



Assignment Cover Letter

(Individual Work)

Student Information:

1.	Surname Saleh Kadarisman	Given Names Amartya	Student ID Number 2201841210
----	---------------------------------------	-------------------------------	--

Course Code	: COMP6502	Course Name	: Introduction to Programming
--------------------	------------	--------------------	-------------------------------

Class	: L1BC	Name of Lecturer(s)	: 1. Monica Hidajat
--------------	--------	----------------------------	---------------------

Major	: CS
--------------	------

Title of Assignment (if any)	: Django Blog Web Application
--	-------------------------------

Type of Assignment	: Final Project
---------------------------	-----------------

Submission Pattern

Due Date	: 20-11-2018	Submission Date	: 20-11-2018
-----------------	--------------	------------------------	--------------

The assignment should meet the below requirements.

1. Assignment (hard copy) is required to be submitted on clean paper, and (soft copy) as per lecturer's instructions.
2. Soft copy assignment also requires the signed (hardcopy) submission of this form, which automatically validates the softcopy submission.
3. The above information is complete and legible.
4. Compiled pages are firmly stapled.
5. Assignment has been copied (soft copy and hard copy) for each student ahead of the submission.

Plagiarism/Cheating

BiNus International seriously regards all forms of plagiarism, cheating and collusion as academic offenses which may result in severe penalties, including loss/drop of marks, course/class discontinuity and other possible penalties executed by the university. Please refer to the related course syllabus for further information.

Declaration of Originality

By signing this assignment, I understand, accept and consent to BiNus International terms and policy on plagiarism. Herewith I declare that the work contained in this assignment is my own work and has not been submitted for the use of assessment in another course or class, except where this has been notified and accepted in advance.

Signature of Student:

1. Amartya Saleh Kadarisman

(Name of Student)

Django Blog Web Application

Name : Amartya Saleh Kadarisman

ID : 2201841210

1. Description

This particular web application that I made was a simple web application that has several features

The web application features :

- Register / Create Account
- Login
- Post (contains: title, content, and image)
- Change User Profile (Username, email, and profile picture)
- Auto resize profile picture (reduces the image size)
- Auto resize posted Image (doesn't break the original image ratio)
- Delete user's own post

Reasons why I made this project :

- I was curious on how Django works
- I want to improve my coding skill specifically in web-development
- It could be fun

2. Design

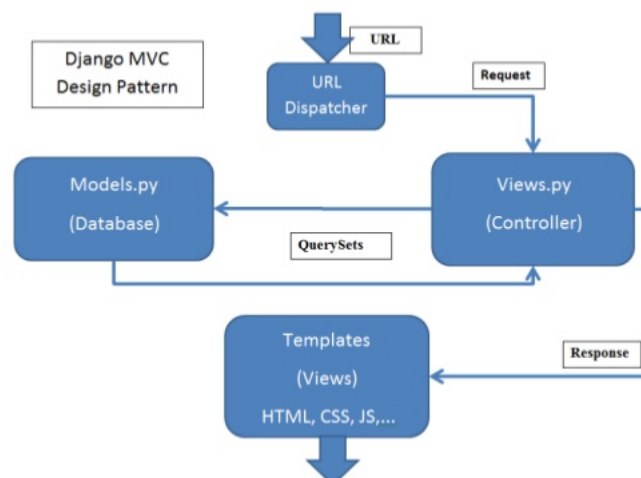
Basic Django Flow

23/08/2012

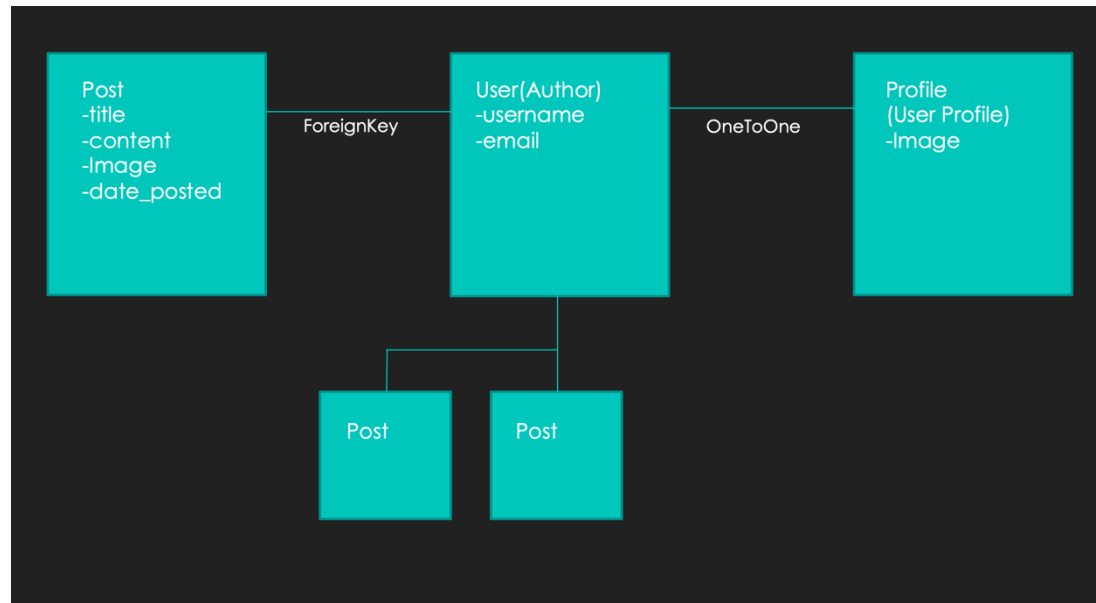
Python - Django Training Course 2012 @HCMUT

28

How things in Django works?



