

Proyecto: Albergue de mascotas*

Christian David Pocol Franco, carne: 201807325¹ and Hector Fernando Carrera Soto, carne 201700923¹

¹Facultad de Ingeniería, Escuela de mecánica eléctrica,
Universidad de San Carlos, Ciudad Universitaria, Zona 12, Guatemala.

I. LIBRERIAS Y APLICACIONES UTILIZADAS

A. Django/framework

Django es un framework web de alto nivel que permite el desarrollo rápido de sitios web seguros y mantenibles. Desarrollado por programadores experimentados, Django se encarga de gran parte de las complicaciones del desarrollo web, por lo que puedes concentrarte en escribir tu aplicación sin necesidad de reinventar la rueda. Es gratuito y de código abierto, tiene una comunidad próspera y activa, una gran documentación y muchas opciones de soporte gratuito y de pago. En un sitio web tradicional basado en datos, una aplicación web espera peticiones HTTP del explorador web (o de otro cliente). Cuando se recibe una petición la aplicación elabora lo que se necesita basándose en la URL y posiblemente en la información incluida en los datos POST o GET. Dependiendo de qué se necesita quizás pueda entonces leer o escribir información desde una base de datos o realizar otras tareas requeridas para satisfacer la petición. La aplicación devolverá a continuación una respuesta al explorador web, con frecuencia creando dinámicamente una página HTML para que el explorador la presente insertando los datos recuperados en marcadores de posición dentro de una plantilla HTML.

Las aplicaciones web de Django normalmente agrupan el código que gestiona cada uno de estos pasos en archivos separados:

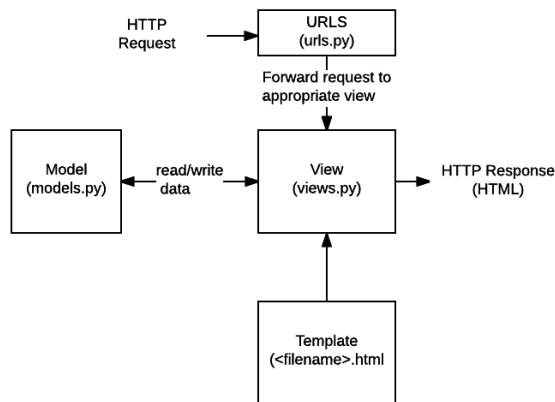


Figura 1: Diagrama de flujo django

B. bootstrap4

Bootstrap es un framework CSS desarrollado por Twitter en 2010, para estandarizar las herramientas de la compañía.

Inicialmente, se llamó Twitter Blueprint y, un poco más tarde, en 2011, se transformó en código abierto y su nombre cambió para Bootstrap. Desde entonces fue actualizado varias veces y ya se encuentra en la versión 4.4.

El framework combina CSS y JavaScript para estilizar los elementos de una página HTML. Permite mucho más que, simplemente, cambiar el color de los botones y los enlaces.

Esta es una herramienta que proporciona interactividad en la página, por lo que ofrece una serie de componentes que facilitan la comunicación con el usuario, como menús de navegación, controles de página, barras de progreso y más.

Además de todas las características que ofrece el framework, su principal objetivo es permitir la construcción de sitios web responsive para dispositivos móviles.

Esto significa que las páginas están diseñadas para funcionar en desktop, tablets y smartphones, de una manera muy simple y organizada.



C. Psycopg2/postgresql

PostgreSQL es uno de los sistemas de gestión de bases de datos relacionales más avanzados y utilizados. Es extremadamente popular por muchas razones, algunas de las cuales incluyen que sea de código abierto, su extensibilidad y su capacidad para manejar muchos tipos diferentes de aplicaciones y cargas variables. Con Python, puede establecer fácilmente una conexión a su base de datos PostgreSQL. Hay muchos controladores de Python para PostgreSQL, siendo «psycopg» el más popular. Su versión actual es psycopg2. Podemos integrar Postgres

* Proyectos de computación aplicados a I.E

con Python usando el psycopg2 módulo. psycopg2 es un adaptador de base de datos de Postgres para Python.

