

Tarea 1

RA1

16 OCTUBRE

IES TRAFALGAR

Creado por: Antonio Miguel Alba Garcia



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:** Usaría una página estática, ya que el contenido no cambiará frecuentemente. Es una introducción básica a la aplicación y no requiere procesamiento en el servidor.
- **Página de introducción de datos:** Esta página sería dinámica, ya que incluye un formulario y necesita procesar y validar la información que los usuarios ingresan (nombre y correo).
- **Página de visualización de datos:** Sería dinámica porque muestra una lista que debe actualizarse cada vez que alguien se registra, lo que requiere consulta a una base de datos.

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** (del lado del cliente) para realizar una validación en tiempo real del formato del correo electrónico. Esto evita recargar la página o enviar datos incorrectos al servidor.

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

En este caso, necesitaríamos validar en el servidor usando **PHP** o **Node.js** para verificar en la base de datos si el correo ya existe. Esto asegura que no se ingresen correos duplicados.

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?

Podemos usar varias arquitecturas, como:

- **LAMP** (Linux, Apache, MySQL, PHP): Utiliza PHP, un lenguaje interpretado y de guiones.
- **MEAN** (MongoDB, Express, Angular, Node.js): Utiliza JavaScript (Node.js) del lado del servidor, un lenguaje interpretado.
- **ASP.NET**: Utiliza C#, que se compila a código intermedio (IL, Intermediate Language) y luego es interpretado por el CLR.

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

Facilidad de desarrollo y mantenimiento, escalabilidad, compatibilidad con el sistema operativo, rendimiento y costes (algunos stacks como MEAN o LAMP son open-source y gratuitos, mientras que ASP.NET puede tener licencias).

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

Si usamos LAMP:

- **Linux**: El sistema operativo.
- **Apache**: El servidor web.
- **MySQL**: El sistema de gestión de bases de datos.
- **PHP**: Para la programación del lado del servidor.

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

Un entorno de desarrollo local con:

- **XAMPP** (incluye Apache, MySQL y PHP) o similar.
- Un **editor de código** como Visual Studio Code.
- **Git** para control de versiones.

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?

En PHP, las direcciones de correo se manipulan como cadenas de texto (tipo string), que permiten almacenar y procesar las direcciones de correo introducidas por los usuarios.