



# Building a World

Ich mach' mir die Welt – widdewidde wie sie mir gefällt...

domfi @ RFCCv1

2021-06-12

# Virtuelle Welten

- Kontaktbeschränkungen verhinder[n|ten] das normale Club-Leben
- Sprach- und Videochats wurden schnell genutzt
- Das Feeling von IRL Konferenzen kommt damit aber nicht auf
- Workadventure to the rescue!
  - 2D-Welten erstellen und mit Avataren drin rumlaufen
  - Jeder Club kann seine eigenen Räume schaffen
  - Wenn sich Avatare begegnen startet automatisch ein Videochat

# Getting started: GitHub-Template benutzen

- Man braucht einen GitHub-Account
- Siehe: <https://workadventu.re/map-building>
- Nachdem man ein neues Repo für seine Map erzeugt hat, wird dieses Repo lokal gecloned:

```
dominik@cray:~$ git clone https://github.com/domfi/myfirstwamap.git
Cloning into 'myfirstwamap'...
remote: Enumerating objects: 27, done.
remote: Counting objects: 100% (27/27), done.
remote: Compressing objects: 100% (21/21), done.
remote: Total 27 (delta 1), reused 21 (delta 1), pack-reused 0
Unpacking objects: 100% (27/27), 331.05 KiB | 4.87 MiB/s, done.
dominik@cray:~$ |
```

# Getting started: Node Module installieren

```
dominik@cray:~$ cd myfirstwamap/  
dominik@cray:~/myfirstwamap$ npm install  
npm WARN workadventure-map-starter-kit@1.0.0 No repository field.  
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.13 (node_modules/fsevents):  
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.13: wanted  
{"os":"darwin","arch":"any"} (current: {"os":"linux","arch":"x64"})  
  
added 673 packages from 379 contributors and audited 678 packages in 9.76s  
  
47 packages are looking for funding  
  run `npm fund` for details  
  
found 3 vulnerabilities (2 moderate, 1 high)  
  run `npm audit fix` to fix them, or `npm audit` for details  
  
New major version of npm available! 6.14.4 → 7.16.0  
Changelog: https://github.com/npm/cli/releases/tag/v7.16.0  
Run npm install -g npm to update!  
  
dominik@cray:~/myfirstwamap$ |
```

