



TECNOLÓGICO
NACIONAL DE MÉXICO®

Tecnológico Nacional de México
Campus Tláhuac



MATERIA

PROGRAMACIÓN PHP CON MVC

NOMBRE DEL TEMA

SISTEMA DE AGENDA CON PHP Y MYSQL

ACTIVIDAD

EXAMEN

ELABORADO POR:

ALONSO MORALES CRISTEL

INGENIERÍA EN SISTEMAS COMPUTACIONALES

DOCENTE

ROLDAN AQUINO SEGURA

FECHA: 17-DIC-2020



Contenido

Introducción	3
Desarrollo	4
Referencias Bibliográficas	17

<https://github.com/CristelAlm/crudconlogin>



Introducción

Vamos a crear paso a paso nuestra primera aplicación web con CodeIgniter MVC y Bootstrap, CodeIgniter es una framework MVC (modelo, vista, controlador) que nos facilita el desarrollo de apps web con php, la parte de las vistas o las pantallas de usuarios estará apoyada por Bootstrap, un framework css, javascript y html para el diseño responsivo y estético. Realizaremos una app capaz de crear reportes y guardarlos en bases de datos para su posterior visualización, también añadiremos algunas funcionalidades extras a nuestro proyecto.



Desarrollo

1) Vamos a la consola y ponemos este código para crear nuestro proyecto

The screenshot shows the 'Composer Installation' page for CodeIgniter 4. The left sidebar contains a navigation menu with the following items: Installation, Manual Installation, Composer Installation, Running Your App, Upgrading From a Previous Version, Troubleshooting, CodeIgniter Repositories, Build Your First Application, CodeIgniter4 Overview, General Topics, Controllers and Routing, Building Responses, Working With Databases, Modeling Data, Managing Databases, Library Reference, Helpers, Testing, Command Line Usage, and Extending CodeIgniter. The main content area is titled 'Installation & Set Up' and includes the following text:

This installation technique would suit a developer who wishes to start a new CodeIgniter4 based project.

Installation & Set Up

In the folder above your project root:

```
composer create-project codeigniter4/appstarter project-root
```

The command above will create a "project-root" folder.

If you omit the "project-root" argument, the command will create an "appstarter" folder, which can be renamed as appropriate.

If you don't need or want phpunit installed, and all of its composer dependencies, then add the "--no-dev" option to the end of the above command line. That will result in only the framework, and the three trusted dependencies that we bundle, being composer-installed.

A sample such installation command, using the default project-root "appstarter":

```
composer create-project codeigniter4/appstarter --no-dev
```

After installation you should follow the steps in the "Upgrading" section.

The screenshot shows a Windows command prompt window with the following text:

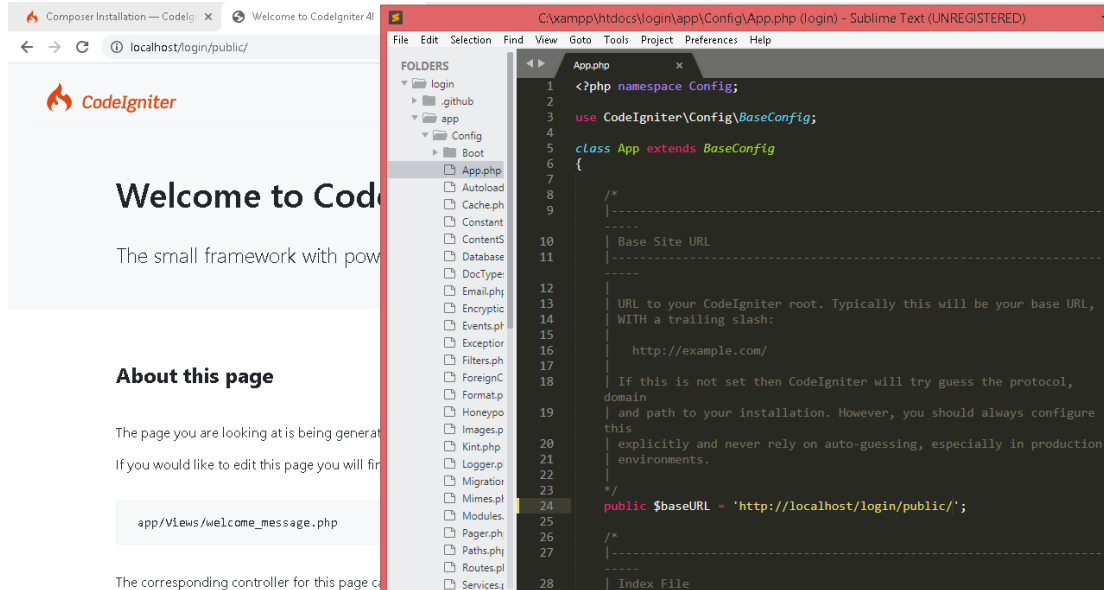
```
C:\WINDOWS\system32\cmd.exe - composer create-project codeigniter4/app...  
Microsoft Windows [Versión 6.3.9600]  
(C) 2013 Microsoft Corporation. Todos los derechos reservados.  
Me C:\Users\Gristel\cd>xampp\htdocs  
C:\xampp\htdocs>composer create-project codeigniter4/appstarter login  
Creating a "codeigniter4/appstarter" project at "/login"  
Warning from https://repo.packagist.org: You are using an outdated version of Co  
poser. Composer 2.0 is now available and you should upgrade. See https://getco  
poser.org/2
```

