S. Xiao Fernandez Marin.
Group 1392

REPORT

Django is a framework that works in Python and based in the MVC architecture.
In Django you work with a project that has applications plugged to it. The application needs to be synced with the database that we are using (in this case PostgreSQL).

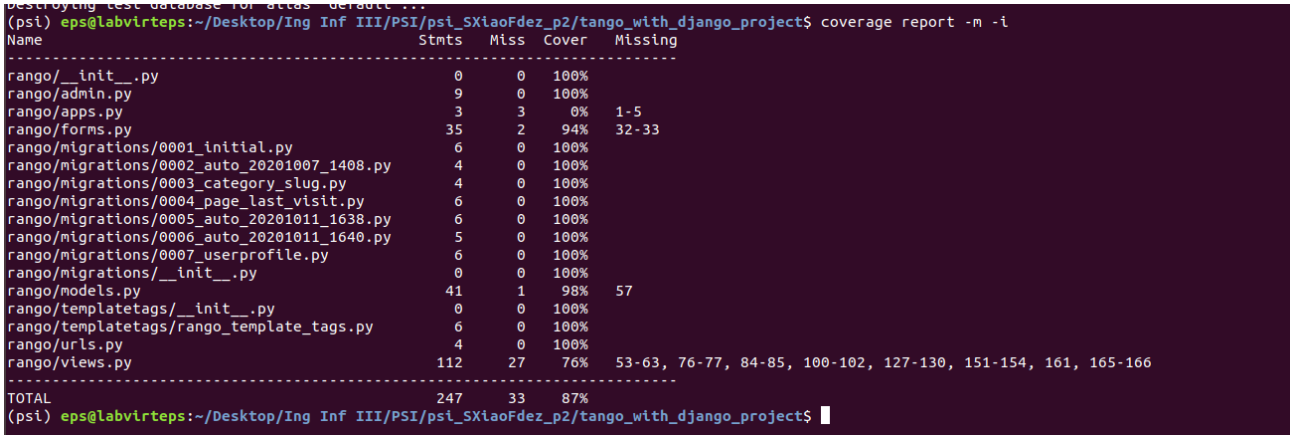
The application URL gets linked to the project one and that is the one that is linked to the web server.

The **model** is designed in the *models.py* file, inside the app directory. This file has classes that are describing the data on the database.

The **view** part is showed to the user through HTML templates.

The functions that the HTML files use to to access data and build the HTML are in the *views.py* file. This file is used by the **controller** to link the request of the user (using the *urls.py* file and analysing which function of the *views.py* should call) with the HTML files.

COVERAGE



```
Destroying test database for alias 'default'...
(psi) eps@labvirteps:~/Desktop/Ing Inf III/PSI/psi_SXiaoFdez_p2/tango_with_django_project$ coverage report -m -i
Name                                             Stmts   Miss  Cover   Missing
------------------------------------------------------------------------------
rango/__init__.py                                    0      0   100%
rango/admin.py                                       9      0   100%
rango/apps.py                                        3      3     0%   1-5
rango/forms.py                                      35      2    94%   32-33
rango/migrations/0001_initial.py                     6      0   100%
rango/migrations/0002_auto_20201007_1408.py          4      0   100%
rango/migrations/0003_category_slug.py               4      0   100%
rango/migrations/0004_page_last_visit.py             6      0   100%
rango/migrations/0005_auto_20201011_1638.py          6      0   100%
rango/migrations/0006_auto_20201011_1640.py          5      0   100%
rango/migrations/0007_userprofile.py                 6      0   100%
rango/migrations/__init__.py                         0      0   100%
rango/models.py                                     41      1    98%   57
rango/templatetags/__init__.py                       0      0   100%
rango/templatetags/rango_template_tags.py            6      0   100%
rango/urls.py                                        4      0   100%
rango/views.py                                     112     27    76%   53-63, 76-77, 84-85, 100-102, 127-130, 151-154, 161, 165-166
------------------------------------------------------------------------------
TOTAL                                              247     33    87%
(psi) eps@labvirteps:~/Desktop/Ing Inf III/PSI/psi_SXiaoFdez_p2/tango_with_django_project$
```

QUESTIONS EXERCISE 1 WEEK 3

• What would happen if you don’t enter in a category name on the add category form?
If you do not enter a category name on the add category form when clicking "Add New Category" it says to you that you have to add a name to the field.

• What happens when you try to add a category that already exists?
It does not allow you because the name has to be unique.

• What happens when you visit a category that does not exist?
It says that the category does not exists and shows you a link to go back to the index.

S. Xiao Fernandez Marin.
Group 1392

• In the section above where we implemented our ModelForm classes, we repeated the max_length values for fields that we had previously defined in the models chapter.
This is bad practice as we are repeating ourselves! How can you refactor your code so that you are not repeating the max_length values?
We could pass as an argument the "max_length" field from the Category class to the CategoryForm class or make "max_length" visible to the class that inherit from the Category field.