



Hochschule für Angewandte Wissenschaften Hamburg
Hamburg University of Applied Sciences

Bachelorthesis

Glenn Bickel

**Konzeptionierung und Implementierungsanalyse eines
Systems zur übergreifenden Darstellung und Koordination
von Patienteninformationen mit Schwerpunkt auf die
Integration in die bestehenden Systeme**

*Fakultät Technik und Informatik
Department Informatik*

*Faculty of Engineering and Computer Science
Department of Computer Science*

Glenn Bickel

Konzeptionierung und Implementierungsanalyse eines Systems zur übergreifenden Darstellung und Koordination von Patienteninformationen mit Schwerpunkt auf die Integration in die bestehenden Systeme

Abschlussarbeit eingereicht im Rahmen der Bachelorprüfung

im Studiengang Bachelor of Science Wirtschaftsinformatik
am Department Informatik
der Fakultät Technik und Informatik
der Hochschule für Angewandte Wissenschaften Hamburg

Betreuender Prüfer: Prof. Dr. Rüdiger Weißbach

Zweitgutachter: Prof. Dr. Ulrike Steffens

Eingereicht am 17.März 2016

Autor

Glenn Bickel

Thema der Arbeit

Konzeptionierung und Implementierungsanalyse eines Systems zur übergreifenden Darstellung und Koordination von Patienteninformationen mit Schwerpunkt auf die Integration in die bestehenden Systeme.

Stichworte

Soarian, myMedis, systemübergreifende Datenbankabfrage, Webentwicklung, IIS 8.5, PHP

Kurzzusammenfassung

In einem komplett digitalisierten Krankenhaus bestehen besondere Ansprüche an die Verfügbarkeit von patientenbezogenen Daten. Da die IT-Landschaft solcher Krankenhäuser aber zumeist historisch gewachsen ist, kann eine Kommunikation der einzelnen Systeme nicht immer über die etablierten Wege erfolgen.

Diese Thesis beschreibt eine derartige Situation am Universitätsklinikum Hamburg-Eppendorf und zeigt einen Lösungsweg zur systemübergreifenden Abfrage und Darstellung von Daten aus Soarian Clinicals der Cerner Corporation und aus myMedis der c.a.r.u.s. HMS GmbH auf. Dies geschieht auch im Hinblick auf den Datenschutz und die Anforderungen der Mitarbeiterinnen und Mitarbeiter in den OP-Sälen des Universitätsklinikums Hamburg-Eppendorf und des Altonaer Kinderkrankenhauses.

Author

Glenn Bickel

Title of the paper

Conception and implementation analysis of a system for managing and displaying patient data focused on the integration into the existing systems.

Keywords

Soarian, myMedis, cross-system data access, Web Engineering, IIS 8.5, PHP

Abstract

In a totally digitalized hospital are particular aspects which have to be considered to provide patient data to the users. Since the IT-Landscape of such hospitals often is historically grown, the established solutions cannot always be used.

This thesis paper describes a similar situation concerning the Universitätsklinikum Hamburg-Eppendorf and suggests a solution to access and display data obtained from Cerner Corporations Soarian Clinicals and c.a.r.u.s. HMS GmbHs myMedis. This is done in regard to the patients' privacy and the employees' requirements who are working in the operating rooms of the Universitätsklinikum Hamburg-Eppendorf and the Altonaer Kinderkrankenhaus.

Inhaltsverzeichnis

1	Einleitung	10
1.1	Das Universitätsklinikum Hamburg-Eppendorf	10
1.1.1	IT-Systeme im Universitätsklinikum Hamburg-Eppendorf	12
1.1.1.1.	Über Soarian.....	13
1.1.1.2.	Über myMedis.....	14
1.2	Aufgabenstellung	16
2	Hauptteil	18
2.1	Anforderungen	18
2.1.1	Interview	18
2.2	Design-Konzeption	19
2.3	Konzept zur Implementation	22
2.3.1	Hardware-Voraussetzungen	22

2.3.2	Software-Voraussetzungen.....	23
2.3.3	Sicherheit	24
2.3.4	Auswahl der Programmiersprache	26
2.3.5	Wartung und Erweiterung	27
2.4	Praxis	30
2.4.1	Server-Konfiguration.....	30
2.4.2	Datenbankzugriff mittels PHP.....	33
3	Zusammenfassung.....	36
3.1	Fazit	36
3.2	Ausblick	38
4	Literatur- und Quellenverzeichnis	40
5	Anhang.....	43

Abkürzungsverzeichnis

AKK.....	<i>Altonaer Kinderkrankenhaus</i>
ASP.....	<i>Active Server Pages .NET</i>
BSI.....	<i>Bundesamt für Sicherheit in der Informationstechnologie</i>
CMS.....	<i>Content-Management-System</i>
EMRAM.....	<i>Electronic Medical Record Adaption Model</i>
HIMSS.....	<i>Healthcare Information an Management Systems Society</i>
HTML.....	<i>Hypertext Markup Language</i>
HTTPS.....	<i>Hypertext Transfer Protocol Secure</i>
IIS.....	<i>Internet Information Services</i>
JSP.....	<i>Java Server Pages</i>
KAS.....	<i>Klinisches Arbeitsplatzsystem</i>
KIS.....	<i>Krankenhausinformationssystem</i>
PHP.....	<i>Hypertext Preprocessor</i>
UKE.....	<i>Universitätsklinikum Hamburg-Eppendorf</i>
UML.....	<i>Unified Modeling Language</i>

Abbildungs- und Tabellenverzeichnis

Abbildung 1: Soarian Clinicals Patientenakte	13
Abbildung 2: myMedis OP-Plan	15
Abbildung 3: W3Techs.com Marktanteil serverseitiger Programmiersprachen.....	26
Abbildung 4: Gegenüberstellung der Ausgaben der Datenbanken	35

1 Einleitung

1.1 Das Universitätsklinikum Hamburg-Eppendorf

Diese Thesis ist in Zusammenarbeit mit der IT-Abteilung des Universitätsklinikum Hamburg-Eppendorf (UKE) entstanden.

Das UKE gehört mit seinen 1.420 Betten und über 380.000 jährlichen Patientinnen und Patienten¹ zu den größten Krankenhäusern Deutschlands. Die Patientendaten verwaltet das UKE komplett digital in einer elektronischen Patientenakte, die auf Soarian Clinicals der Cerner Corporation² basiert und durch die eigene IT-Abteilung verwaltet und ständig erweitert wird.

Für seine elektronische Patientenakte und das papierlose Arbeiten am Klinikum wurde das UKE im November 2011 als erstes deutsches Krankenhaus für die Stufe 7 des „Electronic Medical Record Adaption Model“ (EMRAM) ausgezeichnet.³

Das EMRAM ist ein Model der weltweit tätigen „Healthcare Information and Management Systems Society“ (HIMSS), einer gemeinnützigen Organisation, die ihren Fokus auf die Verbesserung der Gesundheitsversorgung durch den intelligenten Einsatz von IT-Systemen

¹ Siehe: „Geschäftsbericht und Jahrbuch 2014“, 2015.

² Genauere Informationen zu Soarian sind im Abschnitt **1.1.1.1 – über Soarian** nachzulesen.

³ Siehe: „Universitätsklinikum Hamburg-Eppendorf nimmt mit Software von Siemens Spitzenplatz in Europa ein“, 2011.

gelegt hat.⁴ Mit der EMRAM-Auszeichnung bewertet HIMSS den Grad des papierlosen Arbeitens an einem Krankenhaus, Stufe 7 ist in diesem Model die höchste Stufe und beschreibt eine vollkommene Digitalisierung der Patientendaten.

Dem UKE als Gesamtkonzern gehören noch weitere Kliniken und Krankenhäuser an. Diese haben ihren Sitz entweder direkt auf dem Gelände des UKE, wie z.B. die Martini-Klinik und das Universitäre Herzzentrum oder liegen außerhalb, wie das Altonaer Kinderkrankenhaus(AKK). Alle Unternehmen des UKE speisen ihre Patientendaten in die elektronische Patientenakte ein und teilen sich damit weitreichend dieselbe Soft- und Hardware. Dies ermöglicht es einerseits, auf redundante Infrastruktur zu verzichten, andererseits erfordert es aber auch einen weiteren Sicherheitsaspekt bei der Berechtigung zur Einsicht von Patientenakten zu berücksichtigen. Auch wenn die einzelnen Unternehmen zum selben Konzern gehören muss gewährleistet sein, dass Mitarbeiterinnen und Mitarbeiter der anderen Unternehmen die Daten der behandelten Personen nicht ohne deren Einwilligung einsehen können.

⁴ Siehe: „Gesundheitswesen neu“, o.J.

1.1.1 IT-Systeme im Universitätsklinikum Hamburg-Eppendorf

Die IT-Infrastruktur des UKE umfasst eine Vielzahl von Systemen, mit denen die Mitarbeiterinnen und Mitarbeiter Daten über Patientinnen und Patienten erheben, einsehen oder bearbeiten können. Da die IT-Systemlandschaft des UKE historisch gewachsen ist, arbeiten viele der Systeme auf eigenen Datenbanken, was das Abrufen von Daten mitunter sehr stark verkompliziert oder es gänzlich unmöglich macht. So benutzen z.B. einige Fachabteilungen noch ihre eigenen Softwaresysteme zum Erfassen und Verwalten von Patientendaten, die somit nicht vollständig in die gemeinsame Datengrundlage des Konzern UKE laufen.

Diese Thesis bezieht sich ausschließlich auf die Daten aus der elektronischen Patientenakte (Soarian) und auf die des Systems zur OP-Planung, zur OP-Dokumentation und zum Management der Belegung innerhalb der Operationssäle (myMedis).

Für ein besseres Verständnis werden Soarian und myMedis und ihre Funktion im Gesamtkonzern UKE in den folgenden Abschnitten erläutert.

1.1.1.1. Über Soarian

Soarian ist ein Krankenhausinformationssystem (KIS), das von Siemens Health Services entwickelt und im Februar 2015 von der Cerner Corporation übernommen wurde⁵.

