



Django Fundamental



Muhammad Riza Nafis

- ❏ Apa itu Django?
- ❏ Preparations
- ❏ Web CRUD menggunakan Django
- ❏ Improvement

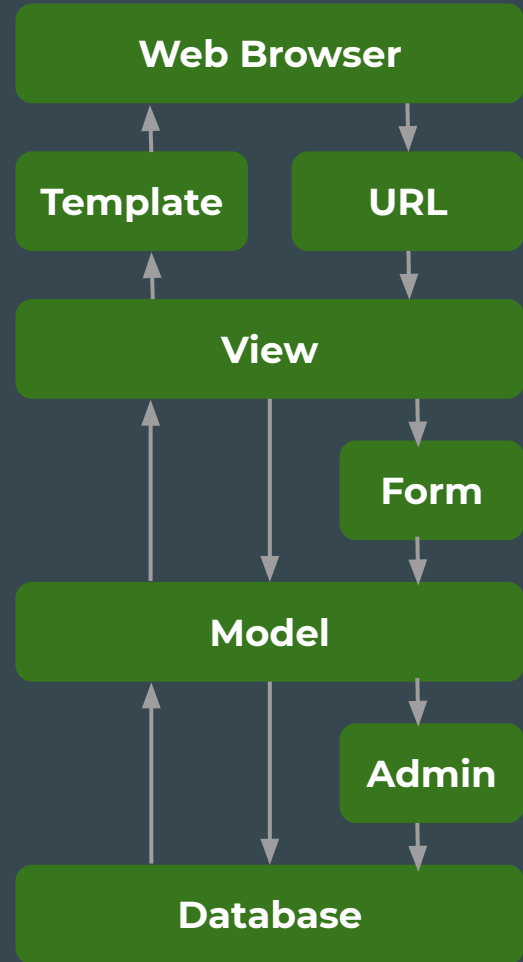


Apa itu Django?



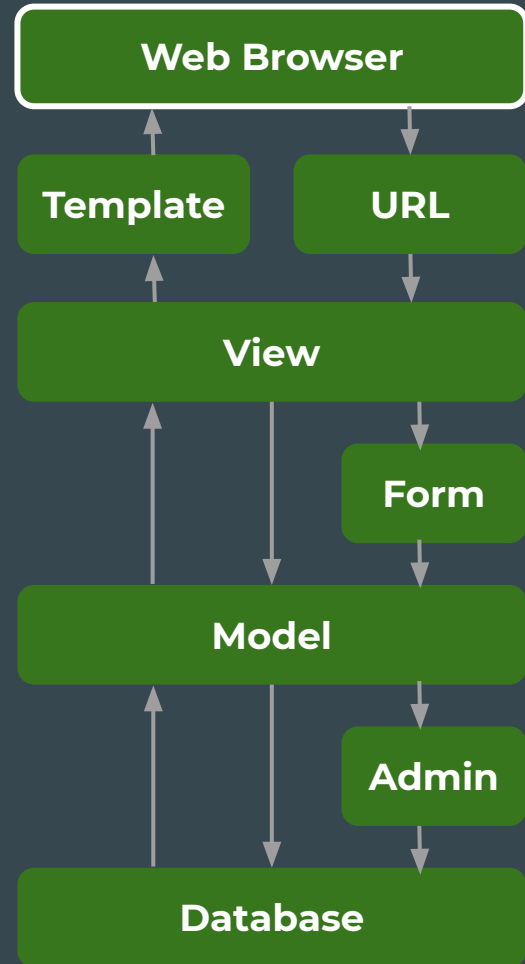
Django merupakan High-Level Web Framework menggunakan Python yang memudahkan penggunaanya untuk membuat suatu website dengan mudah, cepat, dan sedikit kode yang dituliskan.