Overlaid on the command prompt is a Notepad window titled 'Sin título: Bloc de notas' with the following text:

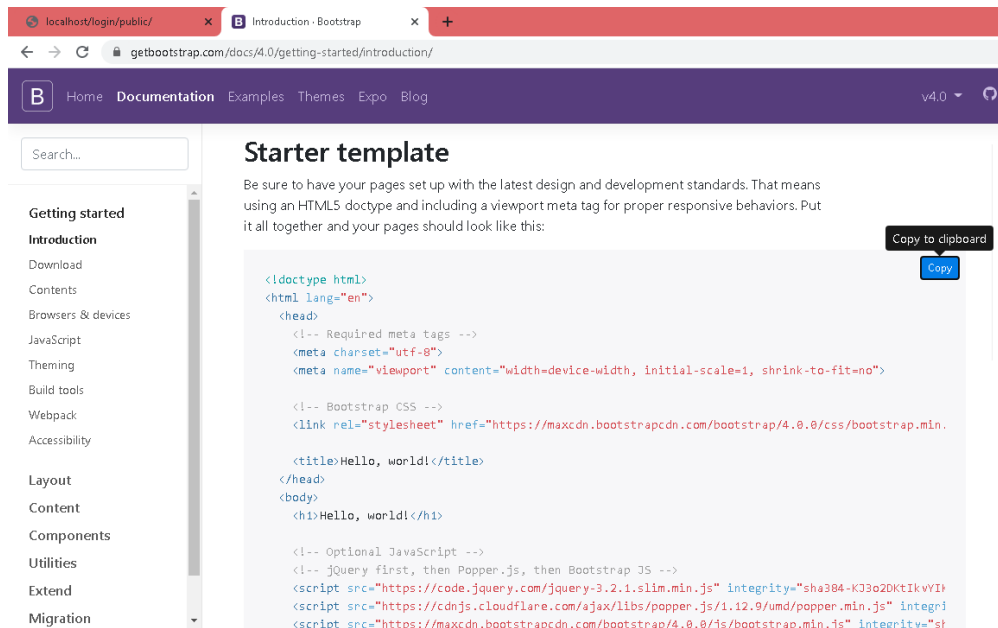
```
composer create-project codeigniter4/appstarter  
login
```



- 2) Vamos a ir a nuestra carpeta App-Config-App y copiamos el url para que funcione