Das KIS ist die zentrale Stelle, in der alle Daten der Patientinnen und Patienten, wie z.B. Name, Adresse, aber auch Risikofaktoren (Vorerkrankungen/Allergien), Diagnosen und Behandlungen gespeichert werden. Über dieses System können die Mitarbeiterinnen und Mitarbeiter des Pflegedienstes alle Informationen abrufen, die sie zur Bewältigung ihrer täglichen Arbeit benötigen. Des Weiteren können die Mitarbeiterinnen und Mitarbeiter des medizinischen Dienstes vorausgegangene Diagnosen einsehen, Behandlungen dokumentieren oder auch Anforderungen stellen, zum Beispiel ans Zentrallabor.

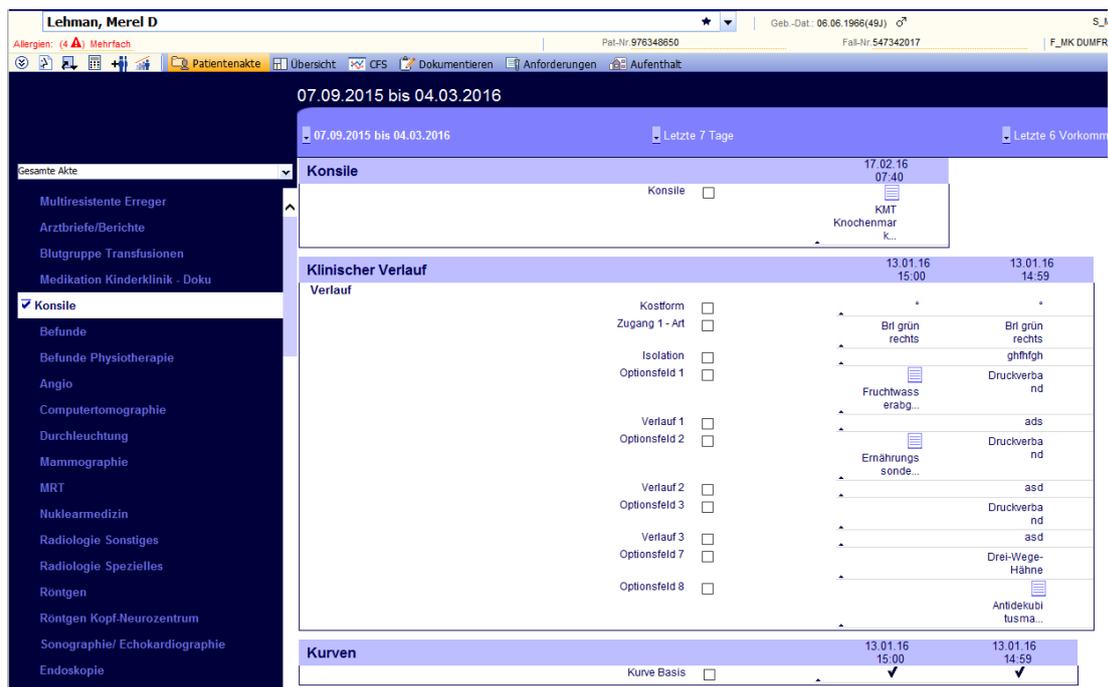


Abbildung 1: Soarian Clinicals Patientenakte

Quelle: Eigene Darstellung, Cerner Soarian Clinicals, Version 4.00 SP06

⁵ Nähere Informationen siehe: „Cerner Completes Acquisition of Siemens Health Services“, 2015.

Neben der Planung und Organisation der Behandlungen, wird aber auch die Verwaltung über Soarian geregelt, wie zum Beispiel das Verfassen von Arztbriefen, die zur Nachbehandlung durch den weiterbehandelnden Arzt benötigt werden oder die Erfassung aller relevanten Daten, die zur Abrechnung mit den Krankenkassen notwendig sind.

Für einige Fachbereiche sind auch weitere Systeme an Soarian angeschlossen, deren erhobene Daten sind schon jetzt zumindest teilweise auch in Soarian einsehbar. Speziell im Falle der Verwaltung und der Belegplanung der Operationssäle ist dies myMedis.

1.1.1.2. Über myMedis

MyMedis ist das modulare KIS des Entwicklers c.a.r.u.s. HMS GmbH. Da das UKE bereits über ein KIS verfügt, findet aus myMedis nur das Modul zur OP-Planung und /-Management Anwendung.

Alle Vorgänge in den Operationssälen im Gesamtkonzern des UKE werden über myMedis geplant, dokumentiert und abgerechnet. Die OP-Belegplanung ist hierbei ein kritischer Punkt, da sie durch die nötigen Vor-/ und Nachbereitungen, sowie unvorhersehbare Umstände laufend angepasst werden muss. Darüber hinaus müssen bestimmte Ruhezeiten für die operierenden Ärzte eingehalten werden. Des Weiteren müssen auch kurzfristig Säle für Notfalloperationen zur Verfügung stehen. Dies verzögert die vorgesehenen Operationen, sodass der Tagesablauf neu geplant und koordiniert werden muss. Diese Faktoren und unvorhersehbaren Ereignisse müssen durch die Software der OP-Verwaltung erfasst und gesteuert werden. Eine unflexible Planung führt gegebenenfalls zu Ausfällen oder Leerstand eines Saals, was sich direkt finanziell auf das Krankenhaus auswirkt.

Der OP-Plan von myMedis (siehe Abbildung 2) ermöglicht es den Anwenderinnen und Anwendern, sich einen genauen Überblick über die Belegung mehrerer OP-Säle zu verschaffen und deren Status einzusehen.

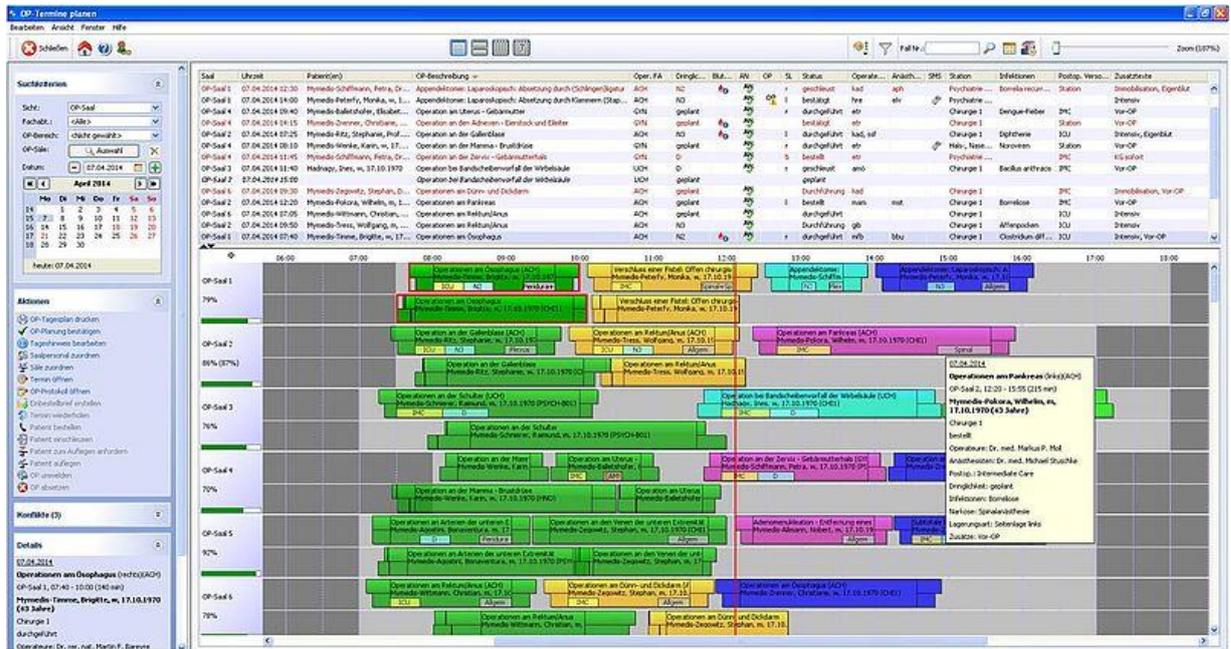


Abbildung 2: myMedis OP-Plan
 Quelle: „OP-Planung/-Management“, o.J.

1.2 Aufgabenstellung

Das Ziel dieser Thesis ist es, eine Analyse der Machbarkeit der Zusammenfügung und Darstellung von Daten aus verschiedenen medizinischen IT-Systemen anzufertigen.

Obwohl in dieser Thesis nur auf Soarian und myMedis eingegangen wird, umfasst die IT-Landschaft des Gesamtkonzern UKE eine Vielzahl an Systemen. Durch die hohe Anzahl sehr unterschiedlicher IT-Systeme im UKE,⁶ besteht gerade bei Patienten mit einem komplizierten Krankheitsbild die Möglichkeit, dass nicht alle Daten im selben System zugänglich sind. Um eine übergreifende Auswertung der Daten zu erstellen, sind Mitarbeiterinnen und Mitarbeiter also zum Teil gezwungen, mehrere Systeme gleichzeitig zu bedienen. Eine quantitative, systemübergreifende Analyse mehrerer Patienten kann es unter Umständen erforderlich machen, dass Daten aus den einzelnen Systemen exportiert werden müssen, um sie in Relation zueinander zu stellen.

Im Rahmen dieser Thesis soll analysiert werden, ob ein übergreifendes System entwickelt werden kann, das auf die Datenquellen mehrerer anderer Systeme (im besten Fall aller Systeme, die im UKE eingesetzt werden) zugreifen und deren Daten zur Ansicht und Aufarbeitung zur Verfügung stellen kann. Wie bereits beschrieben wäre dies eine erhebliche Arbeitserleichterung und ermöglicht es unter Umständen, neue Visualisierungen der Daten zu erstellen, die speziell auf den Arbeitsablauf einer bestimmten Personengruppe innerhalb des UKE abgestimmt sind und somit ihre Produktivität deutlich steigern könnten.

