

Initiales Repo aufsetzen

Kevin Pfeifer – Version 1.0, 2019-07-03

1. Ein neues Repository bei Bitbucket erstellen und dieses lokal auschecken (Beispiel: kcc)
2. Den Master Branch von unserem wordpress Repository herunterladen
https://bitbucket.org/sunlimeitservices/wordpress/src/2019_git_local_setup_oop/
Über das linke Menü "Downloads" ⇒ Branches ⇒ "master" ZIP
3. Den Inhalt dieses ZIPs in das zuvor ausgecheckte neue Repository entpacken.
Darauf achten dass die .gitignore mit kopiert wird!
4. Die mktheme.sh öffnen und die Werte NAME , SLUG und DESC durch die eigenen Theme Strings ersetzen.
5. Die mktheme.sh ausführen ⇒ ./mktheme.sh und danach löschen.
6. Diesen Stand in das Repository commiten und pushen.
7. Am Batman Server mit dem User batcave anmelden und in den Ordner chroot gehen.
8. Dort das zuvor neu erstellte Repository auschecken. Damit sollte an unserem Beispiel ein Ordner kcc im chroot Order liegen.
9. npm i am Batman in dem ausgecheckten Repository durchführen.
10. Nun sollte die Seite zumindest über folgende URL aufrufbar sein: <https://<repo-name>.sunlime.io>
11. Über <https://batman.sunlime.at/phpmyadmin/> sich mit dem batcave Benutzer anmelden und eine neue Datenbank erstellen. An unserem Beispiel heist diese Datenbank dann batcave_kcc
12. Den Wordpress Installer auf der Domain <https://<repo-name>.sunlime.io> durchführen mit der zuvor neu erstellten Datenbank und den Anmeldedaten vom batcave Datenbank-Benutzer (siehe 1Password)
Die Serveradresse **muss** hier 127.0.0.1 sein da (wegen der aktuellen chroot-Umgebung) localhost nicht funktioniert.
13. In dem lokal ausgecheckten Repo das gulpfile.js öffnen und im Task uploads , db und default folgende Werte anpassen:

```
gulp.task('uploads', function() {  
  var remote_user = 'batcave';  
  var remote_server = 'batman.sunlime.at';  
  var remote_path = '/home/batcave/chroot/kcc/docroot/wp-content/uploads/';  
  ...  
});
```

JS

```

gulp.task('db', function() {
  var remote_user = 'batcave';
  var remote_server = 'batman.sunlime.at';
  var remote_path = '/home/batcave/chroot/kcc/docroot'; // without / at the end
  var domain = 'https://sunlime:sunlime.io@kcc.sunlime.io'; // without / at the end
  var mysql_running = fs.existsSync('/tmp/kcc.mysql.sock');

  if(!mysql_running) {
    if(!fs.existsSync('mysql_data')) {
      exec.execSync('mkdir mysql_data && /usr/local/mysql/bin/mysqld --skip-networking --initialize -h
./mysql_data');
    }
    exec.exec('/usr/local/mysql/bin/mysqld --skip-networking --socket=/tmp/kcc.mysql.sock --skip-grant-
tables -h ./mysql_data');
  }
  ...
});

```

```

gulp.task('default', function() {
  ...
  exec.exec('killall -9 mysqld; /usr/local/mysql/bin/mysqld --skip-networking --socket=/tmp/kcc.mysql.sock
--skip-grant-tables -h ./mysql_data');
  ...
});

```

- a. Beim uploads Task muss der remote_path der absolute Pfad zum Uploads Ordner sein
- b. Beim db Task muss der remote_path der absolute Pfad zum Document-Root sein
- c. Der MySQL-Socket muss hier überall /tmp/<repo-name>.mysql.sock sein, hier also /tmp/kcc.mysql.sock

