

Technische Dokumentation

Einleitung

Erstellt wurde ein Online-Tutorial zur Schulung der Informationskompetenz; insbesondere bezüglich wissenschaftlichem Arbeiten. Auftraggeber war die Humboldt-Universität zu Berlin. Ausgeführt wurde der Auftrag durch Studierende der Hochschule der Medien Stuttgart aus den Studiengängen Bibliotheks- und Informationsmanagement (betreut durch Prof. Simon - simon@hdm-stuttgart.de) und Informationsdesign (betreut durch Prof. Mangold - mangold@hdm-stuttgart.de).

Dies ist die technische Dokumentation zum Umgang mit dem Tutorial. Bei Fragen wenden Sie sich bitte an Toni Michel (tm063@hdm-stuttgart.de).

Verwendeten Technologien

Die folgenden Technologien kamen im Tutorial zum Einsatz:

- HTML5
- CSS3
- Javascript / jQuery

Es wurden keine serverseitigen Technologien verwendet, weswegen das Tutorial auf jedem beliebigen Server gehostet werden kann und auch als lokale Anwendung ausführbar ist. Es wurden zudem keine Drittanbieter-Technologien wie Java Applets oder Flash verwendet. Jeder moderne Browser kann das Tutorial also darstellen, ohne auf Plug-Ins angewiesen zu sein. Getestet wurde mit den aktuellen Generationen der Browser Firefox, Internet Explorer, Opera, Google Chrome und Safari.

Mobile Geräte können das Tutorial verwenden, obgleich dieses nicht explizit für mobile Geräte optimiert wurde. Notwendig ist eine ausreichende Display-Größe, was auf den meisten Tablets, jedoch nicht den meisten Handys, gewährleistet ist.

Implementierung des Tutorials

Für die Implementierung des Tutorials muss nichts im Speziellen beachtet werden. Lediglich ein geeignetes Verzeichnis auf dem Server auswählen, die vorliegenden Dateien in das Verzeichnis übertragen und das Tutorial kann aufgerufen werden.

Austauschen der Logos

Zum Austauschen des Logos, je nach Anbieter, der das Tutorial verlinkt, wird die URL des Tutorials genutzt. Dafür wird die URL der Startseite erweitert um einen *accesspoint*. Um beispielsweise das Logo des Bildungsforums Potsdam auf der Seite zu verwenden würde die URL so aussehen:

`domain.de/tutorial_informationskompetenz/index.html?accesspoint=1`

Die verschiedenen Anbieter, die das Tutorial verlinken möchten, und für die ein eigenes Logo vorgesehen ist, erhalten also einen individualisierten Link. Hier eine Übersicht über die verschiedenen *accesspoints*:

Anbieter	URL
Humboldt-Universität	index.html?accesspoint=0 (oder nur index.html - das Logo der Humboldt-Universität wird als Standard verwendet, wenn kein Accesspoint angegeben ist)
Bildungsforum Potsdam	index.html?accesspoint=1
SSB	index.html?accesspoint=2
TU Berlin	index.html?accesspoint=3
TU Berlin Universitätsbibliothek	index.html?accesspoint=4

Hinzufügen neuer Logos

Um der Liste perspektivisch neue Logos hinzuzufügen muss folgendes beachtet werden.

- Die vollständigen Namen der neuen Logos müssen in der Datei **Logolist.json** verzeichnet werden. Dabei kann sich am bestehenden Aufbau der Datei orientiert werden.
 - Die Zahl vor dem Doppelpunkt ist jeweils identisch zum späteren Accesspoint (der an das Ende der URL angefügt wird).
 - Bei *start* muss der Name des Logos eingetragen werden, dass auf der **Startseite** angezeigt wird. Da die Wabe auf der Startseite, auf der das Logo platziert wird, ein farbiges Muster hat, sollten Logos mit Transparenzen auf einem entsprechend ruhigen Hintergrund platziert werden - beispielsweise auf einem weißen Kreis.
 - Bei *content* muss der Name des Logos eingetragen werden, das später auf der **Inhaltsseite** angezeigt werden soll. Hier ist kein zusätzlicher Hintergrund für Logos mit Transparenzen notwendig.
- Beide Dateien für jedes Logo (also je eine für die Verwendung auf der Startseite und für die Verwendung auf der Inhaltsseite) müssen im Ordner **logos/** gespeichert werden.

Verwaltung des Inhalts

Im Tutorial wurden Funktionalität und Inhalt weitestgehend voneinander getrennt. So ist es möglich, den Inhalt bei Bedarf auf eine andere Seite zu übertragen. Außerdem sind Änderungen am Inhalt möglich, ohne dass die Funktionalität der Seite beeinflusst wird.

Inhalt bearbeiten

Im Hauptverzeichnis der Webseite befindet sich ein Ordner mit dem Namen **directory**. In diesem befinden sich sechs Unterordner, einer für jedes Kapitel im Tutorial. In jedem dieser Unterordner befindet sich je eine HTML Datei für jedes Unterkapitel.

