Cristóbal Suárez Abad

Implantación de aplicaciones web  2º ASIR

Despliegue de una aplicación web completa

**Índice**

[**0. Introducción. 2**](#_Toc211808474)

[**1. IP estática. 4**](#_Toc211808475)

[**2. Instalación de LAMP 5**](#_Toc211808476)

[**a. Apache. 5**](#_Toc211808477)

[**b. MySQL. 6**](#_Toc211808478)

[**c. PHP. 8**](#_Toc211808479)

[**d. phpMyAdmin 9**](#_Toc211808480)

[**3. Clonación de GitHub. 12**](#_Toc211808481)

[**a. Renombrar carpeta: 12**](#_Toc211808482)

[**4. Instalar VSCodium o Code. 13**](#_Toc211808483)

[**a. Instalamos extensión Idioma Español. 14**](#_Toc211808484)

[**b. Instalamos extensión SSH. 14**](#_Toc211808485)

[**c. Acceder a la carpeta creada desde el terminal de VSCodium. 14**](#_Toc211808486)

[**5. MySQL: crear base de datos y usuario. 16**](#_Toc211808487)

[**a. phpMyAdmin. 16**](#_Toc211808488)

[**b. Terminal MySQL. 18**](#_Toc211808489)

[**6. Ejecutar script biblioteca.sql 19**](#_Toc211808490)

[**7. Configuración de VirtualHost. 20**](#_Toc211808491)

[**8. Configurar el acceso a la base de datos desde la aplicación. 22**](#_Toc211808492)

[**9. Activar modulo rewrite. 23**](#_Toc211808493)

[**10. Configuración del módulo rewrite en apache2.conf. 24**](#_Toc211808494)

[**NOTA: pasar todos los archivos al usuario de apache2. 25**](#_Toc211808495)

[**NOTA: Poner la dirección y el nombre de la página en “hosts”. 26**](#_Toc211808496)

[**11. Verificar el buen funcionamiento de la aplicación. 27**](#_Toc211808497)

[**12. Subir el proyecto completo a tu perfil de GitHub. 29**](#_Toc211808498)

[0. Introducción.](#A00_Indice)

*Vamos a instalar la aplicación Biblioteca, una aplicación web a medida para llevar el control de los préstamos en una biblioteca. Para realizar la instalación sigue los siguientes pasos:*

1. *Instala la pila LAMP en una MV o contenedor dentro de Proxmox. Instala también la aplicación phpmyadmin para poder gestionar el servidor de base de datos mediante la interfaz gráfica* ***(Incluye captura de pantalla con los componentes instalados).***
2. *Descargar el código fuente de la aplicación y los ficheros configuración mediante la aplicación git. Para ello deberás instalarla en tu MV de Proxmox (sudo apt install git). Una vez instalado, ubícate en la carpeta /var/www/html/ y clona el repositorio mediante el comando “sudo git clone* [*https://github.com/VidaInformatico/Sistema-de-biblioteca-basico-php-8-y-mysql.git*](https://github.com/VidaInformatico/Sistema-de-biblioteca-basico-php-8-y-mysql.git)*”.* ***(Incluye captura de pantalla)***
3. *Renombra la carpeta creada que contiene la aplicación por “biblioteca”. Accede a la nueva carpeta creada desde el entorno de desarrollo Visual Studio Code (o Codium) mediante el plugin de acceso remoto.*
4. *Crea una base de datos en el servidor de MySQL llamada* ***biblioteca*** *y un usuario (usuario=biblioteca, password=usuario) con acceso completo a ella. Lo puedes realizar desde el interfaz gráfico phpmyadmin o mediante sentencias SQL* ***(Incluye captura de pantalla del usuario creado)***
5. *Ejecuta el script biblioteca.sql para crear todas las tablas de la aplicación web.* ***(Incluye captura de pantalla donde se muestre las tablas generadas)***
6. *Crea un virtualhost con el que accederás con el nombre* ***biblioteca.tunombre.org****. Recuerda que el DocumentRoot de sitio virtual es la carpeta biblioteca creada anteriormente.* ***(Incluye captura de pantalla del fichero de configuración)***
7. *Vamos a configurar el acceso a la base de datos desde la aplicación, para ello cambia el fichero biblioteca/Config/Config.php indicando la URL con la que vas a acceder a la aplicación, el usuario de acceso (el que has creado en el punto 2), su contraseña, la base de datos que has creado y la dirección donde se encuentra la base de datos, que en este caso es localhost.* ***(Incluye captura de pantalla del fichero config.php)***

