|  |  |  |
| --- | --- | --- |
| BurbujonDeLosHorrores | | |
|  |  |  |
|  | | |

Índice

[Configuración de Switch 2](#_Toc65266994)

[CMMS 4](#_Toc65266995)

[Oracle Database 19c 8](#_Toc65266996)

[SQL Developer 9](#_Toc65266997)

[Instalacion Symfony 10](#_Toc65266998)

[Creacion de sCript De requisitos(Librerias) 11](#_Toc65266999)

[Creacion de servicio que lanza webservice 12](#_Toc65267000)

[Creacion de webservice 14](#_Toc65267001)

[Hardware 15](#_Toc65267002)

[REPLICACIÓN DFS 16](#_Toc65267003)

[REDIRECCIONAMIENTO CARPETAS (PERFILES MÓVILES) 18](#_Toc65267004)

[INSTALACIÓN DE SOFTWARE DE FORMA REMOTA 20](#_Toc65267005)

[Diagrama Entidad-Relacion y paso a tablas 21](#_Toc65267006)

[Los 6 servidores el CPD 22](#_Toc65267007)

[Esquema de red 23](#_Toc65267008)

[BIBLIOGRAFíA 24](#_Toc65267009)

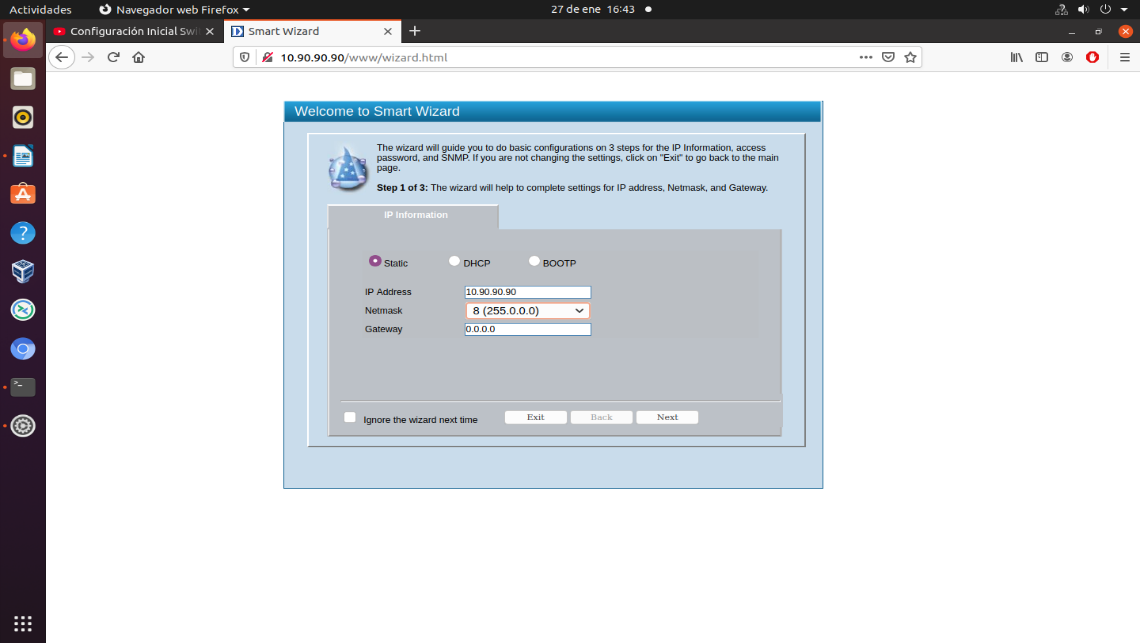
# Configuración de Switch

Para poder acceder al switch tenemos que reiniciar el switch introducir un clip o algo puntiagudo en el agujero donde pone reset y estar 10 segundos.

Una vez realizado el reset tenemos que configurar la red poniendo en la ip 10.90.90.x

Mascara 255.255.255.0 y la puerta de enlace 10.90.90.90.

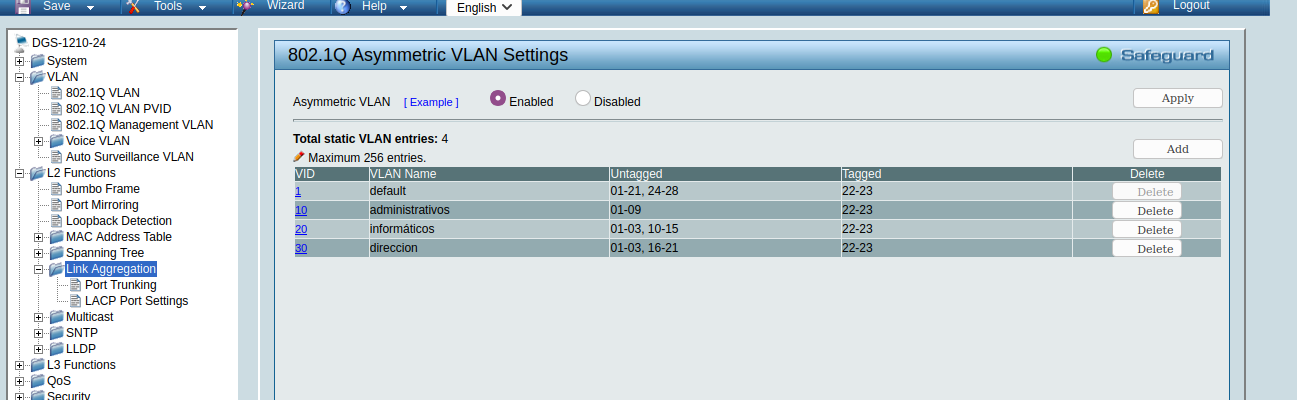
Una vez hecho eso introducimos en el buscador 10.90.90.90 y ya entraremos en el switch, nos pedirá una contraseña en la que tendremos que poner admin porque es la contraseña base del switch.



