

# Taskstack Documentation

(Work in progress)

Julius Krahn

September 28, 2020

## Contents

<b>1</b>	<b>Tech Stack</b>	<b>2</b>
1.1	Backend . . . . .	2
1.2	Frontend . . . . .	2
1.3	Misc . . . . .	2
<b>2</b>	<b>Running Taskstack for development</b>	<b>3</b>
2.1	Requirements . . . . .	3
2.2	Run . . . . .	3
<b>3</b>	<b>Deployment</b>	<b>3</b>
3.1	Server Requirements . . . . .	3
3.2	Initial . . . . .	3
3.3	Updating . . . . .	4

# 1 Tech Stack

## 1.1 Backend

- Web Server: NGINX
- Application Server: uWSGI
- Database: MySQL
- Application Framework: (Python:) Flask
  - SQLAlchemy
  - Flask-SocketIO
  - ... (for all used packages see /taskstack/requirements.txt)
- Hosting: AWS EC2 (server), AWS RDS (db), AWS S3 (static and uploaded files)
- Email Sending: AWS SES
- Docker (+ Docker-Compose)

## 1.2 Frontend

- Client and Serverside Rendering
- No Framework/ big library
- socket.io
- tippy.js for Tooltips
- fontawesome for Icons
- he.js for Escaping
- anchorme.js for URL to Link
- cropper.js for Image Cropping
- sortable.js for Drag and Drop + Sort

## 1.3 Misc

- Domain Name: Namecheap

## 2 Running Taskstack for development

### 2.1 Requirements

- Python 3.7
- Python modules: /taskstack/requirements.txt (except uwsgi, gevent, redis)
- Local MySQL database
- AWS CLI
- Port 5000 open

### 2.2 Run

- set environment variables: /taskstack/.env
- run /taskstack/wsgi.py
- access the server: <http://localhost:5000>

## 3 Deployment

### 3.1 Server Requirements

- OS: Ubuntu
- Docker (+ Docker-Compose)
- Git
- Port 80 and 443 open for all incoming and outgoing connections

### 3.2 Initial

- Get the code  
`git clone https://github.com/juliuskrah/taskstack`
- Set environment variables: /taskstack/.env
- Disable THP  
`echo never /sys/kernel/mm/transparent_hugepage/enabled`  
(needs to be done again if the server is shut down)
- Build and Start  
`docker-compose build`  
`docker-compose up -d`
- Enable https  
`docker exec -it nginx bash`  
`sudo certbot --nginx`

### 3.3 Updating

- Stop affected service (taskstack lnginx lredis)  
docker-compose stop [service]
- Get the code  
git pull
- Re-build and start affected service (taskstack lnginx lredis)  
docker-compose build [service]  
docker-compose up -d [service]  
(If you need to rebuild the nginx service, make sure to enable https again!)