

Introducción.

El propósito de este manual del programador es dar a conocer al lector los códigos fuentes del programa realizado. Para ello tratamos de la forma más concisa de explicar cada uno de los códigos, junto con la programación utilizada en el desarrollo del sistema, esto con el fin de que el usuario pueda modificar a gusto alguno de los valores y parámetros de las funciones que se encuentran expuestas en la programación de Sistema Informático de Registro de Notas Académico.

Objetivos General.

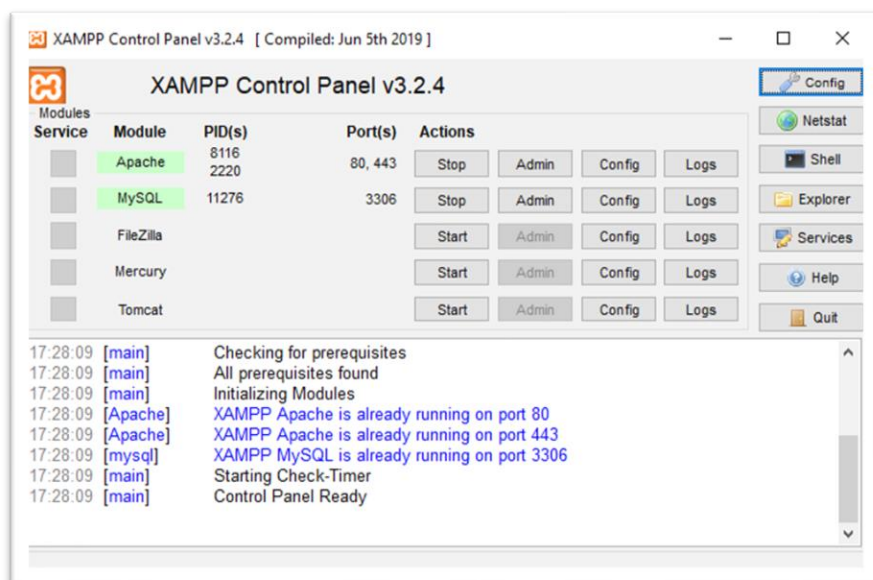
- El desarrollo del manual del programador es dar a conocer a los usuarios los códigos fuentes del sistema y registro de notas, mostrar los pasos a seguir para la configuración del sistema.

Objetivos Específicos.

- El desarrollo del manual del programador instruirá en el uso adecuado del sistema de notas, para el acceso oportuno y adecuado del sistema de notas mismo.
- El desarrollo del manual del programador mostrará los pasos a seguir en el proceso de configuración del sistema, así como la descripción de los archivos relevantes del sistema.
- El desarrollo del manual del programador orientará en la configuración y soporte del Sistema Informático de Registro de Notas Académico. mismo.

Manual de administrador

1. El Administrador deberá ejecutar el gestor base de datos XAMPP activando Apache y MySQL, para poder abrir el sistema en el navegador como la Base de Datos.

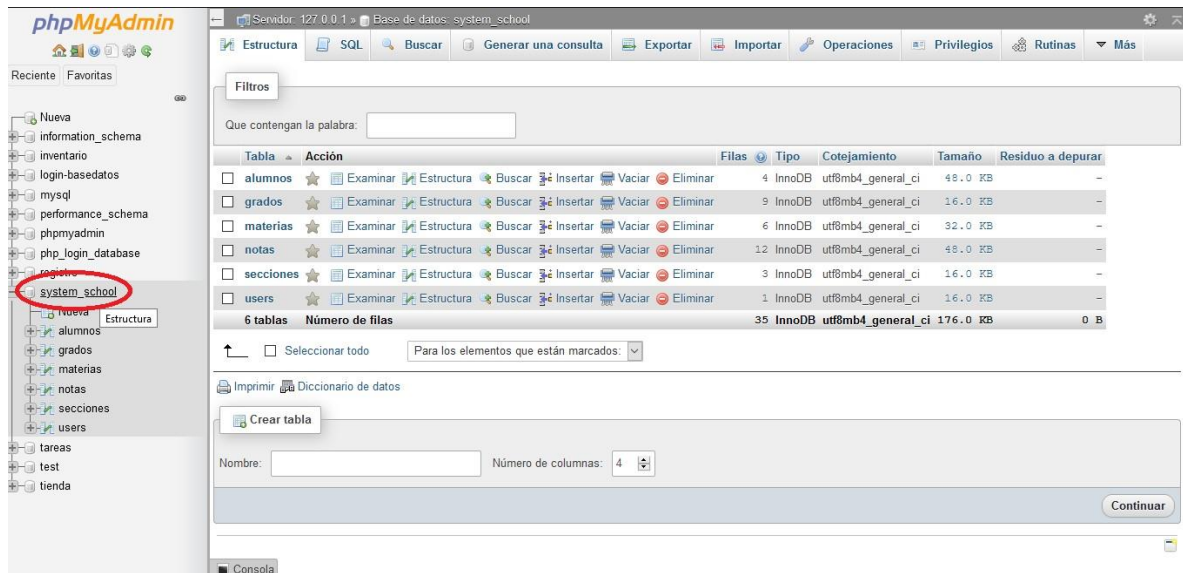


2. El Administrador tendrá que colocar una ruta de enlace (URL) que al actualizarse tiene acceso a PhpMyAdmin. URL de PhpMyAdmin: <http://localhost/phpmyadmin/index.php>

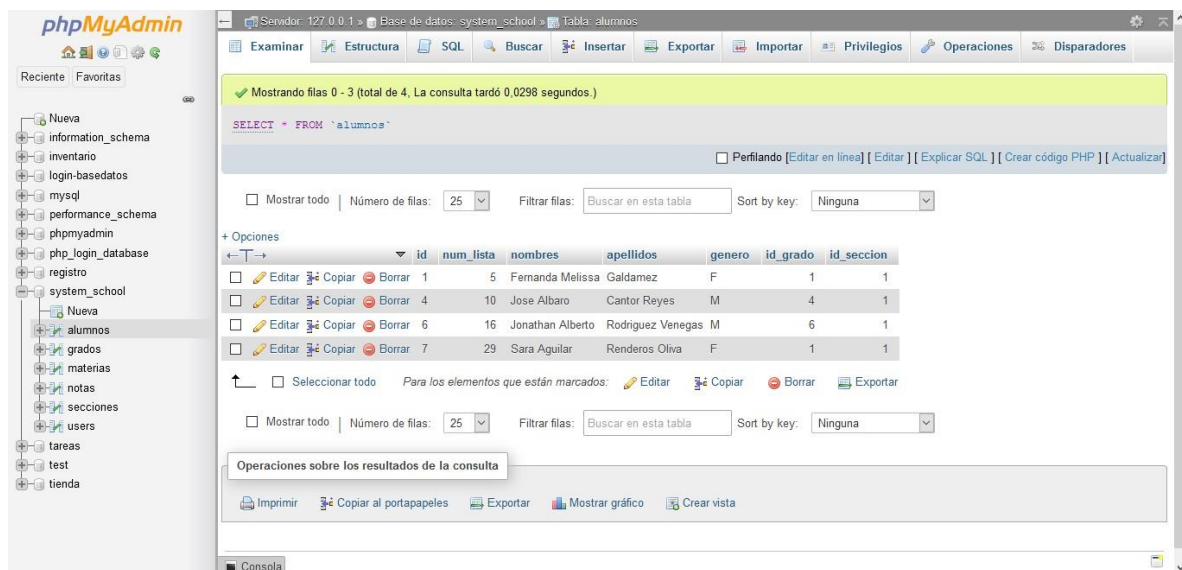
Una vez Importada la BD del sistema, Actualizamos la página de PhpMyAdmin carga el sistema.



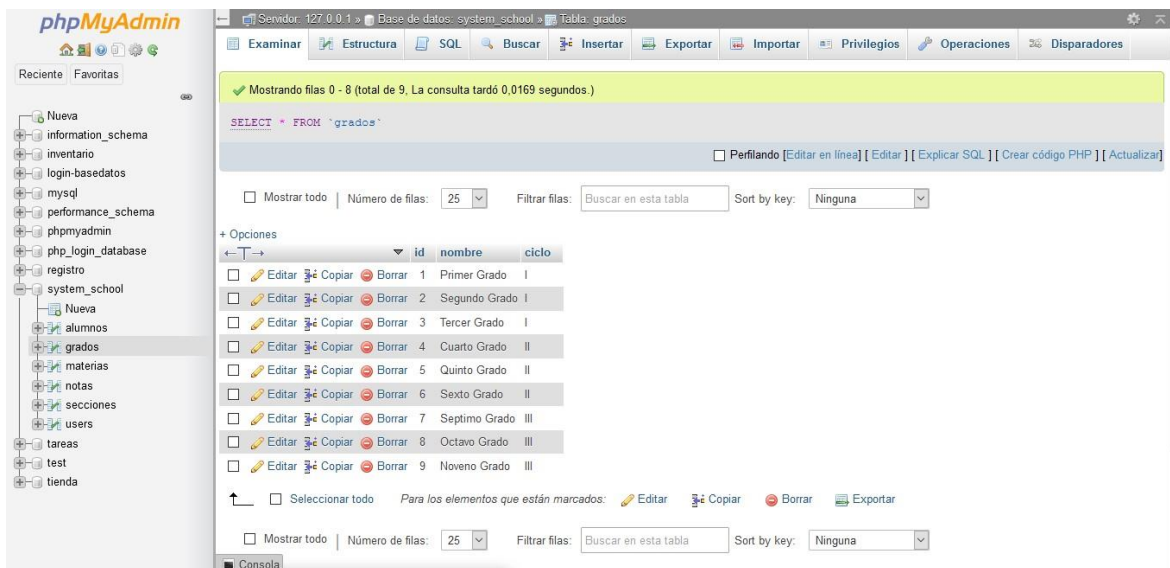
3. El administrador tiene el control desde el gestor de base de datos **MySQL en phpMyAdmin** puede visualizar la base de datos llamada “**system_school**” donde se puede visualizar la estructura de la tabla que obtiene 6 filas, puede ver la acción que se hace en dicha tabla, tiene el acceso a poder buscar, insertar, vaciar y eliminar.



