

Dokumentacja

Aplikacja HelpDesk
FlaskApp
v1.5

16 wrzesień 2017
Albert Szadziński

Spis treści

1	Instalacja	3
1.1	Wersje oprogramowania	3
1.2	Konfiguracja środowiska do wirtualizacji	3
1.3	Instalacja pakietów	4
1.4	Konfiguracja serwera	4
1.4.1	Tworzenie katalogu na aplikacje	4
1.4.2	Pobieranie kodu źródłowego aplikacji wraz z szablonami	4
1.4.3	Tworzenie wirtualnego środowiska dla webaplikacji	4
1.4.4	Instalacja lokalnych modułów dla Pythona i test aplikacji	5
1.4.5	Konfiguracja apache2	5
2	Konfiguracja aplikacji	7
2.1	Pliki konfiguracyjne aplikacji	7
2.1.1	Zarządzanie działami i typami zleceń	7
2.1.2	Podpinanie bazy danych pod aplikacje	7
2.2	Zarządzanie bazą danych	7
2.2.1	Zarządzanie użytkownikami	7
2.2.2	Wiadomości i aktualności	7

1 Instalacja

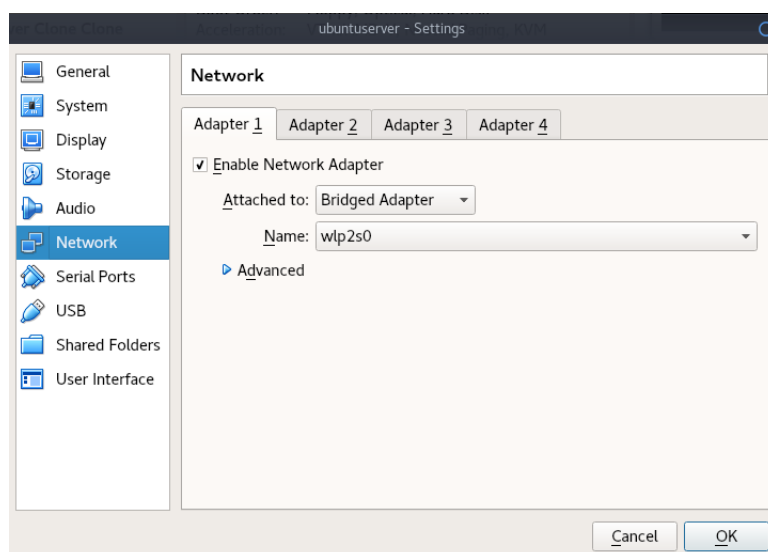
1.1 Wersje oprogramowania

- **System operacyjny**
Ubuntu 16.04 xenial - wersja serwerowa
Linux ubuntu 4.4.0-87-generic
- **apache2**
Server version: Apache/2.4.18 (Ubuntu)
Server built: 2017-07-27T14:34:01
- **MySQL**
mysql 14.14 Distrib 5.7.19
- **pip**
pip 8.1.1 from /usr/lib/python2.7/dist-packages (python 2.7)
- **Flask**
Flask 0.12.2
Python 2.7.12 (default, Nov 19 2016, 06:48:10)

1.2 Konfiguracja środowiska do wirtualizacji

Po instalacji aktualnej wersji **VirtualBoxa** należy przejść do *File>Import* w przypadku posiadania obrazu systemu **.ova** lub do *File>New* aby rozpocząć instalację nowego systemu z obrazu **.iso**. W obu przypadkach należy zatwierdzić domyślne konfiguracje.

Przed uruchomieniem maszyny wirtualnej należy upewnić się, czy włączone jest mostkowanie karty sieciowej na aktualnie używanym interfejsie sieciowym.



1.3 Instalacja pakietów

W przypadku importowania maszyny z pliku .ova wszystkie niezbędne pakiety są zainstalowane, a poniższe instrukcje należy zastosować w momencie wystąpienia problemów w działaniu aplikacji od punktu (1.4.3) po wcześniejszym wykonaniu:

```
$: rm -rf /var/www/FlaskApp/FlaskApp/venv #jako root
```

Po uzyskaniu dostępu do VM (maszyna wirtualna) poprzez VBox lub SSH należy przejść na użytkownika *root*:

```
$: sudo su #lub
```

```
$: su
```

Domyślne hasła:

- server:0e48e4Jmmr
- root:4oo1Jmmr

Aktualizacja listy oraz pakietów:

```
$: apt-get update
```

```
$: apt-get upgrade
```

Instalacja niezbędnych pakietów:

```
$: apt-get install apache2 mysql-client mysql-server libapache2-mod-wsgi  
libmysqlclient-dev python-dev python-pip phpmyadmin  
texlive texlive-lang-polish texlive-fonts-extra python-mysqldb
```

1.4 Konfiguracja serwera

1.4.1 Tworzenie katalogu na aplikacje

```
$: mkdir /var/www/FlaskApp
```

1.4.2 Pobieranie kodu źródłowego aplikacji wraz z szablonami

```
$: cd /var/www/FlaskApp
```

```
$: git clone https://github.com/aszadzinski/help-desk-panel
```

```
$: mv help-desk-panel/ FlaskApp/
```

