**Taller práctico.**

**Actividad N 25.**

**MVC\_CRUD parte 6.**

**Objetivo de la actividad:** Codificar las interfaces gráficas y módulos usando PHP y el servidor apache aplicando el concepto de MVC\_CRUD

Considerando una lógica similar de las vistas anteriores, se generan para el modelo producto en la ruta view/producto del proyecto:

**Paso 1.**

|  |
| --- |
| producto.php |
| <h1 class="page-header">Productos </h1>  <div class="well well-sm text-right">  <a class="btn btn-primary" href="?c=proveedor&a=Nuevo">Nuevo Proveedor</a>  <a class="btn btn-primary" href="?c=producto&a=Nuevo">Nuevo Producto</a>  </div>  <table class="table table-striped">  <thead>  <tr>  <th style="width:180px;">Código Producto</th>  <th style="width:120px;">NIT Proveedor</th>  <th style="width:120px;">Nombre Producto</th>  <th style="width:120px;">Precio Unitario</th>  <th style="width:120px;">Descripción</th>  </tr>  </thead>  <tbody>  <?php foreach($this->model->Listar() as $r): ?>  <tr>  <td><?php echo $r->idProducto; ?></td>  <td><?php echo $r->nit; ?></td>  <td><?php echo $r->nomprod; ?></td>  <td><?php echo $r->precioU; ?></td>  <td><?php echo $r->descrip; ?></td>  <td>  <a href="?c=producto&a=Crud&idProducto=<?php echo $r->idProducto; ?>">Editar</a>  </td>  <td>  <a onclick="javascript:return confirm('¿Seguro de eliminar este registro?');" href="?c=pro-  ducto&a=Eliminar&idProducto=<?php echo $r->idProducto; ?>">Eliminar</a>  </td>  </tr>  <?php endforeach; ?>  </tbody>  </table> |

**Paso 2.**

|  |
| --- |
| producto-nuevo.php |
| <h1 class="page-header">  Nuevo Registro  </h1>  <ol class="breadcrumb">  <li><a href="?c=producto">Productos</a></li>  <li class="active">Nuevo Registro</li>  </ol>  <form id="frm-producto" action="?c=producto&a=Guardar" method="post" enctype="multipart/  form-data">  <div class="form-group">  <label>Código Producto</label>  <input type="text" name="idProducto" value="<?php echo $prod->idProducto; ?>" class="-  form-control" placeholder="Ingrese Código Producto" data-validacion-tipo="requerido|min:20" />  </div>  <div class="form-group">  <label>Nit Proveedor</label>  <input type="text" name="nit" value="<?php echo $prod->nit; ?>" class="form-control" place-  holder="Ingrese NIT Proveedor" data-validacion-tipo="requerido|min:20" />  </div>  <div class="form-group">  <label>Nombre Producto</label>  <input type="text" name="nomprod" value="<?php echo $prod->nomprod; ?>" class="-  form-control" placeholder="Ingrese nombre producto" data-validacion-tipo="requerido|min:100"  />  </div>  <div class="form-group">  <label>Precio Unitario</label>  <input type="text" name="precioU" value="<?php echo $prod->precioU; ?>" class="form-con-  trol" placeholder="Ingrese precio unitario" data-validacion-tipo="requerido|min:20" />  </div>  <div class="form-group">  <label>Descripción del Producto</label>  <input type="text" name="descrip" value="<?php echo $prod->descrip; ?>" class="form-con-  trol" placeholder="Ingrese descripción producto" data-validacion-tipo="requerido|min:240" />  </div>  <hr />  <div class="text-right">  <button class="btn btn-success">Guardar</button>  </div>  </form>  <script>  $(document).ready(function(){  $("#frm-producto").submit(function(){  return $(this).validate();  });  })  </script> |

**Paso 3.**

|  |
| --- |
| producto-editar.php |
| <h1 class="page-header">  <?php echo $prod->idProducto != null ? $prod->nomprod : 'Nuevo Registro'; ?>  </h1>  <ol class="breadcrumb">  <li><a href="?c=producto">Productos</a></li>  <li class="active"><?php echo $prod->idProducto != null ? $prod->nomprod : 'Nuevo Registro';  ?></li>  </ol>  <form id="frm-producto" action="?c=producto&a=Editar" method="post" enctype="multipart/  form-data">  <input type="hidden" name="idProducto" value="<?php echo $prod->idProducto; ?>" />  <div class="form-group">  <label>Nit Proveedor</label>  <input type="text" name="nit" value="<?php echo $prod->nit; ?>" class="form-control" place-  holder="Ingrese NIT Proveedor" data-validacion-tipo="requerido|min:20" />  </div>  <div class="form-group">  <label>Nombre Producto</label>  <input type="text" name="nomprod" value="<?php echo $prod->nomprod; ?>" class="-  form-control" placeholder="Ingrese nombre producto" data-validacion-tipo="requerido|min:100"  />  </div>  <div class="form-group">  <label>Precio Unitario</label>  <input type="text" name="precioU" value="<?php echo $prod->precioU; ?>" class="form-con-  trol" placeholder="Ingrese precio unitario" data-validacion-tipo="requerido|min:20" />  </div>  <div class="form-group">  <label>Descripción del Producto</label>  <input type="text" name="descrip" value="<?php echo $prod->descrip; ?>" class="form-con-  trol" placeholder="Ingrese descripción producto" data-validacion-tipo="requerido|min:240" />  </div>  <hr />  <div class="text-right">  <button class="btn btn-success">Actualizar</button>  </div>  </form>  <script>  $(document).ready(function(){  $("#frm-producto").submit(function(){  return $(this).validate();  });  })  </script> |

Como actividad debes:

1. Crear los elementos descritos desde el paso 1, hasta el paso 6. Has captura de pantalla según la siguiente tabla: (recuerda hacer la descripción de cada paso empleado una buena redacción)

|  |
| --- |
| Captura de pantalla paso1 |
|  |
| Captura de pantalla paso2 |
|  |
| Captura de pantalla paso3 |
|  |

1. Una vez generado todo el código para el proyecto, con el servidor Xampp, php y mysql en ejecución de forma local, puede probar el proyecto en el browser, realice la captura de pantalla del funcionamiento final.

|  |
| --- |
| Captura de pantalla proyecto primerMVC en browser. |
|  |

1. ¿Qué es cakephp, menciona las ventajas y desventajas del framework? Explica en un párrafo de 500 palabras.

CakePHP es un marco de trabajo de código abierto para el lenguaje de programación PHP. CakePHP se basa en el patrón de diseño modelo-vista-controlador (MVC) y proporciona una amplia gama de características y herramientas integradas, como validación de entrada, protección contra CSRF, prevención de inyección SQL y más. CakePHP también ofrece una amplia gama de componentes y ayudantes integrados que facilitan tareas comunes, como el manejo de formularios, AJAX, autenticación, seguridad, caché y más. Además, CakePHP tiene una comunidad activa y amigable que contribuye al proyecto y ofrece soporte.

Entre las ventajas de CakePHP se encuentran su facilidad de uso, su capacidad para crear aplicaciones web responsivas y su amplia gama de componentes y herramientas integradas. Además, CakePHP sigue convenciones claras y predecibles, lo que facilita el desarrollo y mejora la legibilidad del código.

Entre las desventajas se encuentran su tamaño relativamente grande, la necesidad de adaptar el diseño a un grid de 12 columnas y la dificultad para cambiar de versión si se han realizado modificaciones profundas.

1. ¿Qué otros elementos puedo incluir al código del proyecto para mejorar su interfaz gráfica?.