

Formulario en HTML Parte I

Descubre los conceptos y características clave de los atributos fundamentales de los formularios HTML, como **name**, **action**, **method**, y **target**. Aprende a configurar y usar formularios de manera efectiva en tus proyectos web.



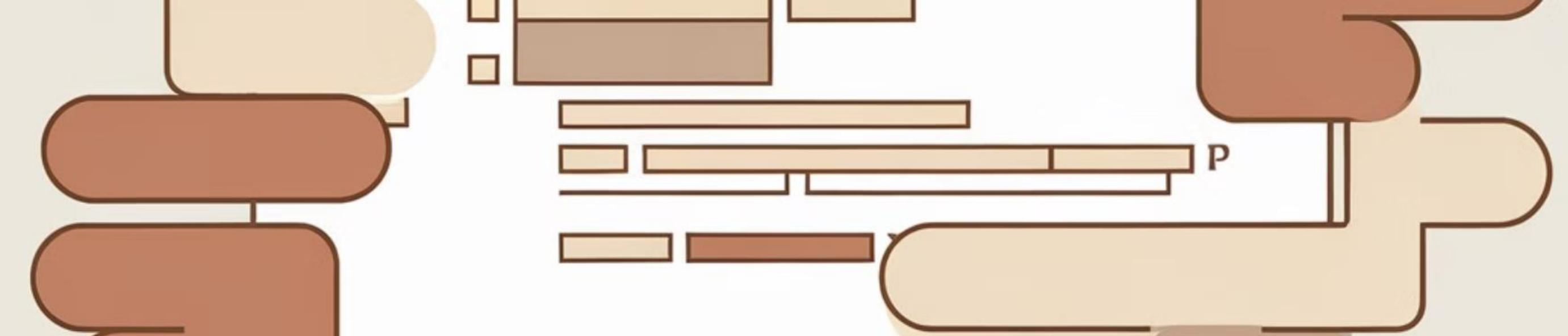
¿Qué es un formulario en HTML?

Definición

Un formulario en HTML es un elemento interactivo que permite a los usuarios enviar información a un servidor web. Está compuesto por diversos controles como campos de texto, botones, menús desplegables, entre otros.

Propósito

Los formularios HTML se utilizan comúnmente para recopilar datos de los usuarios, como iniciar sesión, registrarse, hacer pedidos o enviar comentarios. Facilitan la interacción entre el usuario y el sitio web.



Atributos de un formulario

Atributos Esenciales

Los formularios en HTML tienen atributos clave como name, action, method y target que definen su comportamiento y funcionamiento.

Configuración del Envío

Estos atributos establecen cómo y dónde se enviarán los datos introducidos por el usuario.

Facilidad de Uso

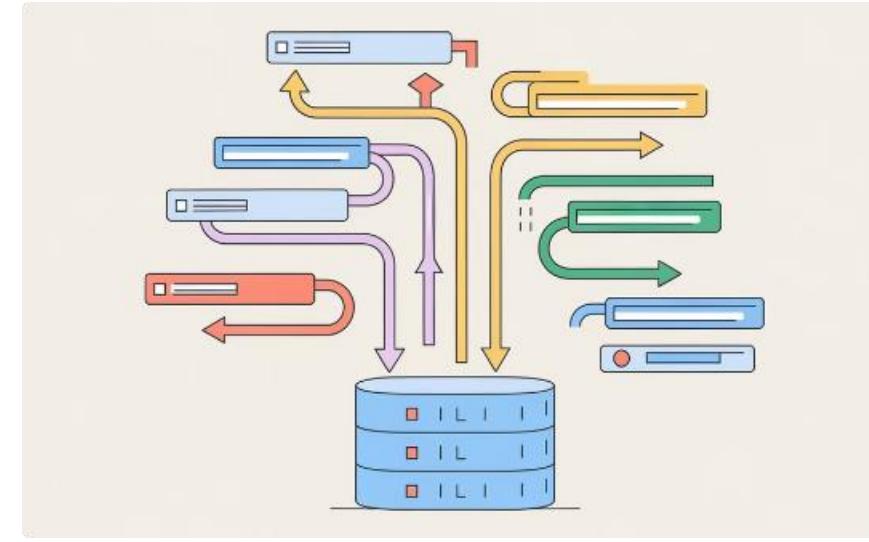
Una correcta configuración de estos atributos asegura una experiencia de usuario fluida y eficiente.

