



Actividad 2 - Pop-Up y Envío de Datos

Desarrollo de Sistemas Web I

Ingeniería en Desarrollo de Software

Tutor: Aarón Iván Salazar Macías

Alumno: Jusi Ismael Linares Gutiérrez

Fecha: 08/09/2023

Índice

Introducción	3
Descripción	3
Justificación	3
Desarrollo:	4
Codificación.....	4
App.js.....	4
Resultados.php	4
Capturas de pantalla	7
Conclusión	9
GitHub	9

Introducción

Para esta actividad número 2 de la materia Desarrollo de sistemas web 1 utilizaremos nuevamente las páginas creadas en nuestra actividad anterior y le añadiremos nuevas opciones para hacerla más dinámica. Con esta nueva actualización a nuestra página también haremos uso de otros lenguajes como lo serán JavaScript y PHP, aunque sean líneas básicas y sencillas las que utilizaremos para esta actividad, nos serán de gran utilidad para ir aprendiendo lo que podemos hacer con estos otros dos lenguajes, además aprenderemos a combinarlos con nuestros dos lenguajes anteriores (HTML y CSS) para poder crear páginas web más completas y complejas. JavaScript nos servirá para crear una ventana emergente que le pregunte al usuario si desea volver a ingresar datos a la página y al dar clic en el botón “Sí quiero” se mandará nuevamente al index para que el usuario pueda realizar el formulario nuevamente. Con PHP se enviarán los datos ingresados por el usuario en el index a la página resultados para que además de la imagen también se puedan visualizar las respuestas dadas por el usuario en el index.

Descripción

Utilizando nuestra actividad anterior actualizaremos el sistema para que ahora cada que el usuario envíe los datos requeridos por el formulario, todos estos se muestren en la página web del archivo resultados.html, además de que esta misma página contara con un nuevo botón el cual sirve para que el usuario regrese al index a ingresar nuevamente los datos requeridos en el formulario. En cuanto a la ventana emergente que se quiere implementar esta debe ser lanzada al momento de crear un nuevo registro, por ejemplo, si registro a mi hermano y doy clic en el botón de enviar datos, el index debe mandarme a la página de resultados y al final de la página debe haber un botón que diga “Volver a ingresar datos” que al darle clic debe lanzarse la ventana emergente que me pregunte si deseo volver a ingresar datos, por lo que si doy clic en el botón “Si quiero” debe mandarme nuevamente al index para que esta vez pueda ingresar mis datos. Estas actualizaciones se realizarán utilizando lenguajes PHP y JavaScript

Justificación

Esta actualización es muy importante para la página ya que lo que hicimos en la actividad 1 solo sirve para llenar datos y mandarte a la página de resultados por lo que no tiene ninguna función que haga uso de estos datos, pues en la actividad 1 nos dedicamos principalmente en que las páginas web se vieran bonitas y no tanto en hacer uso de la información del usuario por más sencilla que sea. Por eso para esta actividad 2 nos vamos a centrar en manejar la información ingresada por el usuario, aunque solo sea el enviar de una página a otra la información del usuario para después desplegarla en pantalla, esto nos será útil como una práctica de introducción al lenguaje PHP, pues hoy son solo datos no tan importantes de un usuario, pero el día de mañana en un trabajo podría tratarse de información más importante, por lo que hay que saber cómo enviarla apropiadamente.

Desarrollo:

Codificación

App.js

Empezaremos con el código del javascript app.js, var Alert = new CustomAlert(); Crea una nueva variable Alert y se le asigna la instancia CustomAlert y function CustomAlert() Esta función constructora se utiliza para crear objetos CustomAlert

```
var Alert = new CustomAlert();  
function CustomAlert() { /*Est
```

this.render = function(): Este método se utiliza para mostrar la alerta

let popUpBox = document.getElementById('popUpBox'); Obtiene el elemento con el ID popUpBox
popUpBox.style.display = "block"; Vuelve visible el elemento con el ID popUpBox como un cuadro emergente.

```
this.render = function() { /* Este método se utiliza par  
    let popUpBox = document.getElementById('popUpBox');  
    popUpBox.style.display = "block"; /* vuelve visible
```

Con esta línea se obtiene un elemento con el ID closeModal y se le establece su contenido HTML, En este caso un botón con un enlace

```
document.getElementById('closeModal').innerHTML = '<a href="index.html"><button type = "button" a href>¡Si quiero</button></a>';  
Esto define una función llamada ok, La función se ejecutará cuando se llame a ok()
```

```
this.ok = function() {
```

