

Wordpress - Bootstrap Blog

Markus Reichl, 28. Januar 2018

Inhaltsverzeichnis

1	Aufgabe	1
2	Installation	2
3	Das Design	3
4	Aufteilung in Module	4
4.1	Header	4
4.2	Footer	6
4.3	Sidebar	7
4.4	Content	7
5	Kombination	8

1 Aufgabe

Es soll ein Webauftritt für ein Diplomprojekt mittels Wordpress, inklusive Präsentation des Projekts und mehreren Postings über den Projektfortschritt gestaltet werden. Relevant hier weniger der Inhalt sondern vielmehr, sondern die Umsetzung. Unsere Ziele sind dabei:

- Das Design baut auf Bootstrap (oder auch einem anderen CSS Framework)
- Es soll ein eigenes Wordpress-Theme, zur Darstellung der Inhalte aus den Posts auf Basis einer Vorlage gestaltet werden.
- Es soll ein eigenes Wordpress-Plugin entwickelt und hinzugefügt werden.
- Inhalte sollen mit speziellen Posts und eigenen Feldern eingebaut werden.

2 Installation

Wordpress[1] kann über die Seite[2] des Content Management System (CMS) heruntergeladen und nach der beiliegenden Anleitung[3] installiert werden.

Wir installieren Wordpress, zusammen mit einer MySQL[6] Datenbank und den zugehörigen Abhängigkeiten über den Virtualisierungsdienst Docker[4].

Die Konfiguration, sowie die Steuerung wird dabei über das Docker Compose[5] Tool vorgenommen, wobei die `docker-compose.yml` wie folgt definiert wird.

```
version: '3.1'

services:
  wordpress:
    image: wordpress
    restart: always
    ports:
      - 8080:80
    environment:
      WORDPRESS_DB_PASSWORD: insywp
    volumes:
      - './var/www/html:/var/www/html'
    container_name: insy-wordpress

  mysql:
    image: mysql:5.7
    restart: always
    environment:
      MYSQL_ROOT_PASSWORD: insywp
    volumes:
      - './var/lib/mysql:/var/lib/mysql'
    container_name: insy-wordpress-mysql
```

Der Container kann dabei über `docker-compose up -d` gestartet und mittels `docker-compose down` heruntergefahren werden.

Die Ordner `/var/lib/mysql/` und `/var/www/html/` werden lokal gespeichert um Änderungen persistent zu halten, auch wenn der Container neu erstellt wird.

3 Das Design

Als Basis für das eigene Theme verwenden wir das Blog Beispiel[7] von Bootstrap und passen dieses anhand der vorhandenen Wordpress Themes an. Zu diesem Zweck speichern wir die Seite einfach mittels STRG+S unter `var/www/html/wp-content/themes/bootstrap/`. Die `index.html` Datei muss nun nur noch in `index.php` umbenannt werden und wir haben ein neues Theme!

Nun wollen wir Bootstrap über ein Content Delivery Network (CDN) einbinden. Dazu werden die Zeilen

```
<!-- Bootstrap core CSS -->
<link href="../../../../dist/css/bootstrap.min.css" rel="stylesheet">

...

<!-- Bootstrap core JavaScript -->
<script src="https://code.jquery.com/jquery-3.2.1.slim.min.js"
  integrity="sha384-KJ3o2DKtIkvYIK3UENzmM7KCkRr/rE9/Qpg6aAZGJwFDMVNA/GpGFF93hJ
  ↳ XpG5KkN"
  crossorigin="anonymous"></script>
<script>window.jQuery || document.write(
  '<script src="../../../../assets/js/vendor/jquery-slim.min.js"><\script>'
)</script>
<script src="../../../../assets/js/vendor/popper.min.js"></script>
<script src="../../../../dist/js/bootstrap.min.js"></script>
<script src="../../../../assets/js/vendor/holder.min.js"></script>
```

durch die jeweiligen Links ersetzt:

```
<!-- Bootstrap core CSS -->
<link href="../../../../dist/css/bootstrap.min.css" rel="stylesheet">

...

<!-- Bootstrap core JavaScript -->
<script src="https://code.jquery.com/jquery-3.2.1.slim.min.js"
  integrity="sha384-KJ3o2DKtIkvYIK3UENzmM7KCkRr/rE9/Qpg6aAZGJwFDMVNA/GpGFF93hJ
  ↳ XpG5KkN"
  crossorigin="anonymous"></script>
<script src="https://cdnjs.cloudflare.com/
  ajax/libs/popper.js/1.12.9/umd/popper.min.js"
  integrity="sha384-ApNbgh9B+Y1QKtv3Rn7W3mgPxhU9K/ScQsAP7hUibX39j7fakFPskvXusJ
  ↳ vfa0b4Q"
  crossorigin="anonymous"></script>
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/js/bootstrap.min.js"
  integrity="sha384-JZR6Spejh4U02d8j0t6vLEHfe/JQGiRRSQQxSfFWpi1MquVdAyjUar5+7J
  ↳ 6PVCmYl"
  crossorigin="anonymous"></script>
```

