

# MARS (Measurement Activity REST Server)

## Installation

- Clone the git repository
- Run the `install_docker.sh` script if docker and docker-compose isn't installed (tested on clean install of Ubuntu Server 18.04)
- Go to the `src` folder
- Copy `.env.sample` and `.env.db.sample` to `.env` and `.env.db` respectively
- Change values in `.env` and `.env.db` to secure credentials and add the servers IP address to the allowed hosts environment variable.
- Run the following commands

```
sudo docker-compose up -d --build
```

### To build and run the container

```
sudo docker-compose exec web python manage.py migrate --noinput
```

### To migrate the tables to the database

```
sudo docker-compose exec web python manage.py createsuperuser --username admin --email admin@notarealdomain.local --noinput
```

### To create an admin user that can then create more users etc. Default password is set in `.env` under `DJANGO_SUPERUSER_PASSWORD`.

```
sudo docker-compose exec web python manage.py collectstatic --no-input --clear
```

### To collect static files (CSS, Javascript, etc) for Nginx

## Admin

Once deployed, login with the admin user with the default password defined in the `.env` file. Changing this password is important for security, this can be done once logged in by pressing the edit button next to the admin user.

When an admin logs in, they default to `/dashboard/admin` and a user defaults to their user-page. An admin can, from the admin-page, get into each user's individual user page to edit/add information for the user (like API keys, devices, etc).

An admin can from the admin-page add Global API keys that can be used for exporting of data via the API. The read/write permissions are for if the API key is used for ingesting data from devices.

At the bottom of the admin-page, there are basic statistics available with amount of data points, activities and devices that can be filtered by user and time.

As an admin, you can also login to `/admin` which gives the default Django web interface where all tables and their data is available and modifiable if necessary.

## User

A user can add/edit API keys (that only work for data ingestion for their own user id and devices) with read/write permissions (what permission is needed is described in the API documentation). The user can also add devices to their account. When a user has generated an API key and added a device, they can use their user id, device id and API key on their device to start sending data.

## API

Described in API documentation and video.