*Texto

El contenido generado por IA puede ser incorrecto.*

1. *El módulo rewrite de Apache2 nos va a permitir acceder a una URL e internamente estar accediendo a otra. En esta aplicación lo tenemos que activar para ello, ejecutamos* ***a2enmod rewrite*** *y reinicia el servidor. Si has activado este módulo previamente, no es necesario repetir el paso.*
2. *La configuración del módulo rewrite está realizada en el fichero de configuración apache2 .htaccess. Tenemos que permitir al servidor web leer este fichero, para ello modifica la directiva AllowOverride con el valor All en el fichero /etc/apache2/apache2.conf en la configuración del directorio /var/www, quedaría de la siguiente forma:*

*<Directory /var/www/>*

*Options Indexes FollowSymLinks*

***AllowOverride All***

*Require all granted*

*</Directory>*

***(Incluye captura de pantalla del fichero apache2.conf)***

1. *Verifica que la aplicación funciona. Recuerda modificar tu fichero hosts del PC para que redireccione correctamente al servidor apache al ingresar en el navegador la url personalizada de tu sitio. El usuario y password de entrada es admin/admin.* ***(Incluye captura de pantalla en donde se visualice que has entrado en la aplicación)***
2. *Por último, sube el proyecto completo a tu perfil de GitHub con el nombre Biblioteca-IAW-2ASIR-TuNombre. Recuerda que previamente tienes que crear un repositorio local dentro de la carpeta raíz del proyecto (var/www/html/biblioteca) y subir todos los archivos al mismo. Después tienes que desconectar tu repositorio local con el repositorio remoto origen (git remote remove origin) y volver a conectarlo con el proyecto de tu perfil de GitHub* ***(Incluye captura de pantalla de los comandos empleados y la URL de tu perfil de GitHub).***

[1. IP estática.](#A00_Indice)

Texto

El contenido generado por IA puede ser incorrecto.

[2. Instalación de LAMP](#A00_Indice)

[a. Apache.](#A00_Indice)

**apt update**

**apt install apache**

**systemctl status apache2**

Ponemos la IP del servidor en el navegador web de nuestro equipo.

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

[b. MySQL.](#A00_Indice)

**apt install mysql-server**

**systemctl status mysql**

**ss -tap | grep mysql**

Configuramos una contraseña para el usuario “**root**” de MySQL porque por defecto no tiene una.

**sudo mysql -u root -p**

**ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql\_native\_password BY '12345';**

Texto

El contenido generado por IA puede ser incorrecto.

Habilitamos el acceso remoto a nuestro servidor MySQL desde IPs externas, para ello debemos modificar el archivo que se encuentra en el directorio “**/etc/mysql/mysql.conf.d/mysqld.cnf**”, usando nano. Debemos poner lo siguiente:

**bind-address = 0.0.0.0**

Imagen de la pantalla de un celular con letras

El contenido generado por IA puede ser incorrecto.

Crear usuario: **CREATE USER 'usuario'@'%' IDENTIFIED BY 'contraseña';**

Otorgar privilegios a un usuario en una base de datos:

**GRANT ALL PRIVILEGES ON [nombre de la base de datos].\* TO 'usuario'@'%';**

**FLUSH PRIVILEGES;**

Mostrar permisos en una base de datos:

**SHOW GRANTS FOR 'biblioteca'@'localhost';**

Borrar usuario: **DROP USER 'biblioteca'@'localhost';**

Mostrar listado de usuarios: **SELECT User, Host FROM mysql.user;**

Mostrar listado de bases de datos: **SHOW DATABASES;**

Mostrar usuario actual**: SELECT USER();**

**SELECT CURRENT\_USER();**

Entrar en base de datos: **USE (nombre base de datos);**

Ver tablas de una base de datos: primero debemos entrar en la base y luego usar:

**SHOW TABLES;**

[c. PHP.](#A00_Indice)

**apt install php libapache2-mod-php -y**

Podemos comprobar que está funcionando, creando un archivo **.php** dentro del directorio **/var/www/html/** y abriendo el navegador.

