

RANGKUMAN
BELAJAR DJANGO 2.2



**UNIVERSITAS
TEKNOLOGI
SUMBAWA**

MATA KULIAH : PEMROGRAMAN MOBILE

DIAMPUH OLEH : HERFANDI, A.Md.,S.Kom.,M.Kom

DISUSUN OLEH:

ABIGAIL PERKASA (20.01.013.064)

PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS REKAYASA SISTEM

UNIVERSITAS TEKNOLOGI SUMBAWA

2022

1. INTRODUCTION

Django adalah web framework gratis (free open source) yang mengikuti pola arsitektur MVT dan ditulis dengan bahasa python. Web framework merupakan kerangka kerja yang membantu membuat pengembangan situs web secara mudah dan cepat. Di dalamnya sudah tersedia fungsi-fungsi siap pakai untuk mengakses ke database, template dan session.

MVT (Model, View, Template) merupakan 3 komponen utama dalam membangun sebuah aplikasi dimana model merupakan definitif dari database atau representasi dari tabel-tabel di database. Views berisi fungsi-fungsi untuk mengatur proses http request dan juga respon. Sedangkan template berhubungan dengan tampilan atau user interface yang isinya file html.

Jadi, Django adalah web framework yang digunakan untuk membantu membuat situs web. Beberapa website yang menggunakan teknologi Django diantaranya:

- codesaya.com merupakan website belajar pemrograman interaktif.
- kelaskita.com
- website gojekindonesia

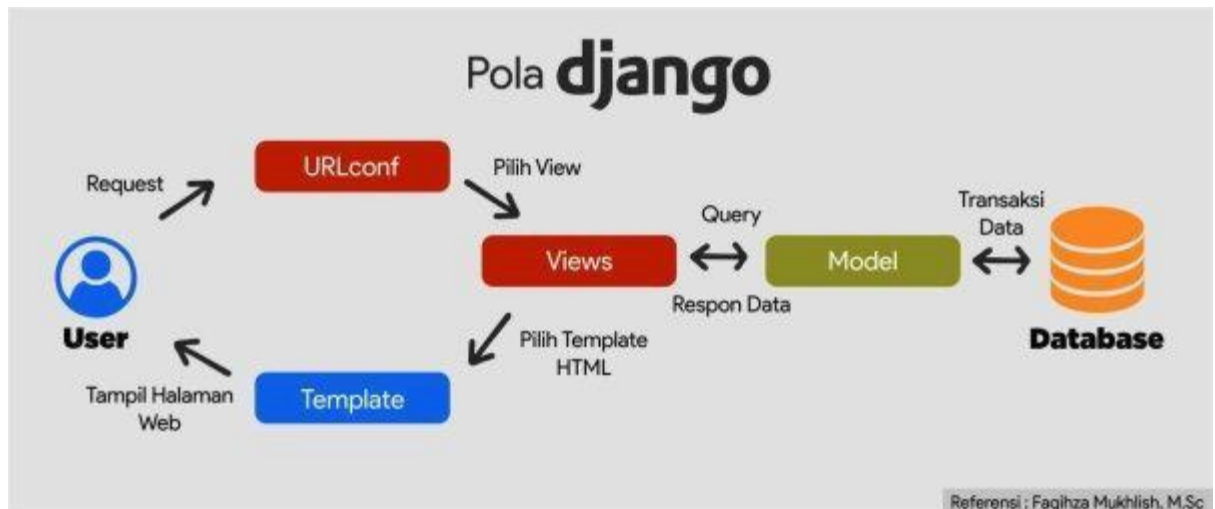
Beberapa skill dasar yang dibutuhkan untuk belajar Django yaitu:

- Dasar-dasar bahasa python
- OOP (Object Oriented Programming) atau Pemrograman Berorientasi Objek
- Paham query SQL
- HTML

2. WORKFLOW

Alur kerja Django dimulai dari client yang melakukan request, misalkan url `zulhilmi.id/profil` lalu akan di cek oleh URL Config apakah request profil ada di aplikasi Django? Jika ada maka akan memilih view yang sesuai lalu dari view akan dicek kembali apakah request dari client membutuhkan data dari database? Jika dibutuhkan maka view akan melakukan queryset ke model lalu model akan mengambil data dari database, direspon ke view lalu data akan dilempar template lalu akhirnya akan direspon oleh client.