Atributo name



Definición del Atributo name

El atributo **name** en un formulario HTML es utilizado para dar un nombre único a cada campo del formulario. Este nombre se utiliza para identificar el campo en el momento de enviar los datos del formulario al servidor.



Importancia del Atributo name

El atributo **name** es crucial para el envío y procesamiento de datos de un formulario, ya que permite al servidor identificar y procesar correctamente cada campo del formulario.



Ejemplos de Atributo name

Algunos ejemplos de valores para el atributo **name** son: **nombre**, **email**, **contraseña**, **comentarios**, etc. Cada uno de estos nombres hace referencia a un campo específico del formulario.

Ejemplos de atributo name

El atributo **name** en un formulario HTML es esencial para identificar los diferentes campos que lo componen. Aquí hay algunos ejemplos de cómo se utiliza este atributo:

1. Nombre: **name="nombre"**
2. Correo electrónico: **name="email"**
3. Contraseña: **name="password"**
4. Número de teléfono: **name="telefono"**
5. Selección de género: **name="genero"**



Atributo action



1 Propósito

El atributo **action** en un formulario HTML especifica la URL del script o programa que procesará los datos enviados desde el formulario.

2 URL de Destino

La URL proporcionada en el atributo **action** es la ubicación donde se enviará el formulario cuando se complete y se envíe.

3 Tipos de Método

El atributo **action** funciona junto con el atributo **method** para determinar cómo se enviarán los datos del formulario.

Ejemplos de atributo action

El atributo **action** en un formulario HTML especifica la URL del servidor que procesará los datos enviados por el formulario. Algunos ejemplos de valores comunes para este atributo son:

1. **/enviar-formulario** - La ruta relativa al servidor donde se enviará el formulario.
2. **https://www.ejemplo.com/procesar-datos** - La URL completa del servidor que procesará los datos.
3. **mailto:example@example.com** - Una dirección de correo electrónico donde se enviarán los datos del formulario.

The image shows a web page with a contact form. The form consists of three input fields: a text field for 'NAME' containing 'name, ss', another text field for 'EMAIL' containing 'forryes', and a third text field for 'MESSAGE'. Below the form is a brown rectangular button with the text 'example.com' on it. The background of the page is a light beige color.

Atributo method (post, get)



Atributo method="post"

Envía los datos del formulario al servidor de forma más segura, ya que los datos se envían en el cuerpo de la solicitud HTTP y no se muestran en la URL.



Atributo method="get"

Envía los datos del formulario al servidor a través de la URL, lo que significa que los datos se hacen visibles y se limita la cantidad de información que se puede enviar.

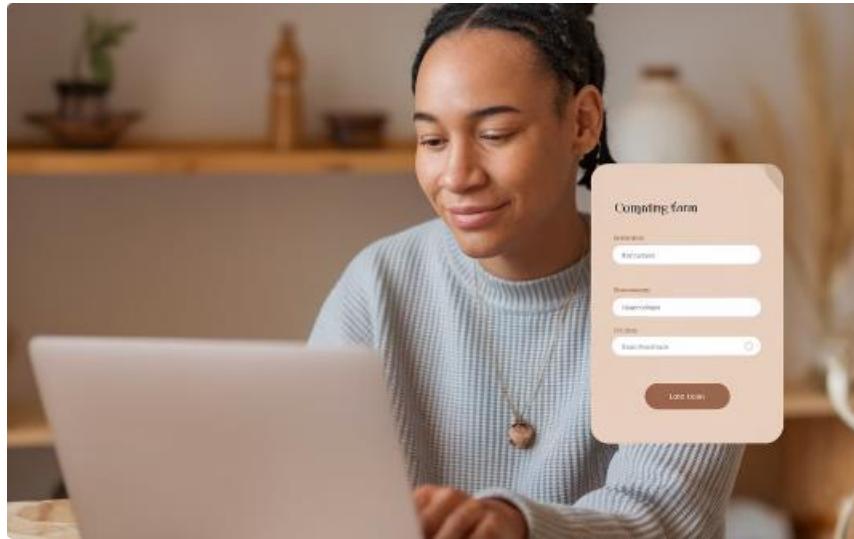


Consideraciones

La elección del método depende de la sensibilidad de los datos y del tamaño de la información que se transmite.



