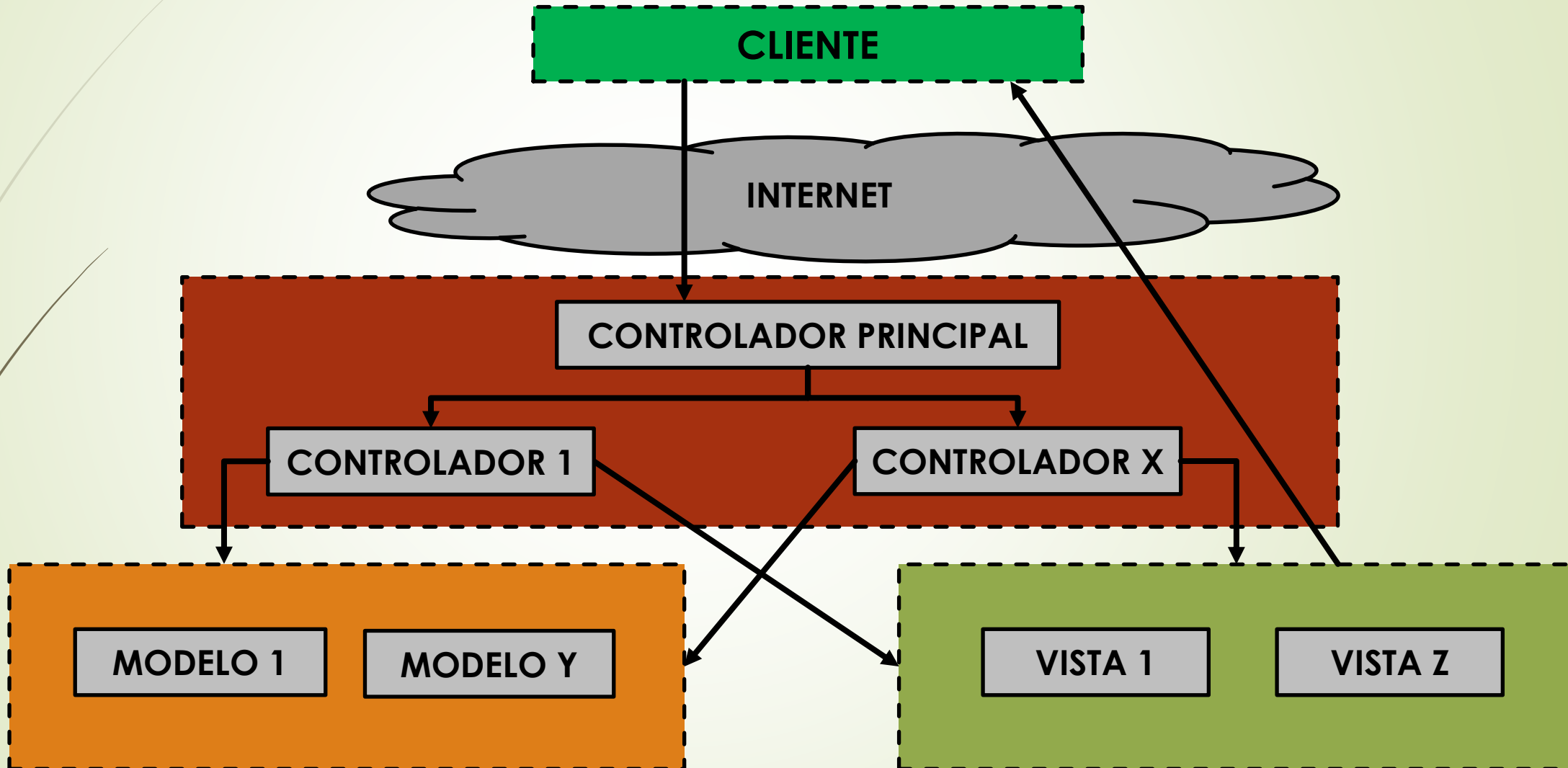
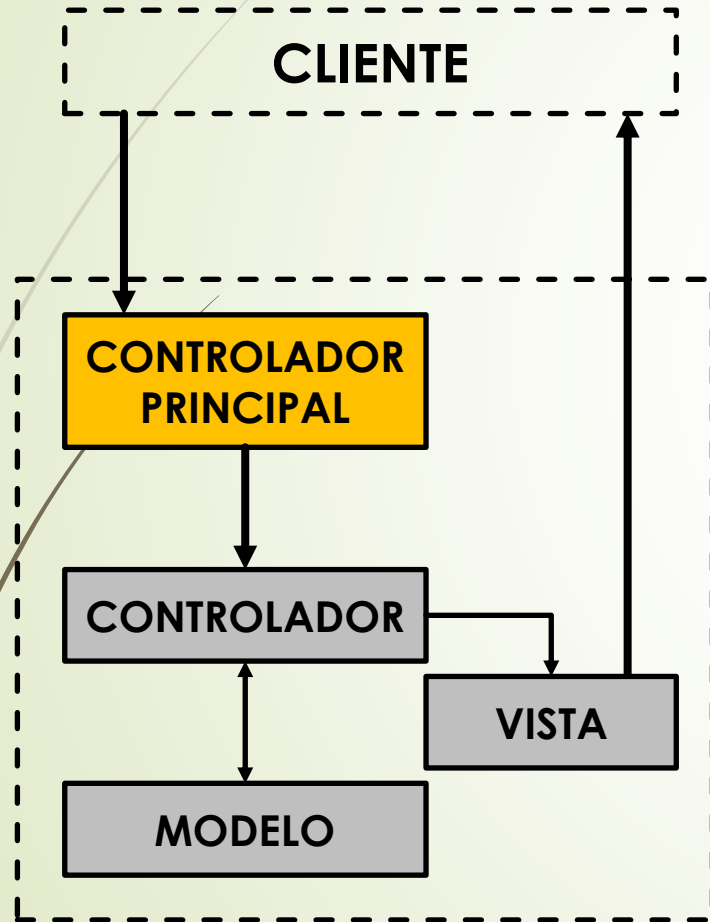


