

# **UNIVERSIDAD AUTÓNOMA GABRIEL RÉNE MORENO**

**FACULTAD DE INGENIERIA EN  
DE LA COMPUTACION Y TELECOMUNICACIONES**



## **Práctica No 5-SA**

### **Modificar Caso de Uso de MENU3C**

**PERTENECE A : Dominguez Molina Roberto Yúnior 218017138**

**GRUPO : 12-SA**

**MATERIA : Tecnología web**

**URL para revisión : <https://mail.tecnoweb.org.bo/inf513/grupo12sa/practica5>**

**Santa Cruz – Bolivia**

## Contenido

1.- Introducción.....	3
2. Antecedentes y Justificación.....	3
3 El problema.....	3
3.1 Situación problemática.....	3
3.2 Formulación del problema .....	4
4. Objetivos .....	4
4.1 Objetivo General .....	4
4.2 Objetivos específicos.....	4
5. Metodología .....	4
6. Marco teórico .....	5
6.1 Arquitectura 3 capas.....	5
6.2 Capas .....	5
7. Desarrollo.....	6
8. Resultados .....	8
9 Conclusión .....	10
10 Recomendación.....	11
11 Bibliografía .....	11

## Ilustraciones

Figure 1 código modificar persona .....	<b>¡Error! Marcador no definido.</b>
Figure 2 código listar persona.....	<b>¡Error! Marcador no definido.</b>
Figure 3 Código vista modificar .....	<b>¡Error! Marcador no definido.</b>
Figure 4 ingresar nueva persona .....	<b>¡Error! Marcador no definido.</b>
Figure 5 vista principal .....	<b>¡Error! Marcador no definido.</b>
Figure 6 Login.....	<b>¡Error! Marcador no definido.</b>
Figure 7 Modificar persona.....	<b>¡Error! Marcador no definido.</b>
Figure 8 Lista personas .....	<b>¡Error! Marcador no definido.</b>

# 1.- Introducción

El desarrollo de tres capas es la base fundamental de la programación Web. Por lo tanto, será usado en las materias de desarrollo de software y en todas aquellas aplicaciones web, tres capas se refieren a presentación, negocio y datos los cuales veremos en el siguiente ejercicio para su mejor entendimiento aplicado a un menú para modificar la agenda que hemos desarrollado.

## 2. Antecedentes y Justificación

La programación por capas es un modelo de desarrollo software en el que el objetivo primordial es la separación (desacoplamiento) de las partes que componen un sistema software o también una arquitectura cliente-servidor: lógica de negocios, capa de presentación y capa de datos. De esta forma, por ejemplo, es sencillo y mantenible crear diferentes interfaces sobre un mismo sistema sin requerirse cambio alguno en la capa de datos o lógica.

La ventaja principal de este estilo es que el desarrollo se puede llevar a cabo en varios niveles y, en caso de que sobrevenga algún cambio, solo afectará al nivel requerido sin tener que revisar entre el código fuente de otros módulos, dado que se habrá reducido el Acoplamiento informático hasta una interfaz de paso de mensajes.

## 3 El problema

### 3.1 Situación problemática

Se tiene una agenda con amigos que de desea subir a internet como un sistema capaz de agregar, modificar y eliminar amigos y datos que éstos tengan para una visualización mas sencilla.

## 3.2 Formulación del problema

Se desea Programar una agenda virtual para tener todos los datos antes descrito a mano desde el servidor tecnoweb.org.bo accesible desde cualquier navegador basándonos en la aplicación de tres capas para menú dinámico

## 4. Objetivos

### 4.1 Objetivo General

Desarrollo del caso de uso modificar agenda en un modelo basado en 3 capas con menú dinámico.

### 4.2 Objetivos específicos

- Elaborar las plantillas necesarias para el Caso de Uso Modificar Amigos de la Aplicación  
Agenda: nombre, teléfono, celular, correo electrónico, Agregue una foto.
- Elaborar el Código en el Negocio necesario para visualizar las plantillas de modificación de amigos.
- Elaborar Código necesario Actualizar los Datos de los Amigos: Especialmente subir fotos archivos validos
- Crear el Código Necesario para Listar los Amigos con sus respectivas Fotografías.

