

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

Utilizaría páginas web dinámicas ya que son las empleadas para la programación de aplicaciones web, para crear aplicaciones que se ejecuten en un servidor web. La razón es que en las páginas web dinámicas la información que se muestra no es definitiva (a diferencia de las páginas web estáticas) sino que cambia dependiendo de las acciones que se efectúen desde el cliente web.

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?

El lenguaje a utilizar es JavaScript, que se ejecutará en el navegador y comprobará los datos que son introducidos en el formulario.

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 AJAX, que posibilita realizar programas en los que el código JavaScript que se ejecuta en el navegador puede comunicarse con un servidor para obtener información (como en este caso, si el correo introducido ya figura en la lista).

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 empleada en el servidor puede ser JavaEE, AMP, CGI, ASP.Net y Node.js.

En JavaEE se usa el lenguaje Java, en AMP el lenguaje PHP/Phyton/Perl, en CGI el lenguaje C/Perl/Phyton, en ASP el lenguaje Visual Basic.Net y en Node.js el lenguaje ECMAScript.

Los lenguajes de guiones son aquellos que se ejecutan directamente a partir de su código fuente (PHP, Phyton, Perl y ASP), los compilados nativos son aquellos en los que el código fuente es traducido a código binario antes de ejecutarse (C) y los compilados a código intermedio son aquellos que antes de ejecutarse su código fuente es traducido a código intermedio (Java).

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

Es necesario tener en cuenta el tamaño del proyecto, los lenguajes que ya se conocen y si merece la pena aprender uno nuevo, si se hará uso de herramientas de código abierto o propietarias (el coste que implicaría su uso), el número de personas del que se dispondrá para programar, si se cuenta ya con un servidor web o gestor de bases de datos previo o se puede escoger uno nuevo libremente en caso de necesitarse y el tipo de licencia a aplicar a la aplicación que se desarrolle.

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.

Será necesario instalar el módulo mod_php en el servidor Apache para que ejecute páginas web dinámicas (para ejecutar lenguaje de guiones en PHP).

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

Una plataforma de programación AMP que dependiendo del sistema operativo que se utilice para el servidor recibirá el nombre de LAMP (para Linux), WAMP (para Windows) o MAMP (para Mac).

En mi caso instalé XAMPP aunque existe también WAMPP.

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

Serían datos de tipo cadena (string) delimitados por comillas simples o dobles, que contendrían un conjunto de caracteres (letras, números y símbolos).

