Universidad de la Cuenca del Plata
Autorización Definitiva. Decreto Poder Ejecutivo Nacional N°931/2006

Paradigmas y Lenguajes de Programación III

Desafío de Halloween

Manuel Zielinski, DNI: 44988773

Estimados alumnos, tenemos un desafío para hacerlo en la semana de halloween

El objetivo es crear una aplicación web simple que permita a los usuarios registrar y votar por sus disfraces de Halloween favoritos.

Pasos:

1. Configuración de la base de datos:

a. Crea una base de datos MySQL llamada "halloween" y configura una tabla llamada "disfraces" con las siguientes columnas:

```
-- -- Estructura de tabla para la tabla disfraces
CREATE TABLE disfraces ( id int(11) NOT NULL, nombre varchar(50) NOT NULL,
descripcion text NOT NULL, votos int(11) NOT NULL, foto varchar(20) NOT NULL,
foto_blob blob NOT NULL, eliminado int(11) NOT NULL DEFAULT 0 );

-- -- Estructura de tabla para la tabla usuarios
CREATE TABLE usuarios ( id int(11) NOT NULL, nombre varchar(50) NOT NULL,
clave text NOT NULL );

-- -- Estructura de tabla para la tabla votos
CREATE TABLE votos ( id int(11) NOT NULL, id_usuario int(11) NOT NULL,
id_disfraz int(11) NOT NULL );
```

2. Desarrollo de la aplicación web:

a. Crea una página principal (index.php) que muestre una lista de disfraces disponibles con sus nombres, descripciones y la cantidad de votos que han recibido.

b. Agrega un botón "Votar" junto a cada disfraz para permitir a los usuarios votar por su disfraz favorito. Debes prevenir votos duplicados de un mismo usuario.

c. Crea una página de registro (registro.php) que permita a los usuarios registrarse con un nombre de usuario y una contraseña.

d. Implementa un sistema de autenticación para asegurarte de que solo los usuarios registrados puedan votar.

e. Crea una página de inicio de sesión (login.php) que permita a los usuarios iniciar sesión con su nombre de usuario y contraseña.

Paradigmas y Lenguajes de Programación III

Desafío de Halloween

f. Desarrolla una página de administración (admin.php) que solo sea accesible para un usuario administrador. En esta página, el administrador puede agregar (**ABM**) nuevos disfraces a la base de datos.

3. Personalización:

a. Añade estilos CSS para darle un toque de Halloween a tu aplicación.

b. Puedes permitir que los usuarios carguen imágenes de sus disfraces junto con la descripción.

¡Este desafío debería ser un proyecto interesante para desarrollar durante Halloween!

Asegúrate de investigar y aprender sobre autenticación, seguridad y buenas prácticas de desarrollo web en PHP y MySQL mientras lo construyes.

¡Diviértete programando!

ANOTACIONES, tenemos algunas cositas que debemos utilizar.
Ustedes deberán ir completando para que utilizamos cada una de ellas en el desarrollo del proyecto.

- ☑ ~~mysqli_connect()~~
- ☑ ~~mysqli_query()~~
- ☑ ~~mysqli_num_rows()~~
- ☑ ~~mysqli_insert_id()~~
- ☑ ~~mysqli_num_fields()~~
- ☑ ~~mysqli_real_escape_string()~~
- ☑ ~~$_FILES['foto']['name']~~
- ☑ ~~explode(".",$archivo)~~
- ☑ ~~end($extension)~~
- ☑ ~~is_uploaded_file($_FILES['foto']['tmp_name'])~~
- ☑ ~~time()~~
- ☑ ~~copy($_FILES['foto']['tmp_name'], "fotos/".$qu.".".end($extension));~~
- ☑ ~~mysqli_error($con)~~
- ☑ ~~unlink('fotos/' . $_POST['foto_actual']);~~
- ☑ ~~isset($_POST['nombre'])~~
- ☑ ~~file_exists("fotos/".$r['foto'])~~
- ☑ ~~number_format($r['Precio'],2,',','.')~~

Universidad de la Cuenca del Plata
Autorización Definitiva. Decreto Poder Ejecutivo Nacional N°091/2006

Paradigmas y Lenguajes de Programación III

Desafío de Halloween

Ustedes deberán ir analizando y listando los puntos críticos donde se imaginan que es necesario aplicar algún mecanismo de seguridad con el fin de que la aplicación sea segura.

