
IDEAS PROYECTO PHP

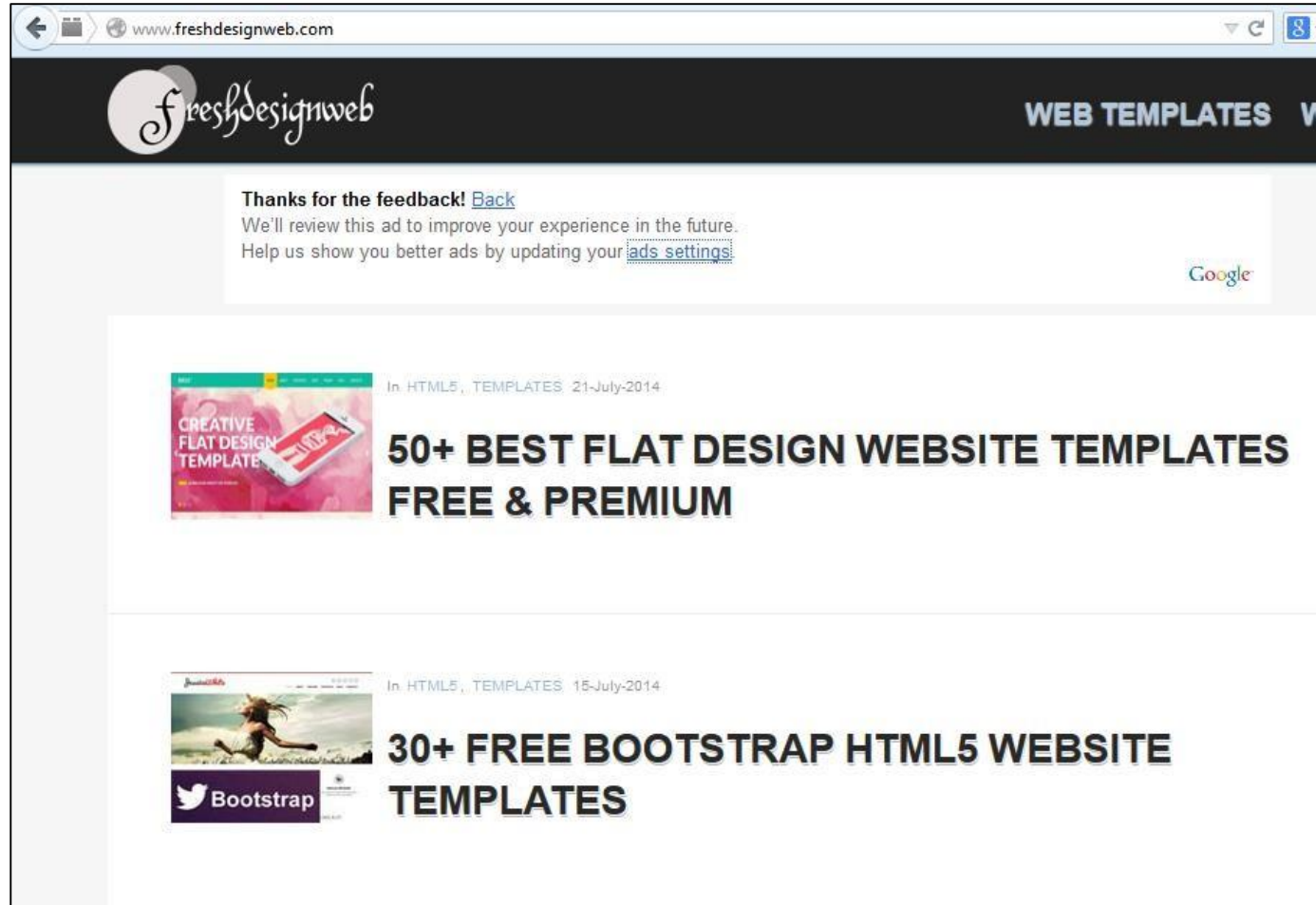
MARTA MORENO

INDICE

1. Seleccionar plantilla
2. Envío de mail (deprecated)
3. Localización GoogleMaps (deprecated)
4. Creación de un combobox con información de la BBDD
5. Inserción de datos a través de un formulario
6. Consulta y borrado de datos
7. Autocompletar una caja de texto con Ajax
8. Completar un combobox desde otro con Ajax
9. Login con Ajax

1. SELECCIONAR PLANTILLA

Paso 1. Web de plantillas gratuitas: FRESHDESIGNWEB.



1. SELECCIONAR PLANTILLA

Paso 2. Otra web de plantillas gratuitas: FREEHTML5TEMPLATES.



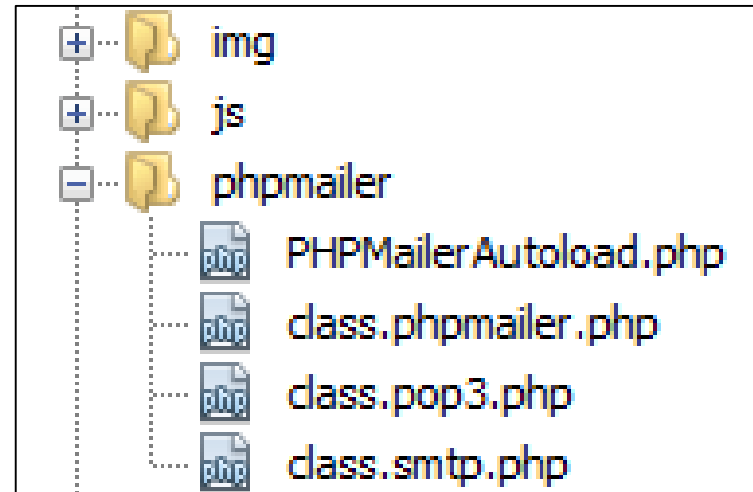
2. ENVIO DE MAIL (deprecated)

