

UNIVERSIDAD DON BOSCO



FACULTAD DE INGENIERÍA ESCUELA DE COMPUTACIÓN

Desarrollo de aplicaciones web con software interpretado
en el servidor (DSS) G05T.

Proyecto de catedra:

Manual técnico.

Alumnos:

José Álvarez Mejía **AM232553**

Diego Canjura Estrada **CE221950**

Oswaldo Henríquez Miranda **HM231279**

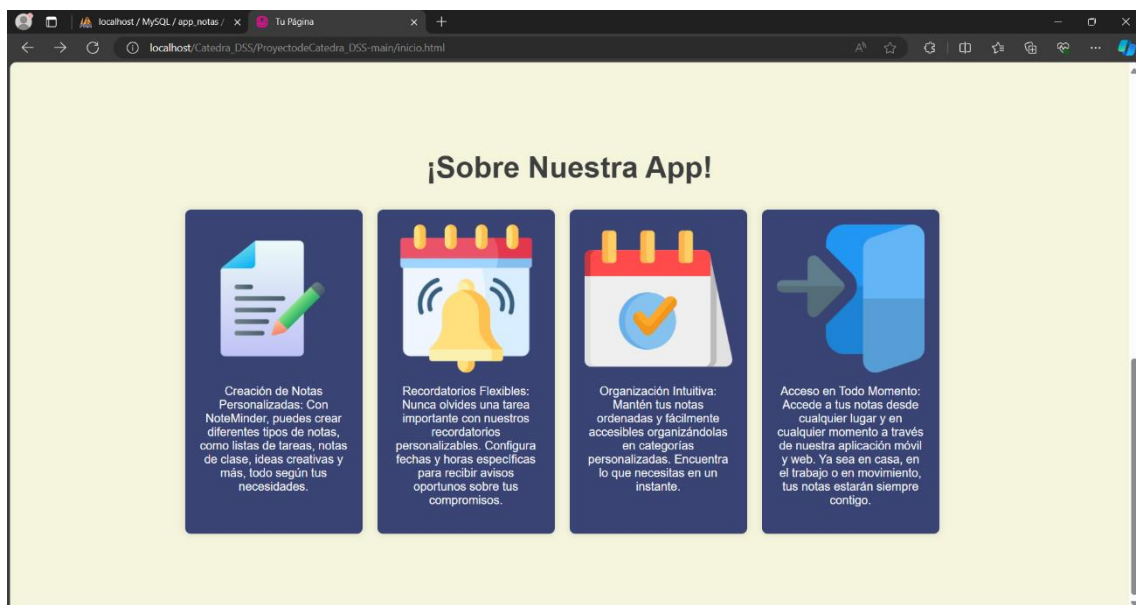
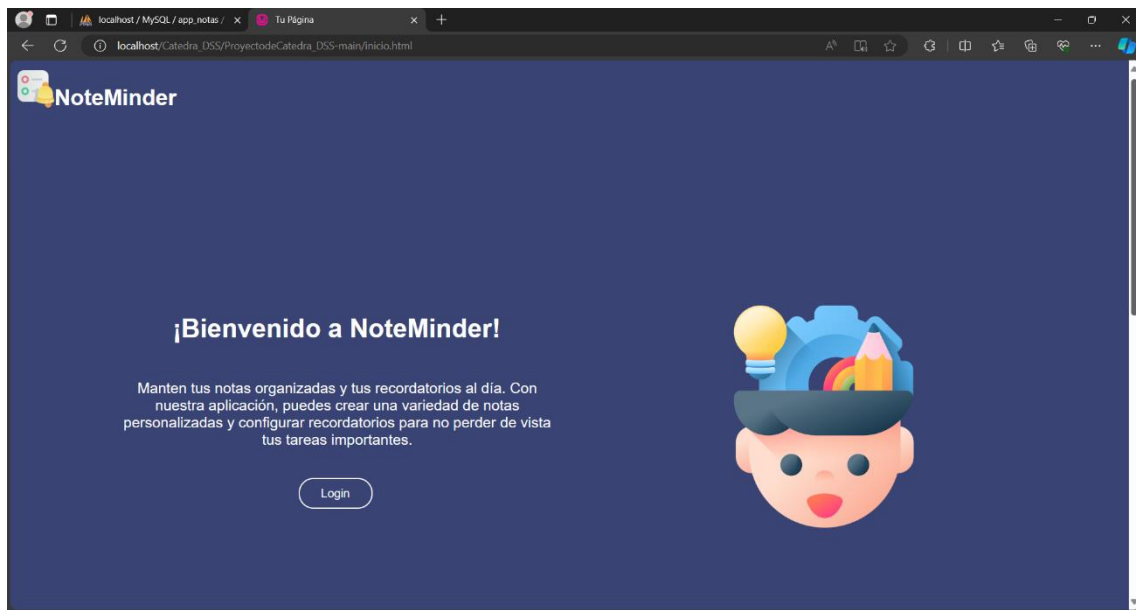
Jorge Mira Flores **MF232232**

Kevin Ramírez Montiel **RM230143**

En este manual técnico se hablará más profundamente sobre las pantallas de la aplicación, en los que se enseñará como funciona y se explicará cada archivo php.

Pantalla 1, Inicio:

Al iniciar el programa, el usuario verá la página principal donde encontrará más información sobre la aplicación, en la cual se podrán crear notas personalizadas, recordatorios flexibles, organización intuitiva y se podrá tener acceso en todo momento. También se encuentra un botón de “Login” que fue hecho con JavaScript, donde los usuarios podrán crear una cuenta y podrán iniciar sesión.



Conexión a la base de datos: db_connection.php.

A lo largo de nuestra aplicación es necesario el uso de una base de datos, para que almacene información importante, como los usuarios que se registran con sus respectivos campos como nombre, correo electrónico, usuario y contraseña.

La base de datos también es necesaria para que se almacene toda la información sobre las notas, con todos sus campos que son título, contenido, color, etc. Así mismo, también es indispensable para que se guarde toda la información de los recordatorios con la fecha y hora en la que se desea la alerta.

