

# Proyecto 1 Grupal: Creación de una Aplicación Web para Visualización de Datos

**Autor:**

Zaira Yorleni Nieto Ocampo

**TALENTO TECH**

**Programación Explorador**

**fecha de entrega:**

Domingo 12 de Mayo del 2025

**Puerto Carreño - VICHADA**

## Actividad a realizar:

Pasos del Proyecto:

Configuración del Entorno:

- Instala Flask y cualquier otra biblioteca necesaria (como Matplotlib o Chart.js) en tu entorno de desarrollo.

Desarrollo del Backend con Flask:

- Crea una aplicación Flask básica que incluya una ruta para cargar una página HTML.
- Implementa una función que permita a los usuarios cargar un archivo CSV.

Procesamiento de Datos con Python:

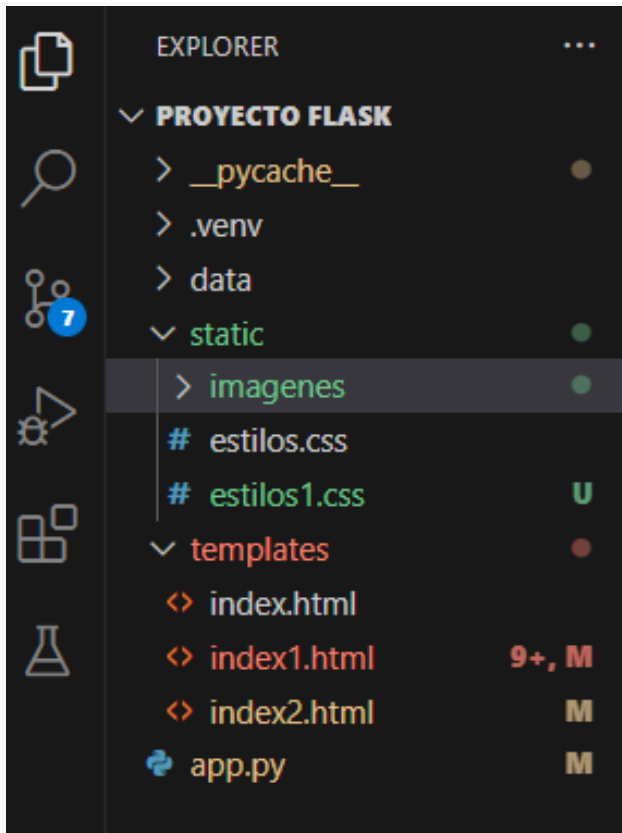
- Escribe un script en Python que lea el archivo CSV cargado por el usuario.
- Utiliza Matplotlib para generar gráficos básicos (por ejemplo, gráficos de barras o de líneas) a partir de los datos cargados.

Desarrollo del Frontend:

- Crea una página HTML que incluya un formulario para cargar archivos y una sección para mostrar los gráficos generados.
- Utiliza Chart.js o D3.js para crear gráficos interactivos a partir de los datos procesados por Python.

## 1. Configuración del Entorno:

Instala Flask y cualquier otra biblioteca necesaria (como Matplotlib o Chart.js) en tu entorno de desarrollo.



Biblioteca necesaria:

# estilos.css	# estilos1.css U	<> index.html	<> index2.html M
<pre> app.py &gt; home 1  from flask import Flask, render_template, request 2  import pandas as pd      #importar pandas al proyecto 3  import re 4  import matplotlib.pyplot as plt 5  import io                #renderizar gráficos 6  import base64 7 8  from datetime import datetime 9 </pre>			

## 2.Desarrollo del Backend con Flask:

A.Crea una aplicación Flask básica que incluya una ruta para cargar una página HTML.

-Definimos la ruta, es decir, que cuando un usuario visite la URL `/pagina2`, Flask ejecutará la función `pagina2()`.

```
@app.route("/pagina2")
```

-Aquí se define la función que se ejecutará al acceder a `/pagina2`.

