

13-10-2023

# Plataformas de programación web en entorno servidor. Aplicaciones LAMP.

## Tarea 01

ROBERTO RODRÍGUEZ JIMÉNEZ  
[roberto.rodjim.1@educa.jcyl.es](mailto:roberto.rodjim.1@educa.jcyl.es)

## Contenido

Tarea online DWES01 .....	2
Enunciado .....	2
Caso práctico.....	2
¿Qué te pedimos que hagas? .....	2
Recursos necesarios y recomendaciones .....	3
Recursos necesarios .....	3
Recomendaciones .....	3
Evaluación de la tarea .....	3
Criterios de evaluación implicados.....	3
¿Cómo valoramos y puntuamos tu tarea? .....	3
RESOLUCIÓN.....	4
1. ¿Qué tipo de páginas, estáticas o dinámicas, utilizarás para programar cada una de las páginas que componen tu aplicación? ¿Por qué? .....	4
2. Si en la página de introducción de datos quieres comprobar, antes de enviar los datos, que el correo electrónico introducido cumple unas ciertas normas (por ejemplo, que tiene una @), ¿qué tecnología/lenguaje utilizarás? .....	5
3. Si en esa misma página, ahora quieres comprobar que el correo electrónico introducido no se haya introducido anteriormente y figure en la lista, ¿qué tecnología/lenguaje utilizarás? .....	5
4. ¿Qué arquitecturas puedes usar en el servidor para ejecutar la aplicación? ¿Cómo es el o los lenguajes que se usa en cada una de esas arquitecturas?: ¿de guiones, compilado a código nativo o compilado a código intermedio? .....	5
5. ¿Qué parámetros debes tener en cuenta para decidirte por usar una arquitectura u otra? .....	6
6. Si te decides por utilizar una arquitectura AMP para la aplicación ¿qué componentes necesitas instalar en tu servidor para ejecutar la aplicación? Indica algún producto concreto para cada componente.....	6
7. ¿Qué necesitas instalar en tu ordenador para poder desarrollar la aplicación? .....	6
8. Si utilizas el lenguaje PHP para programar la aplicación, ¿cuál será el tipo de datos se utilizará para manipular cada una de las direcciones de correo? .....	7

## Tarea online DWES01

**Título de la tarea:** Gestionar direcciones de correo

**Unidad 1:** Plataformas de programación web en entorno servidor. Aplicaciones LAMP.

**Ciclo formativo y módulo:** Ciclo superior de DAW. Módulo de DWES.

**Curso académico:** 2021/2022

### Enunciado

#### Caso práctico

Juan debe explicar a Carlos las necesidades que tendrá un nuevo proyecto para controlar direcciones de correo, para ello le pide que piense detenidamente que tecnología serían las óptimas y cuáles no. Carlos con los conceptos que le ha explicado se pone manos a la obra.

#### ¿Qué te pedimos que hagas?

Quieres programar una aplicación web para gestionar una lista de direcciones de correo. En una primera aproximación, tu aplicación se compondrá de tres páginas:

- ✓ Una página de presentación, donde explicas el cometido de la aplicación y su funcionamiento. Contiene un enlace que te lleva a la siguiente página.
- ✓ Una página de introducción de datos, donde cualquiera puede darse de alta introduciendo su nombre y su dirección de correo. Contiene un formulario que, una vez rellenado, envía los datos y te lleva a la siguiente página.
- ✓ Una página de visualización de datos, en la que se muestra la lista, nombres y direcciones de correo, de todos los que se han anotado en la aplicación.

Tienes que crear un documento en el que respondas a las siguientes preguntas sobre la aplicación que vas a desarrollar:

1. ¿Qué tipo de páginas, estáticas o dinámicas, utilizarás para programar cada una de las páginas que componen tu aplicación? ¿Por qué?
2. Si en la página de introducción de datos quieres comprobar, antes de enviar los datos, que el correo electrónico introducido cumple unas ciertas normas (por ejemplo, que tiene una @), ¿qué tecnología/lenguaje utilizarás?
3. Si en esa misma página, ahora quieres comprobar que el correo electrónico introducido no se haya introducido anteriormente y figure en la lista, ¿qué tecnología/lenguaje utilizarás?
4. ¿Qué arquitecturas puedes usar en el servidor para ejecutar la aplicación? ¿Cómo es el o los lenguajes que se usa en cada una de esas arquitecturas?: ¿de guiones, compilado a código nativo o compilado a código intermedio?
5. ¿Qué parámetros debes tener en cuenta para decidirte por usar una arquitectura u otra?
6. Si te decides por utilizar una arquitectura AMP para la aplicación ¿qué componentes necesitas instalar en tu servidor para ejecutar la aplicación? Indica algún producto concreto para cada componente.
7. ¿Qué necesitas instalar en tu ordenador para poder desarrollar la aplicación?
8. Si utilizas el lenguaje PHP para programar la aplicación, ¿cuál será el tipo de datos se utilizará para manipular cada una de las direcciones de correo?