WORKFLOW DJANGO



Web Browser

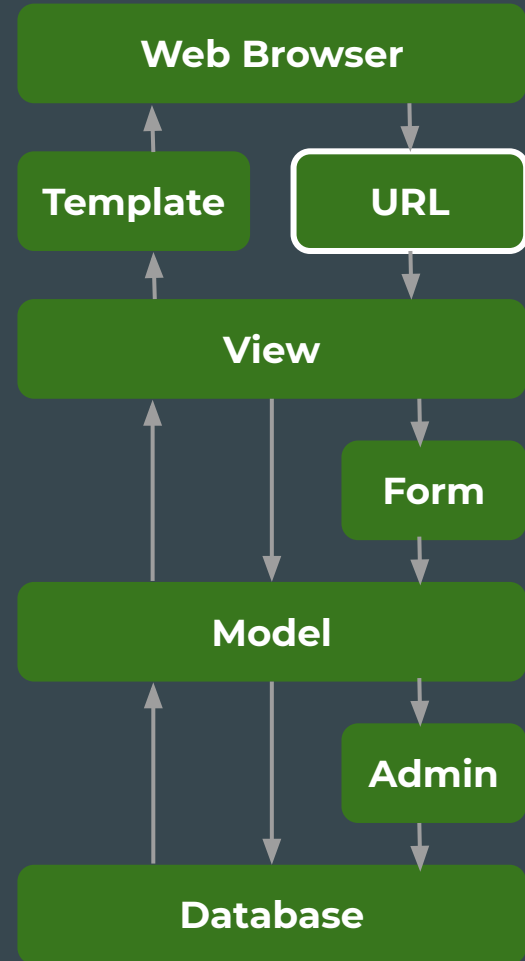
- Tampilan yang dilihat oleh User
- Tempat mengambil Input dari User
 - Input Keyboard
 - Klik Mouse



URL

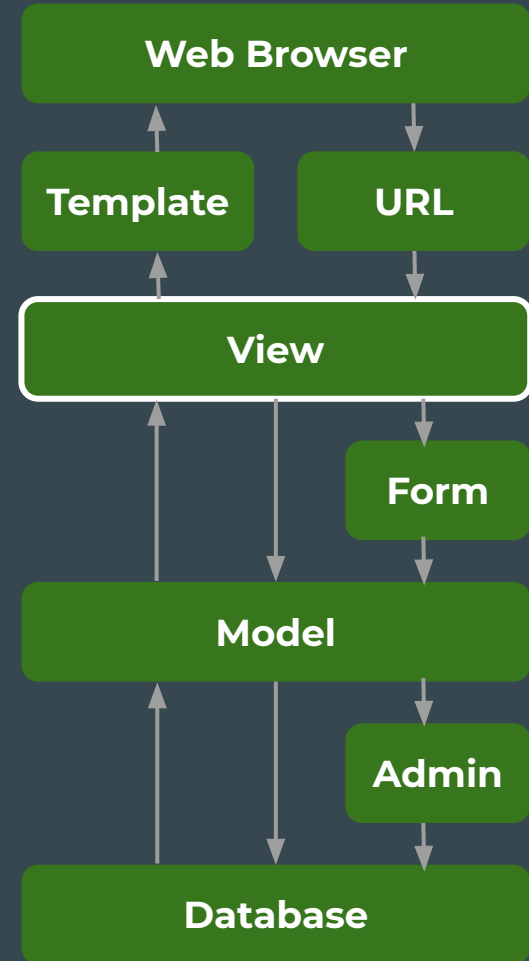
- Alamat Web
- URL memberi **mapping** pada View