PATRON MVC CON PROGRAMACIÓN ORIENTADA A OBJETOS



CONTROLADOR PRINCIPAL



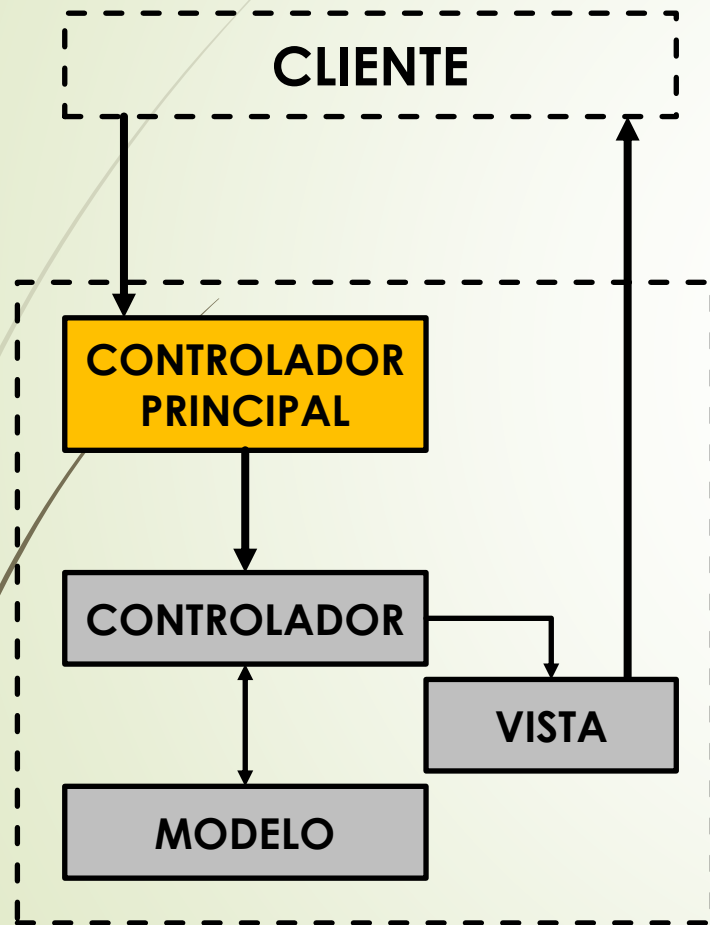
C:\Código MVC\index.php - Notepad++

File Edit Search View Encoding Language Settings Tools Macro Run

index.php

```
1 <?php
2 /*
3  Camino único para acceder a la aplicación.
4  Sólo incluye e inicia el FrontController.
5  */
6
7  //Incluimos el FrontController
8  require 'libs/FrontController.php';
9  //Lo iniciamos con su método estático main.
10 FrontController::main();
11 ?>
```

CONTROLADOR PRINCIPAL



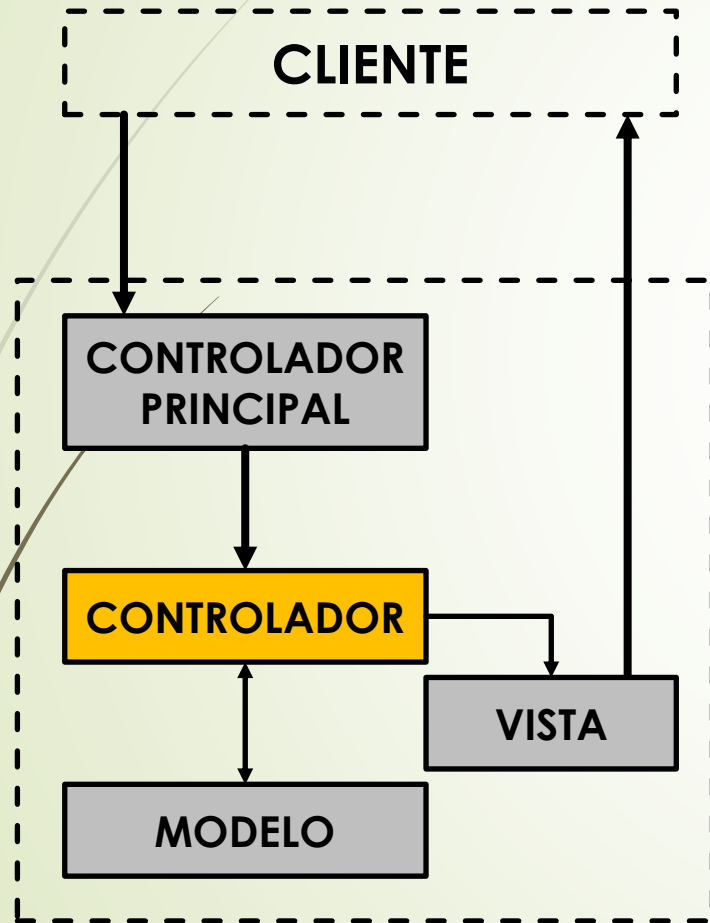
C:\Código MVC\libs\FrontController.php - Notepad++

File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?

index.php FrontController.php

```
1 <?php
2 /*
3  Se encarga de recibir todas las peticiones e incorporar algunos ficheros.
4  Después realiza una selección del controlador encargado de atender la petición
5  y produce el evento correspondiente
6  */
7
8 class FrontController {
9
10     static function main() {
11
12         //Incluimos algunas clases:
13
14         require 'libs/Config.php'; //de configuracion
15         require 'libs/SPDO.php'; //PDO con singleton
16         require 'libs/View.php'; //Mini motor de plantillas
17
18         require 'config.php'; //Archivo con configuraciones.
19
20         //Con el objetivo de no repetir nombre de clases, nuestros controladores
21         //terminarán todos en Controller. Por ej, la clase controladora Items, será ItemsController
22
23         //Formamos el nombre del Controlador o en su defecto, tomamos que es el IndexController
24         if(! empty($_GET['controlador'])) {
25             $controllerName = $_GET['controlador'] . 'Controller';
26         } else {
27             $controllerName = "CdController";
28         }
29
30         //Lo mismo sucede con las acciones, si no hay acción, tomamos index como acción
31         if(! empty($_GET['accion'])) {
32             $actionName = $_GET['accion'];
33         } else {
34             $actionName = "listar";
35         }
36         $controllerPath = $config->get('directorioControlador') . $controllerName . '.php';
```