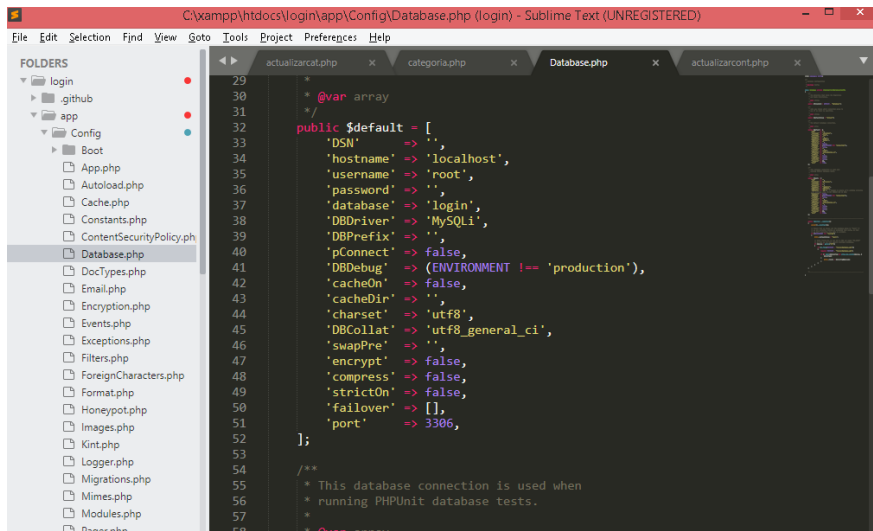


- 3) Vamos a Bootstrap y copiamos Starter Template





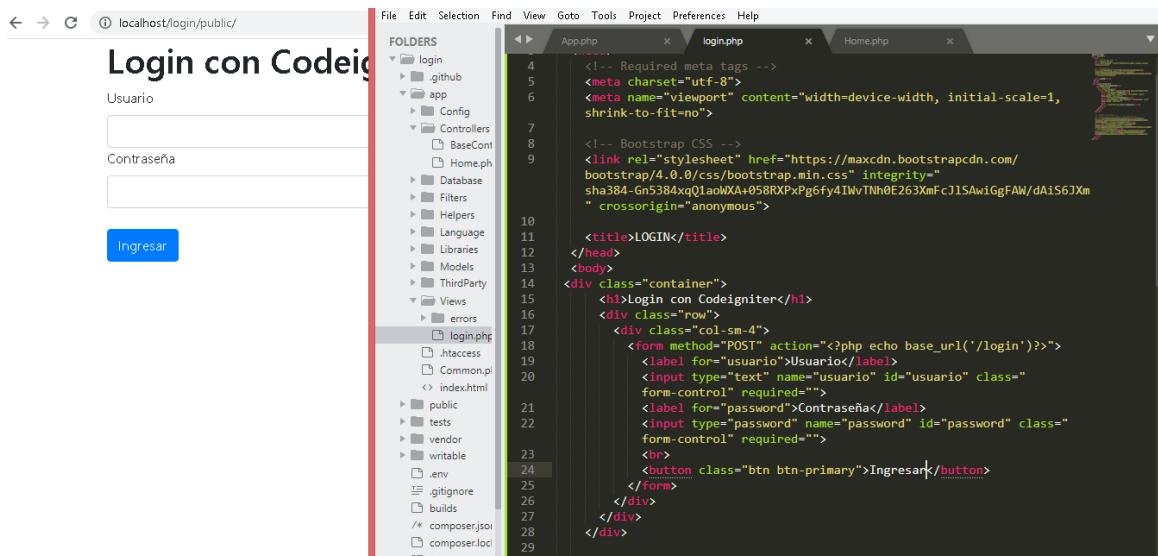
4) Configuramos nuestra base de datos a nuestro proyecto login



The screenshot shows the Sublime Text editor with the file 'Database.php' open. The file contains a PHP array configuration for a database connection. The configuration includes fields for DSN, hostname, username, password, database, DBDriver, DBPrefix, pConnect, DBDebug, cacheOn, cacheDir, charset, DBCollat, swapPre, encrypt, compress, strictOn, failover, and port. The database is configured to use MySQLi on localhost with the 'login' database.

```
29 *  
30 * @var array  
31 */  
32 public $default = [  
33     'DSN' => '',  
34     'hostname' => 'localhost',  
35     'username' => 'root',  
36     'password' => '',  
37     'database' => 'login',  
38     'DBDriver' => 'MySQLi',  
39     'DBPrefix' => '',  
40     'pConnect' => false,  
41     'DBDebug' => (ENVIRONMENT !== 'production'),  
42     'cacheOn' => false,  
43     'cacheDir' => '',  
44     'charset' => 'utf8',  
45     'DBCollat' => 'utf8_general_ci',  
46     'swapPre' => '',  
47     'encrypt' => false,  
48     'compress' => false,  
49     'strictOn' => false,  
50     'failover' => [],  
51     'port' => 3306,  
52 ];  
53  
54 /**  
55  * This database connection is used when  
56  * running PHPUnit database tests.  
57  */  
58
```