Para hacer la conexión del programa con la base de datos, lo hicimos mediante el siguiente código php:

```
<?php
$servername = "localhost";
$username = "root";
$password = "";
$dbname = "app_notas";

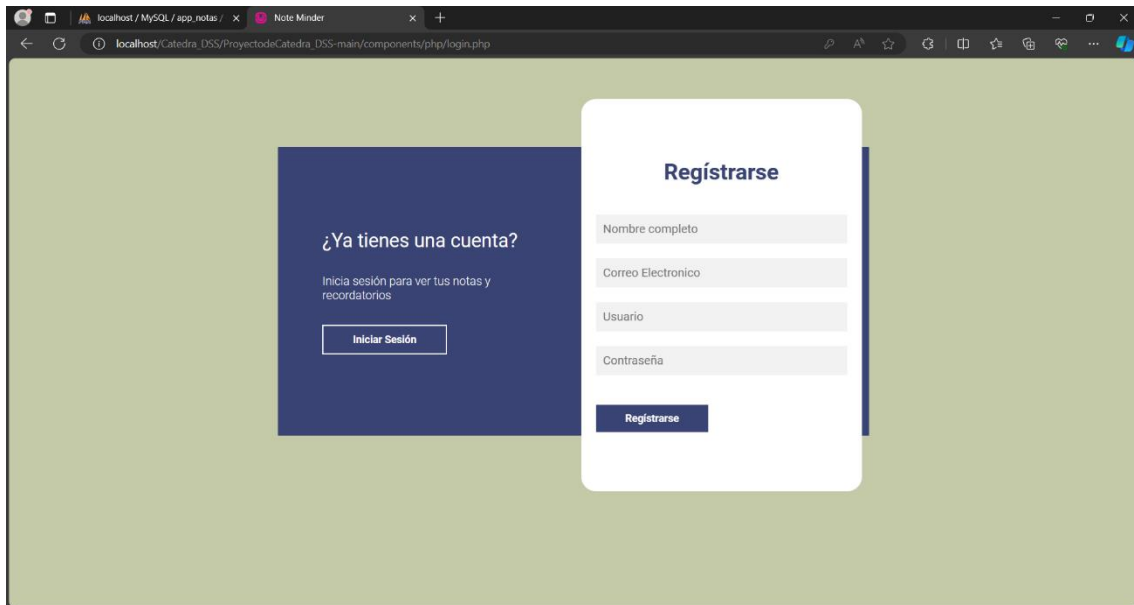
try {
    $conn = new PDO("mysql:host=$servername;dbname=$dbname", $username,
$password);
    $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
    // echo "Conexión exitosa";
} catch(PDOException $e) {
    echo "Error de conexión: " . $e->getMessage();
}

function conexion(){
    $mysqli_conexion = new mysqli("localhost", "root", "", "app_notas");
    if($mysqli_conexion->connect_errno){
        echo "Fallo al conectar a MySQL: (" . $mysqli_conexion-
>connect_errno . ") " . $mysqli_conexion->connect_error;
    }
    return $mysqli_conexion;
}
?>
```

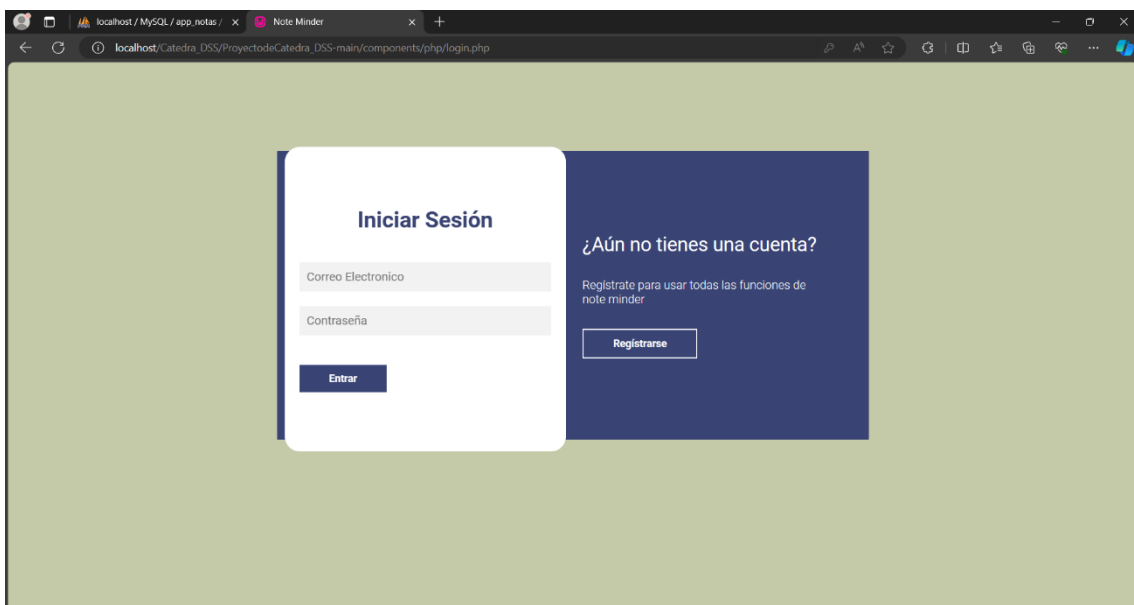
Una vez tenemos una conexión exitosa con la base de datos, podemos proseguir con las siguientes pantallas.

Pantalla 2, Inicio de sesión:

Si el usuario no tiene una cuenta, puede registrarse gratuitamente para tener acceso a todas las funciones de NoteMinder. Para registrarse solamente tiene que llenar todos los campos que aparecen: nombre, correo electrónico, usuario y crear una contraseña, la cual debe tener al menos 8 caracteres, una mayúscula y un número.



Después de crear la cuenta, el usuario puede iniciar sesión ingresando los datos que se piden en los campos del formulario, que son el correo electrónico y la contraseña.



Login: login.php.

En el archivo del login se crearon los contenedores para que almacene las cajas de crear una cuenta e iniciar sesión, también se crearon los formularios para registrarse e ingresar con todos los campos y restricciones:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>Note Minder </title>

  <link
href="https://fonts.googleapis.com/css2?family=Roboto:ital,wght@0,100;0,3
00;0,400;0,500;0,700;0,900;1,100;1,300;1,400;1,500;1,700;1,900&display=sw
ap" rel="stylesheet">

  <link rel="stylesheet" href="../css/estilos.css">
</head>
<body>

  <main>

    <div class="contenedor__todo">
      <div class="caja__trasera">
        <div class="caja__trasera-login">
          <h3>¿Ya tienes una cuenta?</h3>
          <p>Inicia sesión para ver tus notas y
recordatorios</p>
          <button id="btn__iniciar-sesion">Iniciar
Sesión</button>
        </div>
        <div class="caja__trasera-register">
          <h3>¿Aún no tienes una cuenta?</h3>
          <p>Regístrate para usar todas las funciones de
note minder</p>
          <button
id="btn__registrarse">Regístrarse</button>
        </div>
      </div>

      <!--Formulario de Login y registro-->
      <div class="contenedor__login-register">
        <!--Formulario para ingresar-->
```

```

        <form action="iniciarsesion.php" method="POST"
class="formulario__login">
        <h2>Iniciar Sesión</h2>
        <input type="text" placeholder="Correo
Electronico" name="correo" required>
        <input type="password" placeholder="Contraseña"
name="contrasena" required>
        <button>Entrar</button>
        </form>