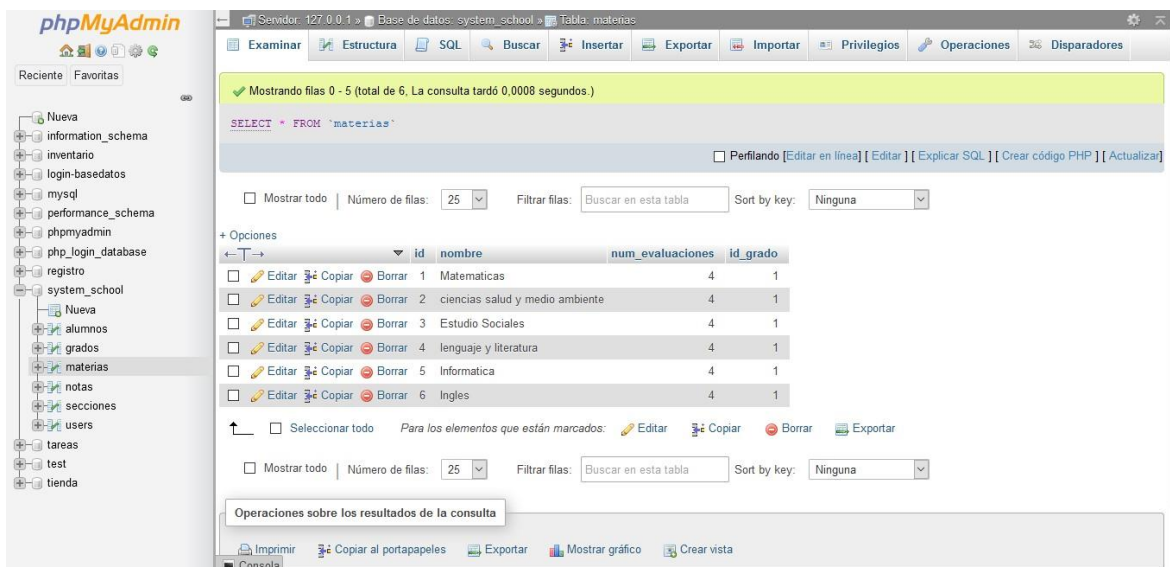
4. Mostrando la fila de alumnos, el Administrador podrá visualizar el registro que se generó desde el sistema de notas en la BD se muestran los resultados de la consulta. Tiene acceso a poder borrar un dato de estudiante editar, puede ver en los datos del alumno y actualizar.



5. **Grados.** Se puede visualizar el **id**, **nombre** y **ciclo**. El **administrador** tiene acceso a **editar**, **borrar** y **actualizar** los registros de la **BD**.



6. **Materias.** Podemos visualizar el **id** de la materia el **nombre** y el **número de evaluaciones** y el **id_grado**. El **administrador** tiene acceso a poder **editar**, **borrar** **agregar** y **actualizar** datos.



7. Visualizamos las opciones que obtiene la tabla Notas id, nota, observaciones, id alumno y un id materia de cada materia. El Administrador tiene acceso a editar, crear y actualizar datos.

The screenshot shows the phpMyAdmin interface with the 'Notas' table selected. The table structure is visible, showing columns: id, nota, observaciones, id_alumno, and id_materia. The data is displayed in a table with 12 rows. The interface includes a sidebar with a database tree, a top navigation bar, and a main content area with various options like 'Examinar', 'Estructura', 'SQL', etc.

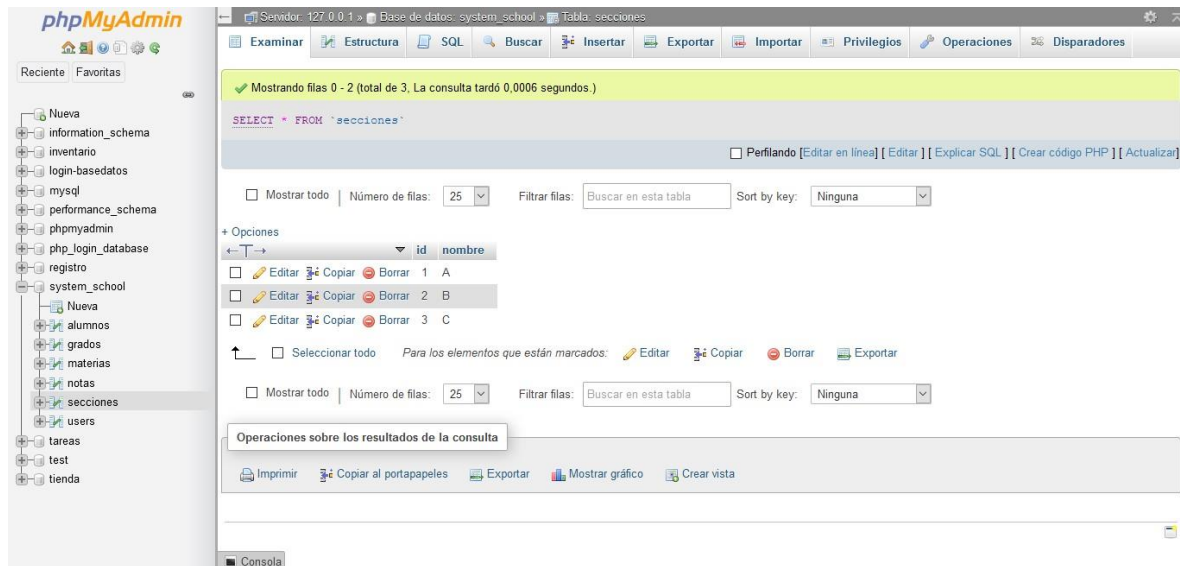
	id	nota	observaciones	id_alumno	id_materia
<input type="checkbox"/>	1	10		1	1
<input type="checkbox"/>	2	10		1	1
<input type="checkbox"/>	3	10		1	1
<input type="checkbox"/>	4	10		1	1
<input type="checkbox"/>	5	0		3	1
<input type="checkbox"/>	6	0		3	1
<input type="checkbox"/>	7	0		3	1
<input type="checkbox"/>	8	0		3	1
<input type="checkbox"/>	9	10		4	2
<input type="checkbox"/>	10	8		4	2
<input type="checkbox"/>	11	9		4	2
<input type="checkbox"/>	12	8		4	2

Se puede visualizar la estructura de la tabla y la acción:

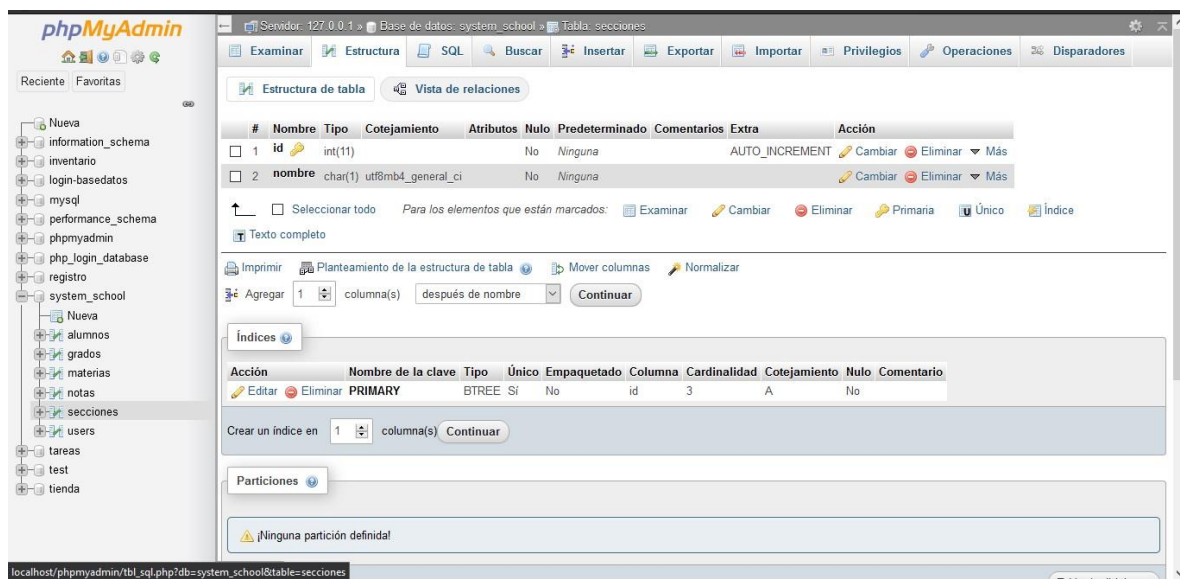
The screenshot shows the phpMyAdmin interface with the 'Notas' table structure selected. The table structure is visible, showing columns: id, nota, observaciones, id_alumno, and id_materia. The interface includes a sidebar with a database tree, a top navigation bar, and a main content area with various options like 'Examinar', 'Estructura', 'SQL', etc.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra	Acción
1	id	int(11)			No	Ninguna		AUTO_INCREMENT	<input type="checkbox"/> Cambiar <input type="checkbox"/> Eliminar <input type="checkbox"/> Más
2	nota	decimal(10,0)			Si	NULL			<input type="checkbox"/> Cambiar <input type="checkbox"/> Eliminar <input type="checkbox"/> Más
3	observaciones	varchar(255)	utf8mb4_general_ci		Si	NULL			<input type="checkbox"/> Cambiar <input type="checkbox"/> Eliminar <input type="checkbox"/> Más
4	id_alumno	int(11)			No	Ninguna			<input type="checkbox"/> Cambiar <input type="checkbox"/> Eliminar <input type="checkbox"/> Más
5	id_materia	int(11)			No	Ninguna			<input type="checkbox"/> Cambiar <input type="checkbox"/> Eliminar <input type="checkbox"/> Más

8. Secciones. Se puede visualizar el id y el nombre. El Administrador tiene acceso a poder editar, copiar, borrar y actualizar datos.



El Administrador puede visualizar la estructura de la tabla. Tiene acceso a poder cambiar, eliminar. Puede ver la acción de cada consulta de la tabla:



9. Users. En la tabla se puede visualizar el username y el password y el nombre y el rol que obtiene. El Administrador tiene acceso a poder editar, borrar y actualizar.

