
Protokoll

Wordpress Plugin

INSY
4AHITM 2015/16

Benedikt Berger

Version 1.0

Note:

Begonnen am 13. April 2016

Betreuer:

Beendet am 20. April 2016

Inhaltsverzeichnis

1. Aufgabenstellung	3
2. Sourcecode	4
Aufgabe 1	4
Aufgabe 2	5
3. Plugin installieren	7
Installation über Wordpressoberfläche	7
Installation über FTP	8
4. Plugin benutzen	9
5. Literaturverzeichnis	10

1. Aufgabenstellung

Einleitung

Stösst man bei der Verwendung eines CMS an die Grenzen der verfügbaren Erweiterungen (Plugins, Extensions, Module, ...), bieten alle gängigen CMS Systeme Schnittstellen zur Anpassung an. Zum Beispiel stellt die Integration externer Datenquellen eine solche Herausforderung dar.

Ziel

Es soll eine Wordpress-Seite aufgesetzt werden, die Daten aus der SchokoDB anzeigen kann, sowohl statisch als auch interaktiv abhängig von Benutzereingaben.

Aufgabenstellung

Installiere und konfiguriere eine Wordpress-Seite fuer unsere Schokofabrik. Die Seite muss keine besonderen statischen Inhalte aufweisen.

Entwickle ein Wordpress-Plugin [1], dass du verwendest, das auf zwei Seiten eingebunden wird:

1. Schnäppchen sind entweder Kunstwerke mit einem Schätzwert unter 2000 Euro oder Süßigkeiten unter 3 Euro. Gib auf der Seite eine Liste von Schnäppchen (bezeichnung und gewicht) aus. Vermerke dabei auch in einer eigenen Spalte, ob es sich um ein Kunstwerk oder ein Produkt aus dem Standardsortiment handelt.
2. Auf einer anderen Seite soll der Benutzer die Möglichkeit haben, eine der in der Datenbank angelegten Kunstschaue auszuwählen. Anschliessend soll zur gewählten Kunstschau eine Liste mit allen ausgestellten Kunstwerken samt erreichten Plätzen angezeigt werden.

Hinweise

- Verwende dazu die mysql-Version der schokoDB (ist im Kurs hochgeladen).
- Nutze Statements um sql injections zu vermeiden.
- Abzugeben sind ein Arbeitsprotokoll sowie das php Plugin

[1] https://codex.wordpress.org/Writing_a_Plugin

[ELE01]

2. Sourcecode

Aufgabe 1

```
function kunstwerk_func(){

    global $wpdb;
    ob_start();

    $res2 = $wpdb->get_results("select bezeichnung, gewicht, 'Kunstwerk' as ist from
    Produkt NATURAL join Kunstwerk where schaeztwert<2000 union select bezeichnung, gewicht,
    'Standard' as ist from Produkt natural join Standardsortiment where preis<3");

    $text = "";
    echo "<h4>billige Kunstwerke: </h4>";
    echo "<table>";
    echo "<tr><td><b>Bezeichnung</b></td><td><b>Gewicht</b></td></tr>";
    foreach($res2 as $r) {
        echo "<tr><td>";
        echo $r->bezeichnung;
        echo "</td><td>";
        echo $r->gewicht;
        echo "</td>";
    }

    echo "</table>";

    return ob_get_clean();
}

add_shortcode('kunstwerk','kunstwerk_func');
```

Die Funktion `kunstwerk_func` wird aufgerufen, wenn in einem Beitrag der shortcode `[kunstwerk]` benutzt wird. Zu allererst wird auf die Datenbank von Wordpress zugegriffen, auf welcher sich auch die Daten der Schokofabrik befinden. Dieser Aufbau ist mit aktuellem Wissensstand falsch, da die Datenbank von Schokofabrik, also auf welche Plugins zugreifen und die Datenbank von Wordpress getrennt sein sollen. In der Variable `$res2` wird der Output der angegebenen SELECT Anweisung abgespeichert. Diese Daten können über eine `foreach`-Schleife ausgegeben werden, wie in diesem Beispiel in einer Tabelle.

