

PHP Fortsetzung

Teil 1: Seiten für Formulareingabe, -validierung und Ausgabe erstellen

1) Erste PHP Seite: index.php

- a) Erzeugen Sie eine PHP-Seite mit einem HTML-Grundgerüst und fügen Sie Überschriften sowie ein HTML-Formular ein – verwenden Sie dazu das Formular aus vorangegangenen Übungen. Ergänzen Sie etwaige fehlende Felder
- b) Beachten Sie, dass das Formular mit den Attributen Action und Method so eingestellt ist, dass es einen **GET-Request** auf die Datei auswertung.php macht.
- c) **Beschriften und formatieren** Sie alle Formular-Felder entsprechend

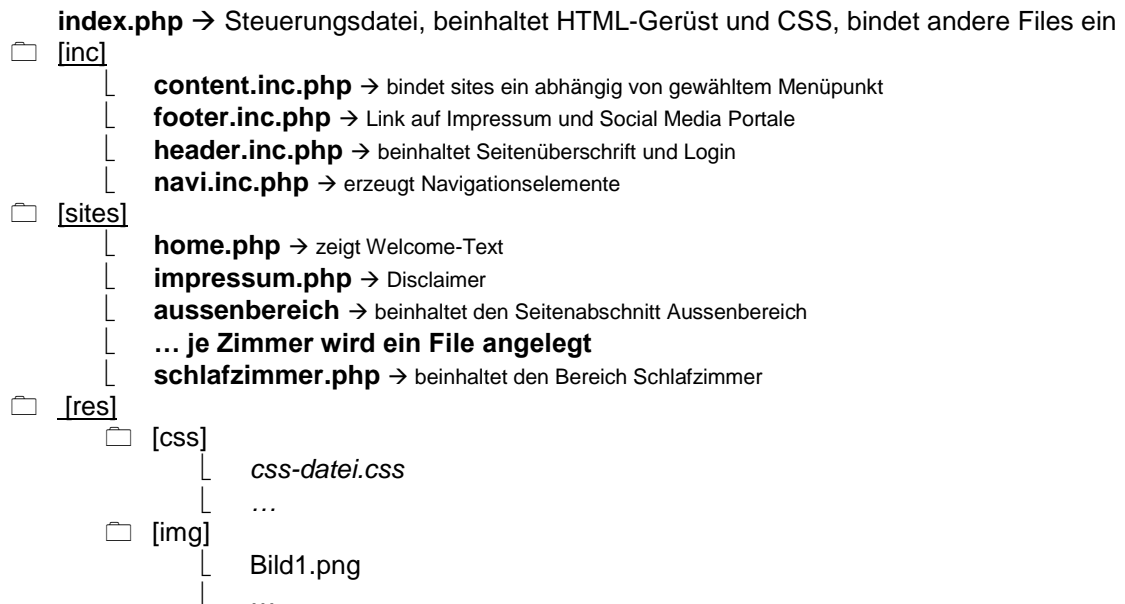
2) Zweite PHP Seite: auswertung.php

- a) Erstellen Sie eine zweite PHP Web Page, welche nun die Daten entgegennimmt. Prüfen Sie, ob alle Daten vollständig übergeben wurden und geben Sie alle Daten in einer übersichtlichen Struktur aus.
- b) Unvollständige Eingaben sollen markiert werden (bspw. rote Schrift mittels Bootstrap od. natives CSS).
- c) Geben Sie weiters die kompletten Inhalte des \$_GET-Array unter Verwendung der Funktion var_dump() aus. Im Realszenario könnten die Daten an dieser Stelle bspw. an einen E-Mail-Rezipienten versendet, in einer DB gespeichert oder an ein Web Service gesendet werden

Teil 2: Smart Homes - Modulares Layout mit PHP

1) Projektstruktur

Um mehr Überblick über Komponenten der Software sowie bessere Wartbarkeit zu erreichen, wird folgende Projektstruktur empfohlen:



2) Grundlayout/Steuerungsdatei

- a) Legen Sie ein neues PHP Projekt für die aktuelle Übung an
 - i) Inhalte und Optik können aus alten Übungen recycelt werden
- b) Erstellen Sie die Datei index.php und erzeugen Sie ein HTML-Grundlayout mit mind. diesen Bereichen
 - i) Header
 - ii) Navi
 - iii) Content
 - iv) Footer
- c) Das Aussehen der Seite wird über eine externe CSS Datei (mittels Bootstrap Framework) bestimmt, welche in die index.php eingebunden wird.
- d) Benutzen Sie ggf. auch eine eigene CSS-Datei, um notwendige Styles umzusetzen.