The screenshot shows the phpMyAdmin interface with the 'users' table selected in the 'system_school' database. The table structure is visible on the left sidebar. The main area displays the table data with columns: id, username, password, nombre, and rol. The data shows one user: id 1, username admin089, password admin012, nombre Administrador, rol Administrador. The interface includes various action buttons like 'Examinar', 'Estructura', 'SQL', 'Buscar', 'Insertar', 'Exportar', 'Importar', 'Privilegios', 'Operaciones', and 'Disparadores'.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra	Acción
1	id	int(11)			No	Ninguna		AUTO_INCREMENT	Cambiar Eliminar Más
2	username	varchar(45)	utf8mb4_general_ci		No	Ninguna			Cambiar Eliminar Más
3	password	varchar(45)	utf8mb4_general_ci		No	Ninguna			Cambiar Eliminar Más
4	nombre	varchar(45)	utf8mb4_general_ci		No	Ninguna			Cambiar Eliminar Más
5	rol	varchar(45)	utf8mb4_general_ci		No	Ninguna			Cambiar Eliminar Más

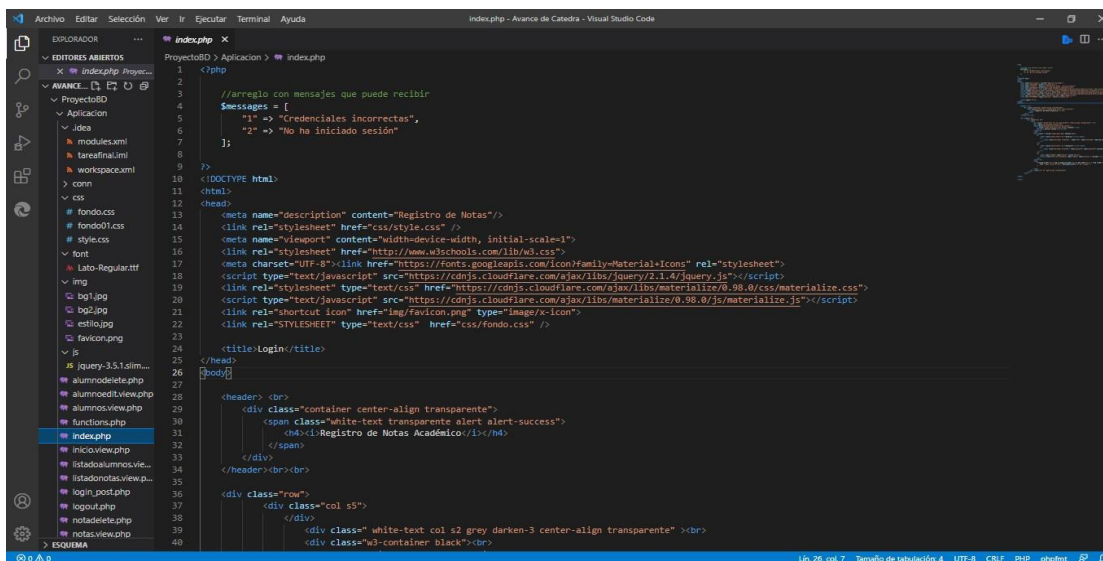
En la estructura de la tabla se visualiza el nombre, el tipo los atributos y la acción de cada dato.

The screenshot shows the phpMyAdmin interface with the 'users' table structure selected. The table structure is visible on the left sidebar. The main area displays the table structure with columns: id, username, password, nombre, and rol. The data shows one user: id 1, username admin089, password admin012, nombre Administrador, rol Administrador. The interface includes various action buttons like 'Examinar', 'Estructura', 'SQL', 'Buscar', 'Insertar', 'Exportar', 'Importar', 'Privilegios', 'Operaciones', and 'Disparadores'.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra	Acción
1	id	int(11)			No	Ninguna		AUTO_INCREMENT	Cambiar Eliminar Más
2	username	varchar(45)	utf8mb4_general_ci		No	Ninguna			Cambiar Eliminar Más
3	password	varchar(45)	utf8mb4_general_ci		No	Ninguna			Cambiar Eliminar Más
4	nombre	varchar(45)	utf8mb4_general_ci		No	Ninguna			Cambiar Eliminar Más
5	rol	varchar(45)	utf8mb4_general_ci		No	Ninguna			Cambiar Eliminar Más

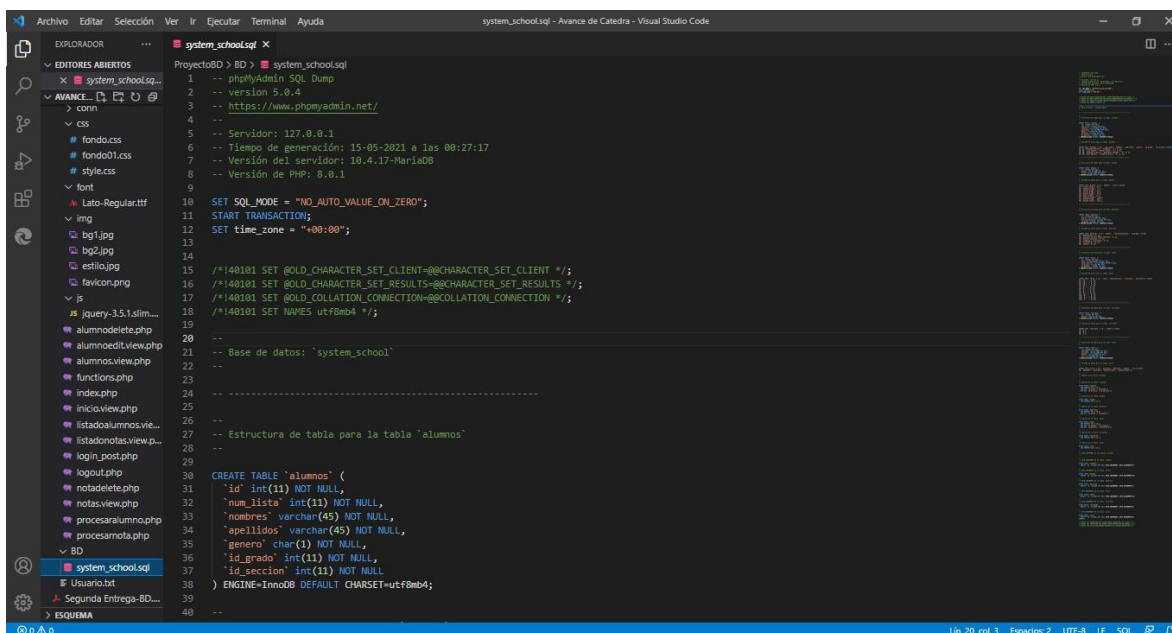
Código Fuente

1. Desde visual estudio code se puede visualizar el código del sistema. Desde la carpeta aplicación/el archivo Index.php se genera la aplicación de sistema de notas. El Administrador puede visualizar la ruta de cada etiqueta del código en pantalla. Puede generar una consulta.



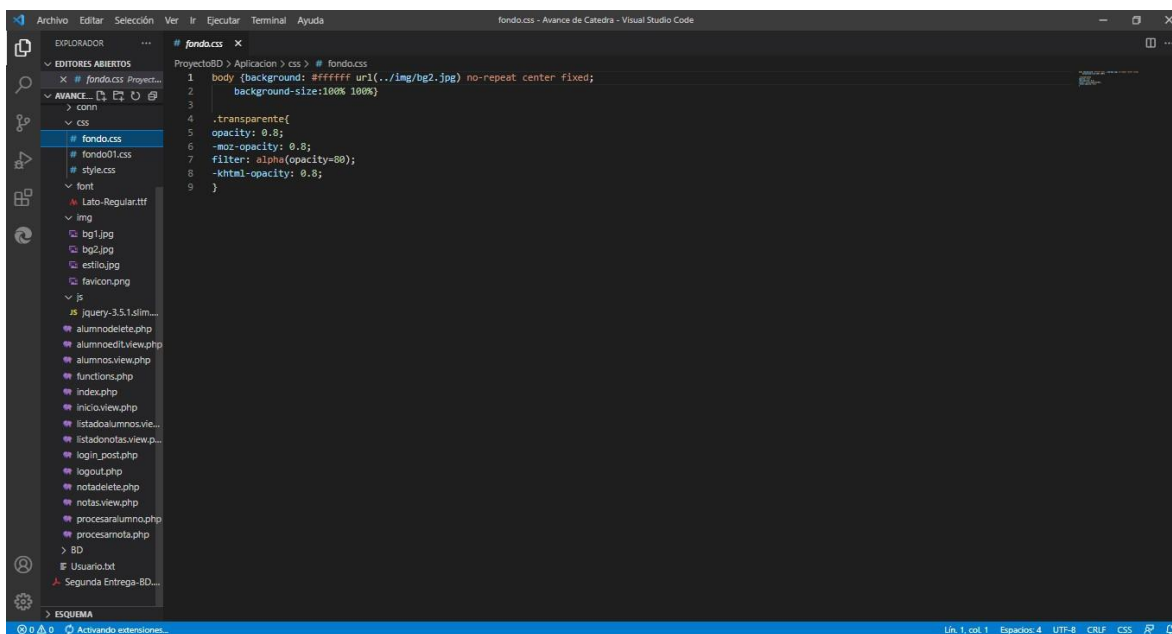
```
1 <?php
2
3 //arreglo con mensajes que puede recibir
4 $messages = [
5     "1" => "Credenciales incorrectas",
6     "2" => "No ha iniciado sesión"
7 ];
8
9
10 <!DOCTYPE html>
11 <html>
12 <head>
13     <meta name="description" content="Registro de Notas"/>
14     <link rel="stylesheet" href="css/style.css" />
15     <meta name="viewport" content="width=device-width, initial-scale=1"/>
16     <link rel="stylesheet" href="http://www.w3schools.com/lib/w3.css"/>
17     <meta charset="UTF-8"/>
18     <script type="text/javascript" src="https://cdnjs.cloudflare.com/ajax/libs/jquery/2.1.4/jquery.js"/>
19     <script type="text/javascript" src="https://cdnjs.cloudflare.com/ajax/libs/materialize/0.98.0/css/materialize.css"/>
20     <script type="text/javascript" src="https://cdnjs.cloudflare.com/ajax/libs/materialize/0.98.0/js/materialize.js"/>
21     <link rel="shortcut icon" href="img/favicon.png" type="image/x-icon"/>
22     <link rel="stylesheet" type="text/css" href="css/fondo.css" />
23
24 <title>Login</title>
25 </head>
26 <body>
27
28 <div class="container center-align transparente">
29     <div class="white-text transparent alert alert-success">
30         <h4>¡Registro de Notas Académico!</h4>
31     </div>
32 </div>
33
34 </div>
35
36 <div class="row">
37     <div class="col s5">
38         <div>
39             <div class="white-text col s2 grey darken-3 center-align transparente"><br>
40                 <div class="w3-container black"><br>
```

2. El administrador controla y supervisa la base de datos desde el código fuente, donde puede gestionar los errores del sistema.