Ya dentro de la configuración del switch para hacer las Vlan tenemos que ir al apartador Vlan y dentro a 802.1Q Vlan. Asignaremos los grupos de vlan, el nombre y el rango de puerto de cada vlan y también asignaremos en cada vlan los puertos de port cannel y los ponemos en modo tangged, los puertos de las vlan en untagged.

Luego tendremos que ir a la opción vlan en 802.1Q Vlan PVID, asignaremos a los puertos la vlan a la que pertenece.

Y por último tendremos que asignar los puertos que van a ser port channel para eso tendremos que ir Vlan>L2 Functions>Link Aggregation>Port Trunking y pondremos los puertos que queremos hacer port channel.



# CMMS

**REQUISITOS:**

-Necesitamos una máquina virtual de win 10 de 32 bit.

-Necesitamos la carpeta CalemEAM que la descargamos de este link:

<https://sourceforge.net/projects/calemeam/files/CalemEAM%20Community%20Edition/Calem%20Community%20Edition%20R2.1e/>

-Necesitamos la versión 1.7.2 de XAMPP, que la descargaremos en el siguiente link:

<https://sourceforge.net/projects/xampp/files/XAMPP%20Windows/1.7.2/>

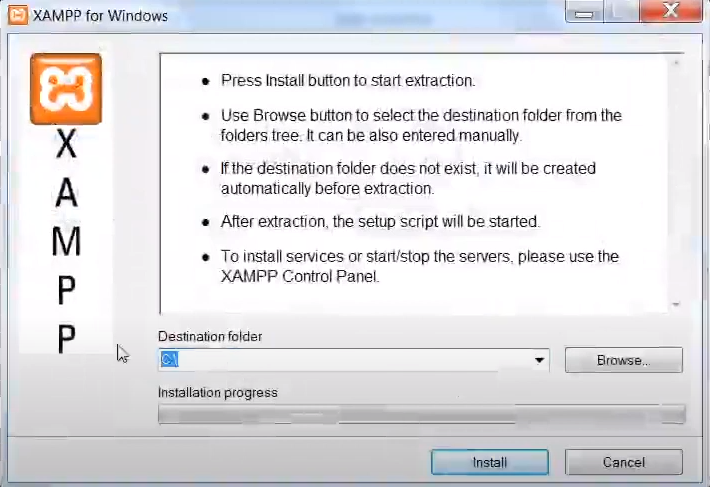
-Necesitamos el programa winrar

**PROCEDIMIENTO:**

-Lo primero que debemos hacer es instalar el servidor web XAMPP que nos aportará a nuestro buscador las herramientas apache y mysql.

Para ello nos metemos en el link que previamente he especificado y elegimos descargar el siguiente archivo:

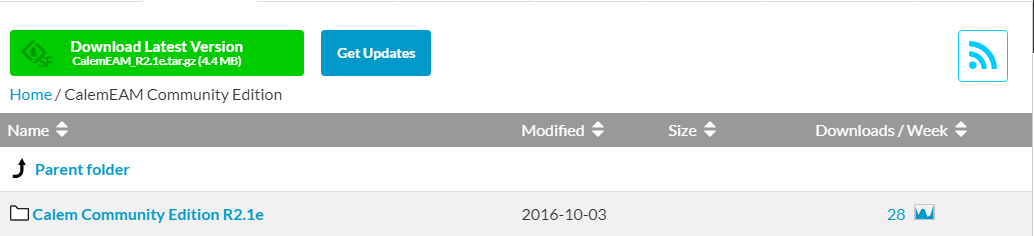
-Después de descargarlo lo ejecutamos y comenzamos la instalación del **XAMPP**



-Una vez instalemos el xampp abrimos el panel de control de xampp y arrancamos los servicios de apache y mysql.

-Una vez arrancados los servicios comenzamos con la descarga e instalación de la herramienta cmms(CalemEAM) ,para ello, accedemos al siguiente link:

<https://sourceforge.net/projects/calemeam/files/CalemEAM%20Community%20Edition/>

-Abrimos la siguiente carpeta

-Instalaremos el winrar para poder descomprimir el archivo descargado.