1.4.3 Tworzenie wirtualnego środowiska dla webaplikacji

```
$: pip install virtualenv
```

```
$: cd /var/www/FlaskApp/FlaskApp
```

```
$: virtualenv venv
```

1.4.4 Instalacja lokalnych modułów dla Pythona i test aplikacji

```
$: source venv/bin/activate
(venv)$: pip install flask
(venv)$: pip install mysql-python
(venv)$: pip install mysql
(venv)$: python __init__.py
```

Po wykonaniu ostatniego polecenia w terminalu powinny pojawić się informacje o rozpoczęciu nasłuchiwania na localhostie:

```
(venv) root@ubuntu:/var/www/FlaskApp/FlaskApp# python __init__.py
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
* Restarting with stat
* Debugger is active!
* Debugger PIN: 717-628-278
```

po otrzymaniu takiej odpowiedzi należy je anulować CTRL+C oraz wyjść z virtualenv:

```
(venv)$: deactivate
```

W przypadku otrzymania błędu należy doinstalować brakujące moduły widoczne w komunikacji błędu jako np:

```
Line 12: import MySQLdb
module not found
```

poprzez wpisanie w środowisku wirtualnym:

```
$: pip install <modul>
```

lub przez manager pakietów:

```
$: apt-get install python-<modul>
```

1.4.5 Konfiguracja apache2

Po uzyskaniu poprawnego działania na localhostie należy dodać aplikację FlaskApp do stron apache'a oraz uruchomić mod *wsgi*.

Aktywacja wsgi:

```
$: a2enmod wsgi
$: service apache2 restart
$: touch /var/www/FlaskApp/flaskapp.wsgi
```

do pliku flaskapp.wsgi wkleić należy:

```
import sys
import logging
logging.basicConfig(stream=sys.stderr)
sys.path.insert(0, "/var/www/FlaskApp/")
from FlaskApp import app as application
application.secret_key = 'foobarspameggs'
```

Następnie po stowrzeniu pliku:

```
$: touch /etc/apache2/sites-available/FlaskApp.conf
```

z zawartością:

```
<VirtualHost *:80>
    ServerName <aktualny adres IP VM>
    ServerAdmin jakis@mail.com
    WSGIScriptAlias / /var/www/FlaskApp/flaskapp.wsgi
    <Directory /var/www/FlaskApp/FlaskApp/>
        Order allow,deny
        Allow from all
    </Directory>
    Alias /static /var/www/FlaskApp/FlaskApp/static
    <Directory /var/www/FlaskApp/FlaskApp/static/>
        Order allow,deny
        Allow from all
    </Directory>
    ErrorLog ${APACHE_LOG_DIR}/error.log
    LogLevel warn
    CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>
```

należy dodać aplikację do aktywnych:

```
$: a2ensite FlaskApp
```

```
$: service apache2 reload
```

2 Konfiguracja aplikacji

2.1 Pliki konfiguracyjne aplikacji

2.1.1 Zarządzanie działami i typami zleceń

W katalogu aplikacji `/var/www/FlaskApp/FlaskApp` w folderze `config_files` znajdują się pliki konfiguracyjne odpowiedzialne za:

- Działy (*branches.data*):
Każda linia tego pliku interpretowana jest przez program jako nowy dział
- Typy zleceń (*typy_zleceń.config*)
jak wyżej
- Incydenty (*incydenty.config*)
Linie rozpoczynające się od `#` oraz od znaku nowej linii są ignorowane.
Linie rozpoczynające się od spacji traktowane są jako opcja wyboru do najbliższej linii nad nią nie rozpoczynającej się od spacji

2.1.2 Podpinanie bazy danych pod aplikację

Pośrednikiem między aplikacją FlaskApp a bazą MySQL jest skrypt `database_connect.py` do którego odwołuje się skrypt `db_func.py` z zapytaniami sql. Aby zmienić aktualną bazę należy zmodyfikować parametry metody `MySQLdb.connect` w pliku `database_connect.py`:

```
conn = MySQLdb.connect(host="localhost", user="root",  
                        passwd="3y0q5rZkn", db="helpdesk")
```

2.2 Zarządzanie bazą danych

Domyślne hasło bazy MySQL na pliku `.ova`:
root: 3y0q5rZkn

2.2.1 Zarządzanie użytkownikami

Aplikacja umożliwia dostęp do bazy użytkowników z poziomu panelu admina po zalogowaniu lub przez phpmyadmin (tabela `users`).

Domyślne hasło admina:
admin: changeme

Pole "Uprawnienia" jest istotne dla programu tylko w momencie gdy przyjmuje wartość `admin`, inne wartości są ignorowane i przydzielają użytkownika do grupy bez uprawnień.

2.2.2 Wiadomości i aktualności(tabele `news` i `notifications`)

W przypadku posiadania przez użytkownika uprawnień admina może od publikować wiadomości z zakładki aktualności widoczne dla każdego użytkownika oraz zarządzać przychodzącymi

zleceniami. Każdy admin w zakładce Lista zleceń ma dostęp do edycji, zamrażania(ukrycie zlecenia, widoczne tylko w zakładce 'zamrożone') oraz delegowania do innych użytkowników z uprawnieniami admina (widoczne w zakładce 'nowe(do mnie)').

W momencie wpisania dowolnej treści w pole 'data zakończenia' zlecenie przenoszone jest do zakładki 'zakończone'