Jika request tidak membutuhkan database, maka dari view akan langsung menuju template dan direspon oleh client.



3. KEBUTUHAN BELAJAR DJANGO

Tools yang dibutuhkan untuk memulai Django:

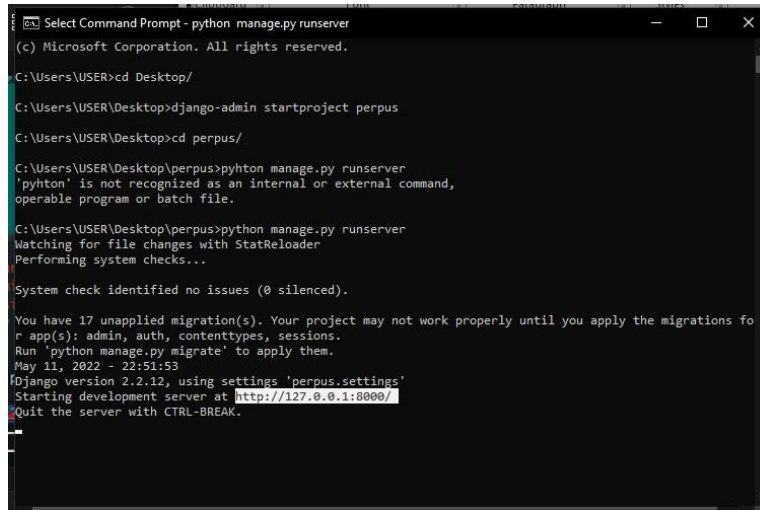
- Cmd/ Terminal digunakan untuk menginstall Django, membuat proyek, menjalankan server dan lain-lain.
- Text editor digunakan untuk menulis kode program. Bisa menggunakan visual studio code atau text editor lainnya.
- Web browser. Bisa menggunakan google chrome.
- Python (cek versi melalui cmd dengan mengetik python maka akan muncul versi python yang sudah diinstall)
- Django (install django melalui cmd dengan mengetik `pip install django==2.2.12` (2.2.12 adalah versi django yang akan diinstall))
- Pada cmd ketik `pip list` untuk melihat apakah django sudah terpasang.

4. MEMBUAT PROJECT

Pada pembahasan ini dan seterusnya akan membahas mengenai project perpustakaan.

- Membuat proyek melalui cmd dengan cara ketik `cd Desktop/` lalu enter. Setelahnya ketik `django-admin startproject perpustakaan` lalu enter. Setelah berhasil, folder perpustakaan akan tampil pada desktop dengan isi terdiri dari beberapa file.
- Pada file `manage.py` berisikan file perintah untuk berinteraksi dengan django.

- Selanjutnya folder perpus yang terdiri dari file `__init__.py` yang memberitahukan bahwa perpus merupakan sebuah package. Ada file `settings.py` berisi file pengaturan/konfigurasi proyek, database, template dan lain-lain. File `urls.py` berisikan pola-pola url. Dan file `wsgi.py` adalah file yang digunakan untuk development proyek yang melibatkan web server kompatibel dengan wsgi.
- Cara menjalankan proyek yaitu masuk dengan folder perpus dengan perintah `cd perpus/` lalu jalankan perintah `python manage.py runserver`



```

Select Command Prompt - python manage.py runserver
(c) Microsoft Corporation. All rights reserved.

C:\Users\USER>cd Desktop/

C:\Users\USER\Desktop>django-admin startproject perpus

C:\Users\USER\Desktop>cd perpus/

C:\Users\USER\Desktop\perpus>python manage.py runserver
'python' is not recognized as an internal or external command,
operable program or batch file.

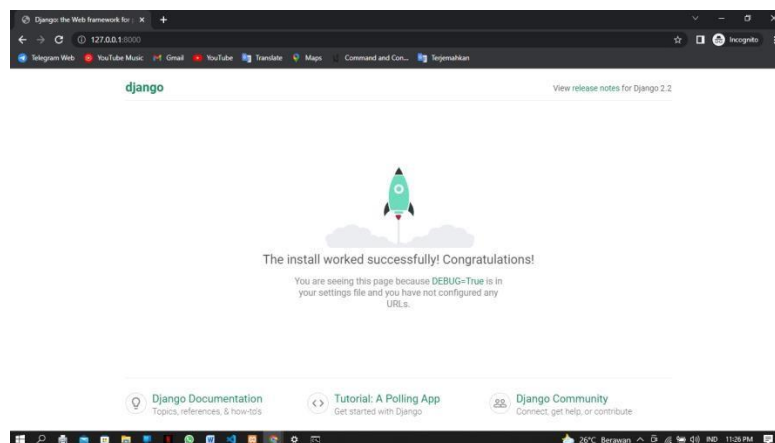
C:\Users\USER\Desktop\perpus>python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).

You have 17 unapplied migration(s). Your project may not work properly until you apply the migrations fo
r app(s): admin, auth, contenttypes, sessions.
Run 'python manage.py migrate' to apply them.
May 11, 2022 - 22:51:53
Django version 2.2.12, using settings 'perpus.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.