-Este es el archivo que debemos descomprimir para poder comenzar con la instalación de la herramienta CMMS



-Una vez descomprimido el archivo nos saldrá una carpeta llamada CalemEAM, está la cortaremos y pegaremos en la carpeta **C:/xampp/htdocs**

-Después de esto, abrimos la shell de mysql desde el panel de control de xampp y procedemos a crear una base de datos en la cual almacenaremos los datos del CalemEAM,

Para esto introducimos el siguiente comando:

**mysql -u root -p**

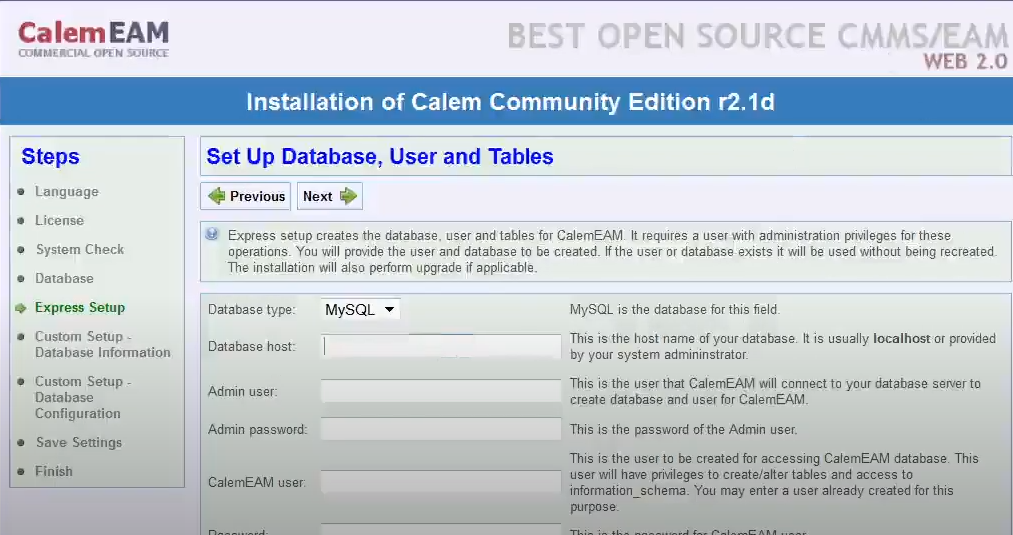
**create database calemeam**

**exit**

-Después de esto accedemos al buscador e introducimos la siguiente ruta:

**localhost/CalemEAM/installation/**

-Si todo está bien nos accederá a la pantalla de instalación del CalemEAM

****

-Introducimos los credenciales de la base de datos, en el database host introducimos (localhost), en el admin user introducimos (root), en la contraseña de admin de db no introducimos nada ya que el usuario root predeterminado de mysql no tiene ninguna password, en el calemeam user introducimos (admin), en la contraseña de este (admin), después introducimos el nombre de la base de datos creada y la damos a siguiente.

-Después de terminar la instalación debemos borrar la carpeta installation y acceder al siguiente link:

<http://localhost/CalemEAM/>

-Introducimos los credenciales predeterminados

USER: **admin**

PASSWORD: **admin\_password**

-Una vez terminado el logue podemos comenzar a utilizar la herramienta.

# Oracle Database 19c

Para empezar, descargamos Oracle Database de la página de Oracle, esta nos proveerá de la versión más reciente (en este caso la 19c), para ello será necesario registrarse proporcionando un email y un teléfono.

Antes de proceder es necesario crear un nuevo usuario de Windows que requerirá un usuario y una contraseña y asegurarse de que no es creada una cuenta de administrador.

Una vez descargado y descomprimido el instalador lo ubicamos en una carpeta dedicada en C (para que todos los usuarios puedan verla)

Para esto es necesario iniciar sesión en Windows con el nuevo usuario y volver a la carpeta en la que habíamos descomprimido el instalador para ejecutar setup.exe con permisos de administrador.

En el primer paso seleccionamos crear y configurar una base de datos de instancia única y pasamos al segundo donde seleccionaremos la clase de escritorio para que sea tan eficiente como pueda. En el tercer paso escogeremos una cuenta Windows existente e introduciremos el nombre de usuario y contraseña creados anteriormente y, en el paso posterior, escogeremos la ruta en la que deseamos realizar la instalación. Para la base de Oracle la ruta será C:\OracleBase19c y es importante quedarse con el nombre de la base de datos global (orcl por defecto) y escoger una contraseña alfanumérica y para terminar otro dato importante es el de la base de datos insertable (orclpdb por defecto).