Leider müssen in Wordpress auch die relativen Links angepasst werden, sodass diese mittels PHP an den jeweiligen Zielort angepasst werden. In diesem Sinne wird vor jedem Pfad folgendes Script eingefügt:

```
<?php echo get_bloginfo('template_directory'); ?>
```

4 Aufteilung in Module

Um das Theme effektiv verwenden zu können sollte dieses in mehrere Komponenten aufgeteilt werden, welche an verschiedenen Pfaden wiederverwendet werden können.

In diesem Beispiel werden wir unseren Blog in die Abschnitte `header.php`, `footer.php`, `sidebar.php` und `content.php` gliedern.

4.1 Header

Der gesamte Kopfbereich von

```
<!DOCTYPE html>
```

bis zum Titel des Blogs bildet den sogenannten Header. Dieser kann zu Beginn jeder Unterseite eingebunden werden und beinhaltet unter anderem importierte Ressourcen. In unserem Fall sieht die **header.php** Datei wie folgt aus:

```
1 <!DOCTYPE html>
2 <html lang="en"><head>
3     <meta http-equiv="content-type" content="text/html; charset=UTF-8">
4     <meta charset="utf-8">
5     <meta name="viewport" content="width=device-width, initial-scale=1,
6     ↪ shrink-to-fit=no">
7     <meta name="description" content="">
8     <meta name="author" content="">
9     <link rel="icon" href="http://getbootstrap.com/favicon.ico">
10
11     <title>Blog Template for Bootstrap</title>
12
13     <!-- Bootstrap core CSS -->
14     <link rel="stylesheet"
15     ↪ href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css"
16     ↪ integrity="sha384-Gn5384xqQ1aoWXA+058RXPxPg6fy4IWvTNh0E263XmFcJlSAwiGgFAW/
17     ↪ dAiS6JXm"
18     ↪ crossorigin="anonymous">
19
20     <!-- Custom styles for this template -->
21     <link href="<?php echo get_bloginfo('template_directory'); ?>/css/css.css"
22     ↪ rel="stylesheet">
23     <link href="<?php echo get_bloginfo('template_directory'); ?>/css/blog.css"
24     ↪ rel="stylesheet">
25 </head>
```

```
23 <body>
24 <div class="container">
25   <header class="blog-header py-3">
26     <div class="row flex-nowrap justify-content-between align-items-center">
27       <div class="col-4 pt-1">
28         <a class="text-muted" href="#">Subscribe</a>
29       </div>
30       <div class="col-4 text-center">
31         <a class="blog-header-logo text-dark" href="#">Large</a>
32       </div>
33       <div class="col-4 d-flex justify-content-end align-items-center">
34         <a class="text-muted" href="#">
35           <svg xmlns="http://www.w3.org/2000/svg" width="20" height="20"
36             viewBox="0 0 24 24" fill="none"
37             stroke="currentColor" stroke-width="2"
38             stroke-linecap="round" stroke-linejoin="round" class="mx-3">
39             <circle cx="10.5" cy="10.5" r="7.5"></circle>
40             <line x1="21" y1="21" x2="15.8" y2="15.8"></line>
41           </svg>
42         </a>
43         <a class="btn btn-sm btn-outline-secondary" href="#">Sign up</a>
44       </div>
45     </div>
46   </header>
47
48   <div class="nav-scroller py-1 mb-2">
49     <nav class="nav d-flex justify-content-between">
50       <a class="p-2 text-muted" href="#">Technology</a>
51       <a class="p-2 text-muted" href="#">Design</a>
52       <a class="p-2 text-muted" href="#">Business</a>
53       <a class="p-2 text-muted" href="#">Politics</a>
54       <a class="p-2 text-muted" href="#">Science</a>
55     </nav>
56   </div>
57 </div>
```

