

Configuración de Servidores

Peñañiel Cabrera Marisol
marisol.penañiel@ucuenca.edu.ec
Docente: Cedillo Priscila

Resumen—Esta práctica tiene como objetivo entender la configuración de los servidores locales, así como la configuración de los servidores de aplicación. Posteriormente para entender la diferencia entre un servidor web local y un servidor de despliegue en la nube.

Palabras clave—Configuración, servidores, puertos, cloud computing, despliegue continuo, hosting.

I. INTRODUCCIÓN

Los servidores web son importantes para el funcionamiento óptimo de los sitios web y aplicaciones web, ya que almacena, procesa y distribuye la información a través de internet. Los servidores web actúan como intermediarios entre el usuario y los sitios web ya que tienen la capacidad de gestionar las solicitudes de los usuarios de tal manera que brinde una información certera y eficiente.

I-A. Marco teórico

I-A1. Servidor Web: Un servidor web es un dispositivo que permite almacenar, procesar y entregar información de sitios web a los usuarios a través de un navegador, están formados por hardware y software los cuales utilizan un Protocolo de Transferencia de Hipertexto conocido como HTTP. Los servidores web también pueden usar el protocolo Simple de Transferencia de Correo (SMTP). [1]

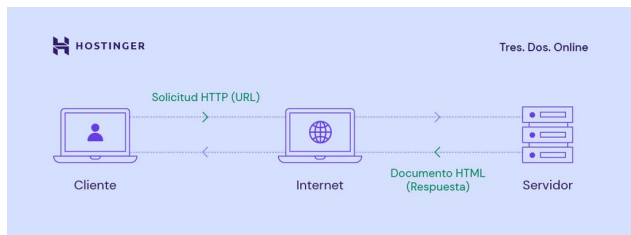


Figura 1. Funcionamiento de un Servidor.

El servidor web sigue un modelo cliente-servidor. En la figura 1 se observa que cuando el usuario de la web carga contenido a la página web se da una petición HTTP a través de internet. El servidor web recibe la petición HTTP y la procesa a través del servidor HTTP, posteriormente el servidor devuelve los archivos del sitio web que envió la solicitud.

Dentro de los servidores web más populares en la actualidad encontramos:

- **Servidor HTTP Apache:** es un servidor gratuito de código abierto que usa sistemas operativos como es Windows, Linux, Mac Os X.
- **Nginx:** es un servidor web de código abierto, también es usado como proxy inverso, cache de HTTP y balanceador de carga. Está diseñado para ofrecer un bajo uso de memoria y alta concurrencia. [2]
- **IIS:** Internet Information Services, es un conjunto de servicios que transforman un sistema Microsoft Windows en un servidor capaz de ofrecer servicios web, FTP y SMTP. [3]

I-A2. Servidor de Aplicaciones: Son programas que se encargan de ejecutar y gestionar aplicaciones web, permitiendo comunicarse con otras aplicaciones, base de datos, servidores web para realizar tareas específicas permitiendo un correcto funcionamiento del sistema. [4]

Dentro de los servidores de aplicaciones más populares en la actualidad encontramos:

- **JOnAs:** es un servidor de código abierto, bajo la licencia de LGPL implementado en Java. Forma parte de Object-Web. [5]
- **JBoss:** es un servidor de aplicaciones Java EE de código abierto. Es el primer servidor de aplicaciones de código de producción y certificado JEE 6.0. Una de sus características es la orientación a la arquitectura de servicios. [6]
- **Apache Tomcat:** desarrollado por Apache Software Foundation, incluye contenedores de servlets y JSP; genera contenido web en respuesta. Además proporciona una consola de administración web para configurar y gestionar el servidor [7]

I-A3. Hosting: El hosting es un servicio de alojamiento digital que permite publicar un sitio web o aplicación en Internet ya que almacena información y contenido. Dentro de las características que presenta un hosting tenemos:

- **Conectividad:** Acceso a la conexión de internet
- **Almacenamiento:** Información de los sitios web se guardan en bases de datos.
- **Capacidad:** Disponibilidad de espacios e infraestructura.
- **Integración:** Incluyen una contratación de un dominio web y una plataforma de correo electrónico.