Aufgabe 2

```

function schokoKunstschauen()
{
    global $wpdb;
    ob_start();

    ?>

    <form method="GET">
        <select name="kunstschau">
            <?php
                $kunstschauen = $wpdb->get_results("SELECT * FROM Kunstschau");
                foreach ($kunstschauen as $k) {
                    list($name, $datum) = explode("|", $_REQUEST['kunstschau']);
                    if ($k->name == $name && $k->datum == $datum) {
                        echo '<option value="" . $k->name . '|' . $k->datum . "" selected="selected">' .
                            $k->name . ' (' . date('Y', strtotime($k->datum)) . ').' .
                            '</option>';
                    } else {
                        echo '<option value="" . $k->name . '|' . $k->datum . "">' .
                            $k->name . ' (' . date('Y', strtotime($k->datum)) . ').' .
                            '</option>';
                    }
                }
            ?>
        </select>
        <input type="submit" value="Anzeigen" class="btn btn-default">
    </form>

    <?php
    if (isset($_REQUEST['kunstschau'])) {
        ?>
        <br>
        <table class="table table-striped table-hover">
            <tr>
                <th>Kunstwerk</th>
                <th>Platz</th>
            </tr>
            <?php
                list($name, $datum) = explode("|", $_REQUEST['kunstschau']);
                $kunstwerke = $wpdb->get_results($wpdb->prepare('SELECT bezeichnung, platz FROM Produkt ' .
                    'INNER JOIN zeigt ON (Produkt.nummer = zeigt.kunstwerknummer) WHERE name = %s AND datum =
                    %s', $name, $datum));

```

```
foreach ($kunstwerke as $kunstwerk) {  
    ?>  
    <tr>  
        <td>  
            <?php echo $kunstwerk->bezeichnung; ?>  
        </td>  
        <td class="small">  
            <?php echo $kunstwerk->platz; ?>  
        </td>  
    </tr>  
    <?php  
  
    }  
}  
  
echo '</table>';  
echo '<br>';  
echo '<br>';  
return ob_get_clean();  
}
```

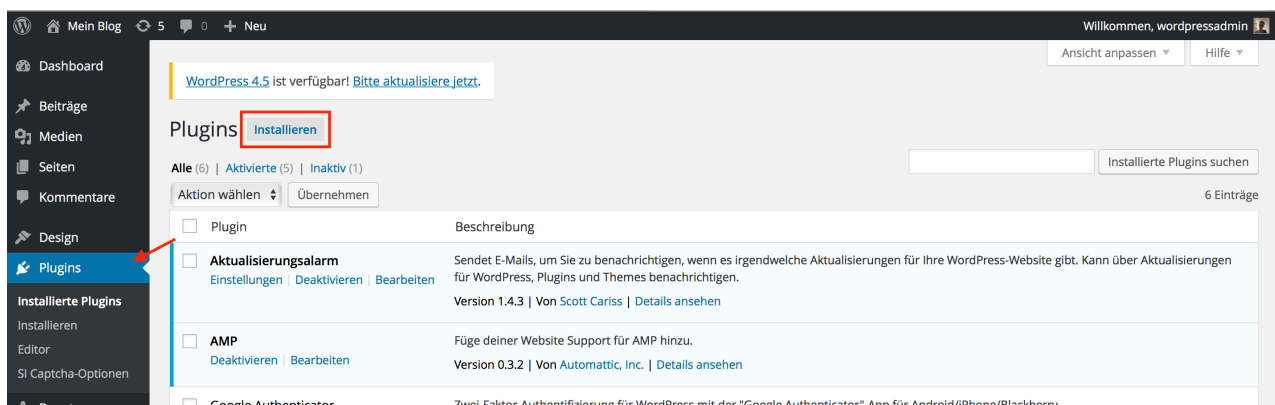
Zu allererst wird auf die Datenbank von Wordpress zugegriffen, auf welcher sich auch die Daten der Schokofabrik befinden. Dieser Aufbau ist mit aktuellem Wissensstand falsch, da die Datenbank von Schokofabrik, also auf welche Plugins zugreifen und die Datenbank von Wordpress getrennt sein sollen.

Es wird ein Formular erstellt, welches eine Combobox mit den verfügbaren Kunstschaueu auflistet. Durch das klicken auf den dazugehörigen Button, werden alle Kunstwerke dieser Kunstschau angezeigt.