        <!--Formulario para registrarse-->
        <form action="registrarse.php" method="POST"
class="formulario__register">
        <h2>Regístrarse</h2>
        <input type="text" placeholder="Nombre completo"
name="nombre_completo" required>
        <input type="text" placeholder="Correo
Electronico" name="correo" required>
        <input type="text" placeholder="Usuario"
name="usuario" required>
        <input type="password" placeholder="Contraseña"
name="contrasena" required>
        <button>Regístrarse</button>
        </form>
    </div>
</div>

</main>

    <script src="../../js/script.js"></script>
</body>
</html>

```

Iniciar sesión: iniciarsesion.php.

En este código php se incluye la base de datos, se extraen los datos ingresados por el usuario, se encripta la contraseña, prepara un procedimiento almacenado para verificar las credenciales, vincula los parámetros, ejecuta y obtiene el resultado de los procedimientos almacenados. Luego, redirige según el resultado del procedimiento almacenado, obtiene el ID para guardarlo en la variable de sesión.

```

<?php
// Incluimos la conexión a la base de datos
include 'db_connection.php';

```

```

// Extraemos los datos ingresados por el usuario
$correo = $_POST['correo'];
$contrasena = $_POST['contrasena'];

// Encriptamiento de contraseña
$contrasena = hash('sha512', $contrasena);

// Preparar el procedimiento almacenado "VerificarCredenciales"
$query = "CALL VerificarCredenciales(?, ?, @resultado)";
$stmt = $conn->prepare($query);

// Vincular parámetros
$stmt->bindParam(1, $correo);
$stmt->bindParam(2, $contrasena);

// Ejecutar el procedimiento almacenado
$stmt->execute();

// Obtener el resultado del procedimiento almacenado
$resultado = $conn->query("SELECT @resultado as resultado")-
>fetch(PDO::FETCH_ASSOC);
$resultado_final = $resultado['resultado'];

// Cerrar el statement

// Redirigir según el resultado del procedimiento almacenado
if ($resultado_final == 1) {

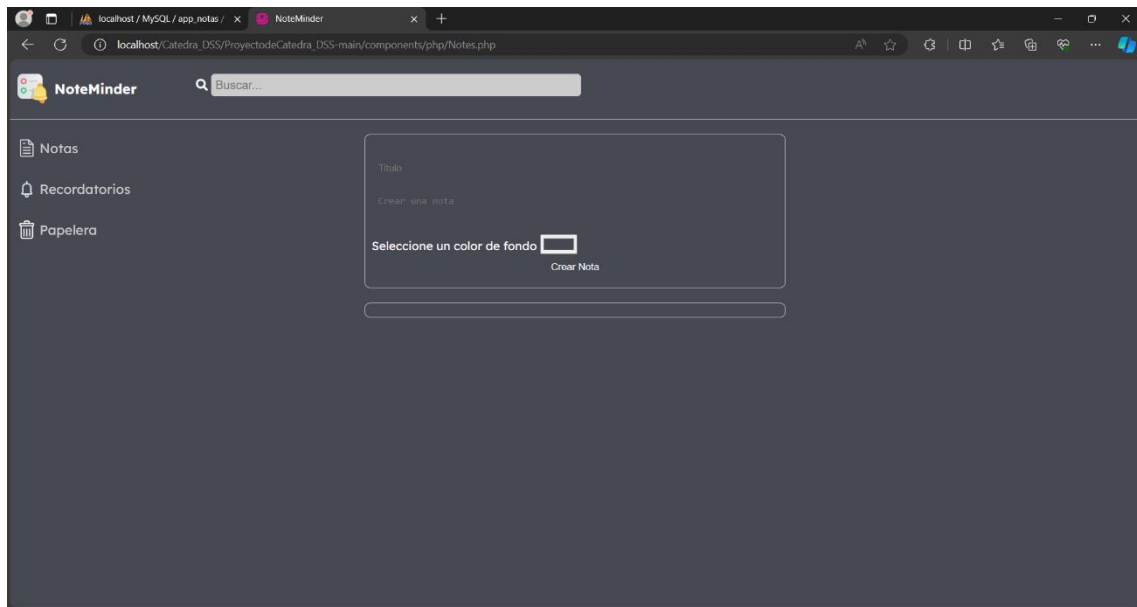
    // Obtener el ID del usuario para guardarlo en la variable de sesión
    $conexion = conexion();
    $ssql = "SELECT `id` FROM `usuarios` WHERE `correo` = '$correo'";
    $result = $conexion->query($ssql);
    session_start();
    if ($result->num_rows > 0) {
        // Usuario autenticado correctamente, obtener su ID
        $row = $result->fetch_assoc();

        // Guardar el ID del usuario en una variable de sesión
        $_SESSION['id'] = $row['id'];
    }

    header("Location: Notes.php");
    exit();
} else {
    echo '
        <script>
            alert("Usuario no encontrado, favor registrarse");
            window.location = "login.php";
        </script>
    
```



Pantalla 3, Notas:



Notes.php.

Este código se encarga de la creación y visualización de notas en la aplicación, presenta el formulario para crear una nota, que incluye campos para el título, el contenido y el color de fondo de la nota.

```
<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>NoteMinder</title>
  <link rel="stylesheet" href="../css/Notes.css">
  <link rel="preconnect" href="https://fonts.googleapis.com">
  <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
  <link
href="https://fonts.googleapis.com/css2?family=Lexend:wght@100..900&displ
ay=swap" rel="stylesheet">
  <link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/5.15.4/css/all.min.css">
</head>
```



```

<body>
  <header class="header">
    <div class="logo">
      
      <h3 class="app-name">NoteMinder</h3>
    </div>
    <div class="search-bar-container">
      <i class="fas fa-search"></i>
      <input type="text" placeholder="Buscar..." class="search-bar"> <!--
Campo de búsqueda -->
    </div>
  </header>
  <div class="formulario-container">
    <p id="agregarNota">Crear una nota...</p> <!-- Texto para agregar una
nota -->
    <form action="../php/crear_nota.php" method="POST" id="formulario"
style="display: none;"> <!-- Formulario para crear una nota -->
      <input type="text" id="titulo" name="titulo" placeholder="Titulo"
required><br> <!-- Campo para el título de la nota -->
      <textarea id="contenido" name="contenido" rows="2" cols="50"
placeholder="Crear una nota" required></textarea><br> <!-- Área para el
contenido de la nota -->
      <label for="color_fondo">Seleccione un color de fondo</label> <!--
Etiqueta para el campo de color de fondo -->
      <input type="color" id="color_fondo" name="color_fondo"
value="#464952" required><br> <!-- Campo para el color de fondo de la
nota -->
      <input type="hidden" name="usuario_id" value="1"> <!-- Campo oculto
para el ID del usuario -->
      <input type="submit" value="Crear Nota"> <!-- Botón para enviar el
formulario y crear la nota -->
    </form>
  </div>