- **Accesibilidad:** Guarda la información en los servidores de tal manera que si ocurre un problema con el equipo, no se pierde la información.

El funcionamiento de un hosting se basa en la integración de los servidores del proveedor del hosting y los usuarios que ingresan al sitio web. [8]

Algunos de los sitios de hosting mas populares tenemos:

- Bluehost
- Google Cloud
- WP Engine
- A2 Hosting
- entre otros

I-A4. Cloud Computing: Cloud Computing o computación en la nube, es un modelo de servicios de computación que permite acceder, agrupar y compartir recursos informáticos a través de internet para no depender de un espacio físico. Dentro de las características de cloud computing tenemos:

- **Conectividad:** Requiere acceso a Internet para funcionar.
- **Escalabilidad:** Capacidad de crecer al ritmo de la empresa o cliente.
- **Seguridad:** Las actualizaciones se realizan automáticamente, brindando las mejores prácticas de seguridad y privacidad.
- **Agilidad:** Todos los equipos pueden acceder a una interfaz compartida.
- **Disponibilidad:** La conexión a internet permite acceder a la información desde cualquier dispositivo.

Las soluciones de Cloud Computing se puede clasificar en 3 tipos [9]:

1. **Software como servicio (SaaS):** El proveedor vende el acceso a un software y a una base de datos, mientras que el cliente puede acceder a través de una aplicación específica o un navegador web.
2. **Plataforma como servicio (PaaS):** El cliente compra acceso virtual a los servicios y la infraestructura necesaria para diseñar e implementar aplicaciones.
3. **Infraestructura como servicio (IaaS):** Dirigida a arquitectos de redes y administradores de TI. Los proveedores venden acceso a recursos virtuales, incluidos servidores, redes y almacenamiento.

I-A5. Despliegue Continuo: Es una práctica que permite automatizar la entrega de software confiable y constante, permitiendo eliminar la fricción y acelerar el proceso [10]. A continuación veremos dos conceptos esenciales:

- **Integración continua:** Involucra fusionar y verificar el código desarrollado por miembros de un equipo de forma automatizada y regular.
- **Entrega continua:** Lleva el proceso de entrega del software. Es decir permite que el software pueda ser implementado en cualquier momento.

II. PRÁCTICA

II-A. Configurar los 3 tipos de servidores

- **APACHE WAMP o APACHE XAMPP:** Puede ser en Windows, Linux o macOS, no importa la distribución.

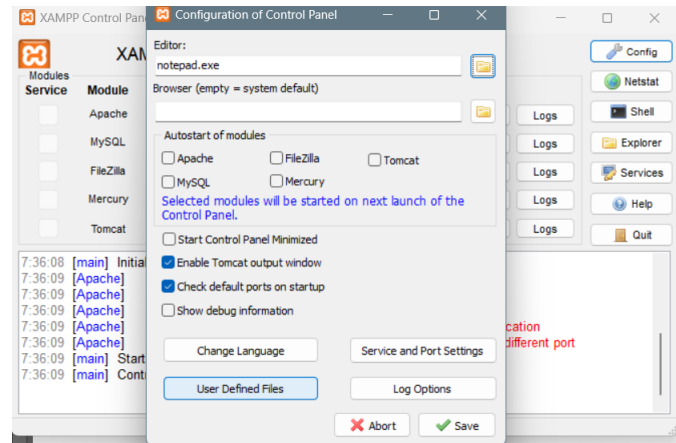


Figura 2. Descarga Apache Xampp

- **APACHE TOMCAT:** Buscar un script del lado del servidor de hola mundo y desplegarlo.