Ejemplos de atributo method



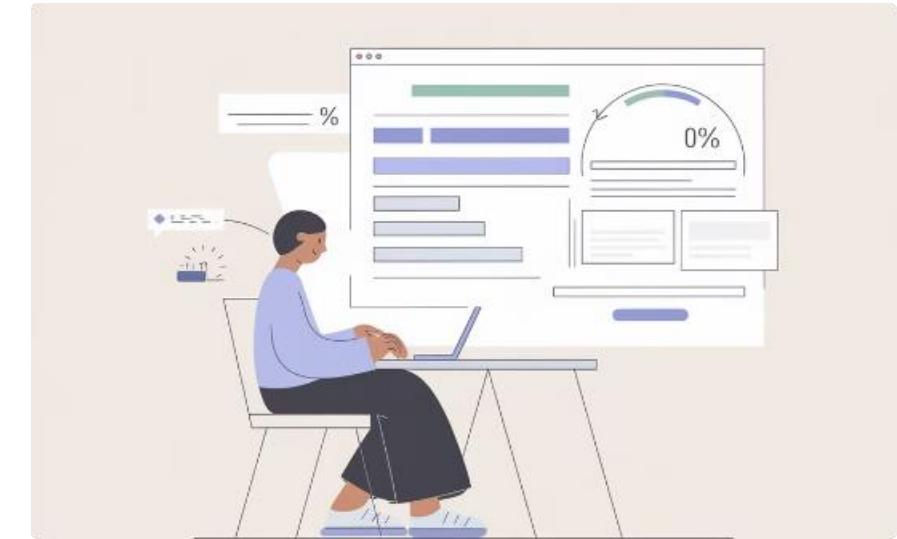
Ejemplo de método GET

En este ejemplo, el método GET se utiliza para enviar los datos del formulario a través de la URL, lo que permite una mejor visibilidad y rastreo de los datos.



Ejemplo de método POST

En este caso, el método POST se emplea para enviar información confidencial, como contraseñas o datos bancarios, de manera más segura a través del cuerpo de la solicitud HTTP.



Diferencias entre GET y POST

GET es mejor para consultas y solicitudes no seguras, mientras que POST es más adecuado para acciones que modifican datos o involucran información sensible.

Atributo target

Definición

El atributo **target** define dónde se abrirá el documento enlazado.

Valores comunes

_self (default): Abre el documento en la misma ventana/pestaña, **_blank**: Abre el documento en una nueva ventana/pestaña.