4.2 Footer

Der Footer beinhaltet häufig hinzugefügte Scripts oder Informationen zur Seite selbst. In unserem Fall ist die **footer.php** Datei wie folgt strukturiert:

```

1 <footer class="blog-footer">
2   <p>Blog template built for <a href="https://getbootstrap.com/">Bootstrap</a> by
   ↳ <a href="https://twitter.com/mdo">@mdo</a>.
3   </p>
4   <p>
5     <a href="#">Back to top</a>
6   </p>
7 </footer>
8
9 <!-- Bootstrap core JavaScript -->
10 <script src="https://code.jquery.com/jquery-3.2.1.slim.min.js"
11   integrity="sha384-KJ3o2DKtIkvYIK3UENzmM7KCKRr/rE9/Qpg6aAZGJwFDMVNA/GpGFF93hX
   ↳ pG5KkN"
12   crossorigin="anonymous"></script>
13 <script
14   ↳ src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.12.9/umd/popper.min.js"
   integrity="sha384-ApNbgh9B+Y1QKtv3Rn7W3mgPxhU9K/ScQsAP7hUibX39j7fakFPskvXusv
   ↳ fa0b4Q"
15   crossorigin="anonymous"></script>
16 <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/js/bootstrap.min.js"
17   integrity="sha384-JZR6Spejh4U02d8j0t6vLEHfe/JQGiRRSQQxSfFWpi1MquVdAyjUar5+76
   ↳ PVCmYl"
18   crossorigin="anonymous"></script>
19
20 <script src="<?php echo get_bloginfo('template_directory');
   ↳ ?>/css/holder.js"></script>
21 <script>
22   Holder.addTheme('thumb', {
23     bg: '#55595c',
24     fg: '#eceeef',
25     text: 'Thumbnail'
26   });
27 </script>
28
29
30 <svg xmlns="http://www.w3.org/2000/svg" width="200" height="250"
31   viewBox="0 0 200 250" preserveAspectRatio="none"
32   style="display: none; visibility: hidden; position: absolute; top: -100%; left:
   ↳ -100%;">
33   <defs><style type="text/css"></style></defs>
34   <text x="0" y="13" style="font-weight:bold;font-size:13pt;font-family:Arial,
   ↳ Helvetica, Open Sans, sans-serif">Thumbnail</text>
35 </svg>
36 </body>
37 </html>

```

4.3 Sidebar

Die Sidebar kann als `texttttsidebar.php` ebenfalls ausgelagert werden:

```

1 <aside class="col-md-4 blog-sidebar">
2   <div class="p-3 mb-3 bg-light rounded">
3     <h4 class="font-italic">About</h4>
4     <p class="mb-0">Etiam porta <em>sem malesuada magna</em> mollis euismod. Cras
      ↪ mattis consectetur purus
5       sit amet fermentum. Aenean lacinia bibendum nulla sed consectetur.</p>
6   </div>
7
8   <div class="p-3">
9     <h4 class="font-italic">Archives</h4>
10    <ol class="list-unstyled mb-0">
11      <li><a href="#">February 2018</a></li>
12      <li><a href="#">January 2018</a></li>
13    </ol>
14  </div>
15
16  <div class="p-3">
17    <h4 class="font-italic">Elsewhere</h4>
18    <ol class="list-unstyled">
19      <li><a href="#">GitHub</a></li>
20      <li><a href="#">Twitter</a></li>
21      <li><a href="#">Facebook</a></li>
22    </ol>
23  </div>
24 </aside>

```

4.4 Content

Die einzelnen Einträge werden ebenfalls getrennt verwaltet und über die `content.php` Datei dargestellt:

```

1 <div class="row">
2   <div class="col-md-8 blog-main">
3     <h3 class="pb-3 mb-4 font-italic border-bottom">
4       From the Firehose
5     </h3>
6
7     <div class="blog-post">
8       <h2 class="blog-post-title">Sample blog post</h2>
9       <p class="blog-post-meta">January 1, 2014 by <a href="#">Mark</a></p>
10      <p>...</p>
11    </div>
12  </div>
13 </div>

```

5 Kombination

Kombiniert werden die einzelnen Komponenten nun in der **index.php** Datei:

```
1 <?php get_header(); ?>
2
3 <main role="main" class="container">
4     <?php get_template_part( 'content', get_post_format() ); ?>
5     <?php get_sidebar(); ?>
6 </main>
7
8 <?php get_footer(); ?>
```


Literatur

- [1] Wordpress. Blog Tool, Publishing Platform, and CMS
<https://wordpress.org>
- [2] Wordpress. Download WordPress
<https://wordpress.org/latest.zip>
- [3] Wordpress. Installing WordPress
https://codex.wordpress.org/Installing_WordPress
- [4] Docker. Build, Ship and Run Any App, Anywhere
<https://www.docker.com>
- [5] Docker. Docker Compose | Docker Documentation
<https://docs.docker.com/compose/>
- [6] MySQL
<https://www.mysql.com/>
- [7] Blog Template for Bootstrap
<http://getbootstrap.com/docs/4.0/examples/blog/>