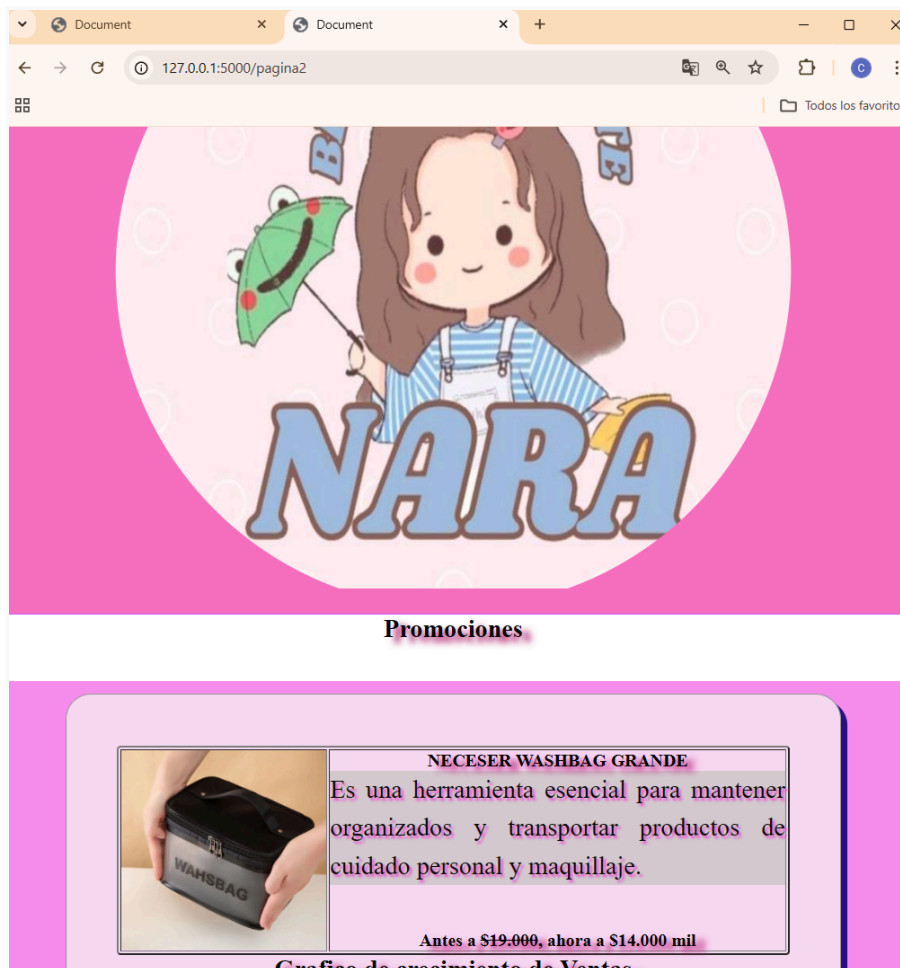
```
def pagina2():
```

-`render_template("index2.html")` carga y devuelve el contenido de la página `index2.html`.

-Flask busca esta plantilla en la carpeta `templates`, por lo que `index2.html` debe estar almacenado ahí.

```
    return render_template("index2.html")
```

Cuando alguien accede a `/pagina2`, Flask envía la página web `index2.html` al navegador del usuario para que la vea.



codigo completo:

```
@app.route("/pagina2")
```

```
def pagina2():
```

```
    return render_template("index2.html")
```

