

UNIDAD 4.

Publicación de aplicaciones web.

(El estudiante demostrará el proceso para el desarrollo de una aplicación web.)

Tema II. Herramientas para publicación de las aplicaciones web.

Evidencia de aprendizaje Tema II –RESUMEN

Temas	Saber	Saber hacer	Resultado de Aprendizaje
Herramientas para publicación de las aplicaciones web	Identificar las herramientas para la transferencia de archivos y publicación de la aplicación web	Determinar el protocolo y la herramienta para la transferencia de archivos de acuerdo con las necesidades de la aplicación web.	Los estudiantes identifican los protocolos y herramientas de transferencia de archivos.
Evidencia de aprendizaje			
A partir de un caso práctico realizar la publicación de una aplicación web y lo documenta en un reporte técnico que incluya: Listado de herramientas, secuencia de configuración, secuencia de publicación, pruebas del funcionamiento y publicación de la aplicación web.			

Información sobre el profesor

Profesor	Correo	Días de Clase
Bernardo Prado Díaz	Tenor_prado@yahoo.com.mx	Lunes y Martes

Saber: ¿Qué herramientas existen?

Para publicar una aplicación web, se utilizan herramientas que permiten transferir archivos desde el entorno local al servidor y configurar el entorno de ejecución. Las más comunes son:

Herramientas de transferencia de archivos

- SFTP (Secure File Transfer Protocol): protocolo seguro para subir archivos. Ejemplo: FileZilla, WinSCP.
- **Git:** sistema de control de versiones que permite desplegar proyectos desde repositorios remotos (GitHub, GitLab).
- Consolas web: como la Bash console en PythonAnywhere para subir, mover y configurar archivos directamente.

Plataformas de publicación

- PythonAnywhere: ideal para proyectos Django; permite subir archivos, configurar entornos virtuales y publicar apps.
- Heroku: plataforma PaaS que automatiza el despliegue desde Git.
- Render: alternativa moderna para publicar apps con backend.

Saber Hacer: ¿Cómo se publica una app web?

Pasos generales para publicar una app Django

1. Preparar el proyecto
 - Tener una app funcional con manage.py, requirements.txt, y configuración de producción en settings.py.
2. Transferir archivos
 - Usar SFTP (FileZilla) o Git para subir el proyecto al servidor.
3. Configurar entorno
 - Crear entorno virtual (venv) y activar.
 - Instalar dependencias con pip install -r requirements.txt.
4. Migrar base de datos
 - Ejecutar python manage.py migrate.
5. Configurar archivos estáticos
 - Ejecutar collectstatic si se usa Django.
6. Publicar
 - Configurar el archivo WSGI y recargar la app desde el panel web.

Resultado de Aprendizaje

Al finalizar, el estudiante será capaz de:

- Reconocer los protocolos de transferencia más seguros y eficientes.
- Elegir la herramienta adecuada según el tipo de aplicación.
- Publicar una aplicación web funcional en una plataforma real.
- Documentar el proceso de despliegue de forma técnica y clara.

Herramientas para Publicación de Aplicaciones Web con Python y Django

1 SABER – Dimensión Conceptual

Objetivo: Identificar y comprender las herramientas y protocolos para la transferencia de archivos y publicación de aplicaciones web utilizando Python y Django.

A) Conceptos Clave

- FTP (File Transfer Protocol):

Definición: Protocolo estándar para transferir archivos entre cliente y servidor.

Ejemplo: Ejemplo: Uso de FileZilla para subir archivos estáticos y código al servidor.

- SFTP (Secure File Transfer Protocol):
Definición: Protocolo de transferencia de archivos sobre SSH, cifrado y seguro.
Ejemplo: Ejemplo: Uso de WinSCP para transferir archivos protegidos.
- FTPS (FTP Secure):
Definición: Extensión de FTP que añade seguridad mediante SSL/TLS.
Ejemplo: Ejemplo: Configuración de conexión segura en FileZilla.
- Git y GitHub/GitLab:
Definición: Sistema de control de versiones distribuido que permite enviar cambios al servidor.
Ejemplo: Ejemplo: Deploy automático con GitHub Actions hacia un servidor de producción.
- rsync:
Definición: Herramienta de sincronización de archivos rápida y eficiente.
Ejemplo: Ejemplo: Actualización de archivos en un VPS desde una terminal Linux.

2 SABER HACER – Dimensión Actuacional

Objetivo: Seleccionar y utilizar el protocolo y herramienta adecuados para la publicación de una aplicación Django.

A) Ejemplo de Uso de SFTP con Django

- # Paso 1: Instalar un cliente SFTP como WinSCP o FileZilla.
- # Paso 2: Conectarse al servidor con usuario, contraseña o clave SSH.
- # Paso 3: Subir el código del proyecto Django a /var/www/mi_proyecto.
- # Paso 4: Reiniciar el servicio web (por ejemplo, gunicorn o apache2).

B) Ejemplo de Deploy con Git

```
# Inicializar repositorio en el servidor
git init
git remote add origin https://github.com/usuario/mi_proyecto.git

# Traer cambios y reiniciar servicio
git pull origin main
sudo systemctl restart gunicorn
```

3 SER Y CONVIVIR – Dimensión Socioafectiva

Objetivo: Promover la colaboración y la seguridad en el proceso de publicación.

- Revisar en equipo la seguridad de las credenciales antes de la publicación.

- Dividir roles: desarrollador, administrador de servidores, tester.
- Mantener comunicación constante durante el proceso de despliegue.
- Documentar los pasos realizados para futuras referencias.

4 Glosario

- Deploy → Proceso de poner en producción una aplicación.
- SSH → Protocolo seguro para administración remota.
- Repositorio → Almacén de código fuente con historial de cambios.
- Producción → Entorno en el que la aplicación es accesible por usuarios finales.
- Hosting → Servicio que aloja aplicaciones y sitios web.