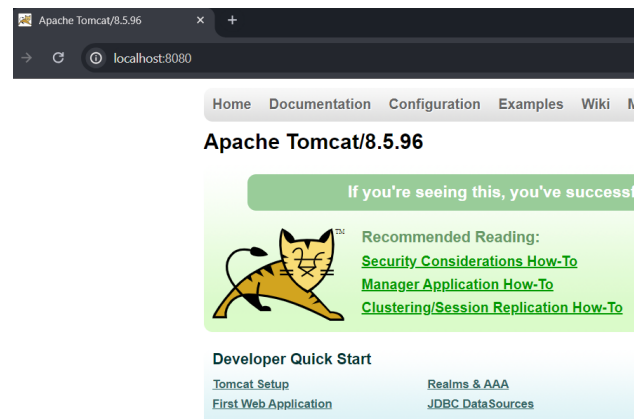


Figura 3. Descarga Apache Tomcat

- **UN SERVIDOR EN CLOUD COMPUTING:** Puede ser en Azure, Heroku, Google Engine, Amazon Web Services, o el que usted elija.



Figura 4. Instalación de heroku

III. DESARROLLO DE LA PRÁCTICA

III-A. Servidores locales

III-A1. APACHE XAMPP:

1. Cambiar los puertos para que funcionen en un puerto desde el 8080 hasta el 8089 (elegir)

- Para cambiar el puerto, ejecutamos XAMPP como administrador, posteriormente nos dirigimos a la opción Config y luego a Service and Port Settings.

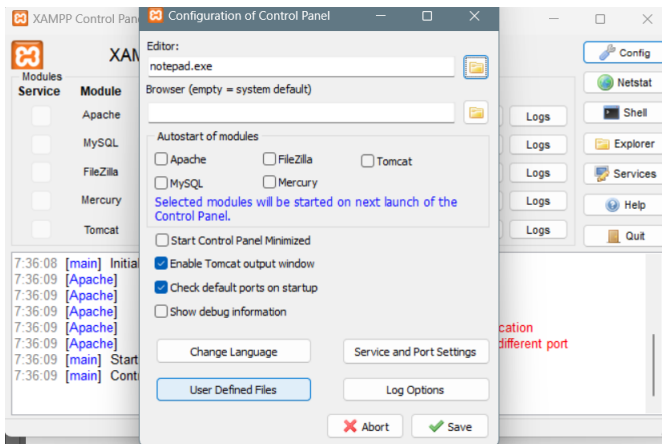


Figura 5. Descarga Apache Xampp

- En el apartado **Main Port** cambiamos nuestro puerto a 8081 y guardamos los cambios realizados.

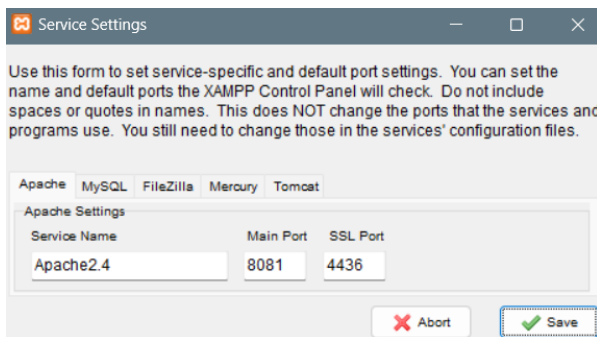


Figura 6. Configuración del puerto Apache Xampp

- En el apartado Config de Apache, ingresamos en Apache httpd.conf en donde modificaremos al nuevo puerto 8081.

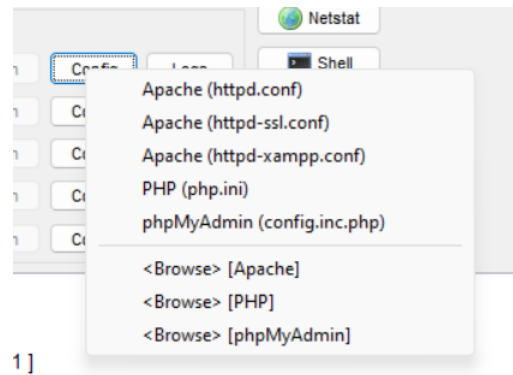


Figura 7. Configuración del puerto en el Script Apache Xampp

- Al ya tener realizado los cambios, procedemos a guardar el script.

