

# Making Profile Website Like an Architect

I'm  
Bima Adhi  
Pratama  
Kharis

Seorang Villain dari Masa Antah Berantah. Bima as Mr. Bipo The SecretDiscorder

Bima Adhi Pratama Kharis

Hello, This is my documentation about my life

Goto To Profiles OR My Apps →

Profile

Wren

Python logo

PBSSI JAYA TEKNIK

1st Practical

BIMALOG V3

Performing system checks...

runserver -l 8000

Semoga Beruntung

Note: This is offline but requires network for all access.

http://127.0.0.1:8000

Semoga Beruntung

If you encounter any other issues or want to discuss features:

Note: This is offline but requires network for all access.

Full Changelog: [bimalogv2...bimalog3](#)

Full Changelog: [bimalogv3...bimalog3](#)

Assets 3

dist.zip 323 MB

Source code (tar.gz)

Project

PROJ

27 minutes ago

45 minutes ago

45 minutes ago

Join discussion

screenrec

## **Daftar Isi**

1. Pengenalan Praktik Pemrograman
  2. Instalasi Bahasa Pemrograman Python
  3. Framework Django
  4. Install Module Python
  5. Membuat Projek Website Pertama
  6. Deploy dan Akses Domain Online Projek
  7. Penutup
- 

Saya Bima Adhi Pratama Kharis telah menyelesaikan buku praktik pemrograman website saintis. hasil projek dapat dilihat pada <https://bimakhzdev.my.id>. Projek yang didasari rasa tahu dan menambah pengalaman mengenai pemrograman website. Bagaimana agar peralatan pendidikan dapat menjadi satu kesatuan dan opensource. Saya inovator dan kreator dari Website Profile ini akan membawa anda pada praktik terpraktis yang pernah ada.

Terima kasih pada Allah SWT, atas karunia, anugrah dan berkahnya projek ini dapat terselesaikan. Terimakasih atas kesabaran rekan rekan apabila ada kesalahan baik disengaja maupun tidak disengaja, saya minta maaf sebesar-besarnya.

penulis Bima Adhi



## Pengenalan Praktik Pemrograman

Pemrograman, terlebih lagi Website sudah ada dari awal tahun 2000-an. Sejak ditemukannya World Wide Web (www) dan protokol TCP/HTTP/HTTPS, perkembangannya sangat pesat. Banyak kemudian mengembangkan software dan hardware untuk mendukung alat globalisasi ini. Muncul bahasa bahasa saintis komputer, seperti halnya yang akan kita bahas yaitu python. Untuk membuat Website sangatlah mudah dan semua bisa. Hanya diperlukan webbrowser, bahasa program dan text editor anda sudah bisa membuat website anda sendiri.

Terdapat HTML (HyperTextMarkupLanguage) , CSS(Cascading Style Sheet), dan Javascript / Python sebagai server side scripting.

*Hypertext Markup Language* yang merupakan sebuah bahasa *markup*. HTML adalah kode untuk membuat struktur halaman suatu *website* yang menarik, saling terhubung satu dengan yang lainnya, dan yang pasti dapat diakses melalui internet.

Awalnya html ditemukan oleh Tim Berners-Lee pada tahun 1991. HTML adalah solusi untuk membantu ilmuwan dalam mengakses dokumen, namun kini html semakin berkembang pesat di dunia pemrograman web.

Seperti halnya membuat sebuah text bahasa, kita memerlukan struktur, dan html adalah jawaban dari itu. gaya struktur dan desain dari text itu jawabannya adalah CSS.

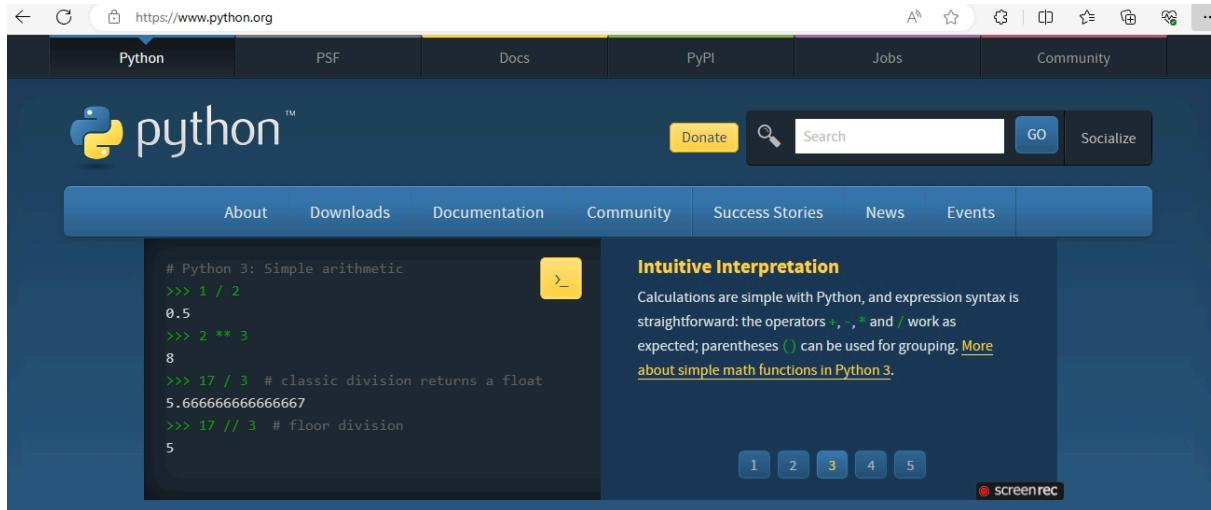
Kita fokuskan ke python. Python (Guido Van Rossum) merupakan bahasa interpreter (dapat dijalankan setelah ditulis) merupakan turunan dari bahasa C (machine lang). Python booming dikalangan programmer memasuki tahun 2020. Bahasa yang menempati peringkat teratas pada waktu itu, memiliki aturan penulisan yang sangat ringkas dan mudah diingat.

Nah kita berkaca lagi pada projek profile website kita. Untuk membangun website terintegrasi dengan Python atau scripting lang, kita memerlukan kerangka pemersatu, (framework). Python sendiri memiliki perpustakaan kode PIP. Python Installer Package yang memuat Kode yang sudah ada dan di tautkan kembali ke kode kita, kalau dalam membuat buku dinamakan Daftar Pustaka.

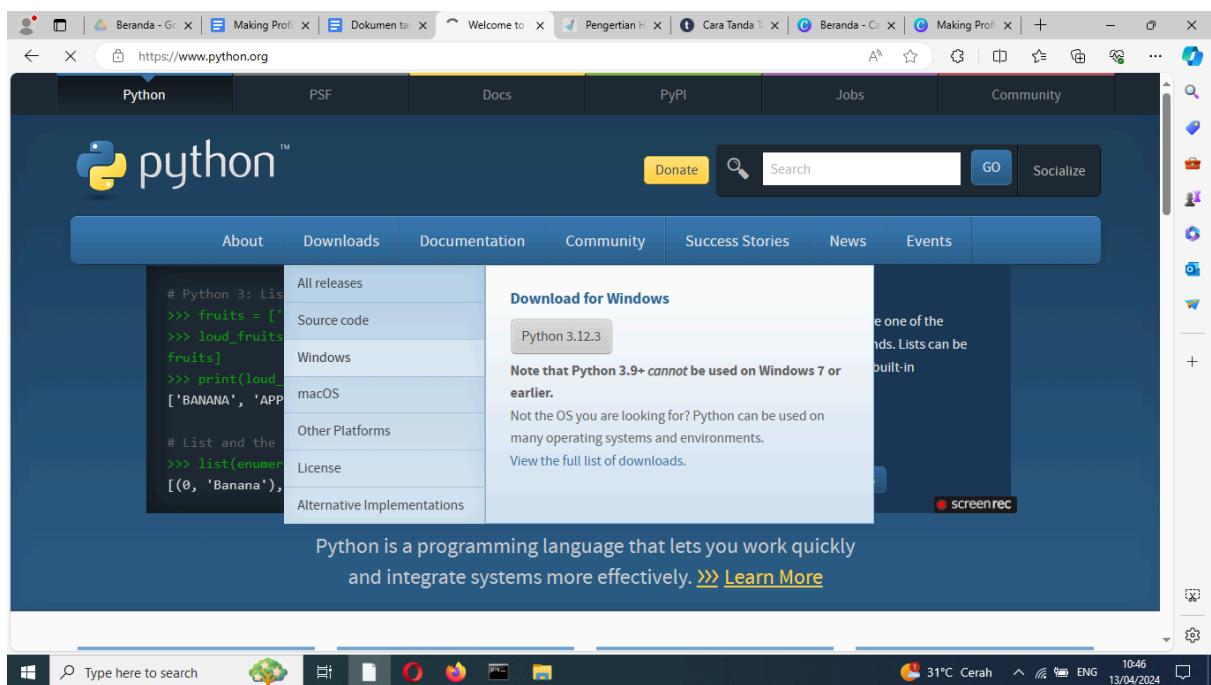
Website PIP <https://pypi.org/project/pip/> sendiri menyimpan kode yang telah teregistrasi, dan pip juga termasuk ke perpustakaan itu sendiri. Python memasuki versi 3.12.3 sudah otomatis menginstall pip dan module dasar. Mendukung Windows, Android, MAC OS, dan Kernel Linux dengan multi talent, meski performanya lebih lambat dari C / C++, namun tidak ada salahnya belajar Python.

## Penginstalan Bahasa Python

Saya disini mencontohkan di Windows 10. Silahkan pergi ke website Python python.org



## Pilih Menu Download Windows



# Python Releases for Windows

- [Latest Python 3 Release - Python 3.12.3](#)

## Stable Releases

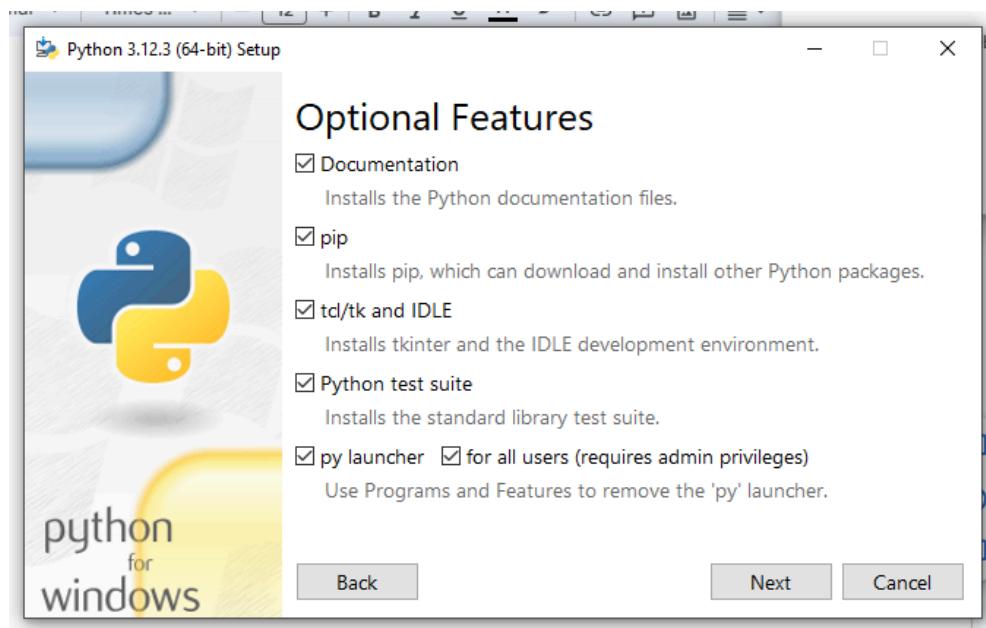
- [Python 3.12.3 - April 9, 2024](#)

**Note that Python 3.12.3 cannot be used on Windows 7 or earlier.**

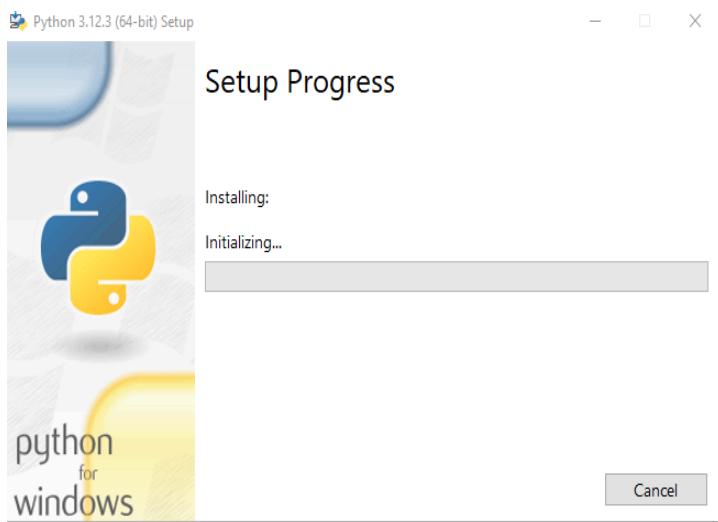
- Download [Windows installer \(64-bit\)](#)
- Download [Windows installer \(ARM64\)](#)
- Download [Windows embeddable package \(64-bit\)](#)
- Download [Windows embeddable package \(32-bit\)](#)
- Download [Windows embeddable package \(ARM64\)](#)
- Download [Windows installer \(32-bit\)](#)

Pilih versi paling baru Latest Version pilih opsi teratas , pastikan Windows mendukung 64-bit dan amd64 / amd32.

