



## **PROCESO DE GESTIÓN DE FORMACIÓN PROFESIONAL INTEGRAL**

### **FORMATO GUÍA DE APRENDIZAJE**

#### **1. IDENTIFICACIÓN DE LA GUIA DE APRENDIZAJE**

- Denominación del Programa de Formación: Tecnólogo en Análisis y Desarrollo de Software
- Código del Programa de Formación: 2560979
- Nombre del Proyecto ( si es formación Titulada): Construcción de Software Orientado a Diferentes Sectores Productivos del Área Metropolitana de Cúcuta, Norte de Santander
- Fase del Proyecto ( si es formación Titulada): CONSTRUCCIÓN
- Actividad de Proyecto(si es formación Titulada)
- Competencia: 220501096 - DESARROLLAR LA SOLUCIÓN DE SOFTWARE DE ACUERDO CON EL DISEÑO Y METODOLOGÍAS DE DESARROLLO.
- Resultados de Aprendizaje Alcanzar:  
CONSTRUIR LA BASE DE DATOS PARA EL SOFTWARE A PARTIR DEL MODELO DE DATOS.
- Duración de la Guía

#### **2. PRESENTACIÓN**

- Esta guía de aprendizaje se presenta como complemento al taller realizado en la última formación: Creación de una base de datos.

#### **3. FORMULACIÓN DE LAS ACTIVIDADES DE APRENDIZAJE**

- CREA LA BASE DE DATOS INTEGRANDO LOS OBJETOS DE ACUERDO CON LA FUNCIONALIDAD DEL SOFTWARE.
- IMPLEMENTA RESTRICCIONES EN LA BASE DE DATOS DE ACUERDO CON LAS REGLAS DE DISEÑO.
- DEFINE ESQUEMAS DE SEGURIDAD EN LA BASE DE DATOS PARA MANTENER LA INTEGRIDAD DE LA INFORMACIÓN.

#### **TEMAS A DESARROLLAR**

##### **3.1. APLICACIÓN DEL PATRÓN MVC EN PHP**

##### **Características**



El patrón Modelo-Vista-Controlador para el diseño de aplicaciones Web es un estándar de la industria en el mundo Java. Hay muchos libros y recursos excelentes disponibles sobre el tema que ayudan a acelerar el proceso de aprendizaje para el equipo de desarrollo. En un breve repaso, MVC viene de Model, View, Controller, o bien: Modelo, Vista y Controlador. La idea básica de este patrón es separar nuestros sistemas en tres capas, el Modelo, la Vista y el Controlador.





- El Modelo se encarga de todo lo que tiene que ver con la persistencia de datos. Guarda y recupera la información del medio persistente que utilizemos, ya sea una base de datos, ficheros de texto, XML, etc.
- La Vista presenta la información obtenida con el modelo de manera que el usuario la pueda visualizar.
- El Controlador, dependiendo de la acción solicitada por el usuario, es el que pide al modelo la información necesaria e invoca a la plantilla(de la vista) que corresponda para que la información sea presentada.

Hay algo de esfuerzo necesario para aprender a utilizar un marco MVC en php. Sin embargo, para el desarrollador de aplicaciones Web grandes, este esfuerzo debe ser recompensado por los numerosos beneficios de utilizar un patrón de diseño MVC, tales como:

- Aplica la modularidad y la partición de aplicación.
- Aumenta la creación de roles específicos en el desarrollo.
- Aumenta la capacidad de gestión de código.
- Aumento de la extensibilidad del código (Capacidad de adaptación a cambios).

### 3.2. ACTIVIDAD

#### 3.2.1. Crear tabla Usuarios

	despachos usuarios
	Username : varchar(50)
	Password : varchar(50)
	Nombre : varchar(50)

#### 3.2.2. Rutinas

Crear en una carpeta para Proyecto MVC: archivo index.php y las subcarpetas Controlador, Vista, Modelo.

##### Index.php

```
<?php
require('Controller/C_verUsuarios.php');
?>
```

##### Controlador/C\_verUsuarios.php



```
<?php
require('Model/Conexion.php');
$con = new Conexion();
$usuarios = $con->getUsers();
require('Views/V_verUsuarios.php');
?>
```

### **Modelo/Conexion.php**

```
<?php

class Conexion{
    private $con;
    public function __construct()
    {
        $this->con = new mysqli('localhost', 'root', '', 'Despachos');
    }

    public function getUsers(){
        $query = $this->con->query('SELECT username, nombre FROM usuarios');
        $retorno = [];

        $i = 0;
        while($fila = $query->fetch_assoc()){
            $retorno[$i] = $fila;
            $i++;
        }
        return $retorno;
    }
}
?>
```

### **Vista/V\_verUsuarios.php**

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <meta http-equiv="X-UA-Compatible" content="ie=edge">
    <title>Lista De Usuarios</title>
</head>
<body>

    <table>
        <thead>
```



```
<tr>
  <td>Usuario</td>
  <!-- <td>Contraseña</td> -->
  <td>Nombre Real</td>
</tr>
</thead>

<tbody>
<?php
  foreach ($usuarios as $user) {
    echo "<tr>";
    echo "<td>". $user['username'] . "</td>";
    // echo "<td>". $user['password'] . "</td>";
    echo "<td>". $user['nombre'] . "</td>";
    echo "</tr>";
  }
?>
</tbody>
</table>
</body>
</html>
```

#### 4. OTRAS FUENTES

#### 5. CONTROL DEL DOCUMENTO

	Nombre	Cargo	Dependencia	Fecha
Autor (es)				

#### 6. CONTROL DE CAMBIOS (diligenciar únicamente si realiza ajustes a la guía)

	Nombre	Cargo	Dependencia	Fecha	Razón del Cambio
Autor (es)					