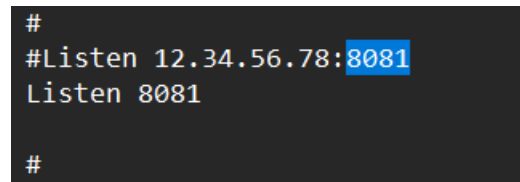


Figura 8. Configuración del puerto en el Script Apache Xampp

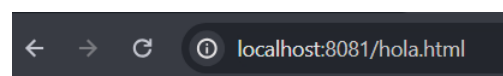
2. Desplegar una página Web sencilla de hola mundo

- Para desplegar la página Web. procedemos a crear un documento html el cuál nos permitirá mostrar el encabezado "Hola Mundo".



Figura 9. Código HTML hola.html

- Posteriormente desde apache Xampp ejecutamos como administrador y observamos la página web desplegada que nos muestra el encabezado "Hola Mundo!"



Hola Mundo!!!!

Figura 10. Despliegue de hola.html

3. Crear una carpeta que se llame "Carpeta Pública" y hacer que sea la carpeta de publicación

- Creamos una carpeta llamada "Carpeta Pública", la cual esta ubicada en el escritorio.
- Posteriormente, configuramos los permisos de seguridad de la carpeta, de tal manera que le damos todos los permisos adecuados.

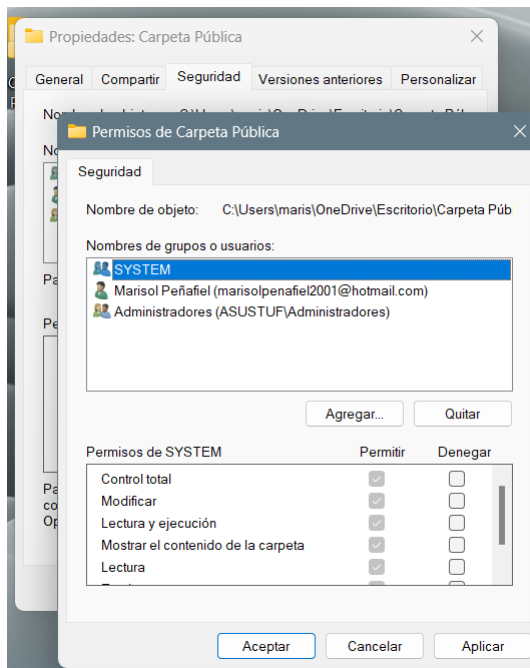


Figura 11. Permisos de Seguridad Carpeta Pública

- Procedemos a ubicarnos en la carpeta conf de apache, para modificar el archivo httpd.conf.

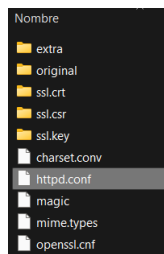


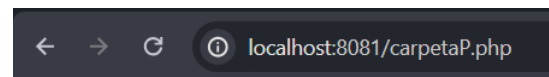
Figura 12. Archivo de Configuración

- En el Script buscamos la palabra DocumentRoot en el cuál configuraremos nuestra ubicación de destino de nuestra carpeta, de igual manera en Directory.

```
# DocumentRoot: The directory out of which you will serve your
# documents. By default, all requests are taken from this directory, but
# symbolic links and aliases may be used to point to other locations.
#
DocumentRoot "C:\Users\maris\OneDrive\Escritorio\Carpeta Pública"
<Directory "C:\Users\maris\OneDrive\Escritorio\Carpeta Pública">
#
# Possible values for the Options directive are "None", "All",
# or any combination of:
#   Indexes Includes FollowSymLinks SymLinksifOwnerMatch ExecCGI MultiViews
#
```

Figura 13. Archivo de Configuración

- Para finalizar guardamos los cambios realizados y creamos un Script en dicha carpeta para ejecutarle y comprobar su funcionamiento.