## B. Implementa una función que permita a los usuarios cargar un archivo CSV:

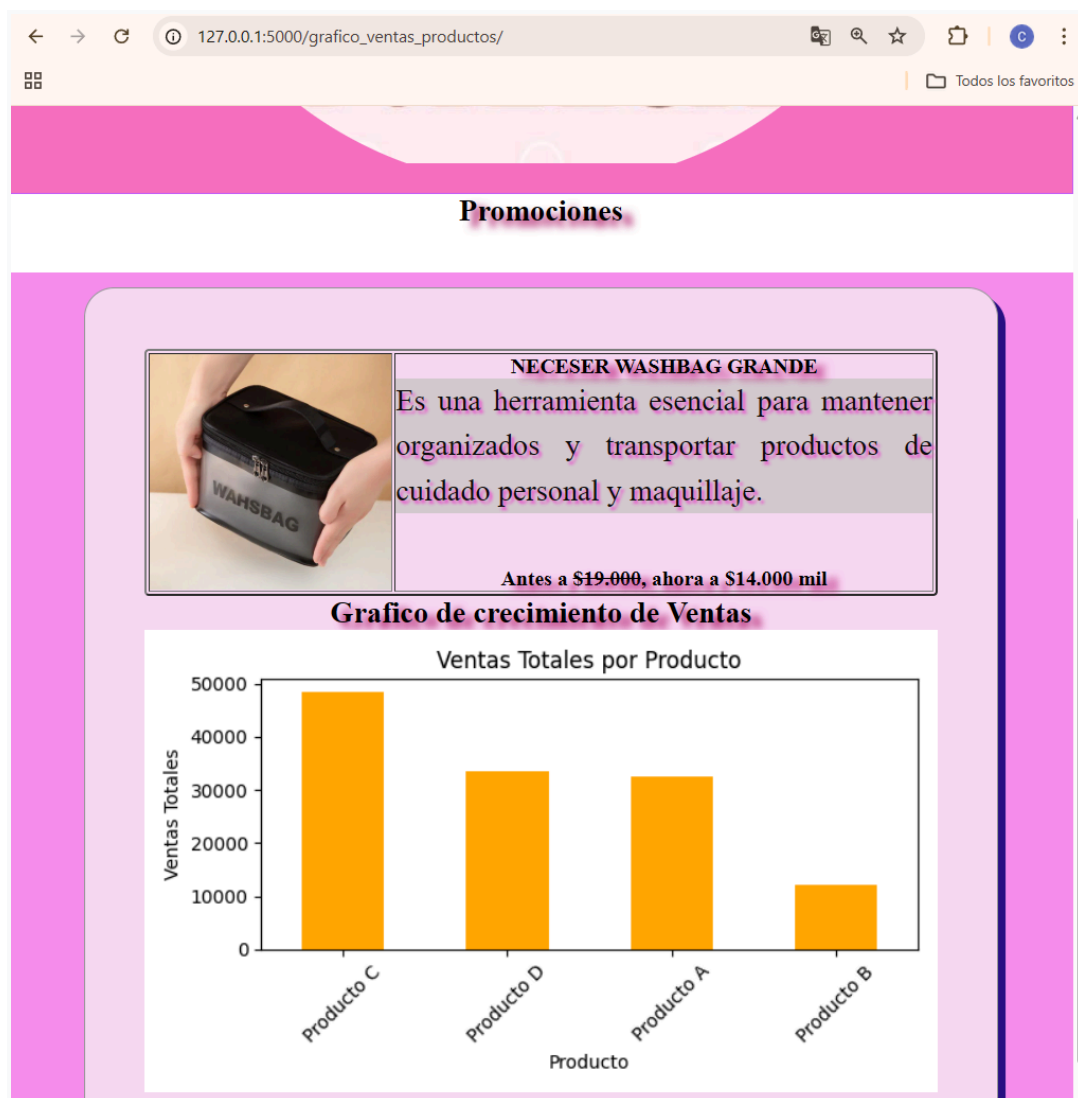
-Se define la ruta, esto configura la URL /grafico\_ventas\_productos/ en Flask. -Cuando un usuario acceda a esta ruta, se ejecutará la función grafico\_ventas\_productos().

```
@app.route('/grafico_ventas_productos/')
```

-Se cargan los datos, leyendo el archivo ventas\_anuales.csv utilizando pandas.

-La información se almacena en el DataFrame df para procesar los datos.

```
df= pd.read_csv('data/ventas_anuales.csv')
```



### 3. Procesamiento de Datos con Python:

**A. Utiliza Matplotlib para generar gráficos básicos (por ejemplo, gráficos de barras o de líneas) a partir de los datos cargados.**

Se define la ruta, es decir, cuando un usuario visite la URL /grafico1/, Flask ejecutará la función grafico1():

```
@app.route('/grafico1/')
def grafico1():
```

**#Datos para el gráfico:**

Después definimos los datos del gráfico: X representa los meses y Y representa las ventas en cada mes.

```
x=[1,2,3,4,5]
```

```
y=[10,14,16,20,25]
```

Se crea una figura de tamaño 6x4 pulgadas, se genera un gráfico de línea con los puntos marcados (marker='o'), se asignan títulos a los ejes y al gráfico para facilitar la interpretación.

```
plt.figure(figsize=(6, 4))
```

```
plt.plot(x, y, marker='o', linestyle='-', color='b')
```

```
plt.title('crecimiento de ventas')
```

```
plt.xlabel('Mes')
```

```
plt.ylabel('Ventas')
```

io.BytesIO() crea un buffer en memoria donde se guarda la imagen.

plt.savefig(buf, format='png') guarda el gráfico en el buffer en formato PNG.

buf.seek(0) reinicia el buffer para que pueda ser leído desde el inicio.