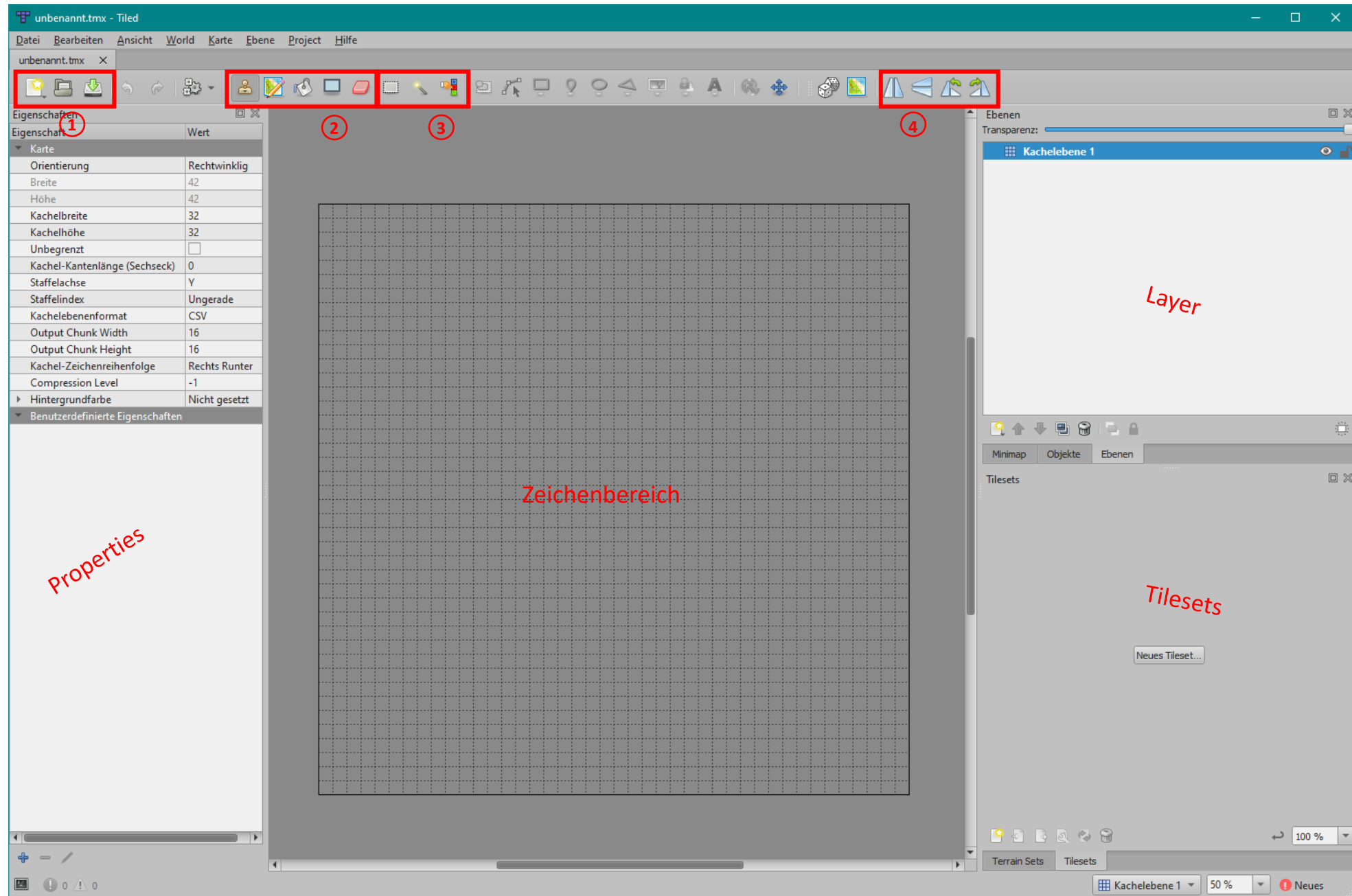
# Tiled – Tiled Map Editor

*„Tiled is a 2D level editor that helps you develop the content of your game.“*

<https://www.mapeditor.org/>

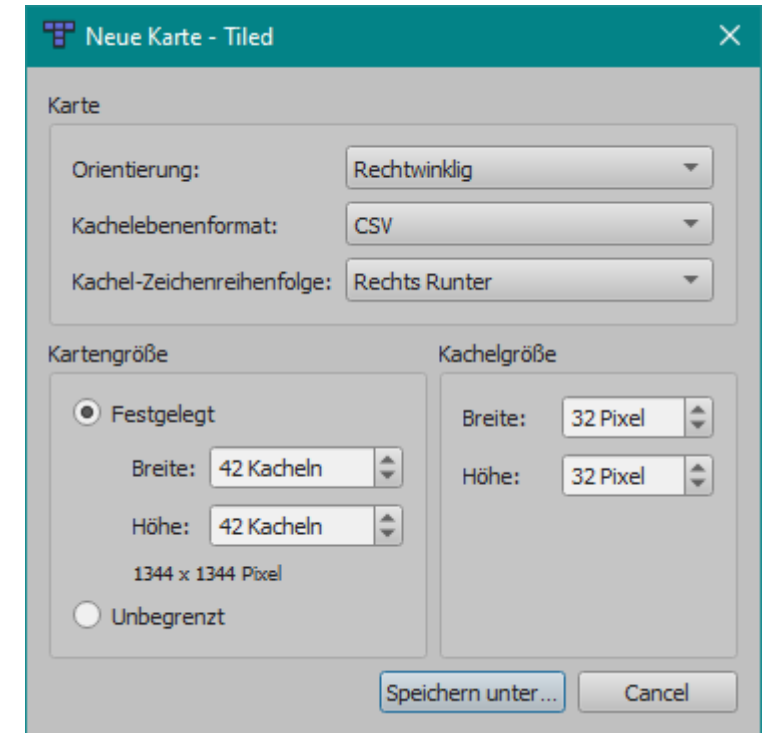
# Tiled

1. Dateioperationen