Mapping dilakukan dengan fungsi yang mengarah pada suatu URL



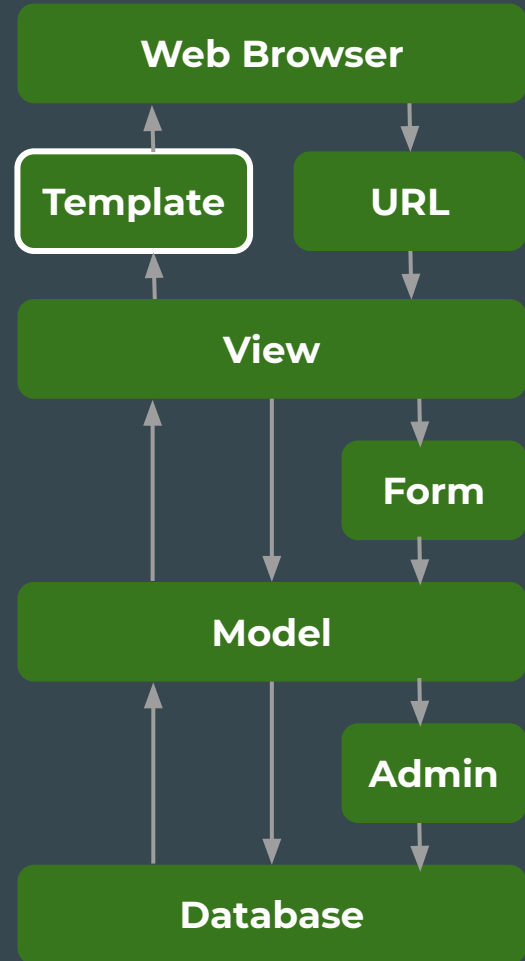
View

- Tempat semua fungsi ditulis
- Menampilkan **Template**
- Mengambil informasi dari **Model** jika diperlukan
- Menaruh data dari **Model** ke **Database** melalui **Form**



Template

- Tempat menyimpan file HTML
- Untuk menempatkan file seperti CSS dan JS, diperlukan suatu folder 'static' untuk menyimpannya



Form

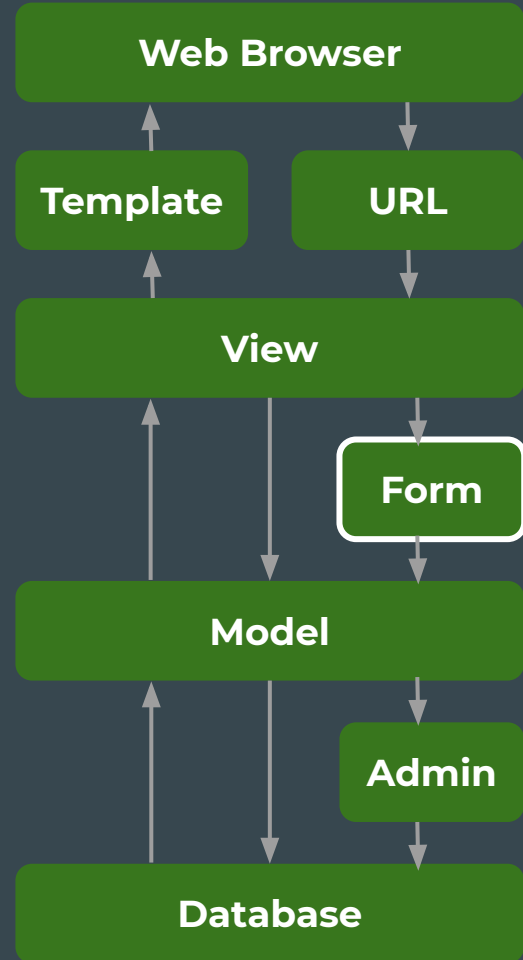
HTML Form \neq Django Form

HTML Form

- Input Element
- Submit Button
 - Dll.