Hola Mundo estoy en mi Carpeta Pública!!!!

Figura 14. Despliegue desde la carpeta pública

4. Buscar la estructura recomendada de un árbol de archivos para publicación (En donde guarden imágenes, frontend, backend, etc). Crear la estructura de carpetas y probar qué hacer para publicar cada una de ellas. Por ejemplo: En la universidad si tuviera una carpeta de cada facultad y dentro de ellas carpetas de imágenes, css, etc. (Consultar en la bibliografía convenciones básicas de estructuras de carpetas).

Para mantener una seguridad con los datos de nuestros proyectos, es necesario mantener un orden en la creación de nuestras carpetas y archivos del documento. Para ello se debe seguir una estructura básica.

En la imagen 15

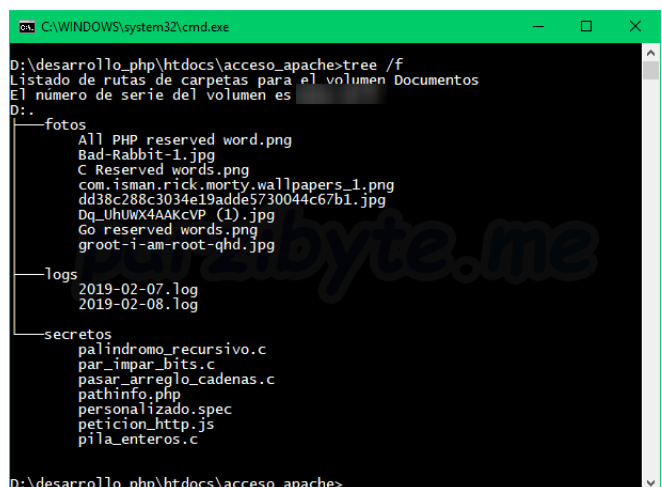


Figura 15. Árbol de Estructura

Procedemos a crear un archivo HTML en el cuál usaremos la estructura para desplegar un Hola Mundo.

- Procedemos a crear nuestra carpeta llamada Estructura Árbol en el cual crearemos nuestro proyecto.

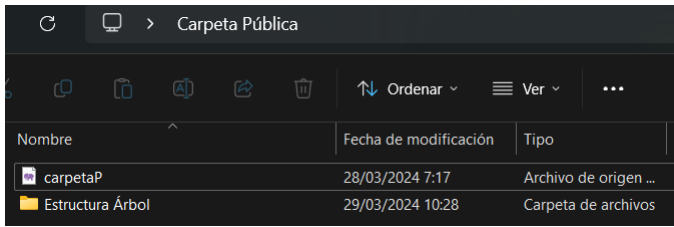
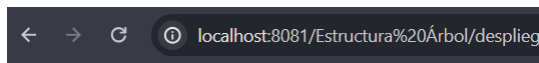


Figura 16. Carpeta para el proyecto con Estructura

- Desplegamos nuestro proyecto, obteniendo el siguiente resultado.



Uso del árbol de estructura

Se presenta una imagen de la carpeta imágenes



Se presenta un video de la carpeta videos

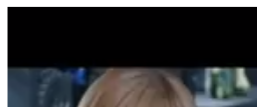


Figura 17. Despliegue con estructura

III-A2. APACHE TOMCAT:

1. **Cambiar los puertos para que funcionen en un puerto desde el 8080 hasta el 8089 (elegir)** Para cambiar el puerto, nos dirigimos a la carpeta de instalación de Tomcat, ingresamos en la carpeta bin y abrimos el archivo server.xml y lo modificamos al puerto 8085.

```
<Connector port="8085" protocol="HTTP/1.1"
connectionTimeout="20000"
redirectPort="8443"
maxParameterCount="1000"
/>
```

Figura 18. Configuración del puerto en Tomcat

2. Desplegar una página Web sencilla de hola mundo

Creamos un script html que nos permita mostrar Hola mundo desde Tomcat, procedemos a desplegarlo desde el servidor Tomcat, con las configuraciones de adecuadas del servidor.