```

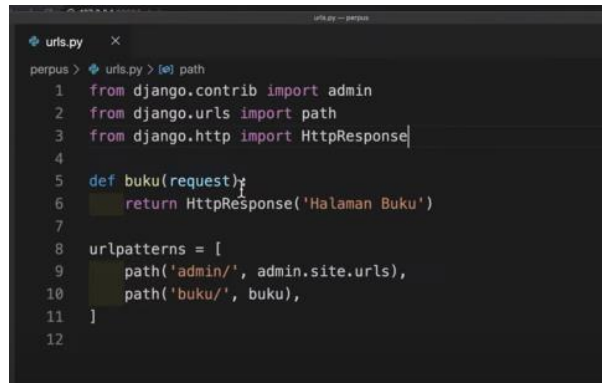
- Copy alamat url lalu buka web server dan paste alamat url. Jika berhasil akan muncul tampilan seperti dibawah



5. BASIC ROUTING

Pada aplikasi perpus yang akan dibuat, client akan membuat request lalu di cek oleh URL conf, lalu view akan dipilih dan view akan direspon oleh client.

- Buka folder perpustakaan pada text editor, dalam hal ini saya menggunakan visual studio code
- Untuk melakukan request dari client, buatlah URL baru seperti gambar berikut

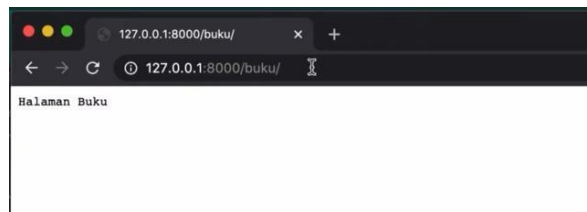


```

1  from django.contrib import admin
2  from django.urls import path
3  from django.http import HttpResponse
4
5  def buku(request):
6      return HttpResponse('Halaman Buku')
7
8  urlpatterns = [
9      path('admin/', admin.site.urls),
10     path('buku/', buku),
11 ]
12

```

- Jika sudah, maka request akan terpenuhi



6. MEMBUAT APPS

Apps adalah sebuah aplikasi pada Django yang mempunyai Model Database, View, Template, dan URLConf. Setiap Project di Django mempunyai Apps dan bisa lebih dari satu Apps.

Ada 2 cara untuk membuat apps:

- Pada cmd membuat perintah `django-admin startapp perpustakaan`
- Pada cmd menjalankan file `manage.py` dengan perintah `python manage.py startapp perpustakaan`

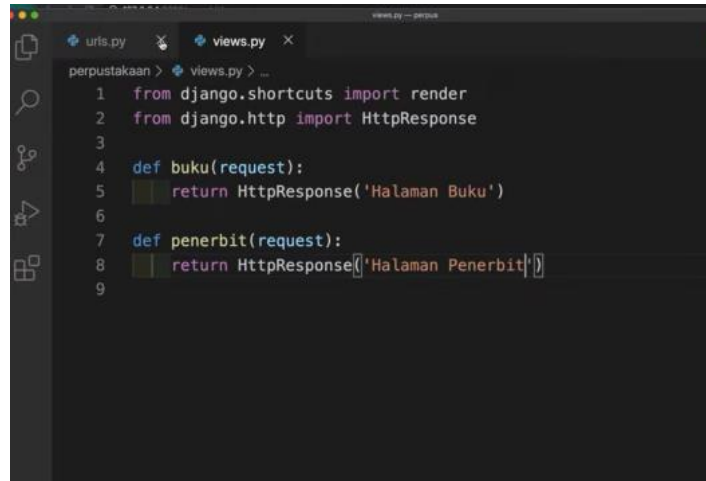
7. MEMBUAT VIEWS

Views menggunakan alur Client, URLConf, View dan memberikan Response kepada Client. Views diciptakan untuk memenuhi Request dari Client.