## Recursos necesarios y recomendaciones

### Recursos necesarios

No se necesitan recursos específicos para esta tarea.

### Recomendaciones

Se recomienda leer detenidamente cada una de las cuestiones planteadas.

## Evaluación de la tarea

### Criterios de evaluación implicados

- ✓ Se han caracterizado y diferenciado los modelos de ejecución de código en el servidor y en el cliente Web.
- ✓ Se han reconocido las ventajas que proporciona la generación dinámica de páginas Web y sus diferencias con la inclusión de sentencias de guiones en el interior de las páginas Web.
- ✓ Se han identificado los mecanismos de ejecución de código en los servidores Web.
- ✓ Se han reconocido las funcionalidades que aportan los servidores de aplicaciones y su integración con los servidores Web.
- ✓ Se han identificado y caracterizado los principales lenguajes y tecnologías relacionados con la programación Web en entorno servidor.
- ✓ Se han verificado los mecanismos de integración de los lenguajes de marcas con los lenguajes de programación en entorno servidor.
- ✓ Se han reconocido y evaluado las herramientas de programación en entorno servidor.
- ✓ Se han caracterizado y diferenciado los modelos de ejecución de código en el servidor y en el cliente Web.
- ✓ Se han reconocido las ventajas que proporciona la generación dinámica de páginas Web y sus diferencias con la inclusión de sentencias de guiones en el interior de las páginas Web.
- ✓ Se han identificado los mecanismos de ejecución de código en los servidores Web.
- ✓ Se han reconocido las funcionalidades que aportan los servidores de aplicaciones y su integración con los servidores Web.
- ✓ Se han identificado y caracterizado los principales lenguajes y tecnologías relacionados con la programación Web en entorno servidor.
- ✓ Se han verificado los mecanismos de integración de los lenguajes de marcas con los lenguajes de programación en entorno servidor.
- ✓ Se han reconocido y evaluado las herramientas de programación en entorno servidor.

### ¿Cómo valoramos y puntuamos tu tarea?

#### **Rúbrica de la tarea (Total 10 puntos.)**

Apartados 1, 2, 3, 5, 7, 8. Responder correcta y justificadamente a las cuestiones planteadas. 1 punto por pregunta.

Apartados 4 y 6. Responder correcta y justificadamente a las cuestiones planteadas. 2 puntos por pregunta.

## RESOLUCIÓN

1. ¿Qué tipo de páginas, estáticas o dinámicas, utilizarás para programar cada una de las páginas que componen tu aplicación? ¿Por qué?

La página de presentación y la página de introducción de datos serán, en principio, páginas estáticas.

La página de presentación debe mostrar información a cerca de la aplicación y un manual sobre su funcionamiento. No debe cambiar si no cambia la manera en la que la aplicación trabaja, o si se ha modificado algún dato que mostrar en la información.