2. Malen/Löschen
3. Auswählen
4. Spiegeln/Drehen



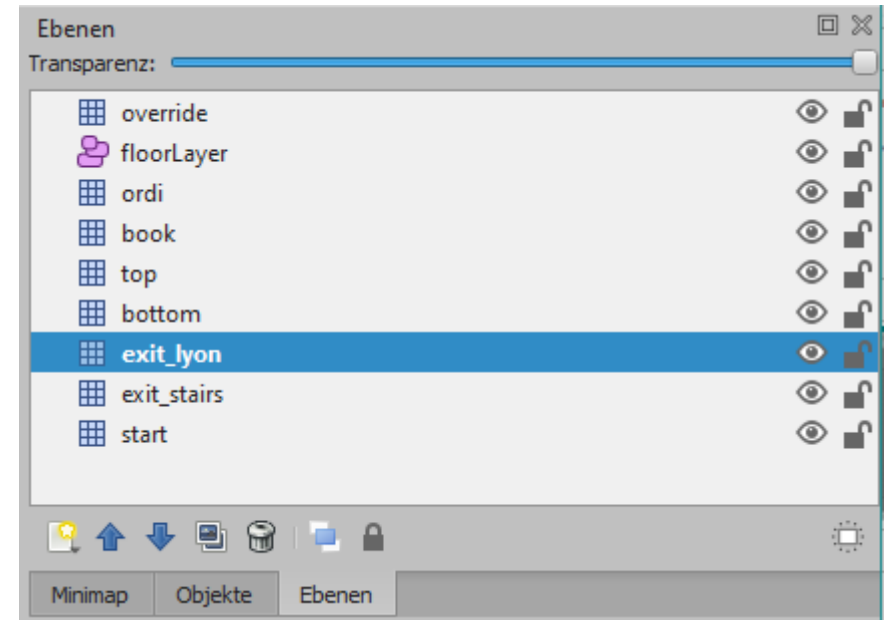
# Maps erstellen

- WA unterstützt nur orthogonale Karten
- Kachelgröße muss 32x32 Pixel sein
- Karten müssen im JSON Format gespeichert werden



