

# Documentation

## Level Editor

Version: 1.0

Autor: Alexander Bifulco

Datum: 05.12.2018

# Installationsanleitung

## Anforderungen

### NodeJS

Die Anwendung verwendet eine Vielzahl von Tools und Bibliotheken, die über NPM installiert werden müssen. Dafür wird NodeJS benötigt. NodeJS ist ein JavaScript Runtime Environment.

Downloadpfad: <https://nodejs.org/en/>

## Projektabhängigkeiten

Die Anwendung erfordert verschiedene Tools, die installiert werden müssen. Diese müssen über die Windows-Konsole installiert werden. Navigiere dazu zum Projektpfad und führe folgenden Befehl aus:

```
npm install
```

Dies kann je nach Internetverbindung einige Minuten dauern.

## Weitere Konsolenbefehle

- npm run start
  - Startet die Anwendung in einem Browser
- npm run watch
  - Pre-Kompiliert die neuen Anpassungen bei jeder Änderung.
- npm run electron
  - Führt die Windows Applikation auf
- npm run dist
  - Kompiliert die Anwendung für den Live-Betrieb

## Level Editor Steuerung

### Canvas

- Linke Maustaste: Malt die ausgewählte Kachel in die Zelle.
- Rechte Maustaste: Löscht die Zelle.

## Navigationsleiste

- Hamburger: Öffnet die Sprite-Sheet Liste.
- Import: Importiert gespeicherte Levels.
- Save: Speichert alle Informationen der Anwendung in einem Json
- Settings: Können die Canvas Eigenschaften angepasst werden.

## Sprite-Sheet Liste

- Sprite-Sheets können mit dem Plus-Symbol hinzugefügt werden.
- Nachdem ein Sprite-Sheet importiert wurde, kann eine Kachel angeklickt und im Canvas gemalt werden.

## Architektur

### Framework

Für die Anwendung wurde Angular benutzt. Wie Angular funktioniert wird hier ausführlich beschrieben:

<https://angular.io/guide/architecture>

### Aufbau

Die Anwendung besteht aus verschiedenen Komponenten und einem Service.

### Komponente

- App: Bootstrap der Anwendung
- Drawer: Zuständig für den Canvas
- Sidebar: Zuständig für die Sprite-Sheets
- Settings: Zuständig für die Canvas-Eigenschaften
- Routing: Wird in dieser App nicht verwendet
- Dialog: Zuständig für die Sprite-Sheet-Eigenschaften
- Brush: Zuständig für die Zellen der Sprite-Sheets

### Service

Der Service hat alle Informationen der Anwendung. Wenn sich der Service verändert, werden alle Komponenten benachrichtigt.