

MODULO 2. DESARROLLO DE APLICACIONES PARA TECNOLOGÍAS DE INTERNET CON PYTHON EN MICROSOFT VISUAL STUDIO CODE

1. INTRODUCCIÓN Y FUNDAMENTOS

CONTENIDOS A DESARROLLAR

- Aplicaciones web.
- Flask y Django.
- Jinja2.
- WSGI Python.
- HTML, CSS y JavaScript.

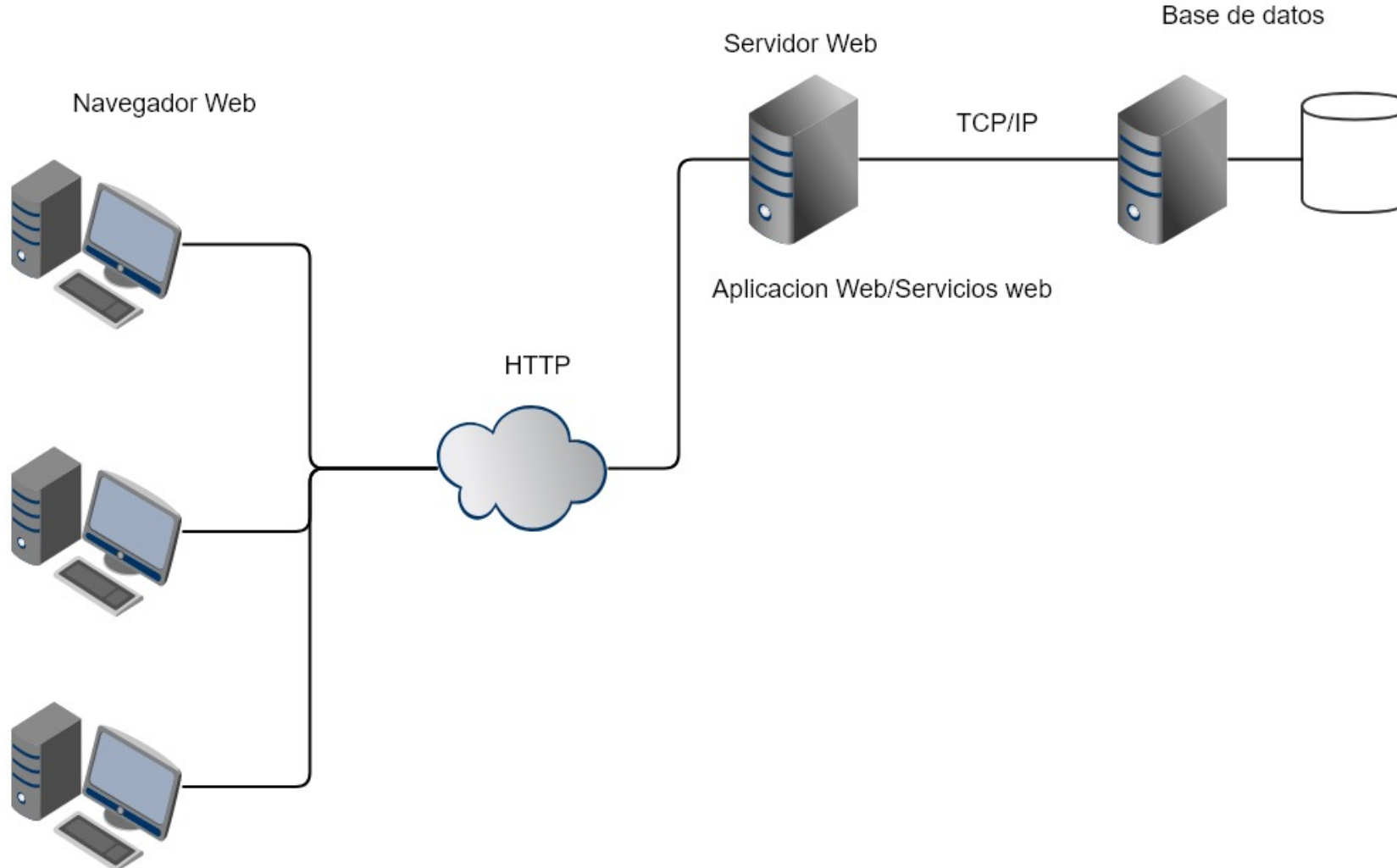
APLICACIONES WEB

- Se denomina aplicación web a aquellas herramientas que los usuarios pueden utilizar accediendo a un servidor web a través de internet o de una intranet mediante un navegador. En otras palabras, es un programa que se codifica en un lenguaje interpretable por los navegadores web en la que se confía la ejecución al navegador.
- Las aplicaciones web son una herramienta que se codifica en un lenguaje que es soportado por los navegadores y que se ejecuta en los mismos. Es decir, son una clase de software que puedes usar accediendo a cierto servidor web a través de Internet o de una Intranet por medio de un navegador que ejecutará la aplicación.