# Layer

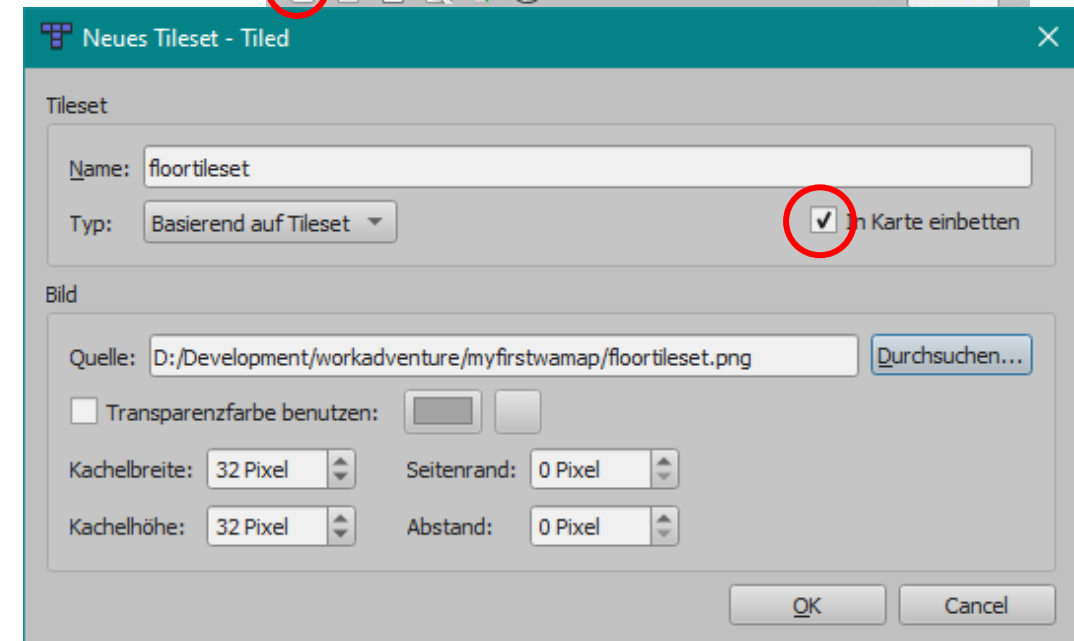
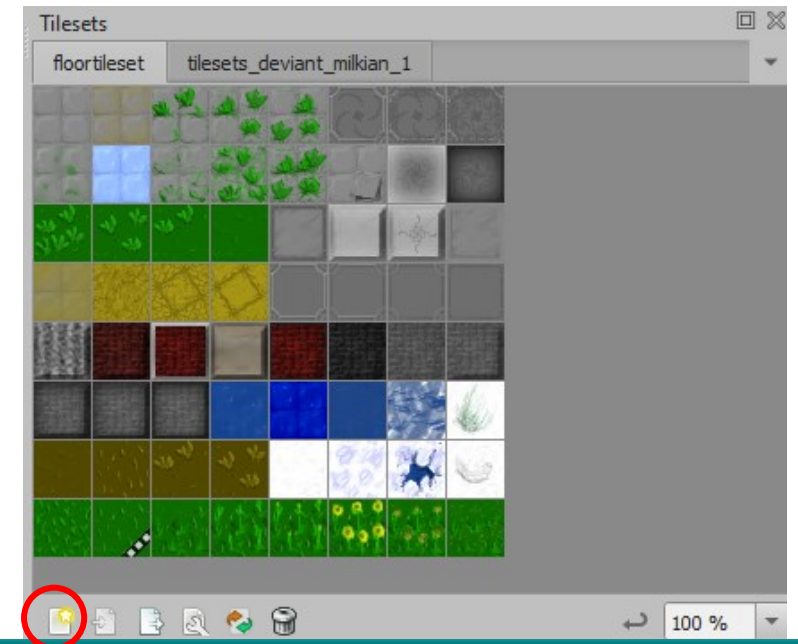
- Die Maps werden in Layern aufgebaut
- Fast alles wird mit Tile-Layern gemacht
- WA erfordert einige spezielle Layer
  - start
  - floorLayer
  - Layer für Ausgänge