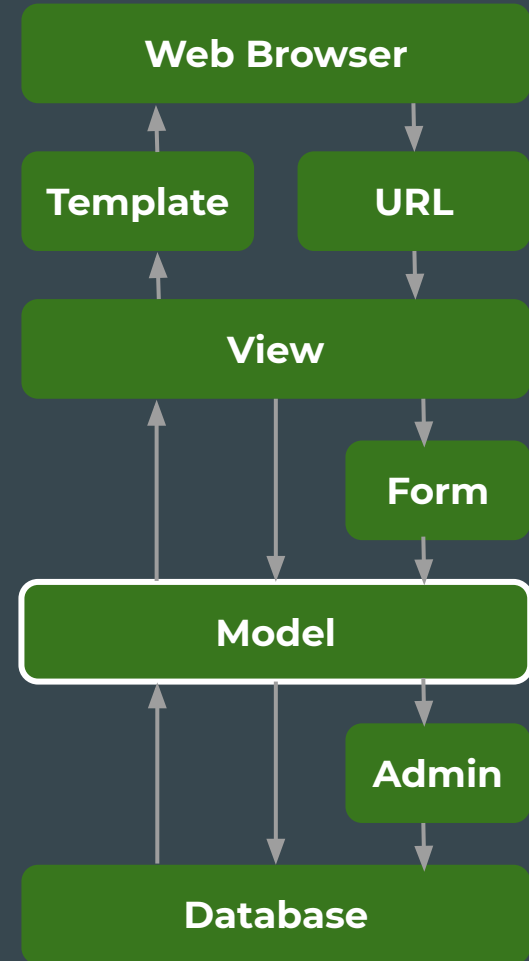
Django Form

Digunakan untuk
membantu
memasukkan data pada
Model



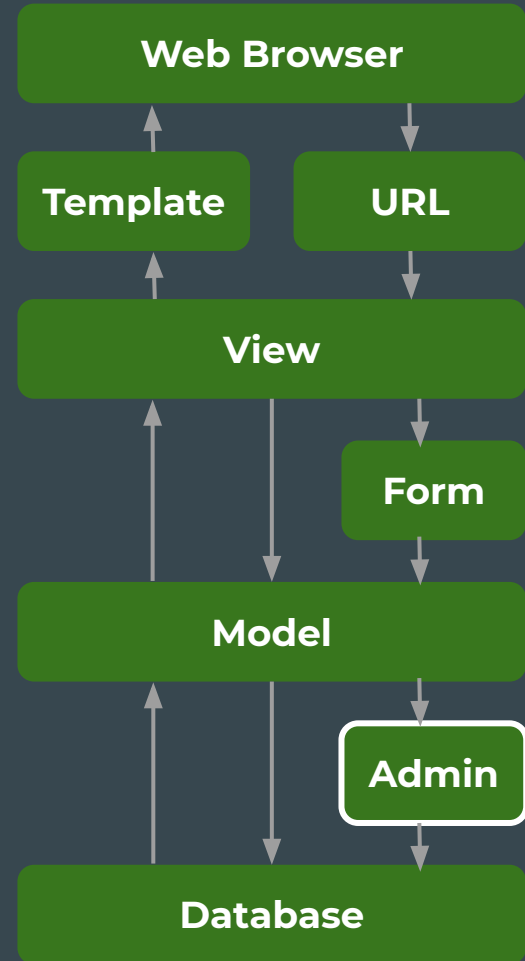
Model

- Merupakan struktur dari **Database** menggunakan ORM (Object Relational Mapping)
<https://docs.djangoproject.com/en/2.2/topics/db/models/>
- Digunakan untuk melakukan CRUD ke **Database**