base64.b64encode(buf.getvalue()).decode('utf8') convierte la imagen en base64, permitiendo incrustarse en una página web sin necesidad de guardarla en disco.

```
buf= io.BytesIO()
```

```
plt.savefig(buf, format='png')
```

```
buf.seek(0)
```

```

    imagen_base64=
    base64.b64encode(buf.getvalue()).decode('utf8')
    Se envía la imagen en formato base64 a una plantilla HTML
    (index2.html), donde podrá ser mostrada.
    return render_template('index2.html', imagen1=
    imagen_base64)

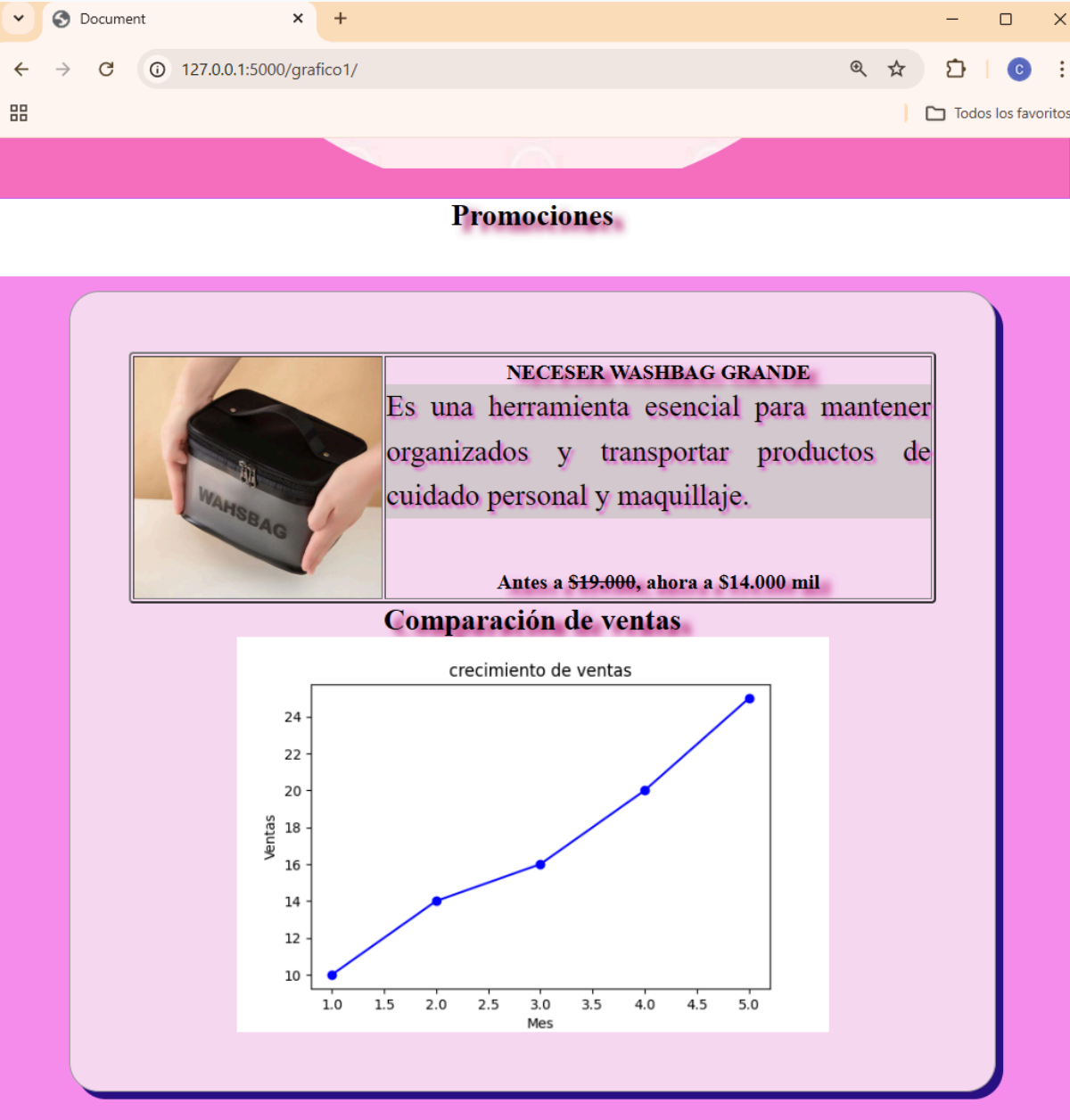
```

En HTML se pondría:

```



```



The screenshot shows a web browser window with the address bar displaying '127.0.0.1:5000/grafico1/'. The page has a pink header with the title 'Promociones'. Below the header, there is a product promotion for 'NECESER WASHBAG GRANDE'. The product is shown in a black and white image. The text describes it as an essential tool for keeping organized and transporting personal care and makeup products. The price is listed as 'Antes a \$19.000, ahora a \$14.000 mil'. Below the product promotion, there is a section titled 'Comparación de ventas' which contains a line graph titled 'crecimiento de ventas'. The graph shows sales growth over 5 months, with the y-axis labeled 'Ventas' and the x-axis labeled 'Mes'.

Mes	Ventas
1.0	10
2.0	14
3.0	16
4.0	20
5.0	24

Codigo completo:

```

@app.route('/grafico1/')
def grafico1():
#Datos para el gráfico
    x=[1,2,3,4,5]
    y=[10,14,16,20,25]
#Creación del gráfico
    plt.figure(figsize=(6, 4))
    plt.plot(x, y, marker='o', linestyle='-', color='b')
    plt.title('crecimiento de ventas')
    plt.xlabel('Mes')
    plt.ylabel('Ventas')
#Guardar gráfico en un buffer
    buf= io.BytesIO()
    plt.savefig(buf, format='png')
    buf.seek(0)
#codificar imagen en base64
    imagen_64= base64.b64encode(buf.getvalue()).decode('utf8')
    return render_template('index2.html', imagen1= imagen_64)

```

#### 4.Desarrollo del Frontend:

A.Crea una página HTML que incluya un formulario para cargar archivos y una sección para mostrar los gráficos generados.