Paso 1. Definimos un formulario:

```
<form method="post" action="enviarmail.php" enctype="multipart/form-data">  
  Nombre: <input type="text" name="nombre">  
  Email: <input type="text" name="email">  
  Website: <input type="text" name="website">  
  Telefono: <input type="text" name="telefono">  
  Mensaje: <textarea name="mensaje"></textarea>  
  <input type="file" name="archivo">  
  <input type="hidden" name="MAX_FILE_SIZE" value="10000000" />  
  <input type="submit" value="Enviar" name="enviar">  
</form>
```

2. ENVIO DE MAIL (deprecated)

Paso 2. Creamos el directorio phpmailer en nuestro proyecto con las siguientes librerías:



2. ENVIO DE MAIL (deprecated)

Paso 3. Debemos agregar en nuestra página web la función SendMail:

```
<?php
function SendMail($ToEmail, $asunto, $filepath, $filename, $MessageTEXT) {
    require_once ( 'phpmailer/class.phpmailer.php' ); // Add the path as appropriate
    $Mail = new PHPMailer();
    $Mail->IsSMTP(); // Use SMTP
    $Mail->Host      = "smtp.gmail.com"; // Sets SMTP server
    $Mail->SMTPDebug  = 2; // 2 to enable SMTP debug information
    $Mail->SMTPAuth    = TRUE; // enable SMTP authentication
    $Mail->SMTPSecure  = "tls"; //Secure connection
    $Mail->Port        = 587; // set the SMTP port
    $Mail->Username     = 'eduard.lara.ochoa@gmail.com'; // SMTP account username
    $Mail->Password     = ''; // SMTP account password
    $Mail->Priority      = 1; // Highest priority - Email priority (1 = High, 3 = Normal, 5 = low)
    $Mail->CharSet      = 'UTF-8';
    $Mail->Encoding     = '8bit';
    $Mail->Subject      = $asunto;
    $Mail->ContentType  = 'text/html; charset=utf-8\r\n';
    $Mail->From         = 'eduard.lara.ochoa@gmail.com';
    $Mail->FromName     = 'Eduard Lara';
    $Mail->WordWrap     = 900; // RFC 2822 Compliant for Max 998 characters per line
    $Mail->AddAttachment($filepath, $filename);
    $Mail->AddAddress( $ToEmail ); // To:
    $Mail->isHTML( TRUE );
    $Mail->Body         = $MessageTEXT;
    //$Mail->AltBody = $MessageTEXT;
    $Mail->Send();
    $Mail->SmtpClose();

    if ( $Mail->IsError() ) { // ADDED - This error checking was missing
        return FALSE;
    } else {
        return TRUE;
    }
}
```

2. ENVIO DE MAIL (deprecated)

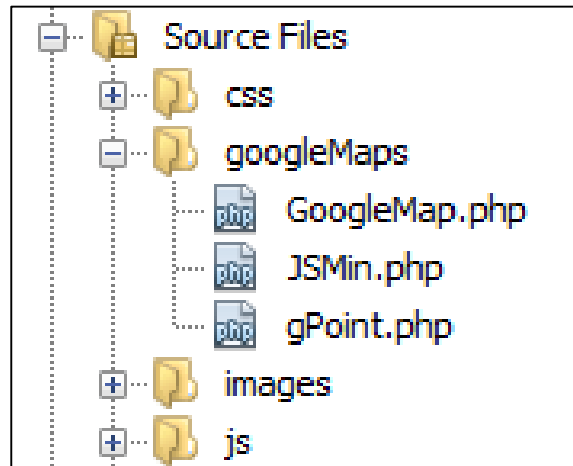
Paso 4. Llamamos a la función Sendmail con los parámetros del formulario recogidos con POST:

```
if ($_POST){
    $ToEmail = $_POST['email'];
    $ToName = $_POST['nombre'];
    $ToWeb = $_POST['website'];
    $ToTelf = $_POST['telefono'];
    $asunto = "Consulta sobre VeganBCN ".$ToName." ".$ToWeb." ".$ToTelf;
    $MessageTEXT = $_POST['mensaje'];
    if (is_uploaded_file($_FILES['archivo']['tmp_name']))
        $Send = SendMail( $ToEmail, $asunto, $_FILES['archivo']['tmp_name'], $_FILES['archivo']['name'], $MessageTEXT);
    else
        $Send = SendMail( $ToEmail, $asunto, "", "", $MessageTEXT);

    if ($Send) {
        echo "<h2>En breve momentos nos pondremos en contacto con usted</h2>";
    }
    else {
        echo "<h2> ERROR</h2>";
    }
    die;
}
```


3. GOOGLEMAPS (deprecated)

Paso 1. Creamos el directorio googleMaps en nuestro proyecto con las siguientes librerías:



3. GOOGLEMAPS (deprecated)

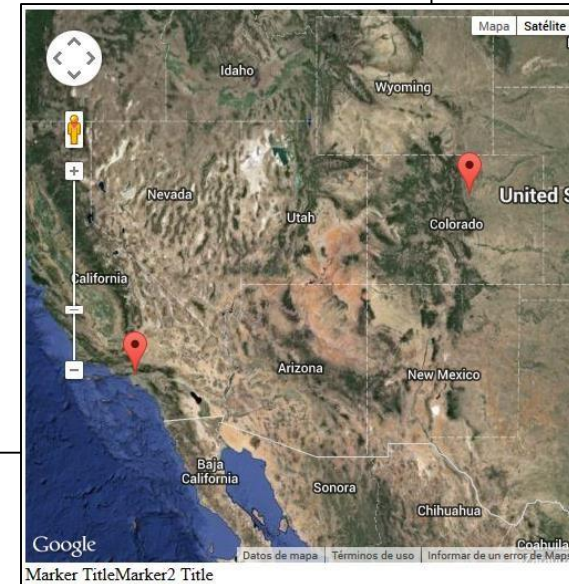
Paso 2. Estructura básica de un mapa de googleMaps. Consultar la web:
<http://bradwedell.com/php-google-maps-api/demos>

```
<?php
include_once("googleMaps/GoogleMap.php");
include_once("googleMaps/JSTMin.php");

$MAP_OBJECT = new GoogleMapAPI();
$MAP_OBJECT->_minify_js = isset($_REQUEST["min"]) ? FALSE : TRUE;
$MAP_OBJECT->setDSN("mysql://root@localhost/negocis");

$MAP_OBJECT->addMarkerByCoords(-104.988,39.755,"Marker Title", "Marker Description");
$MAP_OBJECT->addMarkerByCoords(-118.243,34.0522,"Marker2 Title", "Marker2 Description");
?>

<html>
<head>
<?=$MAP_OBJECT->getHeaderJS();?>
<?=$MAP_OBJECT->getMapJS();?>
</head>
<body>
<?=$MAP_OBJECT->printOnLoad();?>
<?=$MAP_OBJECT->printMap();?>
<?=$MAP_OBJECT->printSidebar();?>
</body>
</html>
```