In jedem Ordner befindet sich des Weiteren eine HTML Datei, die den gleichen Namen hat, wie der Ordner. In dieser befindet sich der Inhalt der Startseite des Kapitels.

Bis auf einige Besonderheiten, auf die nachfolgend eingegangen wird, gelten für die Bearbeitung der Inhaltsseiten die regulären HTML Regeln. Jeder, der mit HTML vertraut ist, sollte keine Schwierigkeiten haben, Inhalte zu editieren.

Verlinkungen

Unterschieden wird zwischen Verlinkungen die auf eine andere Unterseite des Tutorials verweisen und Verlinkungen die auf eine externe Interseite verweisen.

Für Verlinkungen auf eine Unterseite des Tutorials muss dem Link die Klasse *internal-link* gegeben werden. Als Ziel des Links wird keine Webadresse eingetragen, sondern *Kapitelnummer_Unterkapitelnummer*. **Wichtig:** Die Nummerierung der Kapitel beginnt mit 0 und ist daher nicht identisch mit den angezeigten Kapitelnummern, sondern jeweils um eins geringer. Kapitel 1 erhält als Kapitelnummer eine 0, Kapitel 2 erhält eine 1 und so weiter. Eine ausführliche Liste der Nummerierung kann in der Datei **directory.json** eingesehen werden. Beispielsweise soll ein Link erstellt werden, der auf Kapitel *1.2 Themenformulierung* führt. Dieser sieht wie folgt aus:

```
<a href="0_1" class="internal-link">Ein interner Link</a>
```

Externe Seiten können wie in HTML üblich verlinkt werden, jedoch benötigt jeder Link auf eine externe Seite zusätzlich die Klasse *external_link*.

Pop-Ups

Ein Pop-Up besteht aus zwei Teilen. Ein Textabschnitt, der angeklickt werden kann, um das Pop-Up zu öffnen, und dem Pop-Up selbst. Die Inhalte des Pop-Ups befinden sich immer in der gleichen HTML Datei wie der Link, der das Pop-Up öffnet.

Links, die Pop-Ups öffnen, erhalten als Klassen *popup-link* und *popup-x*, wobei x durch die Nummer des jeweiligen Pop-Ups ersetzt wird. Ein Link, der ein Popup-Fenster öffnet, könnte so aussehen:

```
<a href="" class="popup-link popup-1">Hier öffnet sich ein Pop-Up.</a>
```

Dieser Link stellt den Inhalt eines DIV Containers als Pop-Up dar, wenn der DIV Container die folgenden Klassen hat: *popup-wrapper popup popup-1*.

Der Inhalt des Pop-Ups müsste also, auf der gleichen HTML Seite, in dieser Form notiert werden:

```
<div class="popup-wrapper popup popup-1">Inhalt des Pop-Ups.</div>
```

Glossar

Die Inhalte des Glossars befinden sich in der Datei **classes/Glossar.class.js**. Jeder Eintrag im Glossar besteht aus einem *Begriff* und einer *Definition*. Begriffe, die ein Glossar-Popup öffnen sollen, werden mit einem SPAN umschlossen, dass die Klassen *glossar Begriff* erhält, wobei *Begriff* mit der entsprechenden Bezeichnung ersetzt wird, die in **classes/Glossar.class.js** verwendet wurde. Ein Verweis auf den Glossar-Eintrag für KOBV sieht beispielsweise so aus:

```
<span class="glossar KOBV">Hier erfahren Sie mehr über KOBV.</span>
```

Hinzufügen neuer Seiten

Die Kapitel des Tutorials können beliebig durch neue Unterkapitel erweitert werden. Für jedes neue Unterkapitel muss dazu eine neue HTML Seite angelegt werden. Diese HTML Seite braucht weder HEAD noch BODY Tags, sondern lediglich den eigentlichen Inhalt, der dargestellt werden soll. Um sicher zu stellen, dass das Tutorial die HTML Seite findet, müssen zwei Dinge beachtet werden:

- Die HTML Seite muss im Ordner des entsprechenden Kapitels unter **directory/** abgelegt werden.
- Die HTML Seite muss der **directory.json** hinzugefügt werden. Dabei kann sich an den bestehenden Einträgen orientiert werden. Jeder Eintrag besteht aus einem *title* und einem *link*. Der *title* wird vom Tutorial im Inhaltsverzeichnis eingesetzt. Der *link* muss den Namen der HTML Seite haben - **ohne die Dateiendung**, also ohne ".html".

CSS Regeln, die nur auf dieser jeweiligen Unterseite Gültigkeit haben, so wie Javascript für Mini-Spiele auf der jeweiligen Unterseite, werden direkt in die jeweilige HTML Seite aufgenommen.