

Actividad de desarrollo en Servidor.

Acceso a datos desde PHP a bases de datos MYSQL y MONGODB(III)

Índice de apartados

Descripción de la actividad.....	2
Clase VistaHTML.php.....	2
controladorAccesoBBDD.php	5
Cuestiones	6

Descripción de la actividad

Nuestro proyecto va a sufrir un **nuevo cambio para organizar mejor las cosas** . El problema es que el controlador tiene mezclado diferentes cosas en el código. Por un lado accede a la base de datos y por el otro tiene la codificación y formateo de la salida en HTML.

Vamos a encapsular y modularizar la parte de salida HTML a través de una clase. La clase se va a denominar VistaHTML y encapsulará toda la lógica de representación de datos en HTML de salida hacia el cliente

El código se muestra a continuación

Clase VistaHTML.php

Observa como **esta clase define una serie de métodos relacionados con la salida de los datos en HTML exclusivamente**.

Crea la clase en tú proyecto

```
<?php

class VistaHTML {

    // Método para mostrar la lista de personajes
    public function mostrarCharacters(array $characters) {
        echo "<div style='font-family: Arial, sans-serif;'>";
        echo "<h3 style='color: #4CAF50;'>Lista de personajes
del juego:</h3>";
        echo "<div style='display: flex; flex-wrap: wrap; gap:
10px;'>";

        foreach ($characters as $character) {
            echo "<div style='background-color: #f9f9f9;
padding: 10px; border-radius: 5px; width: 250px; box-shadow: 2px
2px 8px rgba(0, 0, 0, 0.2);'>";
            echo "<strong>Nombre:</strong>
{$character['name']}<br>";
            echo "<strong>Nivel:</strong>
{$character['level']}<br>";
        }
    }
}
```

```

        echo "<strong>Salud:</strong>
{$character['health']}<br>";

        echo "</div>";

    }

    echo "</div>";
    echo "</div>";
}

// Método para mostrar la lista de quests
public function mostrarQuests(array $quests) {
    echo "<div style='font-family: Arial, sans-serif;'>";
    echo "<h3 style='color: #4CAF50;'>Lista de desafíos del
juego:</h3>";
    echo "<ul style='list-style-type: none; padding: 0;'>";

    foreach ($quests as $quest) {
        echo "<li style='background-color: #f1f1f1; margin:
10px 0; padding: 10px; border-radius: 5px; box-shadow: 2px 2px
8px rgba(0, 0, 0, 0.1);'>";

        echo "<strong>Título:</strong>
{$quest['title']}<br>";

        echo "<strong>Recompensa:</strong>
{$quest['reward']}<br>";

        echo "<strong>Completada:</strong> " .
($quest['is_completed'] ? '<span style="color:
green;">Sí</span>' : '<span style="color: red;">No</span>') .
"<br>";

        echo "</li>";
    }

    echo "</ul>";
    echo "</div>";
}

```

```

    }

    // Método para mostrar las quests de un personaje específico
    public function mostrarQuestsDePersonaje(string $name, array
$quests) {
        echo "<div style='font-family: Arial, sans-serif;'>";
        echo "<h3 style='color: #4CAF50;'>Lista de desafíos del
personaje: $name</h3>";
        echo "<ul style='list-style-type: none; padding: 0;'>";

        foreach ($quests as $quest) {
            echo "<li style='background-color: #f1f1f1; margin:
10px 0; padding: 10px; border-radius: 5px; box-shadow: 2px 2px
8px rgba(0, 0, 0, 0.1);'>";
            echo "<strong>Título:</strong>
{$quest['title']}<br>";
            echo "<strong>Recompensa:</strong>
{$quest['reward']}<br>";
            echo "<strong>Completada:</strong> " .
($quest['is_completed'] ? '<span style="color:
green;">Sí</span>' : '<span style="color: red;">No</span>') .
"<br>";
            echo "</li>";
        }

        echo "</ul>";
        echo "</div>";
    }

    // Método para mostrar mensajes personalizados
    public function mostrarMensaje(string $mensaje) {
        echo "<div style='font-family: Arial, sans-serif;
padding: 10px; background-color: #e0f7fa; border: 1px solid