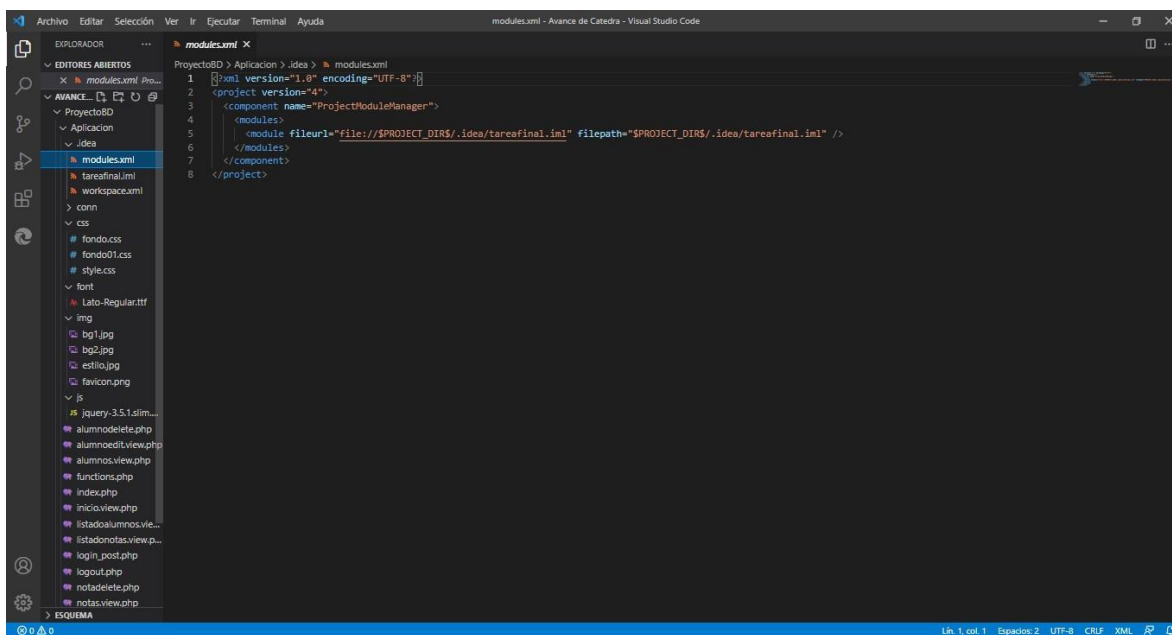


```
1 -- phpMyAdmin SQL Dump
2 -- version 5.0.4
3 -- https://www.phpmyadmin.net/
4 --
5 -- Servidor: 127.0.0.1
6 -- Tiempo de generación: 15-05-2021 a las 00:27:17
7 -- Versión del servidor: 10.4.17-MariaDB
8 -- Versión de PHP: 8.0.1
9
10 SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
11 START TRANSACTION;
12 SET time_zone = "+00:00";
13
14
15 /*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
16 /*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
17 /*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
18 /*!40101 SET NAMES utf8mb4 */;
19
20 --
21 -- Base de datos: 'system_school'
22 --
23
24
25
26 --
27 -- Estructura de tabla para la tabla 'alumnos'
28 --
29
30 CREATE TABLE `alumnos` (
31   `id` int(11) NOT NULL,
32   `num_lista` int(11) NOT NULL,
33   `nombres` varchar(45) NOT NULL,
34   `apellidos` varchar(45) NOT NULL,
35   `genero` char(1) NOT NULL,
36   `id_grupo` int(11) NOT NULL,
37   `id_seccion` int(11) NOT NULL
38 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
39
40 --
```

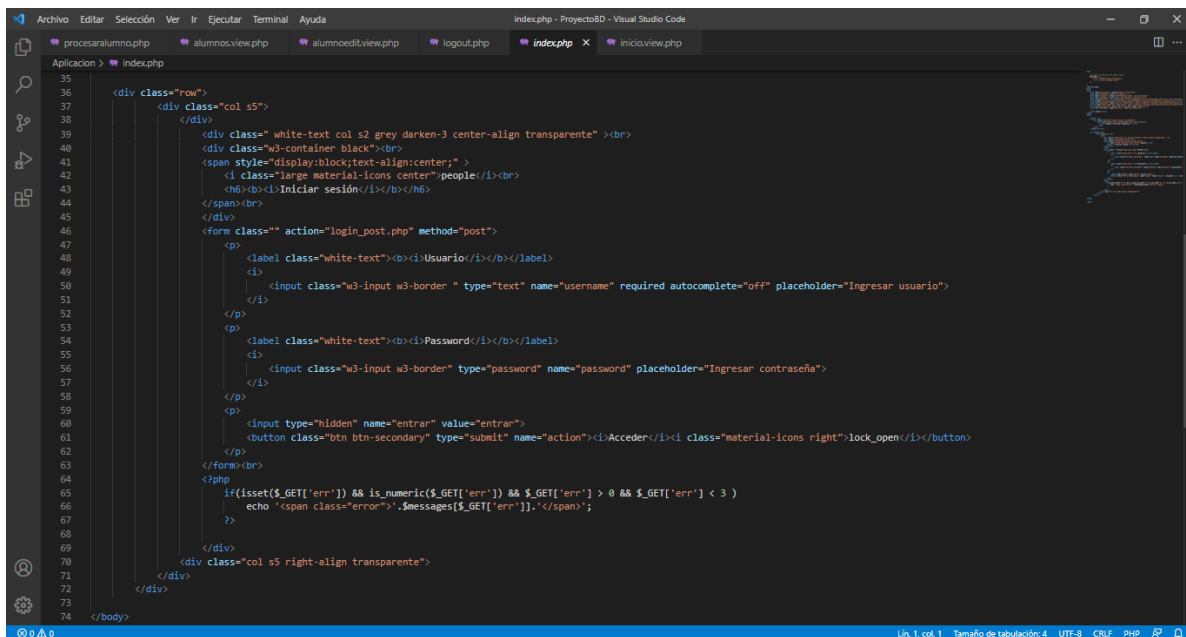

3. En la carpeta CSS se puede visualizar los fondos de los códigos y el estilo.css



4. Módulos.

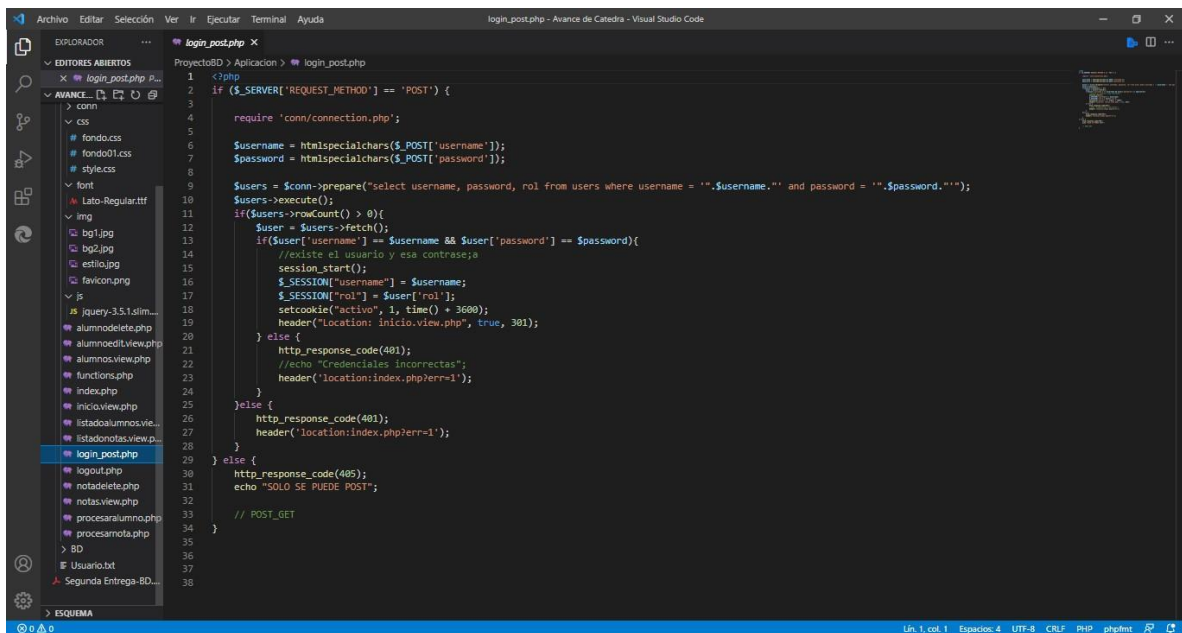


5. Index.php en este archivo.php se muestra el código para el login del sistema, en el cual se ingresa el usuario y contraseña una vez completado esto, lo manda al archivo login_post.php.