En el siguiente paso el instalador comprobara los requisitos previos, incluida la RAM disponible y, una vez realizada esta tarea, nos mostrara y dejara exportar los datos de la instalación, si seleccionamos instalar comenzara el proceso cuya duración variara en base al hardware. El firewall de Windows bloqueara algunas funciones del instalador a no ser que, cuando lo pregunte, se le conceda al instalador permiso para comunicarse. Una vez finalizada la instalación nos dará una URL para conectar mediante el explorador (<https://localhost:5500/em>)

# SQL Developer

Desde SQL PLUS (incluido con la instalación de Oracle Database 19c) podemos iniciar sesión introduciendo: ‘/’ as sysdba

Después crearemos un nuevo usuario común (contienen c## como prefijo): create user c##scott identified by tiger;

Y le concederemos los siguientes permisos con: grant resource,contect,dba to c##scott;

Tras salir volveremos a entrar, pero esta vez como c##scott y contraseña tiger para verificar que podemos contarnos con este usuario y, de ser así, podremos cerrarlo y proceder a conectar SQL Developer.

En la esquina superior izquierda de SQL Developer veremos un gran + verde y si lo pulsamos nos saldrá la ventana para realizar una nueva conexión a una base de datos.

El nombre de la conexión es arbitrario (pero no puede dejarse vacío), el usuario y su contraseña deben ser los creados antes desde SQL PLUS.

El tipo de conexión será básica, el nombre del host localhost, el puerto será el 1521 y el SID será orcl (si no hemos cambiado ninguno de estos valores durante la instalación de Oracle Database19c). Y con esto tendríamos acceso completo a la base de datos desde SQL Developer.

# Instalacion Symfony

Lo primero necesitamos una máquina virtual ubuntu server.

Una vez dentro instalamos todos los módulos de php 7.2:

* **apt install php7.2 php7.2-cli php7.2-common php7.2-mbstring php7.2-gd php7.2-intl php7.2-xml php7.2-mysql php7.2-zip php7.2-pgsql php7.2-dev php7.2-bcmath php7.2-xsl php7.2-xmlrpc**

Después de instalar los módulos instalamos mcrypt:

* **apt install mcrypt**

Luego actualizamos las librerías:

* **apt upgrade**

Lo siguiente que haremos es instalar php-mbstring phpunit:

* **apt install php-mbstring phpunit**

Más adelante instalaremos los módulos que faltan de php:

* **apt install php7.2-common php7.2-cli php7.2-mysql php7.2 php7.2-sqlite3 php7.2-json php7.2-opcache php7.2-readline libapache2-mod-php7.2 php7.2-curl**

Instalaremos el git, curl y composer que son gestores de dependencias

* **apt install git curl composer**

Lo siguiente será crear un proyecto

* **composer create-project symfony/website-skeleton LIBRERÍA**

# Creacion de sCript De requisitos(Librerias)

Normalmente en una página web no es necesario instalar ninguna librería ya que el servidor tiene todo lo básico para el funcionamiento y creación de servicios pero en nuestro caso al haber elegido un framework el cual está orientado a objetos además de creación de APIs complejas deberemos de instalar ciertas librerías para el correcto funcionamiento, por lo tanto sabiendo esto y teniendo en cuenta que estas librerías son necesarias instalarlas la primera vez que se pone en marcha el servidor decidimos hacer un script y olvidarnos de estar atento de que cada vez que se acabase la instalación ejecutáramos la siguiente. En nuestro caso al tener dos máquinas una para la página y otra para el webservice (concretamente la máquina que hace las consultas a la BBDD, ya que esto nos pareció más seguro en vez que el cliente de por si accediese a la BBDD directamente, de esta forma el cliente hace la petición al webservice y este se encargara de manejar esa petición, podríamos llamarlo “firewall”, pero como todo tiene sus fallos etc.). Por lo tanto tendremos dos scripts el del webservicerequisites.sh y requisites.sh. Hay que mencionar que se instala el web-server-bundle en los dos proyectos para poder lanzar el servidor.

requisites.sh:

#!/bin/bash

composer require symfony/web-server-bundle --dev ^4.4.2

composer require annotation

composer requeri symfony/doctrine-bundle

Crearemos el script para el web service y el siguiente para el web-page

webpagerequisites.sh:

#!/bin/bash

composer require sensio/framework-extra-bundle

composer require jms/serializer-bundle

composer require friendsofsymfony/rest-bundle

# Creacion de servicio que lanza webservice

Para crear este servicio que queremos que se pueda también detener además, de iniciar cuando el servidor se inicie, necesitaremos dos scripts .sh (ya que el webservice está definido en una maquina ubuntu)

En el primer script que será el de inicio escribiremos lo siguiente:

#!/bin/bash

cd /home/nombreUsuario/NombreCarpetaProyecto

git pull origin

cd /home/nombreUsuario/NombreCarpetaProyecto/WS

php bin/console server:run 192.168.4.x:8000

En estas líneas básicamente lo que estamos haciendo es acceder a la ruta donde está el proyecto actualizarlo en el branch principal (original) y ejecutar el webservice en modo servidor en una ip específica y en un puerto especifico como en este caso.

En el segundo script que servirá para parar el servicio escribiremos lo siguiente:

#!/bin/bash

cd /home/nombreUsuario/NombreCarpetaProyecto/WS

php bin/console server:stop

Este script es más corto y sencillo y después de haber visto el de antes, es fácil deducir que lo que hacemos es acceder a la carpeta y parar el webservice (o servidor) que esta lanzado en la ip específica y el puerto especifico de antes.