5) Hacemos la vista del login



The screenshot shows a web browser on the left displaying a login form titled 'Login con Codeigniter'. The form has fields for 'Usuario' and 'Contraseña' and an 'Ingresar' button. On the right, the Sublime Text editor shows the 'login.php' file, which contains HTML and PHP code for the login view. The code includes meta tags, a Bootstrap CSS link, a title, and a form with input fields for username and password, and a submit button.

```
4 <!-- Required meta tags -->  
5 <meta charset="utf-8">  
6 <meta name="viewport" content="width=device-width, initial-scale=1,  
7 shrink-to-fit=no">  
8  
9 <!-- Bootstrap CSS -->  
10 <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/  
11 bootstrap/4.0.0/css/bootstrap.min.css" integrity="  
12 sha384-Gn5384xqQ1aoXXA+058RXPxPg6fy4IWvTNh0E263XmFcJ1SAw1GgFAM/dA1S6JXm  
13 " crossorigin="anonymous">  
14  
15 <title>LOGIN</title>  
16 </head>  
17 <body>  
18 <div class="container">  
19 <h1>Login con Codeigniter</h1>  
20 <div class="row">  
21 <div class="col-sm-4">  
22 <form method="POST" action="{php echo base_url('/login')}?">  
23 <label for="usuario">Usuario</label>  
24 <input type="text" name="usuario" id="usuario" class="form-control" required="">  
25 <label for="password">Contraseña</label>  
26 <input type="password" name="password" id="password" class="form-control" required="">  
27 <br>  
28 <button class="btn btn-primary">Ingresar</button>  
29 </form>  
30 </div>  
31 </div>  
32 </body>  
33 </html>
```



6) Hacemos la vista de Inicio con menú

```
11 <title>Inicio</title>
12 </head>
13 <body>
14 <nav class="navbar navbar-expand-lg navbar-light bg-light">
15 <a class="navbar-brand" href="#" style="color:purple; font-size:10em;">
16 <?php echo session('usuario'); ?></a>
17 <button class="navbar-toggler" type="button" data-toggle="collapse"
18 data-target="#navbarSupportedContent" aria-controls="
19 navbarSupportedContent" aria-expanded="false" aria-label="Toggle
20 navigation">
21 <span class="navbar-toggler-icon"></span>
22 </button>
23
24 <a class="nav-link" href="<?php echo base_url('/categoria') ?>"
25 style="color:purple;">Categorías</a>
26 </li>
27 </li>
28
29 <a class="nav-link" href="<?php echo base_url('/contactos') ?>"
30 style="color:purple;">Contactos</a>
31 </li>
32 </li>
33
34 <div class="collapse navbar-collapse" id="navbarSupportedContent">
35 <ul class="navbar-nav mr-auto">
36 <li class="nav-item active">
37 <a class="nav-link" href="<?php echo base_url('/salir') ?>" style=
38 "color:purple;">Salir <span class="sr-only">(current)</span></a>
39 </li>
40 </ul>
41 </div>
42 </body>
43 </html>
```

7) Hacemos la vista contacto

```
login\app\Views\contactos.php (login) - Sublime Text (UNREGISTERED)
Project: Preferences Help
actualizarcat.php x categoria.php x contactos.php x inicio.php x actualizarcont.php x
<div class="table-responsive">
<table class="table table-hover" id="tablaload">
<thead>
<tr>
<td align="center">Nombre</td>
<td align="center">Apellido Paterno</td>
<td align="center">Apellido Materno</td>
<td align="center">Telefono</td>
<td align="center">E-mail</td>
<td align="center">Categorías</td>
<td align="center">Editar</td>
<td align="center">Eliminar</td>
</tr>
</thead>
<tbody>
<?php foreach($datos as $key): ?>
<tr>
<td align="center"><?php echo $key->nombre ?></td>
<td align="center"><?php echo $key->paterno ?></td>
<td align="center"><?php echo $key->materno ?></td>
<td align="center"><?php echo $key->telefono ?></td>
<td align="center"><?php echo $key->email ?></td>
<td align="center"><?php echo $key->id_categoria ?></td>
<td align="center">
<div style="text-align: center">
<a href="<?php echo base_url().'/obtenerCont/'. $key->
id_contacto?>" class="btn btn-warning btn-sm">
Editar</a>
</div>
<div style="text-align: center">
<a href="<?php echo base_url().'/eliminarcont/'. $key->
id_contacto?>" class="btn btn-danger btn-sm">
Eliminar</a>
</div>
</td>
</tr>
</tbody>
</table>
</div>
```