3. GOOGLMAPS (deprecated)

Paso 3. Ejemplo para insertar diferentes localizaciones a partir de una consulta en la base de datos:

```
<?php
include_once("googleMaps/GoogleMap.php");
include_once("googleMaps/JSMin.php");

$MAP_OBJECT = new GoogleMapAPI();
$MAP_OBJECT->_minify_js = isset($_REQUEST["min"]) ? FALSE : TRUE;
$MAP_OBJECT->setDSN("mysql://root@localhost/negocis");

$con = mysql_connect("localhost", "root", "");
mysql_select_db("negocis");
$sql = "SELECT `negoci`.`Nom` , `negoci`.`Lat` , `negoci`.`Lon` FROM `negoci`";

$result = mysql_query($sql, $con);
while ($row = mysql_fetch_array($result)) {
    $nombre = $row[0];
    $latitud = $row[1];
    $longitud = $row[2];
    $MAP_OBJECT->addMarkerByCoords($longitud, $latitud, $nombre, $nombre, $nombre);
}
?>
```

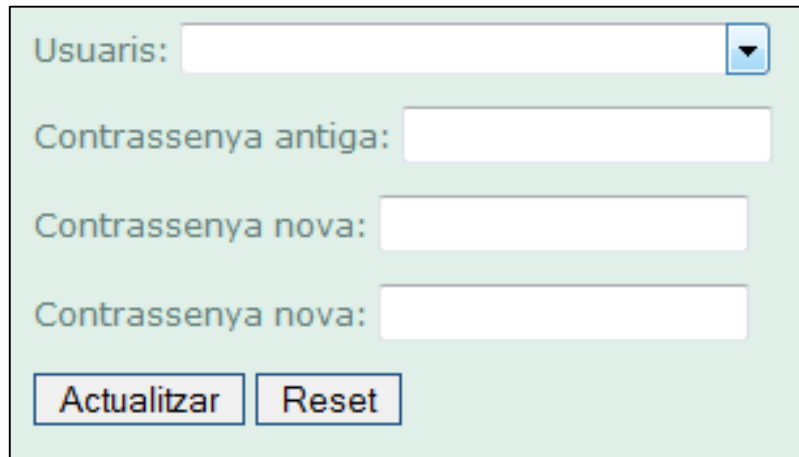
4. CREACION COMBOBOX CON BBDD

Paso 1. La información que se enviara por post del combobox se indica en el atributo value del option. Este valor se recoge con \$_POST['usuario'].

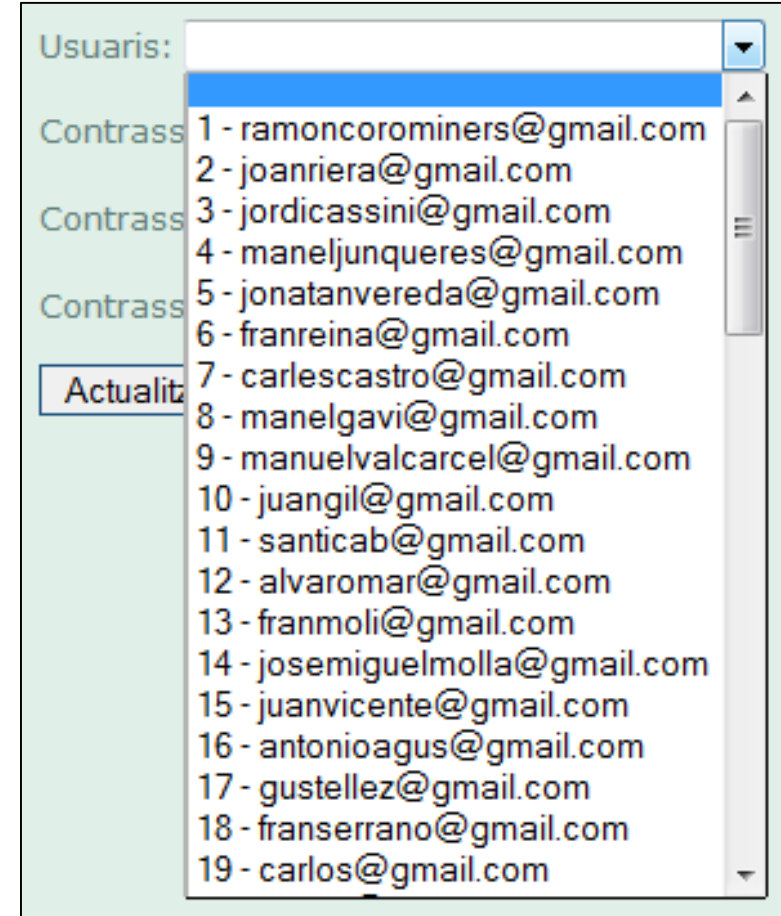
```
<form action="admin.php" method="post">
  Usuaris: <select name="usuario">
    <option value="" selected></option>
    <?php
      $conn = mysqli_connect("localhost", "root", "", "xarxatallers");
      $sql = "SELECT 'ID', 'EMAIL' FROM 'usuaris'";
      $result = mysqli_query($con, $sql);
      while ($row = mysqli_fetch_array($result))
        echo "<option value='$row[0]'>$row[0] - $row[1]</option>";
      echo "</select><br><br>";
    ?>
  Contrassenya antiga:<input type="text" name="password_anterior"/><br><br>
  Contrassenya nova:  <input type="password" name="password_nuevol"/><br><br>
  Contrassenya nova:  <input type="password" name="password_nuevo2"/><br><br>
                      <input type="submit" value="Actualitzar"/>
                      <input type="reset" value="Reset"/>
</form>
```