Lo siguiente será ubicar estos dos archivos en la siguiente ruta: /usr/local/bin. Les daremos permiso de ejecución, es decir hacer un chmod 755 nombreDelArchivo.sh.

Por ultimo en /etc/systemd/system crearemos un archivo .service (el archivo que se ejecutara nada más iniciar el ordenador o más conocido como servicio como su propio nombre indica).

En este archivo lo que escribiremos será lo siguiente:

[Unit]

Description = descripcion que quieras poner

After=networking.target

[Service]

Type=oneshot

ExecStart=/usr/local/bin/nombreArchivo.sh

ExecStop=/usr/local/bin/nombreArchivo.sh

RemainAfterExit=yes

[Install]

WantedBy=multi-user.target

De esta forma definimos que script se tiene que ejecutar dependiendo si el ordenador se está iniciando o estemos diciendo que queremos parar el servicio etc.

Antes de darlo por finalizado haremos ciertas comprobaciones de si el servicio está funcionando bien, las comprobaciones son las siguientes:

systemctl start nombreServicio.service

systemctl status --no-pager nombreServicio.service

systemctl stop nombreServicio.service

Si todo va bien y en las comprobaciones dice que el servicio está activado, lo último que habrá que hacer es habilitar el servicio para que cuando se reinicie el servicio se inicie automáticamente, y el comando es el siguiente:

systemctl enable nombreServicio

# Creacion de webservice

Teniendo en cuenta que solo hay que recibir información, tratarla para después devolver esa información tratada (idea básica de un webservice), tenemos que tener en mente tres cosas la primera que no vamos a necesitar diseño ya que no necesitamos mostrar nada por pantalla, segundo que las peticiones se envían por métodos concretos (POST, GET, etc.) por lo tanto hay que tener en cuenta que datos queremos que se muestren en la url ya que no queremos inyecciones sql (este la diferencia entre post y get, de todas formas sigue habiendo formas de acceder a la bbdd sin necesidad de que utilicemos esto), y por ultimo si hay alguna forma de generar menos accesos es decir menos urls a las que hacer peticiones (ejemplo: <http://192.168.4.x:8000/crearModUsu>) y podemos juntar distintas acciones en una misma url el funcionamiento será más eficaz ya que hay menos código ejecutar por cada petición.

Por lo tanto teniendo en cuenta esto generaremos una función publica, que va a ser la que más vamos a utilizar y es la devolución de los datos en formato json, a partir de ahí solo necesitamos crear la funciones que básicamente lo que harán es recibir los datos, ver que es lo que hay que hacer con ellos es decir, insertar, actualizar, eliminar, mostrar, etc. Después generar la respuesta en este caso un array y llamar a la función que la convierte en json y la devuelve. El más complicado de tratar ha sido el de la creación y modificación de usuario ya que al tener dos tipos de usuarios (Cliente y Trabajador) se ha tenido que crear una petición que especifica que tipo de acción se va a realizar y que tipo de usuario va a ser el que va a recibir esa acción. Otras de las complicadas ha sido la petición de creación y modificación de ya que estas tienen id en el diseño de la bbdd, en cambio en la petición llega el nombre, de esta forma tienes que hacer dos consultas y recorrer dos tipos de datos distintos la tabla artista y la tabla maquetas (otra forma hubiera sido crear una clase propia para después crear un dql y extraer el dato mediante el nombre y aunque en eficacia sea más óptimo decimos que no íbamos a implementarlo, al menos de momento)

# Hardware

TORRE CERRADA:

gigabyte H55M-S2H LGA1150 + intel HD graphics

intel core i3 540

ram ddr3 8gb (2x4gb) 2133mhz

hdd: SEAGATE BARRACURA 7200RPM ST3500418AS

TORRE ABIERTA:

GIGABYTE GA-H81M-S2H LGA1150 + intel HD graphics

ram ddr3 8gb (2x4gb) 2133mhz

hdd SEAGATE BARRACUDA 7200RPM ST3170827AS

Intel(R) Core(TM) i5-4460 CPU@ 3.20GHz

SWITCH X2:

D-LINK DGS-1210-24

# REPLICACIÓN DFS

Primero agregamos el rol de administración DFS, para ello accedemos al apartado de agregar roles y características y dentro de este apartado buscamos la opción de administración DFS y la de replicación DFS.

Una vez terminemos la instalación del rol reiniciamos el ordenador y comenzamos a configurar la herramienta de DFS, el primer paso es crear un espacio de nombre, para ello vamos a herramientas y abrimos la herramienta **Administración de DFS**

Una vez abierta esta herramienta seleccionamos la opción **Crear nuevo espacio de nombres,** después pondremos el servidor que queremos que sea propietario.

Le damos a siguiente y a continuación ponemos el nombre que aparecerá a la hora de abrir la raíz en este caso (**oficina**).

