

Blog

Zofia Dobrowolska

17 kwietnia 2023

1 Introduction

I've fucked up I've fucked up pretty badly. One tinny issue ('/media/' was stubbornly refusing to obey me) caused me to mess up the whole code and the version control so much that I couldn't do anything about it anymore. In a frenzy of anger I deleted everything and am starting again. This time I'm gonna document each and every step of this journey, so (maybe, just tinny maybe) I'll figure out where I went wrong. So, without further ado let's dive into this biggie mess.

2 Initiation – django project

First things first: initiation Poure yourself a big cup of strong eather tea of coffee (but STRONG). Then open comandline (for lazy-ass people like me : **Windows + R**). and type:

```
~ cd $<<$project directory path$>>$
~ venv\Scripts\activate
~ pip install django
~ django-admin startproject $<<$name$>>$
~ python manage.py runserver
  { this comand starts the project in browser}
~ django-admin startapp $<<$name$>>$
```

```
~ python manage.py migrate
  {this one wi will use a lot}

~ python manage.py makemigrations
  {and this one too}

~ python manage.py createsuperuser
  {so you can login to admin website}
```

Et voilà! The monster was created, now we will just decorate it with horns, sharp claws and crooked teeth. Jump on.

3 Handling venv

Storing a Django project and its virtual environment objectively consumes too much space. Yet, for deployment, it's crucial to know the requirements for the project to work. We can generate *requirements.txt* that will be the scribe of our misdeeds (I mean installations). Some more typing, but now in git bash:

```
~ This one for saving installation configs to txt file:
```

```
env1/bin/python -m pip freeze > requirements.txt
```

```
~ and another one for installing whatever is written in the file for given
environment.
```

```
env2/bin/python -m pip install -r requirements.txt
```

This step should be repeated throughout the development. Only then we can include *.venv* in our *.gitignore* file. Yeah, about that...

4 .gitignore

The *.gitignore* file is a textish file that tells Git which files or folders to ignore in a project. A local *.gitignore* file is usually placed in the root directory of a project. So, create a text file under name ".gitignore" (yessss... with a "." at the beginning), then copy/paste the lines from the footnote¹. Now final touches. By force delete the ".txt" file extension, et voilà, once again.

¹[Django .gitignore file content](#)

5 Extend URLs

For each app in the project to have separate *url* addresses it's needed to include them in the project *urls.py* file. As the name of the action we will use function *include(path)*, which takes the path to app's url file as (str) argument.

6 Project *settings.py*

In the *settings.py* file we will fuck up a lot of things durring the process, but lets focus on the things (I think) I know how to set. First, we need to add ourselves (our device) to acces debugging part of website (only if

```
DEBUG = False
```

) –

```
ALLOWED_HOSTS = ["127.0.0.1"]
```

. How the hell I knew the adress, I have no idea, will try to figure it out in the "free" time.

Then increase the memory for one post from default 2,5 MB to 20 MB, since we want to store multiple images.

For the app to even work we need to add it to

```
INSTALLED_APPS = ['my_app']
```

. Also id installed package do not work as expected add it to this peace of code.

The for our files to cooperate with us we need to add their adress to specific places, namely:

```
TEMPLATES = [{'DIRS': ['templates']}]
```

```
STATIC_URL = 'static/'
```

```
STATIC_ROOT = '/static/collection/'
```

```
STATICFILES_DIRS = ['static', 'img']
```

```
MEDIA_URL = 'media/'
```

```
MEDIA_ROOT = 'my_blog/media'
```

7 Rich editor – ckeditor

For rich text fields in django where's a couple of options, I (dunno why) have choosen the *ckeditor*. To install this beauty type in terminal:

```
pip instal django-ckeditor
```

In *settings.py* add such lines:

```
import ckeditor
INSTALLED_APPS = ['ckeditor', 'ckeditor_uploader']

CKEDITOR_UPLOAD_PATH = 'content/ckeditor/'
CKEDITOR_CONFIGS = {
    'default': {
        # Computer modern: the reason is obvious; Monospace: font designed for programming soft;
        # Courier New: for movie scripts and stuff; the others who knows why
        'font_names': 'Computer Modern; Monospace; Courier New; Roboto; Arial; Verdana; Times New Roman',
        'skin': 'moono',
        # 'skin': 'office2013',
        'toolbar_Basic': [
            ['Source', '-', 'Bold', 'Italic']
        ],
        'toolbar_YourCustomToolbarConfig': [
            {'name': 'math', 'items': ['Mathjax', ]},
            {'name': 'document', 'items': ['Source', '-', 'Save', 'NewPage', 'Preview', 'Print', '-', 'Templates']},
            {'name': 'clipboard', 'items': ['Cut', 'Copy', 'Paste', 'PasteText', 'PasteFromWord', '-', 'Undo', 'Redo']},
            {'name': 'editing', 'items': ['Find', 'Replace', '-', 'SelectAll']},
            {'name': 'forms',
             'items': ['Form', 'Checkbox', 'Radio', 'TextField', 'Textarea', 'Select', 'Button', 'ImageButton',
                       'HiddenField']},
            '/',
            {'name': 'basicstyles',
             'items': ['Bold', 'Italic', 'Underline', 'Strike', 'Subscript', 'Superscript', '-', 'RemoveFormat']},
            {'name': 'paragraph',
             'items': ['NumberedList', 'BulletedList', '-', 'Outdent', 'Indent', '-', 'Blockquote', 'CreateDiv', '-',
                       'JustifyLeft', 'JustifyCenter', 'JustifyRight', 'JustifyBlock', '-', 'BidiLtr', 'BidiRtl',
                       'Language']},
            {'name': 'links', 'items': ['Link', 'Unlink', 'Anchor']},
            {'name': 'insert',
             'items': ['Image', 'Flash', 'Table', 'HorizontalRule', 'Smiley', 'SpecialChar', 'PageBreak', 'Iframe']},
            '/',
            {'name': 'styles', 'items': ['Styles', 'Format', 'Font', 'FontSize']},
            {'name': 'colors', 'items': ['TextColor', 'BGColor']},
            {'name': 'tools', 'items': ['Maximize', 'ShowBlocks']},
            {'name': 'about', 'items': ['About']},
            '/', # put this to force next toolbar on new line
            {'name': 'yourcustomtools', 'items': [
                # put the name of your editor.ui.addButton here
                'Preview',
                'Maximize',

            ]},
        ],
        'toolbar': 'YourCustomToolbarConfig; CodeSnippet', # put selected toolbar config here
        'mathJaxLib': '//cdn.mathjax.org/mathjax/latest/MathJax.js?config=TeX-AMS_HTML',
        # 'toolbarGroups': [{ 'name': 'document', 'groups': [ 'mode', 'document', 'doctools' ] }],
        # 'height': 291,
        # 'width': '100%',
        # 'filebrowserWindowHeight': 725,
        # 'filebrowserWindowWidth': 940,
        # 'toolbarCanCollapse': True,
        # 'mathJaxLib': '//cdn.mathjax.org/mathjax/2.2-latest/MathJax.js?config=TeX-AMS_HTML',
        'tabSpaces': 4,
        'extraPlugins': ','.join([
            'codesnippet',
            'mathjax',
            'uploadimage', # the upload image feature
            # your extra plugins here
            'div',
            'autolink',
            'autoembed',
            'embedsemantic',
            'autogrow',
            # 'devtools',
            'widget',
            'lineutils',
            'clipboard',
        ])
```

```
        'dialog',  
        'dialogui',  
        'elementspath'  
    ),  
}
```

8 Installations

1. Pillow;
2. django-ceditor;
3. pygments;
4. rest_framework;