2) „Module“

- d) Legen Sie einen neuen Ordner in Ihrem Projekt an mit der Bezeichnung „inc“
- e) Erzeugen Sie je eine PHP Datei für die Bereiche in der index.php (Header, Navi, ... werden abgebildet durch header.inc.php, navi.inc.php ...) und legen Sie diese im Ordner inc ab

Die Inhalte dieser Module sind folgende:

i) Header

- enthält die Überschrift sowie ein Login-Formular
- dieses nimmt (GET/POST)Parameter entgegen und überprüft diese auf Inhalt und Wert.
- Wenn „wohlgeformte“ Parameter übergeben wurden, werden diese gegen statische Werte überprüft, um einen Login zu simulieren.
- Legen Sie dafür Username und Passwort in je eine Variable ab. Wenn die übergebenen Werte mit den hinterlegten Werten übereinstimmen, ist der Login erfolgreich und als Ausgabe erscheint: „Sie sind eingeloggt als USERNAME“
- *In der Praxis werden die Logindaten gegen bspw. eine Datenbank geprüft oder auch gegen externe Schnittstellen (wie bspw. OAuth).*

ii) Navigation

- enthält 7 Menüeinträge welche in einem assoziativen Array abgelegt werden. Aus diesem Array soll ein listenbasiertes Menü erzeugt werden. Als Schablone verwenden Sie ein Bootstrap Menu. Mindestens folgende Einträge zeigt das Menü:
 - i. Home
 - ii. Schlafzimmer
 - iii. Wohnzimmer
 - iv. Esszimmer
 - v. Küche
 - vi. Außenbereich
 - vii. Kontakt (Verweis auf mailto-Adresse)
- Weiters hat jeder Menü-Eintrag eine Link-Funktion und verweist auf index.php

- Bei jedem Link wird der Name des Menüeintrags mitgegeben, damit später festgestellt werden kann, welcher Link geklickt wurde, bspw:

```
<a href="index.php?selectedMenu=".$menuName">Menu1</a>
```

- Der aktuell ausgewählte Menüeintrag soll in hervorgehoben (zB mit Bootstrap-Klasse active) werden.

iii) Content

- ist verantwortlich für das Einbinden der richtigen Inhalte, abhängig davon, welchen Menüpunkt der User angeklickt hat.
- Basierend auf einer Whitelist in Form einer switch/case Anweisung wird der über das Menü übergebene Parameter (selectedMenu) geprüft und die gewählte Site eingebunden.
 - i. Home
 - a. Bindet home.php ein, welche den Welcome-Text anzeigt
 - ii. Schlafzimmer - Außenbereich
 - a. Bindet jeweils das dazugehörige File mit den Zimmerinformationen ein.
 - iii. Impressum
 - a. Bindet impressum.php ein, welche einen Disclaimer enthält

```
...
switch ($_GET['selectedMenu']) {
    case 'eintrag1': include 'file1.php';
                    break;
    case 'eintrag2': include 'file2.php';
                    break;
}
...
```

iv) Footer

- Enthält Links zu Impressum und Social Media Portalen. Legen Sie für das Impressum ebenfalls eine Inhaltsseite an und zeigen Sie diese – wenn angeklickt - im Content-Bereich an.

Allgemeines

1. Außerdem & Kontrolle

- a. Stellen Sie sicher, dass Sie weiterhin valides Markup erzeugen
 - i. Einhalten des HTML-Grundaufbaus, HEAD-BODY-Bereiche
 - ii. richtige Verschachtelung
- b. Validieren Sie Ihr Ergebnis (das im Browser als Quelltext ankommt) erneut (W3C-Validator)

Form der Abgabe

Laden Sie die gesamte Abgabe bis zum Abend vor der nächsten Übungseinheit als zip/rar File im Moodle hoch. Achten Sie darauf, dass ALLE Ressourcen vollständig enthalten sind (CSS-Files, Bilder, ...)

>>> FASTLANE >>>

1. DataModel: Patienten-Objekt

- a. Legen Sie eine Patientenklasse (in Ordner „model“) an und hinterlegen Sie die dem Formular (Ü1) entsprechenden Eigenschaften.
- b. Wenn das Formular ausgefüllt wurde, befüllen Sie das Patientenobjekt und lesen Sie die Daten für die Daten-Anzeige auch wieder von dort aus.

2. Eingaben-Überprüfung mit Regular Expressions

- a. Erweitern Sie die Überprüfung der Formulardaten durch RegEx, sodass die Eingaben exakter auf ein Muster geprüft werden, bspw. numerische Werte, Datumsformate, Zeicheninhalte von Namen, etc.
- b. Geben Sie entsprechende Fehlermeldungen aus.

3. Utility-Class: DataChecker

- a. Legen Sie eine weitere Klasse in einem neuen Ordner „utility“ an.
- b. Lagern Sie sämtliche Formularüberprüfungen in diese Klasse aus. Wenn Daten im Patientenobjekt gesetzt werden, sollen diese vorher mittels DataChecker-Objekt validiert werden.

Referenzen:

- Formulardatenübergabe
 - https://www.php-einfach.de/php-tutorial/_get-und-_post/
- PHP Forms
 - https://www.w3schools.com/php/php_forms.asp
- Form Validierung
 - https://www.w3schools.com/php/php_form_validation.asp
- Formular ausfüllen
 - https://www.w3schools.com/php/php_form_complete.asp
- PHP Manual → GET
 - <http://php.net/manual/de/reserved.variables.get.php>
- PHP Manual → POST
 - <http://php.net/manual/de/reserved.variables.post.php>
- PHP Include
 - <http://php.net/manual/de/function.include.php>
- Regex Tutorial
 - <https://regexone.com/>
- Regex Training
 - <https://regexr.com/>
- PHP & Objekt Orientierte Programmierung
 - <https://www.php-einfach.de/experte/objektorientierte-programmierung-oop/>