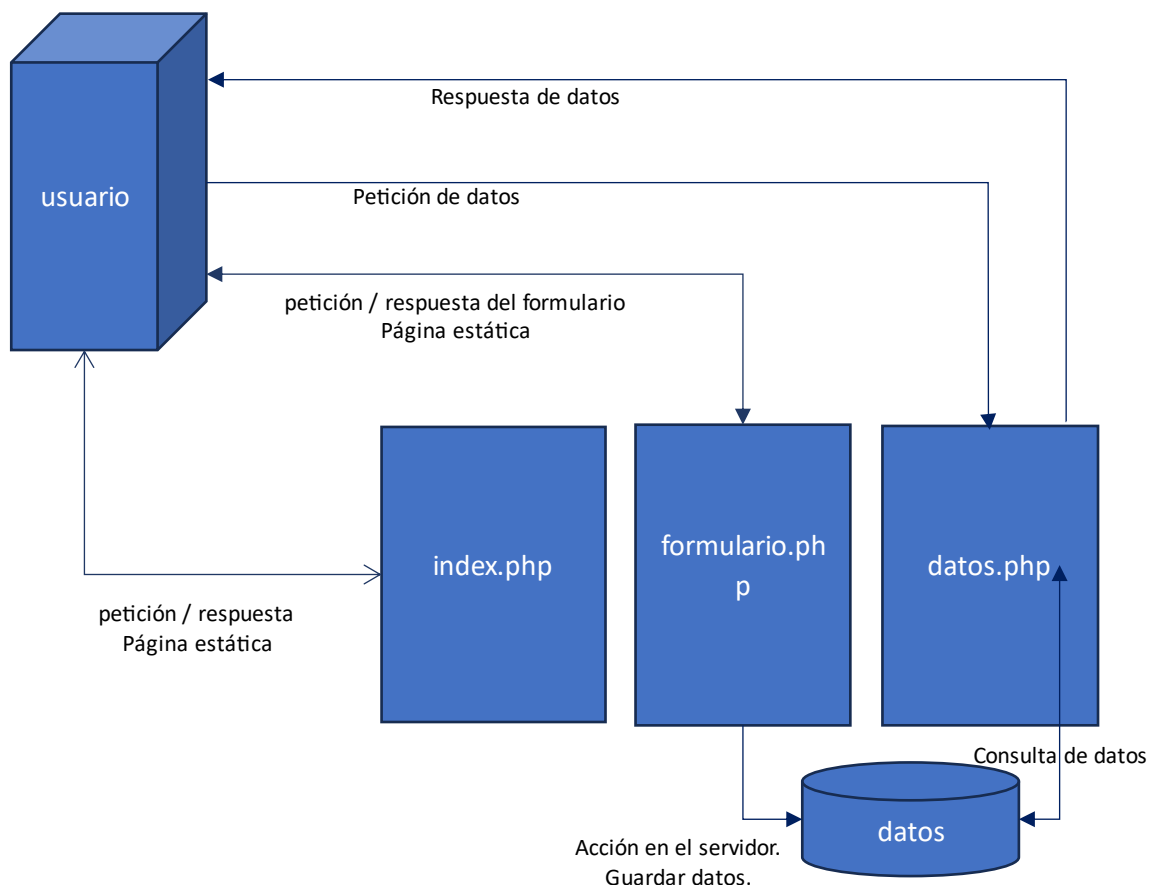
La página de introducción de datos está compuesta por un formulario en cual introducir los datos.

El formulario será el mismo siempre, cargado en su página.

Será una página estática. El destino, una vez enviado el formulario, es una página diferente, en la cual se muestra el listado de los registros.

Podría darse el caso de que se pidiese que en la misma página del formulario se mostrasen los registros, lo que requeriría que la página fuese dinámica, pero en los requisitos se especifica que los registros deben mostrarse en una página única.

La página de visualización de datos debe ser dinámica, ya que su contenido cambiará cada vez que se agregue un nuevo usuario.



2. Si en la página de introducción de datos quieres comprobar, antes de enviar los datos, que el correo electrónico introducido cumple unas ciertas normas (por ejemplo, que tiene una @), ¿qué tecnología/lenguaje utilizarás?

Actualmente (octubre de 2023) y con el estándar HTML5, la validación del email la realiza el navegador al marcar el input como `type="email"`.

Sin embargo, si no se quiere depender de que el usuario tenga un navegador actualizado, o este no soporte HTML5, el lenguaje más apropiado para validar el email es JavaScript.

JavaScript se ejecuta en el navegador y no requiere de conexión con el servidor para poder validar el campo. Habría que codificar una función que recoja el valor del input y devuelva un valor booleano dependiendo de si el campo es válido o no.

El email se evaluaría cuando el input perdiese el foco y la función lo puede obtener a través del DOM. Si no se valida el valor, el input del email recuperaría el foco y se señalaría, con un texto o resaltando con un color llamativo, que el email no es válido.

Para validar el email puede usarse patrón.

3. Si en esa misma página, ahora quieres comprobar que el correo electrónico introducido no se haya introducido anteriormente y figure en la lista, ¿qué tecnología/lenguaje utilizarás?