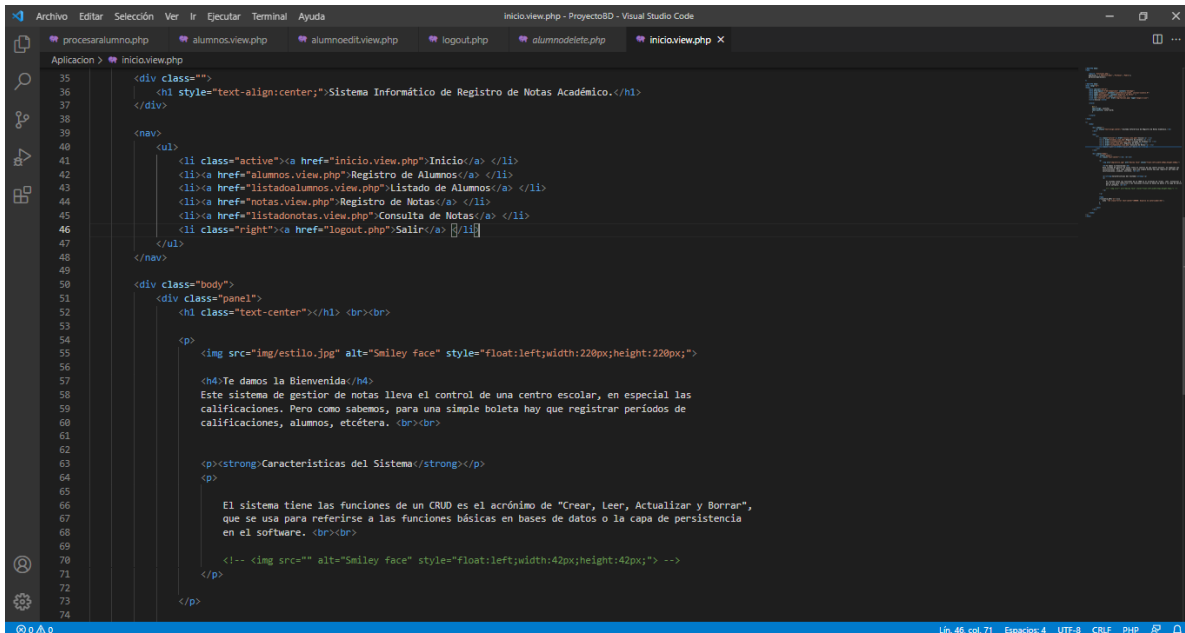
```
35 <div class="row">
36 <div class="col s5">
37 </div>
38 </div>
39 <div class="white-text col s2 grey darken-3 center-align transparente"><br>
40 <div class="w3-container black"><br>
41 <span style="display:block;text-align:center;">
42 <i class="large material-icons center">people</i><br>
43 <h2><i>Iniciar sesión</i></h2>
44 </span><br>
45 </div>
46 <form class="" action="login_post.php" method="post">
47 <p>
48 <label class="white-text"><b><i>Usuario</i></b></label>
49 <i>
50 <input class="w3-input w3-border" type="text" name="username" required autocomplete="off" placeholder="Ingresar usuario">
51 </i>
52 </p>
53 <p>
54 <label class="white-text"><b><i>Password</i></b></label>
55 <i>
56 <input class="w3-input w3-border" type="password" name="password" placeholder="Ingresar contraseña">
57 </i>
58 </p>
59 <p>
60 <input type="hidden" name="entrar" value="entrar">
61 <button class="btn btn-secondary" type="submit" name="action"><i>Acceder</i></button><i class="material-icons right">lock_open</i></button>
62 </p>
63 </form><br>
64 <div>
65 <php
66 <?php
67 <?php
68 <?php
69 <?php
70 <div class="col s5 right-align transparente">
71 </div>
72 </div>
73 </body>
74 </div>
```

- 5.1. El login_post.php es quien verifica si el usuario y contraseña están registrados en la BD si esto es así, permite acceder al sistema caso contrario no podrá acceder.



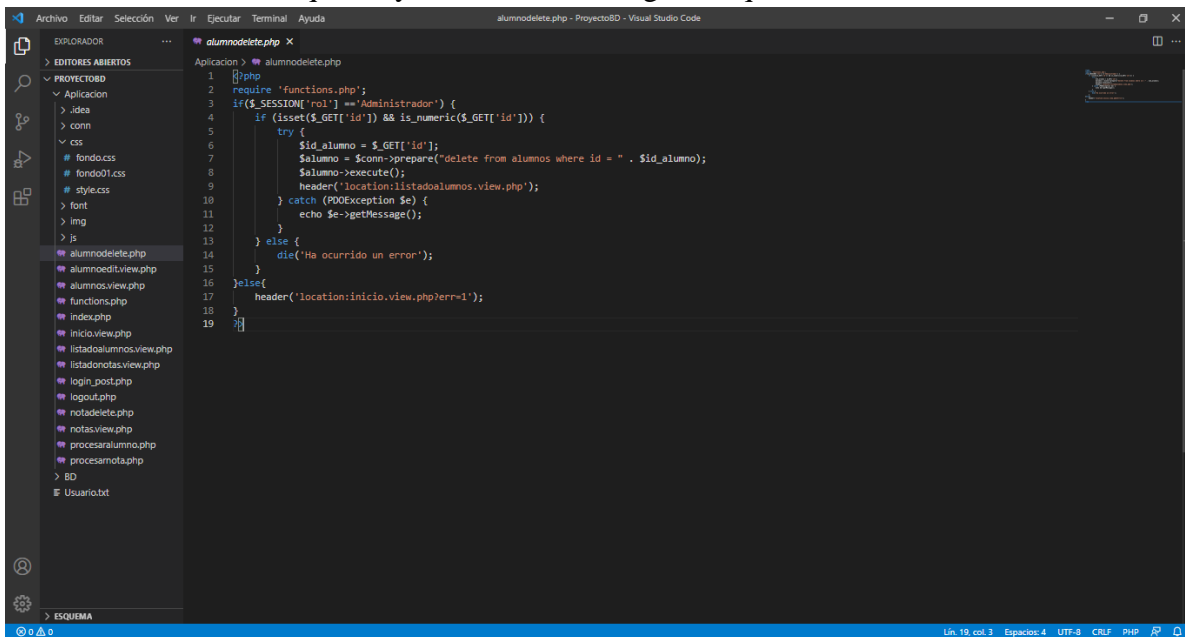
```
1 <?php
2 if ($SERVER['REQUEST_METHOD'] == 'POST') {
3
4     require 'conn/connection.php';
5
6     $username = htmlspecialchars($_POST['username']);
7     $password = htmlspecialchars($_POST['password']);
8
9     $users = $conn->prepare("select username, password, rol from users where username = '".$username."' and password = '".$password."'");
10    $users->execute();
11    if($users->rowCount() > 0){
12        $user = $users->fetch();
13        if($user['username'] == $username && $user['password'] == $password){
14            //existe el usuario y esa contraseña
15            session_start();
16            $_SESSION['username'] = $username;
17            $_SESSION['rol'] = $user['rol'];
18            setcookie("activo", 1, time() + 3600);
19            header("location: inicio.view.php", true, 301);
20        } else {
21            http_response_code(401);
22            //echo "Credenciales incorrectas";
23            header("location:index.php?err=1");
24        }
25    } else {
26        http_response_code(401);
27        header("location:index.php?err=1");
28    }
29 } else {
30     http_response_code(405);
31     echo "SOLO SE PUEDE POST";
32 }
33 // POST_GET
34 }
35
36
37
38
```

6. En este archivo se muestra la pagina principal dentro del sistema donde se da la bienvenida y las características del sistema.



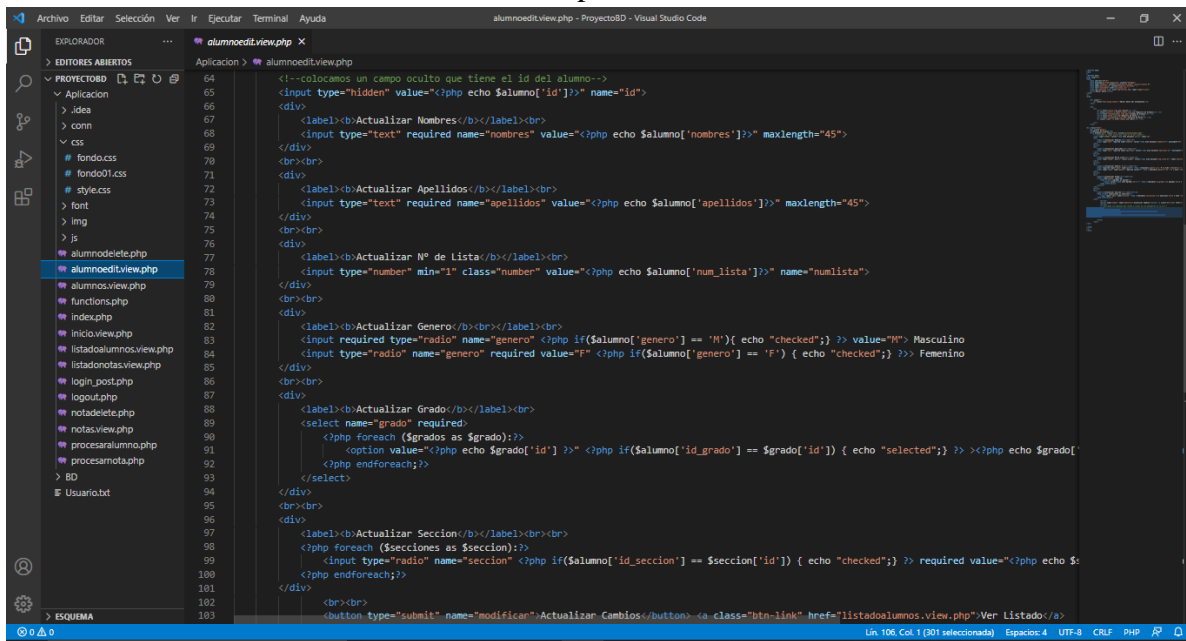
```
35 <div class="">
36 <h1 style="text-align:center;">Sistema Informático de Registro de Notas Académico.</h1>
37 </div>
38
39 <nav>
40 <ul>
41 <li class="active"><a href="inicio.view.php">Inicio</a> </li>
42 <li><a href="alumnos.view.php">Registro de Alumnos</a> </li>
43 <li><a href="listadoalumnos.view.php">Listado de Alumnos</a> </li>
44 <li><a href="notas.view.php">Registro de Notas</a> </li>
45 <li><a href="listadonotas.view.php">Consulta de Notas</a> </li>
46 <li class="right"><a href="logout.php">Salir</a> </li>
47 </ul>
48 </nav>
49
50 <div class="body">
51 <div class="panel">
52 <h1 class="text-center"></h1> <br><br>
53
54 <p>
55 
56
57 <h4>Te damos la Bienvenida</h4>
58 Este sistema de gestor de notas lleva el control de una centro escolar, en especial las
59 calificaciones. Pero como sabemos, para una simple boleta hay que registrar periodos de
60 calificaciones, alumnos, etcétera. <br><br>
61
62 <p><strong>Características del Sistema</strong></p>
63 <p>
64 El sistema tiene las funciones de un CRUD es el acrónimo de "Crear, Leer, Actualizar y Borrar",
65 que se usa para referirse a las funciones básicas en bases de datos o la capa de persistencia
66 en el software. <br><br>
67 <!-- <img src="" alt="Smiley face" style="float:left;width:42px;height:42px;" -->
68 </p>
69
70 </div>
71
72 </div>
73
74 </div>
```