Codigo de mi página:

```

<!DOCTYPE html>
<html lang="es">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>mi sitio web</title>
    <link rel="stylesheet" href='{{ url_for("static" ,
filename="estilos.css" ) }}' type="text/css">
    <script
src="https://cdn.jsdelivr.net/npm/chart.js"></script>
<!--es un conjunto de librerias que se encuentran en la
nube-->

```



```

</head>
<body>
    <button onclick="mostrarmMensaje()"> Haz clic
aqui</button>
    <header id="cabecera"><img class="logo" src='{{
url_for("static" , filename="imagenes/1.jpeg") }}'></header>
<nav id="menu">
    <a href="#inicio" class="nav-enlace">Inicio </a>
    <a href="#productos" class="nav-enlace">Productos </a>
    <a href="{{ url_for('pagina2') }}"
class="nav-enlace">Promociones </a>
    <a href="#contacto" class="nav-enlace">Contacto </a>
</nav>    <!--este es el menu principal-->
    <div id="cuerpo">
    <section id="seccion">
        <article id="inicio"class="articulo"><h1>Bienvenidos
a NARA:</h1><P class="parrafo1">Que somos?, en NARA, nos
apasiona ayudarte a descubrir tu belleza natural y a sentirte
segura en tu propia piel. Somos un emprendimiento dedicado al
cuidado de la piel y el maquillaje, comprometidos con ofrecer
productos de alta calidad que se adapten a tus necesidades
individuales.</P></article>
        <article class="articulo"><img src='{{
url_for("static" , filename="imagenes/2.jpg") }}' width="350px"
height="350px"><P>Te ofrecemos una guia detallada de todos
nuestros productos, destacando sus ingredientes clave y
beneficios para que puedas tomar decisiones
informadas</P></article>
        <article class= "articulo">
            <h1>Grafico de crecimiento de Ventas</h1>
            
        </article>
        <article class="articulo">
            <h1>Comparación de ventas</h1>
            <canvas id="chart5" style="width:500px;
height:300px;"></canvas>
        </article>

```

```

<article id="inicio"class="articulo"><P>En NARA,
estamos comprometidos en proporcionarte todo lo que necesitas
para una rutina de cuidado de la piel y maquillaje completa y
efectiva. <img src='{{ url_for("static" ,
filename="imagenes/3.jpeg") }}' width="350px;
height="350px"><br><h1>¡Es hora de realzar tu
belleza!</h1></P></article>

```

```

<article
id="productos"class="articulo"><h1><strong>KITS</strong></h1>
<table border="2">
    <tr>
        <td><img src='{{ url_for("static" ,
filename="imagenes/kit 1.png") }}' width="500px"
height="450px"></td>
        <td><h2>KIT DE ARROZ X6</h2><p>Este kit es una
solución integral para el cuidado facial, ofreciendo una
variedad de productos que trabajan juntos para mejorar la
salud y la apariencia de tu piel.</p>
        <br><h2>$79.000</h2>
    </td>
    </tr>
    <tr>
        <td><img src='{{ url_for("static" ,
filename="imagenes/Kit-tomate.jpg") }}' width="500px"
height="450px"></td>
        <td><h2>KIT DE TOMATE X4 IMAGES</h2><p>El
tomate es conocido por sus beneficios para la piel debido a su
rico contenido de antioxidantes, vitaminas y minerales.Este
kit incluye cuatro productos como: Limpiador, Serum,
Hidratante y Tónico refrescante.</p>
        <br><h2>$55.000</h2>
    </td>
    </tr>
    <tr>
        <td><img src='{{ url_for("static" ,
filename="imagenes/kit_rojo.jpeg") }}' width="500px"
height="450px"></td>
        <td><h2>KIT REJUVENECIMIENTO X3
IMAGES</h2><p>Un conjunto de productos para el
rejuvenecimiento facial consiste en una crema facial, unsuero

```

y un contorno de ojos. Estos productos están formulados con elastina, lo cual contribuye significativamente a mejorar la hidratación y revitalización de la piel.</p>

<br><h2>\$40.000</h2>

</tr>

</table></article>

<article id="inicio"class="articulo"><h2>MANEJO DE BASE DE DATOS:</h2><P>Lorem ipsum dolor sit amet, consectetur adipisicing elit. Voluptatibus id pariatur quaerat exercitationem architecto corporis, dolore asperiores dolorem velit nesciunt dolor, repudiandae totam odit, aspernatur excepturi quisquam laborum nihil blanditiis!este es el espacio del articulo 2</P></article>