Figura 19. Despliegue Hola Tomcat

3. **Crear una carpeta que se llame "Carpeta Pública" y hacer que sea la carpeta de publicación** Creamos una carpeta llamada Carpeta Pública, dentro del cual organizaremos nuestro proyecto. Esta carpeta debe tener los permisos adecuados.

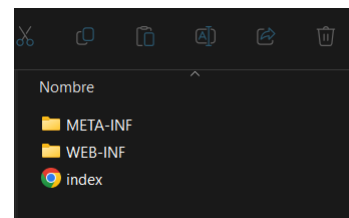


Figura 20. Carpeta Pública

Posteriormente configuramos la dirección de nuestro servidor para ser usado desde nuestro ID.

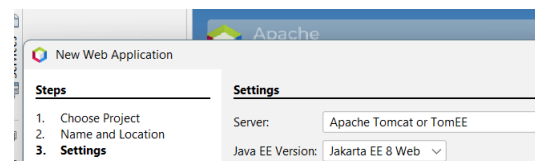


Figura 21. Configuración ruta del servidor

Finalmente ejecutamos nuestro documento html.

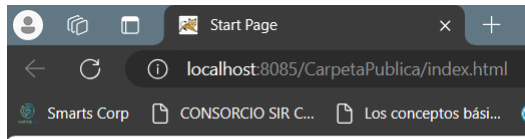


Figura 22. Despliegue Carpeta Pública

4. **Buscar la estructura recomendada de un árbol de archivos para publicación (En donde guarden imágenes, frontend, backend, etc). Crear la estructura de carpetas y probar qué hacer para publicar cada una de ellas. Por ejemplo: En la universidad si tuviera una carpeta de cada facultad y dentro de ellas carpetas de imágenes, css, etc. (Consultar en la bibliografía convenciones básicas de estructuras de carpetas).**
- Siguiendo el proceso antes mencionado procedemos a crear un archivo html en el cual usaremos una imagen y un vídeo, que serán ubicadas en carpetas específicas para su organización.

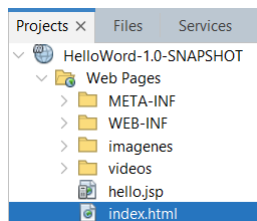


Figura 23. Estructura de las carpetas Tomcat



Figura 24. Despliegue desde la estructura

III-A3. Configurar el firewall para que sirva exclusivamente en el puerto que se quiera indicar.:

- Abrir el Panel de Control
- Seleccionar "Sistema y seguridad"
- Abrir el Firewall de Windows Defender.
- Configurar reglas de entrada: En el panel izquierdo, haz clic en "Configuración avanzada" para acceder a la configuración detallada del firewall.

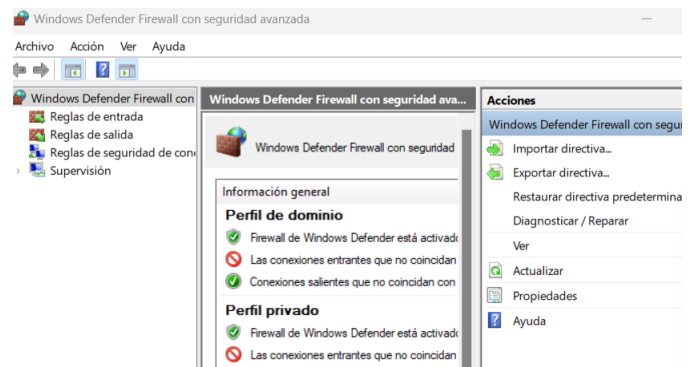


Figura 25. Configurar Firewall

Posteriormente procedemos a ejecutar index.html desde Tomcat.

