

Universidad Nacional Autónoma de México

Facultad de Ciencias



TECNOLOGÍAS PARA DESARROLLOS EN INTERNET

Alumnos:

Lemus Pablo Arturo

Martinez Lorenzo Jiovanni

Pérez Márquez José Alejandro

Sánchez Alcántara Jesús Esteban

Índice

1. Introducción	3
2. Desarrollo	3
2.1. Nombre del framework y lenguaje que usa	3
2.2. Propósito del framework	4
2.3. Frontend o backend	4
2.4. ¿Quién lo usa?	4
2.5. Curva de aprendizaje	4
3. Conclusión	5
4. Referencias	6

1. Introducción

Django nace en el año 2003 gracias a un equipo de desarrolladores de Kansas, comenzaron a usar Python para crear sus aplicaciones.

Desarrollaron por necesidad un framework de desarrollo Web que les ahorrara tiempo era la única forma en que podían crear aplicaciones mantenibles en tan poco tiempo. El equipo de World Online, decidió liberar el framework como software de código abierto. Lo liberaron en julio de 2005 y lo llamaron Django, por el guitarrista de jazz Django Reinhardt.

2. Desarrollo

2.1. Nombre del framework y lenguaje que usa

Django sigue el principio DRY (conocido también como Una vez y sólo una). El principio No te repitas (en inglés Don't Repeat Yourself o DRY, también conocido como Una vez y sólo una) es una filosofía de definición de procesos que promueve la reducción de la duplicación especialmente en programación. Según este principio toda pieza de información nunca debería ser duplicada debido a que la duplicación incrementa la dificultad en los cambios y evolución posterior. Los cambios en cualquier parte del proceso requieren cambios en un único lugar.

Este principio establece que, en un entorno informático, la información no debe repetirse. Es decir, el conocimiento almacenado en un programa informático debe mantenerse en un, y sólo en un, lugar.

En Django, las tres capas básicas son el modelo, la vista, y la plantilla.

1. El Navegador manda una solicitud
2. La vista interactúa con el modelo para obtener datos.
3. La vista llama a la plantilla.
4. La plantilla renderiza la respuesta a la solicitud del navegador

El modelo

El modelo define los datos almacenados, se encuentra en forma de clases de Python, cada tipo de dato que debe ser almacenado se encuentra en una variable con ciertos parámetros, posee métodos también. Todo esto permite indicar y controlar el comportamiento de los datos.

La vista

La vista se presenta en forma de funciones en Python. Su propósito es determinar qué datos serán visualizados, entre otras cosas más. El ORM de Django permite escribir código Python en lugar de SQL para hacer las consultas que necesita la vista. La vista también se encarga de tareas conocidas como el envío de correo electrónico, la autenticación con servicios externos y la validación de datos a través de formularios.

La plantilla

La plantilla es básicamente una página HTML con algunas etiquetas extras propias de Django, en sí no solamente crea contenido en HTML (también XML, CSS, Javascript, CSV, etc).

2.2. Propósito del framework

Citando el slogan de django: The web framework for perfectionists with deadlines. (el framework para aplicaciones web para perfeccionistas con fechas límite) Django fue inventado para cumplir con plazos cortos para desarrollo de proyectos así como para satisfacer los requerimientos de desarrolladores web experimentados. Esta herramienta permite un desarrollo rápido, limpio y pragmático de una aplicación web, proveyendo una estructura para incluir interactividad entre una base de datos y un navegador web.

2.3. Frontend o backend

Este framework es más conocido por lo rápido que lleva un proyecto web de un concepto a su lanzamiento. Enfocándose en la parte de backend de una aplicación web, Django incluye características como una técnica de mapeo objeto-relacional (ORM) que facilita el manejo de una base de datos implementando características de un lenguaje O.O., apoya el uso de formularios para ingresar información de parte del usuario, entre otras herramientas que ayudan a un desarrollo más fácil para el desarrollador.

2.4. ¿Quién lo usa?