<article id="inicio"class="articulo"><h2>MANEJO DE BASE DE DATOS:</h2><P>Lorem ipsum dolor sit amet, consectetur adipisicing elit. Voluptatibus id pariatur quaerat exercitationem architecto corporis, dolore asperiores dolorem velit nesciunt dolor, repudiandae totam odit, aspernatur excepturi quisquam laborum nihil blanditiis!este es el espacio del articulo 2</P></article>

<article id="inicio"class="articulo"><h2>MANEJO DE BASE DE DATOS:</h2><P>Lorem ipsum dolor sit amet, consectetur adipisicing elit. Voluptatibus id pariatur quaerat exercitationem architecto corporis, dolore asperiores dolorem velit nesciunt dolor, repudiandae totam odit, aspernatur excepturi quisquam laborum nihil blanditiis!este es el espacio del articulo 2</P></article>

<article id="inicio"class="articulo"><h2>MANEJO DE BASE DE DATOS:</h2><P>Lorem ipsum dolor sit amet, consectetur adipisicing elit. Voluptatibus id pariatur quaerat exercitationem architecto corporis, dolore asperiores dolorem velit nesciunt dolor, repudiandae totam odit, aspernatur excepturi quisquam laborum nihil blanditiis!este es el espacio del articulo 2</P></article>

</section>

<aside id="lateral">

```

        <form name="Formulario">
        <h1>Formulario web</h1>
        <table>
            <tr><td>Nombre:</td><td><input type="text"
name="txtNombre" required></td></tr>
            <tr><td>Correo:</td><td><input type="email"
name="miEmail" placeholder="Ingrese email"></td></tr>
            <tr><td>Celular:</td><td><input type="number"
name="numEst" step="1"></td></tr>
            <tr><td>Ciudad:</td><td><input type="text"
name="txtNombre" required></td></tr>
            <tr><td><input type="submit" name="btnEnviar"
value="Enviar"></td></tr>
        </table>
        </form>
        <br>
        <table id="contacto"><h1> REDES SOCIALES</h1></table>
        <tr>¡Para brindarte una atencion mas personalizada
y detallada, te invitamos a ponerte en contacto con
nosotros!.</tr>
        <br>
        <tr><img src='{ { url_for("static" ,
filename="imagenes/WAWA.png") } }' width="225"
height="215"><br> WHATSAPP: 315 3463778</tr>
        <tr><img src='{ { url_for("static" ,
filename="imagenes/INTA.png") } }' width="225"
height="215"><br> INSTAGRAM: cuidado_facial_NARA</tr>
        <tr><img src='{ { url_for("static" ,
filename="imagenes/FACE.png") } }' width="225"
height="215"></td><br> FACEBOOK: Cuidado facial</tr>
        </table>
    </aside>
    <footer id="pie"><h2>HORARIOS: Lunes a viernes de 8am a 6
pm </h2><br>
        <h2>DIRECCION: Carrera 11 Cl 25</h2></br>
        <h2>CONTACTO: 3174089322</h2></br>
        <h2>EMAIL: zairanieto29@gmail.com</h2></br>
    </footer>