4. CREACION COMBOBOX CON BBDD

Paso 2. Este es el efecto:



Formulario de actualización de contraseña. El campo 'Usuaris:' es un combobox con una flecha hacia abajo. Debajo hay tres campos de texto para 'Contrassenya antiga:', 'Contrassenya nova:' y 'Contrassenya nova:'. Al final hay dos botones: 'Actualitzar' y 'Reset'.



Detalle del combobox de usuarios. El campo 'Usuaris:' está desplegado, mostrando una lista de 19 usuarios. El primer usuario, '1 - ramoncorominers@gmail.com', está seleccionado y resaltado en azul. Debajo de la lista hay un botón 'Actualitz'.

Id	Usuari
1	ramoncorominers@gmail.com
2	joanriera@gmail.com
3	jordicassini@gmail.com
4	maneljunqueres@gmail.com
5	jonatanvereda@gmail.com
6	franreina@gmail.com
7	carlescastro@gmail.com
8	manelgavi@gmail.com
9	manuelvalcarcel@gmail.com
10	juangil@gmail.com
11	santicab@gmail.com
12	alvaromar@gmail.com
13	franmoli@gmail.com
14	josemiguelmolla@gmail.com
15	juanvicente@gmail.com
16	antonioagus@gmail.com
17	gustellez@gmail.com
18	franserrano@gmail.com
19	carlos@gmail.com

4. CREACION COMBOBOX CON BBDD

Paso 3. Un hipotético procesado del formulario anterior con el select construido a través de la base de datos sería:

```
<?php
if ($_POST){
    $id = $_POST['usuario'];
    $old_passw = $_POST['password_anterior'];
    $passw1 = $_POST['password_nuevo1'];
    $passw2 = $_POST['password_nuevo2'];

    if ($id=="" || $old_passw=="" || $passw1=="" || $passw2==""){
        echo "Hauria d'omplir tots els camps!!!";
    }else{
        $sql = "SELECT 'PASSWORD','EMAIL' FROM 'usuaris' WHERE ID=$id;";
        $result = mysqli_query($con,$sql);
        $row = mysqli_fetch_array($result);
        $passbd = $row[0];
        $mail = $row[1];
        if ($passbd != $old_passw)
            echo "Contrassenya erronia!!!";
        else if ($passw1 != $passw2)
            echo "Contrassenyes noves no coincideixen!!!";
        else{
            $sql = "UPDATE usuaris SET PASSWORD='$passw1' WHERE ID=$id;";
            if (mysqli_query($con,$sql))
                echo "Canvi de contrassenya is l'usuari ".$mail." realitzat correctament";
            else
                echo "Canvi is contrassenya de l'usuari ".$mail." realitzat incorrectament";
        }
    }
}
echo "</div>";
?>
```

5. INSERCIÓN BBDD DESDE FORMULARIO

Paso 1. Formulario para la inserción de datos de usuario:

Nombre:

Apellido:

Edad:

Mail:

Direccion:

```
<form action="admin.php" method="post">
  Nombre:  <input type="text" name="nombre"/><br><br>
  Apellido: <input type="text" name="apellido"/><br><br>
  Edad:    <input type="text" name="edad"/><br><br>
  Mail:    <input type="text" name="mail"/><br><br>
  Direccion: <input type="text" name="direccion"/><br><br>
            <input type="submit" value="Insertar" />
</form>
```

5. INSERCIÓN BBDD DESDE FORMULARIO

Paso 2. Script base de datos usuario:

```
create table USUARIOS
(
  IDUSUARIO      int not null auto_increment,
  NOMBRE         varchar(10) not null,
  APELLIDO       varchar(10) not null,
  EDAD           int not null,
  MAIL           varchar(20) not null,
  DIRECCION      varchar(20) not null,
  primary key (IDUSUARIO)
) engine=InnoDB charset=utf8 auto_increment=0;
```


5. INSERCIÓN BBDD DESDE FORMULARIO

Paso 3. Procesado para la inserción de los datos del formulario:

```
<?php
    if ($_POST){
        $nombre = $_POST['nombre'];
        $apellido = $_POST['apellido'];
        $edad = $_POST['edad'];
        $mail = $_POST['mail'];
        $direccion = $_POST['direccion'];
        $con = mysqli_connect("localhost", "root", "", "sql_db");
        $sql = "INSERT INTO USUARIOS (IDUSUARIO, NOMBRE, APELLIDO, EDAD, MAIL, DIRECCION) "
            . " VALUES (null, '$nombre', '$apellido', $edad, '$mail', '$direccion') ";
        if (mysqli_query($con, $sql))
            echo "Insercion realizada correctamente";
        else
            echo mysqli_error();
    }
?>
```

5. INSERCIÓN BBDD DESDE FORMULARIO

Paso 4. Ejecución del formulario y script de inserción:

Nombre:

Apellido:

Edad:

Mail:

Direccion:

+ Opciones

		IDUSUARIO	NOMBRE	APELLIDO	EDAD	MAIL	DIRECCION
<input type="checkbox"/>	Editar Copiar Borrar	1	edu	lara	33	edu@gmail.com	Calle 2000
	<input type="checkbox"/> Marcar todos	Para los elementos que están marcados:			Cambiar	Borrar	Exportar