Um den Umfang der Thesis einzuschränken, konzentriert sie sich zunächst ausschließlich auf die Datengrundlage der elektronischen Patientenakte und der Software zur Planung und Dokumentation von Operationen im AKK und im Zentral-OP des UKE. Neben der Eingrenzung der Datengrundlage bietet dies den Vorteil, dass im Bereich der OP-Planung bereits Interesse

⁶ Für ausführlichere Informationen siehe 1.1.1 – IT-Systeme im Universitätsklinikum Hamburg-Eppendorf.

an einem neuen System besteht, das ergänzend zu der zurzeit eingesetzten Software verwendet werden und somit aus dem Prototyp direkt ein System entwickelt werden kann, das den Ansprüchen der Mitarbeiterinnen und Mitarbeiter der OP-Säle gerecht wird.

Um die angestrebte Darstellung für den Einsatz in einem OP-Saal zu optimieren, ist allerdings nicht nur ein Feedback der dort Tätigen sinnvoll, sondern auch die Einhaltung grundlegender Erkenntnisse aus dem Bereich der Softwareergonomie.

2 Hauptteil

2.1 Anforderungen

2.1.1 Interview

Da sich diese Abschlussarbeit hauptsächlich mit der Optimierung der Arbeitsabläufe der Mitarbeiterinnen und Mitarbeiter in den OP-Sälen der verschiedenen Fachabteilungen des UKE befasst, hat zu Beginn der Konzeptionierung ein Treffen mit Frau Heike Jipp, die im AKK im Qualitätsmanagement arbeitet und mit Herrn Dr. Philip Gabriel, der am UKE in der Anästhesiologie tätig ist, stattgefunden. Frau Jipp und Herr Dr. Gabriel haben die nötige Erfahrung in der Organisation der OP-Säle und mit den Arbeitsabläufen der Mitarbeiterinnen und Mitarbeiter, um eine qualifizierte Aussage über deren Arbeitsgewohnheiten und Anforderungen treffen zu können.

Im Gespräch haben sich folgende Kernpunkte herauskristallisiert, die in der Konzepterarbeitung besonders berücksichtigt werden sollten:⁷

- Die Darstellung sollte für die unterschiedlichen Operationssäle anpassbar sein. Eventuell sollten diese Einstellungen sogar jedem Anwender einzeln zur Verfügung stehen
- Der Aufruf sollte über myMedis erfolgen, um den Aufwand für den Anwender zu begrenzen

⁷ Jipp & Dr. Gabriel, 2015.

- Die Raum-Wahl bei der Ein- bzw. Ausleitung⁸ ist im AKK strikt vorgeschrieben und kann somit in der Software mit erfasst werden. Im UKE ist sie jedoch flexibel und wird elektronisch nicht erfasst
- Anzeigen von Informationen aus dem Krankenhauslogistik-System SyncroTESS⁹
 - Wie ist der Status des Patienten, wird er rechtzeitig für die Operation vorbereitet sein oder verzögert sie sich?
- Stehen für den Patienten ausreichend Blutkonserven zur Verfügung, Anforderungen müssen in diesem Bereich teilweise noch über das Telefon gestellt werden.

2.2 Design-Konzeption

Für die Entwicklung eines ersten Prototyps, der auf die Datenbanken der Testsysteme zugreifen kann, ist ein Webserver notwendig. Die Bereitstellung eines Webservers im Netzwerk des KIS erfordert spezielle Berechtigungen und kann daher nur durch den IT-Infrastruktur-Dienst des UKE erfolgen. Durch die hohe Arbeitslast des Netzwerkdienstes, hat die Einrichtung aber erst zu einem sehr späten Zeitpunkt stattgefunden, sodass der benötigte Webserver erst gegen Ende der Bearbeitungszeit zur Verfügung stand, obwohl er bereits zu Beginn der Thesis beantragt wurde.

⁸ In der Einleitung werden die letzten Vorbereitungen am Patienten getroffen, bevor dieser in den Operationssaal transportiert wird und die Operation beginnt. In der Ausleitung findet dem entsprechend die Nachbereitung statt.

⁹ SyncroTESS ist eine Logistiksoftware der INFORM GmbH. Mit ihr werden im UKE Patiententransporte geplant, um Patienten in andere Fachabteilungen oder OP-Säle zu verlegen. Darüber hinaus regelt sie auch Warentransporte wie z.B. Blutkonserven und Operationsbesteck.

Um trotz des fehlenden Webservers erste Überlegungen bezüglich des Konzeptes anzustellen, wurde auf einem vom System entkoppelten Computer eine XAMPP-Installation angelegt. „XAMPP ist eine [...] leicht zu installierende Apache-Distribution, die MariaDB, Hypertext Preprocessor PHP und Perl enthält“.¹⁰ XAMPP ermöglicht es einen Webserver unkompliziert auf einem Arbeitsplatzrechner zu betreiben. Ein dedizierter Server ist also nicht nötig, und bietet so gerade zur Entwicklung eine unkomplizierte Umgebung, die unabhängig vom Netz des UKE und durch die Installation auf einem Laptop auch mobil ist.

Für die graphische Darstellung der Patientendaten wurden auf dem provisorischen Webserver WordPress und Joomla installiert, die zu den gängigsten Content-Management-Systemen (CMS) gehören und somit dazu dienen die Anordnung von Daten auf einer Website zu vereinheitlichen und erheblich zu erleichtern. CMS bieten gerade bei wiederkehrenden Datenstrukturen, wie z.B. Foren oder Blogs, den deutlichen Vorteil, dass die Darstellung durch die Konfiguration vorgegeben wird und gleichartige Daten, die in diesem Fall z.B. von einem anderen Patienten oder aus einer anderen Station stammen, ohne größeren Aufwand in die Website integriert werden können.

Nach der Konfiguration der CMS stellte sich allerdings heraus, dass eine Anpassung an die vorliegenden Anforderungen sehr viel aufwendiger ist, als zu Beginn vermutet wurde. So ist weder WordPress, noch Joomla in ihrer Standard-Konfiguration dazu in der Lage, externe Daten aus einer Datenbank auszulesen. Beide Systeme sind darauf spezialisiert Texte und Bilder anzuzeigen, die über die integrierte Oberfläche eingegeben worden sind. Die Darstellung von Daten, die aus einer Datenbankabfrage gewonnen werden, ist zwar möglich

¹⁰ „XAMPP Installers and Downloads for Apache Friends“, o.J.

aber nicht praktikabel¹¹, da sie schlecht zu verwalten ist oder das CMS eine Darstellungsweise vorgibt, die für den angestrebten Verwendungszweck nicht geeignet ist.¹²

Das Konzept zur Darstellung durch ein CMS ist nach diesen ersten Versuchen komplett verworfen worden. Die Extraktion der Daten aus den verschiedenen Systemen, sowie die Darstellung auf der Website muss also grundlegend neu programmiert werden. Die Darstellung für einen ersten Prototypen, der für die Erstellung der Machbarkeitsanalyse ausreicht, kann allerdings sehr schlicht gehalten werden. Für einen produktiven Einsatz im UKE wäre eine visuell aussagekräftige und ansprechende Website sehr vorteilhaft, damit die Anwenderinnen und Anwender nicht durch das Erscheinungsbild und die Menge an Informationen überfordert werden. Für die Machbarkeitsanalyse des grundlegenden Konzeptes der systemübergreifenden Darstellung von Daten ist dies allerdings nachrangig.

¹¹ WordPress ermöglicht es Datenbank-Queries über eine Erweiterung auszuführen, indem man die gewünschten Queries in den Einstellungen hinterlegt und deren Ergebnisse mittels ihrer ID in eigene Texte einbaut. Dieses ist allerdings sehr unflexibel und für Personen, die nicht seit Anfang der Entwicklung involviert sind, sehr schwer nachzuvollziehen.

¹² Durch eine Erweiterung lässt sich in WordPress der gesamte Inhalt einer Datenbanktabelle auf einer Seite anzeigen. Diese Ausgabe kann allerdings nicht gefiltert werden, es werden also alle Tupel in einer Tabelle angezeigt, was die Übersichtlichkeit stark beeinträchtigt.

2.3 Konzept zur Implementation

2.3.1 Hardware-Voraussetzungen

Die Hauptaufgabe des in Abschnitt **2.2 – Design-Konzeption** beschriebenen Webservers ist es, den Anwenderinnen und Anwendern Informationen über das Intranet des Konzern UKE bereitzustellen. Des Weiteren werden im ersten Schritt über ihn Datenbankabfragen auf einem Microsoft SQL Server 2014 und einer Oracle Database 11g gestartet, deren Ergebnisse gegebenenfalls auf einer niedrigen Stufe aufzubereiten sind. Hinzu kommt noch eine Art Benutzerkontenverwaltung auf niedrigem Niveau¹³. Da die OP-Plan-Visualisierung in ihrer ersten Entwicklungsphase auf einige wenige Anwender in den OP-Sälen des UKE und AKK beschränkt ist, muss hier auch nicht mit einer höheren Last durch hohe Zugriffszahlen gerechnet werden.

Für die Entwicklung eines ersten Prototyps reichen die Mindestanforderungen des Windows Server 2012 aus, die sich wie folgt ergeben¹⁴:

- Prozessor:
 - Mindestens 1,4-GHz-Prozessor mit 64 Bit
- RAM:
 - Mindestens 512 MB
- Speicherplatzanforderungen:
 - 32 GB

¹³ Nähere Informationen hierzu können dem Abschnitt **2.3.2 – Software-Voraussetzungen** entnommen werden.

¹⁴ Vergleich „Systemanforderungen und Installationsinformationen für Windows Server 2012 R2, o.J.“

Da sich der Aufgabenbereich des Servers aber gegebenenfalls während der Entwicklung um Abfragen auf den Datenbanken weiterer Systeme erweitern kann und auch die Zahl der Anwenderinnen und Anwender schwer vorherzusehen ist, muss die Leistung des Servers dementsprechend viel Puffer bieten. Für diesen Zweck wird keine neue Hardware bereitgestellt, sondern ein virtueller Server auf einem VMware Server¹⁵ eingerichtet. Dies bietet den großen Vorteil, dass für dieses Projekt ein neuer Server bereitgestellt werden kann, ohne zusätzliche Hardware anzuschaffen und ohne zusätzlichen Aufwand zur Integrierung und anschließenden Wartung in den Serverräumen des UKE zu verursachen.

