

SETUP / INSTALLATION

RaspiApp

„RaspiApp“ benötigt

- RaspberryPi, Modell 3B+
- Python 3.5
- Modul „kivy“
- Modul „websockets“

Auf Kommandozeile zum Ordner RaspiApp wechseln.

```
sudo apt-get update
```

```
sudo apt-get install libSDL2-dev libSDL2-image-dev libSDL2-mixer-dev libSDL2-ttf-dev \
```

```
pkg-config libgl1-mesa-dev libgles2-mesa-dev \
```

```
python-setuptools libgstreamer1.0-dev git-core \
```

```
gstreamer1.0-plugins-{bad,base,good,ugly} \
```

```
gstreamer1.0-{omx,alsa} python-dev cython
```

```
sudo pip3 install git+https://github.com/kivy/kivy.git@master
```

Beim ersten Ausführen einer Kivy-App sind die Buttons noch funktionslos, eine Konfigurationsdatei wird erstellt.

Wurde die App als root ausgeführt, liegt diese unter /root/.kivy/config.ini, ansonsten unter /home/<username>/.kivy/config.ini

Der Bereich [Input] dieser Datei muss so aussehen:

```
mouse = mouse mtdev=%(name)s = probesysfs,provider=mtdev hid_%(name)s = probesysfs,  
provider=hidinput
```

Zum Abschluss das Modul „websockets“:

```
python3 -m pip install websockets
```

ExternApp

In dieser Version sind die Verbindungsdaten für den Websocket noch hardcoded und müssen händisch angepasst werden.

Dazu muss in touch_thing_socket.js Client der Wert in Zeile eins entsprechend geändert werden.

RaspiApp gibt diesen aus, wenn der Socket-Server gestartet wird.

Zur Verwendung der (noch im Experimentier-Stadium befindlichen) Programm-Funktion muss die App mittels Node-server gestartet werden, um das Laden der verwendeten Json-Files vom lokalen Datenträger zu ermöglichen.

Auf Kommandozeile:

```
npm install http-server -g
```

ANLEITUNG / BEDIENUNG

RaspiApp

Kommandozeile öffnen, zum Ordner RaspiApp navigieren.

python3 working_touch_thing.py [optional] [optional] [optional] [optional]

Mit den Parametern „debug“, „info“, „warn“ oder „error“ kann der Logging-Level bestimmt werden.

Die Parameter „-kv“ und „-w“ aktivieren den jeweiligen internen Logger der Module „kivy“, bzw. „websockets“.

Der Parameter „-f“ erzeugt zusätzlich Logfiles im Ordner raspiApp/logs

ExternApp

Grundsätzlich genügt es, zum externApp-Ordner zu navigieren und die html-Datei zu öffnen. Die Trainingsprogramm-Funktion in ihrer derzeitigen Ausführung verlangt jedoch das Starten der App über einen Node-Server:

Kommandozeile öffnen und zum Ordner RaspiApp navigieren, dann:

http-server

Im Browser zu localhost:8080 wechseln und dort die html-datei öffnen.

ExternApp muss zu RaspiApp verbunden sein, um mittels Updates die Anzeigen anpassen zu können und Befehle zu übermitteln.

Weiters akzeptiert ExternApp keine Steuerbefehle außer „ON“, solange der Gerätestatus auf „OFF“ steht.