Para desplegar tu aplicación desde cero en un servidor, el proceso abarcará desde la instalación del sistema operativo hasta la configuración completa del entorno. Aquí vamos a instalar un sistema operativo Linux (Ubuntu Server) por su estabilidad y soporte, además de su compatibilidad con servidores web como Nginx y Apache.

### Paso a Paso para Configurar el Servidor desde Cero

Paso 1. \*\***Instalación del Sistema Operativo**\*\*

- \*\***Descargar Ubuntu Server**\*\*: Descarga la imagen ISO de Ubuntu Server (por ejemplo, Ubuntu Server 22.04 LTS) desde [Ubuntu](https://ubuntu.com/download/server).

- \*\***Instalar el sistema operativo**\*\*:

- Accede a la consola del servidor o monta la ISO mediante un administrador de servidores si trabajas remotamente.

- Sigue las instrucciones de instalación, configurando:

- Idioma, teclado y zona horaria.

- Usuario administrador y contraseña.

- Particionamiento del disco (elige el esquema recomendado si el servidor está vacío).

- Durante la instalación, selecciona "Instalar solo el sistema base" sin paquetes adicionales, ya que los configuraremos manualmente.

Paso 2. \*\*Configuración Inicial del Sistema Operativo\*\*

- \*\*Acceso al sistema\*\*: Una vez instalado, inicia sesión en el servidor con el usuario y contraseña creados durante la instalación.

- \*\*Actualizar el sistema\*\*:

```bash

sudo apt update && sudo apt upgrade -y

```

- \*\*Configurar el firewall (UFW)\*\*:

```bash

sudo ufw allow OpenSSH

sudo ufw enable

sudo ufw status

```

Paso 3. \*\*Instalación del Servidor Web (Nginx)\*\*

- \*\*Instalar Nginx\*\*:

```bash

sudo apt install nginx -y

```

- \*\*Configurar el firewall para Nginx\*\*:

```bash

sudo ufw allow 'Nginx Full'

```

- \*\*Verificar el estado de Nginx\*\*:

```bash

sudo systemctl status nginx

```

Paso 4. \*\*Instalación de Node.js y npm (si la aplicación requiere JavaScript del lado del servidor)\*\*

- \*\*Instalar Node.js y npm\*\*:

```bash

sudo apt install nodejs npm -y

```

- \*\*Verificar las versiones\*\*:

```bash

node -v

npm -v

```

Paso 5. \*\*Instalación de Certbot para HTTPS (Opcional)\*\*

- \*\*Instalar Certbot y el plugin de Nginx\*\*:

```bash

sudo apt install certbot python3-certbot-nginx -y

```

Paso 6. \*\*Configuración del Nombre de Dominio y el DNS\*\*

- \*\*Configurar el subdominio\*\*: Si el subdominio ya está en el DNS y apunta a la IP del servidor, continúa al siguiente paso.

- \*\*Configuración de Nginx para el subdominio\*\*:

- Crea un archivo de configuración para el subdominio:

```bash

sudo nano /etc/nginx/sites-available/geoportal\_ucv

```

- En el archivo, define el servidor para el subdominio:

```nginx

server {

listen 80;

server\_name subdominio.universidad.edu;

root /var/www/geoportal\_ucv;

index index.html;

location / {

try\_files $uri $uri/ =404;

}

}

```

- Habilita el sitio y recarga Nginx:

```bash

sudo ln -s /etc/nginx/sites-available/geoportal\_ucv /etc/nginx/sites-enabled/

sudo systemctl reload nginx

```

Paso 7. \*\*Transferencia de Archivos de la Aplicación\*\*

- \*\*Crear el directorio raíz de la aplicación\*\*:

```bash

sudo mkdir -p /var/www/geoportal\_ucv

```

- \*\*Configurar permisos\*\*:

```bash

sudo chown -R $USER:$USER /var/www/geoportal\_ucv

sudo chmod -R 755 /var/www/geoportal\_ucv

```

- \*\*Transferir los archivos\*\*:

- En tu máquina local, usa `scp` para transferir los archivos:

```bash

scp -r /ruta/local/de/la/aplicacion usuario@dirección\_IP:/var/www/geoportal\_ucv

```

Paso 8. \*\*Configuración del HTTPS con Certbot (Opcional)\*\*

- Si deseas que la aplicación esté disponible a través de HTTPS, ejecuta Certbot:

```bash

sudo certbot --nginx -d subdominio.universidad.edu

```

Paso 9. \*\*Pruebas y Verificación de la Configuración\*\*

- \*\*Acceso al sitio\*\*: Abre un navegador y verifica el acceso a la URL `http://subdominio.universidad.edu`.

- \*\*Verificación del HTTPS\*\*: Si habilitaste HTTPS, también verifica `https://subdominio.universidad.edu`.

### Resumen de Configuraciones Clave y Verificaciones

1. \*\*Revisar permisos\*\*: Verifica que los archivos de la aplicación tengan permisos de lectura por parte de Nginx:

```bash

sudo chown -R www-data:www-data /var/www/geoportal\_ucv

```

2. \*\*Verificar firewall\*\*: Asegúrate de que los puertos necesarios estén abiertos:

```bash

sudo ufw status

```

Con estos pasos, tu servidor debería estar completamente configurado para hospedar la aplicación en el laboratorio.