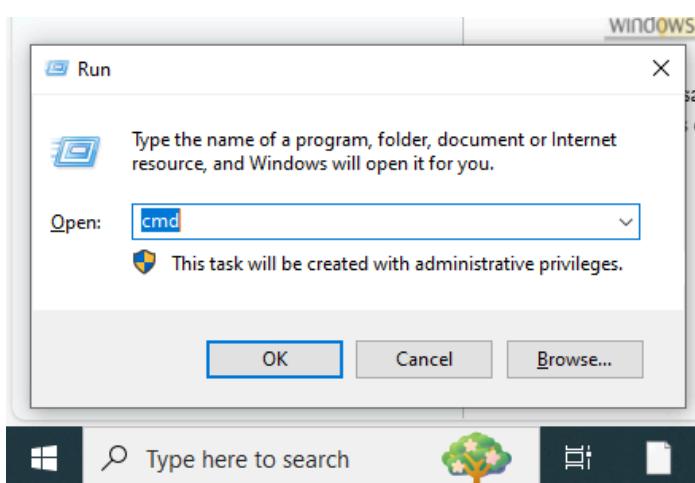
Setelah Download sukses Tekan Installer nya dan pilih instalasi sebagai berikut



(Pastikan Checklist add python to ENV Path (environment variabel). Dikarenakan saya sudah pernah install, jadi cukup upgrade Versi



Tunggu sampai selesai. Pastikan tidak ada kendala gagal atau lain. Tekan Keyboard Logo Windows dan R secara bersamaan



ketik cmd atau cmd.exe lalu OK. ketik di Command Prompt tersebut

python –version

```
Administrator: Command Prompt
Microsoft Windows [Version 10.0.19045.4291]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Administrator>python --version
Python 3.12.3

C:\Users\Administrator>
```

A screenshot of a Windows Command Prompt window with "Administrator" privileges. The title bar says "Administrator: Command Prompt". The window shows the command "python --version" being typed and the output "Python 3.12.3" being returned. The prompt then changes to "C:\Users\Administrator>".

## Instalasi Module Django

Instalasi berhasil. Untuk membuat website di python, kita perlu module perpustakaan yang memuat fungsi itu. ketik django <https://www.djangoproject.com/> . Pergi ke Download pada Website Official tersebut

The screenshot shows the official Django website at <https://www.djangoproject.com/>. The top navigation bar includes links for OVERVIEW, DOWNLOAD, DOCUMENTATION, NEWS, COMMUNITY, CODE, ISSUES, ABOUT, and DONATE. The main content area is titled "Download" and features a section titled "How to get Django". It explains that Django is available open-source under the [BSD license](#) and recommends using the latest version of Python 3. It lists supported Python versions and provides instructions for getting the latest official version. Below this, there's a "Option 1: Get the latest official version" section with installation instructions for Linux/macOS and Windows, including a command-line example for Windows: "python -m pip install Django==5.0.4". To the right, there's a sidebar titled "Diamond and Platinum Members" featuring logos and descriptions for DEFNA and JetBrains PyCharm.

Untuk menginstall Django di windows lebih mudahnya ketik di command prompt (cmd)

*pip install django*

A screenshot of a Windows Command Prompt window titled "Administrator: Command Prompt". The window shows the following command and its output:

```
Administrator: Command Prompt
Microsoft Windows [Version 10.0.19045.4291]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Administrator>python --version
Python 3.12.3

C:\Users\Administrator>pip install django
Requirement already satisfied: django in c:\program files\python312\lib\site-packages (5.0.4)
Requirement already satisfied: asgiref<4,>=3.7.0 in c:\program files\python312\lib\site-packages (from django) (3.7.2)
Requirement already satisfied: sqlparse>=0.3.1 in c:\program files\python312\lib\site-packages (from django) (0.4.4)
Requirement already satisfied: tzdata in c:\program files\python312\lib\site-packages (from django) (2023.4)

C:\Users\Administrator>
```

Untuk membuat projek website dengan django, ketik lagi

*mkdir bimalog*

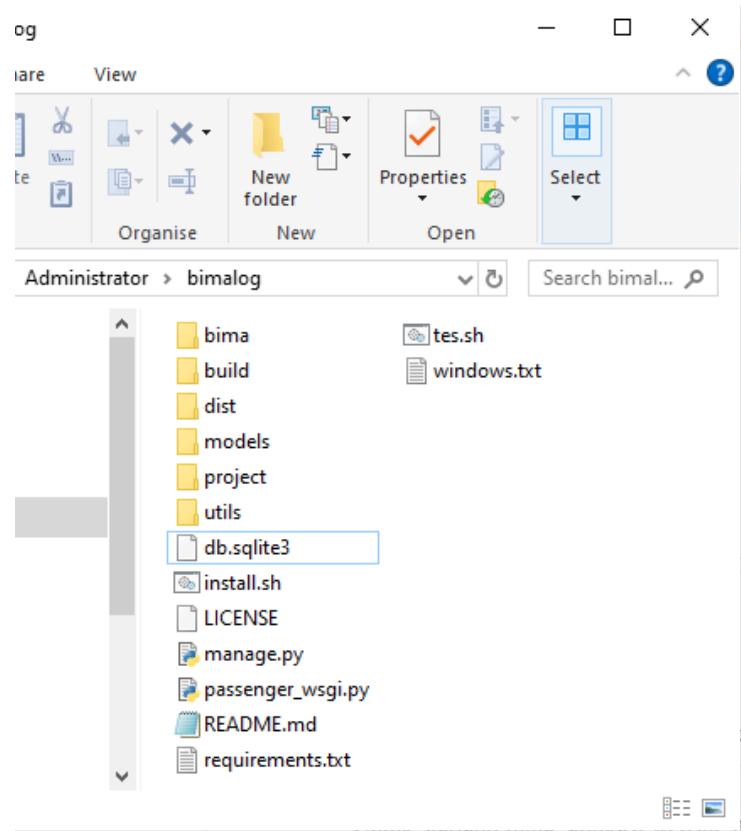
*cd bimalog*

*django-admin startproject project*

*django-admin startapp bima*

Kita akan membuat direktori bimalog lalu menuju direktori folder bimalog. Dengan startproject adalah direktori folder project django dan pengaturan untuk website.

Untuk startapp bima, aplikasi website dengan direktori folder bima. Struktur jika dilihat pada File Explorer (Windows + E)



pergi ke project lalu tekan settings.py Pastikan sudah menginstall Notepad++

```
# SECURITY WARNING: don't run with debug turned on in production!
DEBUG = True

ALLOWED_HOSTS = ['*', 'bimakharizdev.my.id']
CSRF_TRUSTED_ORIGINS = ["https://bimakharizdev.my.id", "http://*"]

# Application definition

INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    'bima',
]

MIDDLEWARE = [
    'django.middleware.security.SecurityMiddleware',
    'django.contrib.sessions.middleware.SessionMiddleware',
    'django.middleware.common.CommonMiddleware',
    'django.middleware.csrf.CsrfViewMiddleware',
    'django.contrib.auth.middleware.AuthenticationMiddleware',
    'django.contrib.messages.middleware.MessageMiddleware',
    'django.middleware.clickjacking.XFrameOptionsMiddleware',
]

ROOT_URLCONF = 'project.urls'
```

```
ALLOWED_HOSTS = ['*']
```

default untuk mengizinkan domain lokal kita, agar project website bimalog bisa diakses di browser. tambahkan nama aplikasi kita (bima) pada INSTALLED\_APPS = [...‘<namaapps>’]

Lalu mengizinkan templates file html pada line

---

```
import os
```

```
TEMPLATES = [
```

```
{
```

```
'BACKEND': 'django.template.backends.django.DjangoTemplates',
```

```
'DIRS': [os.path.join(BASE_DIR, 'bima', 'templates')],
```

```
'APP_DIRS': True,
```

```
'OPTIONS': {
```

```
'context_processors': [
```

```
'django.template.context_processors.debug',
```

```
'django.template.context_processors.request',
```

```
'django.contrib.auth.context_processors.auth',
```

```
'django.contrib.messages.context_processors.messages',
```

```
],
```

```
},
```

```
},
```

```
]
```

templates berada pada direktori aplikasi bima folder baru namanya templates akan diisi file html. 'DIRS': [os.path.join(BASE\_DIR, 'bima', 'templates')],

---

Mengizinkan File Gambar selain html perlu diedit settings.py pada line

---

```
MEDIA_ROOT = os.path.join(BASE_DIR, 'bima', 'media')

MEDIA_URL = '/media/'

# Static files (CSS, JavaScript, Images)

# https://docs.djangoproject.com/en/5.0/howto/static-files/

STATICFILES_DIRS = (
    os.path.join(BASE_DIR, 'bima', 'static'),
)

STATIC_ROOT = os.path.join(os.path.dirname(BASE_DIR), 'bima', 'static')

STATIC_URL = '/static/'
```

---

terdapat static dan media sebagai path dari URL website kita.

Tekan Ctrl+S atau save file settings. kembali ke direktori project, sekarang buka file urls.py

---

```
from django.contrib import admin

from django.urls import path, include, re_path

# urls.py

from django.conf import settings

from django.conf.urls.static import static

from django.views.static import serve
```

```

urlpatterns = [
    path('admin/', admin.site.urls),
    path("", include('bima.urls'), name=""),
    re_path(r'^media/(?P<path>.*$', serve, {'document_root': settings.MEDIA_ROOT}),
    re_path(r'^static/(?P<path>.*$', serve, {'document_root': settings.STATIC_ROOT}),
]

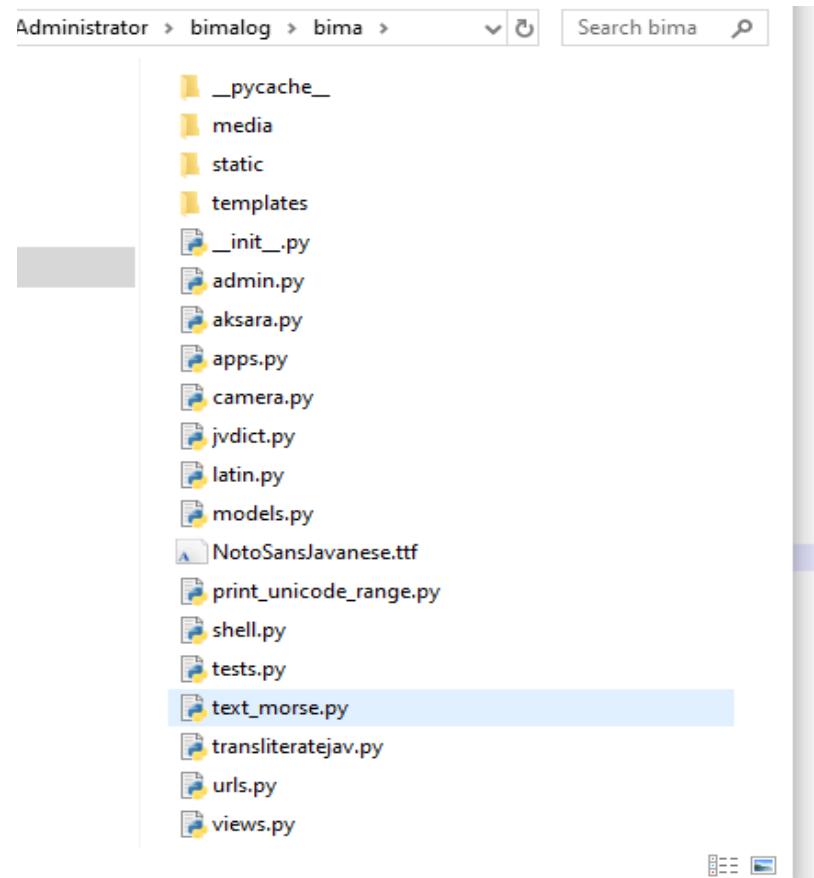
if settings.DEBUG:
    urlpatterns += static(settings.STATIC_URL, document_root=settings.STATIC_ROOT)

```

---

urlpattern perlu ditambah urls.py di folder aplikasi bima dengan cara import include dan re\_path formatting seperti di atas.

project/urls.py Ctrl+S Save File Copy File urls.py ke folder bima lalu pastekan.



buka urls.py lalu edit seperti pada kode di bawah

---

```
from django.contrib import admin

from django.urls import path, include

from . import views

urlpatterns = [
    path('spldtv/', views.solve_linear_system, name='solve_linear_system'),
    path('polino/', views.polino, name='polinomial'),
    path('algebra/', views.algebra, name='algebra'),
    path('monitorb/', views.monitorbelakang, name='monitorbelakang'),
    path('monitor/', views.monitor, name='monitor'),
    path('quran/', views.quran, name='quran'),
    path('feed/', views.webcam_feed, name='webcam_feed'),
    path('process_image/', views.process_image, name='process_image'),
    path("", views.personal, name=""),
    path('project/', views.project, name="project"),
    path('profile/', views.profile, name="profile"),
    path('base/', views.base, name="base"),
    path('satuan/', views.satuan, name="satuan"),
    path('blang/', views.index, name="blang"),
    path('kalkulator/', views.kalkulator, name="kalkulator"),
    path('clock/',views.server_time,name="clock"),
```

```
path('youtube/',views.youtube,name="youtube"),  
path('morse/',views.morse,name="morse"),  
path('prima/',views.prima,name="prima"),  
path('translator/',views.translator,name="translator"),  
path('proibli/', views.proibli, name='proibli'),  
path('translate/', views.translate, name='translate'),  
path('translate/translate_ori/', views.search_original_coordinates, name='translate_ori'),  
path('translate/translate_end/', views.translated_search, name='translate_end'),  
path('aksara_converter/', views.aksara_converter, name='aksara_converter'),  
path('aksara_converter/aksara_converter_image/', views.convert_image,  
name='convert_image'),  
]  
-----
```

Sebenarnya path() adalah letak kode kita pada browser sebagai contoh  
<http://127.0.0.1:8000/<letak kode>/>

dengan letak kode = 'polino/' di views.polino templates polino.html..

