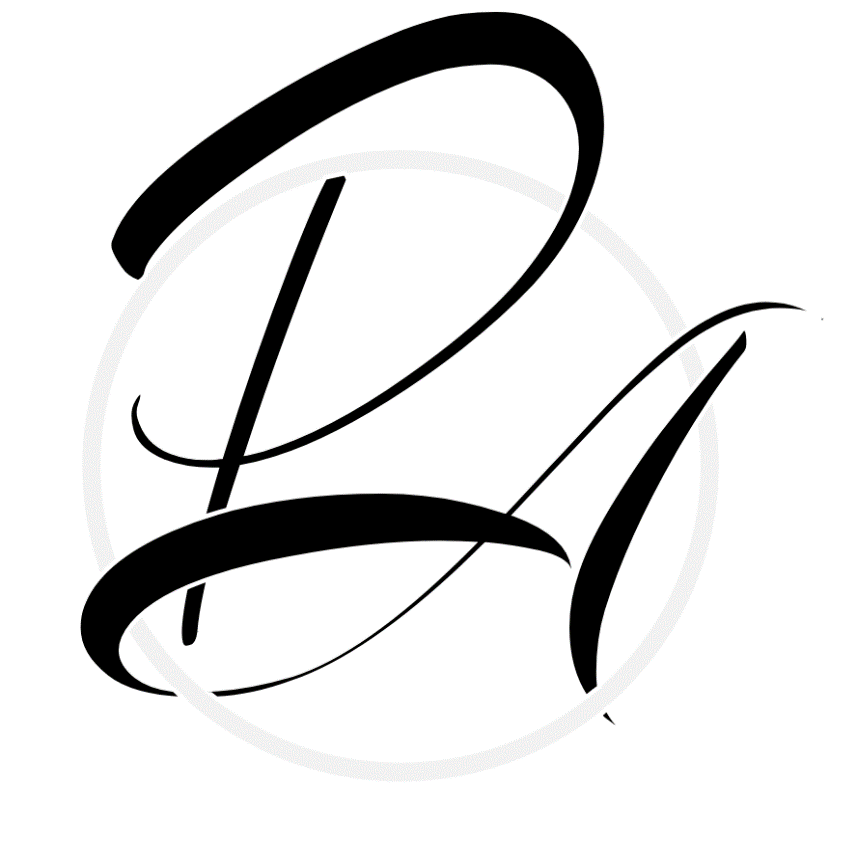
Pflichtenheft: Elektronische Patientenakte



Inhaltsverzeichnis

[1.0 Einleitung 2](#_Toc516491574)

[2.0 Projektumfang 3](#_Toc516491575)

[2.1 Website 3](#_Toc516491576)

[2.1.1 Bestandteile 3](#_Toc516491577)

[2.1.2 Funktionen 3](#_Toc516491578)

[2.1.3 Design 4](#_Toc516491579)

[2.1.4 Sicherheit 4](#_Toc516491580)

[2.2 Datenbank 4](#_Toc516491581)

[3 Website 5](#_Toc516491582)

[3.1 Anmelde Ansicht 5](#_Toc516491583)

[3.2 Auf Patienten zugreifen 6](#_Toc516491584)

[3.3 Neuen Eintrag erstellen 7](#_Toc516491585)

[4 Systemtests 8](#_Toc516491586)

[4.1 Datenbank Allgemein: 8](#_Toc516491587)

[4.2 DB-Queries: 8](#_Toc516491588)

[4.3 Schnittstellen: 9](#_Toc516491589)

[4.4 User Eingabefehler: 9](#_Toc516491590)

[4.5 Overlay: 9](#_Toc516491591)