Basic Database Model



3. Discussion

The things that were used and included in my project was Django, Bootstrap, Crispy Forms, and Pillow

Django is the main framework that is used in this project. Django is a back-end web-development framework that simplifies the process of building a good back-end website system.

Bootstrap is also a crucial framework for this project as it is used for the front-end part of the website. With bootstrap I was able to make the website look good in a short time because of the wide-variety of bootstrap library. I didn't need to create my own css from scratch, I was able to modify just several things that doesn't match or look good.

Crispy Forms is also a very convenient tool that I used in this project. With crispy forms I was able to let it tidy the forms that are displayed on the website without doing anything complicated.

Pillow is also very useful in this project because with pillow I was able to shrink image resolution and most importantly size, so images doesn't take so many space in the database.

How it works:

This website works like any other Django Websites that are operating these days. First the user enters the url that he/she wishes to visit, then the url will find the view that is assigned to the specific url.

Then the views will take or store any information from or to the database depending on what type of request (post or get request) is being sent by the user to the website. After accumulating **or** storing data from **or** to the database, the views will render the assigned template and display the information gathered on it.

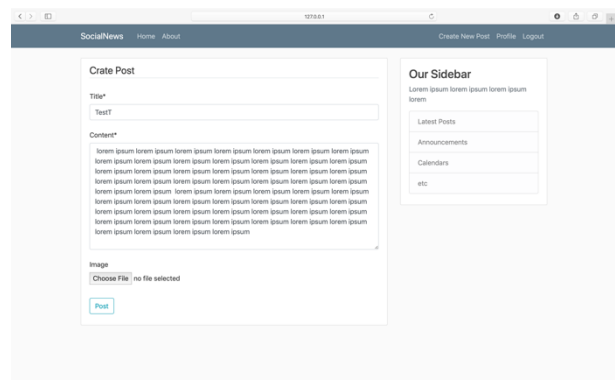
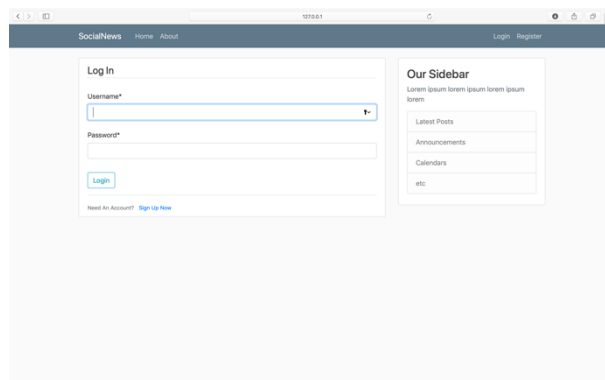
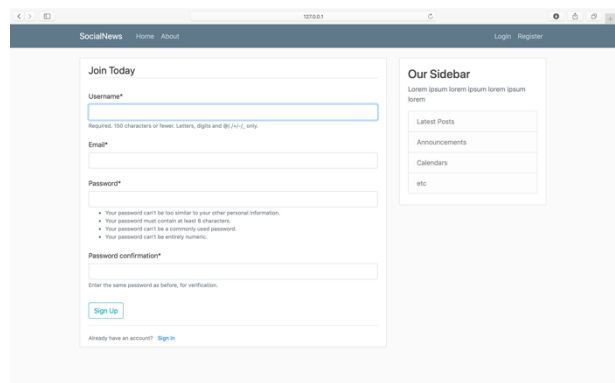
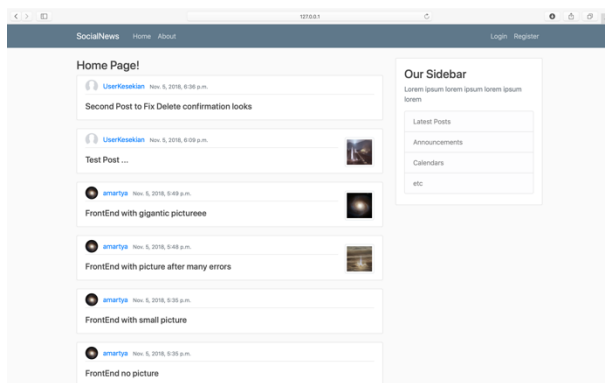
All part of the website works like this on and on again, the only difference is the user might click on a button that is routed to a url of the website (which is basically the same as entering a url but much more convenient), and the type of requests (post or get request). For clearer understanding check the second part (Design) again.