Para comprobar si un correo existe, en el momento en el que se ha introducido, podemos usar JavaScript para realizar una petición al servidor, que este haga la comprobación usando PHP, y que devuelva la información mediante XML o JSON.

Los lenguajes usados son JavaScript, del lado del cliente, y PHP del lado del servidor. Esta tecnología es conocida como AJAX y permite obtener datos del servidor sin tener que abandonar la página.

4. ¿Qué arquitecturas puedes usar en el servidor para ejecutar la aplicación? ¿Cómo es el o los lenguajes que se usa en cada una de esas arquitecturas?: ¿de guiones, compilado a código nativo o compilado a código intermedio?

La arquitectura para el servidor debería ser AMP, para cualquier sistema operativo.

Necesitamos que el servidor ejecute consultas a una base de datos y el lenguaje PHP para realizar dichas consultas.

La respuesta del servidor es HTML, por lo que nos vale un servidor HTTP como Apache HTTP Server.

## Lenguajes

PHP es un lenguaje de guiones, ejecutado directamente en el servidor por el intérprete de PHP.

Para el acceso a los datos y su manipulación, se usa el lenguaje SQL, que es un lenguaje declarativo.

La respuesta es HTML, un lenguaje de marcas que es interpretado por el navegador.

## 5. ¿Qué parámetros debes tener en cuenta para decidirte por usar una arquitectura u otra?

Para elegir una arquitectura adecuada se debe tener en cuenta el tipo de aplicación que vamos a desarrollar:

- El tamaño de la aplicación: Es una aplicación pequeña, no necesita de excesivos recursos.
- Los lenguajes que se van a utilizar también es un punto a tener en cuenta. Dado que vamos a usar PHP y MySQL, podríamos elegir una arquitectura AMP.
- Vamos a usar herramientas de uso libre, tanto para las bases de datos, como para el IDE o cualquier editor de código o imágenes.
- El proyecto se inicia desde cero y podemos usar la base de datos. Ya que queremos software libre y pensamos en una arquitectura WAMP, el motor de la base de datos será MariaDB.

## 6. Si te decides por utilizar una arquitectura AMP para la aplicación ¿qué componentes necesitas instalar en tu servidor para ejecutar la aplicación? Indica algún producto concreto para cada componente.

Definitivamente, se opta por una arquitectura AMP. Necesitaremos instalar en el sistema operativo un servidor HTTP, interpretar un lenguaje para programar la aplicación y un sistema de gestión de bases de datos.

- Sistema operativo:  
Como tratamos de evitar costes o, al menos, minimizarlos, optamos por una distribución Ubuntu (Linux).
- Servidor:  
Instalamos Apache HTTP Server, con licencia *Apache*, la cual permite la libertad de uso para cualquier propósito.
- Lenguaje de programación:  
El lenguaje para atender las peticiones en el servidor será PHP. PHP es interpretado (script) en el servidor por el motor Zend.  
PHP se implementará como un módulo y no como ejecutable CGI.  
El módulo que debemos instalar en el servidor es **mod\_php**.
- SGBD:  
Para la persistencia de los datos se usará MariaDB (licencia GPL).  
La elección de MariaDB en lugar de MySQL se debe a garantizar el mantenimiento de la licencia GPL, ya que MySQL es propiedad de Sun Microsystems.

## 7. ¿Qué necesitas instalar en tu ordenador para poder desarrollar la aplicación?

Para desarrollar la aplicación vamos a usar Visual Studio Code.

VSC es un *editor de código fuente* más que un IDE, pero con los complementos que pueden instalarse cumple perfectamente el cometido.

Tiene licencia *MIT*, por lo que podemos usarlo con muy pocas restricciones.

VSC tiene un módulo de control de versiones *GIT*, con acceso a repositorios como GitHub.

Tiene opciones de resaltado, autocompletado o formateo de código. También puede mostrar la documentación oficial del lenguaje en línea.

VSC puede depurar el código PHP.

8. Si utilizas el lenguaje PHP para programar la aplicación, ¿cuál será el tipo de datos se utilizará para manipular cada una de las direcciones de correo?

PHP permite guardar datos como cadenas de texto.

Para almacenar las direcciones de correo se usará el tipo **String**.

String permite la manipulación del valor guardado, pudiendo comprobar que el formato es válido en cualquier momento o, por ejemplo, separar el nombre del usuario del dominio.