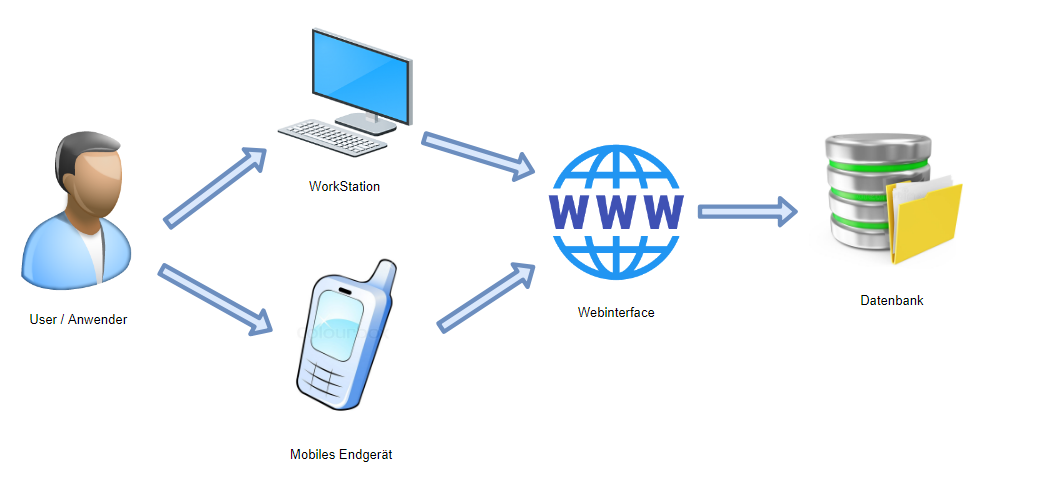
1. Einleitung

Daten Elektronisch zu erfassen bietet vielerlei Vorteile, so werden keine kompletten Kellerräume mehr mit alten Büchern und Ordner befüllt und es kann einfach und gezielt nach bestimmten Datensätzen gesucht werden und diese stets aktualisieren oder verändern.

Heutzutage wird der Großteil aller Daten elektronisch erfasst. Ein Krankenhaus beispielsweise könnte im Jahr mehrere hunderte Ordner mit Patientenakten befüllen. Doch dies gehört der Vergangenheit an. Selbst in kleinen dörflichen Arztpraxen ist ein PC nicht mehr wegzudenken.   
Meist ist die dafür genutzte Software sehr in die Jahre gekommen, die Anwender müssen sich durch eine komplizierte und Altbackene UI klicken.

Doch genau hier kommt die unsere moderne elektronische Patientenakte zum Einsatz. Nach Absprache mit dem Auftraggeber erfolgt die Erfassung und Verwaltung der Patientenakten über eine moderne Website. So ermöglichen wir maximale Flexibilität in jeder Hinsicht. Nicht nur ist der Anwender von nun an nicht mehr an ein lästiges Programm gebunden, welches auch gerne mal Probleme verursacht oder Updates benötigt, sondern wir bringen die Datenverwaltung weg von den großen Servern hinzu mobilen Endgeräten. So dass im Alltagsstress alle wichtigen Daten bequem in der Hosentasche zur Verfügung stehen.

Hinter der Website steht eine aktuelle, zuverlässige, sichere und schnelle Datenbank die von unserem Team regemäßig gewartet und aktualisiert wird. Die Verknüpfung der Website mit der Datenbank erfolgt über PHP und JavaScript.   
Die Website ermöglicht es dem Auftraggeber unter minimalem Aufwand das erfassen und erweitern von Patientenakten.

  
***Abbildung 1:*** *Zugriffs Hierarchie beim Ablegen von Informationen auf der Datenbank.*

2.0 Projektumfang

Ausfolgenden Bestandteilen ist unser Projekt aufgebaut:

2.1 Website

2.1.1 Bestandteile

* HTML Version 5.2
* CSS3
* JavaScript Version 1.8.5
* PHP Version 7.2.6

2.1.2 Funktionen

* Anlegen neuer Patientenakten
* Bearbeiten bestehender Patientenakten
* Bisherige Behandlungen einsehen
* Neue Behandlungen Hinzufügen
* Bearbeiten der Anamnese

2.1.3 Design

* Eingabe der Versicherungsnummer über Label
* Im oberen Bereich der Website ein iFrame mit Patientendaten
* Slider für Einsicht in vergangene und aktuellen Behandlungen
* Slider für Erstellen einer neuen Behandlung