Nah agak bingung kah?? Penasaran?

Lanjut bima/urls.py Ctrl+S Save lalu kita buat file polino.html di folder templates/

Isinya sebagai berikut untuk fitur polinomial

```
<!-- polino.html -->  
-----
```

```
<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Polynomial Operations</title>

<link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">

<style>

.polynomial-input {

    height: 100px;

    resize: none;

}

</style>

</head>

<body>

<div class="container mt-5">

    <h1 class="mb-4">Polynomial Operations</h1>

    <form method="post">

        {% csrf_token %}

        <div class="form-group">

            <label for="polynomial1">Polynomial 1:</label>

            <input type="text" class="form-control polynomial-input" id="polynomial1" name="polynomial1" placeholder="Enter polynomial 1">

        </div>

    </form>

</div>


```

```
</div>

<div class="form-group">

    <label for="polynomial2">Polynomial 2:</label>

    <input type="text" class="form-control polynomial-input" id="polynomial2"
name="polynomial2" placeholder="Enter polynomial 2">

</div>

<div class="form-group">

    <label for="operation">Select operation:</label>

    <select class="form-control" id="operation" name="operation">

        <option value="addition">Addition</option>

        <option value="multiplication">Multiplication</option>

        <option value="division">Division</option>

        <option value="roots">Roots</option>

    </select>

</div>

<button type="submit" class="btn btn-primary">Calculate</button>

</form>
```

```
{% if result %}

<div class="mt-5">

    <h2>Result:</h2>

    <p>{{ result | safe }}</p>

</div>
```

```

{%- endif %}

{%- if result and operation == 'roots' %}

<div class="mt-5">

    <h2>Polynomial Plot:</h2>

</div>

{%- endif %}

</div>

<script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"></script>

<script
src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.5.4/dist/umd/popper.min.js"></script>

<script
src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>

</body>

</html>

```

---

polino.html save. kemudian kembali ke direktori bima, terdapat views.py kita edit file tersebut

```

from django.shortcuts import render

from django.http import JsonResponse, HttpResponse

import json

from .shell import run # Import your custom language interpreter function

```

```
from bs4 import BeautifulSoup
import pytube
from sympy import factorint
import numpy as np
from io import StringIO
from decimal import Decimal, getcontext
import math
import io
import base64
import matplotlib.pyplot as plt
import sympy as sp
from mpmath import mp
from django.views.decorators.csrf import csrf_exempt
from datetime import datetime, timedelta
import time
getcontext().prec = 999999999
mp.dps = 999999999
from functools import reduce
import numpy as np
from math import gcd
import sys
from .text_morse import morse_translate, reverse_morse_translate
# views.py
```

```
from .jvdict import Jvdict

from .transliteratejav import transliterate

from .aksara import dotransliterate

from deep_translator import GoogleTranslator

from langdetect import detect

import roman

import textwrap

import os

from math import factorial

from itertools import permutations

from langdetect.lang_detect_exception import LangDetectException

# Set the path to the tessdata directory

from PIL import Image, ImageDraw, ImageFont

import os

import torch

import cv2

from django.views.decorators.csrf import csrf_exempt

import scipy

import numpy as np

from django.http import JsonResponse

import torch

from PIL import Image, ImageFilter

import argparse
```

```
from models.module_photo2pixel import Photo2PixelModel

from utils import img_common_util

from io import BytesIO

from django.http import StreamingHttpResponse

from django.shortcuts import render

from .camera import VideoCamera

from torchvision.transforms import ToTensor

from torchvision.transforms.functional import resize

from scipy.linalg import inv

from alquran_id import AlQuran as Quran

from pytube.innertube import _default_clients

from scipy.linalg import solve

# views.py

import matplotlib.pyplot as plt

from sympy import sympify, simplify

from django.shortcuts import render

import numpy as np

import sympy as sp

import re

#def module lainnya

#def module lainnya
```

```

def polino(request):
    result = None
    if request.method == 'POST':
        polynomial1 = request.POST.get('polynomial1')
        polynomial2 = request.POST.get('polynomial2')
        operation = request.POST.get('operation')

        x = sp.symbols('x')
        poly1 = sp.Poly(polynomial1, x).as_expr() # Convert to SymPy expression
        poly2 = sp.Poly(polynomial2, x).as_expr() # Convert to SymPy expression

    if operation == 'addition':
        result = sp.expand(poly1 + poly2)
    elif operation == 'multiplication':
        result = sp.expand(poly1 * poly2)
    elif operation == 'division':
        quotient, remainder = sp.div(poly1, poly2)
        result = {'quotient': quotient, 'remainder': remainder}
    elif operation == 'roots':
        result = sp.solve(poly1, x)

    # Convert result to LaTeX format for displaying in template

```

```

if result:

    result = sp.latex(result)

# Plot the polynomial if operation is 'roots'

if operation == 'roots':

    x_vals = np.linspace(-10, 10, 400)

    y_vals = np.array([poly1.subs(x, val) for val in x_vals], dtype=float)

    plt.plot(x_vals, y_vals)

    plt.xlabel('x')

    plt.ylabel('f(x)')

    plt.title('Plot of the polynomial')

    plt.grid(True)

    plt.savefig('polynomial_plot.png')

    plt.close()

return render(request, 'polino.html', {'result': result})

```

Bagian ini merupakan embed untuk menampilkan templates polino.html beserta fitur polinomial (Scripting) di framework Django 5.0.4

Save All semua perubahan pada files. kita pergi ke command prompt

***cd bimalog***

***python manage.py makemigrations***

***python manage.py migrate***

***python manage.py collectstatic***

***python manage.py runserver***

C:\Users\Administrator\bimalog>python manage.py runserver

Watching for file changes with StatReloader

Performing system checks...

runserver

System check identified no issues (0 silenced).

April 13, 2024 - 11:33:57

Django version 5.0.4, using settings 'project.settings'

Starting development server at <http://127.0.0.1:8000/>

Quit the server with CTRL-BREAK.

website kita ada pada <http://127.0.0.1:8000/polino>

jika pada browser akan tampil seperti ini

The screenshot shows a web browser window with multiple tabs open. The active tab displays a form titled "Polynomial Operations". The form has two input fields: "Polynomial 1" containing "x\*\*2+x+" and "Polynomial 2" containing "x\*\*2+9". Below these is a dropdown menu for "Select operation" set to "Addition", and a "Calculate" button. At the bottom, the result is shown as "Result: 2 x^2 + x + 14".

## Install Module Python

Seperti yang telah dibahas tadi, python mempunyai package tempat perpustakaan kode disimpan.

Untuk menginstall module matematis, data, atau scraper web (Youtube / ig Downloader), pastikan sudah menginstall python dengan installer windows.

```
pip install <namamodule>
```

```
pip install <namamodule>==<versi>
```

Lalu jika terdapat error pada waktu install ditandai tidak support dan jangan memaksa install. Library Matematis seperti NumPy, SciPy, Matplotlib, Decimal, SymPy MPMATH banyak alternatif lain dan penggunaan dengan membaca dokumentasi module terkait.

Penggunaan module pada Kode kita ditandai dengan

```
import numpy as np
```

```
from pytube import YouTube
```

Cara termudah backup project agar tidak install module manual satu persatu kembali pada command prompt, lalu ketik perintah

```
pip freeze > requirements.txt
```

---

akan muncul file requirements.txt dengan module yang sudah terinstall sebelumnya

```
aksara_ocr @ file:///C:/Users/Administrator/blang-project/bima
```

```
alquran-id==0.0.3
```

```
altair==5.2.0
```

```
altgraph==0.17.4
```

```
asgiref==3.7.2
```

```
attrs==23.2.0
```

beautifulsoup4==4.12.3

blinker==1.7.0

boto3==1.34.37

botocore==1.34.37

bs4==0.0.2

build==1.0.3

CacheControl==0.13.1

cachetools==5.3.2

certifi==2024.2.2

charset-normalizer==3.3.2

cleo==2.1.0

click==8.1.7

colorama==0.4.6

contourpy==1.2.0

crashtest==0.4.1

cycler==0.12.1

deep-translator==1.11.4

distlib==0.3.8

Django==5.0.4

django-cors-headers==4.3.1

dulwich==0.21.7

fastjsonschema==2.19.1

filelock==3.13.1

fonttools==4.48.1

fpdf==1.7.2

fsspec==2024.2.0

gitdb==4.0.11

GitPython==3.1.41

html5lib==1.1

idna==3.6

importlib-metadata==7.0.1

installer==0.7.0

jaraco.classes==3.3.0

Jinja2==3.1.3

jmespath==1.0.1

jsonschema==4.21.1

jsonschema-specifications==2023.12.1

keyring==24.3.0

kiwisolver==1.4.5

langdetect==1.0.9

lmdb==1.4.1

markdown-it-py==3.0.0

MarkupSafe==2.1.5

matplotlib==3.8.2

mdurl==0.1.2

more-itertools==10.2.0

mpmath==1.3.0

msgpack==1.0.7

networkx==3.2.1

numpy==1.26.4

opencv-contrib-python==4.9.0.80

opencv-python==4.9.0.80

packaging==23.2

pandas==2.2.0

patch-ng==1.17.4

pefile==2023.2.7

pexpect==4.9.0

pillow==10.3.0

pkginfo==1.9.6

platformdirs==3.11.0

poetry==1.7.1  
poetry-core==1.8.1  
poetry-plugin-export==1.6.0  
protobuf==4.25.2  
ptyprocess==0.7.0  
pyarrow==15.0.0  
pydeck==0.8.1b0  
Pygments==2.17.2  
pyinstaller==6.5.0  
pyinstaller-hooks-contrib==2024.3  
pyparsing==3.1.1  
pyproject\_hooks==1.0.0  
pytesseract==0.3.10  
python-dateutil==2.8.2  
pytorch-pretrained-bert==0.6.2  
pytube==15.0.0  
pytube3==9.6.4  
pytz==2024.1  
pywin32-ctypes==0.2.2  
rapidfuzz==3.6.1

referencing==0.33.0

regex==2023.12.25

requests==2.31.0

requests-toolbelt==1.0.0

rich==13.7.0

roman==4.1

rpds-py==0.17.1

s3transfer==0.10.0

scipy==1.13.0

setuptools==69.0.3

shellingham==1.5.4

six==1.16.0

smmmap==5.0.1

soupsieve==2.5

sqlparse==0.4.4

streamlit==1.31.0

sympy==1.12

tenacity==8.2.3

textdistance==4.6.1

textwrapper==1.0.2

toml==0.10.2

tomlkit==0.12.3

toolz==0.12.1

torch==2.2.2

torchvision==0.17.2

tornado==6.4

tqdm==4.66.1

trove-classifiers==2024.1.31

typing\_extensions==4.9.0

tzdata==2023.4

tzlocal==5.2

urllib3==2.0.7

validators==0.22.0

virtualenv==20.25.0

watchdog==4.0.0

webencodings==0.5.1

wheel==0.42.0

whitenoise==6.6.0

zipp==3.17.0

Untuk Menginstall ulang module ketik lagi

```
pip install -r requirements.txt
```

Terkadang di kernel Linux python3 dan pip atau pip3

*Kegunaan Modules Python PIP Di tengah kompleksitas pemrograman, module Python muncul sebagai “penyelamat” yang memungkinkan analyst menyimpan, mengorganisasi, dan mengakses kode dengan cara lebih terstruktur serta modular. Module Python berfungsi sebagai library kode yang dapat dengan mudah diakses dan digunakan ulang. Module memastikan setiap bagian kode memiliki tempatnya sendiri dan dapat ditemukan dengan cepat saat dibutuhkan. Dengan memanfaatkan kekuatan module Python, kamu tidak hanya mempercepat proses pembuatan kode, tetapi juga memastikan kode tersebut clean, terorganisasi, dan siap untuk kolaborasi.*

Modules python memiliki tipe file py sama dengan program python. tipe file di belakang nama file menandakan jenis files tersebut

.mp3 //Tipe Audio

.mp4 //Tipe Video

.ogg // Sama

.txt // Tipe Text Bebas

.py // Tipe Python File

.cpp // Tipe C++ Language

dan banyak lain , tipe file harus di jalankan dengan compiler yang sesuai, Python interpreter sebagai eksekutor py file, Galery dan Video Player sebagai eksekutor file video dan gambar. Web Browser sebagai eksekutor file html, css, php, Bahasa Markup dan Scripting Web.

Projek Bimalog Django pada langkah Framework django merupakan sedikit dari implementasi apa itu pemrograman dan coding. Python Multi Operasi Sistem, Selain Django alternatif lainnya ada Flask.