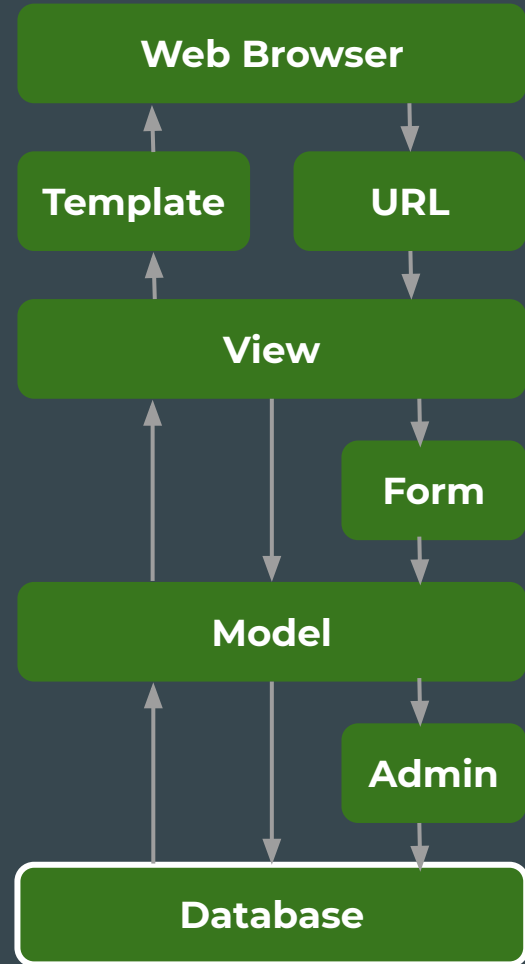
Admin

Membantu meregister dan mengorganisir data dengan **Model** ke **Database**



Database

Kumpulan data yang disimpan oleh web



Preparations (1)

Install python3 (versi python 3.6), pip, dan virtualenv

Ubuntu/Debian

```
$ sudo apt-get install python3 python3-pip python3-virtualenv
```

Windows

Download python di <http://python.org>

pip sudah terinstall pada instalasi python

```
...\> pip install virtualenv
```

Preparations (2)

Install Django (versi django 2.2)

Ubuntu/Debian

```
$ python3 -m virtualenv django -p python3
```

```
$ source django/bin/activate
```

```
(django) $ pip install django
```

Windows

```
...\> virtualenv django
```

```
...\> django\Scripts\activate
```

```
(django) ...> pip install django
```

Preparations (Khusus VPS)

Install Gunicorn

```
(django) $ pip install gunicorn
```

Install Nginx

```
# ./install-nginx.sh django_crud <host_ip>
```


Mengapa tidak menginstall database?

Database

Django menggunakan SQLite untuk Default, jika ingin menggunakan database lain nanti dapat diubah pada setting.py pada folder project

More Info :

<https://docs.djangoproject.com/en/2.2/topics/install/#database-installation>

Web CRUD menggunakan Django

Step 1 Cek Instalasi Django

```
(django) $ python -m django --version
```

```
(django) $ django-admin startproject django_crud
```

Struktur File yang terbentuk

```
django_crud/  
  django_crud/  
    __init__.py  
    settings.py  
    urls.py  
    wsgi.py  
  manage.py
```

Step 1 Cek Instalasi Django (Lanjut)

(Khusus VPS)

```
(django) $ gunicorn django_crud.wsgi --bind <server_ip>:8000 &
```

```
(django) $ python manage.py migrate
```

```
(django) $ python manage.py runserver
```



The install worked successfully! Congratulations!

You are seeing this page because `DEBUG=True` is in your settings file and you have not configured any URLs.



Django Documentation
Topics, references, & how-to's



Tutorial: A Polling App
Get started with Django



Django Community
Connect, get help, or contribute

Tampilan jika Django berhasil dijalankan

django

Django App

Django App

Django App

Django App

Django App

Django App

Django Project

Ilustrasi Program Menggunakan Django

Step 2 Membuat Aplikasi yang ingin dibuat

```
$ python manage.py startapp crud
```

Struktur File yang terbentuk

```
crud/  
  __init__.py  
  admin.py  
  apps.py  
  migrations/  
    __init__.py  
  models.py  
  tests.py  
  views.py
```


Step 2 Membuat Aplikasi yang ingin dibuat (Lanjut)

django_crud/settings.py

```
...  
# Application definition  
  
INSTALLED_APPS = [  
    'django.contrib.admin',  
    'django.contrib.auth',  
    'django.contrib.contenttypes',  
    'django.contrib.sessions',  
    'django.contrib.messages',  
    'django.contrib.staticfiles',  
    'crud.apps.CrudConfig',  
]  
...
```