  <div class="formulario-container">
    <?php include '../php/mostrar_nota.php';?> <!-- Contenido de las
notas -->
  </div>

  <nav class="menu">
    <ul>
      <li><a href="Notes.php" id="notas"> Notas</a></li> <!-- Opción del
menú para notas -->
      <li><a href="Reminder.php" id="recordatorio">
Recordatorios</a></li> <!-- Opción del menú para recordatorios -->
    </ul>
  </nav>

```

```

        <li><a href="#" id="papelera">
Papelera</a></li> <!-- Opción del menú para la papelera -->
    </ul>
</nav>

<script src="../js/Notes.js"></script>
</body>
</html>

```

crear_nota.php:

Este código hace que se cree una nueva nota en la base de datos cuando se envía un formulario mediante POST, y maneja adecuadamente errores y redirecciones según el resultado de la operación de la inserción.

```

<?php
session_start();

// Verifica si la solicitud HTTP es de tipo POST
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    // Requiere el archivo de conexión a la base de datos
    require_once "db_connection.php";

    // Recupera los datos del formulario enviado por el método POST
    $titulo = $_POST["titulo"];
    $contenido = $_POST["contenido"];
    $color_fondo = $_POST["color_fondo"];

    // Verifica si la variable de sesión 'id' está definida
    if (isset($_SESSION['id'])) {
        $usuario_id = $_SESSION['id'];
    } else {
        echo "La variable de sesión 'id' no está definida.";
        exit(); // Sale del script si la variable de sesión no está
definida
    }

    try {
        // Prepara la consulta SQL para insertar una nueva nota en la
base de datos
        $stmt = $conn->prepare("INSERT INTO notas (usuario_id, titulo,
contenido, color_fondo) VALUES (:usuario_id, :titulo, :contenido,
:color_fondo)");

```

```

        // Vincula los parámetros de la consulta con las variables
        obtenidas del formulario
        $stmt->bindParam(':usuario_id', $usuario_id);
        $stmt->bindParam(':titulo', $titulo);
        $stmt->bindParam(':contenido', $contenido);
        $stmt->bindParam(':color_fondo', $color_fondo);

        // Ejecuta la consulta SQL
        $stmt->execute();

        // Redirige al usuario a la página Notes.php con un mensaje
        indicando que la nota se ha creado correctamente
        header("Location: Notes.php?nota_creada=true");
        exit();
    } catch (PDOException $e) {
        // Muestra un mensaje de error si la operación de inserción falla
        echo "Error al crear la nota: " . $e->getMessage();
    }
} else {
    // Si la solicitud no es de tipo POST, redirige al usuario a la
    página Notes.php
    header("Location: Notes.php");
    exit();
}

```

mostrar_nota.php:

Este código maneja la visualización, edición y eliminación de notas para el usuario actual en la aplicación.

```

<?php
session_start();
if (isset($_SESSION['id'])) {
    $usuarioId = $_SESSION['id'];
} else {
    echo "La variable de sesión no está definida.";
}

require_once "db_connection.php";

$conexion = conexion(); // Conexión a la base de datos

// Consulta SQL para mostrar las notas del usuario
$sql = "SELECT `id`, `titulo`, `contenido`, `color_fondo` FROM `notas`
WHERE `usuario_id` = '$usuarioId'";
$result = $conexion->query($sql);
echo "<link rel='stylesheet' href='../css/botonEliminar.css'>";

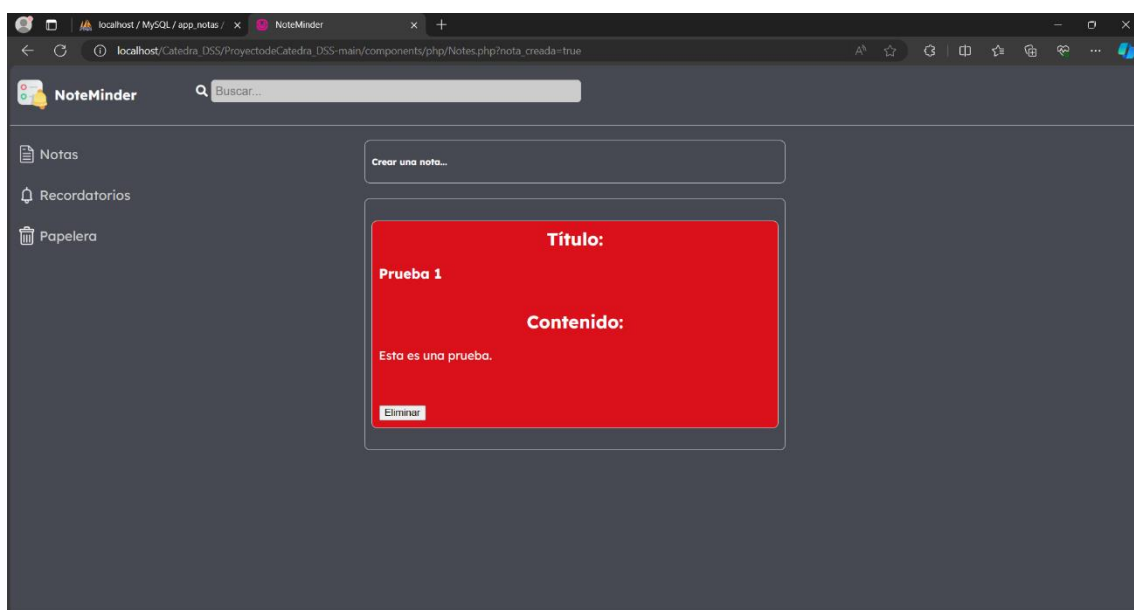
```



```

        // Actualizar la página después de eliminar la nota
        location.reload();
    }
};
xhr.open("POST", "eliminar_nota.php", true);
xhr.setRequestHeader("Content-Type", "application/x-www-form-
urlencoded");
xhr.send("idNota=" + idNota);
}
}
</script>

```



Update_note.php:

Permite al usuario actualizar el título y el contenido de una nota existente en la base de datos y luego redirige al usuario a la página de notas.