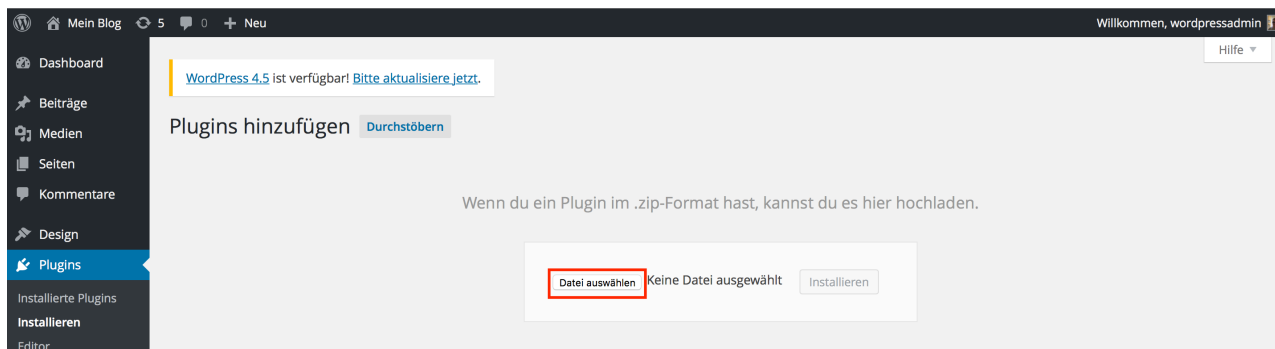
3. Plugin installieren

Installation über Wordpressoberfläche

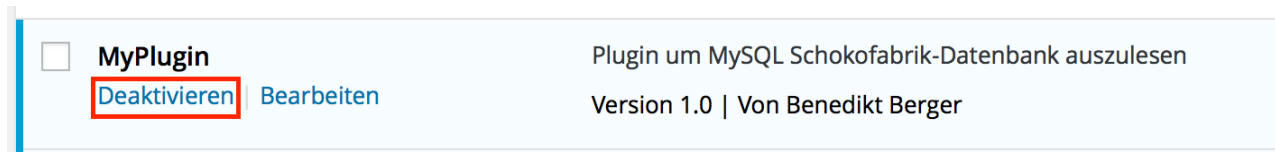
Zuerst muss in die Admin-Oberfläche von Wordpress eingestiegen werden. Hier kann man dann im linken Menü unter *Plugins* und anschließend durch klicken des Buttons *Installieren* ein neues Plugin hinzufügen. Im nächsten Fenster wird auf den Button *Plugin hochladen* ein neues Plugin hochgeladen.



Im nächsten Fenster kann ein zip-Archiv hochgeladen werden, welches das php-File des Plugins enthält.



Zu allerletzte muss das Plugin nur noch in der Plugins Übersicht aktiviert werden.



[WP01]

Installation über FTP

Um das Plugin über den Plugin Ordner, direkt auf dem Server zu installieren kann wie folgt vorgegangen werden:

Zu allererst muss zu dem Plugins Ordner von Wordpress navigiert werden. Dieser Ordner befindet sich im Verzeichnis `/wp-content/plugins`.

In dem Ordner Plugins, muss jetzt, um ein neues Plugin hinzufügen zu können, ein neuer Ordner mit dem Namen des Plugins erstellt werden. In diesen Ordner werden einfach die Sourcecode Files platziert, in meinem Fall das File *BergerPlugin.php*.

Server: /tgm/wp-content/plugins					
Dateiname ^	Dateigröße	Dateityp	Zuletzt geändert	Berechtigung	Besitz
..					
BergerPlugin		Verzeichnis	16.03.2016 12:35:50	0755	2647
amp		Verzeichnis	15.03.2016 13:15:30	0755	2647
google-authenticator		Verzeichnis	03.02.2016 11:12:56	0755	2647
remove-xmllrpc-pingback-ping		Verzeichnis	03.02.2016 11:12:20	0755	2647
si-captcha-for-wordpress		Verzeichnis	08.03.2016 14:27:41	0755	2647

4. Plugin benutzen

Um das Plugin jetzt nutzen zu können, kann beispielsweise in einem neuen Beitrag oder einer neuen Seite der im php-File beschriebene Shortcode genutzt werden.

Shortcodes

können auch als Makros bezeichnet werden und bieten die Möglichkeit Code zu ersetzen durch Funktionen von Plugins beispielsweise. Es gibt zahlreiche schon eingebaute Shortcodes zu benutzen, allerdings gibt es auch Plugins, welche Shortcodes hinzufügen, wie in unserem Beispiel.

Hier ist ein Beispiel für einen Shortcode, wie er auch in diesem Beispiel benutzt wird und einfach in Beiträge eingefügt werden kann:

[kunstwerk]

5. Literaturverzeichnis

- [WP01] BLOG TOOL, PUBLISHING PLATFORM. WORDPRESS [ONLINE] AVAILABLE AT:
[HTTPS://WORDPRESS.ORG](https://wordpress.org)
[ABGERUFEN AM 19. APRIL 2016]
- [ELE01] AUFGABE. TGM MOODLE [ONLINE] AVAILABLE AT:
[HTTPS://ELEARNING.TGM.AC.AT/MOD/ASSIGN/VIEW.PHP?](https://elearning.tgm.ac.at/mod/assign/view.php?id=47516&action=view)
[ID=47516&ACTION=VIEW](https://elearning.tgm.ac.at/mod/assign/view.php?id=47516&action=view)
[ABGERUFEN AM 19. APRIL 2016]