# Python Flask Starter Bundle Instructions

## Description

This is readme and instructions how start using flask backend bundle from Akveo. Backend bundle is integrated solution of Flask micro-framework and Angular Frontend code. Backend code plays mostly API role, giving data to the client side as REST API. Backend part of Flask Starter Bundle is based on modern python framework Flask framework. Flask Documentation helps with understanding of Flask principles.

# Running Instructions

Application requires a database to correctly run. It can be any database SQLAlchemy supports. In development by default Sqlite database is used, which is stored at `api/bundle.db`. If you would like to use other database in development mode, change database URL in `api/config.py` line 46.

### Running the backend

0) Install python 3. You can find here instructions for your operation system.

1) Change directory to the *backend* folder `cd backend`

2) Create virtualenv `virtualenv -p python3 bundle_env`

3) Activate virtualenv `source bundle_env/bin/activate`

4) Install required python modules `pip install -r requirements.txt`

5) Only during the initial launch create database schema for your application `python manage.py recreate_db`

6) Run the application `python manage.py runserver`

That's it! Now your application is running at port 3001 and you can access it by typing `http://localhost:3001/` in your browser.

### Running Angular front end

Before running Angular you need to have node installed

0) Install nodejs. You can download it using you operating system package manager or from here.

1) Go to the *front end* folder `cd frontend`

2) Run commands `npm install` and `npm start` 3) Open `http://localhost:4200` in your browser 4) create new user using interface and start working with app

# Issue tracking

You can post issues and see sample materials at [GitHub Support Repository]

### Other Commands

```
# development
$ npm run start
# development with watch mode - code will be rebuild after each change. it runs `nodemon` module to watch over changes and re-run node api automatically.
$ npm run start:dev
# build dist for prod deployment
$ npm run build
# production mode
$ npm run start:prod
```

# Test

```
# unit tests
$ npm run test
# e2e tests
$ npm run test:e2e
# test coverage
$ npm run test:cov
```

# Style Check

```
# lint
$ npm run lint
```

# Features

- Flask framework, feature based modules
- Compatible with ngx-admin out of the box
- JWT authentication using flask-jwt-extended module
- PostgreSQL, MySQL, Oracle, Microsoft SQL Server, and SQLite can be used as databases
- SQLAlchemy is used as database toolkit for data CRUD operations

- Compression setup for API
- Logging to constole and files
- werkzeug is used for better experience while develop
- 6 months free updates

## Basic Code Structure

Code is organized in following structure - Main Folder - frontend // this floder contains all UI code - backend // server side Flask code - api // this is where all project files located - auth // auth module and code there - user // user module and code there - __init__.py // this is where flask app is created - config.py // configuration reading code is here - core.py // general module independent code - db.py // database initialization - email.py // code that sends emails - email.py // application runner

```
- docs // documentation and licenses
```

## Support

Please post issues in Bundle Support Repository Or send email to Akveo Support

## License

Flask bundle is provided under Single Application or Multi Application license. Please find details in License files.