7. Este archivo es quien ayuda a borrar los registros que están dentro de la BD.



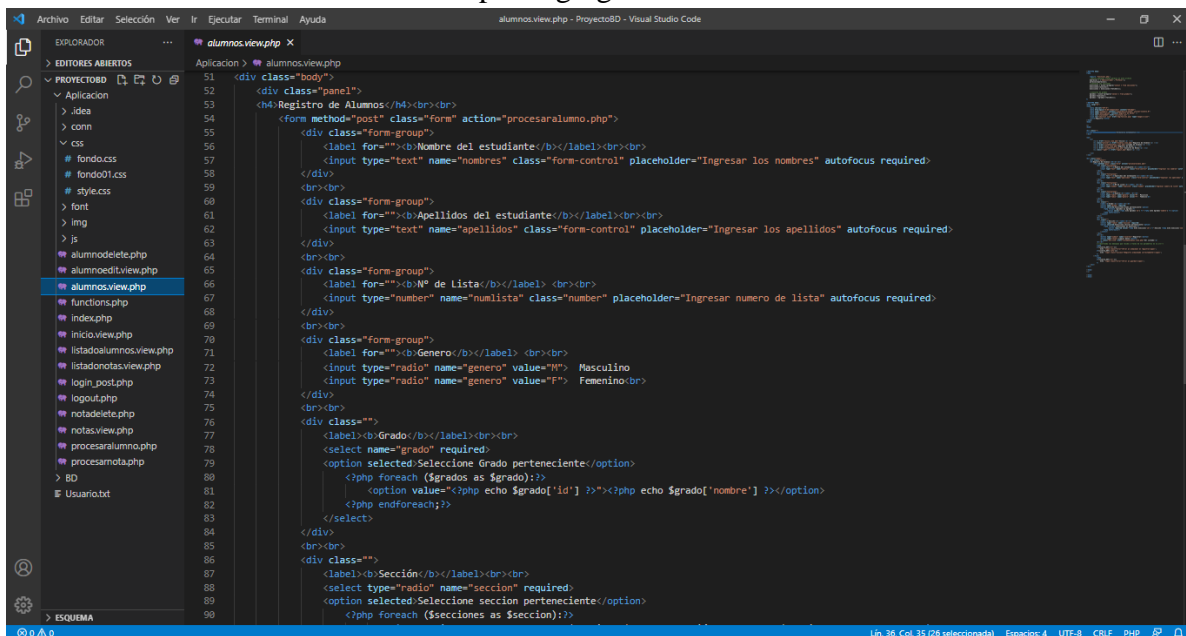
```
1 <?php
2 require "functions.php";
3 if($SESSION['rol'] == "Administrador") {
4     if (isset($_GET['id']) && is_numeric($_GET['id'])) {
5         try {
6             $id_alumno = $_GET['id'];
7             $alumno = $conn->prepare("delete from alumnos where id = " . $id_alumno);
8             $alumno->execute();
9             header("location:listadoalumnos.view.php");
10         } catch (PDOException $e) {
11             echo $e->getMessage();
12         }
13     } else {
14         die('Ha ocurrido un error');
15     }
16 } else {
17     header("location:inicio.view.php?err=1");
18 }
19 }
```

8. En este archivo es nuestro UPDATE para actualizar los datos de la BD.



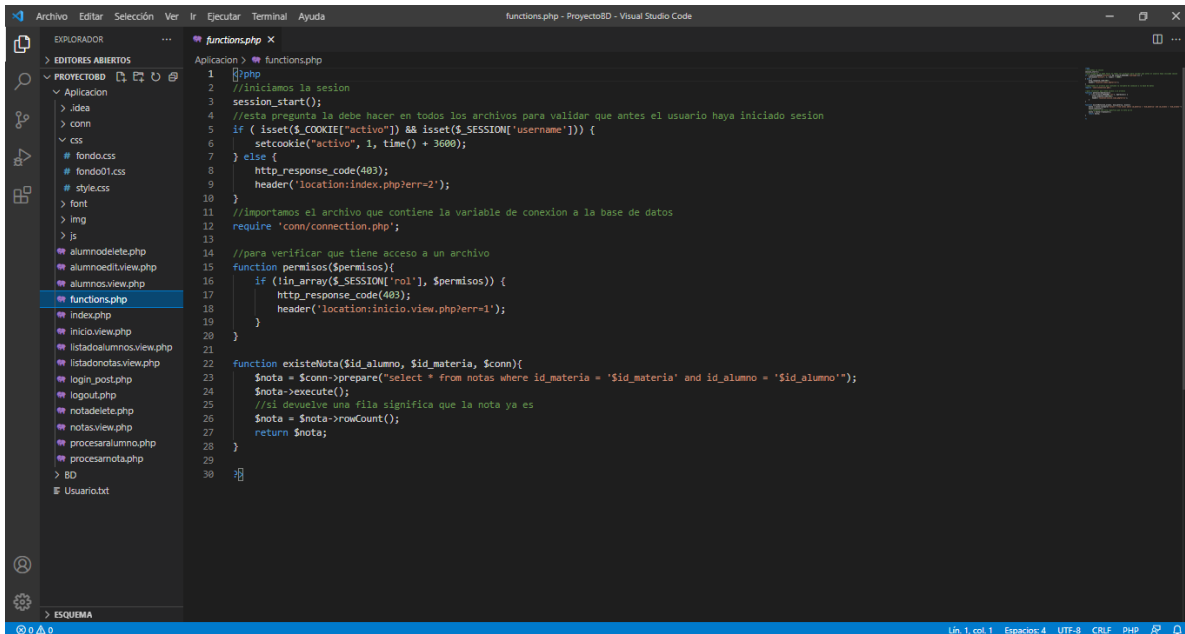
```
64 <!--colocamos un campo oculto que tiene el id del alumno-->
65 <input type="hidden" value="{php echo $alumno['id']}" name="id"/>
66 <div>
67 <label><b>Actualizar Nombres</b></label><br>
68 <input type="text" required name="nombres" value="{php echo $alumno['nombres']}" maxlength="45"/>
69 </div>
70 <br><br>
71 <div>
72 <label><b>Actualizar Apellidos</b></label><br>
73 <input type="text" required name="apellidos" value="{php echo $alumno['apellidos']}" maxlength="45"/>
74 </div>
75 <br><br>
76 <div>
77 <label><b>Actualizar N° de Lista</b></label><br>
78 <input type="number" min="1" class="number" value="{php echo $alumno['num_lista']}" name="numlista"/>
79 </div>
80 <br><br>
81 <div>
82 <label><b>Actualizar Genero</b></label><br>
83 <input required type="radio" name="genero" <?php if($alumno['genero'] == 'M'){ echo "checked";} >? value="M"> Masculino
84 <input type="radio" name="genero" required value="F" <?php if($alumno['genero'] == 'F') { echo "checked";} >? > Feminino
85 </div>
86 <br><br>
87 <div>
88 <label><b>Actualizar Grado</b></label><br>
89 <select name="grado" required>
90 <?php foreach ($grados as $grado)>
91 <option value="{php echo $grado['id']}" <?php if($alumno['id_grado'] == $grado['id']) { echo "selected";} >? >{php echo $grado['nombre']}>
92 </php endforeach>
93 </select>
94 </div>
95 <br><br>
96 <div>
97 <label><b>Actualizar Sección</b></label><br>
98 <?php foreach ($secciones as $seccion)>
99 <input type="radio" name="seccion" <?php if($alumno['id_seccion'] == $seccion['id']) { echo "checked";} >? required value="{php echo $seccion['id']}" <?php echo $seccion['nombre']>
100 </php endforeach>
101 </div>
102 <br><br>
103 <button type="submit" name="modificar">Actualizar Cambios</button><a class="btn-link" href="listadoalumnos.view.php">Ver Listado</a>
```

9. Este archivo es el formulario para agregar nuevos datos de los alumnos.



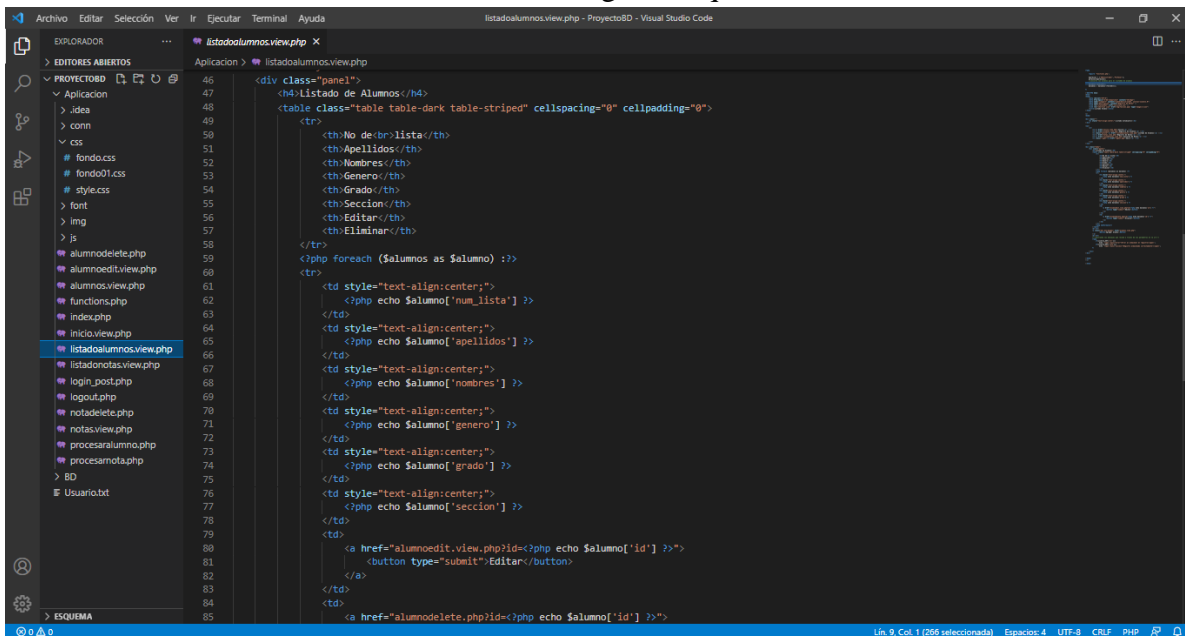
```
51 <div class="body">
52 <div class="panel">
53 <h4>Registro de Alumnos</h4><br><br>
54 <form method="post" class="form" action="procesaralumno.php">
55 <div class="form-group">
56 <label for=""><b>Nombre del estudiante</b></label><br>
57 <input type="text" name="nombres" class="form-control" placeholder="Ingresar los nombres" autofocus required>
58 </div>
59 <br><br>
60 <div class="form-group">
61 <label for=""><b>Apellidos del estudiante</b></label><br>
62 <input type="text" name="apellidos" class="form-control" placeholder="Ingresar los apellidos" autofocus required>
63 </div>
64 <br><br>
65 <div class="form-group">
66 <label for=""><b>N° de Lista</b></label><br>
67 <input type="number" name="numlista" class="number" placeholder="Ingresar numero de lista" autofocus required>
68 </div>
69 <br><br>
70 <div class="form-group">
71 <label for=""><b>Genero</b></label><br>
72 <input type="radio" name="genero" value="M"> Masculino
73 <input type="radio" name="genero" value="F"> Feminino<br>
74 </div>
75 <br><br>
76 <div class="">
77 <label><b>Grado</b></label><br>
78 <select name="grado" required>
79 <option selected>Seleccione Grado perteneciente</option>
80 <?php foreach ($grados as $grado)>
81 <option value="{php echo $grado['id']}" <?php echo $grado['nombre']> <?php echo $grado['nombre']> </option>
82 </php endforeach>
83 </select>
84 </div>
85 <br><br>
86 <div class="">
87 <label><b>Sección</b></label><br>
88 <select type="radio" name="seccion" required>
89 <option selected>Seleccione seccion perteneciente</option>
90 <?php foreach ($secciones as $seccion)>
```