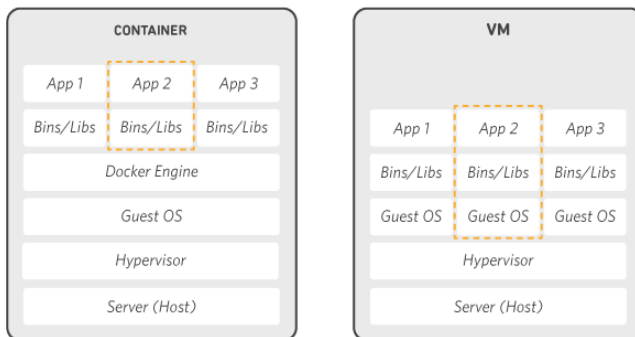
D. Crispy-forms

Django, por defecto, no proporciona ningún método de estilo de formulario de Django debido a que requiere mucho esfuerzo y un tiempo precioso para diseñar un formulario de manera hermosa. django-crispy-forms nos resuelve este problema. Le permitirá controlar el comportamiento de renderizado de sus formularios Django de una manera muy elegante y SECA.

E. Docker

Docker es una plataforma de software que le permite crear, probar e implementar aplicaciones rápidamente. Docker empaqueta software en unidades estandarizadas llamadas contenedores que incluyen todo lo necesario para que el software se ejecute, incluidas bibliotecas, herramientas de sistema, código y tiempo de ejecución. Con Docker, puede implementar y ajustar la escala de aplicaciones rápidamente en cualquier entorno con la certeza de saber que su código se ejecutará.

Docker le proporciona una manera estándar de ejecutar su código. Docker es un sistema operativo para contenedores. De manera similar a cómo una máquina virtual virtualiza (elimina la necesidad de administrar directamente) el hardware del servidor, los contenedores virtualizan el sistema operativo de un servidor. Docker se instala en cada servidor y proporciona comandos sencillos que puede utilizar para crear, iniciar o detener contenedores.



F. Amazon AWS

Amazon Web Services, también conocida como AWS, es un conjunto de herramientas y servicios de cloud computing de Amazon. Este servicio se lanzó oficialmente en 2006 y para junio de 2007 AWS ya contaba con una base de usuarios de aproximadamente 180 mil personas. Entre las empresas que la utilizan se encuentran algunas como Reddit, Foursquare, Pinterest, Netflix, la NASA o

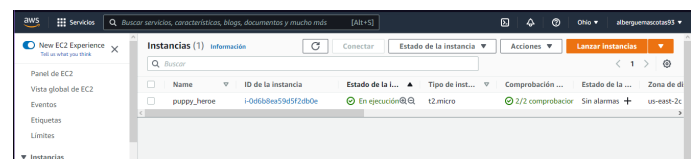
la CIA, y algunas españolas como Mapfre, el FC Barcelona o Interflora. Esto se debe principalmente a la madurez del servicio frente a otros similares y las posibilidades que ofrece el amplio abanico de herramientas disponibles. En la Guía de Cloud Computing podrá encontrar una comparativa de todas las herramientas de Amazon Web Services con las de otras plataformas similares.

Desde el almacenamiento de datos a las herramientas de implementación, y también, desde los directorios a la entrega de contenido, AWS dispone de más de 50 servicios a solo un clic; estos servicios se aprovisionan rápidamente y sin gastos por adelantado, para que empresas, startups, pequeñas y medianas empresas y clientes en general tengan acceso a los elementos básicos que necesitan para responder con rapidez.

G. DOCKER con AWS

AWS proporciona soporte técnico para Docker de código abierto y para sus soluciones comerciales. Existen varias maneras de ejecutar contenedores en AWS, incluido Amazon Elastic Container Service (ECS), un servicio de administración de contenedores altamente escalable y de gran desempeño. Los clientes pueden implementar de manera fácil sus aplicaciones en contenedores del entorno Docker local directamente a Amazon ECS. AWS Fargate es una tecnología para Amazon ECS que permite ejecutar contenedores en producción sin implementar o administrar infraestructura. Amazon Elastic Container Service for Kubernetes (EKS) facilita la ejecución de Kubernetes en AWS. AWS Fargate es una tecnología para Amazon ECS que le permite ejecutar contenedores sin aprovisionar ni administrar servidores. Amazon Elastic Container Registry (ECR) es un repositorio de contenedores privado seguro y altamente disponible que facilita el almacenamiento y la administración de imágenes de contenedores de Docker, así como el cifrado y la compresión de imágenes en reposo de manera que se puedan extraer de manera rápida y segura. AWS Batch le permite ejecutar cargas de trabajo de procesamiento por lotes altamente escalables con contenedores de Docker.