## 5. Metodología

Método de síntesis, la reunión de forma racional de elementos dispersos de un todo para su estudio.

## 6. Marco teórico

### 6.1 Arquitectura 3 capas

La arquitectura del software es el diseño de más alto nivel de la estructura de un sistema. Consiste en un conjunto de abstracciones que forman el “marco” del software. La arquitectura se diseña en la fase posterior a la de requisitos, la llamada fase de diseño. (Victor Gómez, junio 2020)

### 6.2 Capas

- Datos: Esta capa se encarga de guardar los datos. Será donde se gestione todo lo relativo a la base de datos y a la creación, edición y borrado de datos de ésta.
- Negocio: En esta capa se gestiona la lógica de la aplicación. Es donde se dice que se hace con los datos. Por ejemplo, para una aplicación de gestión de una biblioteca será donde se gestione cuántos préstamos puede tener un usuario, que ocurre si un usuario se retrasa al devolver un libro, etc. Estará conectada con la capa de persistencia para poder realizar sus funciones.
- Presentación: En esta capa se crea la interfaz del usuario. Su única función es pasarle las acciones que realice el usuario a la capa de negocio.

### 6.3. Menú dinámico

Respuesta: Se supone que el menú que tienes debería estar centrado, ya que al id header ya le agregamos un ancho y un margen automático, en el extraño caso de que no tuvieras centrado tu menú debes seguir los siguientes pasos.

Es muy sencillo, lo único que tienes que hacer es tomar todo el código que se encuentre dentro de la etiqueta `<ul class=»nav»>` y encerrarlo en otra etiqueta, en este caso utilizare una etiqueta html5 llamada nav, (etiqueta que debí usar en el menú desde el principio, pero para no confundirlos no la puse) y a esta última etiqueta nav agregarle con CSS un ancho y un margen automático para que se centre

## 7. Desarrollo

Creamos el menú responsive.

```
<a href="#" data-target="menu-responsive" class="sidenav-trigger">
  <i class="material-icons">menu</i>
</a>

<ul class="right hide-on-med-and-down">
{section name=op1 loop=$menu}
  <li><a class="dropdown-trigger" data-target="submenu-{$menu[op1].name}" href="#">{$menu[op1].name}</a></li>
  <!-- SubMenu dropdown-->
  <ul class="dropdown-content blue-text" id="submenu-{$menu[op1].name}">
    {section name=op2 loop=$menu[op1].submenu}
      <li><a class="blue-text text-darken-4" href="{$menu[op1].submenu[op2].link}">{$menu[op1].submenu[op2].name}</a></li>
    {/section}
  </ul>
</section>
</ul>
</div>
</div>
```

*Ilustración 1 menu responsive*

Podemos ver el index.php.

```
<?php

include('env.php');
require '/usr/share/php/Smarty/Smarty.class.php';
$smarty = new Smarty;
$smarty->compile_check = true;
$smarty->debugging = false;
$smarty->error_reporting = E_ALL & ~E_NOTICE;
$smarty->assign("titulo","Agenda Personal (Menu Dinamico)");
//Menu
$menu=array();
$link=chop('ps_login.php'. '?' . 'opcion=000','');
$subMenu[0]=array('name'=>'Ingresar','param'=>'opcion=000','link'=>$link);
$menu[0]=array('name'=>'Sistema','file'=>'ps_login.php','subMenu'=>$subMenu);
/*echo "<pre>";
print_r($menu);
echo "</pre>";*/

$smarty->assign("menu",$menu);
$smarty->assign("path",$path);
//Contenido
$smarty->assign("body","<center><h1>Agenda Personal<br>INF513-SA</h1></center>");
$smarty->assign("message","Bienvenido a la Agenda Personal INF513-SA");
$smarty->display('bienvenido.html');
```