```

```

<script>
    const labels= {{ labels | tojson }};
    const Basico= {{ basico | tojson }};
    const Premium= {{ premium | tojson }};
    new Chart(document.getElementById("chart5"), {
        type: 'line',
        data: {
            labels: labels,
            datasets: [
                {
                    label: 'Kit básico',
                    data: Basico,
                    borderColor: 'rgba(255, 99, 132, 1)',
                    fill: false
                },
                {
                    label: 'Kit premium',
                    data: Premium,
                    borderColor: 'rgba(54, 162, 235, 1)',
                    fill: false
                }
            ]
        },
        option: { responsive: false }
    });
</script>

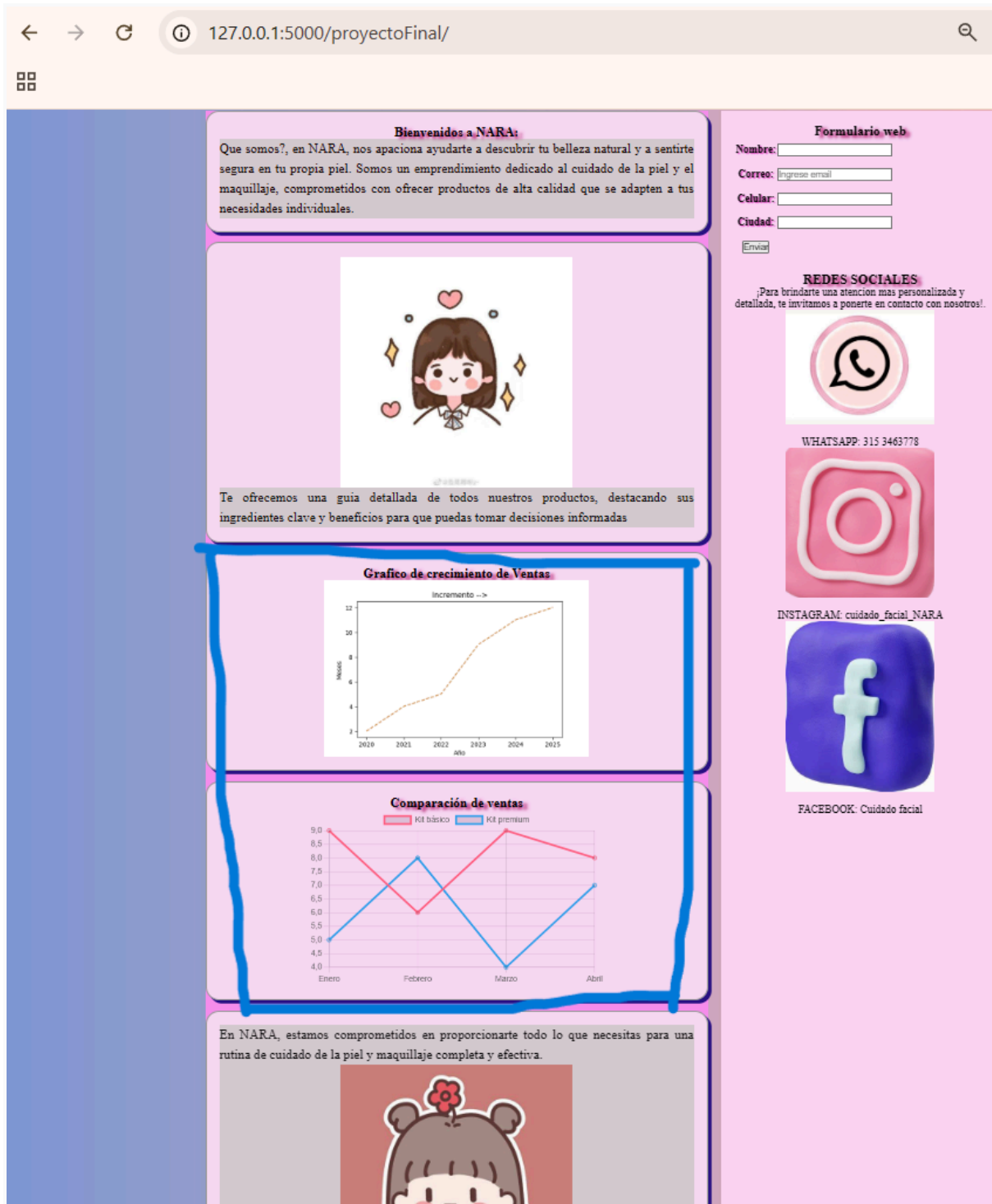
</body>
</html>

```

**Formulario señalado en azul:**



Sección para mostrar los gráficos generados: (señalado en azul)



**B.Utiliza Chart.js o D3.js para crear gráficos interactivos a partir de los datos procesados por Python.**

-Se crean las variables en Python, se crean listas que contienen los nombres de los meses (labels) y los datos de dos conjuntos (basico y premium).

-basico y premium representan valores numéricos asociados a cada mes:

```
labels= ["Enero", "Febrero", "Marzo", "Abril"]
        basico= [9, 6, 9, 8]
        premium= [5, 8, 4, 7]
```

-Render\_template() envía las variables a una plantilla HTML (index1.html) para que puedan ser utilizadas en la página web:

```
        return render_template("index1.html", labels= labels,
        basico= basico, premium= premium, imagen= imagen)
```

-Flask convierte las variables Python a formato JSON para que JavaScript pueda usarlas.

-Se crean constantes (labels, Basico, Premium) con los datos recibidos:

```
const labels= {{ labels | tojson }};
const Basico= {{ basico | tojson }};
const Premium= {{ premium | tojson }};
```

-Se crea un gráfico de líneas (type: 'line') con Chart.js..

-labels define los puntos en el eje X (los meses).

-datasets define las líneas:

-Kit básico → Usa los valores de Basico y color rojo.

-Kit premium → Usa los valores de Premium y color azul.

-fill: false indica que las líneas no tendrán un fondo relleno.