Des Weiteren erfordert die Einrichtung des Servers die Installation und Konfiguration neuer Software; ein dedizierter Server ermöglicht dies ohne Einfluss auf andere Systeme auszuüben. Eine Störung der Arbeitsabläufe der Anwenderinnen und Anwender wird somit nicht riskiert.

2.3.2 Software-Voraussetzungen

Das im UKE verwendete klinische Arbeitsplatzsystem (KAS) bietet den großen Vorteil, dass jeder Benutzer sich an jedem Standard-Arbeitsplatz auf dem Gelände des UKE anmelden und von dort aus auf alle Systeme zugreifen kann, die er benötigt. Um dies zu bewerkstelligen – insbesondere, um viele einzelne Installationen zu vermeiden – sind viele der Systeme webbasiert. Für die OP-Plan-Visualisierung bietet der Verzicht auf einen eigenen Klienten den Vorteil, dass die Darstellung der Daten beim Anwender auf bereits etablierten Wegen erfolgen kann. So wird der für eine Webanwendung benötigte Internet Explorer zum Beispiel bereits durch Soarian verwendet. Aus diesem Grund findet eine regelmäßige Aktualisierung des

¹⁵ Mit Hilfe eines VMware Servers lassen sich virtuelle Rechner generieren, die sich die Hardware des VMware Servers teilen.

Internet Explorers bereits statt und für die OP-Plan-Visualisierung muss er nicht zusätzlich gewartet werden, es bleibt also nur die Betreuung des dedizierten Webservers.

Die Systeme des UKE basieren weitestgehend auf Windows Server 2008 und Windows Server 2012 Installationen. Die Wahl des Betriebssystems für den Server dieses Projekts fiel auf Windows Server 2012 R2 ohne dabei besondere Anforderungen zu beachten, beide Versionen sind dafür geeignet, die erforderlichen Rollen zu erfüllen.

2.3.3 Sicherheit

Da es sich bei den Daten, die für dieses Projekt relevant sind, um medizinische Daten von Patienten und somit um personenbezogene Daten besonderer Art handelt, muss ein besonderer Augenmerk auf den Datenschutz gelegt werden.

Die elektronische Patientenakte des UKE ist bereits durch das Bundesamt für Sicherheit in der Informationstechnologie (BSI) nach ISO 27001 zertifiziert und bietet somit nachweislich einen hohen Schutz für die Daten der Patienten.

Um diese Sicherheitsstandards auch in der neuen Software beizubehalten, muss dies direkt bei der Konzeptionierung in Betracht gezogen werden.

Einen wichtigen Punkt bildet auch die Benutzerverwaltung. In Bezug auf den Datenschutz muss durch sie gewährleistet werden, dass keine unbefugten Personen Zugang zum System und somit zu den Patientendaten haben. Andererseits bedeutet sie aber einen erhöhten Aufwand, was das Anlegen und Sperren von Anwendern, das Zurücksetzen und den Versand von Anmeldeinformationen betrifft. Da sich die Mitarbeiter im UKE schon durch die hohe Anzahl der verschiedenen IT-Systeme viele Kennungen und Passwörter merken müssen, bietet es sich hier an, das System an ein anderes zu koppeln und somit die Anzahl der Anmeldeinformationen der Anwenderinnen und Anwender nicht weiter zu erhöhen.

Zu diesem Zweck wäre es möglich, den Befehl zum Aufruf der OP-Plan-Visualisierung nur den Benutzern anzuzeigen, die eine Berechtigung dafür haben. Da der Plan aber durch einen Webserver dargestellt werden soll, könnte der Aufruf auch durch den Anwender manuell erfolgen, womit er dieses Sicherheitssystem umgangen hätte. Um eine zusätzliche Zugangskontrolle durchzuführen, könnte beispielsweise in einer Parameterübergabe ein Hashwert übergeben werden, der aus dem Benutzernamen errechnet wird. Zugang haben somit also nur die Anwender, die die Zugangsdaten berechtigter Benutzerkonten aus dem Ursprungssystem haben. Gleichzeitig entfällt für die OP-Plan-Visualisierung jegliche Form der Passwortvergabe oder Synchronisation von Zugangsdaten mit anderen Systemen. Es müssen lediglich die berechtigten Anwender hinterlegt und die gelöscht werden, die weiterhin im UKE tätig sind aber keinen Zugriff mehr auf die OP-Plan-Visualisierung haben sollen.

Neben dem Datenschutz ist auch die Datenintegrität ein ausgesprochen wichtiger Faktor für das Ergebnis dieses Projekts. Es muss sichergestellt werden, dass die im neuen System angezeigten Daten auch tatsächlich den Daten entsprechen, die in Soarian und myMedis dokumentiert worden sind. Eine Abweichung kann im schlimmsten Falle eine Gefährdung für den Patienten bedeuten. Es ist auch es wichtig, dass die Datenintegrität der anderen Systeme nicht beeinflusst werden kann. Da weder Soarian noch myMedis eine entsprechende Schnittstelle anbieten, um Daten an andere Systeme weiterzuleiten oder zu empfangen, muss der Zugriff direkt über die Datenbank erfolgen.¹⁶ Dies birgt allerdings das Risiko, durch einen fehlerhaften Eintrag in die Datenbanken, die bestehenden Systeme zu korrumpieren und somit in ihrer Funktionsweise zu stören. Da das Verhalten der Systeme auf eine Änderung ihrer Datengrundlage innerhalb dieses Projekts nicht vorhersehbar ist, darf ein Zugriff ausschließlich lesend erfolgen.

¹⁶ Lenser, 2016.

2.3.4 Auswahl der Programmiersprache

Zur Erfüllung der zu erfüllenden Aufgabenstellung eignen sich mehrere Programmiersprachen. Um eine geeignete Programmiersprache für die Verwendung in diesem Projekt auszuwählen, werden die drei Sprachen untersucht, die im Bereich der Webentwicklung die weiteste Verbreitung haben. Namentlich sind das PHP, Microsofts Active Server Pages .NET (ASP.NET oder kurz ASP) und Java (Java Server Pages, kurz JSP), wie der Abbildung 3 entnommen werden kann.

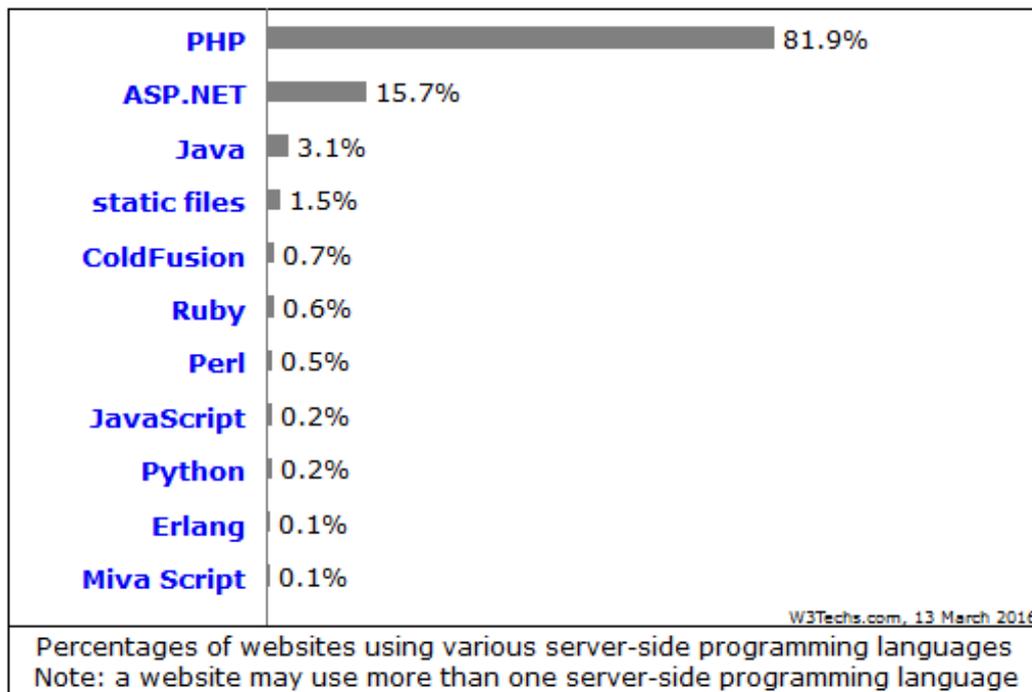


Abbildung 3: W3Techs.com Marktanteil serverseitiger Programmiersprachen

Quelle: „Usage Statistics and Market Share of Server-side Programming Languages for Websites“, 2016.

Die von W3Techs.com erhobene Statistik zeigt innerhalb der Webentwicklung einen deutlichen Trend hin zu PHP und ASP. JSP gehört mit einem Marktanteil von 3,1% eher zu den weniger verbreiteten Programmiersprachen.

Ein gravierender Faktor in der Softwareentwicklung ist die Erfahrung der Entwicklerinnen und Entwickler. Zwar gehen dieser Thesis bereits Entwicklungen in PHP, Java und C#, das bei ASP zum Einsatz kommt, voraus, speziell im Bereich der Webentwicklung konzentriert sich die Erfahrung allerdings auf PHP und JSP. In Betracht gezogen werden muss also auch, dass einer möglichen Entwicklung in ASP auf jeden Fall eine Lernphase vorausgehen muss, die in den Zeitplan integriert werden muss. Da es im Gesamtkonzern UKE bereits vergleichbare Projekte gibt, die mittels PHP umgesetzt wurden, kann die Verwendung von PHP auch in diesem Projekt ein großer Vorteil sein, da das Knowhow bereits bei den Mitarbeiterinnen und Mitarbeitern des UKE vorhanden ist. Eine Beteiligung neuer betreuenden Personen am Projekt könnte so bei Bedarf ohne viel vorhergehenden Zeitaufwand erfolgen.

Da sich jede der drei untersuchten Programmiersprachen ohne Einschränkungen für die Erfüllung des Projekts eignet, bleibt die Erfahrung der Programmiererinnen und Programmierer der einzige relevante Faktor. Der Datenbankzugriff und die Darstellung erfolgt somit mittels PHP.