*Ilustración 2 index.php*

El login y su clase que lo llama también es implementada.

```
{[extends file="layout.html"]
{block name="cuerpo"}

<div class="row">
  <form class="col s12 center" action="{$_GET}" method="GET">
    <div class="row card-panel blue-grey darken-1 white-text">
      <h4>Ingreso</h4>
      <div class="input-field col s12">
        <input class="white-text" name="user" id="user" type="text" class="validate" required>
        <label class="white-text" for="user">Usuario</label>
      </div>
      <div class="input-field col s12">
        <input class="white-text" name="pass" id="pass" type="password" class="validate" required>
        <label class="white-text" for="pass">Password</label>
      </div>
      <div>
        <button class="btn blue darken-4" type="submit">Ingresar</button>
      </div>
    </div>
  </form>
</div>
{/block}
```

```
</php>
include('../usr/share/php/adodb/adodb.inc.php');
include('env.php');
$user=$_GET['user'];
$password=$_GET['password'];

/**AHORA UTILIZAMOS ADODB PARA CONECTAR A LA BASE DE DATOS Y LOGRAR INDEPENDENCIA DE LA DB Y LA LG
$db = ADONewConnection($dbdriver);
$db->debug = false;
$db->Connect($dbserver, $dbuser, $dbpassword, $dbname);
$rs = $db->Execute("SELECT * FROM permiso WHERE per_cod='$user' AND perm_pass='$password'");
$db->Close();
$perm=$rs->GetArray();

/*echo "cpre";
print_r($perm);
echo "</pre";
echo "Nivel : ".$_user[0]['per_type'];*/
if (!$rs){
  //header ("Location: pc_mensaje.php?ref=Detalle&error=?". $db->ErrorMsg());
  print $db->ErrorMsg();
}
```

Ilustración 3 Login

Codificamos lo necesario para que la persona pueda logearse e ingresar a un menú principal donde estará contenida la agenda.

Vamos directamente con el motivo del tema y es la sección de amigos: modificar eliminar y listar.

```
case 1102:{ //Modificar
  $cod=$_GET['cod'];
  $nom=$_GET['nom'];
  $appm=$_GET['appm'];
  $tel=$_GET['tel'];
  $cel=$_GET['cel'];
  $email=$_GET['email'];
  $foto=$_GET['foto'];

  /**AHORA UTILIZAMOS ADODB PARA CONECTAR A LA BASE DE DATOS Y LOGRAR INDEPENDENCIA DE
  $db = ADONewConnection($GLOBALS[$dbdriver]);
  $db->debug = false;
  $sqlper="UPDATE persona " .
  "SET per_nom='$nom', per_appm='$appm', " .
  "per_telf='$tel', per_cel='$cel', per_email='$email', per_foto='$foto' " .
  "WHERE per_cod='$cod'";
  $db->Connect($GLOBALS[$dbserver], $GLOBALS[$dbuser], $GLOBALS[$dbpassword], $GLOBALS[$dbname]);
  $rs = $db->Execute($sqlper);
  $content="<h1>La Operacion se realizo con Exito.</h1>";
  $db->Close();

  $usuario = $usuario.$_GET['nom']." ";
  $usuario = $usuario.$_GET['appm']." ";
  if ($rs == false){
    return $this->showMensaje("Modificar Amigo","ERROR AL MODIFICAR","Ocurrio un Error
  }
  else {
    return $this->showMensaje("Modificar Amigo","EXITO AL MODIFICAR","Se Modifico Exito
  }
}
break;
```

Ilustración 4 modificar amigo

```

case 1104: { //Listar
//return $this->showMensaje("Listar Usuario","Aquí puedo colocar el código suficiente para <BR> LISTAR <BR> Puedes Listar","");
$db = ADONewConnection($GLOBALS[dbdriver]);
$db->debug = false;
$db->Connect($GLOBALS[dbserver], $GLOBALS[dbuser], $GLOBALS[dbpassword], $GLOBALS[dbname]);
$sql="SELECT * FROM persona;";
$rs = $db->Execute($sql);
$db->Close();
$amigos-$rs->GetArray();
/*echo "<pre>";
print_r($amigos);
echo "</pre>";*/

if ($rs == false){
return $this->showMensaje("Listar Amigo","ERROR AL LISTAR","Ocurrió un Error al listar <BR>".$sql,"ERROR al Listar...!!");
}else {
session_start();
$smarty = new Smarty;
$smarty->compile_check = true;
$smarty->debugging = true;
$smarty->assign("titulo","Listado de Amigos");
$smarty->assign("amigos",$amigos);
$smarty->assign("gestor","gestorAdmPer.php");
$smarty->assign("nuevo","1101");
$smarty->assign("mostrar","1105");
//Visualizar la información en el TEMPLATE
return $smarty->fetch('pc_per-listar.html');
}
}

