

# **Tareas Unidad 1**

## **Desarrollo Web en Entorno Servidor**

Autor: Derimán Tejera Fumero

Fecha: 25/09/2024

# Respuesta a las preguntas

## ¿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?

**Página de presentación:** Será una página de tipo estática ya que solo muestra información que no cambia (estática), como textos, una descripción del funcionamiento de la aplicación etc... .

**Página de introducción de datos:** Será una página dinámica ya que se necesitará procesar los datos del formulario que el usuario ha introducido y deberá validarlos.

**Página de visualización de datos:** Será una página dinámica, ya que mostrará información que cambia según las personas que se registren en la lista de correos.

## 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?

Utilizaría JavaScript, utilizando por ejemplo un regex: `^[^\\s@]+@[\\s@]+\\.[^\\s@]+$` en una función.

(Información sacada de la [web](#))

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

Utilizaría PHP desde el lado del servidor para consultar si la información ya está dentro de la base de datos.

**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?**

**Java EE** se utiliza para aplicaciones medianas o grandes, usando Java compilado a código intermedio (bytecode) con tecnologías como servlets y JSP.

**AMP (Apache, MySQL, PHP/Perl/Python)** es utilizada para aplicaciones pequeñas o medianas, basada en lenguajes de guiones como PHP, con versiones LAMP para Linux, WAMP para Windows....

**CGI/Perl** es una de las más antiguas, permite ejecutar scripts en Perl o C, pero es menos eficiente que otras opciones.

**ASP.Net**, utiliza lenguajes como C# y Visual Basic.Net compilados a código intermedio (MSIL) en servidores IIS, siendo una plataforma completa pero que utiliza código propietario.

**Node.js** es un entorno en tiempo real para aplicaciones escalables, basado en JavaScript asíncrono (arquitectura basada en eventos y asíncrona), ideal para manejar muchas conexiones simultáneas.

**(Información sacada del apartado 2.1 de UT1 DWES)**

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

- ¿Qué tamaño tiene el proyecto?
- ¿Qué lenguajes de programación conozco? ¿Vale la pena el esfuerzo de aprender uno nuevo?
- ¿Voy a usar herramientas de código abierto o herramientas propietarias? ¿Cuál es el coste de utilizar soluciones comerciales?
- ¿Voy a programar la aplicación yo solo o formaré parte de un grupo de programadores?
- ¿Cuento con algún servidor web o gestor de base de datos disponible o puedo decidir libremente utilizar el que crea necesario?
- ¿Qué tipo de licencia voy a aplicar a la aplicación que desarrolle?

**Tamaño del proyecto:** Si es pequeño o mediano, AMP sería suficiente, si es grande se podría considerar Java EE o ASP.NET.

**Coste:** AMP es de código abierto, sin embargo ASP.NET es una solución comercial.

**Experiencia del equipo:** Si el equipo tiene experiencia en Java o .NET podría ser más fácil usar.

**Escalabilidad:** Los proyectos que requieren escalabilidad pueden estar mejor contruidos en Java EE o Node.js.

**Requisitos del servidor:** Si el servidor ya tiene ciertos componentes instalados como puedan ser Apache...

**(Información sacada del apartado 2.1.1 de UT1 DWES)**

**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.**

**Sistema operativo:** Linux (por ejemplo: Ubuntu).

**Servidor web:** Apache.

**Base de datos:** MySQL o MaríaDB.

**Lenguaje de programación:** PHP.

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

**Entorno de desarrollo (IDE):** Visual Studio Code o NetBeans.

**Un servidor local:** XAMPP o LAMP.

**Componentes de la arquitectura AMP (PHP, Apache y MySQL):** como parte del entorno AMP para probar la aplicación. Gestión de bases de datos MySQL, lenguaje de programación PHP, Apache como servidor.

**(Información sacada del apartado 3.2 de UT1 DWES)**

**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?**

El tipo de datos deberá ser string, porque las direcciones de correo son cadenas de texto, además podría ser conveniente convertirlas a mayúscula o a minúsculas para verificar si el email ya está en la base de datos o no.

**(Información sacada del apartado 3.4.1 de UT1 DWES)**

**FIN**