- Pinterest: Red social que permite intercambiar recetas, artículos, negocios, ideas etc. Con millones de usuarios, era claro que debía ser amigable con el usuario y el uso de Django cumplía con las necesidades de los creadores.
- Instagram: Otra red social que se basa en compartir fotos entre los usuarios. Guardando momentos, atrayendo gente joven y usada entre famosos, Django fue utilizado en su creación y uso.
- Mozilla Firefox: Una buena herramienta para un buen navegador.

2.5. Curva de aprendizaje

Django tiene una curva de aprendizaje relativamente moderada si uno no ha trabajado con ningún framework MVC, ya que es MTV (Model Template View), aunque por otra parte es más sencillo de aprender que otros frameworks ya que hay gran cantidad de información disponible, que está muy bien redactada y completa.

En los proyectos que se hagan en Django se tendrán estructuras pequeñas, sin tantas carpetas, aunque el problema que se tendrá será en el manejo eventual de un esquema de la base de datos que se tenga.

3. Conclusión

En este trabajo aprendimos como crear un proyecto con Django y vimos que el API que ofrece es muy amplio para facilitar el desarrollo rápido de aplicaciones web basadas en un patrón CRUD.

Uno de los puntos a favor que Django tiene es que usa Python y esto permite que sea un framework de alto nivel debido a que está escrito en el, es decir, Django heredó sus características y facilidades de Python, la más clara es que el código es muy fácil de comprender, aparte de que uno puede usar puro Python para manejar la base de datos.

Otro punto a su favor es que Django nos permite, gracias a su estructura, que sus aplicaciones web se creen de una forma rápida, además de fomentar con la filosofía DRY la reutilización de código, esto hace una forma rápida de trabajar.

4. Referencias

- Django Software Foundation. (2016). “Empezando” en Django Documentation . [En línea]. EE UU, disponible en <https://docs.djangoproject.com/es/1.9/intro/>, consultado el día [13 -septiembre-2016]
- ¿Qué es Django? , <http://tutorial.djangogirls.org/es/django/>, consultado el día [14 -septiembre-2016]
- Django Tutorial - Tutorialspoint , <http://www.tutorialspoint.com/django/> , consultado el día [13 -septiembre-2016]
- Django, [https://es.wikipedia.org/wiki/Django_\(framework\)](https://es.wikipedia.org/wiki/Django_(framework)), consultado el día [13 -septiembre-2016]
- Próximo curso: Django, el web framewrok para perfeccionistas, <http://www.maestrosdelweb.com/curso-django-introduccion/>, consultado el día [14 -septiembre-2016]
- Cómo se utiliza Python y Django en Mejorando.la , <http://www.maestrosdelweb.com/python-django-mejorandola/>, consultado el día [13 -septiembre-2016]
- Tu primera app CRUD en Django con Class Based Views - Platzi , <https://platzi.com/blog/django-class-based-views/>, consultado el día [14 -septiembre-2016]
- Django overview, <https://www.djangoproject.com/start/overview/>, consultado el día [16 -septiembre-2016]
- python - Whats the best way to start learning django?, <http://stackoverflow.com/questions/9081123/is-django-for-the-frontend-or-backend>, consultado el día [16 -septiembre-2016]
- Top 10 sites built with Django Framework, <https://www.linkedin.com/pulse/top-10-sites-built-django-framework-vladimir-bogdanov>, consultado el día [17 -septiembre-2016]
- What is Django & why should I use it?, <https://www.djangorocks.com/what-is-django.html>, consultado el día [17 -septiembre-2016]
- Qué es MVC, <http://www.desarrolloweb.com/articulos/que-es-mvc.html>, consultado el día [13 -septiembre-2016]
- DRY: Don't Repeat Yourself (¡No Te Repitas!), <http://joaquin.medina.name/web2008/documentos/informatica/documentacion/logica/OOP/Principios/201>, consultado el día [14 -septiembre-2016]
- Python - DjangoFramework de desarrollo web para perfeccionistasBasado en el Modelo MTV, http://www.revistasbolivianas.org.bo/scielo.php?pid=S1997-40442012000200016&script=sci_arttext, consultado el día [13 -septiembre-2016]