10. Funciones del sistema,



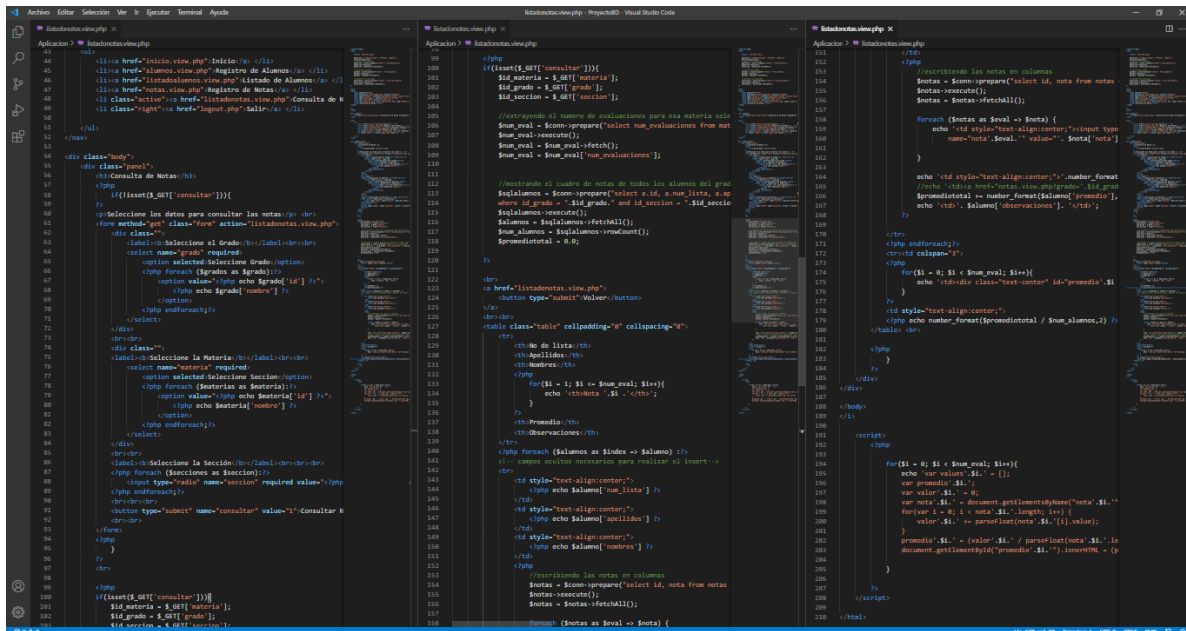
```
1  <?php
2  //iniciamos la sesion
3  session_start();
4  //esta pregunta la debe hacer en todos los archivos para validar que antes el usuario haya iniciado sesion
5  if ( !isset($_COOKIE["activo"]) && !isset($_SESSION["username"])) {
6      setcookie("activo", 1, time() + 3600);
7  } else {
8      http_response_code(403);
9      header('location:index.php?err=2');
10 }
11 //importamos el archivo que contiene la variable de conexión a la base de datos
12 require 'conn/connection.php';
13
14 //para verificar que tiene acceso a un archivo
15 function permisos($permisos){
16     if (!in_array($_SESSION['rol'], $permisos)) {
17         http_response_code(403);
18         header('location:inicio.view.php?err=1');
19     }
20 }
21
22 function existeNota($id_alumno, $id_materia, $conn){
23     $nota = $conn->prepare("select * from notas where id_materia = 'id_materia' and id_alumno = 'id_alumno'");
24     $nota->execute();
25     //si devuelve una fila significa que la nota ya es
26     $nota = $nota->rowCount();
27     return $nota;
28 }
29
30
```

11. En este archivo mostramos todos los registros, que están en la BD.

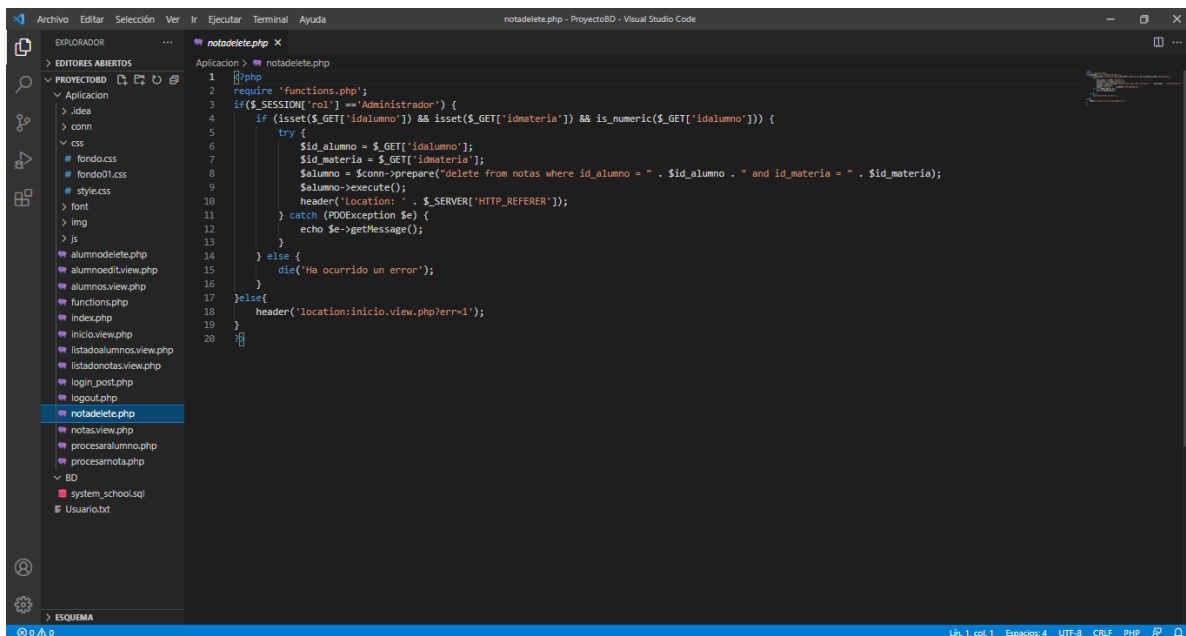


```
46 <div class="panel">
47     <h4>Listado de Alumnos</h4>
48     <table class="table table-dark table-striped" cellspacing="0" cellpadding="0">
49         <tr>
50             <th>No de<br>lista</th>
51             <th>Apellidos</th>
52             <th>Nombres</th>
53             <th>Genero</th>
54             <th>Grado</th>
55             <th>Seccion</th>
56             <th>Editar</th>
57             <th>Eliminar</th>
58         </tr>
59         <?php foreach ($alumnos as $alumno) :?>
60             <tr>
61                 <td style="text-align:center;">
62                     <?php echo $alumno['num_lista'] ?>
63                 </td>
64                 <td style="text-align:center;">
65                     <?php echo $alumno['apellidos'] ?>
66                 </td>
67                 <td style="text-align:center;">
68                     <?php echo $alumno['nombres'] ?>
69                 </td>
70                 <td style="text-align:center;">
71                     <?php echo $alumno['genero'] ?>
72                 </td>
73                 <td style="text-align:center;">
74                     <?php echo $alumno['grado'] ?>
75                 </td>
76                 <td style="text-align:center;">
77                     <?php echo $alumno['seccion'] ?>
78                 </td>
79                 <td>
80                     <a href="alumnoedit.view.php?id=<?php echo $alumno['id'] ?>">
81                         <button type="submit">Editar</button>
82                     </a>
83                 </td>
84                 <td>
85                     <a href="alumnodelete.php?id=<?php echo $alumno['id'] ?>">
```

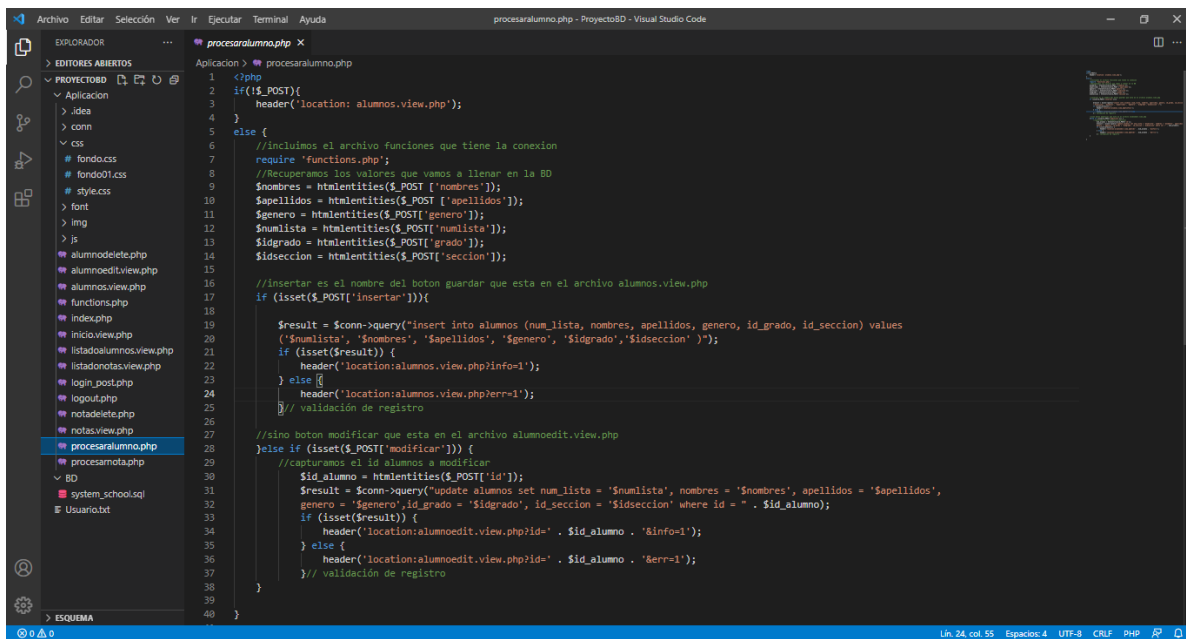

12. Este archivo es un poco largo, en lo dividí en 3 partes. Este archivo muestra todas las notas de los alumnos, quien hace la lógica para que el resultado sea perfecto.