- Crear una nueva regla de entrada
- Seleccionar el tipo de regla -¿Selecciona "Puerto".

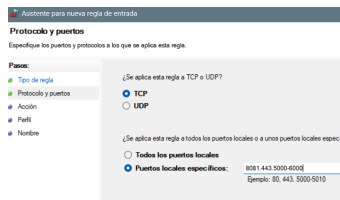


Figura 27. Selección del Puerto

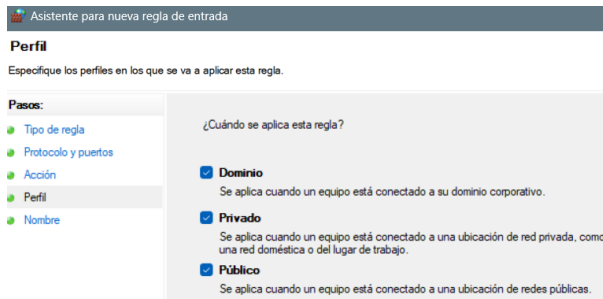


Figura 28. Dominio de alcance

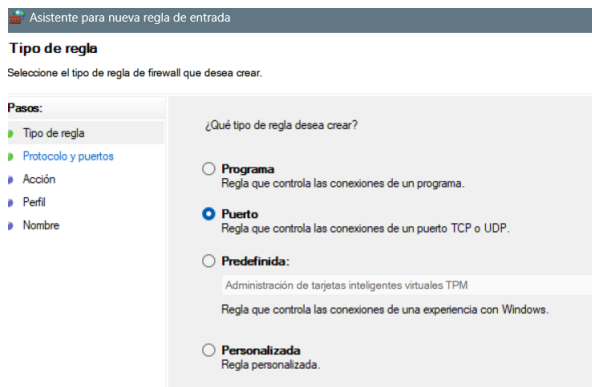


Figura 26. Crear Nueva Regla

- Especificar el puerto: Elije "TCP." o "UDP" según el tipo de tráfico que deseas permitir y especifica el número de puerto que deseas abrir.
- Elegir la acción de la regla: Selecciona "Permitir la conexión" haz clic en "Siguiente".
- Especificar cuándo aplicar la regla ya sea doméstico, público y privado.
- Al ya tener la regla, le damos un nombre. La nueva regla puede ser desactivada o activada.

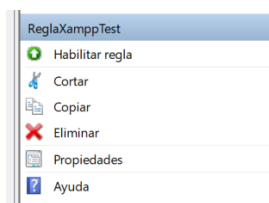


Figura 29. Regla creada

III-B. Servidor Cloud

III-B1. Buscar un servidor con trial o transacciones limitadas, revisarlos costos, restricciones de tecnología. Ver qué ofrecen y hacer una reflexión sobre dicho contenido.:

Microsoft Azure ofrece una versión de prueba gratuita, la cual permite a los usuarios probar varios servicios en la nube durante un período de tiempo limitado de 30 días. Durante este período de prueba, los usuarios tienen acceso a una cantidad limitada de créditos para gastar en los servicios de Azure, lo que les permite experimentar con diferentes recursos y funcionalidades.

La versión de prueba gratuita de Microsoft Azure incluyen:

■ Costos

Durante el período de prueba, Microsoft proporciona créditos gratuitos que permiten probar los servicios de Microsoft Azure. Algunos servicios pueden consumir créditos más rápido que otros, y una vez que se agoten los créditos gratuitos, se aplicarán cargos estándar por el uso adicional.

■ Restricciones de tecnología

La versión de prueba gratuita de Azure permite a los usuarios acceder a una amplia gama de servicios y funcionalidades de Azure, incluidos servicios de computación, almacenamiento, bases de datos, inteligencia artificial, Internet de las cosas (IoT), etc. Sin embargo hay restricciones en la cantidad de recursos disponibles o el acceso a características avanzadas.