2.3.5 Wartung und Erweiterung

Obwohl es sich bei diesem Projekt in erster Linie um eine Machbarkeitsanalyse handelt, die der Entwicklung eines Systems zur systemübergreifenden Daten-Verarbeitung zu Grunde liegen soll, können bereits bei der Entwicklung des Prototyps Komponenten entwickelt werden, die anschließend produktive System einfließen sollen. Neben dem fertig eingerichteten Webserver betrifft das auch Softwarekomponenten, die eigens dafür geschrieben worden sind, um das erstellte Konzept zu beweisen. Um für das anschließende produktive System zu gewährleisten, dass es gewartet und erweitert werden kann, müssen bei der Programmierung etablierte Grundsätze berücksichtigt werden.

Zu den wichtigsten dieser Grundsätze gehört die objektorientierte Programmierung. Da es sich bei PHP 5 nicht grundlegend um eine objektorientierte Programmiersprache handelt, kann der Programmcode auch prozedural verfasst werden. Es gibt zwar nachvollziehbare Gründe Programmcode prozedural zu schreiben, um zum Beispiel kurze Funktionen vor der endgültigen Implementierung zu testen. Objektorientierte Programmierung ist zwar in der Regel mit etwas mehr Aufwand verbunden, für den produktiven Einsatz ist sie allerdings im Hinblick auf die Wartung und Erweiterung dringend zu empfehlen.

Die Wartung wird beispielsweise durch den Einsatz objektorientierter Programmierung erheblich erleichtert, da er die Übersichtlichkeit durch Kapselung¹⁷ des Programmcodes in verschiedene Klassen erhöht. Ein weiterer Vorteil der Kapselung ist es, dass bei einer Erweiterung des Funktionsumfangs die bereits bestehenden Funktionen einzelner Klassen genutzt werden können und somit redundanter Programmcode vermieden wird. Dies hat den Vorteil, dass die Anzahl an Zeilen Programmcode verringert wird, was wiederum die Übersichtlichkeit erhöht. Außerdem muss im Falle einer Änderung diese in der Regel nur in einer Klasse durchgeführt werden, da andere Klassen die durch die zu ändernde Klasse bereitgestellten Funktionen nutzen.¹⁸

Ein wichtiger Faktor hierbei ist es auch, dass häufig nicht nur eine Person an der Entwicklung eines Systems beteiligt ist. Die entwickelnden Personen sollten sich zwar zum Zeitpunkt der Entwicklung über die einzelnen Klassen und Funktionen ihres Programmcodes im Klaren sein, oftmals ist ein solches System aber über Jahre oder Jahrzehnte im Einsatz, sodass sich die

¹⁷ Kapselung dient in der Programmierung vor allem der Abschottung von Daten eines Objekts, damit sie nur über die zur Verfügung gestellten Methoden abgefragt und manipuliert werden können. Dies dient der Sicherheit und erhöht im gleichen Zug die Übersichtlichkeit, da die Kommunikation der Objekte nur über vorgeschriebene Wege erfolgt.

¹⁸ Arndt, Hermanns, Prof. Dr. Kuchen, & Poldner: Best Practices in der Softwareentwicklung, Münster, 2009, S.25 f.

Entwicklerinnen und Entwickler wieder neu im Programmcode zurechtfinden müssen. Auch ist es durchaus denkbar, dass sich über die Lebensspanne des Systems die Betreuer durch personelle Änderungen oder Vergrößerung des Personalstabs ändern. Um diesen Mitarbeiterinnen und Mitarbeitern die Einarbeitung in die grundlegenden Funktionsweisen des Systems zu erleichtern, sollte über die Anwendung der objektorientierten Programmierung eine Dokumentation der einzelnen Komponenten und die Generierung einiger üblicher Diagramme aus der Unified Modeling Language (UML)¹⁹ in Betracht gezogen werden.

Auf das vorliegende Projekt bezogen bietet es sich zum Beispiel an, die Kommunikation mit den Datenbankservern über einzelne Klassen zu realisieren, die entsprechend auf die verschiedenen Datenbanken angepasst sind. Dies betrifft auf der einen Seite die Verbindungsdaten, wie Datenbankserver, Benutzerkennung und Passwort, auf der anderen Seite aber auch den Aufbau der Datenbanken selbst. So ist es für die Klasse zur Verbindung mit der Soarian Datenbank denkbar, dass sie eine Funktion aufweist, die sämtliche Stammdaten eines Patienten, wie Name, Geburtsdatum und Adresse ausliest. Die Klasse, die auf die Datenbank von myMedis zugreift hingegen, muss hauptsächlich die Daten zu Operationen abfragen, eine Unterteilung in verschiedene Klassen ergibt hierbei also durchaus Sinn.

¹⁹ „Die UML stellt [...] insgesamt sechs Strukturdiagramme zur Beschreibung statischer Strukturen, wie z.B. der Beziehungen zwischen Objekten, Klassen und Paketen [...] zur Verfügung.“ (Arndt, Hermanns, Prof. Dr. Kuchen, & Poldner: Best Practices in der Softwareentwicklung, Münster, 2009, S.28)

2.4 Praxis

2.4.1 Server-Konfiguration

Wie bereits beschrieben, erfolgt die Hardware-Konfiguration der UKE-internen Server über eine eigene Abteilung, sodass zum Beispiel die Konfiguration des Subnetzes²⁰ beantragt werden muss und nicht am Server direkt bearbeitet werden kann. Da dieses Projekt hauptsächlich an Soarian gebunden ist, ist der Webserver dem Soarian-Subnetz zugeteilt, was den Zugriff auf dessen Datenbanken ermöglicht. Um auch auf die Datenbank von myMedis zugreifen zu können, wurde eine Portfreigabe eingerichtet, die es speziell dem Webserver erlaubt auf den Port²¹ des myMedis-Servers zuzugreifen, über den diese Datenbank zu erreichen ist. Diese Praxis macht die Einrichtung eines neuen Servers etwas komplizierter, bietet aber einen zusätzlichen Schutz vor dem Zugriff Unbefugter auf die jeweiligen Server. Des Weiteren gilt für den Webserver auch die UKE-weite Richtlinie, dass Webserver nur über das Hypertext Transfer Protocol Secure (HTTPS) erreichbar sein dürfen, deren Kommunikation also prinzipiell verschlüsselt erfolgt.

Da es sich bei dem Webserver, wie bereits beschrieben, um einen virtuellen Server handelt, erfolgt der Zugriff darauf über das Remote Desktop Protocol. Über diese Verbindung kann die Steuerung des Servers über einen Arbeitsplatzrechner desselben Netzwerkes erfolgen. Die Benutzeroberfläche unterscheidet sich rein optisch nur im Detail von der eines Windows-

²⁰ Ein Subnetz definiert einen Teilbereich des Gesamt-Netzwerks des UKE. Ohne weitere Einstellungen ist es für einen Rechner nicht möglich, auf einen anderen Rechner oder Server zuzugreifen, der sich außerhalb seines Subnetzes befindet.

²¹ „[...] Um den konkreten (technischen) Dienst, der genutzt bzw. angesprochen wird, zu identifizieren, besteht unter TCP (und UDP) die Notwendigkeit, über Port-Nummern eine exaktere Identifikation zu ermöglichen. [...]“ siehe Hübscher, Petersen, Rathgeber, Richter, & Dr. Scharf: IT-Handbuch IT-Systemelektroniker/-in Fachinformatiker/-in, Braunschweig, 2009, S.322.

Arbeitsplatzrechners. Die Konfiguration des Servers kann über den sogenannten Server-Manager erfolgen, der sich automatisch nach der ersten Inbetriebnahme des Servers startet. Der Server-Manager als Werkzeug ermöglicht es verschiedene angeschlossene Server zu konfigurieren und gegebenenfalls zu einem Cluster²² zu verbinden. Da allerdings nur der Webserver allein benötigt wird und auch kein anderer Server freigegeben ist, steht nur die Konfiguration des Webserver selbst zur Verfügung.

Über den Server-Manager wurden auf dem Server die Internet Information Services(IIS) eingerichtet, Microsofts Implementation eines Webserver. Diese Einrichtung ermöglicht es, mittels eines Webbrowsers²³ auf den Server zuzugreifen, um eine Website angezeigt zu bekommen. In seiner Standard-Konfiguration ist IIS nur in der Lage, Webseiten mittels der Hypertext Markup Language (HTML) darzustellen. HTML lässt sich dazu nutzen, den Inhalt einer Webseite, also Texte, Tabellen und Bilder zu strukturieren. Die Daten aus einer Datenbank abfragen kann es allerdings nicht, sodass hierzu ein zusätzliches Programm benötigt wird. Die Wahl ist dabei auf PHP gefallen. Da es sich bei PHP um eine weit verbreitete Sprache handelt, werden die Mitarbeiterinnen und Mitarbeitern des UKE das System verstehen und weiter entwickeln können. Die einzelnen Schritte der PHP-Installation erfolgten nach dem offiziellen Handbuch der PHP Group zur Installation von PHP unter Microsoft IIS 7.0 oder höher und können diesem entnommen werden.²⁴

Obwohl zum Zeitpunkt der Installation bereits PHP 7 veröffentlicht war, ist die Wahl dennoch auf PHP 5.6.18 gefallen. Grund dafür ist die Microsofteigene PHP-Erweiterung, „Microsoft

²² Ein Cluster ist in diesem Zusammenhang ein Verbund mehrerer Server, zur Erfüllung einer Aufgabe. Ein Datenbank-Cluster ermöglicht es zum Beispiel, die Auslastung aufzuteilen und somit die Performanz zu erhöhen.

²³ Ein Webbrowser ermöglicht es den Benutzerinnen und Benutzern, über das Internet – oder in diesem Fall das lokale Netzwerk, auch Intranet genannt – Webseiten anzeigen zu lassen. Zu den am weitest verbreiteten Webbrowsern gehört der Internet Explorer von Microsoft, der im Gesamtkonzern des UKE der Standard ist.

²⁴ Vergleiche: „Microsoft IS 7.0 and later – Manual“, o.J.