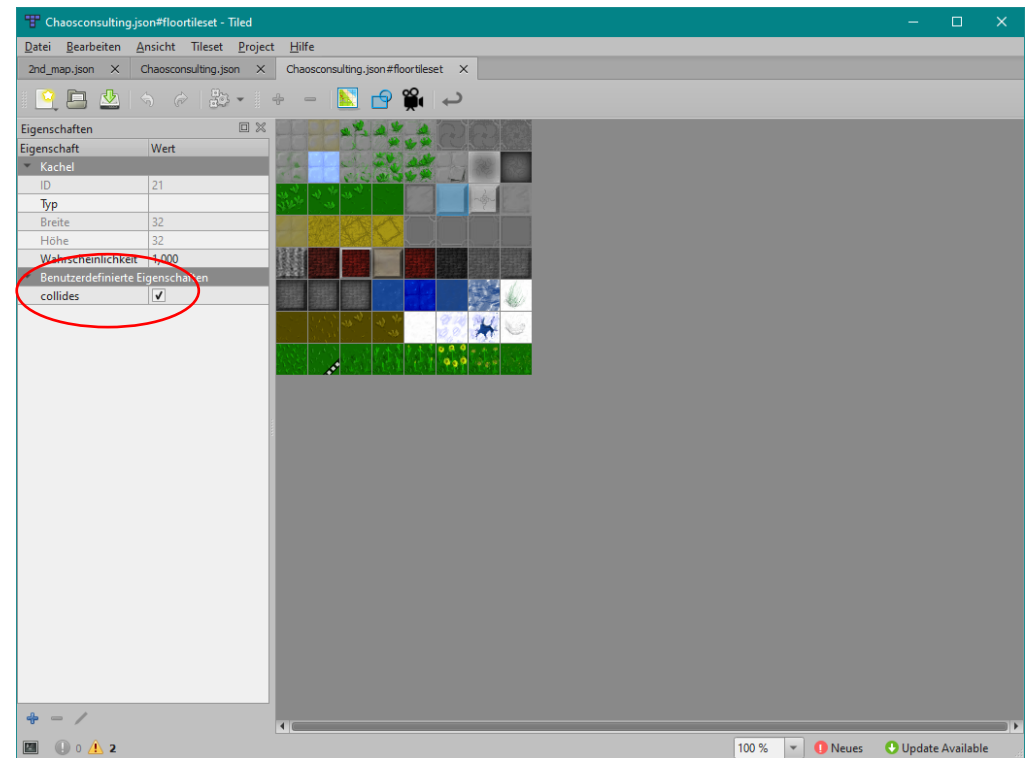
# Tilesets

- Tilesets fassen einzelne Tiles zusammen
- Durch Tiles mit der Eigenschaft `collides: true` kann man nicht durchgehen
- WA erfordert, daß Tilesets in die Map embedded werden
- Quellen für Tilesets z.B.:
  - <https://opengameart.org/>



# Tiles editieren

- Mit dem eingebauten Tileset-Editor kann man die Tiles bearbeiten
- Braucht man, um bestimmte Tiles in Hindernisse umzubauen:
  - Eigenschaft `collides` (boolean)



# Map malen

- Auswahl des Layers, auf dem die Tiles dargestellt werden sollen
- Aktivierung der Stamp-Funktion
- Auswahl eines oder mehrerer Tiles
- Stempeln...
- Bei Bedarf können die ausgewählten Tiles gespiegelt oder gedreht werden

# Default-Eingang festlegen

- WA erfordert einen Layer `start`
- Mindestens eine Kachel muss gesetzt sein
- Auf dieser Kachel starten die Avatare
- Wenn mehrere Kacheln gesetzt sind, wird zufällig eine ausgewählt

# Ausgänge festlegen

- Ähnlich wie beim Eingang: eigener Layer mit mindestens einem gesetzten Tile
- Beim Betreten dieses Tiles wird auf die in der Tile-Layer Eigenschaft `exitUrl` verlinkte Map gewechselt

# floorLayer

- Mit dem (leeren) Object-Layer namens `floorLayer` wird festgelegt, wo WA die Avatare einblendet
- Layer oberhalb von `floorLayer` überdecken den Avatar

# Spezielle Bereiche

- In WA gibt es folgende spezielle Bereiche:
  - Webseite öffnen
    - Eigener Layer mit Eigenschaft `openWebsite` (string)
  - Audio abspielen
    - Eigener Layer mit Eigenschaften
      - `playAudio` (string)
      - `audioLoop` (boolean)
      - `audioVolume`
  - Ruhebereich (bei Begegnungen werden keine Gespräche gestartet)
    - Eigener Layer mit Eigenschaft `silent` (boolean)
  - Videokonferenz
    - Eigener Layer mit Eigenschaften
      - `jitsiRoom` (string)
      - `jitsiTrigger` (string) „onaction“ startet Jitsi nur, wenn man eine Aktion ausführt
      - `jitsiTriggerMessage` (string)

# Scripting

- Karteneigenschaft `script` mit dem Namen des auszuführenden Scripts
- Details:
  - <https://workadventu.re/map-building/scripting>
  - <https://workadventu.re/map-building/api-reference>



