



UNIVERSIDAD DE SAN CARLOS DE GUATEMALA

FACULTAD DE INGENIERÍA

ESCUELA MECÁNICA ELÉCTRICA

PROYECTOS APLICADOS A LA INGENIERÍA ELÉCTRICA

ING. JOSÉ ANÍBAL SILVA DE LOS ÁNGELES

PROYECTO:
PAGINA WEB

24/09/2022

ESTUDIANTES

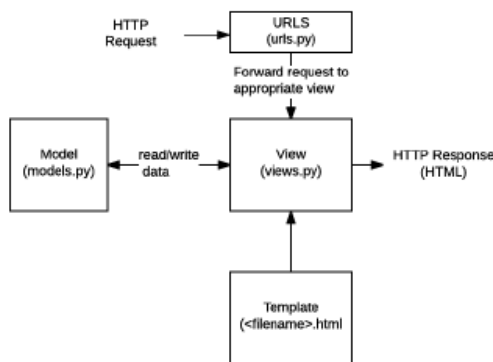
Bryan Geovanny Medina Umul, 201908094

Luis Arturo Quiñonez Bolaños, 201801101

I. LIBRERIAS Y APLICACIONES UTILIZADAS

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 ficheros separados:

Figura 1: Diagrama del proyecto general



II. 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.

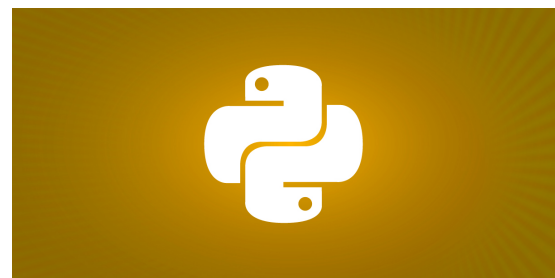
Figura 2: Bootstrap



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

Figura 3: Python



IV. 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.

Figura 4: Admin Django

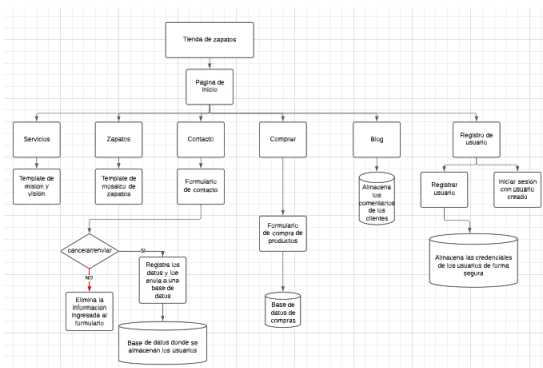
Example Form

Login

V. 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.

Figura 5: Diagrama de Flujo



VI. MANAGE

Este archivo realiza la función de trabajar todos los programas en la aplicación

Figura 6: Archivo manage

```
1 #!/usr/bin/env python
2 """Django's command-line utility for administrative tasks."""
3 import os
4 import sys
5
6
7 def main():
8     """Run administrative tasks."""
9     os.environ.setdefault('DJANGO_SETTINGS_MODULE', 'Albergue_mascotas.settings')
10    try:
11        from django.core.management import execute_from_command_line
12    except ImportError as exc:
13        raise ImportError(
14            "Couldn't import Django. Are you sure it's installed and "
15            "available on your PYTHONPATH environment variable? Did you "
16            "forget to activate a virtual environment?"
17        ) from exc
18    execute_from_command_line(sys.argv)
19
20
21 if __name__ == '__main__':
22     main()
```

VII. SETTINGS

Este archivo realiza la función de configurar los códigos necesarios para su buen funcionamiento.