```

Ilustración 5 eliminar y listar amigo

## 8. Resultados

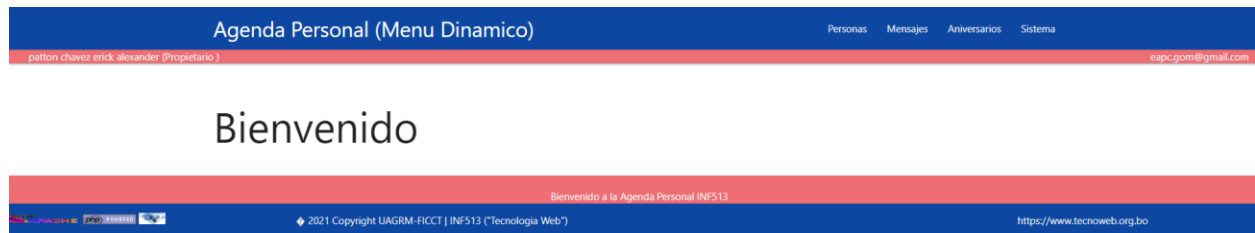





Ilustración 6 menu de entrada

Ilustración 7 Registrar amigo



Agenda Personal (Menu Dinamico)				Personas	Mensajes	Aniversarios	Sistema
Listado de Amigos							
3885821	Ureña Inarra pedro		<a href="#">Editar</a>				
123456	canchari juan		<a href="#">Editar</a>				
4012065	Colque Paulasanz Jose Manuel		<a href="#">Editar</a>				

*Ilustración 8 Listar amigo*

Sistema funcionando en internet explorer, problemas en el formulario sobre todo en los datos numéricos y tipo fecha, el menú desplega correctamente

Agenda Personal (Menu Dinamico)				Personas	Mensajes	Aniversarios	Sistema
						Registrar	
						Modificar	
						Eliminar	
						Listar	

### Registrar Amigo

Contraseña	Nombres	Apellidos
<input type="text"/>	<input type="text"/>	<input type="text"/>

C.I.	Fecha Nac.	Lugar Nac.
<input type="text"/>	<input type="text"/>	<input type="text"/>

Telefono	Celular	Email
<input type="text"/>	<input type="text"/>	<input type="text"/>

Profesion	Direccion
<input type="text"/>	<input type="text"/>

*Ilustración 9 fromulario en internet explorer*

Funcionando en Firefox, correcta las imágenes y el menú desplegable, error con el tipo de dato número.

Agenda Personal (Menu Dinamico)

PersonasMensajesAniversariosSistema

Registrar

Modificar

Eliminar

Listar

Registrar Amigo

Contraseña

•••••

Nombres

Apellidos

C.I.

Fecha Nac.  
dd/mm/aaaa

Lugar Nac.

Telefono

Celular




Email

Profesion

Direccion

REGISTRAR

Ilustración 10 formulario en firefox

3885821	Ureña Inarra pedro		<a href="#">Editar</a>
123456	canchari juan		<a href="#">Editar</a>
4012065	Colque Paulasanz Jose Manuel		<a href="#">Editar</a>

## 9 Conclusión

Podemos notar cómo el modelo 3 capas ayuda enormemente a la creación de cualquier sistema en la actualidad, dependiendo de la complejidad o de la necesidad podría ser una arquitectura a tomar en cuenta a la hora de trabajar de manera ordenada y limpia.

## 10 Recomendación

Hay que tomar mucha importancia con las fallas que pueda tener los navegadores como internet Explorer y Firefox en algunos casos, muchas personas usan éstos y pueden quedar inconformes si no se aplica una solución adecuada de antemano, en cuanto al fin de ésta práctica, todos los exploradores usados resultaron de manera satisfactoria a la hora de mostrar el menú desplegable.

## 11 Bibliografía

*Víctor Gómez, 23 de junio del 2020, sacado de [instintobinario.com](https://instintobinario.com) con el link <https://instintobinario.com/arquitectura-en-tres-capas/>*