Das Dokumentieren ist ein integraler Bestandteil eines Projektes. Deshalb muss in Anlehnung an Kap.12 aus dem Lehrmittel «Web Engineering» auch die zu erstellende Webseite dokumentiert werden.

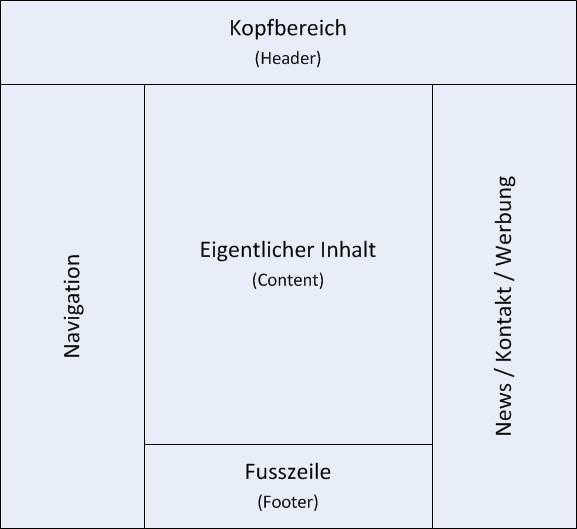
Damit bei dieser Dokumentation nicht auf dem leeren Blatt gestartet werden muss, kann diese Vorgabe verwendet werden.

Die blauen Texte beschreiben den geforderten Inhalt der jeweiligen Kapitel.

|  |  |  |  |
| --- | --- | --- | --- |
| Rev. | Datum | Autor | Bemerkungen |
| 0.1 | 13. Sept. 2019 | SOP | Initiale Version |
|  |  |  |  |

# Anforderungen an die Webseite

* Was für Inhalte sollen auf der Webseite präsentiert werden?
* Es wird eine einfache Web-Applikation für einen Verein erstellt. Technisch gesprochen wird dazu eine dynamische Web-Applikation erstellt, welche Daten aus einer Datenbank liest und neue Daten in die Datenbank schreibt. Drei statischen Seiten mit Bildern (Vereinsgeschichte, Aktuelles, Kontakte) sind ein Bestandteil der Website. Ein Gästebuch ist ebenfalls implementiert. Alle statischen Elemente auf der Webseite (z.B. Hintergrundbilder, Struktur der Webseite, usw.) werden als Template gespeichert und von allen Seiten dynamisch geladen. Dadurch sind die statischen Seiten auch in PHP zu erstellen.
* Es gibt auch eine Navigation (navbar.php), dadurch können die anderen Seiten erreicht werden. Diese Navigation ist nicht statisch, denn sie wird durch php generiert.
* Wie soll die die Struktur der Webseite aussehen?



* Skizze des GUIs
* Navigation zwischen den Seiten.
* Welche Funktionen sollen beim Aufruf bzw. Verlassen einer Seite aufgerufen werden?
* Welche Ausnahmesituationen können auftreten, und wie wird mit denen umgegangen?
* Beim Erstellen eines Gästebucheintrags kann es vorkommen, dass der Benutzer ein oder mehrere Felder nicht ausfüllt. Das kann entweder auf der Clientseite mit Javascript oder auf der Serversite mit php kontrolliert werden. In diesem Fall wurde ein php file geschrieben, dass die Felder auf Richtigkeit überprüft, bevor das Formular überhaupt abgeschickt wird.

# Technische Beschreibung

* Wie wird die Struktur der Webseite sichergestellt? Gibt es Vorlagen (Templates)? Wie werden diese geladen?
* Wie werden die Dateien auf dem Webserver organisiert? Siehe Beispielorganisation
* In welcher Datei wird welche Funktionalität abgedeckt? (Zugriff auf DB, Speichern von Einträgen, Darstellen von Inhalten, etc.)
* Wie läuft das Erstellen eines Gästebucheintrages ab?
* Die Einträge des Gästebuches (Name des Autors, E-Mail vom Autor, Datum der Erstellung, eigentlicher Text) werden in einer Datenbank verwaltet. Wie läuft die Darstellung der Gästebucheinträge ab?
* Beim Aufruf der Gästebuch-Seite sollen alle Einträge nach Datum sortiert auf den Bildschirm ausgegeben werden (jedoch ohne E-Mail vom Autor).
* Jeder Besucher dieser Seite kann nun einen eigenen Eintrag erstellen. Dieser Beitrag wird anschliessend ebenfalls in der Datenbank abgelegt. Das Erstellungsdatum wird durch die Datenbank selbstständig ermittelt.

Abbildung 1: Beispielorganisation

* Headers
  + Header\_oben.html
  + Header\_unten.html
* Bilder
  + Logo.png
  + Bild1.jpg
* Index.html
* …

# Testen

* Die Webseite muss an Anlehnung an Kapitel 12.5.3 auch getestet werden.
* Mindestens 2 manuelle Testfälle werden erwartet. Zum Beispiel: Gästebucheintrag erstellen, Gästebucheintrag darstellen
* Was für Felder müssen beim Erstellen ausgefüllt sein?
* Ein Testfall wurde bereits begonnen:

## Testfall „Gästebucheinträge darstellen“

|  |  |
| --- | --- |
| **Name** | Gästebucheinträge darstellen |
| **Vorbedingung** | * Webseite muss auf Webserver (Apache 2.4.4) hochgeladen sein. * Testfall „…“ muss ausgeführt worden sein |
| **Ablauf** | 1. Webseite aufrufen 2. In Navigation auf „Gästebuch“ klicken 3. … |
| **Erwartetes Ergebnis** | Einträge welche im Testfall „…“ erstellt wurden werden dargestellt |

## Testfall …

* Ein weiterer Testfall

# Testergebnisse

* Zusammenfassung der Testergebnisse:  
  Wann hat wer gestestet?  
  Welche Testfälle wurden durchgeführt?   
  Stimmte das tatsächliche Ergebnis mit dem erwarteten Ergebnis überein?