**<?php**

**echo "Hello World!";**

**?>**



Otra forma de ver que lo hemos instalado bien es hacer:

**ls -l /etc/apache2/mods-enabled/**

Texto

El contenido generado por IA puede ser incorrecto.

[d. phpMyAdmin](#A00_Indice)

**apt install phpmyadmin**

Marca la opción de “**apache2**”.

Interfaz de usuario gráfica, Texto, Correo electrónico

El contenido generado por IA puede ser incorrecto.

Le decimos que **NO** queremos crear **una base de datos propia** de **phpMyAdmin**, porque ya tenemos una.

Texto

El contenido generado por IA puede ser incorrecto.

Una vez instalado, ponemos en el navegador <http://IP_SERVIDOR/phpmyadmin/>. E introducimos las credenciales de un usuario de MySQL. En nuestro caso usamos el “root”.

Interfaz de usuario gráfica

El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

El contenido generado por IA puede ser incorrecto.

[3. Clonación de GitHub.](#A00_Indice)

Instalamos “**git**”:

**apt install git**

Nos posicionamos:

**cd /var/www/html/**

Nos bajamos el código fuente:

**sudo git clone** [**https://github.com/VidaInformatico/Sistema-de-biblioteca-basico-php-8-y-mysql.git**](https://github.com/VidaInformatico/Sistema-de-biblioteca-basico-php-8-y-mysql.git)

Texto

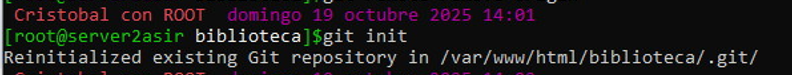
El contenido generado por IA puede ser incorrecto.

[a. Renombrar carpeta:](#A00_Indice)

**mv Sistema-de-biblioteca-basico-php-8-y-mysql/ biblioteca**



**git init**

****

[4. Instalar VSCodium o Code.](#A00_Indice)

<https://github.com/VSCodium/vscodium/releases>

Descarga la versión “VSCodiumUserSetup.exe”.Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

El contenido generado por IA puede ser incorrecto.

[a. Instalamos extensión Idioma Español.](#A00_Indice)

Captura de pantalla de un celular

El contenido generado por IA puede ser incorrecto.

[b. Instalamos extensión SSH.](#A00_Indice)

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

[c. Acceder a la carpeta creada desde el terminal de VSCodium.](#A00_Indice)

Nos vamos a **Terminal** (o tres puntos) **🡪 Nuevo terminal**.

Captura de pantalla de un celular

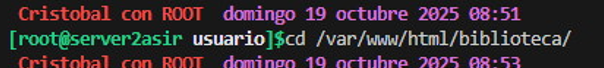
El contenido generado por IA puede ser incorrecto.

Nos conectamos con SSH al servidor.

Texto

El contenido generado por IA puede ser incorrecto.

Navegamos hasta el directorio.



[5. MySQL: crear base de datos y usuario.](#A00_Indice)

[a. phpMyAdmin.](#A00_Indice)

Le damos a “**Bases de datos**”, elegimos un nombre y luego “**utf8mb4\_general\_ci**”. Y le damos a **Crear**.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

El contenido generado por IA puede ser incorrecto.

Para crear usuarios pinchamos en la base de datos y luego le damos a “**Privilegios**”. Ahí debemos darle a “**Nuevo**”.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

El contenido generado por IA puede ser incorrecto.

Nombre, contraseña, nivel de permisos, etc.

Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

[b. Terminal MySQL.](#A00_Indice)

**CREATE DATABASE “nombre base de datos”;**

**SHOW DATABASES;**

**Texto

El contenido generado por IA puede ser incorrecto.**

**CREATE USER 'biblioteca'@'%' IDENTIFIED BY 'contraseña';**

**GRANT ALL PRIVILEGES ON biblioteca.\* TO 'biblioteca'@'%';**

**FLUSH PRIVILEGES;**

**SHOW GRANTS FOR 'biblioteca'@'%';**

**Texto

El contenido generado por IA puede ser incorrecto.**

[6. Ejecutar script biblioteca.sql](#A00_Indice)

Ejecuta el script **biblioteca.sql** para crear todas las tablas de la aplicación web.

Le damos a **Examinar…**  y navegamos hasta donde tengamos el archivo. Luego, en la parte de abajo, le damos a **Continuar**.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.

En la parte izquierda nos aparecerán todos los elementos de la base de datos.

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

[7. Configuración de VirtualHost.](#A00_Indice)

*Crea un virtualhost con el que accederás con el nombre* ***biblioteca.tunombre.org****. Recuerda que el DocumentRoot de sitio virtual es la carpeta biblioteca creada anteriormente.*

Deshabilitamos el sitio que Apache tiene habilitado por defecto:

**a2dissite 000-default.conf**

Copiamos el default para usarlo como plantilla.

**cp sites-available/000-default.conf sites-available/biblioteca.conf**

Usamos nano para modificarlo:

**nano sites-available/biblioteca.conf**

Y cuando acabemos de modificarlo lo activamos:

**a2ensite biblioteca.conf**

**systemctl reload apache2**

**Texto

El contenido generado por IA puede ser incorrecto.**

**Texto

El contenido generado por IA puede ser incorrecto.**

[8. Configurar el acceso a la base de datos desde la aplicación.](#A00_Indice)

*“Vamos a configurar el acceso a la base de datos desde la aplicación, para ello cambia el fichero biblioteca/Config/Config.php indicando la URL con la que vas a acceder a la aplicación, el usuario de acceso (el que has creado en el punto 2), su contraseña, la base de datos que has creado y la dirección donde se encuentra la base de datos, que en este caso es localhost”.*

**nano /var/www/html/biblioteca/Config/Config.php**

**Texto

El contenido generado por IA puede ser incorrecto.**

[9. Activar modulo rewrite.](#A00_Indice)

*“El módulo rewrite de Apache2 nos va a permitir acceder a una URL e internamente estar accediendo a otra. En esta aplicación lo tenemos que activar para ello, ejecutamos* ***a2enmod rewrite*** *y reinicia el servidor. Si has activado este módulo previamente, no es necesario repetir el paso”.*

**a2enmod rewrite**

**systemctl restart apache2**

**Interfaz de usuario gráfica, Texto

El contenido generado por IA puede ser incorrecto.**

[10. Configuración del módulo rewrite en apache2.conf.](#A00_Indice)

*“La configuración del módulo rewrite está realizada en el fichero de configuración apache2 .htaccess. Tenemos que permitir al servidor web leer este fichero, para ello modifica la directiva AllowOverride con el valor All en el fichero /etc/apache2/apache2.conf en la configuración del directorio /var/www, quedaría de la siguiente forma:”*

**nano /etc/apache2/apache2.conf**

**systemctl restart apache2**

**Texto

El contenido generado por IA puede ser incorrecto.**

[NOTA: pasar todos los archivos al usuario de apache2.](#A00_Indice)

Para evitar problemas con los permisos y que Apache no pueda acceder a los archivos, vamos a pasar la propiedad al usuario de Apache.

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

El contenido generado por IA puede ser incorrecto.

**chown -R www-data:www-data /var/www/html/biblioteca/\***

**Texto, Aplicación

El contenido generado por IA puede ser incorrecto.**

[NOTA: Poner la dirección y el nombre de la página en “hosts”.](#A00_Indice)

En el archivo “**hosts**” de tu equipo (no en el servidor). Debes introducir la IP del servidor y el nombre de las páginas web, los que has puesto en los archivos “.conf”.

