

Einrichtung des Backend

1. Benötigte Software

- Vagrant: <https://www.vagrantup.com/downloads.html>
- SSH Dienst für Windows: <https://www.putty.org>
- VirtualBox: <https://www.virtualbox.org/wiki/Downloads>

Nach den Vagrant erfolgreich installiert wurde, muss noch der Vagrant hostmanager installiert werden. Dies erledigt man mit folgenden Befehl im Terminal:

```
vagrant plugin install vagrant-hostmanager
```

2. Homestead einrichten

Um Homestead einzurichten, müssen folgende Befehle ausgeführt werden:

```
vagrant box add laravel/homestead (Als Provider muss VirtualBox ausgewählt werden)
```

Danach:

```
git clone https://github.com/laravel/homestead.git <Pfad_wo_gespeichert_werden_soll>
```

Dann muss zum Ordner gewechselt werden und folgende Befehle ausgeführt werden:

```
cd <Pfad_wo_gespeichert_wurde>  
git checkout v7.7.0  
init.bat
```

SSH-Key unter Unix/Linux basierten Systeme einrichten:

```
ssh-keygen -t rsa (als Name/Pfad sollte der Default Wert genommen werden)
```

SSH-Key unter Windows einrichten:

Unter Windows muss das Programm PuTTYgen ausgeführt werden und mit Generate ein neuer Key generiert werden. Ist dies geglückt, muss der Private Key sowie Public Key gespeichert werden (der Pfad muss sich gemerkt werden.)

Zurück zum Ordner, in dem Homestead eingerichtet wurde.

Nun muss die Homestead.yaml eingerichtet angepasst werden und muss wie folgt aussehen:

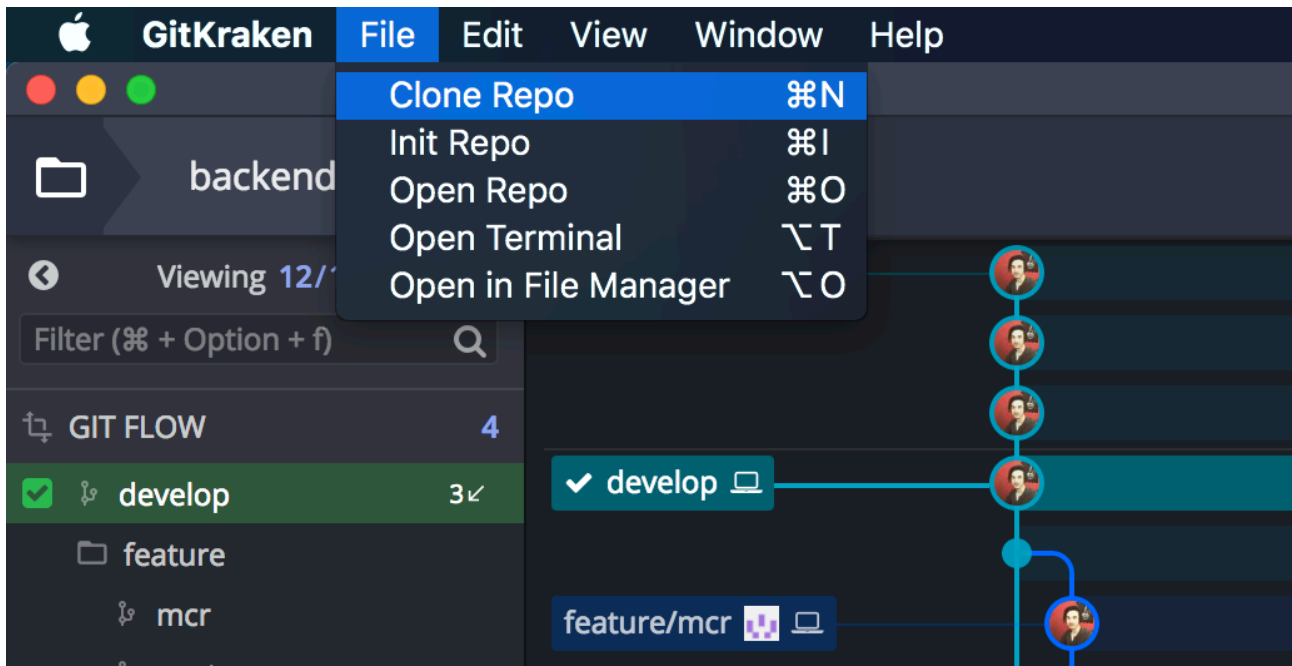
```
1  ip: "192.168.10.10"
2  memory: 2048
3  cpus: 1
4  provider: virtualbox
5
6  authorize: ~/.ssh/id_rsa.pub
7
8  keys:
9  |   - ~/.ssh/id_rsa
10
11  folders:
12  |   - map: ./code
13  |     to: /home/vagrant/code
14
15  sites:
16  |   - map: yourchoice.local
17  |     to: /home/vagrant/code/backend/public
18
19  databases:
20  |   - homestead
21
22  # blackfire:
23  #   - id: foo
24  #     token: bar
25  #     client-id: foo
26  #     client-token: bar
27
28  # ports:
29  #   - send: 50000
30  #     to: 5000
31  #   - send: 7777
32  #     to: 777
33  #     protocol: udp
34
```