Esta línea sirve para seleccionar un elemento HTML con el atributo id igual a "popUpBox". Al establecer style.display en "none", el cuadro de diálogo emergente se ocultará y ya no será visible en la página. (Y al estar dentro de la función ok solo de ejecutara al ser llamada la función)

```
document.getElementById('popUpBox').style.display = "none";
```

Ocultar el elemento con el id popUpOverlay (Y al estar dentro de la función ok solo de ejecutara al ser llamada la función)

```
document.getElementById('popUpOverlay').style.display = "none";
```

Así queda finalmente nuestro archivo app.js

```
var Alert = new CustomAlert(); /* Crea una nueva variable Alert y se le asigna la instancia CustomAlert */  
function CustomAlert() { /*Esta función constructora se utiliza para crear objetos CustomAlert. */  
    this.render = function() { /* Este método se utiliza para mostrar la alerta*/  
        let popUpBox = document.getElementById('popUpBox'); /* Obtiene el elemento con el ID popUpBox */  
        popUpBox.style.display = "block"; /* Vuelve visible el elemento con el ID popUpBox como un cuadro emergente,  
        none es para ocultarlo */  
        document.getElementById('closeModal').innerHTML = '<a href="index.html"><button type = "button" a href>¡Si quiero</button></a>';  
        /*^ Se obtiene un elemento con el ID closeModal y se le establece su contenido HTML, En este caso un botón con un enlace*/  
    }  
    this.ok = function() { /* Esto define una función llamada ok, La función se ejecutará cuando se llame a ok() */  
        document.getElementById('popUpBox').style.display = "none"; /* Para seleccionar un elemento HTML con el atributo id igual a  
        "popUpBox". Al establecer style.display en "none", el cuadro de diálogo emergente se ocultará y ya no será visible en la página.  
        document.getElementById('popUpOverlay').style.display = "none"; /* Oculta el elemento con el id popUpOverlay */  
    }  
}
```

Resultados.php

Ahora pasamos con las nuevas líneas de código implementadas en nuestro archivo resultados.php
Adentro de este div se mostrará el contenido de la ventana emergente. Además gracias a app.js se volverá visible todo lo que este adentro de este div

```
<div id="popUpBox">
```

En este contenedor, se coloca el contenido específico de la ventana emergente, como texto, imágenes o elementos interactivos.

```
<div id="box">
```

Con i ponemos el texto en cursiva y con H1 como título dentro de los div popUpBox y box

```
<i class="fas fa-question-circle fa-5x"><h1>¿Volver a ingresar datos?</h1></i>
```

En este div se colocará el botón de app.js

```
<div id="closeModal"></div>
```

Cerramos los div popUpBox y box

```
</div>  
</div>
```

especifica una acción que se debe realizar cuando el usuario hace clic en el botón. En este caso, cuando se hace clic en el botón, se ejecutará la función JavaScript Alert.render del archivo app.js

```
<button onclick="Alert.render('You look very pretty today.')" class = "btn">¿Volver a Ingresar!</button>
```

Está cargando nuestro archivo JavaScript llamado "app.js" y ejecutando su contenido en la página web actual

```
<script src="./app.js"></script>
```

Finalmente así se ve todo nuestro nuevo código (aun lenguaje HTML) en resultados.php

```
<div id="popUpOverlay"></div>  
<div id="popUpBox"> <!--Representa el contenedor principal de la ventana emergente.  
Es dentro de este contenedor que se mostrará el contenido de la ventana emergente.-->  
  <div id="box"> <!--En este contenedor, se coloca el contenido específico de la ventana emergente,  
  como texto, imágenes o elementos interactivos.-->  
    <i class="fas fa-question-circle fa-5x"><h1>¿Volver a ingresar datos?</h1></i> <!-- Con i ponemos  
    con H1 como título -->  
    <div id="closeModal"></div> <!-- En este div se colocara el botón de app.js -->  
  </div>  
</div>  
<button onclick="Alert.render('You look very pretty today.')" class = "btn">¿Volver a Ingresar!</button>  
que se debe realizar cuando el usuario hace clic en el botón. En este caso, cuando se hace clic en el botón  
la función JavaScript Alert.render -->  
<script src="./app.js"></script> <!-- Está cargando nuestro archivo JavaScript llamado "app.js" y  
ejecutando su contenido en la página web actual. -->
```