# Membuat Projek Website Pertama

Direktori bimalog sudah opensource terdapat di [github.com/SecretDiscorder/bimalog](https://github.com/SecretDiscorder/bimalog)

The screenshot shows the GitHub repository page for 'bimalog' owned by 'SecretDiscorder'. The repository has 5 branches and 3 tags. The 'Code' tab is selected. The repository description is 'PYTHON WEBSITE AND EXECUTABLE WINDOWS FILE FRAMEWORK DJANGO'. It has 50 commits, 1 star, 1 watching, and 0 forks. A release titled 'BIMALOG V3' is shown with a download link for 'dist.zip'. The release notes mention a system check and a Django server start. The assets section lists three files: 'dist.zip', 'Source code (zip)', and 'Source code (tar.gz)'. The date for all assets is 27 minutes ago.

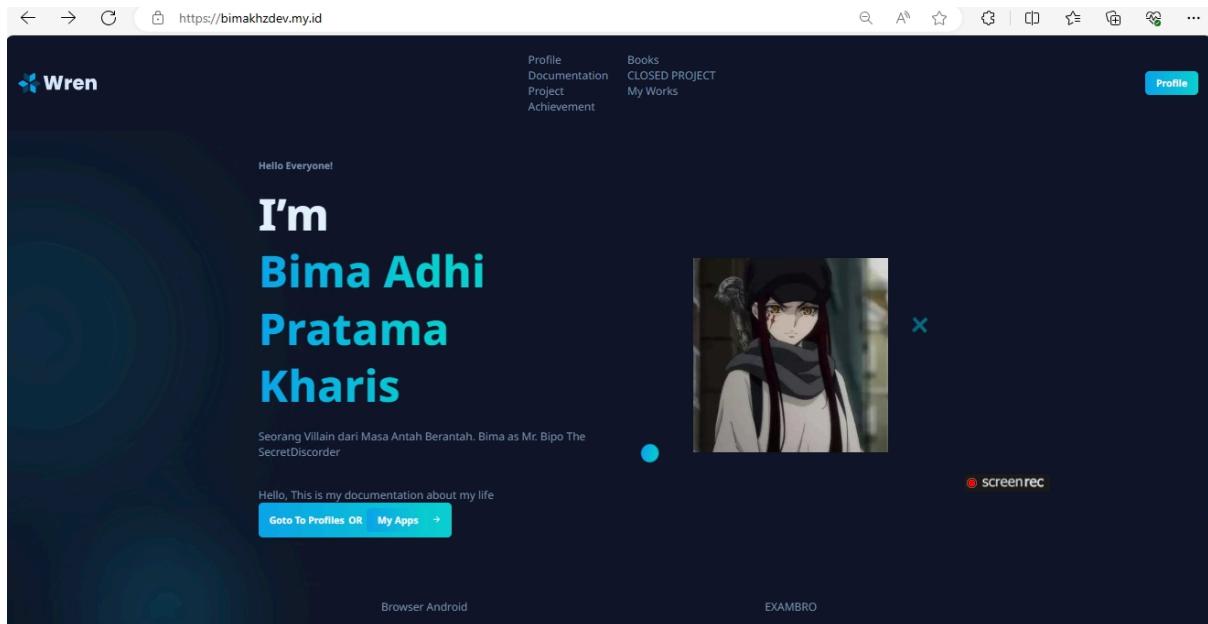
## PBSSI JAYA TEKNO APPS



Untuk menu pada url path /base/ Konsep yang dipakai sama dengan polino.

Perlu pengetahuan mengenai HTML CSS JS AI PYTHON, jadi disarankan membaca dan berlatih mengetik kode dasar yang diperlukan, aktif bertanya di komunitas terkait, atau practical dengan AI ChatGPT, sebagai referensi.

Saya tidak menjabarkan kode terkait bimalog dikarenakan banyaknya fungsi dan luasnya module yang dipakai. Jadikan ini sebagai referensi untuk membuat Fungsi Website anda sendiri.



Domain saya pada bimakhzdev.my.id

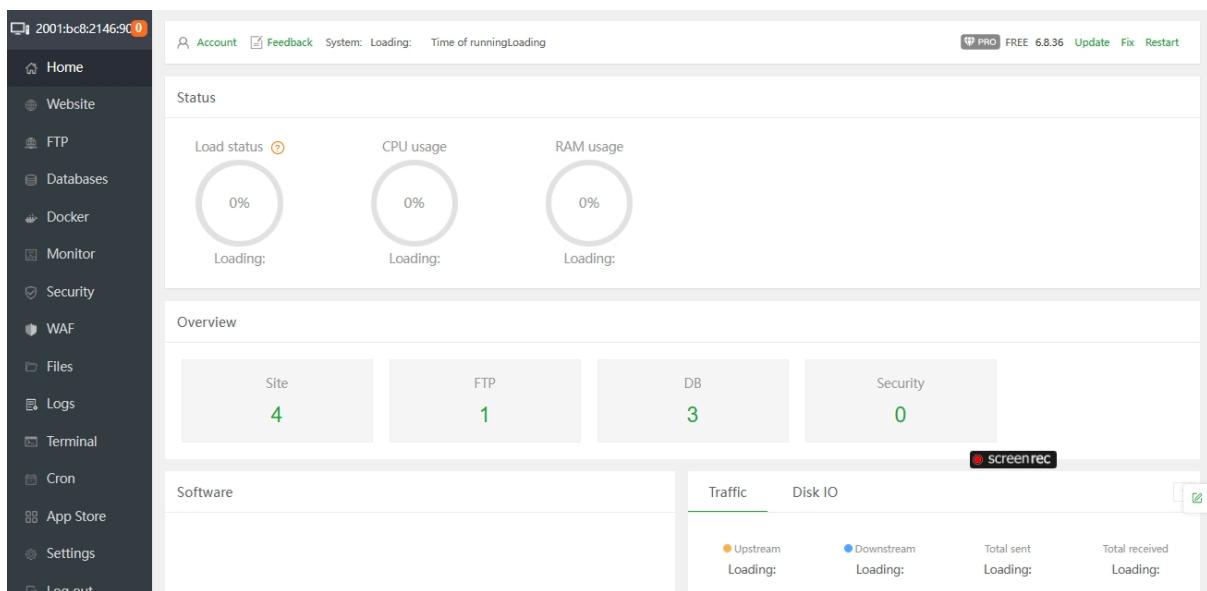
## Deploy Website Python Online

Untuk mengonlinekan Code 24 jam, diperlukan Hosting ( cara termudah ) ataupun manual (Lengkap namun Triky) VPS.

A screenshot of a Microsoft Bing search results page. The search query 'vps adalah' is entered in the search bar. The results section is titled 'Virtual Private Server' and includes a snippet from 'hostinger.co.id' and another from 'dewaweb.com'. To the right, there's a detailed article snippet about VPS from 'screenrec'. The page also shows various search filters like 'SEARCH', 'COPilot', 'MY BING', etc.

Saya Menggunakan VPS TIPE NAT, yang support Domain Fordwarding dan Operasi Sistem Ubuntu Linux Jammy.

banyak persyaratan jika deploy melalui vps. Harus akses IP VPS dan Port Forwarding, menginstall Panel (AAPanel), dan Domain Forwarding, Nameserver juga diperlukan (Cloudflare).



Contoh akses IP Website AAPanel untuk deploy Django Website.

Panduan VPS bisa dilihat di google untuk domain dan portnya, Khusus AAPanel saya merujuk ke pembahasan deploy django di link [Saya Copy Paste saja dan translate ke Indonesia](#)

<https://forum.aapanel.com/d/13338-python-manager-deploy-djangoblog>

1.Pertama instal alat git untuk mengkloning proyek DjangoBlog::

RedHat|CentOS:

```
yum install git -y
```

Debian|Ubuntu:

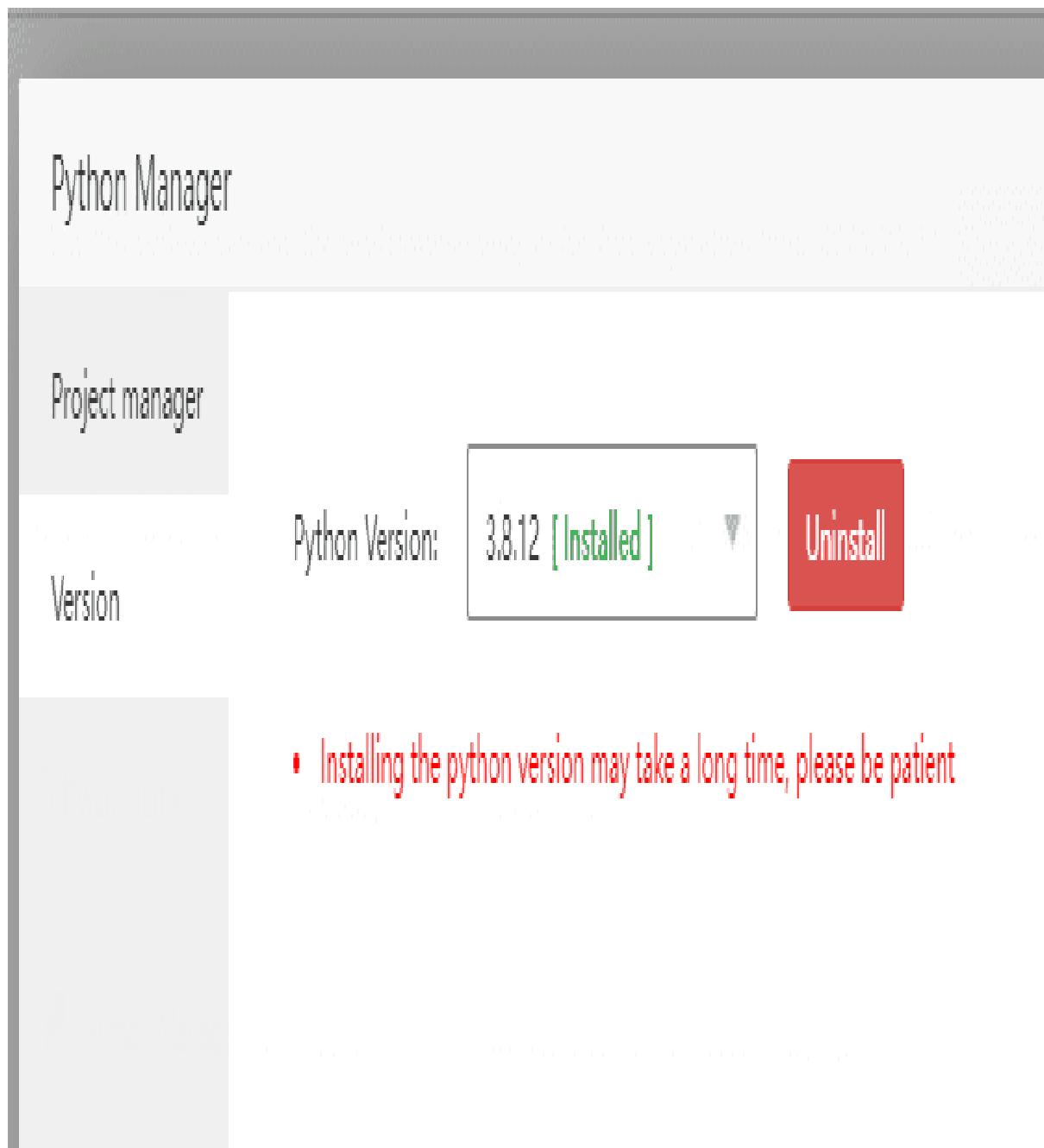
```
apt-get update  
apt-get install git -y
```

Clone the DjangoBlog project:

```
cd /www/wwwroot/  
git clone https://github.com/liangliangyy/DjangoBlog
```

```
root@test:~# cd /www/wwwroot/
root@test:/www/wwwroot# git clone https://github.com/liangliangyy/DjangoBlog
Cloning into 'DjangoBlog'...
remote: Enumerating objects: 4759, done.
remote: Counting objects: 100% (26/26), done.
remote: Compressing objects: 100% (22/22), done.
remote: Total 4759 (delta 7), reused 21 (delta 4), pack-reused 4733
Receiving objects: 100% (4759/4759), 3.79 MiB | 2.35 MiB/s, done.
Resolving deltas: 100% (2871/2871), done.
root@test:/www/wwwroot#
```

## 2. Install Python 3.8.12



### 3.Tambahkan DjangoBlog project

Add a project

|                  |  |
|------------------|--|
| Name             | DjangoBlog   |
| Path             | /www/wwwroot/DjangoBlog             |
| Version          | 3.8.12                              |
| Framework        | django                              |
| Startup mode     | gunicorn                            |
| Startup file/dir | /www/wwwroot/DjangoBlog/djangoblog  |
| Port             | 8000   |
| Run user         | root                                |
| Command          | Enter the project port, no port is left blank  |

Install module now

Start with the sys

- Version:To add another version of Python, please **Installed** in python manager
- Gunicorn Is a widely used Python WSGI UNIX HTTP server
- uWsgi Is a high performance WEB server
- python Run the project directly using python
- The project log needs to be located in the project /project\_dir/logs/error.log otherwise the log cannot be obtained
- Customized project path needs to fill in the full path

**cancel** **confirm**

Parameter Deskripsi:

- Nama:Beri nama proyek Anda
- Path:Pilih direktori root proyek,Pilih direktori kloning sebelumnya
- Versi:Pilih versi python yang dibutuhkan proyek Anda,Pilih Versi Python 3.8.12 di sini
- Framework: Kerangka proyek proyek, proyek saya di sini adalah Flask, jadi pilih django
- Mode startup: Pilih gunicorn di sini, Anda dapat mengganti opsi lain sesuai dengan kebutuhan Anda
- startuo file/dir:diango pilih direktori proyek untuk memulai
- Port:DjangoBlog default ke 8000
- Jalankan user:Start dengan root privileges
- Instal modul sekarang:Saat menambahkan proyek, instal modul yang diperlukan sesuai dengan dokumentasi direktori akar proyek requirements.txt.
- Mulai dengan sys:Configure startup untuk proyek