```

<?php
session_start();
if(isset($_SESSION['id'])) {
    $usuarioId = $_SESSION['id'];
} else {
    echo "La variable de sesión no está definida.";
}
$title = $_POST["titulo"];
$content = $_POST["contenido"];
$idnota = $_POST["idnota"];

require_once "db_connection.php";

```

```

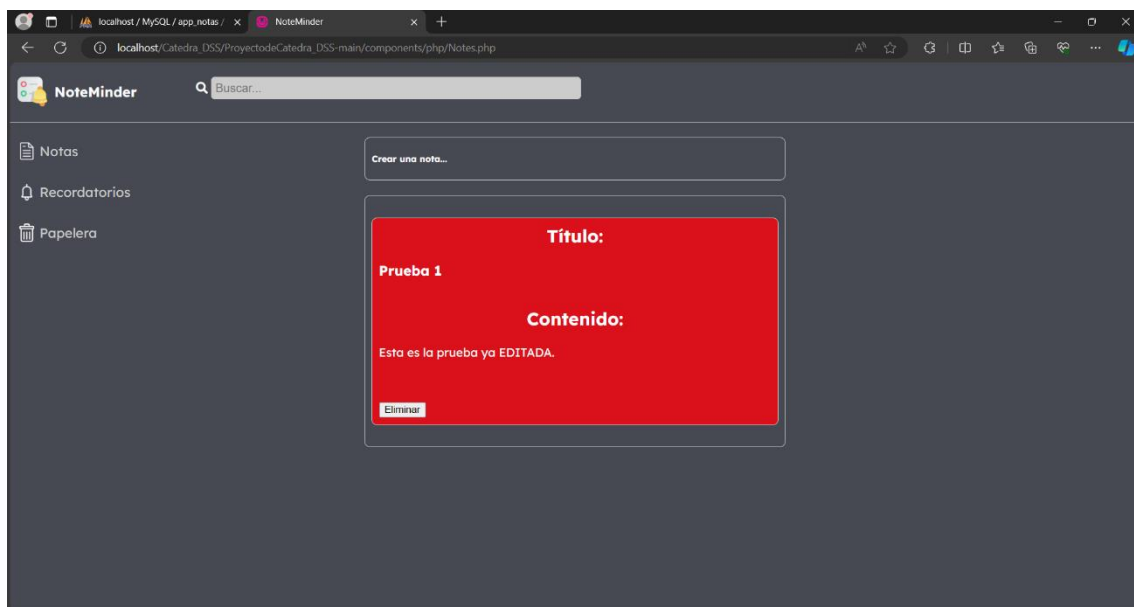
    $conexion = conexion(); // Conexión a la base de datos
    // Consulta SQL para mostrar las notas del usuario
    $ssql = "UPDATE notas SET titulo = '$titulo', contenido = '$contenido' WHERE `id` = '$idnota'";
    $result = $conexion->query($ssql); //Ejecutar consulta

    header("Location: Notes.php");

?>

```

Las notas ya editadas se ven de la siguiente manera:



Eliminar_notas.php:

Este código maneja la eliminación segura de notas de la base de datos, asegurándose de que solo se eliminen las notas del usuario actual y proporcionando mensajes claros sobre el resultado de la operación.

```

<?php
session_start();
if (isset($_SESSION['id'])) {
    $usuarioId = $_SESSION['id'];
} else {
    echo "La variable de sesión no está definida.";
    exit;
}

```

```

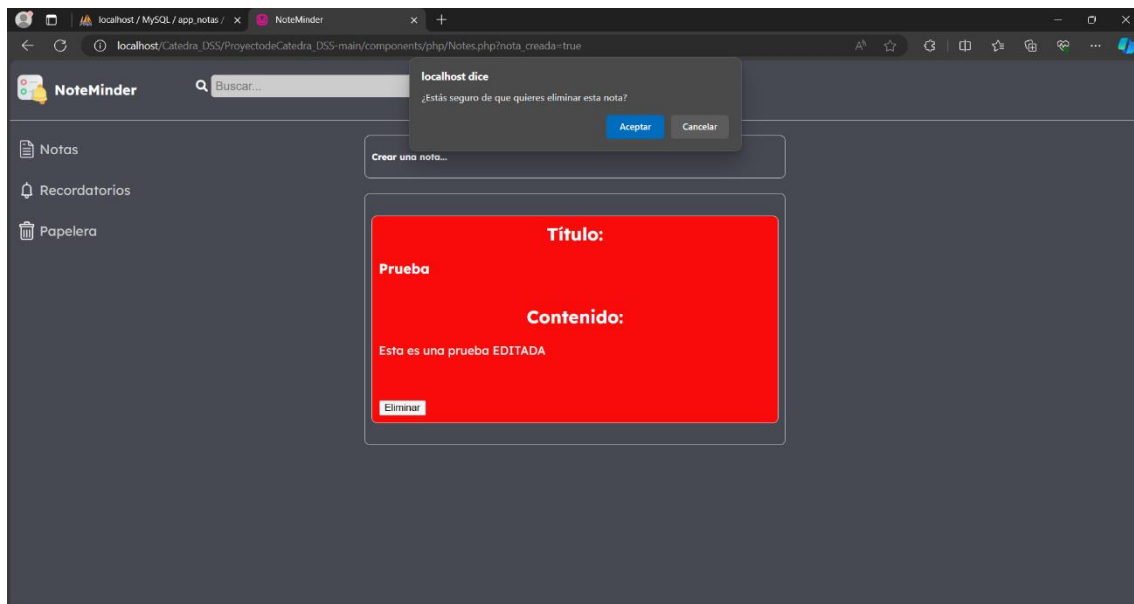
if (isset($_POST['idNota'])) {
    $idNota = $_POST['idNota'];

    require_once "db_connection.php";
    $conexion = conexion(); // Conexión a la base de datos

    $ssql = "DELETE FROM notas WHERE id = '$idNota' AND usuario_id = '$usuarioId'";
    $resultado = $conexion->query($ssql);

    if ($resultado) {
        echo "La nota se eliminó correctamente.";
    } else {
        echo "Error al eliminar la nota.";
    }
} else {
    echo "No se proporcionó un ID de nota válido.";
}

```



Reminder.php.

Este código proporciona una interfaz para que los usuarios puedan crear, ver y gestionar recordatorios en la aplicación.