Treiber für PHP für SQL Server“. Diese Erweiterung ermöglicht es, mittels vordefinierten PHP-Funktionen auf eine SQL-Datenbank eines Microsoft SQL-Server zuzugreifen. Durch die Verwendung von Microsofts Treibern ist sichergestellt, dass der Zugriff geregelt abläuft. Darüber hinaus ist die Umsetzung unkompliziert, da die gewünschte Erweiterung nur mittels Befehl in der Konfigurationsdatei von PHP geladen werden muss und somit wenig Eigenentwicklung in Anspruch nimmt. Die neuste Version dieser Erweiterung ist 3.2. Sie unterstützt offiziell nur PHP 5.6²⁵ und gibt somit die zu verwendende PHP-Version vor. Für die Datenbank von myMedis ist dies allerdings nicht anwendbar, da es sich dabei um eine Datenbank von Oracle handelt. Da PHP aber bereits Funktionen zur Kommunikation mit einer Oracle-Datenbank mitbringt, muss hier keine weitere Erweiterung eingebracht werden.

Damit der Webserver die Zugriffe aus PHP auf die Datenbanken richtig verarbeiten kann, müssen darüber hinaus die entsprechenden Treiber installiert sein. Für die Oracle und die Microsoft SQL Datenbank sind das der Microsoft ODBC Driver, beziehungsweise der Microsoft SQL Server Native Client. Diese beiden Treiber setzen außerdem voraus, dass auf dem Server bestimmte Laufzeitkomponenten installiert sind. Bei diesen Komponenten handelt es sich um die Microsoft Visual C++ Redistributables²⁶, die auf dem Webserver in mehrfacher Ausführung benötigt werden, da die verschiedenen Treiber spezielle Versionen voraussetzen. Zusätzlich zu den benötigten Versionen wurde auch die aktuelle Version von 2015 installiert, da ein

²⁵ Siehe Details unter: „Download Microsoft Driver for PHP for SQL Server from Official Microsoft Download Center“, o.J.

²⁶ „Die Visual C++ Redistributable Packages installieren Laufzeitkomponenten von Visual C++-Bibliotheken. Diese Komponenten sind zur Ausführung von C++-Anwendungen erforderlich, die mit Visual Studio [...] entwickelt wurden und dynamisch mit Visual C++-Bibliotheken verknüpft sind.“ siehe „Download Visual C++ Redistributable für Visual Studio 2015 from Official Microsoft Download Center“ o.J.

Treiber diese eventuell bevorzugen könnte²⁷ und sie vermeintlich die zuverlässigste und sicherste ist.

2.4.2 Datenbankzugriff mittels PHP

Da sich das Projekt zu Beginn der Programmierung nicht mehr im Zeitplan befand, wurde Prototyping als Vorgangsmodell gewählt. Dies bot den Vorteil, dass sich schnell Ergebnisse erzielen ließen, anhand derer ein Fazit für diese Thesis abgeleitet werden konnte. Für die Durchführung einer Machbarkeitsanalyse war dieses Vorgehen sehr gut geeignet, da ohne viele Umwege die Grundlagen überprüft werden konnten, auf denen die Funktionen des Systems basieren. Des Weiteren konnten Teile des Programmcodes zur Entwicklung des produktiven Systems übernommen werden, dies konnte die für den Prototypen aufgewandte Zeit zum Teil ausgleichen.

Wie bereits beschrieben erfolgt der Zugriff auf die Datenbanken von Soarian und myMedis mittels PHP. Um den zuvor in Abschnitt **2.3.4 – Wartung und Erweiterung** festgelegten Ansprüchen gerecht zu werden, wurde jeweils eine Klasse zur Kommunikation mit der Soarian- und der myMedis-Datenbank konzipiert. Da die Entwicklung des Systems prinzipiell in den Testsystemen abläuft, bietet es sich an, in der Entwicklung des Programmes dem Data Access Objekt Entwurfsmuster zu folgen.²⁸ Nach diesem Entwurfsmuster erfolgt die Kommunikation mit den Datenbanken über eigens dafür erstellte Klassen, die nur diesen einen Zweck erfüllen. Dies hat den großen Vorteil, dass bei einem Umzug auf eine andere Datenquelle, wie zum Beispiel auf das produktive System des UKE, Änderungen nur an diesen Klassen durchgeführt

²⁷ Obwohl für viele Programme eine Version von Visual C++ Redistributable vorgegeben ist, können sie auch so konfiguriert sein, dass sie automatisch die neueste nutzen.

²⁸ Siehe "Core J2EE Patterns – Data Access Object", o.J.

werden müssen. Selbst falls sich die Struktur der Datenbank grundlegend ändern sollte, müsste in der neuen Klasse nur die Methode zur Datenabfrage überschrieben werden, damit sie die Daten in der gewohnten Form an die darstellenden Objekte übergeben kann. Dies ist in den Punkten der Wartung, der Erweiterung und der Übersichtlichkeit ein deutlicher Gewinn.

Da die Verbindung zu den Datenbanken und die Abfragen auf ihnen über die integrierten Funktionen von PHP 5 oder eine installierte Erweiterung erfolgen, ist der Programmieraufwand sehr viel geringer, als es bei einer komplett eigenen Implementation dieser Funktionen gewesen wäre. Prinzipiell folgen die Zugriffe der Dokumentation über OCI8²⁹ und SQLSRV³⁰ von The PHP Group. Der Programmcode der beiden Klassen zum Zugriff auf die Datenbanken kann dem Anhang unter Abschnitt **6.1. – Programmcode** entnommen werden.

Die SQL-Queries zur Abfrage der Daten aus den Datenbanken sind für sich genommen sehr umfangreich und wichtig, bieten für die Leserinnen und Leser aber keine inhaltlich neuen Aspekte. Für den Beleg der Funktion des Konzepts wurde eine Abfrage zu allen Patienten-Stammdaten aus Soarian und eine Abfrage zu den Daten der letzten OP aus myMedis geschrieben. Modelle der betreffenden Datenbanken können dem Anhang unter Abschnitt **6.3 – Datenbank-Schemata** entnommen werden.

²⁹ Vergleiche: „PHP: OCI8 Funktionen – Manual“, o.J.

³⁰ Vergleiche: „PHP: SQLSRV – Manual“, o.J.

Der Webserver liefert auf die Eingabe einer Fallnummer hin folgendes Ergebnis:

Daten aus Soarian	Daten aus myMedis
Fallnummer: 395792544	Fallnummer: 395792544
Patientennummer: 55040456	
internePatientennummer: 2016000387	
VIP: nein	
Registrationsdatum: 16.02.2016	Aufnahmedatum: 16.02.16 13:02:20
Vorname: ECP ist Mega	Vorname: ECP ist Mega
Nachname: T3-Soarian	Nachname: T3-Soarian
Land: Deutschland	
Geschlecht: männlich	Geschlecht: m??nnlich
Nationalitaet: DE	
Geburtsdatum: 01.02.2016	Geburtsdatum: 01-FEB-16
	Aufnahmegrund 1&2: Krankenhausbehandlung, vollstation??r
	Aufnahmegrund 3&4: Normalfall
	OP-Bezeichnung: Cross-Linking
	OP-Beginn: 07.03.16 13:03:00
	OP-Ende: 07.03.16 14:03:00

Abbildung 4: Gegenüberstellung der Ausgaben der Datenbanken

Quelle: Eigene Darstellung

Das Abfragen von Daten auf den Datenbanken und deren Darstellung ist somit zweifelsfrei über den implementierten Weg zu erreichen. Im nächsten Schritt muss geklärt werden welche Daten für die Arbeit in den OP-Sälen des UKE und des AKK relevant sind und dargestellt werden sollten, die Machbarkeitsanalyse zum grundsätzlichen Konzept ist aber mit diesem Schritt abgeschlossen.

3 Zusammenfassung

In der Zusammenfassung werden die vorhergegangenen Kapitel noch einmal analysiert und es wird Bezug darauf genommen, inwiefern die erarbeiteten Konzepte in die Praxis umgesetzt werden konnten. Außerdem wird festgehalten welche unerwarteten Komplikationen dabei aufgetreten sind und in welcher Phase der Entwicklung das System sich zum Ende der Thesis befand. Anschließend wird ein Ausblick auf die dieser Thesis folgenden Schritte gegeben, die als Leitfaden dafür zu verstehen sind, wie aus dem Konzept und dem bestehenden Prototyp ein System entwickelt werden kann, das für den produktiven Einsatz geeignet ist.

3.1 Fazit

Das Projekt hat sich im abschließenden Überblick als voller Erfolg erwiesen. Zwar gab es innerhalb des Projekts Rückschläge aber wichtig ist es in erster Linie, dass ein praktikabler Weg zur Umsetzung gefunden wurde. Diese Thesis belegt zweifelsfrei die Machbarkeit des Konzepts zur Umsetzung und dokumentiert darüber hinaus den Weg dorthin zum späteren Nachvollziehen durch neue und bestehende Entwicklerinnen und Entwickler des Systems.

Bezüglich der Konzeptionierung des System-Designs muss festgehalten werden, dass die verschiedenen CMS im Vorfeld näher im Hinblick auf die Abfrage und die Darstellung von Daten aus externen Datenbanken hin untersucht hätten werden müssen. Zwar fielen die Untersuchung und der letztendliche Verwurf des Konzeptes nicht in die kritische Phase des Projekts aber sie hätten noch früher erfolgen und somit Zeit für andere Bereiche freigeben können. Einen deutlich größeren Einfluss auf die Zeitplanung hatte allerdings der Verzug bei der Bereitstellung des benötigten Webservers. Das er erst ca. einen Monat vor dem Abgabetermin dieser Thesis zur Verfügung gestellt wurde, hat er einen vermeidbaren

Zeitdruck auf den Verlauf des Projekts ausgeübt. Positiv sind hierbei die auf dem, mittels XAMPP eingerichteten provisorischen Webserver angestellten Versuche zur Programmierung zu betrachten. So konnten bereits einige Möglichkeiten abgeschätzt und bei der eigentlichen Programmierung Zeit gespart werden. Dennoch sollte, auch im Hinblick auf zukünftige Projekte, der Frage nachgegangen werden, wie es zu dieser Verzögerung kommen konnte und wie sie in Zukunft vermieden werden kann.