Ketika kami menyelesaikan pembuatan proyek, kami tidak dapat membuka proyek secara normal. Kita perlu menginisialisasi proyek

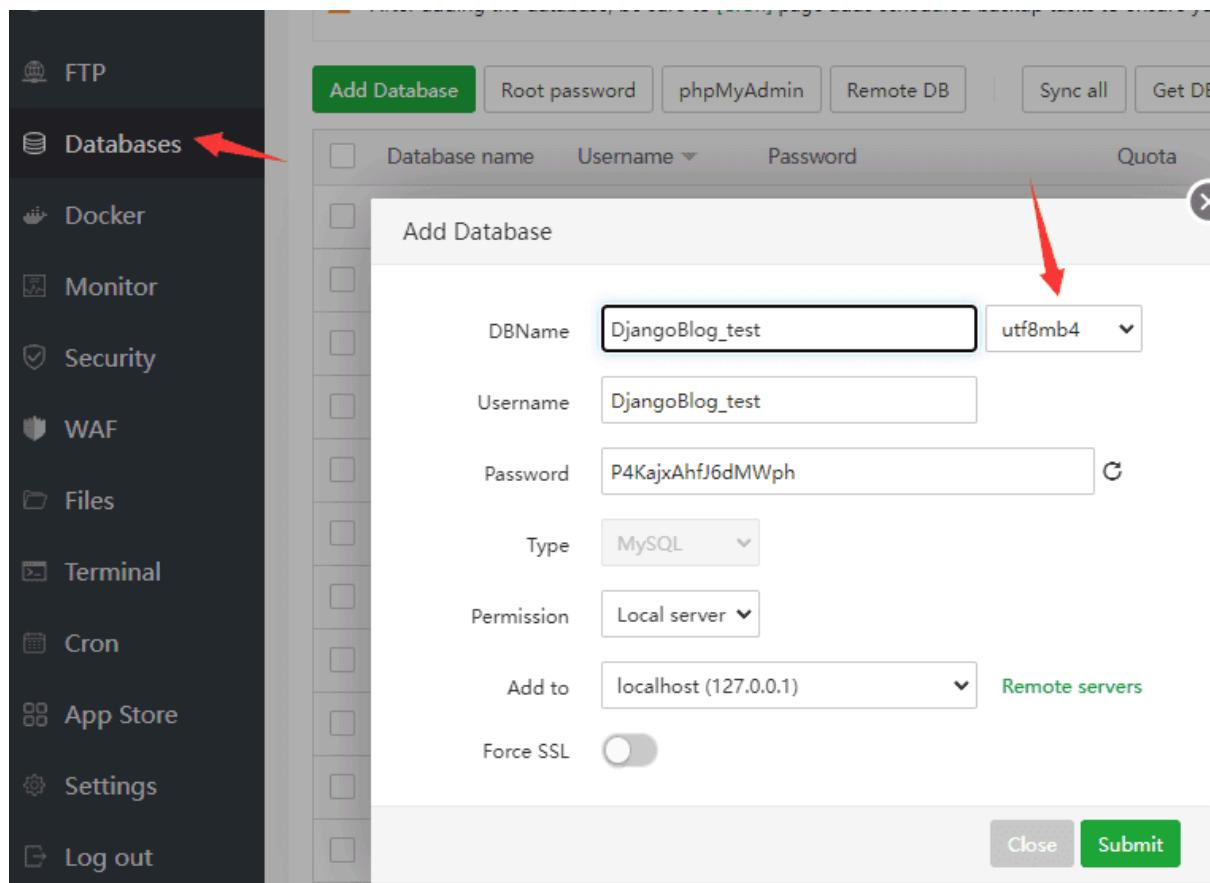
Setelah instalasi selesai, tampilannya adalah sebagai berikut:

The screenshot shows the Python Manager interface. On the left, there's a sidebar with 'Project manager' and other options like 'Logs', 'Version', and 'Python Manager'. The main area has a heading 'Add Project' and a table listing projects. One project is listed: 'Djang...' with path '/www/wwwroot/Dj...', port 8000, mode 'gunicorn', version 3.8.12, CPU 51%, memory 79.73 MB. There are buttons for 'Open', 'Log | Mapping | Restart | Config | Module | Del'.

| Name     | Path               | Port | Mode     | Ver    | CPU | Mem      | Status | Auto start | Operating                                       |
|----------|--------------------|------|----------|--------|-----|----------|--------|------------|---|
| Djang... | /www/wwwroot/Dj... | 8000 | gunicorn | 3.8.12 | 51% | 79.73 MB | ▶      | Open       | Log   Mapping   Restart   Config   Module   Del |

#### 4. Membuat database untuk proyek DjangoBlog

Saat membuat, Anda perlu memperhatikan pemilihan set karakter: utf8mb4



⚠ After adding the database, be sure to [Cron] page adds scheduled backup tasks to ensure your data security.

| Add Database             |                 | Root password   | phpMyAdmin | Remote DB | Sync all           | Get DB from server |
|--------------------------|-----------------|-----------------|------------|-----------|--------------------|--------------------|
| <input type="checkbox"/> | Database name   | Username        | Password   | Quota     | Backup             |                    |
| <input type="checkbox"/> | djangoblog_test | DjangoBlog_test | *****      | Not set   | Exists(3)   Import |                    |

## 5.Modify konfigurasi DjangoBlog:

Pergi ke direktori DjangoBlog, masukkan direktori djangoblog, dan ubah file konfigurasi: settings.py

Setelah mengomentari database asli, kemudian tambahkan konten berikut, perhatikan untuk memodifikasinya ke nama database Anda, nama pengguna, kata sandi, perhatikan kasus database.

```
/www/wwwroot/DjangoBlog/djangoblog
```

```
DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.mysql',
```

```

        'NAME': 'djangoblog_test',
        'USER': 'DjangoBlog_test',
        'PASSWORD': 'P4KajxAhfJ6dMWph',
        'HOST': 'localhost',
        'PORT': 3306,
        'OPTIONS': {'charset': 'utf8mb4'},
    }
}

```

```

# Save | Save All | Refresh | Search | Replace | JumpLine | Font | Theme | Set | Help |
Directory: /www/wwwroot/DjangoBlog...
Back Refresh + New Search settings.py ✎
WHOOSH_INDEX
_init_.py
admin_site.py
blog_signals.py
elasticsearch_backend.py
feeds.py
logentryadmin.py
settings.py
sitemap.py
spider_notify.py
tests.py
urls.py
utils.py
whoosh_cn_backend.py
wsgi.py

89     ], [
90         'django.template.context_processors.request',
91         'django.contrib.auth.context_processors.auth',
92         'django.contrib.messages.context_processors.messages',
93         'blog.context_processors.seo_processor'
94     ],
95 ],
96 ]
97
98 WSGI_APPLICATION = 'djangoblog.wsgi.application'
99
100 # Database
101 # https://docs.djangoproject.com/en/1.10/ref/settings/#databases
102
103
104 #DATABASES = {
105 #     'default': {
106 #         'ENGINE': 'django.db.backends.mysql',
107 #         'NAME': os.environ.get('DJANGO_MYSQL_DATABASE') or 'djangoblog',
108 #         'USER': os.environ.get('DJANGO_MYSQL_USER') or 'root',
109 #         'PASSWORD': os.environ.get('DJANGO_MYSQL_PASSWORD') or 'djangoblog_123',
110 #         'HOST': os.environ.get('DJANGO_MYSQL_HOST') or '127.0.0.1',
111 #         'PORT': int(
112 #             os.environ.get('DJANGO_MYSQL_PORT') or 3306),
113 #         'OPTIONS': {
114 #             'charset': 'utf8mb4',
115 #         })
116
117 DATABASES = {
118     'default': {
119         'ENGINE': 'django.db.backends.mysql',
120         'NAME': 'djangoblog_test',
121         'USER': 'DjangoBlog_test',
122         'PASSWORD': 'P4KajxAhfJ6dMWph',
123         'HOST': 'localhost',
124         'PORT': 3306,
125         'OPTIONS': {'charset': 'utf8mb4'},
126     }
127 }
128
129 # Password validation
130 # https://docs.djangoproject.com/en/1.10/ref/settings/#auth-password-validators
131
132 AUTH_PASSWORD_VALIDATORS = [

```

Comment the configuration below, otherwise it will cause database errors later, only keep:  
TIME\_ZONE = 'Asia/Shanghai'

#LANGUAGE\_CODE = 'zh-hans' (English will be displayed after adding annotations)

#USE\_I18N = True

```
#USE_L10N = True
```

```
#USE_TZ = True
```



Konfigurasi pada proyek

Proyek Python env yang digunakan oleh proyek

```
root@test:/www/wwwroot/DjangoBlog# ls -al
total 120
drwxr-xr-x 17 root root 4096 Jun  7 11:38 .
drwxr-xr-x 45 www www 4096 Jun  7 11:30 ..
drwxr-xr-x  6 www www 4096 Jun  7 11:35 8658305af42d6efded53c296d677d3ba_venv
drwxr-xr-x  5 root root 4096 Jun  7 11:38 accounts
drwxr-xr-x  2 root root 4096 Jun  7 11:30 bin
drwxr-xr-x  7 root root 4096 Jun  7 11:38 blog
drwxr-xr-x  5 root root 4096 Jun  7 11:38 comments
-rw-r--r--  1 root root 144 Jun  7 11:30 .coveragerc
drwxr-xr-x  3 root root 4096 Jun  7 11:38.djangoblog
-rw-r--r--  1 root root   0 Jun  7 11:38.djangoblog.log
-rw-r--r--  1 root root 1095 Jun  7 11:30 docker-compose.es.yml
-rw-r--r--  1 root root 1538 Jun  7 11:30 docker-compose.yml
-rw-r--r--  1 root root  487 Jun  7 11:30 Dockerfile
-rw-r--r--  1 root root  159 Jun  7 11:30 .dockerignore
drwxr-xr-x  2 root root 4096 Jun  7 11:30 docs
drwxr-xr-x  8 root root 4096 Jun  7 11:30 .git
-rw-r--r--  1 root root 126 Jun  7 11:30 .gitattributes
drwxr-xr-x  3 root root 4096 Jun  7 11:30 .github
-rw-r--r--  1 root root 1093 Jun  7 11:30 .gitignore
-rw-r--r--  1 root root  409 Jun  7 11:38 gunicorn.conf
-rw-r--r--  1 root root 1075 Jun  7 11:30 LICENSE
drwxr-xr-x  2 root root 4096 Jun  7 11:38 logs
-rwrxr-xr-x  1 root root  808 Jun  7 11:30 manage.py
drwxr-xr-x  5 root root 4096 Jun  7 11:38 oauth
drwxr-xr-x  4 root root 4096 Jun  7 11:38 owntracks
drwxr-xr-x  2 root root 4096 Jun  7 11:38 __pycache__
-rw-r--r--  1 root root 4605 Jun  7 11:30 README.md
-rw-r--r--  1 root root  509 Jun  7 11:30 requirements.txt
drwxr-xr-x  5 root root 4096 Jun  7 11:38 servermanager
drwxr-xr-x  9 root root 4096 Jun  7 11:30 templates
root@test:/www/wwwroot/DjangoBlog#
```

Harap ganti "8658305af42d6efded53c296d677d3ba\_venv" dengan env proyek Anda dan jalankan perintah.

