FULL STACK DEVELOPER NANODEGREE PROJECT P1 MOVIE TRAILER WEBSITE

BY THEODOR F PURCARU

CONTENT

PREFACE

This documentation is written to explain the first project required in my full stack developer nanodegree. The system built here allows for a very stripped out view of a website which keeps track of media such as movies and shows with ways for users to get info on their favorite movies. It also has an administration panel for admins to be able to do all CRUD operations needed to run such a web app.

The application uses the django framework and can be easily deployed on any virtual machine by following the given instructions.
During this project, I have learned how to properly use the django framework, it's templating system and python in general.

TECHNOLOGY

Django is a free open source web app framework, made with Python, which follows the model-view-controller pattern. It is maintained by the DSF, an independent organization. It's primary goal is to ease the creation of complex, database-driven websites. Django emphasizes reusability and 'pluggability' of components, rapid development and the principle of don't repeat yourself. Python is used throughout, even for settings, files and data models. Django also provides an optional administrative CRUD interface that is generated dynamically through introspection and configured via admin models.

Examples of sites using it : Pinterest, Instagram, Mozilla, Disquis, Bitbucket.

Also included in the core framework are:
1. A lightweight and standalone web server for development and testing.
2. A form serialization and validation system which can translate between HTML forms and values suitable for storage in the database.
3. A template system that utilizes the concept of inheritance borrowed from object-oriented programming.
4. A caching framework which can use any of several cache methods.
5. Support for middleware classes which can intervene at various stages of request processing and carry out custom functions.
6. An internal dispatcher system which allows components of an application to communicate events to each other via pre-defined signals.
7. An internationalization system, including translations of Django's own components into a variety of languages.
8. A serialization system which can produce and
read XML and/or JSON representations of Django model instances.
9. A system for extending the capabilities of the template engine.
10. An interface to Python's built in unit test framework.

I used this framework for all the reasons I showed while describing it. With it, I am able to create any web apps very fast. Since it even uses Python in things such as settings and data models, I feel that I completed the task where the script was supposed to be written in Python, all this being is a framework to make my work into an actual usable product, not just a script which loads a site.

INSTRUCTIONS

Heroku – Windows – How to setup and run a new instance of the application

- Make a free account on Heroku
- Open command line
- do 'heroku login'
- do 'heroku git:clone -a freshertomatoes'
- do 'heroku create'
- do 'git push heroku master'
- do 'git run python manage.py migrate'
- If static files don't show, do 'heroku run python manage.py collectstatic'
- If you want, check logs via 'heroku logs --tail'

Heroku – How to access the administration page of the application

- Go to 'http://freshertomatoes.herokuapp.com/fresh_tomatoes/' to see the content in use.
- Open 'http://freshertomatoes.herokuapp.com/admin/login/?next=/admin/' to get to the administration login page.
- Username: 'test' , Password: 'test'
- Here you can add/remove/edit/read any of the movies or shows