Figura 7: Archivo Settings

```
1 """
2 Django settings for Albergue_mascotas project.
3
4 Generated by 'django-admin startproject' using Django 4.0.3.
5
6 For more information on this file, see
7 https://docs.djangoproject.com/en/4.0/topics/settings/
8
9 For the full list of settings and their values, see
10 https://docs.djangoproject.com/en/4.0/ref/settings/
11 """
12
13 import os
14 from pathlib import Path
15
16 # Build paths inside the project like this: BASE_DIR / 'subdir'.
17 BASE_DIR = Path(__file__).resolve().parent.parent
18
19
20 # Quick-start development settings - unsuitable for production
21 # See https://docs.djangoproject.com/en/4.0/howto/deployment/checklist/
```

VIII. LIBRERIAS

Este archivo realiza la función de instalar las librerías en el docker.

Figura 8: Archivo de librerías

```
1 Django==4.0.3
2 django-multiselectfield==0.1.12
3 DateTime
4 Pillow==7.0.0
5 django-admin-interface==0.19.0
6 django-colorfield==0.6.3
7 django-crispy-forms==1.14.0
8 django-flat-responsive==2.0
9 django-flat-theme==1.1.4
10 django-import-export==2.8.0
11 django-jet==1.0.8
12 django-multiselectfield==0.1.12
13 django-phonenumber-field==6.1.0
14 django-suit==0.2.28
15 gunicorn==20.1.0
16 psycopg2==2.8.4
17 css-parser==1.0.4
18 cssselect==1.1.0
19 cssutils==1.0.2
20 imutils==0.5.4
21 ipython_genutils==0.2.0
22 django-extensions==3.1.5
23 Django==4.0.3
24 django-extensions==3.1.5
25 django-enviro==0.8.1
26 django-axes==5.32.0
27 django-crispy-forms==1.14.0
28 django-ipware==4.0.2
```

```

7 <section class="page-section clearfix">
8   <div class="container">
9     <div class="intro">
10
11       
15
16       <div class="intro-text left=0 text-center bg-faded p-3 rounded">
17         <h2 class="section-heading mb-4">
18           <span class="section-heading-upper">Quiénes/</span>
19
20           <span class="section-heading-lower">somos/</span>
21         </h2>
22
23         <p class="mb-3">Somos una organización dedicada a la protección animal.</p>
24       </div>
25     </div>
26   </section>
27
28   <!-- Message -->
29
30 </section class="page-section cta">
31   <div class="container">
32     <div class="row">
33       <div class="col-xl-9 mx-auto">
34         <div class="cta-inner text-center rounded">
35           <h2 class="Section-heading mb-4">
36             <span class="Section-heading-upper">Comenzamos total/</span>

```

XIV. INICIO DE SESIÓN

Este archivo realiza la función de registrarse a la cuenta personal previamente registrada.

Figura 14: Archivo inicio de sesión

```
1 1
2 2
3 3
4 4
5 5
6 6
7 7
8 8
9 9
10 10
11 11
12 12
13 13
14 14
15 15
16 16
17 17
18 18
19 19
20 20
21 21
22 22
23 23
24 24
25 25
26 26
27 27
28 28
29 29
30 30
```

XV. SERVICIOS

Este archivo realiza la función de mostrar los servicios que ofrece la página web.

Redes de computadoras y arquitectura de comunicaciones, Barcia, N., Pearson, 2005 Como funcionan las redes, Derfler, F., Anaya Multimedia, 2006

Figura 15: Archivo servicios

```
1 1
2 2
3 3
4 4
5 5
6 6
7 7
8 8
9 9
10 10
11 11
12 12
13 13
14 14
15 15
16 16
17 17
18 18
19 19
```

XVI. LOGIN

Este archivo realiza la función de mostrar los requisitos necesarios del Login.

Figura 16: Archivo Login

```
1 1
2 2
3 3
4 4
5 5
6 6
7 7
8 8
9 9
10 10
11 11
12 12
13 13
14 14
15 15
16 16
17 17
18 18
19 19
20 20
21 21
22 22
23 23
24 24
25 25
26 26
27 27
28 28
29 29
30 30
31 31
32 32
33 33
```

Introducción a las telecomunicaciones modernas, Herrera, E., Limusa, 2006 Fundamentos de redes, Hallberg, B., Mc Graw Hill, 2006