14. Ebenso muss der Pfad zur mysql.sock Datei in der sqldump.php und php.ini Datei auch angepasst werden:

- a. <repo-name> ⇒ an unserem Beispiel kcc

```

1 <?php
2
3 /** PHP version of mysqldump cli that comes with MySQL ...*/
15
16 namespace Ifsnop\Mysqldump;
17
18 use ...
21
22 if(PHP_SAPI_NAME() === 'cli' && !file_exists( filename: 'docroot/wp-config.php') && file_exists( filename: 'docroot/wp-config-sample.php')) {
23     $chars = 'ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789!@#$%^&*+,-./:;<=>?@[_`{}~';
24
25     $sample = file( filename: 'docroot/wp-config-sample.php');
26
27     foreach($sample as $key => $line) {
28         if(strpos($line, needle: '$table_prefix') !== false) {
29             $sample[$key] = '$table_prefix = \'.exec( command: "grep \'DROP TABLE IF EXISTS\' sqldump.sql | cut -d _ -f 1 | cut -d \'\' -f 2")._\';'.PHP_EOL;
30         }
31
32         if(strpos($line, needle: 'Füge hier deine Zeichenkette ein') !== false) {
33             $sample[$key] = str_replace( search: 'Füge hier deine Zeichenkette ein', substr(str_shuffle($chars), start: 0, length: 64), $line);
34         }
35     }
36
37     unset($sample[0]);
38     array_unshift( &array: $sample, ...vars: 'define(\'WP_SITEURL\', \'http://localhost:8000\');'.PHP_EOL);
39     array_unshift( &array: $sample, ...vars: 'define(\'WP_HOME\', \'http://localhost:8000\');'.PHP_EOL);
40     array_unshift( &array: $sample, ...vars: '<?php'.PHP_EOL);
41
42     file_put_contents( filename: 'docroot/wp-config.php', $sample);
43
44
45     foreach(file( filename: PHP_SAPI_NAME() === 'cli' ? 'docroot/wp-config.php' : 'wp-config.php') as $line) {
46         if(strpos($line, needle: 'DB_HOST') !== false) eval($line);
47         if(strpos($line, needle: 'DB_USER') !== false) eval($line);
48         if(strpos($line, needle: 'DB_PASSWORD') !== false) eval($line);
49         if(strpos($line, needle: 'DB_NAME') !== false) eval($line);
50     }
51
52     if(PHP_SAPI_NAME() === 'cli') {
53         shell_exec( cmd: 'echo CREATE DATABASE IF NOT EXISTS \'.DB_NAME.\' | /usr/local/mysql/bin/mysql -S /tmp/<repo-name>.mysql.sock');
54         shell_exec( cmd: '/usr/local/mysql/bin/mysql -S /tmp/<repo-name>.mysql.sock \'.DB_NAME.\' < sqldump.sql');
55     } else {
56         $dump = new Mysqldump( dsn: 'mysql:host=\.DB_HOST.;dbname=\.DB_NAME; user= DB_USER, pass: DB_PASSWORD);
57         $dump->start();
58     }
59
60 /**
61  * PHP version of mysqldump cli that comes with MySQL
62  *
63  * mysql mysqldump pdo php7 php5 database php sql
64  *
65  * @category Library
66  * @package Ifsnop\Mysqldump
67  * @author Diego Torres <ifsnop@github.com>
68  * @license http://www.gnu.org/copyleft/gpl.html GNU General Public License
69  * @link https://github.com/ifsnop/mysqldump-php
70  *
71  */

```

```

1 zend_extension = xdebug.so
2 xdebug.remote_enable = 1
3
4 max_execution_time = 300
5 mysqli.default_socket = "/tmp/<repo-name>.mysql.sock"
6 pdo_mysql.default_socket = "/tmp/<repo-name>.mysql.sock"

```

15. Die Node-Modules installieren `npm i`

16. Der Befehl `gulp db` sollte nun ohne Fehlermeldung durchlaufen.

17. Mit dem Befehl `gulp` startet der lokale Webserver und die lokale Entwicklung der Webseite kann gestartet werden.

18. Damit die in das Repo gepushten Änderungen auch auf den "DEV"-Server durchgeführt werden muss ein Webhook bei Bitbucket eingerichtet werden.



Settings

GENERAL

- [Repository details](#)
- [User and group access](#)
- [Access keys](#)
- [Username aliases](#)

WORKFLOW

- [Branch permissions](#)
- [Branching model](#)
- [Merge strategies](#)
- [Webhooks](#)**
- [Links](#)
- [Services](#)

PULL REQUESTS

- [Default reviewers](#)
- [Default description](#)
- [Excluded files](#)

FEATURES

- [Git LFS](#)
- [Wiki](#)
- [Issue tracker](#)

TRELLO

- [Settings](#)

CHAT NOTIFICATIONS

- [Settings](#)

PIPELINES

- [Settings](#)


Webhooks

Webhooks allow you to extend what Bitbucket does when the repository changes (for example, new code is pushed or a pull request is merged).

To learn more about how webhooks work, check out the [documentation](#).


[Add webhook](#)

Title	URL	Actions
Bitbucket code search	https://bb-search-prod-ingester.prod.public.atl-paas.net/r...	View requests READ ONLY
Pipelines	https://bitbucket-pipelines.prod.public.atl-paas.net/rest/bit...	View requests READ ONLY
Pipelines	https://bitbucket-pipelines.prod.public.atl-paas.net/rest/bit...	View requests READ ONLY





Q


+





<>




















Sunlime Web Innovations / Intern / Wordpress

Settings

GENERAL

[Repository details](#)

[User and group access](#)

[Access keys](#)

[Username aliases](#)

WORKFLOW

[Branch permissions](#)

[Branching model](#)

[Merge strategies](#)

Webhooks

[Links](#)

[Services](#)

PULL REQUESTS

[Default reviewers](#)

[Default description](#)

[Excluded files](#)

Webhooks

Add new webhook

To learn more about how webhooks work, check out the [documentation](#).

Title

Batcave

URL

https://batman.sunlime.at/batcave.php

Status

☒ Active

Inactive webhooks don't trigger requests.

SSL / TLS

☐ Skip certificate verification

Untrusted or self-signed certificates may not be secure. [Learn more](#)

Request History

☐ Enable request history collection

Request history collection will be automatically disabled after 12 hours.

Triggers

☒ Repository push

☐ Choose from a full list of triggers

Save

Cancel

a. Title: Batcave

URL: https://batman.sunlime.at/batcave.php

Version 1.0

Last updated 2019-07-16 13:36:21 +0200