COMPONENTES NECESARIOS EN UNA APLICACIÓN WEB

- Una base de datos
- Código de la aplicación
- Servidor web
- Navegador de internet

FUNCIONAMIENTO



ARQUITECTURA DE 3 CAPAS

Se distinguen tres componentes básicos de software:

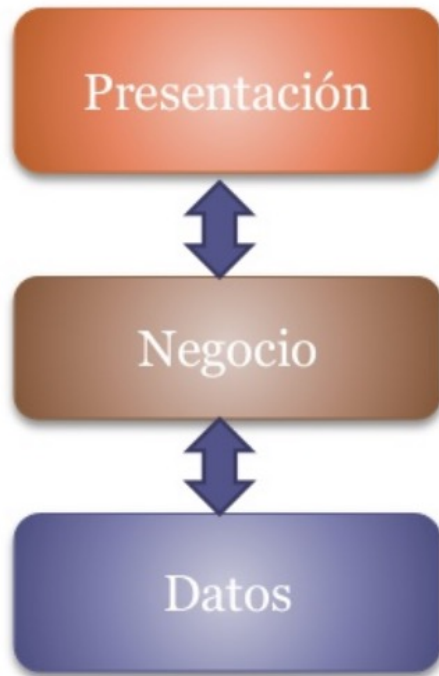
Presentación.- Tiene que ver con la presentación al usuario de un conjunto de objetos visuales y llevar a cabo el procesamiento de los datos producidos por el mismo y los devueltos por el servidor.

Lógica de aplicación.- Esta capa es la responsable del procesamiento de la información que tiene lugar en la aplicación.

Base de datos.- Esta compuesta de los archivos que contienen los datos de la aplicación.

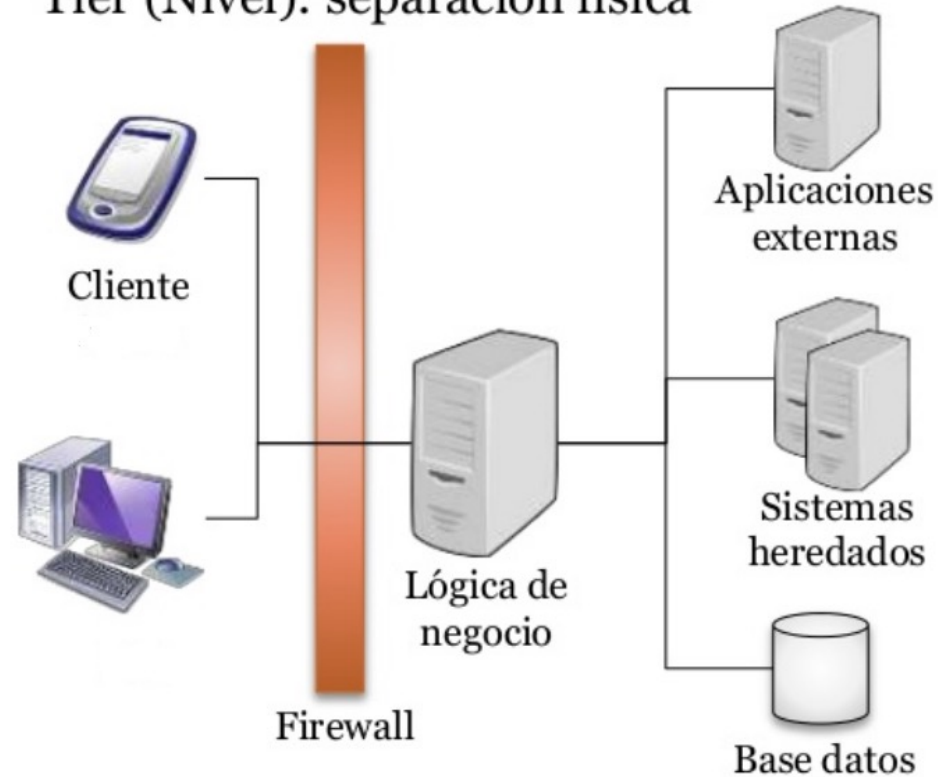
ARQUITECTURA DE 3 CAPAS

Layer (capa): separación conceptual



3-Capas (3-Layers)

Tier (Nivel): separación física



Presentación

Negocio

Datos

3-niveles (3-tier)

ARQUITECTURA MULTINIVEL



- Interfase de usuario
- Administración de las transacciones

- Lógica del negocio
- Caché
- Administración de las transacciones
- Transparencia en la localización de los datos
- Balance de carga

- Administración de los datos

FRAMEWORKS PYTHON PARA DESARROLLO WEB

- Existen 2 frameworks concretos que son los más empleados en desarrollo web Python, por ser proyectos que han ido evolucionando con una mayor frecuencia.
- Flask
- Django

DJANGO

- Django es un framework web de alto nivel que permite el desarrollo rápido de sitios web seguros y mantenibles.
- 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

FLASK

- Flask es un marco web de Python minimalista que proporciona herramientas y funciones útiles que hacen que crear aplicaciones web en Python sea más fácil.
- Ofrece a los desarrolladores flexibilidad y un marco más accesible para los nuevos desarrolladores ya que puede crear una aplicación web rápidamente usando únicamente un archivo Python.
- Flask también es extensible y no fuerza una estructura de directorio concreta ni requiere código estándar complicado antes de iniciarse.

