

Publicare un progetto Django su pythonanywhere.com

I passaggi necessari ad ottenere il risultato voluto, sono i seguenti:

1. carica il tuo codice su PythonAnywhere
2. imposta un virtualenv e installa Django e tutti gli altri requisiti
3. configura la tua app web usando l'opzione **manual config**

1. Carica il tuo codice su PythonAnywhere

La soluzione più immediata sarebbe quella di clonare il repository presente su GitLab o GitHub direttamente all'interno della propria virtual machine, ma avendo registrato un account gratuito, pythonanywhere permette connessioni https ed ssh solamente a siti o servizi presenti in una loro **whitelist**, che non include il server **dw.gnet.it**.

Perciò l'unica alternativa è caricare manualmente il progetto django compresso. E' consigliabile utilizzare il formato **.tar.gz** in modo da poterlo comprimere direttamente dalla **bash** della vm con il comando:

```
tar -xvzf cruscanalyzer_base_folder.tar.gz.
```

2. Imposta un virtualenv e installa Django e tutti gli altri requisiti

Fatto questo, è necessario creare un virtual environment sulla macchina per poter installare la versione più recente di Django, in particolare:

```
mkvirtualenv --python=/usr/bin/python3.6 cruscanalyzer-venv
```

3. Configura la tua app web usando l'opzione manual config

Seguendo la procedura suggerita dal sito, andare sul tab **Web** del sito, crea una nuova applicazione e impostare la **configurazione manuale (manual config)**.

Inserisci il nome del tuo virtualenv

Una volta fatto, nella sezione **Virtualenv**, inserisci il path del tuo virtualenv creato, per esempio:

```
/home/cruscanalyzer/.virtualenvs/cruscanalyzer-venv
```

Opzionale: inserisci il path alla cartella del codice sorgente

Sopra alla sezione **Virtualenv**, nella sezione **Code**, puoi inserire il path alla cartella del progetto Django per avere un link veloce ai file contenuti in essa, il path sarà il seguente:

```
/home/cruscanalyzer/cruscanalyzer_base_folder
```

Modifica il file WSGI

Clicca sul link al file WSGI, nella sezione **Code**, per modificare le impostazioni e far funzionare l'applicazione Django. Il link è il seguente:

https://www.pythonanywhere.com/user/cruscanalyzer/files/var/www/cruscanalyzer/pythonanywhere_com_wsgi.py?edit

In esso, cancella **tutto** il contenuto ed inserisci **solamente** le seguenti righe di codice:

```
# ++++++ DJANGO ++++++
# To use your own Django app use code like this:
import os
import sys

# assuming your Django settings file is at
'/home/myusername/mysite/mysite/settings.py'
```

```

path = '/home/cruscanalyzer/cruscanalyzer_base_folder'
if path not in sys.path:
    sys.path.insert(0, path)

os.environ['DJANGO_SETTINGS_MODULE'] = 'cruscanalyzer.settings'

## Uncomment the lines below depending on your Django version
##### then, for Django >=1.5:
from django.core.wsgi import get_wsgi_application
application = get_wsgi_application()
##### or, for older Django <=1.4
#import django.core.handlers.wsgi
#application = django.core.handlers.wsgi.WSGIHandler()

```

Successivamente, **salva** il file e premi il tasto **Reload** all'interno del tab Web della tua vm.

NOTA BENE:

Per far funzionare il nuovo applicativo, è importante inserire nel file **settings.py** del progetto Django la URL del servizio cloud nel nostro progetto nella riga **ALLOWED_HOST**:

```
ALLOWED_HOSTS = ['cruscanalyzer.pythonanywhere.com']
```

Dopo di che sarà necessario fare nuovamente il **Reload** della Web App.

Visualizzare il database

Per poter visualizzare lo schema e il contenuto del database di django (in automatico viene creato un database SQLite, ma è possibile cambiarlo), è necessario seguire i seguenti passi:

- Cliccare sulla tab **database** in alto a destra di Pycharm; si aprirà una finestra in cui sarà possibile visualizzare i database presenti nel progetto.

- Premere **MAIUSC+INVIO**, o cliccare sull'icona del database con i settings *Data Source Properties*.
- Dovrebbe essere già presente in automatico il database db, collegato al file db.sqlite3. Nel caso non fosse così, aggiungere questo file.
- Aggiornare i driver se necessario (messaggio a fondo finestra).
- In *comment*, inserire il percorso per raggiungere il file **settings.py** del progetto.
- Applicare i cambiamenti e selezionare ok.