```

```

#00796b; border-radius: 5px; color: #004d40; margin-top:
15px; '>";

    echo "<strong>$mensaje</strong>";
    echo "</div>";
}

// Método para mostrar la información general del juego
public function mostrarInformacionJuego(array $characters,
array $quests) {
    echo "<div style='font-family: Arial, sans-serif; '>";
    echo "<h1 style='color: #2c3e50; '>Información del juego:
LISTA DE PERSONAJES Y LISTA DE DESAFÍOS</h1>";
    $this->mostrarCharacters($characters);
    $this->mostrarQuests($quests);
    echo "</div>";
}

// Método para mostrar errores
public function mostrarError(string $error) {
    echo "<div style='font-family: Arial, sans-serif; color:
#c0392b; background-color: #f9e2e2; padding: 10px; border: 1px
solid #e74c3c; border-radius: 5px; margin-top: 15px; '>";
    echo "<strong>Error:</strong> $error";
    echo "</div>";
}
}

```

controladorAccesoBBDD.php

Ahora **el controlador deberá usar esta clase cada vez que quiera imprimir** la salida. Para ello desde el controlador debes instanciar la clase VistaHTML y usarla para imprimir cada una de las salidas

En el primer caso (acción 1) quedaría algo como esto:

```

require_once 'VistaHTML.php';

$vista= new VistaHTML();

$repository = new RPGRepositoryMySQL("mysql", "rpg_game",
"root", "1234");

if (isset($_GET["accion"]) && ($_GET["accion"] == 1)) {
    $characters = $repository->findAllCharacters();
    $quests = $repository->findAllQuests();
    $vista->mostrarInformacionJuego($characters,$quests);
}

```

NOTA: Fíjate cómo ahora desde el controlador se obtiene la información de la base de datos (\$characters y \$quests) y se le pasa al componente VistaHTML para que lo muestre al Cliente.

Como puedes observar el controlador ahora “sale más limpio”

La nueva salida se parece a la siguiente pantalla

Información del juego: LISTA DE PERSONAJES Y LISTA DE DESAFÍOS

Lista de personajes del juego:

Nombre: Ulises
Nivel: 5
Salud: 100

Nombre: Penélope
Nivel: 4
Salud: 80

Nombre: Telemaco
Nivel: 3
Salud: 70

Nombre: Circe
Nivel: 6
Salud: 90

Nombre: Calipso
Nivel: 5
Salud: 85

Nombre: Atenea
Nivel: 7
Salud: 95

Lista de desafíos del juego:

Título: Viajar a Ítaca
Recompensa: 500
Completada: No

Título: Enfrentar a Polifemo
Recompensa: 400
Completada: Si

Título: Superar las Sirenas
Recompensa: 300
Completada: No

Título: Destruir a los pretendientes
Recompensa: 600
Completada: Si

Cuestiones

1. Haz un nuevo controlador **controladorAccesoRPG2.php** y utiliza dentro de él la nueva clase VistaHTML.php para mostrar todas las salidas hacia el cliente.
2. Imagina que quieres cambiar el color de la salida de los personajes en la primera opción ¿Qué componentes cambiarías? Hazlo y pruébalo

Los nuestros colores son los siguientes para characters y quests respectivamente

```
background-color: #e0f7e9
```

```
background-color: #e0f1ff
```

La nueva salida tendría el siguiente aspecto

Información del juego: LISTA DE PERSONAJES Y LISTA DE DESAFÍOS

Lista de personajes del juego:

Nombre: Ulises
Nivel: 5
Salud: 100

Nombre: Penélope
Nivel: 4
Salud: 80

Nombre: Telemaco
Nivel: 3
Salud: 70

Nombre: Circe
Nivel: 6
Salud: 90

Nombre: Calipo
Nivel: 5
Salud: 85

Nombre: Atenea
Nivel: 7
Salud: 95

Lista de desafíos del juego:

Título: Viajar a Ática
Recompensa: 500
Completada: No

Título: Enfrentar a Polifemo
Recompensa: 400
Completada: Si

Título: Superar las Sirenas
Recompensa: 300
Completada: No

Título: Destruir a los pretendientes
Recompensa: 600
Completada: Si

3. Imagina que la salida debe la aplicación debe ser en una interfaz móvil con dimensiones más reducidas ¿Qué cambiarías?