6. Mulai melakukan migrasi data Menghasilkan data:

```
/www/wwwroot/DjangoBlog/8658305af42d6efded53c296d677d3ba_venv/bin/python3
manage.py makemigrations
/www/wwwroot/DjangoBlog/8658305af42d6efded53c296d677d3ba_venv/bin/python3
manage.py migrate
```

```

root@test:/www/wwwroot/DjangoBlog# /www/wwwroot/DjangoBlog/8658305af42d6efded53c296d
677d3ba_venv/bin/python3 manage.py makemigrations
/www/wwwroot/DjangoBlog/8658305af42d6efded53c296d677d3ba_venv/lib/python3.8/site-pac
kages/django/core/management/commands/makemigrations.py:121: RuntimeWarning: Got an
error checking a consistent migration history performed for database connection 'def
ault': (1044, "Access denied for user 'DjangoBlog_test'@'localhost' to database 'Dja
ngoBlog_test'")
    warnings.warn(
                                                Pay attention to the capitalization of the database.

Migrations for 'owntracks':
  owntracks/migrations/0001_initial.py
    - Create model OwnTrackLog
Migrations for 'servermanager':
  servermanager/migrations/0001_initial.py
    - Create model commands
    - Create model EmailSendLog
Migrations for 'accounts':
  accounts/migrations/0001_initial.py
    - Create model BlogUser
Migrations for 'blog':
  blog/migrations/0001_initial.py
    - Create model BlogSettings
    - Create model Links
    - Create model SideBar
    - Create model Tag
    - Create model Category
    - Create model Article
Migrations for 'comments':
  comments/migrations/0001_initial.py
    - Create model Comment
Migrations for 'oauth':
  oauth/migrations/0001_initial.py
    - Create model OAuthConfig
    - Create model OAuthUser

```

```

root@test:/www/wwwroot/DjangoBlog# /www/wwwroot/DjangoBlog/8658305af42d6efded53c296d
677d3ba_venv/bin/python3 manage.py migrate
Operations to perform:
  Apply all migrations: accounts, admin, auth, blog, comments, contenttypes, oauth,
  owntracks, servermanager, sessions, sites
Running migrations:
  Applying contenttypes.0001_initial... OK
  Applying contenttypes.0002_remove_content_type_name... OK
  Applying auth.0001_initial... OK
  Applying auth.0002.Alter_permission_name_max_length... OK
  Applying auth.0003.Alter_user_email_max_length... OK
  Applying auth.0004.Alter_user_username_opts... OK
  Applying auth.0005.Alter_user_last_login_null... OK
  Applying auth.0006_require_contenttypes_0002... OK
  Applying auth.0007.Alter_validators_add_error_messages... OK
  Applying auth.0008.Alter_user_username_max_length... OK
  Applying auth.0009.Alter_user_last_name_max_length... OK
  Applying auth.0010.Alter_group_name_max_length... OK
  Applying auth.0011_update_proxy_permissions... OK
  Applying auth.0012.Alter_user_first_name_max_length... OK
  Applying accounts.0001_initial... OK
  Applying admin.0001_initial... OK
  Applying admin.0002_logentry_remove_auto_add... OK
  Applying admin.0003_logentry_add_action_flag_choices... OK
  Applying blog.0001_initial... OK
  Applying comments.0001_initial... OK
  Applying oauth.0001_initial... OK
  Applying owntracks.0001_initial... OK
  Applying servermanager.0001_initial... OK
  Applying sessions.0001_initial... OK
  Applying sites.0001_initial... OK
  Applying sites.0002_Alter_domain_unique... OK
root@test:/www/wwwroot/DjangoBlog#

```

Hasilkan pengguna super: (buat pengguna administrator sesuai dengan prompt)

```
/www/wwwroot/DjangoBlog/8658305af42d6efded53c296d677d3ba_venv/bin/python3  
manage.py createsuperuser
```

```
root@test:/www/wwwroot/DjangoBlog# /www/wwwroot/DjangoBlog/8658305af42d6efded53c296d  
677d3ba_venv/bin/python3 manage.py createsuperuser  
用户名: kern123  
电子邮件地址: kern@aapanel.com  
Password:  
Password (again):  
[2022-06-07 12:18:09,874] INFO [djangoblog.utils.news:52 utils] cache_decorator set  
cache:get_current_site key:c9e31772484fbedeb0a29223db6723c850ee5b39c1d5cea19c72f96a5  
c56c115  
[2022-06-07 12:18:09,874] INFO [djangoblog.utils.news:52 utils] cache_decorator set  
cache:get_current_site key:c9e31772484fbedeb0a29223db6723c850ee5b39c1d5cea19c72f96a5  
c56c115  
[2022-06-07 12:18:10,147] INFO [djangoblog.spider_notify.baidu_notify:16 spider_notify]  
{"remain":100000,"success":0,"not_same_site":["https://example.com/author/kern12  
3.html"]}  
[2022-06-07 12:18:10,147] INFO [djangoblog.spider_notify.baidu_notify:16 spider_notify]  
{"remain":100000,"success":0,"not_same_site":["https://example.com/author/kern12  
3.html"]}  
Superuser created successfully.  
root@test:/www/wwwroot/DjangoBlog#
```

Hasilkan test data:

```
/www/wwwroot/DjangoBlog/8658305af42d6efded53c296d677d3ba_venv/bin/python3  
manage.py createtestdata
```

```
cache:get_current_site key:9a40b26f160c591f034d7073a86/ecf090cbd0fe84e1ed1/bd04bc/b9  
6fd11bc  
[2022-06-07 12:19:23,659] INFO [djangoblog.spider_notify.baidu_notify:16 spider_notify]  
{"remain":100000,"success":0,"not_same_site":["https://example.com/tag/biao-qian  
-19.html"]}  
[2022-06-07 12:19:23,659] INFO [djangoblog.spider_notify.baidu_notify:16 spider_notify]  
{"remain":100000,"success":0,"not_same_site":["https://example.com/tag/biao-qian  
-19.html"]}  
[2022-06-07 12:19:23,669] INFO [djangoblog.utils.news:52 utils] cache_decorator set  
cache:get_current_site key:9a40b26f160c591f034d7073a867ecf090cbd0fe84e1ed17bd04bc7b9  
6fd11bc  
[2022-06-07 12:19:23,669] INFO [djangoblog.utils.news:52 utils] cache_decorator set  
cache:get_current_site key:9a40b26f160c591f034d7073a867ecf090cbd0fe84e1ed17bd04bc7b9  
6fd11bc  
[2022-06-07 12:19:23,924] INFO [djangoblog.spider_notify.baidu_notify:16 spider_notify]  
{"remain":100000,"success":0,"not_same_site":["https://example.com/article/2022/  
6/7/19.html"]}  
[2022-06-07 12:19:23,924] INFO [djangoblog.spider_notify.baidu_notify:16 spider_notify]  
{"remain":100000,"success":0,"not_same_site":["https://example.com/article/2022/  
6/7/19.html"]}  
created test datas
```

Collect static files:

```
/www/wwwroot/DjangoBlog/8658305af42d6efded53c296d677d3ba_venv/bin/python3  
manage.py collectstatic --noinput  
/www/wwwroot/DjangoBlog/8658305af42d6efded53c296d677d3ba_venv/bin/python3  
manage.py compress --force
```

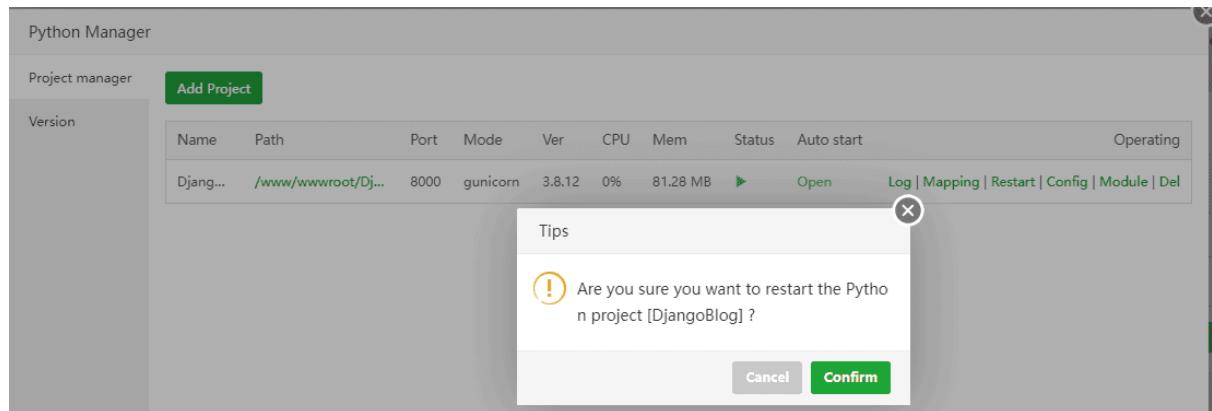
```

root@test:/www/wwwroot/DjangoBlog# /www/wwwroot/DjangoBlog/8658305af42d6efded53c296d
677d3ba_venv/bin/python3 manage.py collectstatic --noinput

581 static files copied to '/www/wwwroot/DjangoBlog/collectedstatic'.
root@test:/www/wwwroot/DjangoBlog# /www/wwwroot/DjangoBlog/8658305af42d6efded53c296d
677d3ba_venv/bin/python3 manage.py compress --force
Compressing... done
Compressed 4 block(s) from 28 template(s) for 1 context(s).

```

Restart ulang DjangoBlog project



## 7. Port 8000 yang digunakan oleh DjangoBlog dirilis di Keamanan

| Firewall    |                    |      |                   |           |                                     |
|-------------|--------------------|------|-------------------|-----------|-------------------------------------|
| Accept Port |                    | Port | Notes/Description | Open      | SYS Firewall                        |
| 66          | Accept port:[8000] |      |                   | Listening | 2022-06-07 11:45:50<br>DjangoBlog   |
| 64          | Accept port:[88]   |      |                   | Listening | 2022-05-14 14:24:50<br>www.load.com |

Coba akses

Buka browser: <http://127.0.0.1:8000/> dan Anda dapat melihat efeknya

Ganti alamat ip dengan alamat server Anda.

192.168.247.129:8000

# djangoblog

基于Django的博客系统

首页 我是父类目 文章归档

**nice title 19**

发表评论 0 views

VIEW

nice content 19

[Read more](#)

本条目发布于 2022-06-07。属于子类目 分类，被贴了 标签， 标签19 标签。作者是 测试用户。

nice title 1 - 0 views  
 nice title 2 - 0 views  
 nice title 3 - 0 views  
 nice title 4 - 0 views  
 nice title 5 - 0 views  
 nice title 6 - 0 views

8. Petakan proyek DjangoBlog ke layanan web (nginx/apache), dan nginx/apache bertanggung jawab untuk meneruskan permintaan ke proyek Python kami di backend

Python Manager

Project manager Add Project

| Name     | Path               | Port | Mode    | Ver    | CPU   | Mem     | Status | Auto start   | Operating |
|----------|--------------------|------|---------|--------|-------|---------|--------|--|-----------|
| Djang... | /www/wwwroot/Dj... | 8000 | unicorn | 3.8.12 | 26.5% | 79.9 MB | ▶      | Open Log   Mapping   Restart   Config   Module   Del |           |

Mapping project [DjangoBlog]

Domain name  (arrow pointing to this field)

Cancel Confirm

Where is the project env:  
 project env path: project\_path/project\_name\_md5\_venv/bin/  
 E.g.: /data/python/1fc54ae5e387be0d369edddc2010e5e\_venv/bin/

**Website** (arrow pointing to this tab)

After the website is created, the file owner will be set to [www], you can add your backup plan in after the site is successfully established, please[Cron];

PHP Project Node Project

Add site Default Page Default Website PHP CLI version Category manager

| Site name         | Status    | Back up   | Document Root          | Expired date | Note                                    | PH  |
|-------------------|-----------|-----------|------------------------|--------------|---|-----|
| 129djangoblog.com | Running ▶ | Not exist | /www/wwwroot/129dja... | Perpetual    | Mapping site for Python project [Dja... | Sta |

9. Konfigurasi file statis

Tambahkan yang berikut ini ke penulisan ulang URL konfigurasi situs:

```
location /static/ {  
  
    alias /www/wwwroot/DjangoBlog/collectedstatic/;  
  
    expires max;  
  
    access_log      off;  
  
    log_not_found   off;  
  
}  
  
location /media {  
  
    alias /www/wwwroot/DjangoBlog/uploads/;  
  
    expires max;  
  
}  
  
location ~ \.py$ {  
  
    return 403;  
  
}
```

Domain Manager      0.Current      Rewrite rule converter: [Apache to Nginx](#)

Subdirectory binding  
Site directory  
Limit access  
Traffic control  
URL rewrite  
Default indexes  
Config  
SSL  
PHP version  
Composer  
Redirect  
Reverse proxy

```
2 | location ~ \.well-known{  
3 | | | | allow all;  
4 | | | }  
5 | location /static/ {  
6 | | | alias /www/wwwroot/DjangoBlog/collectedstatic/;  
7 | | | expires max;  
8 | | | access_log off;  
9 | | | log_not_found off;  
10| | }  
11| location /media {  
12| | alias /www/wwwroot/DjangoBlog/uploads/;  
13| | expires max;  
14| | }  
15| location ~ \.py$ {  
16| | return 403;  
17| }
```

[Save](#)   [Save as template](#)

- Please select your application.
- If the site cannot be accessed after the rewrite rules set, please try to reset to default.