[illegible]

```
<?php
```

```
$nombre = $_POST['Name'];
```

```
$edad = $_POST['Edad']; //Hacemos lo mismo con la edad
$ciudad = $_POST['Ciudad'];
$fec_nac = $_POST['Fec_Nac'];
$pasatiempo = $_POST['Pasatiempo'];
```

```
echo '<p><b>Nombre: </b>' . $nombre . '</p>';
```

Repetiremos el código de arriba, pero cambiando los nombres de las variables por las que correspondan

```
echo '<p><b>Ciudad: </b>' . $ciudad. '</p>'; //Lo mismo
echo '<p><b>Fecha Nacimiento: </b>' . $fec_nac. '</p>';
echo '<p><b>Pasatiempo: </b>' . $pasatiempo. '</p>';
```

Finalmente cerramos el espacio de código PHP

```
?>
```

Finalmente, así se ve todo junto nuestro código PHP y terminamos la implementación de nuestras nuevas líneas de código, por lo que solo queda probar la funcionalidad del programa

```
<?php
$nombre = $_POST['Name']; // Creamos la variable nombre
//donde name = Name, este dato se recibe y se guarda en
$edad = $_POST['Edad']; //Hacemos lo mismo para los demás
$ciudad = $_POST['Ciudad'];
$fec_nac = $_POST['Fec_Nac'];
$pasatiempo = $_POST['Pasatiempo'];

echo '<p><b>Nombre: </b>' . $nombre. '</p>'; //Con este
//ponemos código HTML para darle formato al texto, por n
echo '<p><b>Ciudad: </b>' . $ciudad. '</p>'; //Lo mismo
echo '<p><b>Fecha Nacimiento: </b>' . $fec_nac. '</p>';
echo '<p><b>Pasatiempo: </b>' . $pasatiempo. '</p>';
?>
```

Capturas de pantalla

Empezamos ingresando nuestros datos en la página de index.html y una vez ingresados los datos damos clic en el botón para mandarlos al resultados.php

CAPTURA DE DATOS PERSONALES

INGRESA LOS DATOS QUE SE TE PIDEN

MI PRIMERA ENCUESTA

Nombre	Jusi Ismael Linares Gutierr
Edad	20
Ciudad	Cuernavaca
Fecha de Nacimiento	27/10/2002
Pasatiempo Favorito	Videojuegos y Basketball

¡Ingresamos Datos!

Como podemos ver, los datos que ingrese anteriormente en index.html ahora se muestran en la página de resultados.php, ya que comprobamos que esto funciona correctamente daremos clic en el botón volver a ingresar



Nos lanza correctamente la ventana emergente por lo que solo nos queda dar clic en el botón "Sí quiero" que nos regresara a index.html



Nos manda correctamente al index.html, por lo que podemos ingresar distintos datos todas las veces que consideremos necesarias.



The image shows a web form titled "CAPTURA DE DATOS PERSONALES" (Personal Data Capture) with the subtitle "INGRESA LOS DATOS QUE SE TE PIDEN" (Enter the data requested). The form is set against a dark blue background. It includes the following fields:

- Nombre** (Name): A text input field with the placeholder "Ingresa tu nombre".
- Edad** (Age): A text input field.
- Ciudad** (City): A text input field with the placeholder "Ingresa tu ciudad".
- Fecha de Nacimiento** (Date of Birth): A date picker field showing "dd/mm/aaaa".
- Pasatiempo Favorito** (Favorite Hobbies): A text input field with the placeholder "Ingresa tu pasatiempo".

At the bottom of the form is a blue button labeled "¡Ingresamos Datos!" (We are entering data!).

Conclusión

En mi vida cotidiana y en mi trabajo, es esencial contar con la capacidad de crear formularios web y procesar datos. Esta práctica me ha brindado una comprensión sólida de cómo recopilar información de manera efectiva a través de formularios en línea y cómo presentar resultados de una manera organizada y atractiva.

Además, esta actividad me ha ayudado a familiarizarme con conceptos básicos de programación y desarrollo web, lo que podría ser útil en proyectos más avanzados y ambiciosos. A medida que continúe adquiriendo experiencia y habilidades, estoy seguro de que podré aplicar lo que he aprendido aquí en situaciones más complejas y desafiantes.

Aunque esta actividad pueda parecer simple, su importancia radica en la base sólida que proporciona para futuros proyectos y desafíos en mi campo laboral o en mi vida diaria. Las habilidades adquiridas aquí me servirán como un trampolín para abordar tareas más avanzadas y significativas en el futuro, pues hoy manejo datos utilizando php y html pero el día de mañana podría estar manejando mucha más información utilizando bases de datos como MySQL.

GitHub

https://github.com/JusiLinGu/Practicas_UMI