MICRO FRAMEWORK

- El "micro" en microframework significa que Flask tiene como objetivo mantener el núcleo simple pero extensible.
- Flask puede ser todo lo que se necesita y nada que no se necesite.
- Flask admite extensiones para agregar funcionalidades a la aplicación como si estuviera implementada en el propio Flask. Numerosas extensiones brindan integración de bases de datos, validación de formularios, manejo de carga, varias tecnologías de autenticación abierta y más.
- Flask depende esencialmente del motor de plantillas Jinja y del kit de herramientas Werkzeug WSGI.

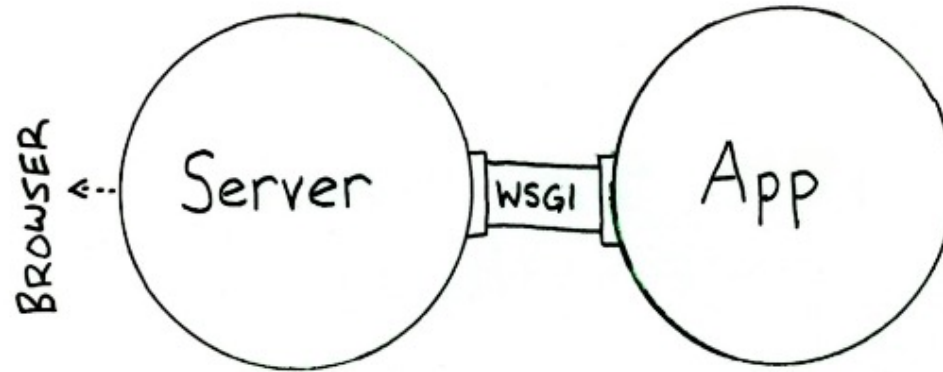
JINJA2

- Jinja2 es un motor de plantilla desarrollado en Python. Flask utiliza jinja2 para generar documentos HTML válidos de una manera muy sencilla y eficiente.
- Por dependencias al instalar Flask instalamos jinja2.

```
{% extends "layout.html" %}
{% block body %}
    <ul>
        {% for user in users %}
            <li><a href="{{ user.url }}">{{ user.username }}</a></li>
        {% endfor %}
    </ul>
{% endblock %}
```

WSGI PYTHON

- Es una especificación que describe cómo se comunica **un servidor web** con **una aplicación web**, y cómo se pueden llegar a encadenar diferentes aplicaciones web para procesar una solicitud/petición (o *request*).



WSGI PYTHON

- WSGI tiene dos partes:
 - El servidor, a menudo servidores web de alto perfil como Nginx o Apache.
 - La aplicación web creada a partir de un script de Python
- El servidor ejecuta la aplicación web y envía información relacionada y una función de devolución de llamada a la aplicación. La solicitud se procesa en el lado de la aplicación y se envía una respuesta al servidor utilizando la función de devolución de llamada.
- Elaborar ejemplo de HTTPServer según indicaciones del instructor

WERKZEUG

- werkzeug Sustantivo alemán: "herramienta". Etimología: werk ("trabajo"), zeug ("cosas")
- Werkzeug es una biblioteca completa de aplicaciones web WSGI. Comenzó como una simple colección de varias utilidades para aplicaciones WSGI y se ha convertido en una de las bibliotecas de utilidades WSGI más avanzadas.
- Flask envuelve Werkzeug, usándolo para manejar los detalles de WSGI mientras proporciona más estructura.

LENGUAJES DEL LADO DEL CLIENTE

- HTML
- CSS
- Javascript

HTML

- **HTML** es un lenguaje de marcado que se utiliza para el desarrollo de páginas de Internet. Se trata de la siglas que corresponden a HyperText Markup Language, es decir, Lenguaje de Marcas de Hipertexto

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
    <title></title>
```

```
</head>
```

```
<body>
```

```
</body>
```

```
</html>
```

Elaborar ejemplos segun indicaciones del instructor

HTML

This is my form

Email:

Password:

Confirm Password:

Upload Your Picture:

Browse...

Gender:

☐

Male

☐

Female

Select Country:

United Kingdom



Hobby:

☐

Cricket

☐

Football

Address

Submit

CSS CASCADE STYLE SHEET

- CSS es el lenguaje que usamos para diseñar un documento HTML.
- CSS describe cómo se deben mostrar los elementos HTML.
- Elaborar ejemplos según indicaciones del instructor

JAVASCRIPT

- JavaScript (JS) es un lenguaje de programación ligero, interpretado, o compilado justo-a-tiempo (just-in-time) con funciones. Si bien es más conocido como un lenguaje de scripting (secuencias de comandos) para páginas web, y es usado en muchos entornos fuera del navegador, tal como Node.js, Apache CouchDB
- Se ejecuta del lado del cliente.
- Elaborar ejemplos segun indicaciones del instructor

PREGUNTAS?

Gracias por su atención!!!!