Uso en formularios

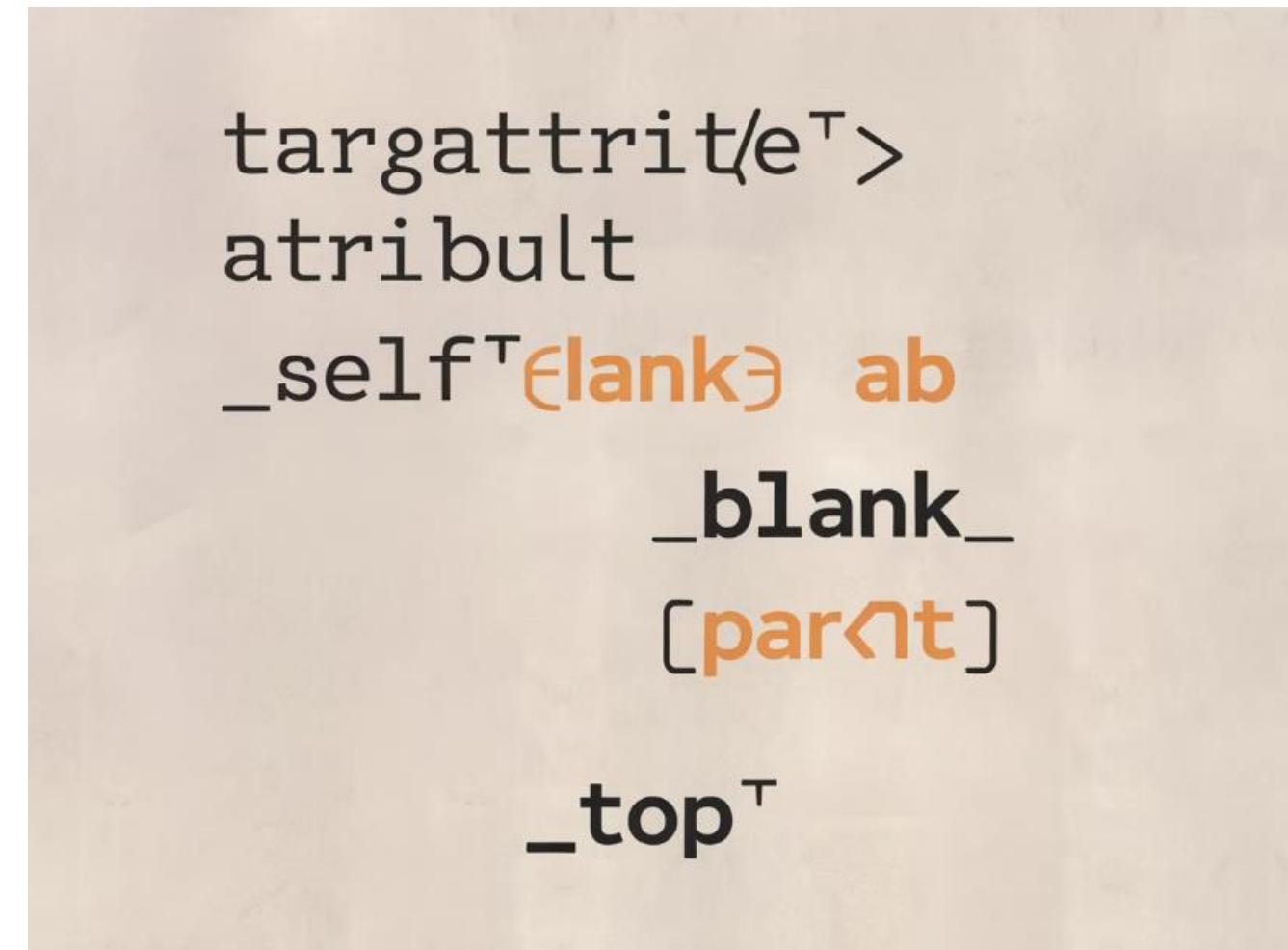
Permite especificar dónde se mostrará la respuesta del servidor después de enviar el formulario.