8) Vista categoría

```
xampp\htdocs\login\app\views\categoria.php (login) - Sublime Text (UNREGISTERED)
to Tools Project Preferences Help
49 <button class="btn btn-danger">
50 <i class="fas fa-id-card-alt fa-6x"></i>
51 </button>
52 <h2>Categorías</h2>
53 <button class="btn btn-primary" data-toggle="modal" data-target="#modalInsertar">
54 <span class="fas fa-clipboard-check"></span> Agregar nuevo
55 </button>
56 <hr>
57 <div class="table-responsive">
58 <table class="table table-hover" id="tablaLoad">
59 <thead>
60 <tr>
61 <td align="center">Categorías</td>
62 <td align="center">Descripción</td>
63 <td align="center">Editar</td>
64 <td align="center">Eliminar</td>
65 </tr>
66 </thead>
67 <tbody>
68 <?php foreach($datos as $key): ?>
69 <tr>
70
71 <td align="center"><?php echo $key->categoria ?></td>
72 <td align="center"><?php echo $key->descripcion ?></td>
73 <td style="text-align: center">
74 <a href="<?php echo base_url().'/obtenerCategoria/'.$key->id_categoria>" class="btn btn-warning btn-sm">
75 <i class="fas fa-edit"></i> Editar</a>
76 </td>
77 <td style="text-align: center">
78 <a href="<?php echo base_url().'/eliminar/'.$key->id_categoria>" class="btn btn-danger btn-sm">
79 <i class="fa fa-trash"></i> Eliminar</a>
```

9) Creamos la migración de la tabla t_categoria

```
MINGW64:/c:/xampp/htdocs/login
Cristel@PC1 MINGW64 /c:/xampp/htdocs/login (master)
$ php spark migrate:create t_categoria

CodeIgniter CLI Tool - Version 4.0.4 - Server-Time: 2020-12-17 11:15:17am
Created file: App/Database/Migrations/2020-12-17-171517_t_categoria.php
```

10) Ejecutamos la migración

```
Cristel@PC1 MINGW64 /c:/xampp/htdocs/login (master)
$ php spark migrate

CodeIgniter CLI Tool - Version 4.0.4 - Server-Time: 2020-12-17 11:15:45am
Running all new migrations...
Running: (App) 2020-12-17-171517_App\Database\Migrations\TCategoria
Done
```




11) Agregamos los valores

```
atabase\Migrations\2020-12-17-171517_t_categoria.php (login) - Sublime Text (UNREGISTERED)
Tools Project Preferences Help
actualizarcat.php x categoria.php x Database.php x 2020-12-17-171517_t_categoria.php x mt.php x
1 k?php namespace App\Database\Migrations;
2
3 use CodeIgniter\Database\Migration;
4
5 class Tcategoria extends Migration
6 {
7     public function up()
8     {
9         $this->forge->addField([
10             'id_categoria' => [
11                 'type' => 'INT',
12                 'constraint' => 5,
13                 'unsigned' => true,
14                 'auto_increment' => true,
15             ],
16             'categoria' => [
17                 'type' => 'VARCHAR',
18                 'constraint' => '255',
19             ],
20             'descripcion' => [
21                 'type' => 'VARCHAR',
22                 'constraint' => '255',
23             ],
24         ]);
25         $this->forge->addKey('id_categoria', true);
26         $this->forge->createTable('t_categoria');
27     }
28
29     public function down()
30     {
31         $this->forge->dropTable('t_categoria');
32     }
33 }
```

12) Creamos la migración de t_agenda

```
Cristel@PC1 MINGW64 /c:/xampp/htdocs/login (master)
$ php spark migrate:create t_agenda

CodeIgniter CLI Tool - Version 4.0.4 - Server-Time: 2020-12-17 11:16:15am
Created file: App\Database\Migrations\2020-12-17-171615_t_agenda.php
```

13) Ejecutamos la migración

```
Cristel@PC1 MINGW64 /c:/xampp/htdocs/login (master)
$ php spark migrate

CodeIgniter CLI Tool - Version 4.0.4 - Server-Time: 2020-12-17 11:18:12am
Running all new migrations...
Running: (App) 2020-12-17-171615_App\Database\Migrations\TAagenda
Done
```