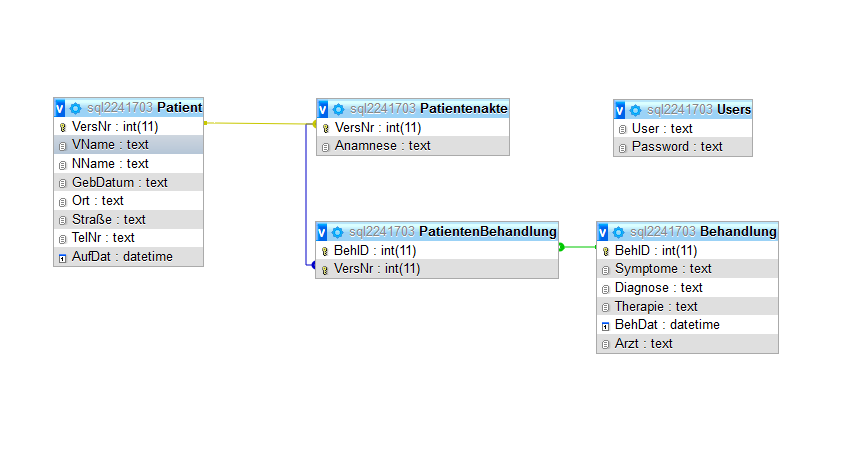
2.1.4 Sicherheit

* Nutzer muss sich über Nutzernamen und geheimes Passwort anmelden
* Bei Änderungen Unterschrift/Elektrischer Stempel des Arztes notwendig
* Bei Änderung wird ein Zeitstempel hinzugefügt
* Verbindung nur über sichere Kanäle

2.2 Datenbank

Für die Patientenakte werden 5 Entitäten benötigt. Die Patientenentität besteht aus den Attributen Versicherungsnummer (PK), Vorname, Nachname, Geburtsdatum, Ort, Straße, Telefonnummer sowie das Aufnahmedatum. Sie wiederspiegelt die Persönlichen Daten des Patienten. Patientenankteentität beinhaltet die Versicherungsnummer (PK, FK) und ein Attribut Anamnese. Für eine Behandlung wurden die Attribute Behandlungs-ID (PK), Symptome, Diagnose, Therapie, Arzt und Behandlusngsdatum eingeführt. Die 1 : n Beziehung zwischen Patientenakte und Behandlung wird mit einer Patientenbehandlungentität, welche Behandlungs-ID (FK,PK) und Versicherungsnummer (FK,PK) als Attribut beinhaltet, dargestellt. Für die Userlogins wird eine separate Userentität mit den Attributen User (PK) sowie Passwort (PK) benötigt.

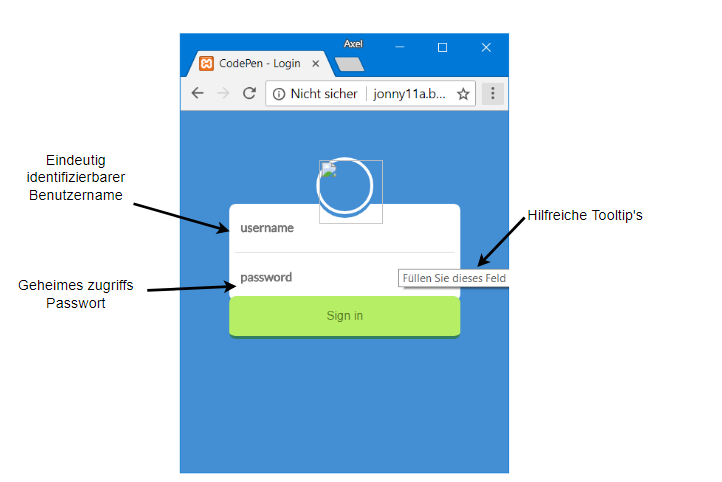
Als Datenbankmanagementsystem wird MySQL verwendet.

Das ER-Modell wird wie folgt dargestel

3 Website

3.1 Anmelde Ansicht

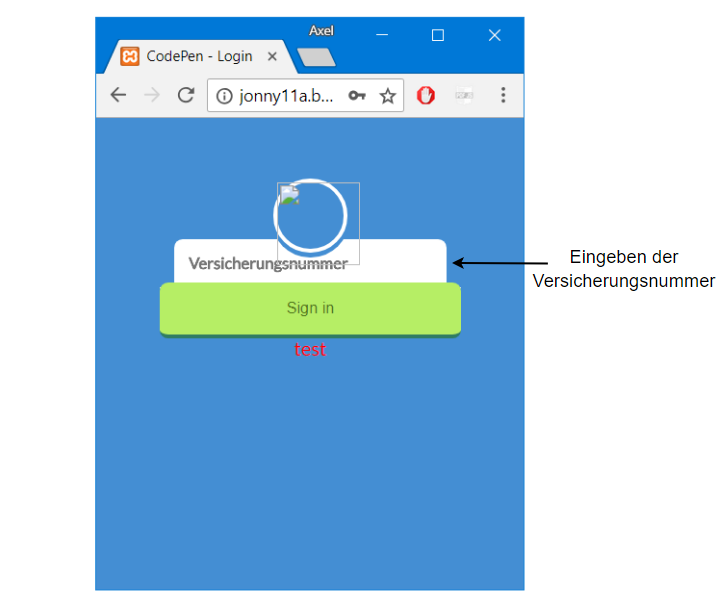
Das von uns Designte Anmelde Fenster ist recht schlicht und einfach gehalten. Da es sich um eine Business Anwendung handelt, haben wir auf Ressourcen verschwendende Verschönerungen verzichtet und alles so einfach wie möglich gehalten.  
Es geht uns auch darum das Leute mit der elektronischen Patientenakte arbeiten können, die bisher noch nicht so viel Erfahrung mit der elektronischen Datenverarbeitung sammeln konnten, ohne dass diese direkt abgeschreckt werden.  
Liefert bei einer falschen Eingabe eine schlüssige Fehlermeldung.