H. Servicio de nube en AWS en funcionamiento



I. Contenedores en docker activos

```

alberque@172-31-32-118: ~/0980_Proyectos-Alberque_de_mascotas/alberque_mascotas
$ sudo docker-compose up
Creating network "alberque_mascotas_default" with the default driver
Creating alberque_mascotas_db_1 ... done
Creating alberque_mascotas_django_app_1 ... done
Creating alberque_mascotas_nginx_1 ... done
Attaching to alberque_mascotas_db_1, alberque_mascotas_django_app_1, alberque_mascotas_nginx_1
alberque_mascotas_db_1 | PostgreSQL Database directory appears to contain a database: Skipping initialization
alberque_mascotas_db_1 |
alberque_mascotas_db_1 | 2022-05-12 23:28:01.433 UTC [1] LOG: starting PostgreSQL 12.9 (Debian 12.9-1.pgdg110+1) on x86_64-pc-linux-gnu, compiled by gcc (Debian
alberque_mascotas_db_1 | 10.2.1-6) 10.2.1, 20210118, 64-bit
alberque_mascotas_db_1 |
alberque_mascotas_db_1 | 2022-05-12 23:28:01.434 UTC [1] LOG: listening on IPv4 address "0.0.0.0", port 5432
alberque_mascotas_db_1 |
alberque_mascotas_db_1 | 2022-05-12 23:28:01.434 UTC [1] LOG: listening on IPv6 address ":::", port 5432
alberque_mascotas_db_1 |
alberque_mascotas_db_1 | 2022-05-12 23:28:01.439 UTC [1] LOG: listening on unix socket "/var/run/postgresql/.s.PGSQL.5432"
alberque_mascotas_db_1 |
alberque_mascotas_db_1 | 2022-05-12 23:28:01.472 UTC [240] LOG: database system was shut down at 2022-05-12 23:27:53 UTC
alberque_mascotas_db_1 |
alberque_mascotas_db_1 | 2022-05-12 23:28:01.473 UTC [1] LOG: database system is ready to accept connections
alberque_mascotas_nginx_1 |
alberque_mascotas_nginx_1 | 2022-05-12 23:28:02.400000 [1] [Info] Starting unicorn 20.1.0
alberque_mascotas_nginx_1 |
alberque_mascotas_nginx_1 | 2022-05-12 23:28:02.400000 [1] [Info] Listening at: http://0.0.0.0:8000 (1)
alberque_mascotas_nginx_1 |
alberque_mascotas_nginx_1 | 2022-05-12 23:28:02.400000 [1] [Info] Using worker: sync
alberque_mascotas_nginx_1 |
alberque_mascotas_nginx_1 | 2022-05-12 23:28:02.400000 [8] [Info] Booting worker with pid: 8
alberque_mascotas_nginx_1 |
alberque_mascotas_nginx_1 | 2022-05-12 23:28:02.400000 [9] [Info] Booting worker with pid: 9
alberque_mascotas_nginx_1 |
alberque_mascotas_nginx_1 | /docker-entrypoint.sh: /docker-entrypoint.d/ is not empty, will attempt to perform configuration
alberque_mascotas_nginx_1 |
alberque_mascotas_nginx_1 | /docker-entrypoint.sh: Looking for shell scripts in /docker-entrypoint.d/
alberque_mascotas_nginx_1 |
alberque_mascotas_nginx_1 | /docker-entrypoint.sh: Launching /docker-entrypoint.d/20-listen-on-ipv6-by-default.sh
alberque_mascotas_nginx_1 |
alberque_mascotas_nginx_1 | 20-listen-on-ipv6-by-default.sh: error: /etc/nginx/conf.d/default.conf is not a file or does not exist
alberque_mascotas_nginx_1 |
alberque_mascotas_nginx_1 | /docker-entrypoint.sh: Launching /docker-entrypoint.d/20-envsubst-on-templates.sh
alberque_mascotas_nginx_1 |
alberque_mascotas_nginx_1 | /docker-entrypoint.sh: Configuration complete; ready for start up

```

J. Repositorio de github

https://github.com/hefecaso/0980_Proyectos-Alberque_de_mascotas.git

II. DIAGRAMA DE FLUJO