13. Este archivo ayuda a eliminar las notas que están registradas en la BD.

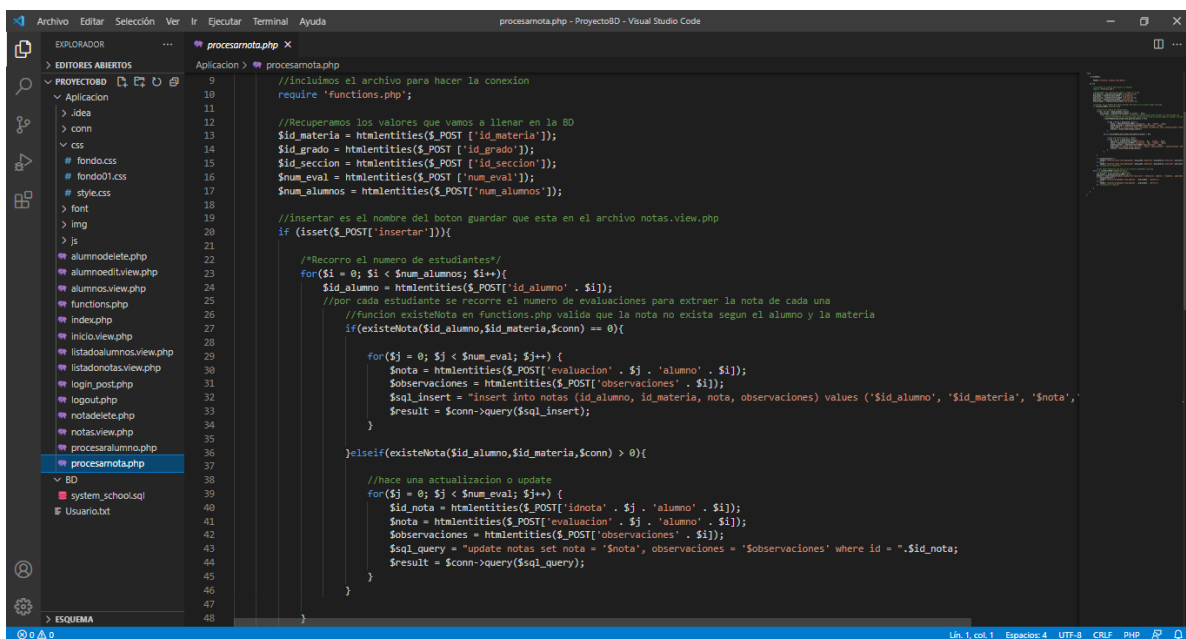


14. Este archivo es de mucha importancia dentro del sistema para actualizar y eliminar cada registro que está dentro de la BD. Este archivo ayuda a procesar los datos del BD para que el resultado tanto de actualizar como eliminar sea satisfactorio.



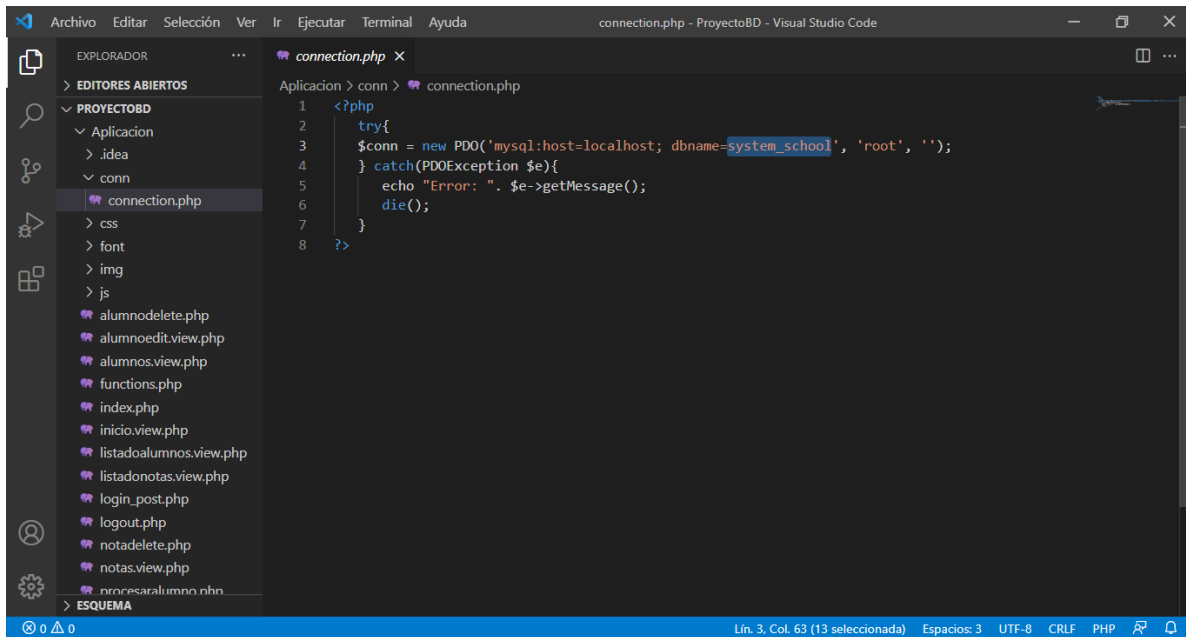
```
1 <?php
2 if(isset($_POST)){
3     header('location: alumnos.view.php');
4 }
5 else {
6     //incluimos el archivo funciones que tiene la conexion
7     require 'functions.php';
8     //Recuperamos los valores que vamos a llenar en la BD
9     $nombres = htmlentities($_POST ['nombres']);
10    $apellidos = htmlentities($_POST ['apellidos']);
11    $genero = htmlentities($_POST['genero']);
12    $numlista = htmlentities($_POST['numlista']);
13    $idgrado = htmlentities($_POST['grado']);
14    $idseccion = htmlentities($_POST['seccion']);
15
16    //insertar es el nombre del boton guardar que esta en el archivo alumnos.view.php
17    if (isset($_POST['insertar'])) {
18
19        $result = $conn->query("insert into alumnos (num_lista, nombres, apellidos, genero, id_grado, id_seccion) values
20        ('$numlista', '$nombres', '$apellidos', '$genero', '$idgrado', '$idseccion' );");
21        if (isset($result)) {
22            header('location:alumnos.view.php?info=1');
23        } else {
24            header('location:alumnos.view.php?err=1');
25        } // validación de registro
26
27        //sino boton modificar que esta en el archivo alumnoedit.view.php
28    } else if (isset($_POST['modificar'])) {
29        //capturamos el id alumnos a modificar
30        $id_alumno = htmlentities($_POST['id']);
31        $result = $conn->query("update alumnos set num_lista = '$numlista', nombres = '$nombres', apellidos = '$apellidos',
32        genero = '$genero', id_grado = '$idgrado', id_seccion = '$idseccion' where id = " . $id_alumno);
33        if (isset($result)) {
34            header('location:alumnoedit.view.php?id=' . $id_alumno . '&info=1');
35        } else {
36            header('location:alumnoedit.view.php?id=' . $id_alumno . '&err=1');
37        } // validación de registro
38    }
39 }
40 }
```

15. Archivo de procesar de notas.



```
9 //incluimos el archivo para hacer la conexion
10 require 'functions.php';
11
12 //Recuperamos los valores que vamos a llenar en la BD
13 $id_materia = htmlentities($_POST ['id_materia']);
14 $id_grado = htmlentities($_POST ['id_grado']);
15 $id_seccion = htmlentities($_POST ['id_seccion']);
16 $num_eval = htmlentities($_POST ['num_eval']);
17 $num_alumnos = htmlentities($_POST ['num_alumnos']);
18
19 //insertar es el nombre del boton guardar que esta en el archivo notas.view.php
20 if (isset($_POST['insertar'])) {
21
22     //Recorro el numero de estudiantes*/
23     for($i = 0; $i < $num_alumnos; $i++){
24         $id_alumno = htmlentities($_POST['id_alumno' . $i]);
25         //por cada estudiante se recorre el numero de evaluaciones para extraer la nota de cada una
26         //funcion existeNota en functions.php valida que la nota no exista segun el alumno y la materia
27         if(existeNota($id_alumno,$id_materia,$conn) == 0){
28
29             for($j = 0; $j < $num_eval; $j++) {
30                 $nota = htmlentities($_POST['evaluacion' . $j . 'alumno' . $i]);
31                 $observaciones = htmlentities($_POST['observaciones' . $i]);
32                 $sql_insert = "insert into notas (id_alumno, id_materia, nota, observaciones) values ('$id_alumno', '$id_materia', '$nota',
33                 '$observaciones' );";
34                 $result = $conn->query($sql_insert);
35             }
36         } elseif(existeNota($id_alumno,$id_materia,$conn) > 0){
37
38             //hace una actualizacion o update
39             for($j = 0; $j < $num_eval; $j++) {
40                 $id_nota = htmlentities($_POST['idnota' . $j . 'alumno' . $i]);
41                 $nota = htmlentities($_POST['evaluacion' . $j . 'alumno' . $i]);
42                 $observaciones = htmlentities($_POST['observaciones' . $i]);
43                 $sql_query = "update notas set nota = '$nota', observaciones = '$observaciones' where id = " . $id_nota;
44                 $result = $conn->query($sql_query);
45             }
46         }
47     }
48 }
```

16. Conexión de la PHP con MySQL ... Como se puede visualizar hemos colocado el nombre de la BD para que, tenga conectividad con el fin de que todos los datos procesados dentro del sistema se guarden en la BD.



```
1 <?php
2 try{
3     $conn = new PDO('mysql:host=localhost; dbname=system_school', 'root', '');
4 } catch(PDOException $e){
5     echo "Error: ". $e->getMessage();
6     die();
7 }
8 ?>
```

Visual Studio Code interface showing the file explorer on the left with the project structure expanded. The main editor displays the contents of `connection.php`. The status bar at the bottom indicates the current cursor position and file encoding.