8. TEMPLATES

- Alur yang digunakan mulai berkembang. Dimulai dari Client, URLConf, View, dan Template.

- Pada bagian Settings.py Line 58. Tambahkan 'DIRS' : ['template']
- Buat Folder templates satu Level dengan settings, lalu buat buku.html
- Membuat HTML sederhana
- Lalu ubah Views.py menjadi seperti berikut



```

1  from django.shortcuts import render
2  from django.http import HttpResponse
3
4  def buku(request):
5      return HttpResponse('Halaman Buku')
6
7  def penerbit(request):
8      return HttpResponse('Halaman Penerbit')
9

```

9. DJANGO TEMPLATE LANGUAGE

Dibagi menjadi 3:

- Substitusi Variabel : Untuk menampilkan Variabel ke Template yang dilemparkan dari Views
- Filter : Untuk memodifikasi Variabel yang akan ditampilkan
- Tags : Untuk melakukan control Flow seperti Looping, If Else, Include, dll untuk mengambil file eksternal

10. TEMPLATE EXTENDING

- Membuat Base Template / Template dasar untuk semua Halaman.
- Template utama yang isinya Base.html adalah file html utama yang akan menampung konten – konten dari Template Apps
- Didalam Apps akan dibuat folder template yang isinya hanya bagian – bagian konten
- Bagian Konten akan diextend / dimasukkan kedalam Base.html

11. STATIC FILE

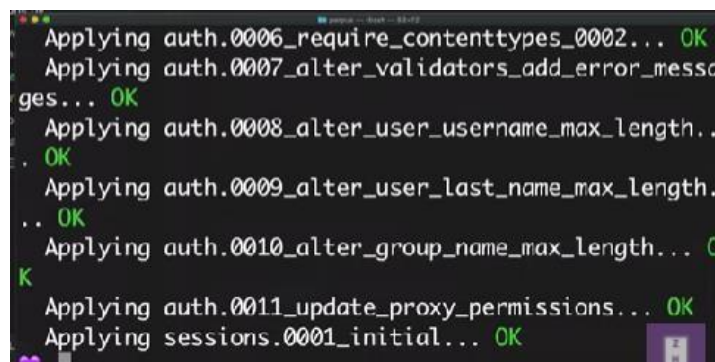
Static File adalah kumpulan File CSS, Java Script, dan gambar. Static File ini digunakan untuk mempercantik / memperindah tampilan Aplikasi yang dibuat dan memberikan pengalaman kenyamanan saat Aplikasi digunakan.

12. SET UP BOOTSTRAP

- Memindahkan file – file CSS dan Java Script kedalam Folder static yang telah di Set Up.
- Download Bootstrap dan JQuery yang akan digunakan. -
Selanjutnya panggil file – file tersebut di Base.html

13. SET UP DATABASE PADA DJANGO

- Secara default Django menggunakan DBMS Sqlite 3 dengan nama ‘db.sqlite3’. Ini bisa di Rename sesuai dengan keinginan kita seperti ‘perpustakaan.sqlite3’
- Pada saat pertama kali melakukan Runserver, ini akan mengcreate database saja tidak termasuk tabel – tabel nya database
- Selanjutnya dilakukan Migrasi dilakukan untuk menyebarkan / menginisialisasi tabeltabel kedalam db.sqlite3 terhadap database project yang akan dibuat.
- Jika berhasil melakukan Migrasi, maka akan tampil seperti dibawah ini



```
Applying auth.0006_require_contenttypes_0002... OK
Applying auth.0007_alter_validators_add_error_mes... 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 sessions.0001_initial... OK
```

14. SET UP DATABASE: MYSQL

- Melakukan Konfigurasi MySql sebagai DBMS untuk project Django yang akan dibuat.
- Tahap ini bersifat Opsional, bisa tidak digunakan jika ingin menggunakan sqlite3 sebagai DBMS
- Jika ingin menggunakan MySql sebagai DBMS, selanjutnya install MySql Installer

- Jika berhasil diinstall, selanjutnya buka MySQL Command Line lalu create Database -
Konfigurasi Database ke settings.py dan akan tampil seperti gambar berikut

```

Running type help, or (h) for help. type (c) to clear the buffer.
Applyin mysql> create database perpustakaan;
Applyin Query OK, 1 row affected (0.01 sec)
Applyin mysql> use perpustakaan;
Applyin Database changed
Applyin mysql> show tables;
Applyin +-----+
Applyin | Tables in perpustakaan |
Applyin +-----+
Applyin | auth_group              |
Applyin | auth_group_permissions  |
Applyin | auth_permission         |
Applyin | auth_user               |
Applyin | auth_user_groups        |
Applyin | auth_user_user_permissions |
Applyin | django_admin_log        |
Applyin | django_content_type     |
Applyin | django_migrations       |

```

15. MODELS

Models merupakan definitive dari database atau representasi tabel pada database. Dengan menggunakan models ini, kita tidak perlu lagi menggunakan Query SQL untuk membuat tabel di database.

Ketika melakukan Migrasi pada model buku, maka Django akan melakukan Create Tabel Buku sesuai dengan field – field yang ada pada model buku ini. Maka jadilah Tabel pada Database yaitu Tabel Buku.

16. MODELS: FOREIGN KEY

- Foreign Key digunakan untuk membuat Relasi antar tabel dalam database Relational.
- Kelompok_id pada gambar dibawah merupakan Foreign Key yang nanti nya akan diisi oleh id tabel kelompok

17. DJANGO ADMIN

Django Admin merupakan salah satu fitur yang powerfull yang ada pada Django. Dikatakan Powerfull karena dapat melakukan CRUD sederhana untuk mengelola data pada model yang kita buat.

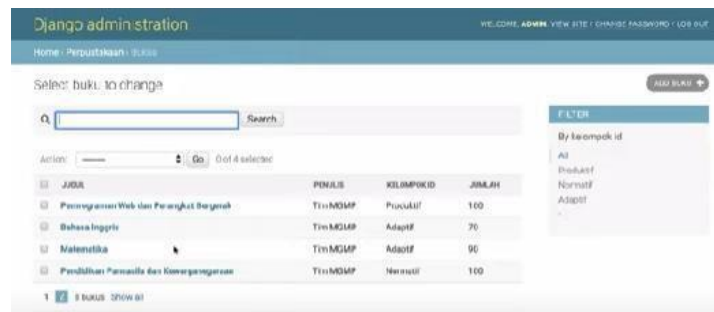
Model – Model yang kita buat akan ditambahkan kedalam Django Admin. Django admin ini bersifat Private karena diperlukan Login terlebih dahulu untuk dapat mengakses nya. Django admin Sederhana namun sangat membantu.

Saat kita membuat projek ada URL yang sudah dibuatkan oleh djangonya yaitu admin, URL inilah untuk mengakses django admin, apabila ingin akses langsung /admin tampilannya langsung django admin log in. Lalu buat akun buka terminal buat username nya dengan nama admin, lalu email address, dan password 2 kali. Jalankan kembali servernya jika sudah membuat akun. Akun tersebut di masukkan ke dalam log in.

Apabila sudah masuk ke dalam akun selanjutnya kita menampilkan model yang sudah kita buat di dalam django admin. Lalu save, model-model yang kita sudah buat sudah masuk sudah terdaftar. Perpustakaan adalah nama X nya, Buku dan kelompok adalah model nya. Power fullnya terdapat di tombol add kalau di klik akan menampilkan ada sebuah form untuk menambahkan data dalam Buku, kita hanya membuat kelas model saja di datakan di dalam django admin. Mengisi judul,penulis,penerbit,jumlah,dan kelompok id dengan mengisi nama dan keterangan dan save. Ini udah termasuk ke database. Kita buka lagi di model kelompok yang sudah terdapat 3 kelompok seperti adaptif,normatif,dan produktif.

18. DJANGO ADMIN: MODEL ADMIN

Melakukan custom sederhana terhadap tampilan data buku, fill-fill apa saja yang ditampilkan sebagai informasi seperti judul,penulis,penerbit dan lain-lain. Kita akan menampilkan kotak pencarian dan filter kelompok buku.