Unter Windows muss allerdings “authorize” und “keys” angepasst werden. “authorize” ist der Pfad zum Public Key (Absolut Pfad) und “keys” zum Private Key (Absolute Pfad).

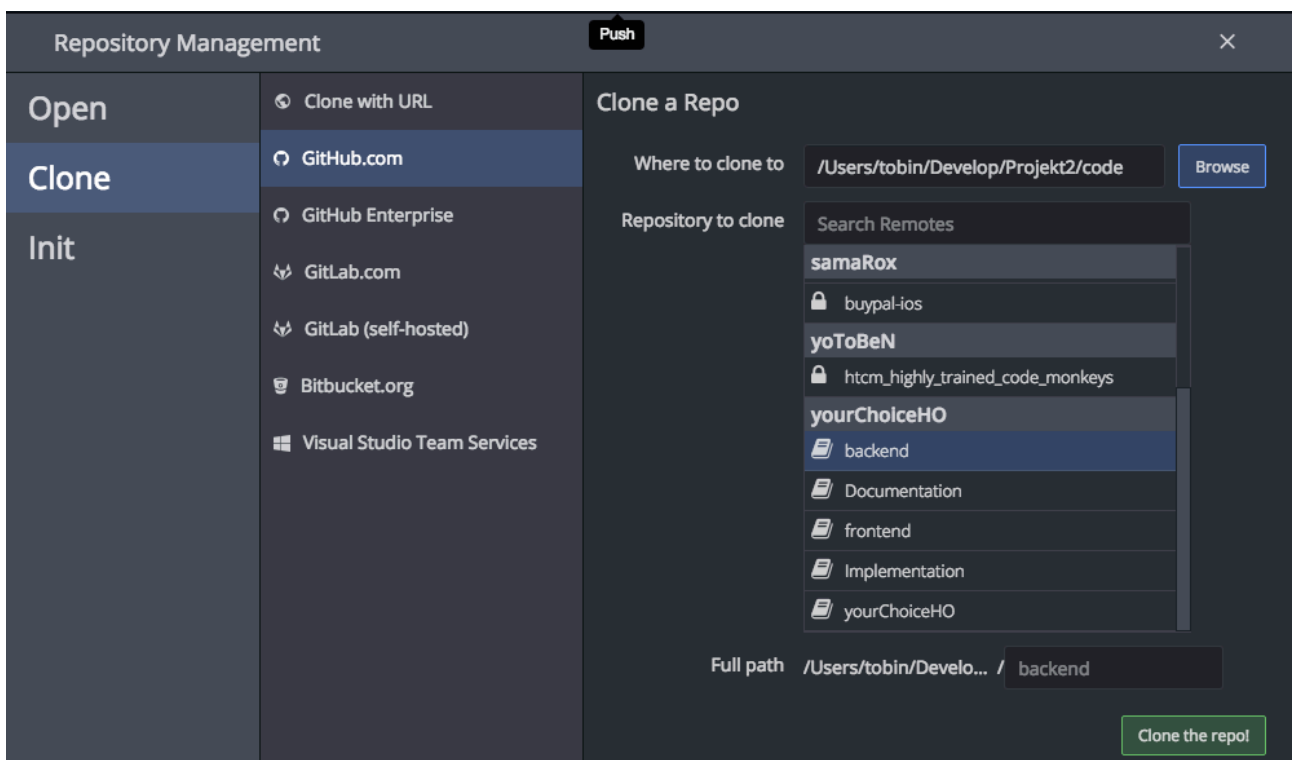
3.Backend einrichten

Um das Backend einzurichten muss in den Homestead Ordner, der Ordner "code" erstellt werden. Danach muss das Backend in diesen Ordner "gecloned" werden. Dies erledigt man wie folgt:

Als erstes muss das Backend aus GitHub gepullt werden. Dafür geht man unter GitKraken auf "File" -> "Clone Repo":



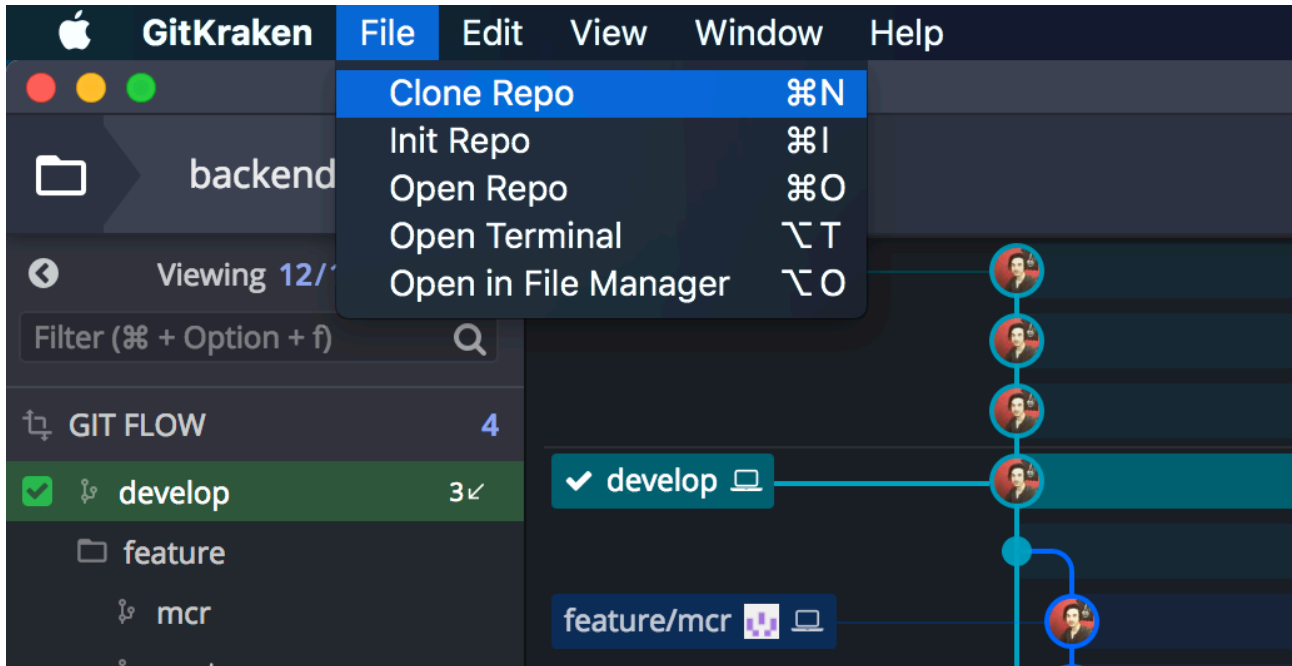
Danach wählen wir GitHub.com aus. Unter "Where to clone to" muss der code Ordner ausgewählt werden, der sich im *Homestead* Verzeichnis befindet. Als "Repository to clone" wird unter "yourChoiceHO" das "backend" ausgewählt ("Full Path" muss nicht erweitert/ausgefüllt werden).