14) Agregamos los valores

```
p:\Database\Migrations\2020-12-17-171615_t_agenda.php (login) - Sublime Text (UNREGISTERED)
Tools Project Preferences Help
actualizarcat.php x categoria.php x Database.php x 2020-12-17-171615_t_agenda.php x font.php x
9
10 $this->forge->addField([
11     'id_contacto' => [
12         'type' => 'INT',
13         'constraint' => 5,
14         'unsigned' => true,
15         'auto_increment' => true,
16     ],
17     'nombre' => [
18         'type' => 'VARCHAR',
19         'constraint' => '255',
20     ],
21     'paterno' => [
22         'type' => 'VARCHAR',
23         'constraint' => '255',
24     ],
25     'materno' => [
26         'type' => 'VARCHAR',
27         'constraint' => '255',
28     ],
29     'telefono' => [
30         'type' => 'VARCHAR',
31         'constraint' => '255',
32     ],
33     'email' => [
34         'type' => 'VARCHAR',
35         'constraint' => '255',
36     ],
37     'id_categoria' => [
38         'type' => 'INT',
39         'constraint' => 5,
40         'unsigned' => true,
41     ],
42 ];
```

15) Agregamos el Modelo de Usuario

```
pp\htdocs\login\app\Models\Usuarios.php (login) - Sublime Text (UNREGISTERED)
Tools Project Preferences Help
actualizarcat.php x categoria.php x Database.php x Usuarios.php x actualizarcont.php x
1 <?php namespace App\Models;
2
3 use CodeIgniter\Model;
4
5 class Usuarios extends Model{
6
7     public function obtenerUsuario($data){
8         $Usuario = $this->db->table('t_usuarios');
9         $Usuario->where($data);
10        return $Usuario->get()->getResultArray();
11    }
12
13 }
```



16) Agregamos el modelo Contacto

```
1 <?php namespace App\Models;
2
3 use CodeIgniter\Model;
4
5 class contactosModel extends Model{
6     public function listarContactos(){
7         $Contactos = $this->db->query('SELECT t_agenda.id_contacto,
            t_agenda.nombre, t_agenda.paterno, t_agenda.materno,
            t_agenda.telefono, t_agenda.email, t_agenda.id_categoria,
            t_categoria.categoria FROM t_agenda, t_categoria where
            t_categoria.id_categoria=t_agenda.id_categoria');
8         return $Contactos->getResult();
9     }
10    public function insertarContacto($datos){
11        $Contactos = $this->db->table('t_agenda');
12        $Contactos->insert($datos);
13
14        return $this->db->insertID();
15    }
16
17    public function obtenerContactos($data){
18        $Contactos = $this->db->table('t_agenda');
19        $Contactos->where($data);
20        return $Contactos->get()->getResultArray();
21    }
22
23    public function actualizarContacto($data, $idContacto){
24        $Contactos = $this->db->table('t_agenda');
25        $Contactos->set($data);
26        $Contactos->where('id_contacto',$idContacto);
27        return $Contactos->update();
28    }
```

17) Agregamos el modelo Categorías

```
1 <?php namespace App\Models;
2
3 use CodeIgniter\Model;
4
5 class categoriasModel extends Model{
6     public function listarCategoria(){
7         $Categorias = $this->db->query("SELECT * FROM t_categoria");
8         return $Categorias->getResult();
9     }
10    public function insertar($datos){
11        $Categorias = $this->db->table('t_categoria');
12        $Categorias->insert($datos);
13
14        return $this->db->insertID();
15    }
16
17    public function obtenerCategoria($data){
18        $Categorias = $this->db->table('t_categoria');
19        $Categorias->where($data);
20        return $Categorias->get()->getResultArray();
21    }
22
23    public function actualizar($data, $idCategoria){
24        $Categorias = $this->db->table('t_categoria');
25        $Categorias->set($data);
26        $Categorias->where('id_categoria',$idCategoria);
27        return $Categorias->update();
28    }
29
30    public function eliminar($data){
31        $Categorias = $this->db->table('t_categoria');
32        $Categorias->where($data);
33        return $Categorias->delete();
34    }
```



18) Agregamos los modelos al controlador

```
Home.php
1 <?php namespace App\Controllers;
2 use App\Models\categoriasModel;
3 use App\Models\contactosModel;
4 use App\Models\Usuarios;
5 class Home extends BaseController
6 {
7     public function index()
8     {
9         $mensaje = session('mensaje');
10        return view('login', ["mensaje" => $mensaje]);
11    }
12
13    public function inicio(){
14        return view('inicio');
15    }
16
17    public function login(){
18        $usuario = $this->request->getPost('usuario');
19        $password = $this->request->getPost('password');
20        $Usuario = new Usuarios();
21
22        $datosUsuario = $Usuario->obtenerUsuario(['usuario' => $usuario]);
23
24        if (count($datosUsuario) > 0 && password_verify($password, $
25            datosUsuario[0]['password'])) {
26
27            $data = [
28                "usuario" => $datosUsuario[0]['usuario'],
29                "type" => $datosUsuario[0]['type']
30            ];
31
32            $session = session();
33            $session->set($data);
34        }
35    }
36 }
```