19. ORM (Object Relational Mapping)

ORM (object relational mapping) merupakan teknik yang digunakan dalam pemrograman untuk menggunakan basis data relasional sebagai penyimpanan data dengan bentuk objek. Perlu diketahui django menggunakan teknik ini untuk menggunakan database relasional. Agar kode python yang ditulis tidak campur aduk dengan query sql. Jadi, ORM ini bertugas sebagai penghubung aplikasi yang dibuat menggunakan database relasional.

20. FORMS: MODEL FORM

- Form biasanya ditulis dengan HTML. Namun di Django tidak wajib menggunakan HTML, bisa saja menggunakan Models
- Input Type harus disesuaikan dengan Type data pada tabel database.
- Selanjutnya Membuat file baru di perpustakaan dengan nama form. Lalu membuat viewsnya.

21. FORMS: WIDGED

- Pada tahap ini, kita menambahkan atribut dengan menggunakan widget pada output yang telah dibuat sebelumnya.
- Didalam forms.py ditambahkan widgets, text Input merupakan typenya.
- Tambahkan atribut kelas didalamnya.

22. CRUD: MENAMBAH DATA

- Masuk ke dalam queryset. Disini kita menambah data, menampilkan, mengubah dan menghapus data Dari data yang sudah dibuat.
- Selanjutnya kita akan menyimpannya ke dalam database. Prosesnya ada di views yang nantinya akan mengecek apakah data yang di submit oleh user sudah benar atau sudah valid.
- Apabila sudah maka akan di simpan ke database.
- Lalu buka text editor, buka file views. Fungsi dari csrf sendiri adalah untuk mngamankan form yang dibuat.

23. CRUD: MENAMPILKAN DATA

- Menampilkan data buku yang sudah dibuat, seperti menampilkan nilai 90/100 dan
- menampilkan kelompok seperti produktif maka hasilnya pasti error karena kelompok_id merupakan type integer.
- Namun Apabila menambah __nama, maka akan bisa mengeluarkan hasilnya.
- Penggunaan queryset tidak sepanjang query. Untuk menampilkan limit pada queryset cukup ditambah diujungnya [:3], sedangkan dalam query limit 3.

24. CRUD : Mengubah Data

- Tahap Mengubah data ini perannya sangat penting karena saat kita salah dalam melakukan input data, maka kita harus mengubahnya atau mengeditnya dengan menggunakan fitur update atau fitur ubah data ini. Ini lebih efektif dibandingkan dengan hapus, lalu input ulang
- Ubah data sebenarnya sama dengan form menambah data. Bedanya form ubah data sudah terisi oleh data yang ingin kita ubah, misalnya seperti ingin mengubah judul yang dari Bhs.Indonesia menjadi IPA, lalu klik simpan dan selesai.

25. CRUD : Hapus Data

- Hapus data juga penting dalam aplikasi, karena apabila data tersebut tidak lagi digunakan kita dapat menghapus nya. Karena Fungsi fitur ini sendiri adalah untuk menghapus data yang sudah tidak digunakan oleh user.
- Sebelum menggunakan action/aksi hapus, terlebih dahulu membuat konfirmasi yang berisi apakah yakin ingin menghapus atau tidak untuk menghindari ketidaksengajaan user dalam menghapus data.

26. AUTHENTICATION : LOGIN & LOGOUT

Authentication merupakan proses verifikasi/validasi identitas user yang terdaftar sebelum mengakses system. Dengan ini kita jadi bisa membatasi user mana saja yang boleh menambahkan data, mengubah data, dan menghapus data. Jadi tidak sembarang user mengakses halaman-halaman tersebut . django ada system authentication yang akan kita gunakan yang bernama class LoginView untuk membuat namanya sama seperti membuat log in django admin. Berarti saat ingin membuat yang baru bisa membuat di terminal.

27. MENGAKSES USERNAME DI TEMPLATE

Mengakses user yang sedang aktif untuk ditampilkan ke dalam template.

Biasanya saat kita login ke dalam aplikasi, terdapat informasi user atau siapa saja yang saat ini sedang login atau yang sedang menggunakan aplikasi. Dan Apabila kita log out dan tidak terdapat informasi maka kita bisa menggunakan attributes `is_authenticated`.