Linux: **/etc/hosts**

Windows: **C:\windows\system32\drivers\etc\hosts**



[11. Verificar el buen funcionamiento de la aplicación.](#A00_Indice)

Interfaz de usuario gráfica, Sitio web

El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

[12. Subir el proyecto completo a tu perfil de GitHub.](#A00_Indice)

*“Por último, sube el proyecto completo a tu perfil de GitHub con el nombre Biblioteca-IAW-2ASIR-TuNombre. Recuerda que previamente tienes que crear un repositorio local dentro de la carpeta raíz del proyecto (var/www/html/biblioteca) y subir todos los archivos al mismo. Después tienes que desconectar tu repositorio local con el repositorio remoto origen (git remote remove origin) y volver a conectarlo con el proyecto de tu perfil de GitHub”.*

En el directorio **/var/www/html/biblioteca/**

Previamente teníamos nuestro repositorio local funcionando.

Texto

El contenido generado por IA puede ser incorrecto.

Ver que archivos han sido modificados o creados: **git status -s**

Añadir todos los archivos modificados: **git add .**

Confirma las modificaciones de los archivos, tras lo cual se pueden “subir”:

**git commit -m “Comentario”**

**Texto, Chat o mensaje de texto

El contenido generado por IA puede ser incorrecto.**

Muestra historial de “commits”: **git log --oneline**

**Texto

El contenido generado por IA puede ser incorrecto.**

Creamos un nuevo repositorio en nuestro perfil de GitHub.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

El contenido generado por IA puede ser incorrecto.

Antes de añadirlo debemos crear un **Token**:

Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.

Podemos elegir entre el modo clásico o el “**fine-grained**”.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

El contenido generado por IA puede ser incorrecto.

Nos mandarán un correo a nuestro email

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

El contenido generado por IA puede ser incorrecto.

Elegir nombre.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

El contenido generado por IA puede ser incorrecto.

Fecha caducidad

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

El contenido generado por IA puede ser incorrecto.

A que repositorios afecta.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

El contenido generado por IA puede ser incorrecto.

Permisos.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

El contenido generado por IA puede ser incorrecto.

Generar Token.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

El contenido generado por IA puede ser incorrecto.

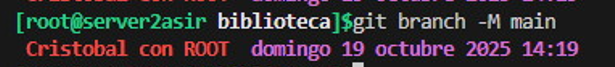
Debemos copiar el token en lugar seguro, porque una vez que cerremos la pestaña no se volverá a mostrar.

Ahora, para subir nuestro repositorio local a GitHub, usaremos tres códigos que nos muestran cuando hacemos un nuevo repositorio en GitHub.

**git remote add origin URL\_REPOSITORIO\_GITHUB**



**git branch -M main**



**git push -u origin main**

Aquí tendremos que poner el nombre de nuestro usuario de GitHub y como “**password**” el **Token** que hemos configurado antes.

**Texto

El contenido generado por IA puede ser incorrecto.**

Probando que estamos conectado a un repositorio remoto.

**git remote -v**

Texto

El contenido generado por IA puede ser incorrecto.

Si quisiéramos dejar de estar conectado al repositorio debemos usar:

**git remote remove origin**

*“Use the git remote remove command followed by the name of the remote repository you want to remove. Conventionally, the default remote name is origin. Replace origin with the name of the remote you want to remove if it differs from origin[[1]](#footnote-1).”*

Cada vez que queramos subir algo al repositorio de GitHub tendremos que hacer uso del **token**. Primer hacer “**commit**” y luego “**push**”.

**git push origin main**

Texto

El contenido generado por IA puede ser incorrecto.

<https://github.com/cristobal90palma/Biblioteca-IAW-2ASIR-Cristobal>

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

El contenido generado por IA puede ser incorrecto.

1. <https://betterstack.com/community/questions/how-to-remove-remote-origin/> [↑](#footnote-ref-1)