## 10.DjangoBlog project deploys SSL

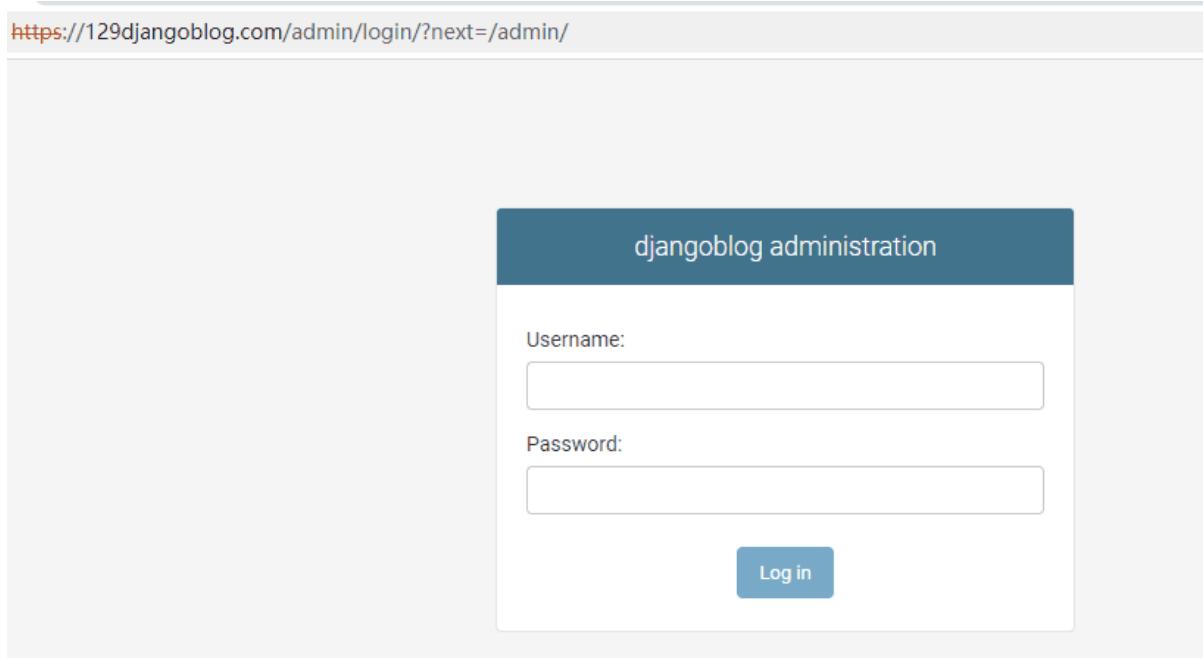
Site modification[129djangoblog.com] -- Time added[2022-06-07 14:27:28]

|                      |  |  |
|----------------------|--|--|
| Domain Manager       | <input type="button" value="Let's Encrypt"/> <input type="button" value="Other certificate"/> <input type="button" value="Turn off"/> <input type="button" value="Certificate holder"/>  | <input checked="" type="button"/> Force HTTPS <input type="checkbox"/> |
| Subdirectory binding | Verification mode <input checked="" type="radio"/> File verification <input type="radio"/> DNS verification  |  |
| Site directory       |  |  |
| Limit access         | Domain name <input checked="" type="checkbox"/> Select All<br><input checked="" type="checkbox"/> 129djangoblog.com  |  |
| Traffic control      |  |  |
| URL rewrite          | <input type="button" value="Apply"/>   |  |
| Default indexes      | <ul style="list-style-type: none"> <li>Please check domain resolution before applying, unresolved domain leads to audit failure.</li> <li>Let's Encrypt free certificate, valid for 3 months, supports wildcard domain. Auto-renew by default.</li> <li>If your site uses CDN or 301 redirect, you will not be able to apply for and renew certificate through file verification.</li> <li>When you do not specify a SSL default site, if you use HTTPS to access a site that does not have SSL enabled, you will directly access a site that has SSL enabled</li> </ul> |  |
| Config               |  |  |
| SSL                  |  |  |
| PHP version          |  |  |
| Composer             |  |  |

Site modification[129djangoblog.com] -- Time added[2022-06-07 14:27:28]

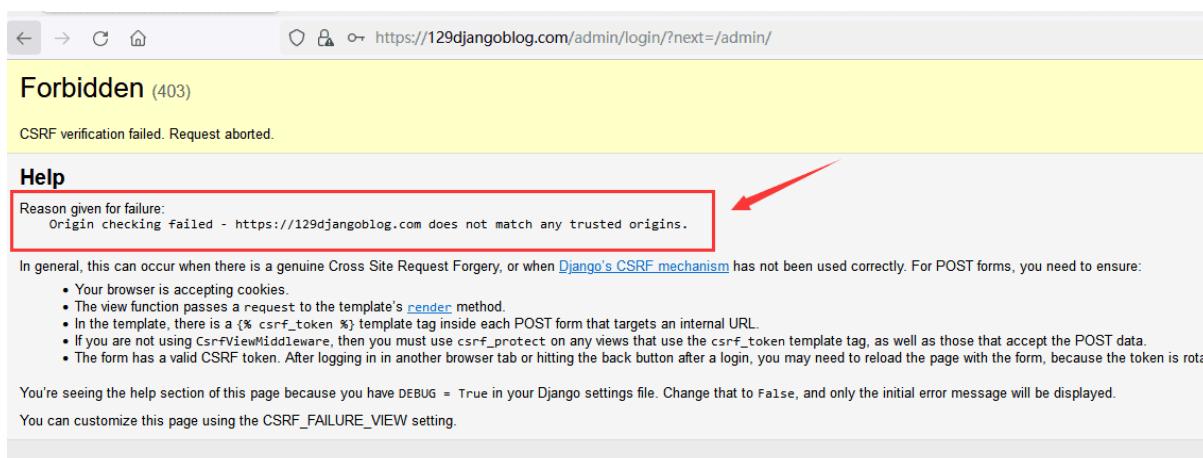
|                      |  |  |
|----------------------|--|--|
| Domain Manager       | <input type="button" value="Let's Encrypt"/> <input type="button" value="Other certificate"/> <input type="button" value="Turn off"/> <input type="button" value="Certificate holder"/>  | <input checked="" type="button"/> Force HTTPS <input type="checkbox"/> |
| Subdirectory binding | <b>Successfully deployed:</b> Please replace the new certificate before the certificate expires<br><b>Brand:</b> 129djangoblog.com <b>Certified domain(s):</b> 129djangoblog.com<br><b>Expire date:</b> 2032-06-04   |  |
| Site directory       |  |  |
| Limit access         | Private key (KEY)  | Certificate (CRT/PEM)  |
| Traffic control      | -----BEGIN RSA PRIVATE KEY-----<br>MIIEpAIBAAKCAQEaxtaa+/A2Ae52pKHPsKX<br>TD/y8kcmS318M2XheTqbDsMIM/tjL<br>Q9/rZMHL5erGbZPaAoqV5KkVEkE4fL+HN0y<br>puh4Qf+ZWwEaLfEJSR/O8RhT3IPT0<br>G+cbohEIOilgmlN9pp/OoVciHgyhOP2Kp/Nz<br>UNa9m/4NebzFzg4n/g7LkTt690AP<br>eGydmOqtENOZM2ra/XcxkqPpyAhImLYHft5<br>Qh2JZve/4FcxrTjrLDYa0kRPM9Auo<br>/9AeYxBEwrHB5DL/fonT70IP3TYddRmqqnP |  |
| URL rewrite          | -----BEGIN CERTIFICATE-----<br>MIIDazCCAlMCFC1DeB5aA9gaq+AotFcMHef<br>NiTFXMA0GCSqGSIb3DQEBCwUAMHIx<br>CzAJBgNVBAYTAmNuMQswCQYDVQQIDAjK<br>ZzELMAkGA1UEBwwCZGcxCzAJBgNVBAoM<br>AmtrMQswCQYDVQQLDAJrazEaMBgGA1UE<br>AwwRMTI5ZGphbmdvYmxvZy5jb20xEzAR<br>BgkqhkiG9w0BCQEWBGFhCAGwHhcNMjlwN<br>jA3MDcxOTI3WhcNMzlwNjA0MDcxOTI3<br>WjByMQswCQYDVQQGEwJbjELMAkGA1UEC        |  |
| Default indexes      |  |  |
| Config               |  |  |
| SSL                  |  |  |
| PHP version          |  |  |
| Composer             | <input type="button" value="Save"/> <input type="button" value="Disable SSL"/>   |  |

Access effect:



Login background error after deploying SSL:

<https://129djangoblog.com/admin>



Solution:

Modifikasi File Konfigurasi: /www/wwwroot/DjangoBlog/djangoblog/settings.py

```
ALLOWED_HOSTS = ['*', '127.0.0.1', '129djangoblog.com']  
# django 4.0新增配置
```

```
CSRF_TRUSTED_ORIGINS = ['https://129djangoblog.com']
```

Ganti djangoblog.com Dengan domainmu domain name, kemudian restart project (go to Python Manager to restart the DjangoBlog project.)

Directory: ./www/wwwroot/DjangoBlog...

```

settings.py ✘
16 def env_to_bool(env, default):
17     str_val = os.environ.get(env)
18     return default if str_val is None else str_val == 'True'
19
20 # Build paths inside the project like this: os.path.join(BASE_DIR, ...)
21
22 BASE_DIR = os.path.dirname(os.path.dirname(os.path.abspath(__file__)))
23
24 # Quick-start development settings - unsuitable for production
25 # See https://docs.djangoproject.com/en/1.10/howto/deployment/checklist/
26
27 # SECURITY WARNING: keep the secret key used in production secret!
28 SECRET_KEY = os.environ.get(
29     'DJANGO_SECRET_KEY') or 'n9ceqv38#&mwuat@mjb_p%em$e8$qyr#fw9ot!=ba6lijx-6'
30
31 # SECURITY WARNING: don't run with debug turned on in production!
32 DEBUG = env_to_bool('DJANGO_DEBUG', True)
33 # DEBUG = False
34 TESTING = len(sys.argv) > 1 and sys.argv[1] == 'test'
35
36 # ALLOWED HOSTS = []
37 ALLOWED_HOSTS = ['*', '127.0.0.1', '129djangoblog.com']
38 # django 4.0新增配置
39 CSRF_TRUSTED_ORIGINS = ['https://129djangoblog.com']
40
41 # Application definition
42
43 INSTALLED_APPS = [
44     # 'django.contrib.admin',
45     'django.contrib.admin.apps.SimpleAdminConfig',
46     'django.contrib.auth',
47     'django.contrib.contenttypes',
48     'django.contrib.sessions',
49 ]

```

← → ⌂ ⌂ https://129djangoblog.com/admin/

djangoblog administration

Site administration

| ACCOUNTS |  |
|----------|--|
| 用户       | <a href="#">+ Add</a> <a href="#">Change</a> |

| ADMINISTRATION |                        |
|----------------|------------------------|
| Log entries    | <a href="#">Change</a> |

| BLOG |  |
|------|--|
| 侧边栏  | <a href="#">+ Add</a> <a href="#">Change</a> |
| 分类   | <a href="#">+ Add</a> <a href="#">Change</a> |
| 友情链接 | <a href="#">+ Add</a> <a href="#">Change</a> |
| 文章   | <a href="#">+ Add</a> <a href="#">Change</a> |
| 标签   | <a href="#">+ Add</a> <a href="#">Change</a> |
| 网站配置 | <a href="#">+ Add</a> <a href="#">Change</a> |

| COMMENTS |  |
|----------|--|
|----------|--|

Recent actions

My actions

[kern@apanel.com](#) [用户](#)

Sumber terkait:

<https://github.com/liangliangyy/DjangoBlog/blob/master/docs/README-en.md>  
<https://www.bt.cn/bbs/thread-90005-1-1.html>

Ini Translator Konten Deploy Django AAPanel Official websitenya. Perlu digaris bawahi memungkinkan error terjadi, meski sudah mengikuti panduan, Semua itu kembali lagi ke anda dan project pemrograman yang anda buat.

*Bertanya itu penting, dan rasa ingin tahu akan pengetahuan itu HARUS.*

Banyak pengembang senior dan yang pemula tetap belajar , Ilmu akan terus berkembang seiring perkembangan zaman.

*Selama waktu masih ada, belajarlah dan gapailah ilmu setinggi dan semaksimal mungkin.*

## Penutup

Terimakasih Karya ini telah di jadikan siang ini dan waktu ini pula. Sebagai rasa terimakasih saya ada sedikit bagian tentang copy projek opensource dari github

Pada Windows pastikan sudah install GIT BASH , penginstalan terletak di git-scm.com



### Git - Downloads

WEB Download the latest source release of Git for Windows, or use Git via Git to get the development version. Find GUI tools, logos, and the Git repository on GitHub.

#### Windows

winget install -id Git.Git -e -source winget.  
The current source code release is ...

#### First-Time Git Setup

The first thing you should do when you install Git is to set your user name and ...

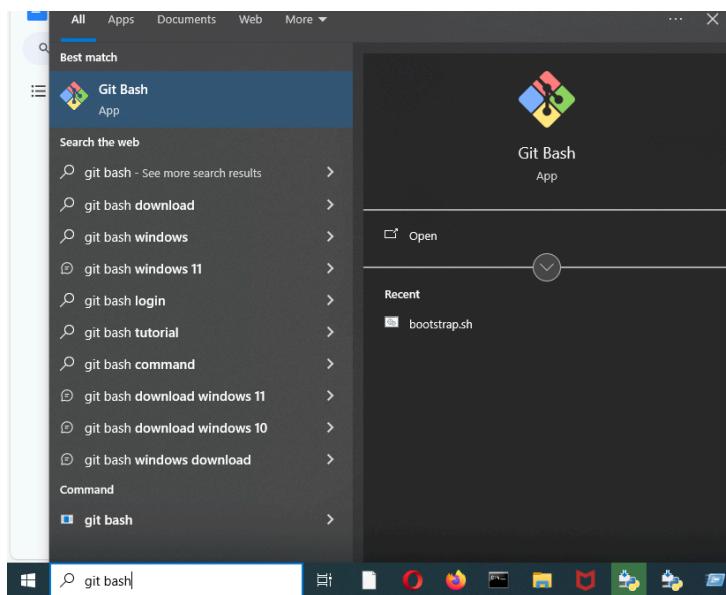
### Git - Downloads

Git comes with built-in GUI tools (git-gui, gitk), but there are several third-party ...