6. CONSULTA Y BORRADO DE DATOS (GET)

Paso 1. Mostrar los datos de un tabla con un link de borrado

```
<?php
$con = mysqli_connect("localhost", "root", "", "sql_db");
$sql = "SELECT IDUSUARIO, NOMBRE, APELLIDO, EDAD, MAIL, DIRECCION FROM USUARIOS;";
$res = mysqli_query($con,$sql);
echo "<table border=2>";
while ($row = mysqli_fetch_array($res)) {
    echo "<tr><td>$row(0)<td>$row(1)<td>$row(2)<td>$row(3)<td>$row(4)<td>$row[5]" .
        "<td><a href=admin.php?accion=borradocid=$row[0]>Borrar</a></td>";
} echo "</table><br>";
?>
```

1	Edu	lara	23	edu@gmail.com	calle 2000	Borrar
2	Marc	Martinez	33	marc@gmail.com	calle 3000	Borrar
3	Josep	Pujol	43	josep@gmail.com	calle 5000	Borrar

Nombre:

Apellido:

Edad:

Mail:

Direccion:

6. CONSULTA Y BORRADO DE DATOS (GET)

Paso 2. Procesado de la acción de borrado de un determinado usuario, haciendo click en el link de borrado (los datos son pasados por GET[]).

```
<?php
    if (isset($_GET[accion]) && $_GET[accion]=="borrado"){
        $id=$_GET[id];
        $con = mysqli_connect("localhost", "root", "", "sql_db");
        $sql = "DELETE FROM USUARIOS WHERE IDUSUARIO=$id;";
        if (mysqli_query($con,$sql))
            echo "Borrado de usuario id=$id realizado correctamente";
        else
            echo mysqli_error ();
    }
?>
```

6. CONSULTA Y BORRADO DE DATOS (GET)

Paso 3. Borrado de un usuario al hacer click en su link de borrado y actualización de la tabla de usuarios:

1	Edu	lara	23	edu@gmail.com	calle 2000	Borrar
2	Marc	Martinez	33	marc@gmail.com	calle 3000	Borrar
3	Josep	Pujol	43	josep@gmail.com	calle 5000	Borrar

Nombre:

Apellido:

Edad:

Mail:

Direccion:

Borrado de usuario id=3 realizado correctamente

1	Edu	lara	23	edu@gmail.com	calle 2000	Borrar
2	Marc	Martinez	33	marc@gmail.com	calle 3000	Borrar

Nombre:

Apellido:

Edad:

Mail:

Direccion:

6. CONSULTA Y BORRADO DE DATOS (POST)

Paso 1. Para conseguir que desde un enlace <a href> se pasen parámetros por POST, se debe hacer uso de una función javascript, la cual hará el envío de los datos usando un formulario oculto en el fichero.

```
<?php
    $con = mysqli_connect("localhost", "root", "", "sql_db");
    $sql = "SELECT IDUSUARIO, NOMBRE, APELLIDO, EDAD, MAIL, DIRECCION FROM USUARIOS;";
    $res = mysqli_query($con,$sql);
    echo "<table border=2>";
    while ($row = mysqli_fetch_array($res))
        echo "<tr><td>$row[0]<td>$row[1]<td>$row[2]<td>$row[3]<td>$row[4]<td>$row[5] "
        . "<td><a href=javascript:EnviarPOST('borrado','$row[0]')>Borrar</a></td>";
    echo "</table><br>";
?>
```

6. CONSULTA Y BORRADO DE DATOS (POST)

Paso 2. El formulario tendrá dos inputs ocultos (hidden) que serán cargados y enviados (submit) a través del DOM de Javascript:

```
<form name="formulario" action="admin.php" method="post">
  <input type="hidden" name="accion"/>
  <input type="hidden" name="id"/>
</form>
```

```
<script type="text/javascript">
  function EnviarPOST(accion, id){
    document.forms.formulario.accion.value= accion;
    document.forms.formulario.id.value = id;
    document.forms.formulario.submit();
  }
</script>
```

6. CONSULTA Y BORRADO DE DATOS (POST)

Paso 3. El procesamiento será el mismo que en el caso del GET, pero ahora con POST:

```
<?php
    if (isset($_POST[accion]) && $_POST[accion]=="borrado"){
        $id=$_POST[id];
        $con = mysqli_connect("localhost", "root", "", "sql_db");
        $sql = "DELETE FROM USUARIOS WHERE IDUSUARIO=$id;";
        if (mysqli_query($con,$sql))
            echo "Borrado de usuario id=$id realizado correctamente";
        else
            echo mysqli_error ($);
    }
?>
```


7. COMPLETAR UN COMOBOBOX DESDE OTRO

AJAX = JAVASCRIPT + PHP

Paso 1. Creamos dos select box: el primero se rellena con datos de la base de datos, y el segundo, que es dependiente del primero, lo dejamos vacío.

```
Serveis: <select name="idservei" id="servei">
    <option value='' selected></option>
    <?php
        $con      = mysqli_connect("localhost", "root", "", "xarxatallers");
        $sql       = "select ID, nom from serveis;";
        $result    = mysqli_query($con, $sql);
        while ($row = mysqli_fetch_array($result)) {
            echo "<option value='$row(0) '$row(0] - $row(1]</option>";
        }
    ?>
    </select> <br> <br>

Tallers: <select name="idtaller" id="taller" >
    </select>
```



Serveis:

Tallers:

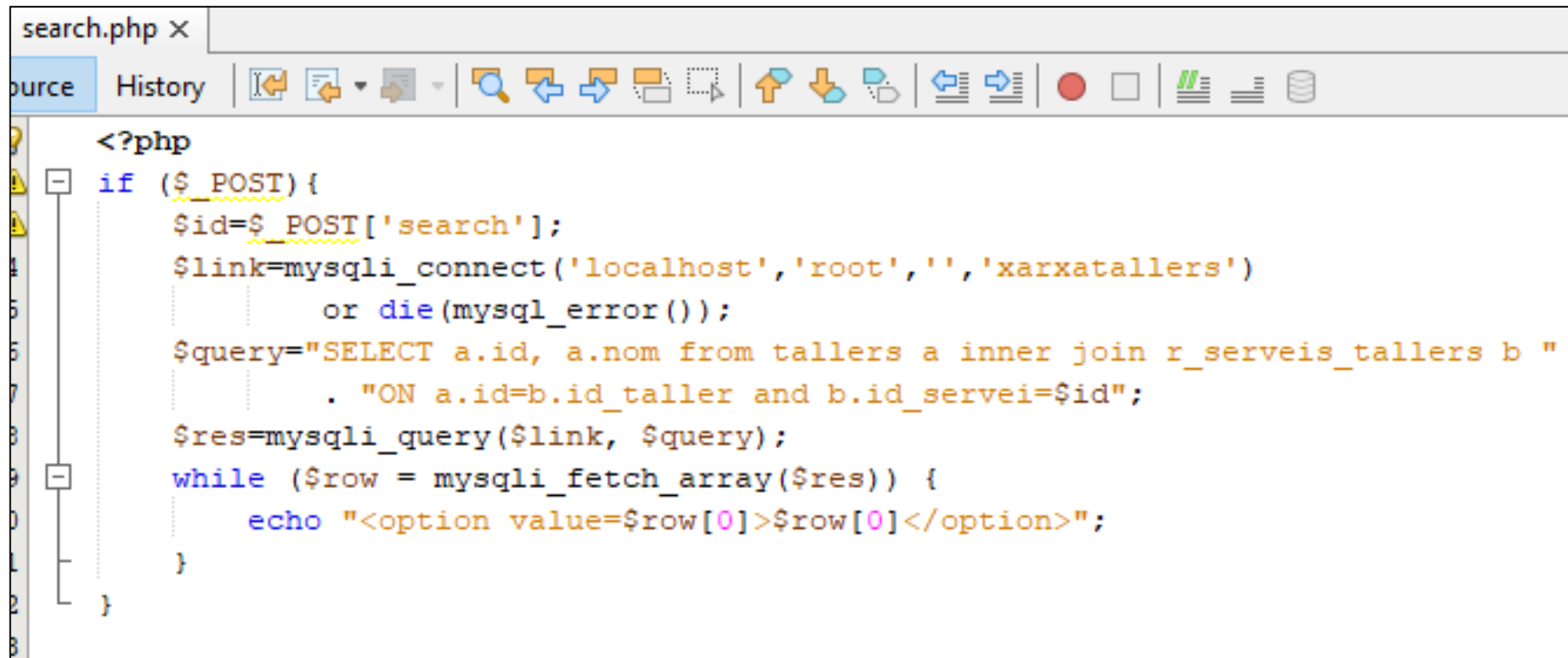
7. COMPLETAR UN COMOBOBOX DESDE OTRO

Paso 2. Definimos el evento change sobre el combobox, de manera que al cambiar su valor, nos salga un alert con el valor seleccionado y se llame al fichero search.php mediante ajax, que nos va a devolver el contenido del combobox dependiente.

```
<script src="https://code.jquery.com/jquery-3.3.1.js"></script>
<script type="text/javascript">
    $(document).ready(function() {
        $('#servei').change(function() {
            var selectedValue = this.value;
            var dataString = 'search='+selectedValue;
            alert( selectedValue);
            $.ajax({
                type: "POST",
                url: 'search.php',
                data: dataString,
                cahe: false,
                success: function(html) {
                    $('#taller').html(html).show();
                }
            });
        });
    });
</script>
```

7. COMPLETAR UN COMOBOBOX DESDE OTRO

Paso 3. El fichero search.php recoge por post el parámetro de selección del combobox independiente, y realiza la búsqueda en función de él. El resultado es el contenido del combobox dependiente.



```
<?php
if ($_POST){
    $id=$_POST['search'];
    $link=mysqli_connect('localhost','root','','xarxatallers')
        or die(mysqli_error());
    $query="SELECT a.id, a.nom from tallers a inner join r_serveis_tallers b "
        . "ON a.id=b.id_taller and b.id_servei=$id";
    $res=mysqli_query($link, $query);
    while ($row = mysqli_fetch_array($res)) {
        echo "<option value=$row[0]>$row[0]</option>";
    }
}
```

7. COMPLETAR UN COMBOBOX DESDE OTRO

Paso 4. Ejecución completa de la secuencia.

Serveis:

Tallers:

- 1 - Canvi de rodes
- 2 - Canvi oli
- 3 - Canvi filtre aire
- 4 - Canvi filtre benzina
- 5 - Canvi de frens
- 6 - Pintura
- 7 - Vidres

Serveis: 2 - Canvi oli

Tallers:

2

Ace

Serveis: 2 - Canvi oli

Tallers:

- Tallers Mubarclaret
- Tallers Mubarclaret
- Tallers Mur
- Tallers Berma
- Tallers Olimpik
- Tallers Jc
- Tallers Jordi's
- Tallers Puertolas
- Tallers Torrents
- Tallers Isidoro

8. AUTOCOMPLETAR UNA CAJA DE TEXTO

AJAX = JAVASCRIPT + PHP

Paso 1. Debajo de la caja de texto que queremos autocompletar añadimos un div con identificador #result, donde dejaremos el resultado del autocompletar.

```
<form action="autocompletar.php" method="post">
  Nombre:  <input type="text" class="search" id="name" name="nombre" placeholder="Busca usuario"/>
           <div id="result"></div>
  Apellido: <input type="text" name="apellido"/><br><br>
  Edad:    <input type="text" name="edad"/><br><br>
  Mail:    <input type="text" name="mail"/><br><br>
  Direccion: <input type="text" name="direccion"/><br><br>
           <input type="submit" value="Insertar" />
</form>
```