***Abbildung 2:*** *Das anmelde Fenster auf der Website*

3.2 Auf Patienten zugreifen

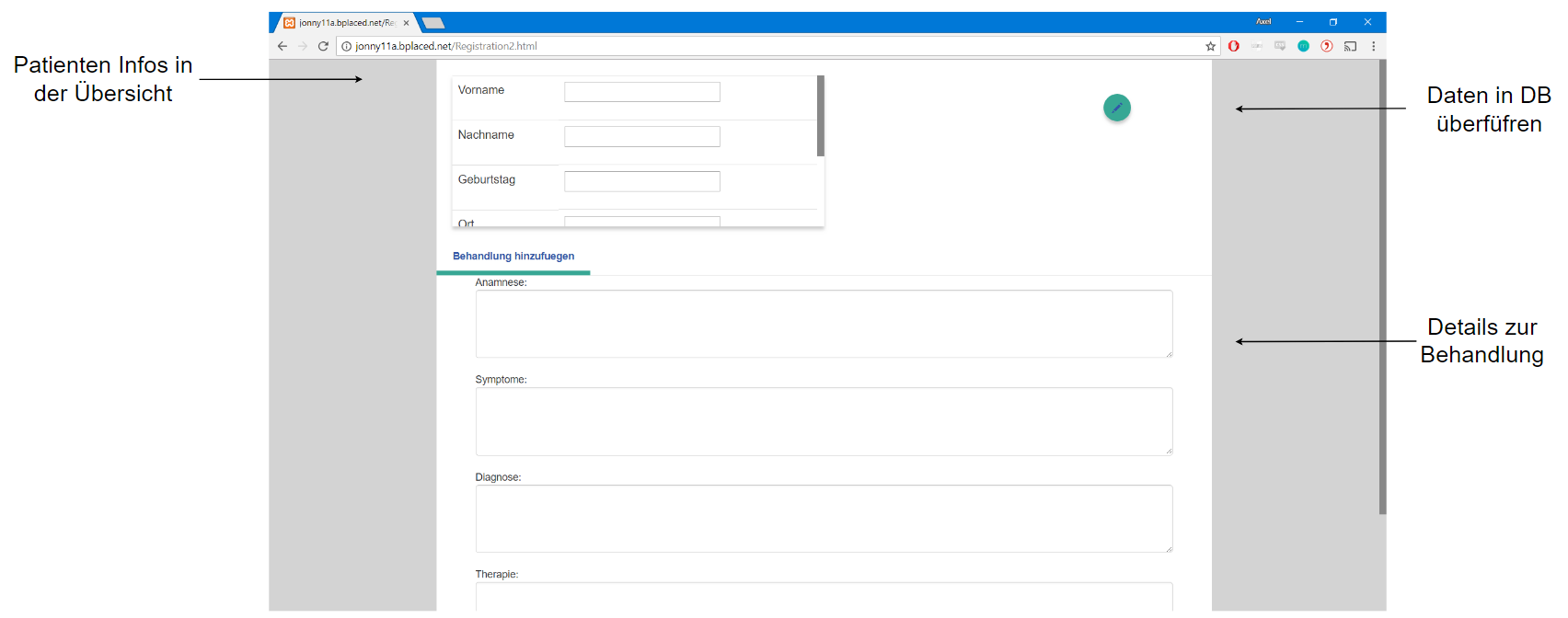
Direkt nach der Eingabe der Benutzerdaten wird der Kunde dazu aufgefordert eine gültige Versicherungsnummer einzugeben wie in ***Abbildung 3***.   
Die Versicherungsnummer wurde von uns als primary key gewählt, die sich diese Ideal eignet um jeden Kunden eindeutig zu identifizieren.   
Sollte es zu der eingetragen Versicherungsnummer bereits einen Eintrag auf der Datenbank geben so werden die Patientendaten sofort geladen und können betrachtet und bearbeitet werden.  
Sollte es eine der Datenbank unbekannte Versicherungsnummer sein so erkennt das System direkt dass ein neue Akte erstellt wird.

