
web_blabla
Release 0.1.0

May 03, 2020

Contents:

| | | |
|----------|--|----------|
| 1 | README of Web_blabla | 1 |
| 1.1 | Presentation | 1 |
| 1.2 | Requirements | 1 |
| 1.3 | Getting Started | 2 |
| 1.4 | Dependencies | 2 |
| 1.5 | ToDo | 2 |
| 2 | Notes on web_blabla | 3 |
| 2.1 | Presentation | 3 |
| 3 | Notes on OVH-VPS deployment | 5 |
| 3.1 | OVH Virtual Server | 5 |
| 3.2 | Web_blabla environment variables | 5 |
| 3.3 | Server Preparation | 5 |
| 3.4 | Starting the apps | 6 |
| 3.5 | Access log | 7 |
| 3.6 | DNS Records | 7 |
| 4 | Indices and tables | 9 |

1.1 Presentation

Web_blabla is my boilerplate for web projects.

It tends to be **JAMstack** compliant, i.e. that most of the code is in the front-end and could be delivered by **CDN**. The back-end should be reduced to the minimum, ideally a pure static sever. Real back-end functionalities are delivered via a **REST API**. The front-end should be easily transformed into an **electron** app, **NW** app or **PWA**.

All distributed files must be generated by **transpilation** to help to find bugs early and get proper code. The following languages are used:

- **pug** for HTML
- **Sass** for CSS
- **TypeScript** for JavaScript

HTML5, **CSS3** and **ES6** are targeted even if they might restrict the compatibility with old browsers.

This documentation is also available on **readthedocs**. An instance of this website is accessible at <https://blabla.billet.ovh>

1.2 Requirements

This project has been developed nodejs and npm:

```
> node -v
v12.16.2
> npm -v
6.14.4
```

To upgrade your npm version, you can try:

```
sudo npm i -g npm
```

Additional requirements:

```
sudo apt install rst2pdf evince curl
pip install --user pipenv
```

1.3 Getting Started

In a bash terminal, type:

```
npx degit charlyoleg/web_blabla my_foo_project
cd my_foo_project
npm install
npm upgrade
npm run
npm run intall_py
# rename the project
find . -iname "*web_blabla*"
grep -i web_blabla -r .
# initiate the git repo
git init
git add .
git commit -m 'first commit'
git remote add origin git@github.com:charlyoleg/my_foo_project
git push --set-upstream origin master
git pull
git push
```

1.4 Dependencies

You can check the node-package dependencies with:

```
npm ls --depth=0
npm ls --prod --depth=0
npm ls --dev --depth=0
```

1.5 ToDo

1.5.1 Frontend

- JWT

1.5.2 Backend

- certificate

2.1 Presentation

The goal of *web_blabla* is to provide me a boilerplate for my web projects.

3.1 OVH Virtual Server

OVH proposes price competitive *virtual server* called VPS. It is a complete Ubuntu server, where you can install anything, inclusive nodejs.

3.2 Web_blabla environment variables

This boilerplate `express` application reads several environment variables:

- `NODE_ENV` (it also influences `express` behavior)
- `HTTP_ENABLE` (http is only used for redirecting to https)
- `PORT_NUM`
- `KEY_FILE`
- `CERT_FILE`

3.3 Server Preparation

3.3.1 Add a new user

OVH pre-installs Ubuntu and provides access to the `root` user. As it is not a good practice to run server-application directly as root, we create a new user called `bob`:

```
adduser bob
groups bob
usermod -aG sudo bob
```

To access per ssh this VPS without password, run from your laptop:

```
ssh-copy-id bob@vps12345.ovh.net
ssh bob@vps12345.ovh.net
```

3.3.2 Install PM2

Inspiration from <http://www.drmpop.com/index.php/2016/09/06/installing-and-running-node-js-on-a-vps/>

Install pm2 globally, because systemd will use it as well:

```
sudo npm i -g pm2
```

To restart all pm2 process after the server reboots:

```
pm2 startup # and follow the instructions

systemctl list-units --type service --all | grep pm2
systemctl status pm2-bob
ps aux | grep pm2
pm2 ls
```

On your development laptop, you don't need to install pm2 globally. If you want to practice it, just install it locally:

```
npm i pm2
npx pm2 ls
npx pm2 logs blabla
npx pm2 stop blabla
npx pm2 kill
```

3.3.3 port access right

On Ubuntu, non-root user can not use per default the port-number smaller than 1024. You can grant the privileges with:

```
sudo apt-get install libcap2-bin
which node
ls -l /usr/bin/node
sudo setcap cap_net_bind_service=+ep /usr/bin/node
```

3.4 Starting the apps

In the VPS-terminal:

```
NODE_ENV='production' HTTP_ENABLE=1 pm2 start node backend/dist/web_blabla_app.js --
↪name blablae
pm2 ls
pm2 save
pm2 logs blablae
sudo reboot
pm2 stop blablae
pm2 restart blablae
ps aux | grep pm2
```

To update the *systemd* startup scripts:

```
pm2 unstartup
pm2 startup
```

3.5 Access log

The *web_blabla express* app uses the middleware *morgan* to log the http-requests. The log-files are stored under *backend/log/* and can be analyzed with *goaccess* for example.

3.5.1 Getting started with goaccess

In a bash terminal:

```
sudo apt install goaccess
cd web_blabla
goaccess backend/log/access.log --log-format=COMBINED
goaccess backend/log/access.log --log-format=COMBINED -a -o web_blabla_access_report.
↪html
sensible-browser web_blabla_access_report.html
```

3.6 DNS Records

Use the wildcard '*' to redirect all the non-registered subdomains to a server-IP.

In a bash terminal, test the DNS registry:

```
host mydomain.ovh
host www.mydomain.ovh
host blabla.mydomain.ovh
host abc.mydomain.ovh
host abc.mydomain.ovh
host abc.mydomain.ovh
host abc.mydomain.ovh
host abc.ddd.mydomain.ovh
```


CHAPTER 4

Indices and tables

- `genindex`
- `modindex`
- `search`