8. AUTOCOMPLETAR UNA CAJA DE TEXTO

Paso 2. Definimos el evento keyup sobre el elemento input a autocompletar. Al apretar una tecla sobre el input, se llamará al fichero dp.php mediante ajax, y nos devolverá el resultado de la consulta.

```
<script type="text/javascript" src="js/jquery-1.11.0.min.js"></script>
<script type="text/javascript">
    $(function() {
        $(".search").keyup(function() {
            var searchid = $(this).val();
            var dataString = 'auto=' + searchid;
            if(searchid!='')
            {
                $.ajax({
                    type: "POST",
                    url: "dp.php",
                    data: dataString,
                    cache: false,
                    success: function(html) {
                        $("#result").html(html).show();
                    }
                });
            }
        });
    });
    return false;
});
</script>
```

8. AUTOCOMPLETAR UNA CAJA DE TEXTO

Paso 3. El fichero dp.php recoge por post el parámetro entrado en la caja de texto, y realiza la búsqueda en la base de datos mediante un like.

```
<?php

if ($_POST){
    $q=$_POST['auto'];
    $con = mysqli_connect('localhost','root','','sql_edu')
        or die(mysqli_error());
    $sql = "SELECT nombre FROM usuarios where nombre like '$q%' order by nombre LIMIT 5";
    $res = mysqli_query($con, $sql);
    while($row = mysqli_fetch_array($res)){
        $username=$row[0];
        $b_username='<strong'. $q.'</strong';
        $final_username = str_ireplace($q, $b_username, $username);
        echo "<div class='show' align='left'>";
        echo "<span class='name'>$final_username</span>$nbsp;<br/>";
        echo "</div>";
    }
}
?>
```

8. AUTOCOMPLETAR UNA CAJA DE TEXTO

Paso 4. Muestra de la ejecución:

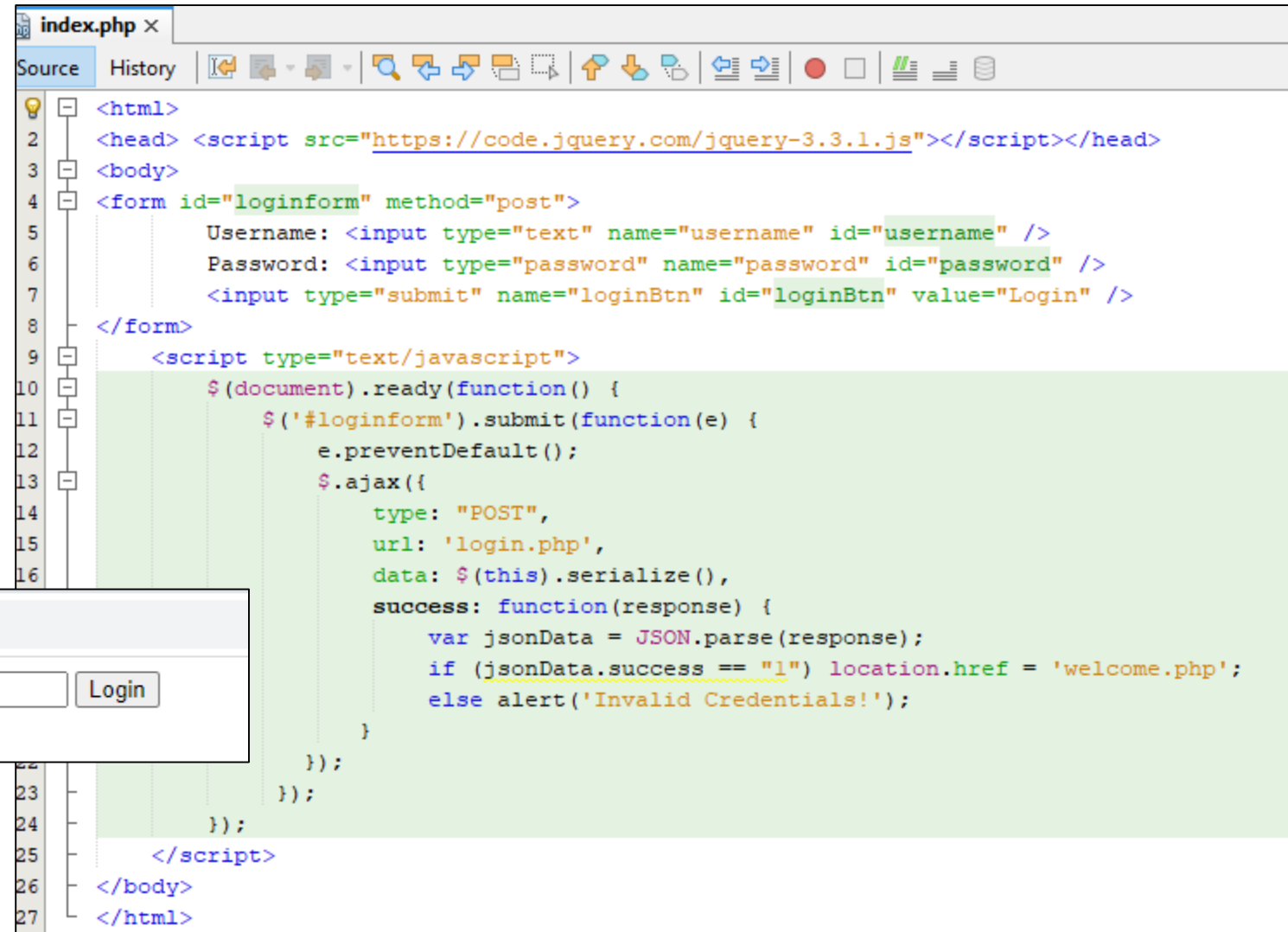
Nombre:
marc
maria
miam
Apellido:
Edad:
Mail:
Direccion:

Nombre:
marc
maria
Apellido:
Edad:
Mail:
Direccion:

Nombre:
edu
esteban
eudaldo
Apellido:
Edad:
Mail:
Direccion:

9. LOGIN CON AJAX

Paso 1. Este es un proceso login realizado con AJAX. index.php es el formulario. El submit es capturado con javascript, y a su vez mediante AJAX se llama a login.php. Este procesa la información y redirige a welcome.php si ha ido bien, o se queda en index.php



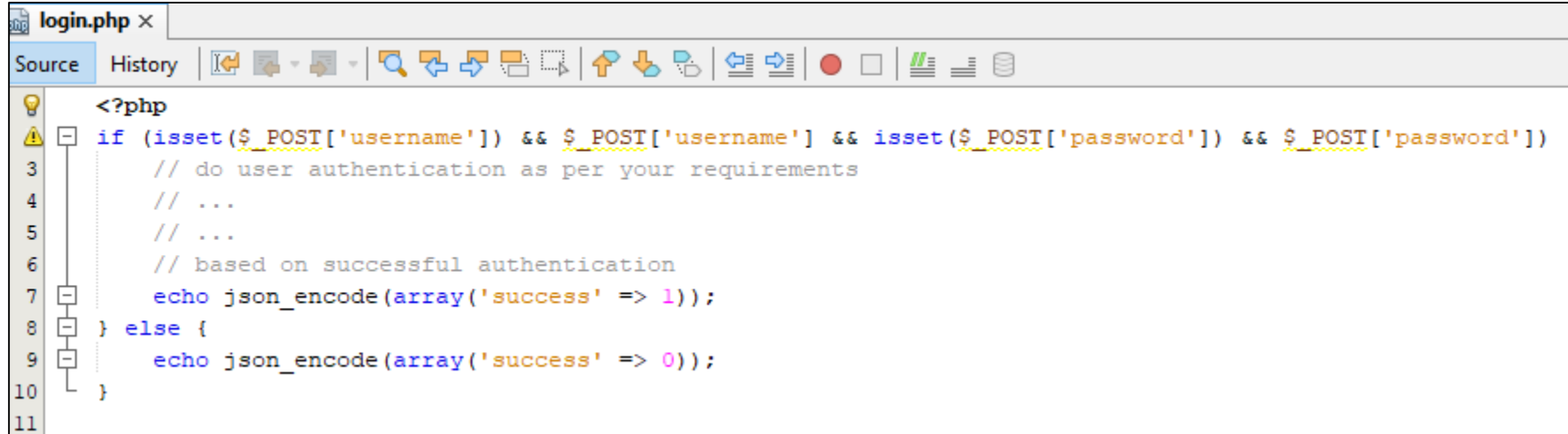
The image shows a web browser window at the top and a code editor window at the bottom. The browser window displays a login form with two input fields labeled 'Username:' and 'Password:', and a 'Login' button. The code editor window shows the source code of the 'index.php' file. The code includes a form with a post method, a jQuery script to capture the submit event, and an AJAX call to 'login.php' with the form data. The success callback checks the response and either redirects to 'welcome.php' or shows an alert for invalid credentials.

```
<html>
<head> <script src="https://code.jquery.com/jquery-3.3.1.js"></script></head>
<body>
<form id="loginform" method="post">
  Username: <input type="text" name="username" id="username" />
  Password: <input type="password" name="password" id="password" />
  <input type="submit" name="loginBtn" id="loginBtn" value="Login" />
</form>
<script type="text/javascript">
$(document).ready(function() {
  $('#loginform').submit(function(e) {
    e.preventDefault();
    $.ajax({
      type: "POST",
      url: 'login.php',
      data: $(this).serialize(),
      success: function(response) {
        var jsonData = JSON.parse(response);
        if (jsonData.success == "1") location.href = 'welcome.php';
        else alert('Invalid Credentials!');
      }
    });
  });
});
</script>
</body>
</html>
```

Este proceso se podría hacer perfectamente sin AJAX

9. LOGIN CON AJAX

Paso 2. Muestra de la ejecución:



```
login.php x
Source History
<?php
1 if (isset($_POST['username']) && $_POST['username'] && isset($_POST['password']) && $_POST['password']) {
2     // do user authentication as per your requirements
3     // ...
4     // ...
5     // based on successful authentication
6     echo json_encode(array('success' => 1));
7 } else {
8     echo json_encode(array('success' => 0));
9 }
10
11
```



← → ↻ ⓘ localhost/php_ajax/welcome.php

BIENVENIDO USUARIO

10. PROYECTO

1. Crea una aplicación que se inicia con un formulario de login (index.php)
2. Realizar la comprobación de login mediante Ajax (login.php)
3. Si nos hemos logado correctamente entramos en la pagina miidealista.php
4. Rellenar mediante base de datos un combobox con las provincias de Cataluña.
(No vale ponerlas tal cual)
5. Si seleccionamos una provincia, mediante Ajax, se deben de rellenar otro combobox con los municipios.
6. Cuando seleccionamos un municipio, se debe de mostrar mediante Ajax, su latitud, longitud y nº de habitantes

1º Opción - LOGIN + IDEALISTA. Rellenar combobox de municipios a partir del combobox de provincias. Al seleccionar combobox municipios mostrar mediante ajax la población de ese municipio en un div → **Son los puntos 9, 4 y 7**

2º Opción - LOGIN + CRUD PHP sobre una tabla de Mysql, por ejemplo clientes
→ **Son los puntos 9, 5 y 6**

10. PROYECTO

CRUD Database

ID	First	Last	Email	Phone	Location	Hobby	Actions	
35	Jane	Smith	js@gmail.com	503-555-5555	Portland, Oregon	Photography	Edit	Del
36	Tristan	Rodriguez	tr@gmail.com	201-555-5555	New York, New York		Edit	Del
37	Andrea	Jones	aj@gmail.com	503-555-5555	Portland, Oregon	Programming	Edit	Del
38	Elliott	Richardson	er@gmail.com		Boise, Idaho		Edit	Del
39	June	Bug	jb@gmail.com			Cooking	Edit	Del

[Download CSV](#)[Add Item](#)