4. Frontend aus dem Git pullen

INFO: Wenn das Frontend schon “gecloned”/“gepullt” wurde, muss nur der Ordner als frontend in den code Ordner verschoben werden, im frontend Ordner, der Ordner dist erstellt werden und danach kann man zum **Schritt 6. Homestead starten** springen.

Als erstes muss das Frontend aus GitHub gepullt werden. Dafür geht man unter GitKraken auf “File” -> “Clone Repo”:



Danach wählen wir [GitHub.com](https://github.com) aus. Unter “Where to clone to” muss der code Ordner ausgewählt werden, der sich im *Homestead* Verzeichnis befindet (im code Ordner befindet auch das *backend*). Als “Repository to clone” wird unter “yourChoiceHO” das “frontend” ausgewählt (“Full Path” muss nicht erweitert/ausgefüllt werden).

Wurde das Frontend erfolgreich “gepullt” muss im frontend Ordner noch ein “dist” Ordner erstellt werden. Der endgültige Pfad sollte dann sein `.../<Homestead>/code/frontend/dist`.

5. Frontend einrichten

Um das Frontend einzurichten, muss man mit dem Terminal in den Homestead Ordner gehen und die vagrant Maschine starten (`vagrant up`). Ist dies geglückt, muss man sich mit der vagrant Maschine verbinden (`vagrant ssh`).

Nun wechselt man in den frontend Ordner (`cd code/frontend`) und gibt folgenden Befehl ein:

```
npm install
```

Ist dies geglückt (Warnings können ignoriert werden), muss folgender Befehl ausgeführt werden:

6. Homestead starten

Um Homestead zu starten, müssen wir mit dem Terminal in den Homestead Ordner und dort `Vagrant up --provision` eingeben. Danach verbinden wir uns mit der Vagrant Maschine mit dem Befehl `vagrant ssh`. Danach müssen wir in den backend Ordner wechseln (`cd code/backend`) und folgende Befehle ausführen:

```
composer update
cp .env.example .env
php artisan key:generate
```

7.Backend updaten und Frontend “builden”

Ist dies dahin alle geglückt, kann das Backend auf den aktuellen Stand gebracht werden. Dafür wird der Branche *develop* genutzt. Danach müssen die Tabellen neu gesetzt werden, dafür muss man mit dem Terminal in den Homestead Ordner gehen und die vagrant Maschine starten (*vagrant up*). Ist dies geglückt, muss man sich mit der vagrant Maschine verbinden (*vagrant ssh*) und zum backend Ordner wechseln (*cd code/backend*). Zum Schluss müssen noch beide Befehle eingegeben werden:

```
php artisan migrate:fresh  
composer dump-autoload  
php artisan db:seed
```

Um das Frontend zu “builden”, muss man mit dem Terminal in den Homestead Ordner gehen und die vagrant Maschine starten (*vagrant up*). Ist dies geglückt, muss man sich mit der vagrant Maschine verbinden (*vagrant ssh*).

Nun wechselt man in den frontend Ordner (*cd code/frontend*) und gibt folgenden Befehl ein:

```
sudo npm run build
```

Jetzt sind Backend und Frontend vereint, und man müsste, wenn man im Browser <http://yourchoice.local> eingibt, auf die Anmelde Seite gelangen.