EL USUARIO ADMIN ES:

NOMBRE: ADMIN
CONTRASEÑA: ADMIN

Desarrollo

Composición de la base de datos:

- Tabla disfraces con columnas id, nombre, descripcion, votos, foto, foto_blob, y eliminado.
- Tabla usuarios con id, nombre, y clave.
- Tabla votos con id, id_usuario, y id_disfraz.

2. Desarrollo de la aplicación web:

a. Página principal (index.php):

Aquí, se muestra la lista de disfraces con nombre, descripción y votos recibidos, los cuales se almacenan en la base de datos.

Se ha utilizado mysqli_query() para realizar la consulta a la base de datos y obtener así los disfraces. mysqli_num_rows() se utiliza para saber la disponibilidad de los disfraces.

b. Botón "Votar":

Se ha logrado la prevención de duplicados utilizando la tabla votos, donde se verifica si un usuario ya ha votado por un disfraz, asegurando de esta manera que mysqli_num_rows() no devuelva registros existentes para un disfraz y usuario.

c. Página de registro (registro.php):

Se permite que los usuarios se registren con nombre de usuario y contraseña. Para asegurar la correcta inserción de datos, se utiliza mysqli_real_escape_string() para evitar inyecciones SQL.

d. Sistema de autenticación:

Con el uso mysqli_query() y mysqli_num_rows() verificamos si el usuario existe y la contraseña es la correcta.

e. Inicio de sesión (login.php):

Solo los usuarios registrados pueden iniciar sesión y acceder a la votación.

f. Panel de administración (admin.php):

Este panel solo es accesible mediante el usuario administrador, se implementa un control de acceso para verificar si el usuario es administrador o no a través de $_SESSION['es_admin'].

3. Personalización:

b. Cargar imágenes de disfraces:

Permite cargar imágenes de los disfraces junto con la descripción. Se verifica que las imágenes se hayan subido correctamente con is_uploaded_file() y se guarda la imagen en la carpeta correcta. Se utiliza explode(".",$archivo) y end($extension) para manejar extensiones de archivo y asegurar que estas se guarden de manera correcta.

4. Anotaciones:

mysqli_connect(): establece la conexión a la base de datos.

mysqli_query(): se utiliza para ejecutar las consultas SQL tales como: cómo obtener disfraces y verificar los votos.

mysqli_num_rows(): se utiliza para verificar si existen registros como lo son la validación de votos duplicados o usuarios registrados.

mysqli_real_escape_string(): se ha implementado para evitar las inyecciones SQL en consultas de autenticación e inserción.

$_FILES['foto']['name']: obtiene el nombre del archivo de la imagen que se ha subido.

is_uploaded_file(): verifica que el archivo se ha subido correctamente.

explode() y end(): se utilizan para trabajar con extensiones de archivo de imágenes.

time(): genera nombres únicos para las imágenes cargadas, evitando así conflictos de nombres.

copy(): se ha utilizado para mover la imagen a la carpeta deseada.

mysqli_error(): se utiliza para prevenir posibles errores en la db.

unlink(): NO se ha utilizado, pero puede ser útil en la edición de disfraces.

isset(): se ha utilizado para verificar si los datos del formulario han sido enviados.

file_exists(): No se ha utilizado, verifica si la imagen existe.

number_format(): NO se ha utilizado en la aplicación.

5. Seguridad:

- Se ha agregado la validación de datos con mysqli_real_escape_string() para evitar las inyecciones SQL.
- La autenticación y control de acceso han sido implementados. Asegurando así que solo los usuarios registrados pueden votar y solamente el administrador puede acceder a la página de administración y agregar distintos disfraces.
- Las imágenes que son cargadas, se gestionan adecuadamente para evitar la transferencia de archivos maliciosos.