19) Las clases de categoría

```
Home.php
47
48 public function crear(){
49     $datos = [
50         "categoria" => $_POST['categoria'],
51         "descripcion" => $_POST['descripcion']
52     ];
53
54     $Crud = new categoriasModel();
55     $respuesta = $Crud->insertar($datos);
56
57     if ($respuesta > 0){
58         return redirect()->to(base_url().'/categoria')->with('mensaje',
59             '1');
60     }else{
61         return redirect()->to(base_url().'/categoria')->with('mensaje',
62             '0');
63     }
64 }
65
66 public function categoria(){
67     $Crud = new categoriasModel();
68     $datos = $Crud->listarCategoria();
69
70     $mensaje = session('mensaje');
71
72     $data = [
73         "datos" => $datos,
74         "mensaje" => $mensaje
75     ];
76     return view('categoria', $data);
77 }
```



20) Las vistas de login

The screenshot shows a web browser window with the URL `localhost/login/public/`. The page has a title "Iniciar Sesión". Below the title, there are two input fields: "Usuario" with the value "admin" and "Contraseña" with three dots indicating a password. A blue button labeled "Ingresar" is positioned below the password field. The browser's tab bar shows several open tabs, including "WhatsApp", "LOGIN", "Address Book Icon - Regular | F...", "CRUD codeigniter", and "Primera Apl".

21) Vista de inicio

The screenshot shows a web browser window with the URL `localhost/login/public/inicio`. The page has a navigation bar with links: "admin" (active), "Categorías", "Contactos", and "Salir". The main content area features a large profile picture of a woman. Below the picture, the text "Datos del Admin" is displayed. Underneath, there are two input fields: "Nombre" with the value "Cristel Alonso Morales" and "Edad" with the value "25 Años".



22) Vista de contactos

← → ↻ localhost/login/public/contactos

admin Categorías Contactos Salir

Agenda

Agregar nuevo

Nombre	Apellido Paterno	Apellido Materno	Telefono	E-mail	Categoria	Editar	Eliminar
Teresa	Gutierrez	Vazquez	58425364	ter@gmail.com	2	Editar	Eliminar
Edder	Tapia	Gomez	5583927350	tageunk@gmail.com	3	Editar	Eliminar

23) Editamos un contacto

← → ↻ localhost/login/public/obtenerCont/2

admin Categorías Contactos Salir

Actualizar contacto

Nombre
Teresa

Apellido Paterno
Gutierrez

Apellido Materno
Vazquez

Telefono
58425364

E-mail
ter@gmail.com

Categoria
amigos

Guardar



24) Se actualizo correctamente

← → ↻ localhost/login/public/contatos

admin Categorías Contatos Salir

Agenda

Agregar nuevo

Nombre	Apellido Paterno	Apellido Materno	Categoría	Editar	Eliminar
cristel	Gutierrez	Vaz	2	Editar	Eliminar
Edder	Tapia	Go	3	Editar	Eliminar

OK

25) Vista de categoría

← → ↻ localhost/login/public/categoria

admin Categorías Contatos Salir

Categorías

Agregar nuevo

Categoría	Descripción	Editar	Eliminar
amigos	Secundaria	Editar	Eliminar
familia	Alonso	Editar	Eliminar



26) Actualizamos una categoría

← → ↻ localhost/login/public/obtenerCategoria/2

admin Categorías Contactos Salir

Actualizar categorías

Categoría

Descripción

Guardar

27) Se actualizo correctamente

← → ↻ localhost/login/public/categoria ☆

admin Categorías Contactos Salir

Categorías

Agregar nuevo

Categoría	Descripción	Editar	Eliminar
amigos	Preparatoria	Editar	Eliminar
familia	Alonso	Editar	Eliminar



TECNOLÓGICO
NACIONAL DE MÉXICO®

Tecnológico Nacional de México
Campus Tláhuac



Referencias Bibliográficas

Alonso Morales Cristel