Escogemos la opción de espacio de nombre basado en dominio y formará la siguiente raíz:

**\\burbujondeloshorrores.com\oficina.**

Después de esto le damos a crear, una vez configurados ambos servidores, en nuestro caso añadimos el S2 al espacio de nombres, como copropietario de este, después de acabar con este proceso comenzaremos con la replicación.

Para comenzar con la configuración de la **replicación DFS**, debemos elegir la opción **agregar nuevo grupo de replicación**, una vez hecho esto escogemos la opción **Grupo de replicación multipropósito.**

Seguido a esto llegaremos a la opción **Miembros de replicación,** en este punto debemos poner el nombre del grupo de replicación, que en nuestro caso es “**Replicación DFS”,** le damos a siguiente, nos aparecerá una pestaña en la cual insertamos qué servidores queremos que formen parte de nuestra replicación, ahí agregaremos los servidores S1,S2, después de esto le damos a siguiente, nos aparecerá la ventana **Programación del grupo de replicación y ancho de banda**, ahí elegimos la primera opción (Replicar de forma continua usando el ancho de banda especificado), siguiente a este paso elegiremos el miembro principal, que en nuestro caso es “S1” , ahora elegiremos la ruta de la raíz para que la replique, para eso le damos a examinar y elegimos la ruta de nuestro espacio de nombres, en nuestro caso sería (**C:\\Raíces DFS\oficina**).

Una vez hechas todas las configuraciones le damos a siguiente confirmamos la instalación y finalizamos el proceso

# REDIRECCIONAMIENTO CARPETAS (PERFILES MÓVILES)

En este proyecto decidimos usar la directiva de grupo de redireccionamiento de carpetas para realizar los perfiles móviles de ciertos usuarios y de apartados distintos de su cuenta de cliente, por ejemplo en el grupo **administración** decidimos redireccionar el escritorio y las descargas, eso quiere decir que cada vez que un usuario de este grupo inicie sesión en el dominio y cree algún archivo o carpeta, éstos nos aparecerán dentro de una carpeta en la raíz DFS con el nombre del usuario que inicie sesión.

Para ello previamente debemos tener bien configurada y compartida la raíz DFS (que previamente hemos explicado),primero daremos permisos de control total a la carpeta oficina de los dos servidores, esto se hará en el apartado propiedades, compartir, uso compartido avanzado, permisos, y seleccionamos todos, a estos les habilitaremos la opción control total. Dentro de esta debemos crear carpetas aplicadas a los grupos que queremos que redireccione los datos, en nuestro caso lo hicimos con todos los grupos del dominio, aunque variamos los apartados que queríamos que se redireccionaran

Para ello dentro de la raíz \**\burbujondeloshorrores.com\oficina** hacemos click derecho seleccionamos nuevo y elegimos la opción carpeta, le ponemos el nombre del grupo que queremos redireccionar y damos los siguientes permisos:

**- Mostrar carpeta o leer datos (permisos avanzados)**

**- Crear carpetas o anexar datos (permisos avanzados)**

**- Leer atributos (permisos avanzados)**

**- Leer atributos extendidos (permisos avanzados)**

**- Permisos de lectura (permisos avanzados)**

**- Atravesar carpeta o ejecutar archivo (permisos avanzados)**

Una vez compartida y configurada la carpeta del grupo a redireccionar debemos crear la directiva de grupo, para ello vamos al **administrador de servidor**, abrimos las herramientas y elegimos la opción **administración directivas de grupo**.

Haz clic con el botón derecho en el dominio donde quieres configurar Redirección de carpetas y, después, selecciona **Crear un GPO en este dominio y vincularlo aquí**,

En el cuadro de diálogo Nuevo GPO, escribe un nombre para el GPO (por ejemplo, Configuración de Redirección de carpetas) y selecciona Aceptar.

Selecciona la GPO. En la sección Filtrado de seguridad de la pestaña Ámbito, selecciona Usuarios autenticados y, a continuación, seleccionamos **Quitar** para evitar que la GPO se aplique a todos los usuarios.

Ahora comenzaremos con la configuración final de la directiva de redireccionamiento:

En Administración de directivas de grupo, hacemos clic con el botón derecho en la GPO que creamos (por ejemplo, Configuración de Redirección de carpetas) y, a continuación, seleccionamos Editar.

En la ventana Editor de administración de directivas de grupo, navegamos hasta **Configuración de usuario**, desplegamos **Directivas**, **Configuración de Windows** y **Redirección de carpetas**.

Haz clic con el botón derecho en una carpeta que quieras redirigir (por ejemplo, Documentos) y, a continuación, seleccionamos **Propiedades**.

En el apartado **Propiedades**, en el cuadro **Configuración**, selecciona la opción **Avanzada**. Le damos al botón agregar agregamos el grupo de seguridad al que queremos aplicar la directiva, después en la sección **Ubicación de la carpeta de destino**, selecciona **Crear una carpeta para cada usuario en la ruta raíz** y, a continuación, en el cuadro **Ruta de acceso raíz**, escribe la ruta de acceso a la raíz DFS, donde se almacenan las carpetas redirigidas, como, por ejemplo: [**\\burbujondeloshorrores.com\oficina\$user**](file:///\\burbujondeloshorrores.com\oficina\$user)

Después de esto, si hemos realizado todos los pasos bien debemos abrir una **cmd(Win +r),**

Abrimos la terminal e introducimos el siguiente comando:

**gpupdate/force** (Así aplicaremos al momento las directivas de grupo, recomendable hacerlo en todos los servidores del dominio)

Para comprobar que todo está bien, abrimos una máquina cliente conectada al dominio y creamos algo en algún apartado redireccionado (ejemplo: Documentos).