Der Zeitaufwand zur Konfiguration des Webserver wurde etwas unterschätzt, was die Installation und Inbetriebnahme der einzelnen Softwarekomponenten angeht. Da der Zeitaufwand allerdings nicht erheblich größer war als geplant und er nach eigenem Ermessen auf die fehlende Erfahrung mit einem Windows Server und den IIS 8.5 zurückzuführen ist, erscheint eine weitere Prüfung hier überflüssig.

PHP als Programmiersprache zu wählen hat sich im Gegensatz dazu als vollkommen richtig erwiesen. Die durch den Entwickler bereitgestellten Erweiterungen zum Datenbankzugriff und die dazugehörigen Dokumentationen haben sich als ausgesprochen hilfreich erwiesen. Die Programmierung in PHP hat sogar weniger Zeit in Anspruch genommen, als erwartet und so konnte bereits mehr Augenmerk auf die Datenbanken von Soarian und myMedis gerichtet werden. Dies ist gerade deshalb vorteilhaft, weil so schon wichtige Schritte unternommen werden konnten, die eine Voraussetzung für die weitere Entwicklung des Systems bis hin zum produktiven Einsatz im UKE sind.

3.2 Ausblick

Trotz des vorausgegangenen positiven Verlaufs dieses Projekts sind noch viele Punkte zu beachten, bis der Einsatz des neuen Systems im Geschäftsbetrieb des Gesamtkonzerns UKE möglich ist.

Im Bereich der Sicherheit sind noch Punkte der Verschlüsselung offen. Die Kommunikation mit dem Webserver erfolgt bereits mittels HTTPS, aber eine Zertifizierung des Servers hat noch nicht stattgefunden. Aktuell weist der Server sich über ein selbst ausgestelltes Zertifikat aus, was beim Zugriff der Mitarbeiterinnen und Mitarbeiter zu einer Fehlermeldung führt. Der Internet Explorer bietet der anwendenden Person an, diese Fehlermeldung zu ignorieren, was eine weitere Entwicklung und das Testen des Systems ermöglicht. Vor einem Einsatz in den Krankenhäusern muss aber dringend ein Zertifikat hinterlegt werden, das durch eine anerkannte Stelle ausgestellt wurde. Darüber hinaus muss die Zugriffskontrolle implementiert werden, sodass nur berechtigte Personen Zugriff auf das System haben und diese nur Daten von Patienten angezeigt bekommen können, die ihren Bereichen zugeteilt sind. Wie bereits beschrieben bietet es sich an, das System an Soarian zu koppeln und einen Parameter zu übergeben, der die Benutzerinnen und Benutzer ausweist. Diese Praxis findet auch schon in anderen Bereichen Anwendung und ist als Lösung somit etabliert.

Weiterhin ist es ausgesprochen wichtig, dass das System so abgesichert wird, dass Benutzerinnen und Benutzer keine eigenen Anweisungen auf der Datenbank ausführen können. Einerseits sollte zu diesem Zweck im System eine eigene Kennung hinterlegt werden, über die auf den Datenbanken nur ein lesender Zugriff erfolgen kann, andererseits muss das System gegenüber Angriffen wie SQL-Injection abgesichert werden. „Direkt SQL Command Injection ist eine Technik, wo ein Angreifer SQL Kommandos erstellt oder existierende verändert, um versteckte Daten sichtbar zu machen, wertvolle Daten zu überschreiben, oder

sogar gefährliche Kommandos auf Systemebene des Datenbank-Hosts auszuführen.“³¹ Eine Dokumentation zur Absicherung gegen SQL-Injection wird ebenfalls durch The PHP Group bereitgestellt.

Neben der Beachtung dieser technischen Aspekte müssen die Wünsche und Ansprüche, sowie die Grundlagen der Softwareergonomie in Betracht gezogen werden, um das System erfolgreich in den bestehenden Arbeitsablauf zu integrieren. Die im Abschnitt **2.1.1 – Interview** dokumentierten Anmerkungen von Frau Jipp und Herrn Dr. Gabriel sollten hierbei als richtungsweisend für den Funktionsumfang angesehen werden, eine Analyse bezüglich des Aufwands und der Möglichkeiten zur Umsetzung hat allerdings noch nicht stattgefunden. Im Rahmen des Interviews haben sich Frau Jipp und Herr Dr. Gabriel dazu bereit erklärt, auf Nachfrage weitere Hinweise für die Entwicklung zu liefern und in der Zukunft einsatzbereite Versionen zu testen und an ihre Mitarbeiterinnen und Mitarbeiter heranzuführen. Dieses Angebot sollte in Hinblick auf die Integration des Systems im UAE auf jeden Fall berücksichtigt werden; darauf einzugehen wird sich voraussichtlich als deutlicher Vorteil erweisen.

³¹ "PHP: SQL Injection – Manual", o.J.

4 Literatur- und Quellenverzeichnis

Arndt, C., Hermanns, C., Prof. Dr. Kuchen, H., & Poldner, M. (16. Februar 2009). Best Practices in der Softwareentwicklung. Münster, Nordrhein-Westfalen, Deutschland.

BitRock Inc. (o.J.). *XAMPP Installers and Downloads for Apache Friends*.

Abgerufen am 14. Februar 2016 von <https://www.apachefriends.org/de/index.html>

c.a.r.u.s. HMS GmbH. (o.J.). *c.a.r.u.s. HMS GmbH: OP-Planung/-Management*.

Abgerufen am 14. Februar 2016 von <https://www.carus-it.com/de/menu-links/produkte/krankenhausloesungen/mymedis/op-anaesthesie/op-planung-management.html> abgerufen

Cerner Corporation. (02. Februar 2015). *Cerner Completes Acquisition of Siemens Health Services*.

Abgerufen am 08. Januar 2016 von http://www.cerner.com/Cerner_Completes_Acquisition_of_Siemens_Health_Services/

HIMSS Europe GmbH. (o.J.). *Gesundheitswesen neu*. (HIMSS Europe GmbH)

Abgerufen am 23. November 2015 von <http://www.himss.eu/de/about>

Hübscher, H., Petersen, H.-J., Rathgeber, C., Richter, K., & Dr. Scharf, D. (2009). In *IT-Handbuch IT-Systemelektroniker/-in Fachinformatiker/-in* (S. 322). Braunschweig: Bildungshaus Schulbuchverlage.

Jipp, H., & Dr. Gabriel, P. (25. November 2015). Interview zu den Systemanforderungen. (M. Lenser, & G. Bickel, Interviewer)

Krause, J. (2004). *PHP5 und MySQL - Schritt für Schritt zum Profi*. München: Markt+Technik Verlag.

Lenser, M. (8. Februar 2016). persönliches Gespräch. (G. Bickel, Interviewer)

Microsoft. (o.J.). *Download Microsoft Drivers for PHP for SQL Server from Official Microsoft Download Center*.

Abgerufen am 08. März 2016 von <https://www.microsoft.com/en-us/download/details.aspx?id=20098&751be11f-ed8-5a0c-058c-2ee190a24fa6=True&e6b34bbe-475b-1abd-2c51-b5034bcdd6d2=True>

Microsoft. (o.J.). *Download Visual C++ Redistributable für Visual Studio 2015 from Official Microsoft Download Center*.

Abgerufen am 08. März 2016 von <https://www.microsoft.com/de-de/download/details.aspx?id=48145&751be11f-ed8-5a0c-058c-2ee190a24fa6=True>

Microsoft. (o.J.). *Systemanforderungen und Installationsinformationen für Windows Server 2012 R2*.

Abgerufen am 28. Februar 2016 von <https://technet.microsoft.com/de-de/library/dn303418.aspx>

Sun Microsystems, Inc. (o.J.). *Core J2EE Patterns - Data Access Object*.

Abgerufen am 12. März 2016 von <http://www.oracle.com/technetwork/java/dataaccessobject-138824.html>

The PHP Group. (o.J.). *PHP: Microsoft IIS 7.0 and later - Manual*.

Abgerufen am 08. März 2016 von <http://php.net/manual/de/install.windows.iis7.php>

The PHP Group. (o.J.). *PHP: OCI8 Funktionen - Manual*.

Abgerufen am 12. März 2016 von <http://php.net/manual/de/ref.oci8.php>

The PHP Group. (o.J.). *PHP: SLQ Injection - Manual*.

Abgerufen am 13. März 2016 von <http://php.net/manual/de/security.database.sql-injection.php>

The PHP Group. (o.J.). *PHP: SQLSRV - Manual*.

Abgerufen am 12. März 2016 von <http://php.net/manual/de/book.sqlsrv.php>

Universitätsklinikum Hamburg-Eppendorf - Geschäftsbereich Unternehmenskommunikation.

(2015). *Geschäftsbericht und Jahrbuch 2014*. Hamburg: Vorstand des Universitätsklinikum Hamburg-Eppendorf.

Universitätsklinikum Hamburg-Eppendorf nimmt mit Software von Siemens Spitzenplatz in Europa ein | Management-Krankenhaus.de. (08. November 2011).

Abgerufen am 06. März 2016 von <http://www.management-krankenhaus.de/news/universitaetsklinikum-hamburg-eppendorf-nimmt-mit-software-von-siemens-spitzenplatz-europa-ein>

W3Techs.com. (09. August 2015). *Usage Statistics and Market Share of Server-side Programming Languages for Websites, March 2016*.

Abgerufen am 13. März 2016 von http://w3techs.com/technologies/overview/programming_language/all

5 Anhang

1 Programmcode.....	1
1.1 Soarian Datenbank-Connector.....	2
1.2 myMedis Datenbank-Connector.....	3
2 Datenbank-Schemata.....	4
2.1 Soarian.....	4
2.2 myMedis.....	5

1 Programmcode

In diesem Abschnitt können die wichtigsten Zeilen Programmcodes eingesehen werden, die zur Durchführung dieses Projekts geschrieben worden sind. Die Darstellung erfolgt weitestgehend unkommentiert, Details können den entsprechenden Dokumentationen von The PHP Group entnommen werden.