-responsive: false evita que el gráfico se ajuste automáticamente al tamaño de la pantalla:

```
new Chart(document.getElementById("chart5"), {
    type: 'line',
    data: {
        labels: labels,
        datasets: [
            {
                label: 'Kit básico',
                data: Basico,
                borderColor: 'rgba(255, 99, 132, 1)',
```

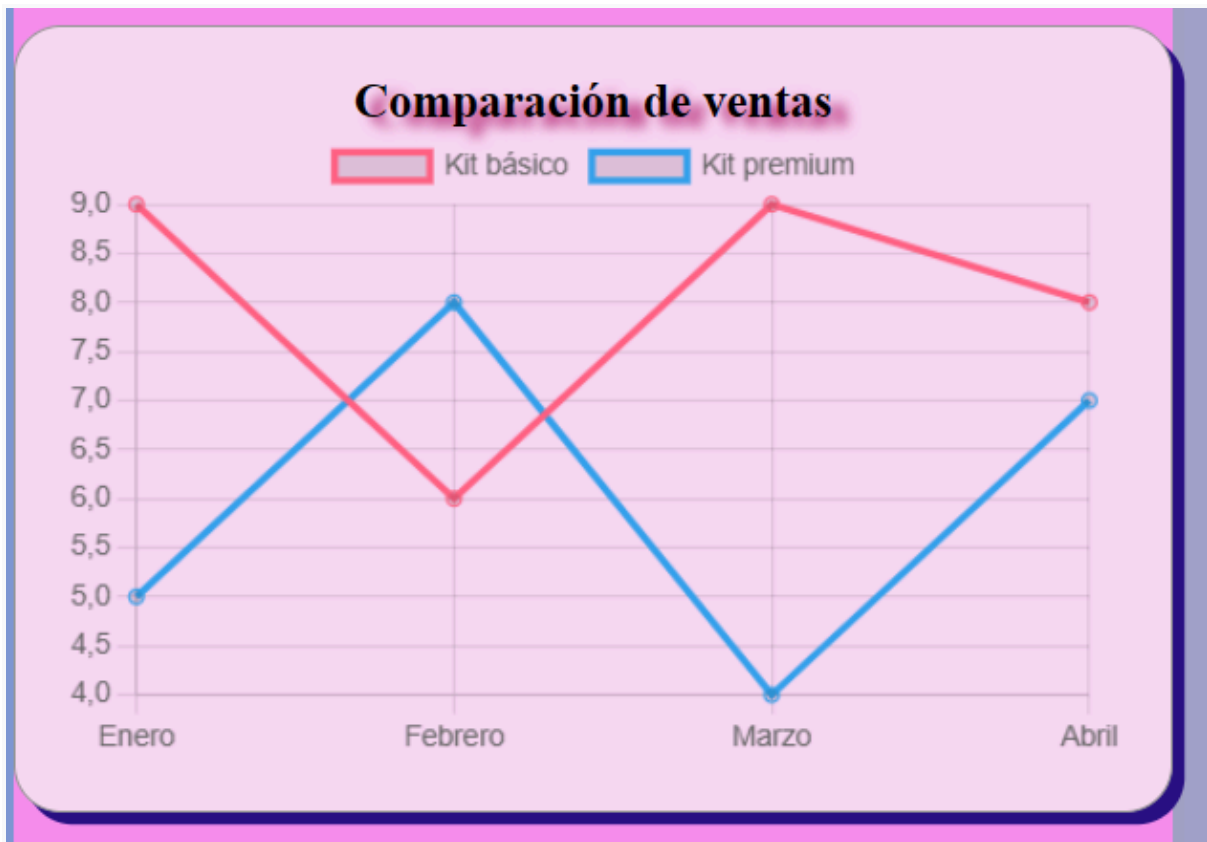


```

        fill: false
      },
      {
        label: 'Kit premium',
        data: Premium,
        borderColor: 'rgba(54, 162, 235, 1)',
        fill: false
      }
    ]
  },
  option: { responsive: false }
});

```

En la página se debe ver así:



Código html completo:

```
<script>
```

```

    const labels= {{ labels | tojson }};
    const Basico= {{ basico | tojson }};
    const Premium= {{ premium | tojson }};
    new Chart(document.getElementById("chart5"), {

```

```

        type: 'line',
        data: {
            labels: labels,
            datasets: [
                {
                    label: 'Kit básico',
                    data: Basico,
                    borderColor: 'rgba(255, 99, 132, 1)',
                    fill: false
                },
                {
                    label: 'Kit premium',
                    data: Premium,
                    borderColor: 'rgba(54, 162, 235, 1)',
                    fill: false
                }
            ]
        },
        option: { responsive: false }
    });
</script>

```

Y LISTO! Tenemos un código que generará un gráfico de líneas que compara dos conjuntos de datos mes a mes.