```

<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>NoteMinder</title>
  <link rel="stylesheet" href="../css/Notes.css">
  <link rel="stylesheet" href="../css/Reminder.css">
  <link rel="preconnect" href="https://fonts.googleapis.com">
  <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
  <link
href="https://fonts.googleapis.com/css2?family=Lexend:wght@100..900&displ
ay=swap" rel="stylesheet">
  <link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/5.15.4/css/all.min.css">
</head>
<body>
<header class="header">
  <div class="logo">
    
    <h3 class="app-name">NoteMinder</h3>
  </div>
  <div class="search-bar-container">
    <i class="fas fa-search"></i>
    <input type="text" placeholder="Buscar..." class="search-bar"> <!--
Campo de búsqueda -->
  </div>
  <div class="cerrar-sesion-container">
    <a href="../../inicio.html" class="cerrar-sesion-btn">Cerrar
sesión</a>
  </div>
</header>
<?php
date_default_timezone_set('America/Mexico_City');
// Obtiene la hora actual
$hora_actual = date('H:i');
$fecha_actual = date('Y-m-d');
// Suma 30 minutos a la hora actual
$nueva_hora = date('H:i', strtotime($hora_actual . ' + 30 minutes'));
?>
<div class="formulario-container">
  <p id="agregarRecordatorio">Crear Recordatorio...</p> <!-- Texto para
agregar un recordatorio -->
  <form action="../../php/crear_recordatorio.php" method="POST"
id="formularioRecordatorio" style="display: none;">
    <input type="text" name="titulo" class="input-titulo"
placeholder="Título del recordatorio" required><br>

```



```

        <textarea name="contenido" class="input-contenido"
placeholder="Contenido del recordatorio" required></textarea><br>

        <label for="color_fondo">Seleccione un color de fondo</label> <!--
- Etiqueta para el campo de color de fondo -->
        <input type="color" id="color_fondo" name="color_fondo"
value="#464952" required><br><br> <!-- Campo para el color de fondo de la
nota -->

        <input type="date" id="date" name="fecha_recordatorio"
class="input-fecha" min="<?php echo $fecha_actual ?>" value="<?php echo
$fecha_actual; ?>" required /><br>
        <input type="time" id="time" name="hora_recordatorio"
class="input-fecha" value="<?php echo $hora_actual; ?>" required /><br>

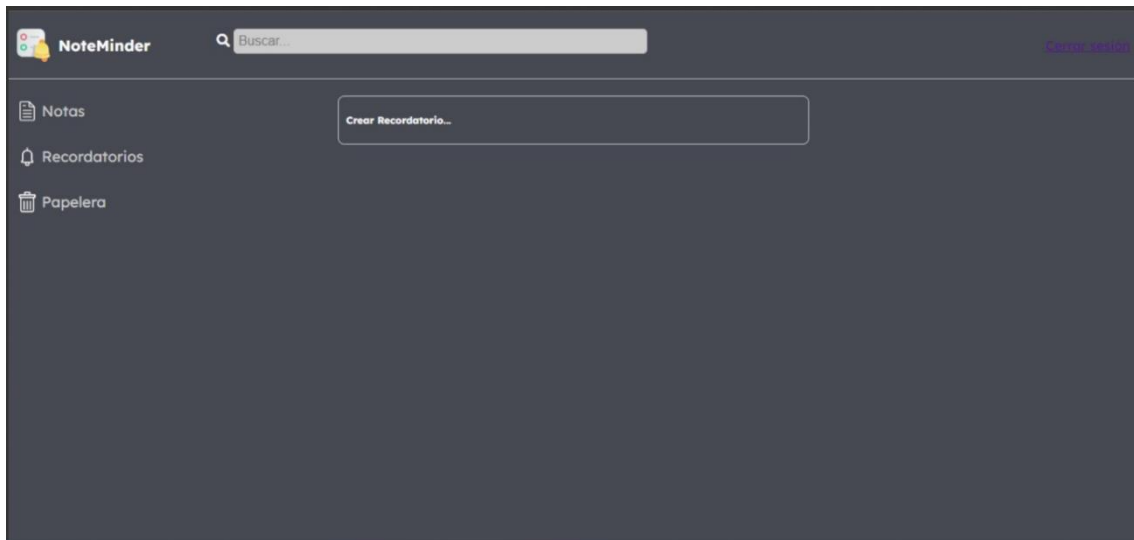
        <input type="submit" class="btn-crear" value="Crear
Recordatorio">
    </form>
</div>

<div class="recordatorios-container">
    <?php include '../php/mostrar_recordatorio.php'; ?>
</div>

<nav class="menu">
    <ul>
        <li><a href="Notes.php" id="notas"> Notas</a></li> <!-- Opción del
menú para notas -->
        <li><a href="Reminder.php" id="recordatorio">
Recordatorios</a></li> <!-- Opción del menú para recordatorios -->
        <li><a href="#" id="papelera">
Papelera</a></li> <!-- Opción del menú para la papelera -->
    </ul>
</nav>

    <script src="../js/recordatorios.js"></script>
</body>
</html>

```



Crear_recordatorio.php:

Este código php hace que se cree un nuevo recordatorio en la base de datos cuando se envía un formulario mediante POST, y maneja adecuadamente los errores y redirecciones según el resultado de la operación de inserción.

```
<?php
session_start();

// Verifica si la solicitud HTTP es de tipo POST
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    // Requiere el archivo de conexión a la base de datos
    require_once "db_connection.php";

    // Recupera los datos del formulario enviado por el método POST
    $nota_id = $_POST["nota_id"];
    $titulo = $_POST["titulo"];
    $contenido = $_POST["contenido"];
    $color_fondo = $_POST["color_fondo"];
    $fecha_recordatorio = $_POST["fecha_recordatorio"];
    $hora_recordatorio = $_POST["hora_recordatorio"];

    if (isset($_SESSION['id'])) {
        $usuario_id = $_SESSION['id'];
    } else {
        echo "La variable de sesión 'id_usuario' no está definida.";
        exit();
    }

    try {
        // Prepara la consulta SQL para insertar un nuevo recordatorio en
        la base de datos
```

```

        $stmt = $conn->prepare("INSERT INTO recordatorios (usuario_id,
nota_id, titulo, contenido, fecha_recordatorio, hora_recordatorio,
color_fondo) VALUES (:usuario_id, :nota_id, :titulo, :contenido,
:fecha_recordatorio, :hora_recordatorio, :color_fondo)");

        // Vincula los parámetros de la consulta con las variables
obtenidas del formulario
        $stmt->bindParam(':usuario_id', $usuario_id);
        $stmt->bindParam(':nota_id', $nota_id);
        $stmt->bindParam(':titulo', $titulo);
        $stmt->bindParam(':contenido', $contenido);
        $stmt->bindParam(':color_fondo', $color_fondo);
        $stmt->bindParam(':fecha_recordatorio', $fecha_recordatorio);
        $stmt->bindParam(':hora_recordatorio', $hora_recordatorio);

        // Ejecuta la consulta SQL
        $stmt->execute();

        // Redirige al usuario a la página Notes.php con un mensaje
indicando que el recordatorio se ha creado correctamente
        header("Location: Reminder.php?recordatorio_creado=true");
        exit();
    } catch (PDOException $e) {
        // Muestra un mensaje de error si la operación de inserción falla
        echo "Error al crear el recordatorio: " . $e->getMessage();
    }
} else {
    // Si la solicitud no es de tipo POST, redirige al usuario a la
página Notes.php
    header("");
    exit();
}
?>

```

Mostrar_recordatorio.php:

Muestra los recordatorios del usuario y proporciona una funcionalidad de notificación para alertar al usuario cuando llega el momento de un recordatorio. También permite eliminar los recordatorios.