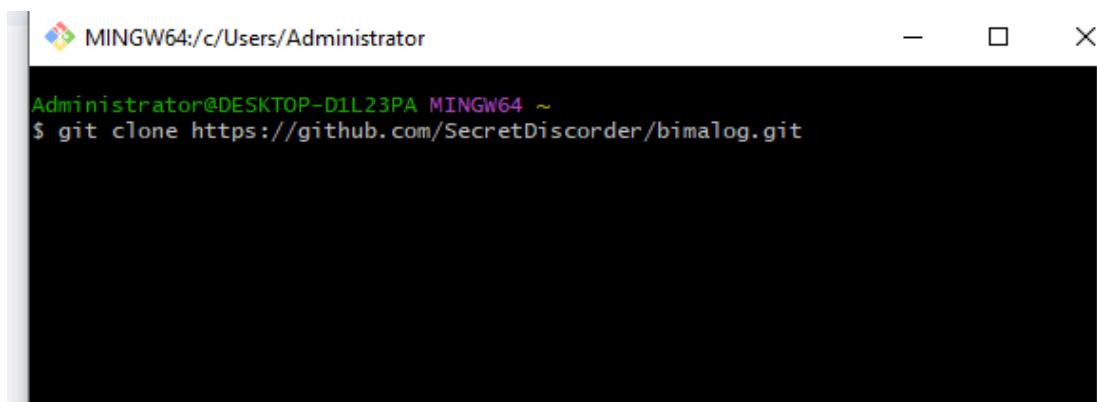
[See results only from git-scm.com](#)

Ikuti arahan penginstalan pada website Git Bash. Setelah terinstall buka di Windows Search

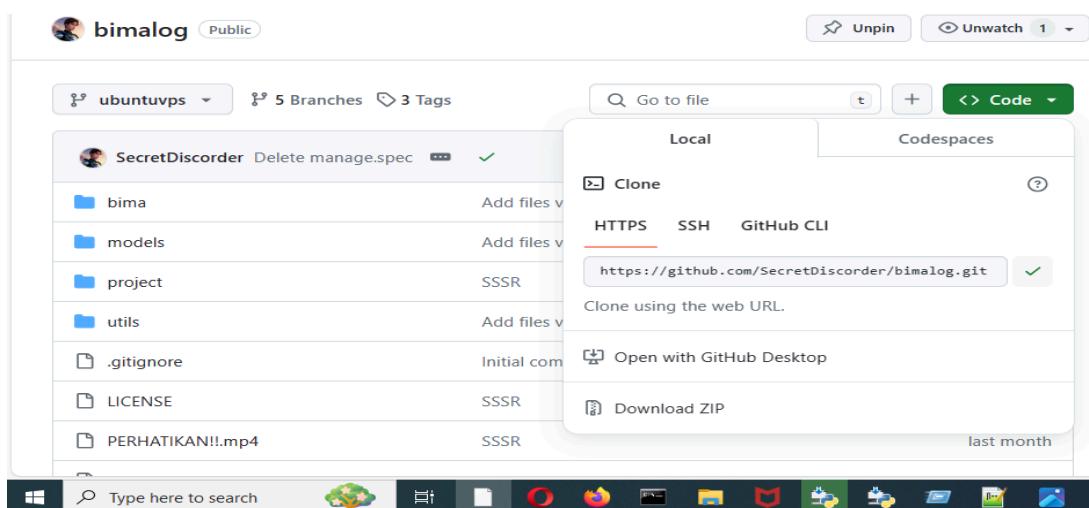
Ketik Git Bash lalu enter



git bash terminal sudah dipasang.



untuk url project opensource dapat dilihat di Layar Desktop



```
git clone <urlprojectCOPY>
```

untuk menuju direktori project ketik cd <namaproject>

```
cd bimalog
```

*Instalasi semua module pip*

```
pip install -r requirements.txt
```

*Install pula Tesseract-Ocr Bila perlu*

```
python manage.py runserver
```

```
C:\Users\Administrator\bimalog>python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

runserver
System check identified no issues (0 silenced).
April 13, 2024 - 11:33:57
Django version 5.0.4, using settings 'project.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.

[13/Apr/2024 11:35:13] "GET / HTTP/1.1" 200 7183
Not Found: /favicon.ico
[13/Apr/2024 11:35:14] "GET /favicon.ico HTTP/1.1" 404 7034
[13/Apr/2024 11:35:19] "GET /polino/ HTTP/1.1" 200 2117
[13/Apr/2024 11:35:25] "POST /polino/ HTTP/1.1" 200 2233
```

Untuk merubah Django ke Windows Apps (exe)

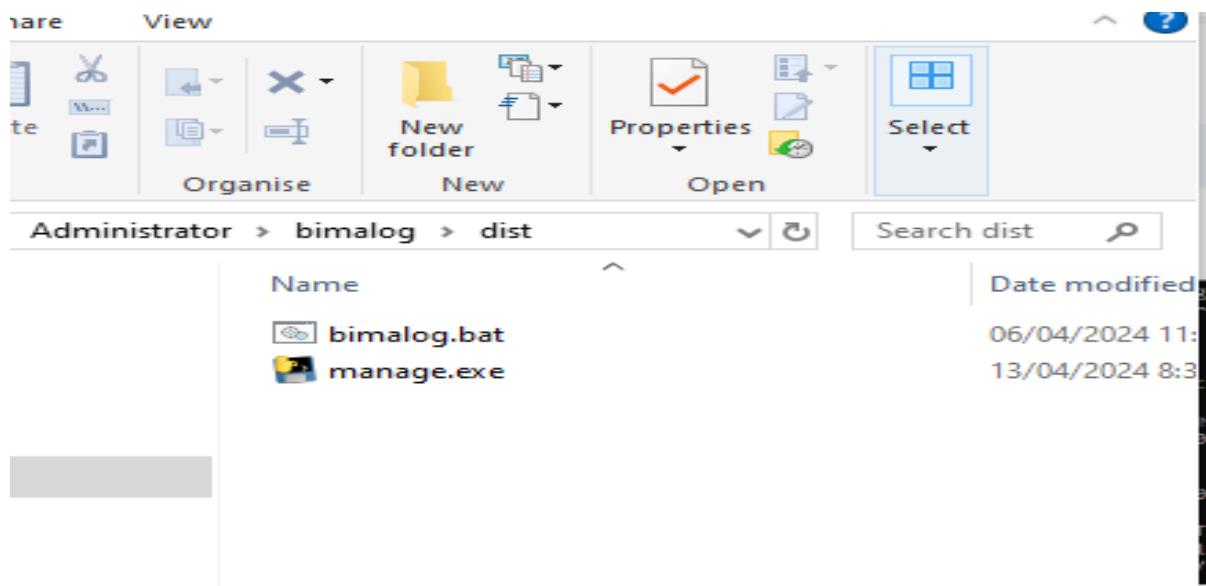
silahkan install module pyinstaller di windows.

```
pip install pyinstaller
```

setelah terinstall ketik perintah di direktori project Django

```
pyinstaller --onefile manage.py
```

Proses nya agak lama, tunggu beberapa menit sampai selesai. output keluaran di folder **dist**



Untuk menjalankan File **manage.exe** tersebut buat file baru **bimalog.bat** dan isi dengan konten ini

---

```
start http://127.0.0.1:8000
```

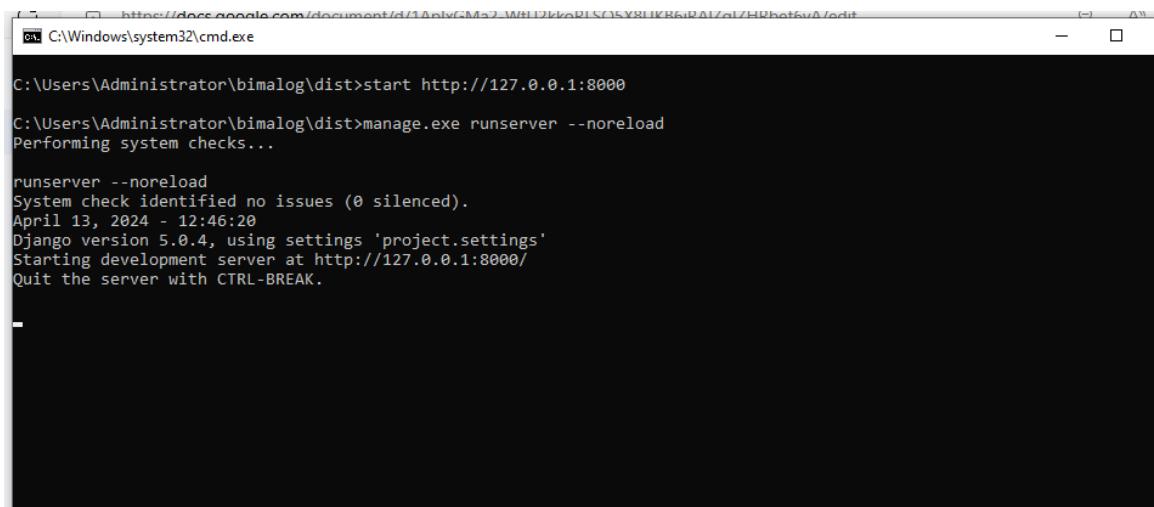
```
manage.exe runserver --noreload
```

---

save **bimalog.bat** dan klik dua kali file tersebut (Sedikit lama startup nya)

```
C:\Windows\system32\cmd.exe
C:\Users\Administrator\bimalog\dist>start http://127.0.0.1:8000
C:\Users\Administrator\bimalog\dist>manage.exe runserver --noreload
-
```

Tunggu sekitar 3-5 menit sampai muncul seperti di bawah ini

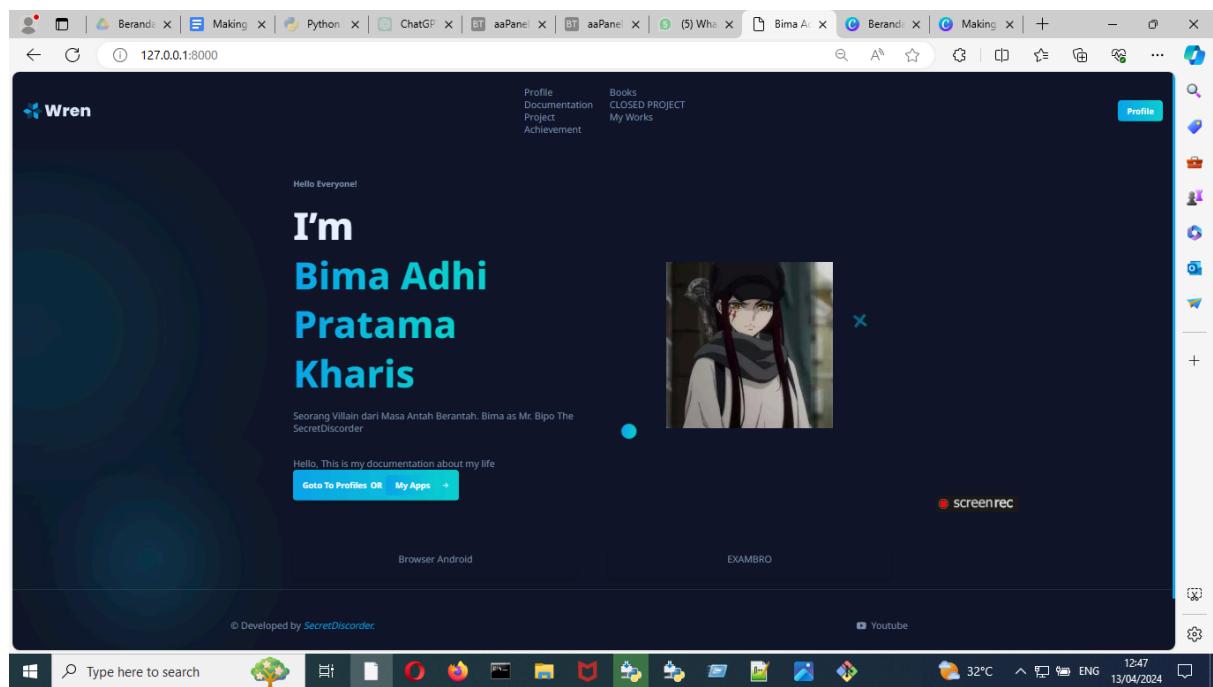


```
https://docs.google.com/document/d/1AnlvCMzJ-WH12k2S7cSYtKRaDAlz17RbtkuA/edit
C:\Windows\system32\cmd.exe

C:\Users\Administrator\bimalog\dist>start http://127.0.0.1:8000
C:\Users\Administrator\bimalog\dist>manage.exe runserver --noreload
Performing system checks...

runserver --noreload
System check identified no issues (0 silenced).
April 13, 2024 - 12:46:20
Django version 5.0.4, using settings 'project.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

buka dibrowser <http://127.0.0.1:8000>



Website Project Django telah berhasil di clone.

Demikian penutup dari buku ini, saya akhirnya dapat menyelesaikan kegiatan harian saya menulis, mengetik, dan membaca. Salam Sehat Salam Semangat

**Making Profile Website Like an Architect**

**BIMALOG V3**

SecretDiscorder / SecretDiscorder / Actions / Project / Security / Insights / Settings

1st Practical

Performing system checks...

runserver - Load

System check results: No issues found.

April 13, 2024 12:33

Django version 3.4, settings module `bimalog.settings`

Quit the server with CTRL-BREAK.

This URL is for local apps (requires internet):

`http://127.0.0.1:8000`

Semoga Beruntung

Note: This is offline but requires network for all access.

`http://127.0.0.1:8000`

Semoga Beruntung

If you encounter any other issues or want to discuss features:

Note: This is offline but requires network for all access.

Full Changelog: [bimalogv2...bimalogv3](#)

Full Changelog: [bimalogv3...bimalogv3](#)

Wren

I'm **Bima Adhi Pratama Kharis**

Seorang Villain dari Masa Antah Berantah. Bima as Mr. Bipo The SecretDiscorder

**Bima Adhi Pratama Kharis**

Hello, This is my documentation about my life

Goto To Profiles OR My Apps → screenrec

Profile

Assets 3

- list.zip 323 MB
- Source code (zip)
- Source code (tar.gz)

Project

27 minutes ago

45 minutes ago

45 minutes ago

Join discussion

screenrec

>>exit() #THANKS