Step 2 Membuat Aplikasi yang ingin dibuat (Lanjut)

crud/views.py

```
from django.shortcuts import render

# Create your views here.
def index(request):
    return render(request, 'index.html')
```

Step 2 Membuat Aplikasi yang ingin dibuat (Lanjut)

crud/templates/index.html

```
<!DOCTYPE html>
<html>
  <head>
    <title>CRUD APP</title>
  </head>
  <body>
    <h1>Hello, World</h1>
  </body>
</html>
```

Step 2 Membuat Aplikasi yang ingin dibuat (Lanjut)

crud/urls.py

```
from django.urls import path

from . import views

urlpatterns = [
    path('', views.index, name='index'),
]
```

Step 2 Membuat Aplikasi yang ingin dibuat (Lanjut)

django_crud/urls.py

```
from django.contrib import admin
from django.urls import include, path

urlpatterns = [
    path('', include('crud.urls')),
    path('admin/', admin.site.urls),
]
```

Untuk referensi lebih lanjut tentang *path()*

<https://docs.djangoproject.com/en/2.2/ref/urls/#django.urls.path>

Step 3 Membuat Database

crud/models.py

```
from django.db import models

# Create your models here.
class Mahasiswa(models.Model):
    nrp = models.CharField(max_length=10)
    nama = models.CharField(max_length=50)
    umur = models.IntegerField()
```

Contoh penggunaan yang lebih complex

<https://docs.djangoproject.com/en/2.2/intro/tutorial02/>

Referensi lebih lanjut

<https://docs.djangoproject.com/en/2.2/ref/models/instances/#django.db.models.Model>

Step 3 Membuat Database (Lanjut)

```
# Menambahkan model pada Database  
$ python manage.py makemigrations  
  
$ python manage.py migrate
```


Step 4 Menambahkan Halaman Website

- Index : menampilkan seluruh data yang telah dimasukkan dalam database
- Insert : memasukkan data ke dalam database
- Update : mengubah data yang ada pada database
- Delete : menghapus data yang ada pada database

Step 4 Menambahkan Halaman Website

crud/views.py

```
from django.shortcuts import render, redirect
from .models import Mahasiswa

# Create your views here.
def index(request):
    mahasiswa_list = Mahasiswa.objects.all()
    context = {'mahasiswa_list': mahasiswa_list}
    return render(request, 'index.html', context)
```

Membuat File index.html
(check di layar)

Step 4 Menambahkan Halaman Website (Lanjut)

crud/views.py

```
...  
def insert(request):  
    if(request.method == 'POST'):  
        mhs = Mahasiswa(  
            nrp=request.POST['nrp'],  
            nama=request.POST['nama'],  
            umur=request.POST['umur'])  
        mhs.save()  
        return redirect(index)  
    else:  
        return render(request, 'insert.html')
```

**Membuat File insert.html
(check di layar)**

Step 4 Menambahkan Halaman Website (Lanjut)

crud/views.py

```
...  
def update(request, nrp):  
    if(request.method == 'POST'):  
        mhs = Mahasiswa.objects.get(nrp=nrp)  
        mhs.nrp = request.POST['nrp']  
        mhs.nama = request.POST['nama']  
        mhs.umur = request.POST['umur']  
        mhs.save()  
        return redirect(index)  
    else:  
        mhs = Mahasiswa.objects.get(nrp=nrp)  
        context = {'detail_mhs': mhs}  
        return render(request, 'update.html', context)  
  
def delete(request, nrp):  
    Mahasiswa.objects.filter(nrp=nrp).delete()  
    return redirect(index)
```

**Membuat File update.html
(check di layar)**

Step 4 Menambahkan Halaman Website (Lanjut)

crud/urls.py

```
from django.urls import path

from . import views

urlpatterns = [
    path('', views.index, name='index'),
    path('insert/', views.insert, name='insert'),
    path('update/<int:nrp>/', views.update, name='update'),
    path('delete/<int:nrp>/', views.delete, name='delete')
]
```