****

***Abbildung 3:*** *Eingeben der Versicherungsnummer als Primär Key um Zugriff auf Patientendaten zu erhalten.*

3.3 Neuen Eintrag erstellen

Nun kann der Anwender einen neue Patientenakte erstellen mit allen Infos die dazugehören.  
Durch drücken auf das Grüne Stift Symbol werden die Daten in die Datenbank übernommen und gespeichert.

****

***Abbildung 4:*** *Einen neuen Patient anlegen*

4 Systemtests

4.1 Datenbank Allgemein:

|  |  |  |  |
| --- | --- | --- | --- |
| **Anforderung** | **Technologie** | **Validierung** | **Geprüft?** |
| Host eines DBMS | mySQL | Manuelle Querie Validierung | Ja |
| Schnittstelle über mysqli von PHP realisiert | PHP | Modultest | Ja |

4.2 DB-Queries:

|  |  |  |  |
| --- | --- | --- | --- |
| **Anforderung** | **Technologie** | **Validierung** | **Geprüft?** |
| Passwort Querie | Ajax / PHP / SQL | Durch eigenes HTML Testmodul | Ja |
| Patientendaten Querie über PK | Ajax / PHP / SQL | Durch eigenes HTML Testmodul | Ja |
| Anamnese Querie über PK | Ajax / PHP / SQL | Durch eigenes HTML Testmodul | Ja |
| Behandlungs Querie über FK-Table | Ajax / PHP / SQL | Durch eigenes HTML Testmodul | Ja |
| Persistierung der Patientendaten | Ajax / PHP / SQL | Durch eigenes HTML Testmodul | Ja |
| Persistierung der Anamnese | Ajax / PHP / SQL | Durch eigenes HTML Testmodul | Ja |
| Persistierung der Behandlung | Ajax / PHP / SQL | Durch eigenes HTML Testmodul | Ja |

4.3 Schnittstellen:

|  |  |  |  |
| --- | --- | --- | --- |
| **Anforderung** | **Technologie** | **Validierung** | **Geprüft?** |
| Einbindung der Datenbank | Ajax / PHP / SQL | Durch eigenes HTML Testmodul | Ja |

4.4 User Eingabefehler:

|  |  |  |  |
| --- | --- | --- | --- |
| **Anforderung** | **Technologie** | **Validierung** | **Geprüft?** |
| Falsches Passwort | JavaScript / PHP | Modultest | Ja |
| Keine Signierung des Arztes hinterlegt | JavaScript / PHP | Modultest | Ja |
| Wenn Patient noch nicht existiert -> Patient anlegen | JavaScript / PHP | Modultest | Ja |
| Wenn Patient bereits existiert -> Patientendaten / Anamnese / vorherige Behandlungen Aufrufen | JavaScript / PHP | Modultest | Ja |

4.5 Overlay:

|  |  |  |  |
| --- | --- | --- | --- |
| **Anforderung** | **Technologie** | **Validierung** | **Geprüft?** |
| Übersichtliches Login | HTML / Bootstrap | Modultest | Ja |
| Aufnahme der Versichertennummer | HTML / Bootstrap | Modultest | Ja |
| Patientendatenübersicht mit diversen Behandlungen | HTML / Bootstrap | Modultest | Ja |
| Aufnahme der Patientendaten sowie die Anamnese | HTML / Bootstrap | Modultest | Ja |
| Eindeutige Dialoge oder ggf. verständliche Fehlermeldungen | HTML / Bootstrap | Modultest | Ja |
| Darstellung auf Workstations unabhängig des Betriebssystems | JavaScript / CSS | Durch eigenes HTML Testmodul | Ja |
| Darstellung auf Tablets | JavaScript / CSS | Durch eigenes HTML Testmodul | Ja |