CONTROLADOR



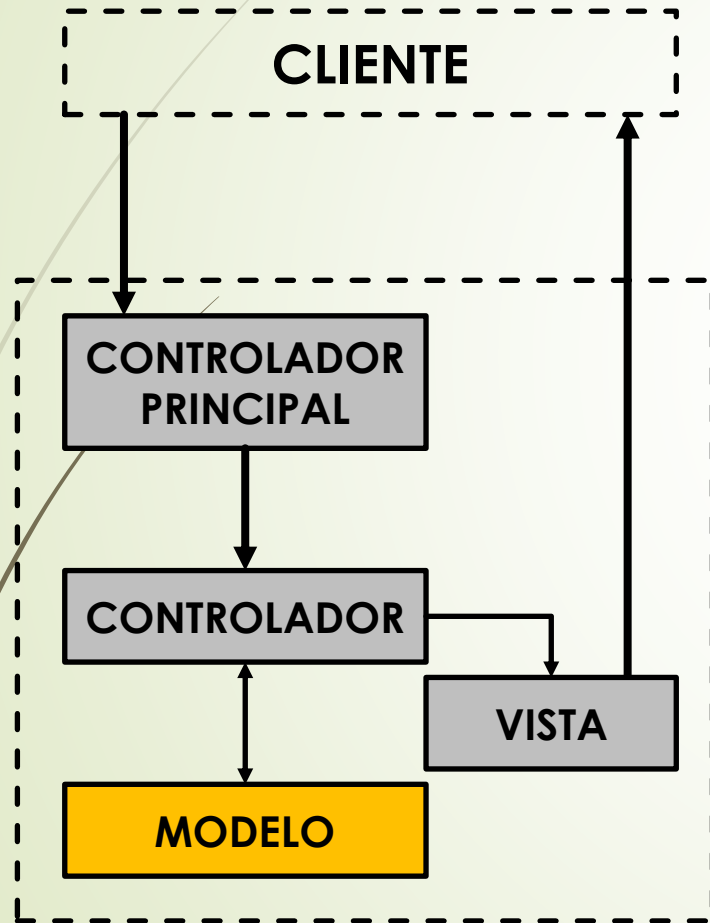
C:\Código MVC\controlador\CdController.php - Notepad++

File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?

index.php FrontController.php CdController.php

```
1 <?php
2 class CdController {
3     private $view;
4
5     function __construct() {
6         //Creamos una instancia de nuestro mini motor de plantillas
7         $this->view = new View();
8     }
9
10    public function listar() {
11        //Incluye el modelo que corresponde
12        require 'modelo/CdModelo.php';
13        //Creamos una instancia de nuestro "modelo"
14        $items = new CdModelo();
15        //Le pedimos al modelo todos los items
16        $listado = $items->devuelveTodosCds();
17        $parametrosParaVista = array();
18        //Pasamos a la vista toda la informaciOn que se desea representar
19        $parametrosParaVista['listado'] = $listado;
20        //Finalmente presentamos nuestra plantilla
21        $this->view->show("listado.php", $parametrosParaVista);
22    }
23
24    public function agregar() {
25        echo 'Aquí incluiremos nuestro formulario para insertar items';
26    }
27
28    public function eliminar() {
29        $idCd = $_REQUEST['id'];
30        require 'modelo/CdModelo.php';
31        $modelo = new CdModelo();
32        $cabecera = generaCabecera($usuario);
33        $pie = generaPie(array('alta', 'listado'));
34        $contenido = generoContenido();
35        $platillaPagina->show(array('estilo.css', 'fondos.css'),
36                                array())
```