Alle kritischen Informationen über die IT-Landschaft des UKE, wie zum Beispiel die Namen der Datenbankserver, Benutzerkennungen und Passwörter, sind in diesem Abschnitt verändert worden. Obwohl es sich bei diesen Informationen nur um die der Testsysteme handelt, sind sie im Interesse des Gesamtkonzerns UKE und der Patientinnen und Patienten zu schützen. Darüber hinaus wurden keine Änderungen vorgenommen, sodass die Syntax gleich der des tatsächlichen Programmcodes ist.

1.1 Soarian Datenbank-Connector

```
1 <?php
2 require_once "aDBConnector.php";
3 class MSSQLDB_SoarianT3 extends aDBConnector
4 {
5     private $dbname;
6     private $db_username;
7     private $db_passwort;
8     private $db_servername;
9
10    function __construct()
11    {
12        // Zugangsdaten zum Datenbankserver wurden aus Sicherheitsgründen verändert.
13        $this->dbname = "Soarian_DB_Test3";
14        $this->db_username = "administrator";
15        $this->db_passwort="dlAie349A!";
16        $this->db_servername="Soarian_Server_Test3";
17    }
18
19    protected function connectToDB()
20    {
21        $connectionInfo = array("Database" =>$this->dbname
22                                , "UID"=>$this->db_username
23                                , "PWD"=>$this->db_passwort);
24        $connection = sqlsrv_connect($this->db_servername
25                                    , $connectionInfo);
26        if (!$connection) {
27            die(print_r( sqlsrv_errors(), true));
28        }
29        return $connection;
30    }
31
32    function executeQuery($query)
33    {
34        $connection = $this->connectToDB();
35        $stmt = sqlsrv_query($connection, $query);
36        if( $stmt === false )
37        {
38            echo "Error in statement execution.</b></b>";
39            die( print_r( sqlsrv_errors(), true));
40        }
41        $result = sqlsrv_fetch_array($stmt);
42        return $result;
43    }
44
45    function getPatientData($patientAccountID)
46    {
47        // Datenbankabfrage wurde aus Sicherheitsgründen verändert.
48        $query = "select
49                    something
50                from
51                    somewhere
52                where
53                    Patientnummer = '". $patientAccountID. "'
54                order by
55                    created";
56
57        return $this->executeQuery($query);
58    }
59 }
60 ?>
```

Abbildung 5: Soarian DB-Connector

Quelle: Eigene Darstellung

1.2 myMedis Datenbank-Connector

```
1 <?php
2 require_once "aDBConnector.php";
3 class OracleDB_myMDBT extends aDBConnector
4 {
5     private $db_port;
6     private $db_username;
7     private $db_passwort;
8     private $db_servername;
9     private $db_servicename;
10
11     function __construct()
12     {
13         // Zugangsdaten zum Datenbankserver wurden aus Sicherheitsgründen verändert.
14         $this->db_port = "4212";
15         $this->db_username = "admin";
16         $this->db_passwort = "555Nase";
17         $this->db_servername = "myMedis_DB_TestServer";
18         $this->db_servicename = "db_test";
19     }
20
21     protected function connectToDB()
22     {
23         $connectionDescription = "DESCRIPTION =
24             (ADDRESS =
25                 (PROTOCOL = TCP)
26                 (HOST = ".$this->db_servername.")
27                 (PORT = ".$this->db_port.")
28             )";
29         $connectionData = "(CONNECT_DATA = (SERVICE_NAME = ".$this->db_servicename."))";
30         $connection = oci_connect($this->db_username, $this->db_passwort
31             , "(".$connectionDescription.$connectionData."");
32         return $connection;
33     }
34
35     protected function executeQuery($query)
36     {
37         $connection = $this->connectToDB();
38         $stmt = oci_parse($connection, $query);
39         oci_execute($stmt);
40         $result = oci_fetch_array($stmt, OCI_ASSOC);
41         oci_close($connection);
42         return $result;
43     }
44
45     function getLastOPData($patientAccountID)
46     {
47         // Datenbankabfrage wurde aus Sicherheitsgründen verändert.
48         $query = "select
49             something
50             from
51             somewhere
52             where
53             Fallnummer = '".$patientAccountID.'"
54             order by
55             created";
56
57         return $this->executeQuery($query);
58     }
59 }
60 ?>
```

Abbildung 6: myMedis DB-Connector

Quelle: Eigene Darstellung

2 Datenbank-Schemata

Um sich einen Überblick über die Relationen der wichtigsten Tabellen innerhalb von Soarian und myMedis zu verschaffen, können die Schemata aus den nächsten Abschnitten hinzugezogen werden. Aus Sicherheitsgründen wurden allerdings alle Tabellen- und Feldnamen verändert.

2.1 Soarian

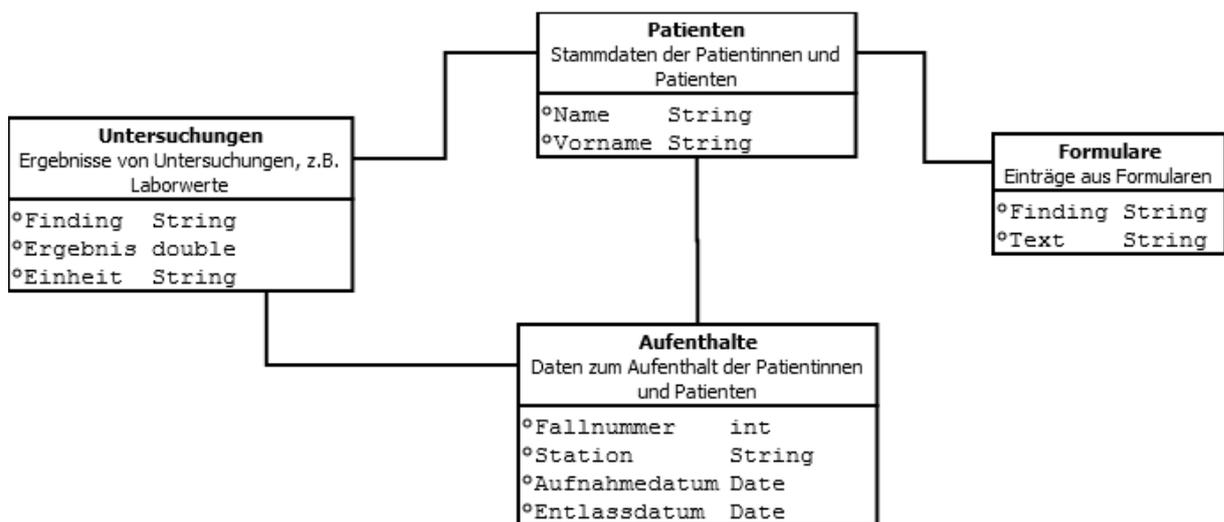


Abbildung 7: vereinfachtes Datenbankschema von Soarian

Quelle: Eigene Darstellung

2.2 myMedis

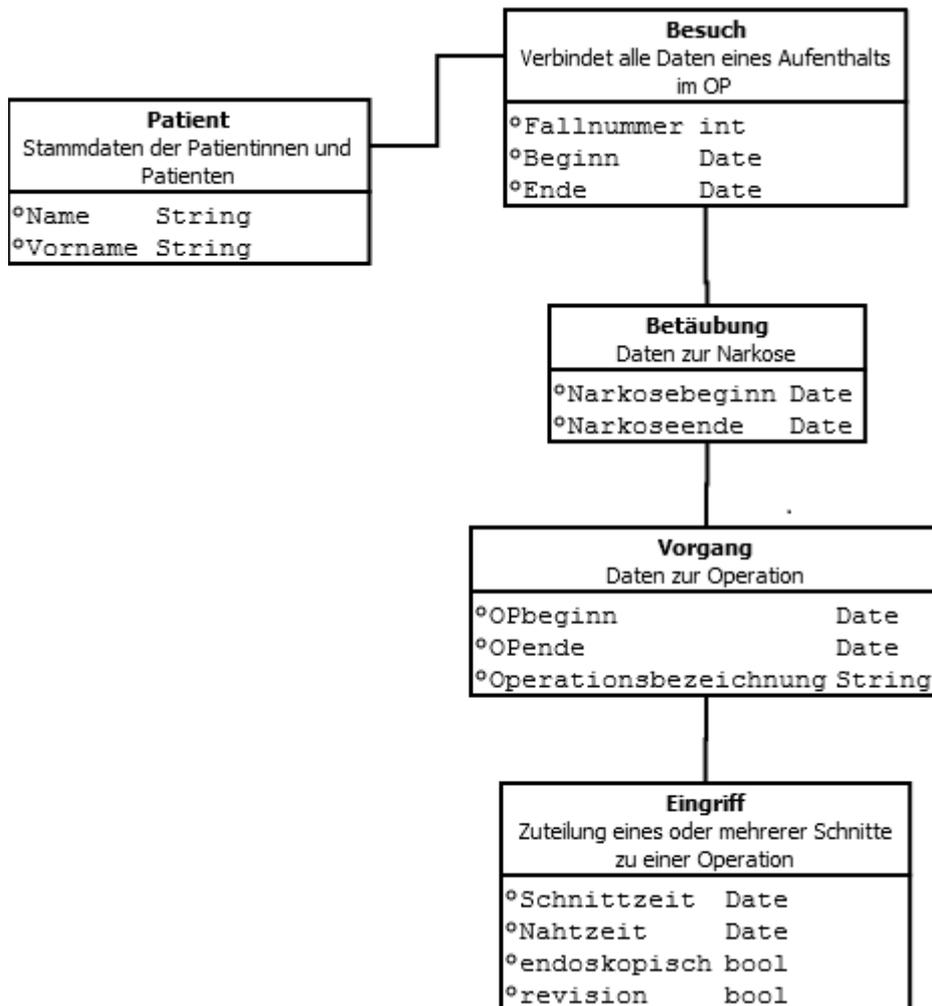


Abbildung 8: vereinfachtes Datenbankschema von myMedis

Quelle: Eigene Darstellung

Versicherung über Selbstständigkeit

Hiermit versichere ich, dass ich die vorliegende Arbeit ohne fremde Hilfe selbstständig verfasst und nur die angegebenen Hilfsmittel benutzt habe. Wörtlich oder dem Sinn nach aus anderen Werken entnommene Stellen sind unter Angabe der Quelle kenntlich gemacht.

Hamburg, den _____