28. SIGN UP

Membuat form sign up, agar user bisa login ke aplikasi perpustakaan yang dibuat. Tahap ini berhubungan dengan django admin karena di dalamnya terdapat username django adminnya. Jika lihat di dokumentasi resmi Django, ada sebuah user form yang bernama `UserCreationForm` untuk membuat user baru.

29. UPLOAD FILE

Membuat tools upload file, seperti mengupload cover buku. Kita akan menambahkan file baru pada model buku yang bernama `cover` sehingga di dalam tambah buku nanti akan muncul yang baru seperti `cover` atau muncul di ubah buku. Di buku kita akan menambahkan kolom baru yaitu `cover` sebagai informasi buku yang kita masukkan.

30. EXPORT FILE

Membuat export file. Export file ini biasa digunakan untuk membuat report atau laporan ke dalam bentuk file excel, pdf, atau file lainnya. Dalam aplikasi biasanya terdapat fitur laporan yang harus di export data yang terdapat dalam database ke dalam file excel contohnya. Cara menggunakan export ini bisa menggunakan `djangoimportexport`.

31. VIRTUAL ENVIRONMENT

Membahas tentang virtual environment atau lingkungan virtual, dalam kasus ini lingkungan virtual berguna untuk mengisolasi proyek yang kita buat. Setiap kita membuat proyek didalam lingkungan virtual ini, proyek tersebut akan terisolasi dari direktori system dan memiliki paket python sendiri yang terinstal di lingkungan virtual tersebut. Lingkungan system operasi lebih besar yang didalam nya sudah ada django versi 2.2.12, pillow versi 7.1.2, dan `django-import-export` versi 2.2.0. kita juga sudah membuat aplikasi perpustakaan di lingkungan system tersebut dengan menggunakan django versi 2.2.12. apabila kita membuat proyek di dalam lingkungan virtual environment 1 dan virtual environment 2. Di dalam virtual env 1 terdapat aplikasi perpustakaan yang dibangun dengan django 1.11 sedangkan di dalam virtual env 2 terdapat juga aplikasi yang dibangun dengan django 3.0.8 mysqlclient.

32. PERSIAPAN DEPLOYMENT

Deployment adalah kegiatan untuk menyebarkan aplikasi yang telah dikerjakan oleh developer (pengembang). Artinya kita mempublish projek atau aplikasi yang kita buat untuk dilihat oleh orang lain. Kita juga melewati siklus pembuatan aplikasi yaitu :

- Development : mengerjakan projek aplikasi kita atau proses pembuatan, pengembangan . kita di awal-awal akan membuat form,model,view,tampilan itu adalah tahapan development atau pengembangan.
- Testing : setelah dilakukan pembuatan otomatis kita akan melakukan uji coba agar mengetahui kekurangan dari kegiatan tersebut. Mengecek apakah sudah sesuai atau apakah ada bug. Testing ini juga dilakukan saat production.
- Production : aplikasi kita yang sudah di buat sudah di deploy / sudah bisa di publish / sudah bisa digunakan oleh orang lain. Bisa juga user melakukan testing pada saat tahap ini. Alur kerja nya dari client yang melakukan request langsung ditangani oleh web server lalu ke wsgi lalu terakhir ke django atau perputaran yang sudah kita buat yaitu aplikasi perpustakaan tersebut. Yang perlu disiapkan adalah app (github,writelab,perpus), terminal, server, domain.

33. DEPLOYMENT

Pada tahap ini, kita akan mendeploymen aplikasi yang telah dibuat ke server. Dengan awalan mengcoding di local lalu mengupload ke github lalu di clon atau ditarik ke computer server untuk dideploy disana. Saat aplikasi kita dalam mode production disetting dan debug masih nilainya masih dalam true jadi kalau misalkan salah menuju ke halaman pasti akan eror . Saat kita dalam mode production kita tidak boleh dibiasakan mengubah tampilan saat mode dalam production, apabila terjadi eror akan berakibat fatal maka itu kalau kita ingin mengembangkan aplikasi yang dalam production kita hanya perlu mengembangkannya di local dan kita masuk ke tahap development atau tahap pengembangan lagi seperti mengubah tampilan atau menambah fitur yang diinginkan.