MODELO



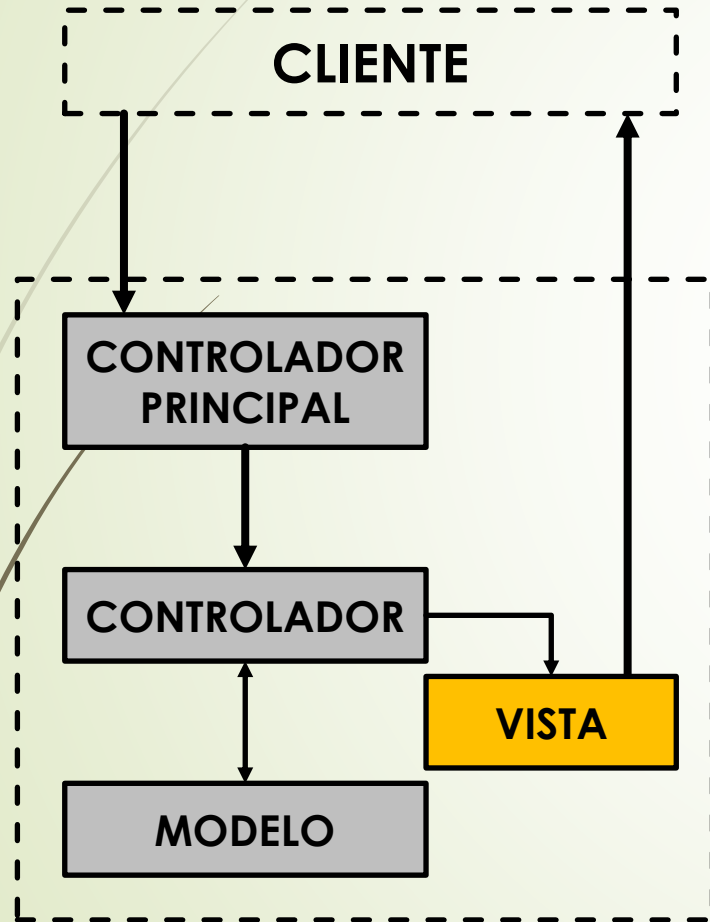
C:\Código MVC\modelo\CdModelo.php - Notepad++

File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?

index.php FrontController.php CdController.php CdModelo.php

```
1 <?php
2 /*
3  Clase modelo. Usa SPDO.
4  */
5
6 class CdModelo {
7     protected $db;
8
9     public function __construct() {
10         //Traemos la Única instancia de PDO
11         $this->db = SPDO::singleton();
12     }
13
14     public function devuelveTodosCds() {
15         //realizamos la consulta de todos los items
16         $consulta = $this->db->prepare('SELECT * FROM cds');
17         $consulta->execute();
18         //devolvemos la colección para que la vista la presente.
19         return $consulta;
20     }
21
22     public function devuelveCdPorId($id) {
23         $elCd = null;
24         // ...
25         // recuperar el Cd de la BD y crear un objeto Cd
26         // $elCd = new Cd(X,Y,Z,W);
27         // ...
28         return $elCd;
29     }
30 }
31 ?>
```

VISTA



C:\Código MVC\vista\listado.php - Notepad++

File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?

index.php FrontController.php CdController.php CdModelo.php listado.php

```
1 <!DOCTYPE html>
2
3 <html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
4
5 <head>
6     <meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
7     <title>MVC - Modelo, Vista, Controlador - Jourmoly</title>
8 </head>
9
10 <body>
11 <table>
12 <tr>
13     <th>ID</th>
14     <th>Título</th>
15     <th>Intérprete</th>
16     <th>Año</th>
17 </tr>
18 <?php
19     // $listado es una variable asignada desde el controlador ItemsController.
20     while($item = $listado->fetch()) {
21     <?>
22 <tr>
23     <td><?php echo $item['id']?></td>
24     <td><?php echo $item['titel']?></td>
25     <td><?php echo $item['interpret']?></td>
26     <td><?php echo $item['jahr']?></td>
27 </tr>
28 <?php
29     }
30 <?>
31 </table>
32 </body>
33 </html>
```




EJERCICIO 1

- Plantea un diseño basado en MVC sobre el ejercicio de la protectora de animales que realizamos en la unidad anterior.
- Implementa el diseño propuesto.