■ Ofertas específicas

Durante prueba la prueba gratuita, Microsoft puede ofrecer promociones especiales o acceso a características premium.

La versión de prueba gratuita de Azure permite que los usuarios exploren y evalúen los servicios en la nube de Microsoft sin incurrir en costos iniciales. Esto permite a los usuarios conocer la plataforma de Azure y determinar si cumple con los requisitos y objetivos del proyecto.

Sin embargo, al finaliza el período de prueba, se aplicaran cargos estándar por el uso continuado de los servicios, por lo que es importante monitorear y administrar el consumo de recursos para evitar costos inesperados al concluir el periodo de prueba.

IV. CONCLUSIONES

En conclusión, podemos decir que un servidor web local y un servidor en la nube se diferencian por su ubicación física, el propósito y sus capacidades.

El servidor web local permite el desarrollo y pruebas de aplicaciones web en un entorno controlado, se ejecuta desde un computador. Tiene limitaciones a la red local y requiere configuraciones para ser accesible desde el internet.

En cambio, el servidor en la nube ofrece un acceso global a las aplicaciones web, ya que se ejecutan remotamente por un proveedor de servicios en la nube. Estos servidores pueden tener costos asociados que ofrecen soluciones rentables a la necesidad de alojamiento de las aplicaciones web.

REFERENCIAS

- [1] V., B. (2023) ¿Qué es un servidor web y cómo funciona?, Tutoriales Hostinger.<https://www.hostinger.es/tutoriales/que-es-un-servidor-web>. Recuperado el 26 de Marzo de 2024.
- [2] ¿Qué es nginx y cómo funciona? (2022) Kinsta@.<https://kinsta.com/es/base-de-conocimiento/que-es-nginx/> Recuperado el 26 de Marzo de 2024.
- [3] León, Á.D. (2019) Servidor IIS: ¿Qué es? Características, Ventajas, Requisitos E Instalación, Infranetworking.https://blog.infranetworking.com/servidor-iis/#Que_es_IIS. Recuperado el 26 de marzo de 2024.
- [4] Casero, A. (2024) ¿Qué son los servidores de aplicaciones?, KeepCoding Bootcamps.<https://keepcoding.io/blog/que-son-los-servidores-de-aplicaciones/>. Recuperado el 26 de marzo de 2024.
- [5] JOnAS, D. (no date) Jonas, Wikiwand.<https://www.wikiwand.com/es/JOnAS>. Recuperado el 26 de marzo de 2024.
- [6] Mora, R.C. (2018) Curso jboss de red hat, Adictos al trabajo.<https://www.adictosaltrabajo.com/2014/09/19/curso-jboss-de-red-hat/>. Recuperado el 26 de marzo de 2024.
- [7] ANW Hosting (no date) ¿Qué es apache tomcat? El Servidor de Aplicaciones Java más popular., ANW.<https://www.anw.es/servidores/apache-tomcat.html>. Recuperado el 26 de marzo de 2024.
- [8] Coppola, M. (2023) Qué es un hosting: Características, Tipos y Ejemplos, Blog de HubSpot. <https://blog.hubspot.es/website/hosting>. Recuperado el 28 de marzo de 2024.
- [9] ¿Qué es cloud computing?. <https://www.salesforce.com/mx/cloud-computing/>. Recuperado el 28 de marzo de 2024.
- [10] Casero, A. (2024) ¿Qué es el despliegue continuo en el desarrollo web?, KeepCoding Bootcamps.<https://keepcoding.io/blog/despliegue-continuo-en-el-desarrollo-web/#:~:text=Para%20comprender%20completamente%20el%20despliegue%20continuo%2C%20es%20esencial,aprobados%20y%20verificados%20en%20el%20c%C3%B3digo%20fuente.%20>. Recuperado el 28 de marzo de 2024.