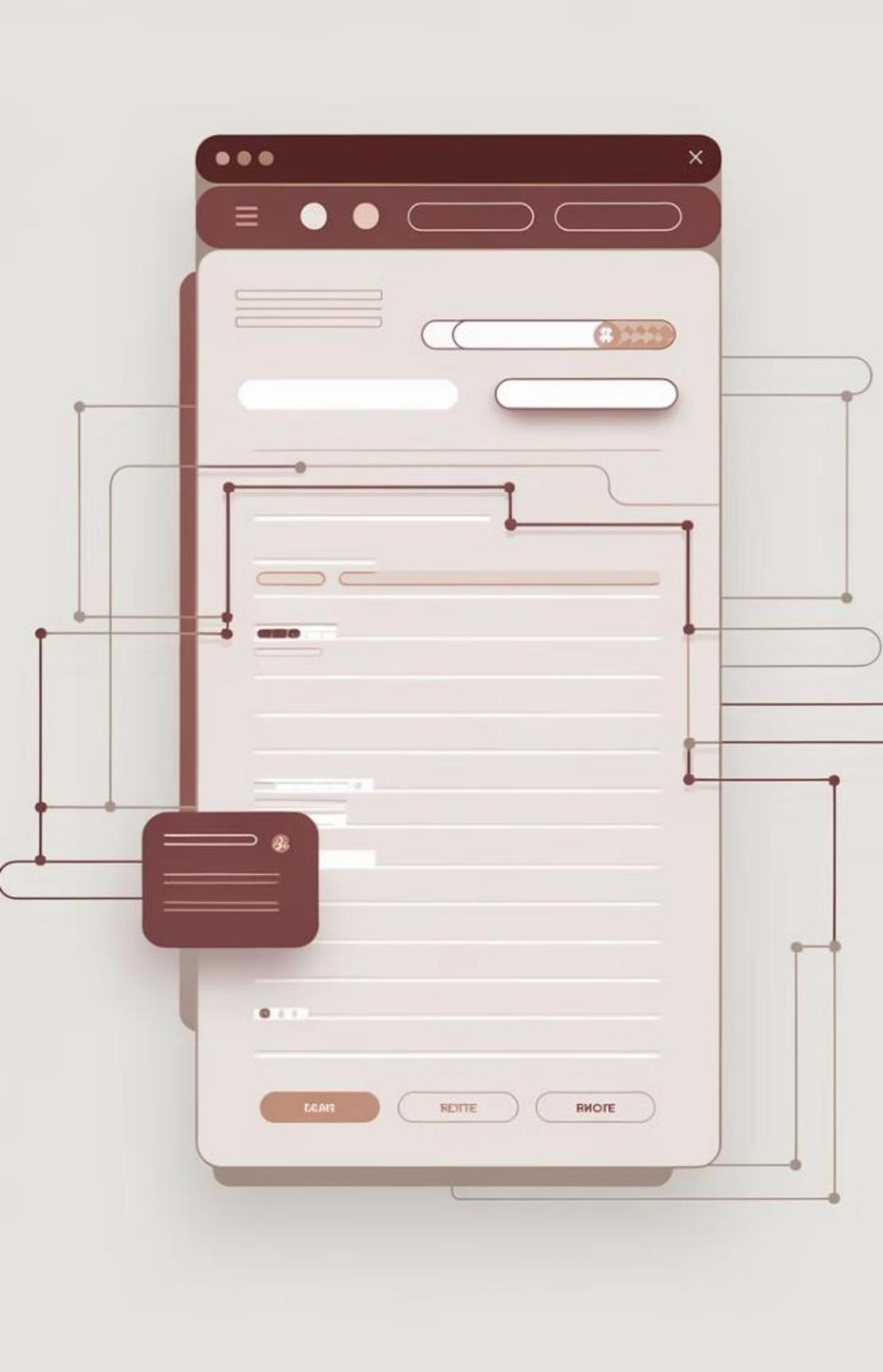


Ejemplos de atributo target

El atributo **target** se utiliza en los formularios HTML para especificar dónde se debe abrir la página de destino cuando se envía el formulario. Algunos valores comunes son:

- **_self** (valor predeterminado): Abre la página de destino en el mismo marco o ventana.
- **_blank**: Abre la página de destino en una nueva ventana o pestaña.
- **_parent**: Abre la página de destino en el marco padre.
- **_top**: Abre la página de destino en la ventana principal.





Resumen y conclusiones



Resumen

En esta presentación, hemos explorado los principales atributos de los formularios en HTML: name, action, method y target. Estos atributos permiten estructurar y configurar el comportamiento de los formularios web.



Conclusiones

Los formularios son elementos clave en el desarrollo web, ya que permiten a los usuarios interactuar con los sitios web y enviar información. Conocer a fondo estos atributos es fundamental para crear formularios robustos y funcionales.



Próximos pasos

A partir de esta base, se puede profundizar en el uso de los diferentes tipos de controles de formulario, la validación de datos y las técnicas de envío y procesamiento de la información.