```

<?php
session_start();
if (isset($_SESSION['id'])) {
    $usuarioId = $_SESSION['id'];
} else {
    echo "La variable de sesión no está definida.";
}

```

```

}

require_once "db_connection.php";

$conexion = conexion(); // Conexión a la base de datos

// Consulta SQL para mostrar las notas del usuario
$sql = "SELECT `id`, `titulo`, `contenido`,
`fecha_recordatorio`, `hora_recordatorio`, `color_fondo` FROM
`recordatorios` WHERE `usuario_id` = '$usuarioId'";
$result = $conexion->query($sql);

echo "<link rel='stylesheet' href='../css/reminder2.css'> ";
echo "<link rel='stylesheet' href='../css/botonEliminar.css'> ";
// Mostrar los resultados
while ($row = $result->fetch_array()) {
    $hora_alerta = $row['hora_recordatorio'];
    $fecha_alerta = $row['fecha_recordatorio'];
    $titulo = $row['titulo'];

    echo "<article class='notaMostrada' id='" . $row['id'] . "'
style='background-color: " . $row['color_fondo'] . "'>
    <h2>Título:</h2> <h3>" . $row['titulo'] . "</h3><br>
    <h2>Contenido:</h2> <p>" . nl2br($row['contenido']) . "</p><br>
    <div class='container'>
        <div class='fecha'>
            <h3>Fecha:</h3>
            <p>" . nl2br($row['fecha_recordatorio']) . "</p>
        </div>
        <div class='hora'>
            <h3>Hora:</h3>
            <p>" . nl2br($row['hora_recordatorio']) . "</p>
        </div>

        <div style='text-align: center;'>
            <button class='button'
onclick='enviarAPapelera(\".$row['id'].")'>Eliminar</button>
        </div>
    </div>

    </article>";

    // Utilizar javascript para mostrar la notificación
    echo "<script>
function checkHora() {
    var titulo = '$titulo';
    var fechaAlerta = '$fecha_alerta';
    var horaAlerta = '$hora_alerta';

```

```

        var fechaActual = new Date().toLocaleDateString('en-US', { year:
'numeric', month: '2-digit', day: '2-digit' });
        fechaActual = fechaActual.split('/').reverse().join('-'); //
Formatea la fecha como YYYY-MM-DD
        var fechaActualReordenada = fechaActual.split('-')[0] + '-' +
fechaActual.split('-')[2] + '-' + fechaActual.split('-')[1]; // Reordena
la fecha actual como YYYY-DD-MM
        var horaActual = new Date().toLocaleTimeString('en-US', {hour12:
false});

        console.log('Esperando la fecha de: ' + titulo);
        console.log('Fecha actual: ' + fechaActualReordenada);
        console.log('Fecha alerta: ' + fechaAlerta);
        console.log('Hora actual: ' + horaActual);

        if(fechaActualReordenada === fechaAlerta) {
            if (horaActual === horaAlerta) {
                console.clear();
                toggleNotification();
            }
        } else {console.log('La fecha no coincide');}
    }
    function clear () {console.clear();}
    setInterval(checkHora, 1000);
    setInterval(clear, 60000);
</script>";

// Componente oculto de la notificacion
echo "<figure class='notification hide none' style='display: flex;
justify-content: center; align-items: center; position: fixed; top: 50%;
left: 50%; transform: translate(-50%, -50%); z-index: 9999;'"
<div class='notification__body'>
    <div class='notification__body__first'>
        <svg focusable='false' viewBox='0 0 24 24'>
            <path d='M20,12A8,8 0 0,1 12,20A8,8 0 0,1 4,12A8,
            8 0 0,1 12,4C12.76,4 13.5,4.11 14.2,
            4.31L15.77,2.74C14.61,2.26 13.34,2
            12,2A10,10 0 0,0 2,12A10,10 0 0,0
            12,22A10,10 0 0, 0 22,12M7.91,10.08L6.5,11.5L11,
            16L21,6L19.59,4.58L11,13.17L7.91,10.08Z'
            >>/path>
        </svg>
        <h3>
            Es hora del recordatorio: '$titulo' &#128640; <br>
            que estaba programado el <b>'$fecha_alerta'</b> a las
            <b>'$hora_alerta'</b>
        </h3>
        <p>
        </p>
    </div>
    </div>
    </figure>";

```

```

    </div>
    <svg
        onclick='toggleNotification()'
        xmlns='http://www.w3.org/2000/svg'
        width='24'
        height='24'
        viewBox='0 0 24 24'
        stroke-width='2'
        stroke='currentColor'
        fill='none'
        stroke-linecap='round'
        stroke-linejoin='round'
    >
        <path stroke='none' d='M0 0h24v24H0z' fill='none'></path>
        <path d='M18 6l-12 12'></path>
        <path d='M6 6l12 12'></path>
    </svg>
</div>
</figure>";
/*
echo "<div id='myModal' class='modal'>";
echo "<div class='modal-content'>";
echo "<span class='close'>&times;</span>";
// Iniciar el formulario
echo "<form action='../php/update_note.php' method='post'>";

    echo "<input type='text' id='idnota' name='idnota' value='" .
htmlspecialchars($row['id']) . "' style='display: none;'><br>";
    // Campo de entrada para el título
    echo "<label for='titulo'>Título:</label>";
    echo "<input type='text' id='titulo' name='titulo' value='" .
htmlspecialchars($row['titulo']) . "'><br>";
    // Campo de entrada para el contenido
    echo "<label for='contenido'>Contenido:</label>";
    echo "<textarea id='contenido' name='contenido'>" .
htmlspecialchars($row['contenido']) . "</textarea><br>";
    // Botón de envío del formulario
    echo "<input type='submit' value='Guardar'>";
    echo "</form>"; // Cerrar el formulario
    echo "</div>";
    echo "</div>";*/
}