# INSTALACIÓN DE SOFTWARE DE FORMA REMOTA

Creamos una carpeta de red compartida en la raíz DFS, donde colocaremos el paquete de Windows Installer (archivo .msi) que quieras distribuir.

Establecemos permisos en el recurso compartido para permitir el acceso al paquete de distribución.

Dentro de la carpeta metemos los archivos .msi que queramos aplicar en la directiva, en nuestro caso **firefox.msi** y **blender.msi**

Una vez creada y compartida la carpeta debemos ir a las directivas de grupo y crear una nueva directiva dentro de nuestro dominio, en la carpeta donde se encuentren los usuarios a los que queramos aplicarle esta (por ejemplo Edificio), la cual llamaremos “software”, una vez creada le daremos clic derecho y elegiremos la opción editar y nos ubicamos en la siguiente ubicación de directiva:

**Configuración de usuario/directivas/ajustes de software/instalación de software**

Comprobamos que en la ventana filtrado de seguridad tenemos añadidos al grupo **Usuarios autentificados.**

Una vez ubicados en esta ruta hacemos clic derecho sobre instalación de software, seleccionamos **nuevo y paquete**, nos abrirá la ruta de red, donde debe aparecernos la carpeta que previamente hemos creado y compartido en la raíz DFS, de la cual la directiva cogerá los paquetes .msi para poder cumplir la instalación remota.

Una vez seleccionemos el instalador nos saldrá una ventana en la cual nos pedirá que especifiquemos qué configuración queremos que tenga, debemos elegir la opción **avanzada.**

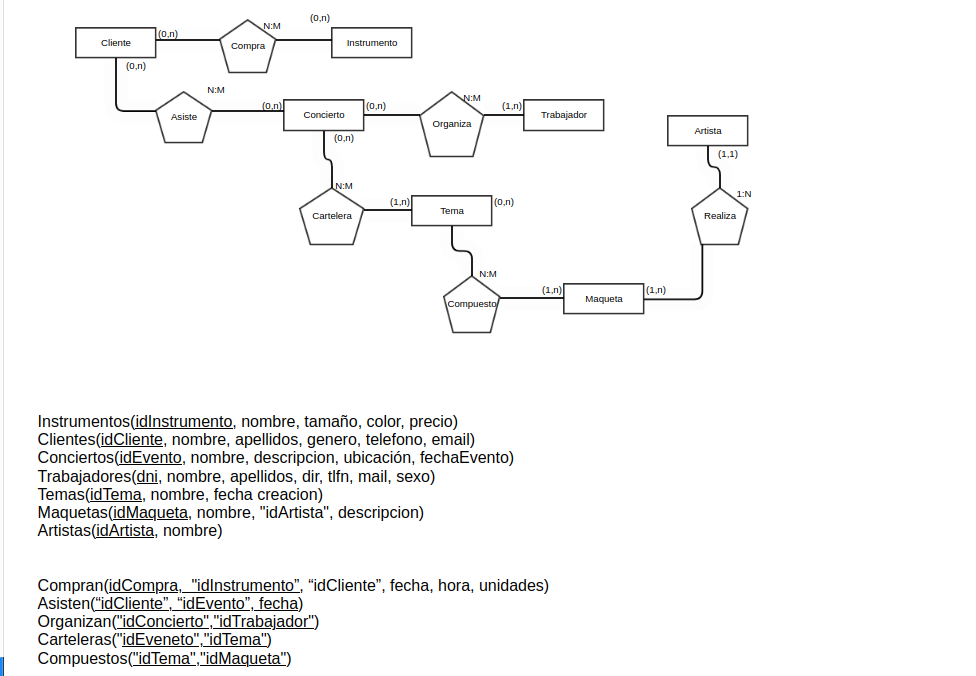
A continuación se nos abrirá una ventana de configuración, elegimos la opción tipo de despliegue y seleccionamos asignado, en esa ventana habilitamos también la casilla de instalar aplicación al inicio de sesión.

Una vez hechos todos los pasos, abrimos la cmd e introducimos el comando:

**gpupdate/force**

Nos cerrará la sesión del server y aplicará los cambios de la directiva, con este paso hecho, abrimos una máquina cliente conectada al dominio y comprobamos que según el cliente inicie sesión se le instalen los paquetes .msi que previamente configuramos.

# Diagrama Entidad-Relacion y paso a tablas

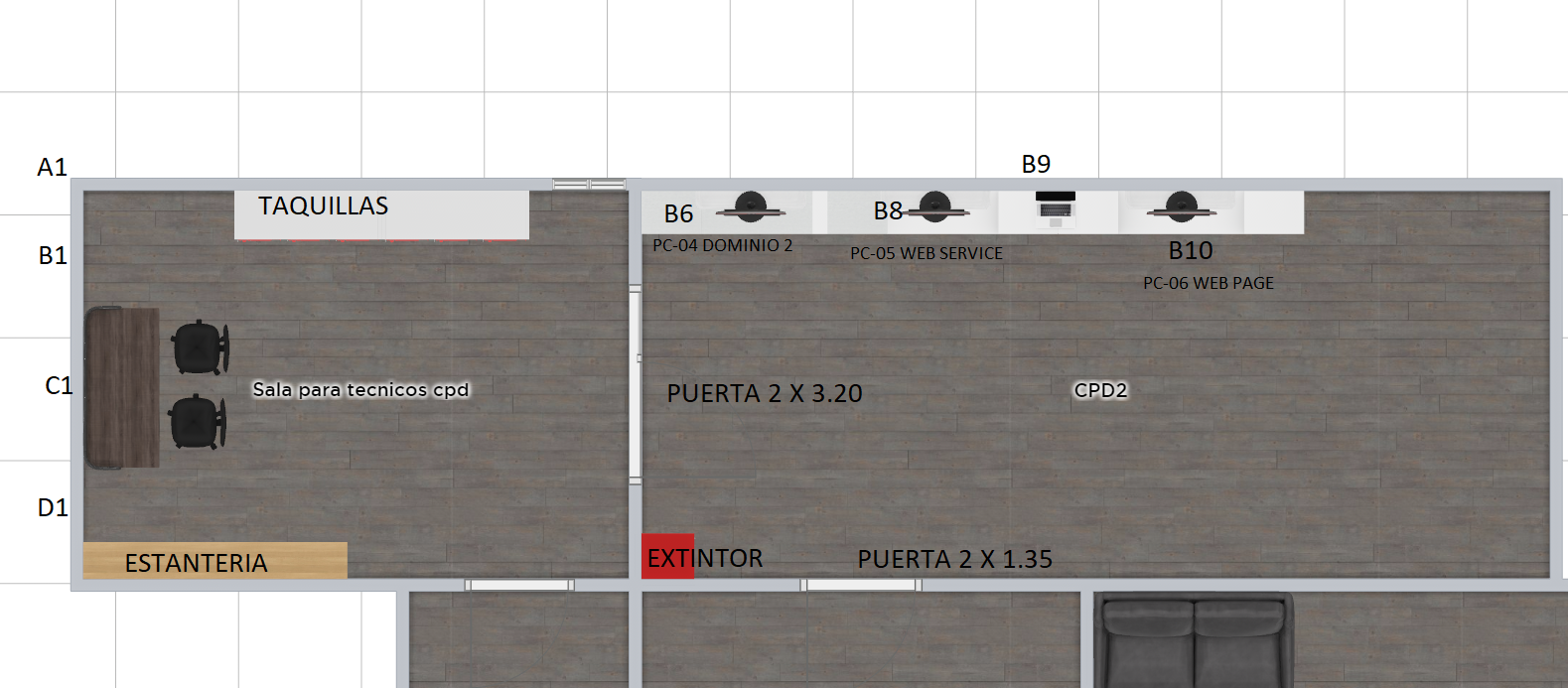


# Los 6 servidores el CPD

WebService y Webpage son servidores Ubuntu dedicados independientes ya que reciben muchas más solicitudes que el resto y se ubican, junto a S2, en la planta superior.

La base de datos está en un sistema Windows de 64 bits por que requería el uso de SQL Developer y el CMMS está en otro sistema Windows pero de 32 bits (ya que el CMMS escogido parecía tener alguna incompatibilidad con los sistemas de 64 bits) y ambos se encuentran en la planta inferior junto a S1.

S1 y S2 contienen ambos el dominio y están asignados uno a cada planta pero, en caso de no estar cualquiera de ellos disponible, los equipos se conectarán al otro.

Todo ello constituye el Centro de Procesamiento de Datos.

El cpd básicamente esta formado en dos plantas, en la planta baja está diseñado para los tres primeros server, el primer dominio, el server BBDD y para el server de CMMS. En la planta de arriba estarían ubicados los otros 3 ordenadores el segundo dominio, y los servidores para el web-page y web-service. Las puertas las hemos elegido con una altura de 3.2metros para que entren los racks, servidores, el material etc… Las dos habitaciones del CPD están dotadas de 2 extintores, taquillas para los empleados, oficinas para los empleados y un almacén para cada cpd.

# Esquema de red

# BIBLIOGRAFíA

DESCARGA CALEMEAM

<https://sourceforge.net/projects/calemeam/files/CalemEAM%20Community%20Edition/Calem%20Community%20Edition%20R2.1e/>

DESCARGA XAMPP

<https://sourceforge.net/projects/xampp/files/XAMPP%20Windows/1.7.2/>

Symfony:

<https://symfony.com/doc/current/setup.html>

# Trello

