürich, 2009

[LTBHB21] LUTZE, M.; TRAUZETTEL, F.; BUSCHHEIZMANN, A.; OVENSCHULTE, M.: Potenziale einer Pflege 4.0. (2021), S. 92 [https://www.bertelsmannstiftung.de/fileadmin/files/user\\_upload/Pflege\\_4.0\\_final.pdf](https://www.bertelsmannstiftung.de/fileadmin/files/user_upload/Pflege_4.0_final.pdf)

[SSBa18] SCHERA, F.; SCHÄFER, M.; BUCUR, A.; et al.: iManageMyHealth and iSupportMyPatients: Mobile decision support and health management apps for cancer patients and their doctors. In: Ecancermedalscience. Online journal 12 (2018), S. Art. 848, 11

## KONTAKT

Prof. Dr. Peter Edelmann  
Technische Hochschule Mittelhessen  
Wilhelm-Leuschner-Straße 13  
61169 Friedberg  
Peter.Edelman@mnd.thm.de

Prof. Dr. Armin Wagenknecht  
Technische Hochschule Mittelhessen  
Wiesenstraße 14  
35396 Gießen  
Armin.Wagenknecht@mni.thm.de

M.Sc. Melvin Schmidt  
Technische Hochschule  
Mittelhessen  
Wiesenstraße 14  
35396 Gießen  
melvin.schmidt92@gmail.com