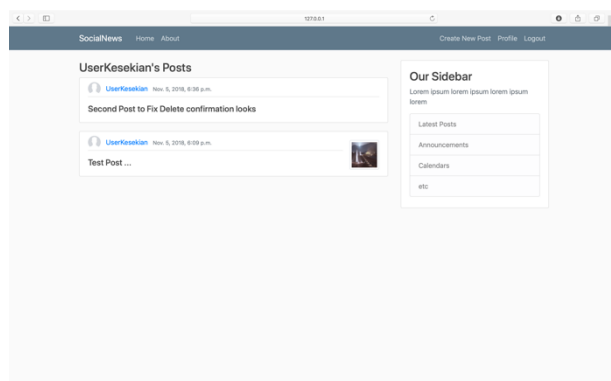
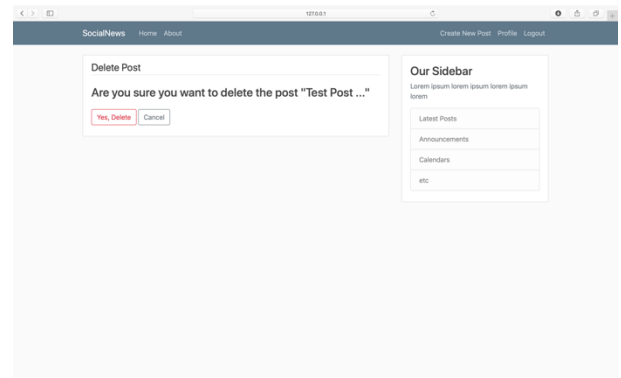
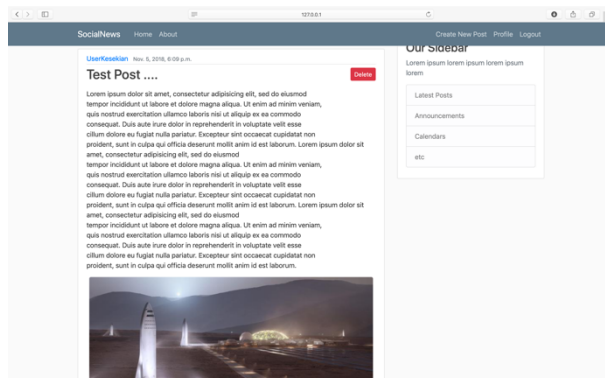
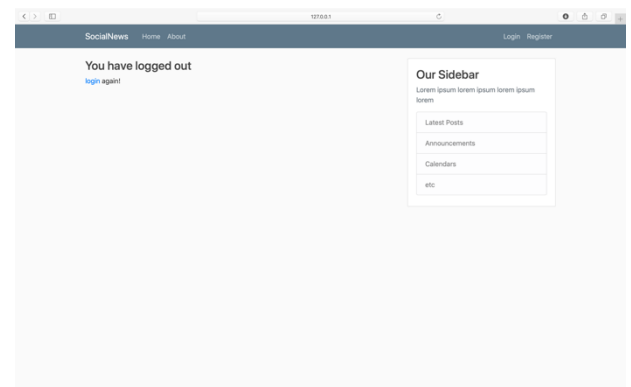
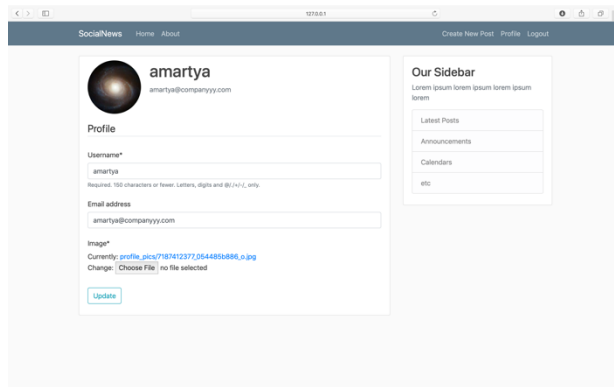
Class Explanation:

There is a lot of class in this project. But classes are used in two parts of this project. The first one is in the models.py file which is part of the database. A class represents a single database table in models.py.

The second part is in views.py which is used to simplify and make it possible to reuse and inherit the views from Django's library. With class based views I don't have to write everything from scratch, I can use something that is already available and modify it based on my website needs.

4. Evidence





5. Resources

- <https://www.pinterest.com/pin/522980575473232728/>
- <https://opensource.com/life/15/2/resize-images-python>
- <https://stackoverflow.com/questions/5213025/how-to-check-imagefield-is-empty>
- <https://www.youtube.com/watch?v=UmljXZlypDc&list=PL-osiE80TeTtoQCKZ03TU5fNfx2UY6U4p>
- <https://getbootstrap.com/docs/4.0/getting-started/introduction/> and a lot of bootstrap documentation
- <https://tutorial.djangogirls.org/en/>
- <https://www.youtube.com/watch?v=ZDa-Z5JzLYM&list=PL-osiE80TeTsqhluOqKhwlXsIBIdSeYtc>