# Map testen

- Wenn man seine Map erstellt hat, kann man sie lokal testen

```
dominik@cray:~/myfirstwamap$ npm run start

> workadventure-map-starter-kit@1.0.0 start /home/dominik/myfirstwamap
> webpack serve --open

[wds]: Project is running at http://localhost:8080/
[wds]: webpack output is served from /
[wds]: Content not from webpack is served from .
[wdm]: asset script.js 889 KiB [emitted] (name: main)
runtime modules 432 bytes 3 modules
cacheable modules 336 KiB
  modules by path ./node_modules/webpack-dev-server/ 21.2 KiB
    modules by path ./node_modules/webpack-dev-server/client/ 20.9 KiB 10 modules
    modules by path ./node_modules/webpack-dev-server/node_modules/ 296 bytes 2 modules
  modules by path ./node_modules/html-entities/lib/*.js 61 KiB 5 modules
  modules by path ./node_modules/url/ 37.4 KiB 3 modules
  modules by path ./node_modules/querystring/*.js 4.51 KiB
    ./node_modules/querystring/index.js 127 bytes [built] [code generated]
    ./node_modules/querystring/decode.js 2.34 KiB [built] [code generated]
    ./node_modules/querystring/encode.js 2.04 KiB [built] [code generated]
  modules by path ./node_modules/webpack/hot/*.js 1.42 KiB
    ./node_modules/webpack/hot/emitter.js 75 bytes [built] [code generated]
    ./node_modules/webpack/hot/log.js 1.34 KiB [built] [code generated]
  ./node_modules/webpack/hot/ sync nonrecursive ^\./log$ 170 bytes [built] [code generated]
webpack 5.31.2 compiled successfully in 2382 ms
[wdm]: Compiled successfully.
|
```

# Getting started: Erster Testlauf



# Server selber betreiben

- Am einfachsten mit Ansible
  - devnope hat ein schönes Playbook geschrieben
    - Vielen Dank!
    - Siehe auch:
      - <https://github.com/devnope/ansible-workadventure>
      - <https://media.ccc.de/v/divoc-reboot-to-respawn-2021-39-workadventure-selbst-betreiben-ein-serviervorschlag>
- Zusätzlich brauchen wir noch einen Turnserver
  - -> Fork. Hack. Pull-Request. Upstream: Merge?
  - Turnserver testen mit <https://webrtc.github.io/samples/src/content/peerconnection/trickle-ice/>

# Server installieren

- Voraussetzungen:
  - VM mit Debian 10
  - DNS-Einträge
    - workadventure
    - api.workadventure
    - back.workadventure
    - maps.workadventure
    - play.workadventure
    - pusher.workadventure
    - uploader.workadventure
    - coturn.workadventure
- Login als root

# Installation per Ansible

- `apt-get update`
- `apt-get install -y git-core ansible`
- `git clone https://github.com/chaos-consulting/ansible-workadventure.git`
- `cd ansible-workadventure`
- `nano deployment.yml`
- `cp vault-template.yml vars.vault.yml`
- `ansible-vault encrypt vars.vault.yml`
- `ansible-vault edit vars.vault.yml --ask-vault-pass`
- `ansible-playbook deployment.yml -e @vars.vault.yml --ask-vault-pass`
- `reboot`

# Workadventure starten

- `cd /opt/workadventure/contrib/docker`
- `docker-compose up -d`
- **Logs verfolgen mit:**
  - `docker-compose logs -f`
- **Dauert eine Weile...**
- **Wenn der front\_1 Container irgendwann ...**
  - `[Fri Jun 11 18:08:43.282139 2021] [core:notice] [pid 7:tid 139663110737984] AH00094: Command line: 'apache2 -D FOREGROUND'`
- **... ausspuckt, läuft die Umgebung**
- **Im Browser öffnen: <https://play.workadventure.tolle.domain>**

# Eigene Map integrieren

- **Workadventure stoppen:**
  - `docker-compose down`
- **Im `ansible-workadventure` Verzeichnis `deployment.yml` anpassen:**
  - `workadventure_start_room` auf die eigene Map-URL ändern
- **Ansible-playbook erneut durchlaufen lassen:**
  - `ansible-playbook deployment.yml -e @vars.vault.yml --ask-vault-pass`
- **Workadventure wieder starten:**
  - `cd /opt/workadventure/contrib/docker`
  - `docker-compose up -d`