?>

<script> /*Utilizamos javascript para mostrar una alert
        para confirmar la eliminación de la nota */

```

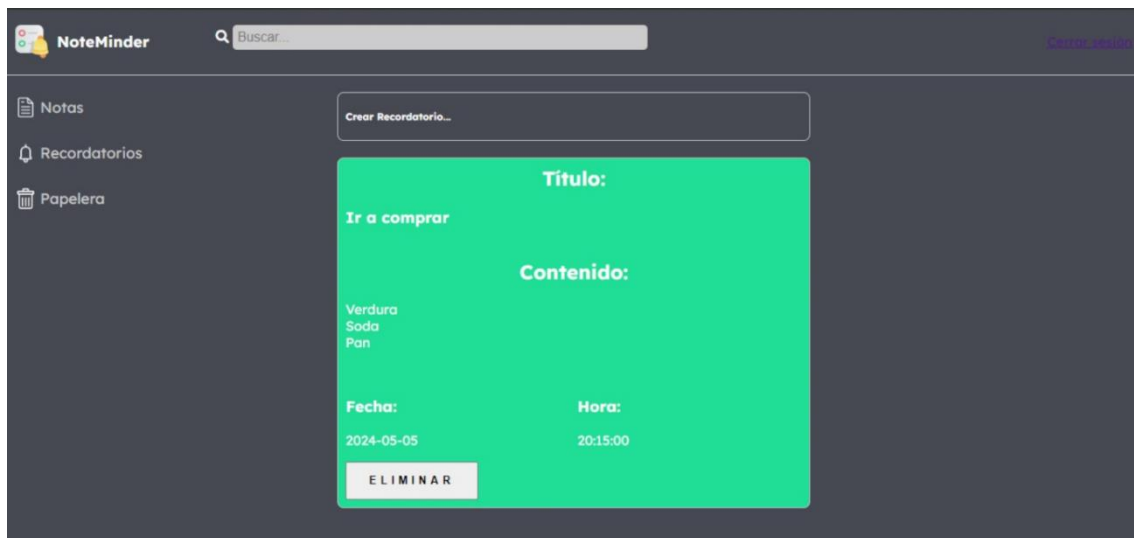
```

function enviarAPapelera(idNota) {
  if (confirm("¿Estás seguro de que quieres eliminar esta nota?")) {
    var xhr = new XMLHttpRequest();
    xhr.onreadystatechange = function() {
      if (xhr.readyState == 4 && xhr.status == 200) {
        // Actualizar la página después de eliminar la nota
        location.reload();
      }
    };
    xhr.open("POST", "eliminar_recordatorio.php", true);
    xhr.setRequestHeader("Content-Type", "application/x-www-form-urlencoded");
    xhr.send("nota_id=" + idNota);
  }
}

</script>

<script src="../js/scriptNotificacion.js"></script>

```



Update_reminder.php:

Este código verifica si el usuario ha iniciado sesión, luego obtiene los datos del formulario de actualización del recordatorio mediante POST, se establece una conexión con la base de datos y se construye y ejecuta una consulta SQL para actualizar el título y contenido del recordatorio en la base de datos. Después de la actualización, el usuario es redirigido a la página de recordatorios.

```

<?php
session_start();

// Verifica si la variable de sesión 'id' está definida
if(isset($_SESSION['id'])) {
    $usuarioId = $_SESSION['id']; // Obtiene el ID del usuario de la
    sesión
} else {
    echo "La variable de sesión 'id' no está definida."; // Muestra un
    mensaje si la variable de sesión no está definida
}

// Obtiene los valores del formulario de actualización de recordatorio
$titulo = $_POST["titulo"];
$contenido = $_POST["contenido"];
$idRecordatorio = $_POST["idRecordatorio"];

// Incluye el archivo de conexión a la base de datos
require_once "db_connection.php";

// Establece la conexión con la base de datos
$conexion = conexion();

// Construye la consulta SQL para actualizar el recordatorio en la base
de datos
$sql = "UPDATE recordatorios SET titulo = '$titulo', contenido =
'$contenido' WHERE id = '$idRecordatorio' AND usuario_id = '$usuarioId'";

// Ejecuta la consulta SQL para actualizar el recordatorio
$resultado = $conexion->query($sql);

// Redirige al usuario a la página de recordatorios después de actualizar
el recordatorio en la base de datos
header("Location: Reminder.php");
?>

```

Eliminar_recordatorios.php.

Verifica si el usuario ha iniciado sesión, comprueba si se ha enviado un ID de nota mediante el método POST, si se proporciona un ID de nota, se procede a eliminar el recordatorio correspondiente. Luego hace una conexión a la base de datos y construye y ejecuta una consulta SQL para eliminar el recordatorio de la base de datos, verificando que el ID del recordatorio coincida con el solicitado.


```

<?php
session_start();
if (isset($_SESSION['id'])) {
    $usuarioId = $_SESSION['id'];
} else {
    echo "La variable de sesión no está definida.";
    exit;
}

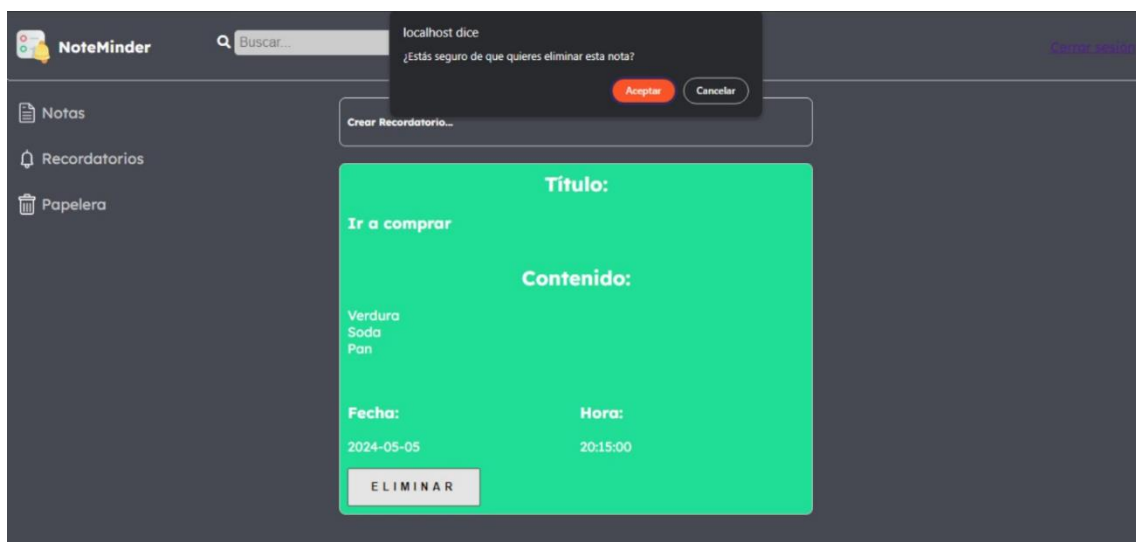
if (isset($_POST['nota_id'])) {
    $idRecordatorio = $_POST['nota_id'];

    require_once "db_connection.php";
    $conexion = conexion(); // Conexión a la base de datos

    $ssql = "DELETE FROM recordatorios WHERE id = '$idRecordatorio' AND
usuario_id = '$usuarioId'";
    $resultado = $conexion->query($ssql);

    if ($resultado) {
        echo "La nota se eliminó correctamente.";
    } else {
        echo "Error al eliminar la nota.";
    }
} else {
    echo "No se proporcionó un ID de nota válido.";
}
?>

```



La alerta de los recordatorios se muestra de la siguiente manera:

