



# salesianos

## PORTAL EDUCATIVO



**EDUCADOS**

Trabajo de Fin de Ciclo ASIR  
Ciclo Formativo de Grado Superior  
Salesianos Atocha  
Tutor: Jose Manuel Prieto Gordo

**Sergio Acosta Guijarro**  
**JUNIO DE 2023**

# Índice

<b>INTRODUCCIÓN.....</b>	<b>3</b>
DESCRIPCIÓN GENERAL DEL PROYECTO Y SU PROPÓSITO.....	3
OBJETIVOS Y METAS.....	3
PLANIFICACIÓN.....	3
<b>ARQUITECTURA DE LA BASE DE DATOS.....</b>	<b>4</b>
DESCRIPCIÓN DE LA BASE DE DATOS.....	4
EXPLICACIÓN DE LA ESTRUCTURA.....	4
MODELO ENTIDAD-RELACIÓN.....	9
<b>DISEÑO Y FUNCIONALIDAD WEB.....</b>	<b>10</b>
EXPLICACIÓN DE LA PÁGINA WEB.....	10
FUNCIONALIDAD.....	19
DISEÑO.....	22
<b>ARQUITECTURA DE DOCKER.....</b>	<b>23</b>
EXPLICACIÓN DE DOCKER.....	23
MANUAL DOCKER - GITHUB.....	23
FUNCIONAMIENTO.....	24
<b>RESULTADOS.....</b>	<b>26</b>
CRONOLOGÍA.....	26
CONCLUSIÓN.....	28
<b>POSIBLES AMPLIACIONES.....</b>	<b>29</b>
<b>WEBGRAFÍA.....</b>	<b>30</b>

## **INTRODUCCIÓN**

### **DESCRIPCIÓN GENERAL DEL PROYECTO Y SU PROPÓSITO**

Este proyecto consiste en la creación de Educados, un ecosistema educativo digital diseñado específicamente para colegios con el propósito de conectar a los profesores, alumnos y padres. El objetivo es facilitar la comunicación entre estos y monitorizar el rendimiento de los alumnos. Por otro lado, los profesores tendrán una interfaz para la facilitación de administrar las calificaciones y entrega de boletines.

### **OBJETIVOS Y METAS**

El principal objetivo es mejorar la experiencia educativa mediante la implementación de un entorno web accesible desde cualquier lugar, además, se aspira a optimizar la gestión de los recursos brindando un servicio educativo de calidad.

Para lograr esto, se han establecido las siguientes metas:

- ❖ Crear una interfaz amigable, fácil de usar y agradable para los usuarios.
- ❖ Garantizar la seguridad y privacidad de los datos en todo momento.
- ❖ Ofrecer una disponibilidad del portal en múltiples servicios, permitiendo el acceso en cualquier lugar y momento.

### **PLANIFICACIÓN**

La planificación del proyecto consta de cuatro procesos. En primer lugar, se desarrollará la idea y se definirán las necesidades del proyecto, como el nombre, logo, diseño y las funciones que tendrá la plataforma. Luego, se creará la base de datos y se asegurará su seguridad y automatización. A continuación, se creará la página web y su interfaz. Y por último, se configurará el servidor Linux con los servicios contenerizados necesarios para el funcionamiento del portal educativo.

## ARQUITECTURA DE LA BASE DE DATOS

La arquitectura de la base de datos es un elemento fundamental para el correcto funcionamiento del proyecto. Su función es almacenar y gestionar la información relacionada con las tablas. Una base de datos bien desarrollada garantiza un soporte sólido y eficiente para todas las funcionalidades del sistema.

## DESCRIPCIÓN DE LA BASE DE DATOS

La base de datos de Educados es el soporte necesario para almacenar y gestionar la información relacionada con los usuarios, grupos y materias, así como de las calificaciones. Su desarrollo es clave para el correcto funcionamiento del proyecto.

La base de datos se basa en un modelo relacional, es decir los datos se organizan en las tablas relacionadas entre sí. Esta elección se debe a la flexibilidad y capacidad de gestión entre los datos. La base de datos almacena información clave sobre los usuarios, como nombre, dirección de correo electrónico, contraseña, así como detalles relacionados con los cursos, como el nombre del curso, las materias y las calificaciones.

## EXPLICACIÓN DE LA ESTRUCTURA

La base de datos comienza con la tabla usuarios, en la tabla usuarios se almacenan los datos de todos los usuarios.

Field	Type	Null	Key	Default	Extra
id_usuario	int(4)	NO	PRI	NULL	auto_increment
nombre	varchar(50)	YES		NULL	
correo	varchar(100)	NO	UNI	NULL	
contraseña	varchar(100)	NO		NULL	
tipo_usuario	varchar(20)	NO		NULL	

El id\_usuario es de tipo numérico, es el **primary key** de la tabla y es de tipo **auto\_increment**. También se almacena el **nombre**, el **correo electrónico** (de tipo **UNI**Q para evitar repeticiones), la **contraseña** (almacenada de forma encriptada) y el **tipo\_usuario** en el que existen cuatro tipos:

### ❖ Administrador

No tiene tabla propia, el administrador es el único usuario con permisos de modificación en la tabla de inserción, actualización y de borrado, entre sus características se encuentran la capacidad de agregar usuarios, materias y cursos así como borrarlos y actualizarlos.

### ❖ Profesor

El profesor tiene una función importante en el proyecto Educados. Su labor principal es la inserción de notas en sus materias y la administración de su curso. El profesor tiene la función de inserción de notas en sus materias y en la administración de su curso.

Field	Type	Null	Key	Default	Extra
id_usuario	int(4)	NO	PRI	NULL	auto_increment
nombre	varchar(50)	YES		NULL	
correo	varchar(100)	NO	UNI	NULL	
contraseña	varchar(100)	NO		NULL	

### ❖ Padre

La función del padre es más sencilla en comparación con la del administrador y el profesor. Su labor principal es la de ver la información de su hijo, como las calificaciones y el progreso académico.

Field	Type	Null	Key	Default	Extra
id_usuario	int(4)	NO	PRI	NULL	auto_increment
nombre	varchar(50)	YES		NULL	
correo	varchar(100)	NO	UNI	NULL	
contraseña	varchar(100)	NO		NULL	
id_hijo	int(4)	YES	MUL	NULL	
hijo	varchar(50)	YES		NULL	
primer_login	tinyint(1)	YES		1	

En la propia tabla se guardan los datos del hijo, que se relacionan mediante un trigger que automatiza este paso, funciona mediante la columna "**primer\_login**" que indica si el padre ha iniciado sesión por primera vez. Si este valor es 1 o True, el sistema dirigirá al padre a un formulario donde podrá ingresar los datos de su hijo. Una vez que el padre envíe el formulario, el trigger modificará el valor de "**primer\_login**" a 0 o False y rellenará las columnas "**id\_hijo**" e "**hijo**".

## ❖ Alumno

El alumno o hijo es el usuario más importante, a partir de él se crean todas las relaciones con las tablas.

Field	Type	Null	Key	Default	Extra
id_usuario	int(4)	NO	PRI	NULL	auto_increment
nombre	varchar(50)	YES		NULL	
correo	varchar(100)	NO	UNI	NULL	
contraseña	varchar(100)	NO		NULL	
id_curso	int(4)	YES	MUL	NULL	
curso	varchar(50)	YES	MUL	NULL	
primer_login	tinyint(1)	YES		1	

Al igual que en la tabla padre, tiene una tabla "**primer\_login**", solo que en este caso, le envía a un formulario en el que se muestra un select option, conectado a la base de datos que muestra todos los cursos existentes. El alumno elige el curso y al enviar ese formulario hay otro trigger que modifica el "**primer\_login**" y rellena las columnas **id\_curso** y **curso**.

Estas cuatro tablas (usuario, padre, alumno y profesor) son la base de la estructura, a partir de estos se crean tres tablas más:

## ❖ Curso

En la tabla curso, se almacena el **id** del curso, el nombre del **curso** y el **tutor**. Es la tabla más sencilla de todas.

Field	Type	Null	Key	Default	Extra
id_curso	int(4)	NO	PRI	NULL	auto_increment
curso	varchar(50)	YES	UNI	NULL	
tutor	varchar(50)	YES		NULL	

## ❖ Materia

La tabla materia es muy importante, en ella se recogen los datos de las materias, así como los del profesor de cada materia y el curso al que pertenece, así como de sus id 's.

Field	Type	Null	Key	Default	Extra
id_materia	int(4)	NO	PRI	NULL	auto_increment
materia	varchar(50)	YES		NULL	
id_profesor	int(4)	YES	MUL	NULL	
profesor	varchar(50)	YES		NULL	
id_curso	int(4)	YES	MUL	NULL	
curso	varchar(50)	YES		NULL	

Esta tabla está automatizada mediante triggers donde las columnas de cada **materia** y **curso** están automatizadas.

## ❖ Calificaciones

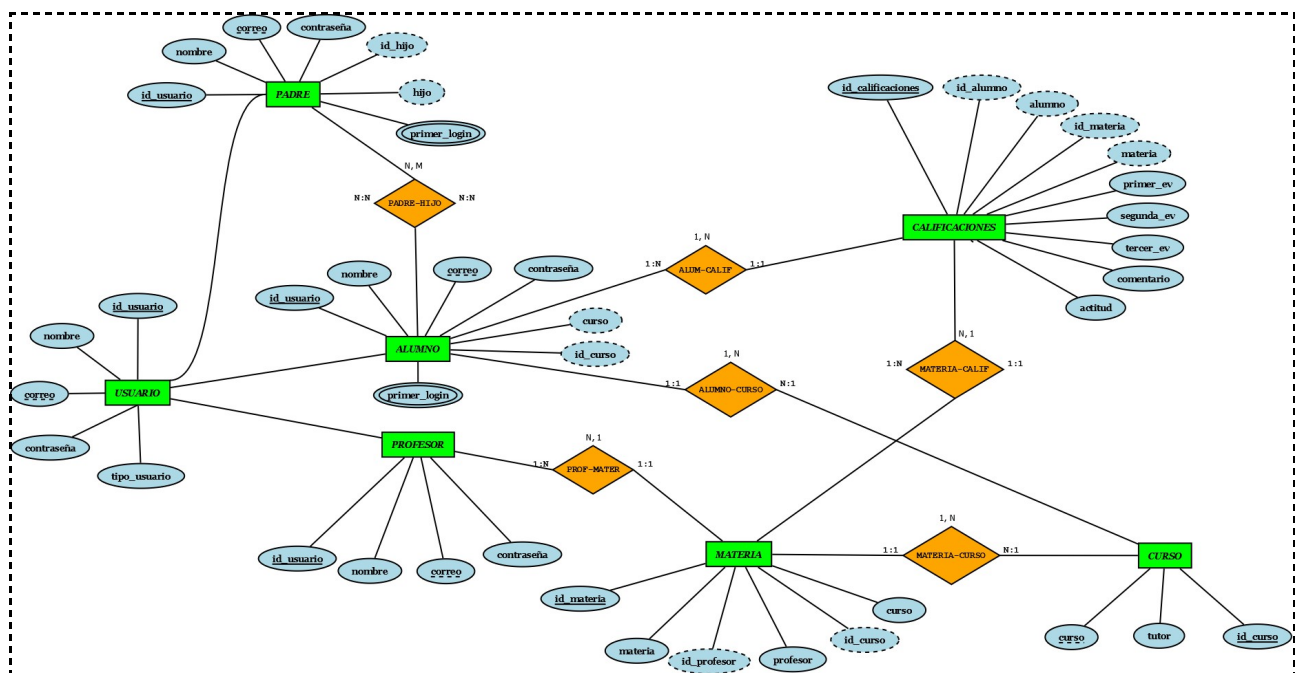
La tabla de calificaciones es la más compleja, larga y automatizada de todas. En esta tabla se almacenan los datos de las notas de los alumnos en cada materia, y los usuarios no pueden crear ningún registro en ella, excepto las notas.

Field	Type	Null	Key	Default	Extra
id_calificaciones	int(11)	NO	PRI	NULL	auto_increment
id_alumno	int(4)	YES	MUL	NULL	
alumno	varchar(50)	YES		NULL	
id_materia	int(4)	YES	MUL	NULL	
materia	varchar(50)	YES		NULL	
primer_ev	int(2)	YES		NULL	
segunda_ev	int(2)	YES		NULL	
tercer_ev	int(2)	YES		NULL	
actitud	int(2)	YES		NULL	
comentario	varchar(100)	YES		NULL	
nota_media	float	YES		NULL	STORED GENERATED

Existen varios triggers que almacenan la información de los alumnos y de las materias. La columna **nota\_media** es de tipo **númeroico**, y es un **STORED GENERATED**, esto significa que es una tabla que recoge las notas de las tres evaluaciones y las divide entre tres, esta nota media no es funcional al 100%, hasta que las tres evaluaciones tengan nota.

```
1 ALTER TABLE calificaciones ADD COLUMN nota_media
2 FLOAT AS ((primer_ev + segunda_ev + tercer_ev) / 3) STORED;
```

## MODELO ENTIDAD-RELACIÓN



La base de datos cuenta con **14 tablas**, de las cuales **7** son **relaciones**, entre ellas existen triggers de **inserción**, **actualización** y **borrado**, para la completa automatización de esta, está pensada para la completa facilitación y simplificación de las tablas así como de su seguridad. El usuario administrador tiene la responsabilidad de crear los datos relacionados con las materias, los cursos y las calificaciones, así como de crear los usuarios con el rol de profesor. Es importante destacar que una vez que se ha registrado un alumno en la tabla de calificaciones, no se pueden borrar los datos relacionados con él, como su curso o materia. Esto se hace para mantener la integridad de la información. Por esta razón, desde la interfaz de la página web no se permite la modificación de esta tabla. Las notas de los alumnos son introducidas por los profesores de cada asignatura, no por el profesor del curso en su conjunto. Esto permite que cada profesor introduzca las calificaciones correspondientes a su asignatura, garantizando así la exactitud de los registros.

Es importante mencionar que no existen relaciones directas entre las tablas de usuarios. Esto se debe a que la información de estas tablas es redundante, lo que significa que los datos se repiten en varias tablas.



## DISEÑO Y FUNCIONALIDAD WEB

### EXPLICACIÓN DE LA PÁGINA WEB

La web de Educados está diseñada de forma intuitiva y funcional. Al entrar, nos encontramos con una página dividida en tres secciones principales: la cabecera, el cuerpo y el pie de página.



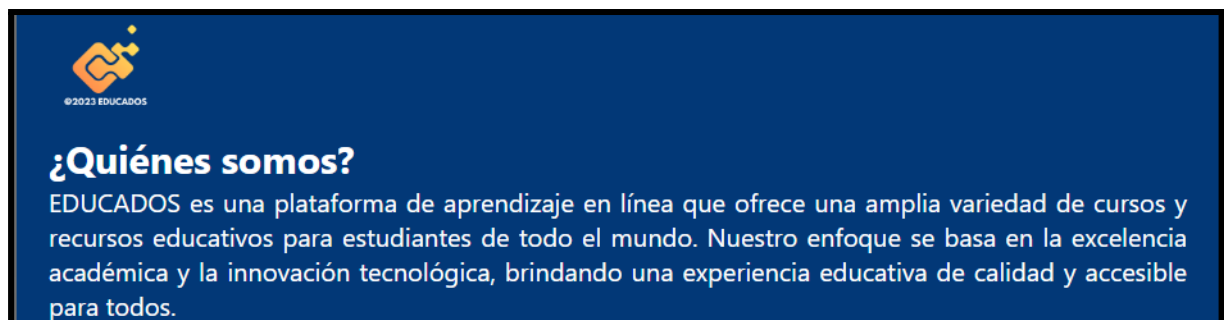
En la **cabecera**, podemos ver el logotipo de la empresa y un icono que nos lleva a la página de inicio de sesión de nuestra interfaz. También encontramos un menú que nos permite navegar por las diferentes secciones de la página. En la sección de contacto, se incluye un formulario donde los usuarios pueden enviar correos electrónicos a una cuenta de correo específica para esta función.

El formulario de contacto está encabezado por el título 'Contáctanos'. A continuación, hay cuatro campos de entrada: 'Nombre' con el placeholder 'Escriba aquí su nombre completo...', 'Correo electrónico' con 'Escriba aquí su email...', 'Asunto' con 'Escriba aquí su asunto...', y 'Mensaje' con 'Escriba aquí su mensaje...'. El campo de mensaje es un área de texto más grande. En la parte inferior del formulario, hay un botón azul con el texto 'Enviar'.

El **cuerpo** de la página se ha diseñado de forma ordenada y accesible, para aquellos usuarios administradores que no tengan conocimientos técnicos en desarrollo web. Se ha implementado un sistema sencillo en el que la etiqueta "**article**" se utiliza para crear cajas blancas en las que se puede escribir contenido. Esto permite añadir información de forma fácil y rápida.



El **footer**, situado al final del cuerpo y de la página en general, proporciona un bonito acabado visual sin crear espacios en blanco innecesarios.



El icono situado a la derecha de la cabecera nos lleva a la sección de conexión a la base de datos, es decir, a la página de acceso y registro. Al hacer clic en este icono, se nos redirige a una página que muestra un formulario para acceder a nuestra interfaz. Además, se ofrece la opción de registrarse y la posibilidad de restablecer la contraseña en caso de olvido.

A login form with a white background and a thin black border. At the top, the text "Iniciar sesión" is centered in a gray, sans-serif font. Below this, there are two input fields stacked vertically. The first field has a blue envelope icon and the placeholder text "Email...". The second field has a blue shield icon and the placeholder text "Contraseña...". Below the input fields, there is a horizontal line. At the bottom left, there are two links: "¿No tienes cuenta?, regístrate" and "He olvidado mi contraseña". At the bottom right, there is a blue button with the text "INICIAR SESION" in white, uppercase letters.

El proceso de registro de usuarios en Educados es sencillo y seguro. Se presenta un formulario donde los usuarios pueden introducir su nombre, correo electrónico y contraseña. Este formulario se encarga de validar los datos introducidos para garantizar que no se repite una dirección de correo electrónico existente en la base de datos y que la contraseña tiene al menos 5 caracteres. Aunque esta validación podría ser más estricta, hemos optado por mantenerla simplificada. Además, se incluye una casilla de verificación obligatoria en la que los usuarios deben aceptar haber leído las políticas establecidas.



○ Soy padre   ○ Soy alumno

☐ Ha leído y acepta todas nuestras políticas

[¿Ya tiene una cuenta?, inicie sesión](#)

**ACEPTAR**

Una vez introducidos correctamente los datos en el formulario de registro, se envía un correo electrónico personalizado a la dirección de Gmail proporcionada. Este correo contiene un número aleatorio generado entre 100.000 y 900.000.



Código de verificación en educados Recibidos x

**educadosinfo@gmail.com**  
para mí ▾

14:24 (hace 0 minutos) ☆ ↶ ⋮

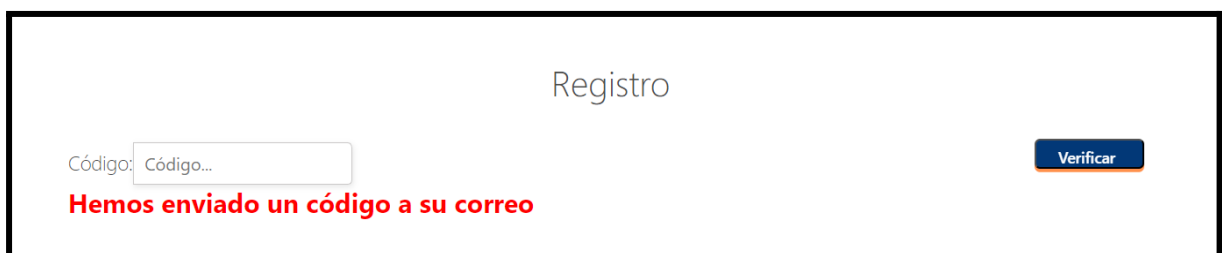
**Hola, Sergio**

Este es el código: **746498**

Si no ha sido usted ignore y borre este correo

↶ Responder   ↷ Reenviar

El usuario debe introducir este número en otro formulario proporcionado. Si el número introducido coincide con el generado, los datos del usuario se guardan en la base de datos y se le redirige de nuevo a la página de inicio de sesión.



Registro

Código:

**Hemos enviado un código a su correo**

**Verificar**

Una vez completado el proceso de registro y conectado el usuario, en el caso de un usuario padre, se requiere una relación con el hijo. La interfaz mostrará la información del usuario, pero ninguna opción adicional.

**Elena** Foro clase

**HOLA, Elena**  
**Como es tu primera vez, que tal si nos das los datos de tu hijo**

Nombre

Correo electrónico

**Cambiar**

En este se realiza una verificación de todos los alumnos disponibles, y si los datos introducidos coinciden, un trigger modificará la tabla correspondiente y redirigirá automáticamente a la interfaz de inicio.

**Elena** Foro clase

Calificaciones  
Circulares  
Incidencias  
Boletines

Hoy junio de 2023

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 1 de jul 2

Festividad del Día de Canarias  
Día de la Región de Castilla-La Mancha  
Corpus Christi (Castilla-La Mancha)  
Día de la Región de Murcia (M)  
Día de la Rioja (La Rioja)  
San Juan (festivo regional)  
Eid al-Adha (festivo regional)

Google Calendar

La educación es el arma más poderosa que puedes usar para cambiar el mundo

© 2023 EducadosSA

Lo mismo pasa con el usuario alumno solo que el formulario es distinto, en este no es necesario hacer ninguna comprobación de seguridad.

**Sergio** Foro clase

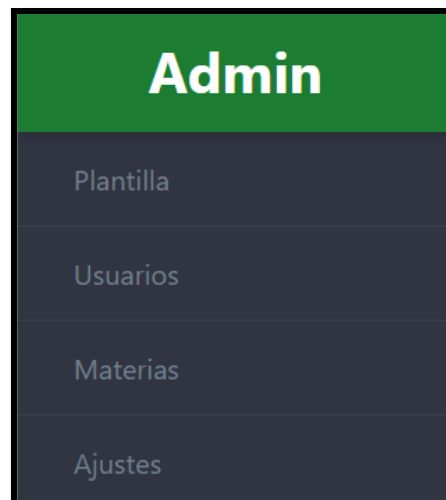
**Hola, Sergio**  
**¿Porqué no nos dices tu curso?:**

---  
ASIR1  
ASIR2

**Añadir**

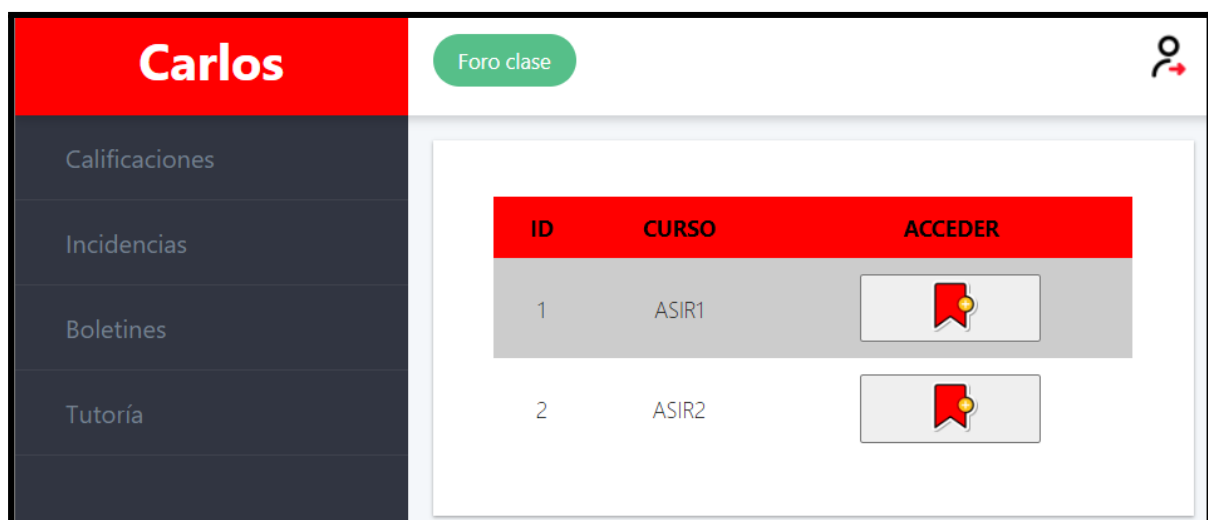
Todas las interfaces comparten estructura y estilo, la única diferencia notable es en el color de la paleta, la esquina superior izquierda que funciona como enlace a la página principal o de las tablas.

En la interfaz del administrador nos da a elegir todas estas opciones:



En ellas este usuario administrador puede hacer las modificaciones necesarias para cada una de las opciones, tanto de inserción, actualización y borrado.

En la interfaz del profesor, tenemos tres opciones, calificaciones, donde se muestran todos los cursos, seas o no profesor.



El botón que se muestra sirve de enlace, donde se guardan los datos de la tabla, así cada vez que sea necesario ir a una página anterior los datos no se pierden.

```
1 print("<TD><a href='calificaciones2.php?curso=" . urlencode($resultado['curso']) . "'>
2     <button id='boton_' . $resultado['id_curso'] . "' title='Notas'
3     onclick='mostrarMenu(" . $resultado['id_curso'] . ")'><img src='./img/acceso.png'
4     width='35%'></button></a></TD>\n");
```

Por ejemplo:

[http://localhost/proyecto\\_educados/profesor/calificaciones2.php?curso=ASIR1](http://localhost/proyecto_educados/profesor/calificaciones2.php?curso=ASIR1)

[http://localhost/proyecto\\_educados/profesor/calificaciones3.php?curso=ASIR1&materia=SO](http://localhost/proyecto_educados/profesor/calificaciones3.php?curso=ASIR1&materia=SO)

Si el curso seleccionado tiene asignaturas asignadas, se mostrará una lista de estas asignaturas. Si no, simplemente nos dirá que no tenemos materias en ese curso.

No tiene ninguna materia en este curso.

En el caso de que sí haya asignaturas disponibles, se mostrará un menú desplegable con tres evaluaciones y un botón "Mostrar". Una vez seleccionada una evaluación nos muestra un listado de todos los alumnos con esa materia.

La otra opción disponible es un sistema para enviar archivos o boletines a un correo, para que el correo se envíe, tiene que existir en la base de datos, si no, no lo envía, hay dos, uno para los padres y otro para el alumno.

Correo del alumno:

Subir archivo:  Ninguno archivo selec.

Correo del padre:

Subir archivo:  Ninguno archivo selec.

La tercera y última opción es exclusiva para el tutor. Si no lo es, aparecerá un mensaje indicando que no tienes acceso a esa sección.

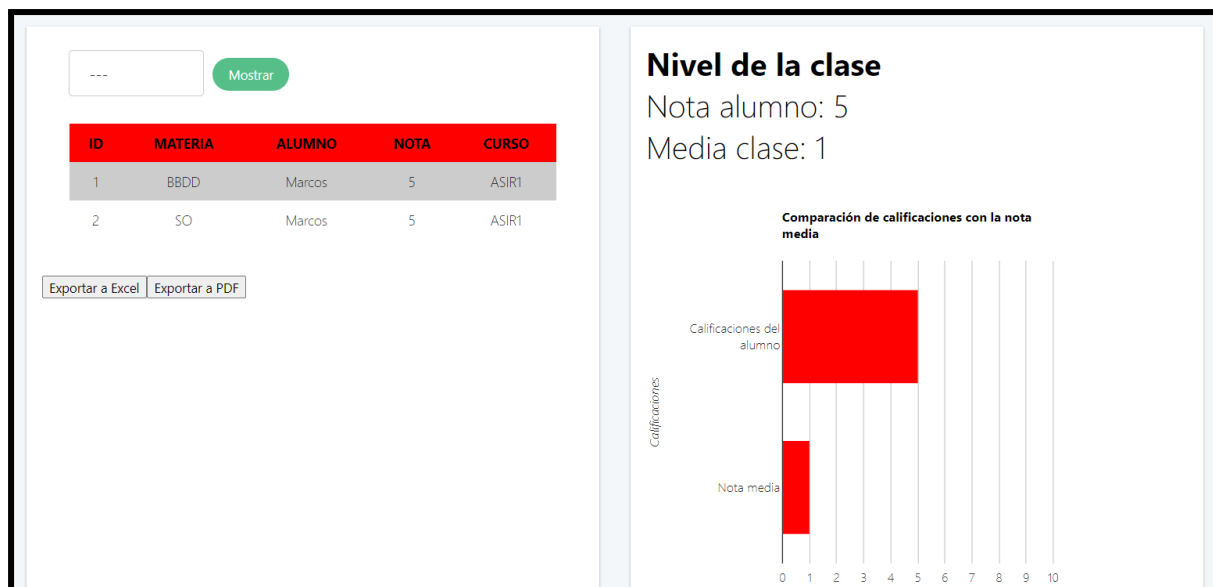
No tienes cursos asignados como tutor.

En caso de ser tutor de algún curso nos mostrará ese curso en particular, y dentro de este se listarán todos los alumnos con un botón “mostrar”.

NOMBRE	CORREO	CURSO	OPCIÓN
Sergio	sergio@gmail.com	ASIR1	Mostrar
Marcos	marcos@gmail.com	ASIR1	Mostrar
Alejandra	seracogui@gmail.com	ASIR1	Mostrar

Al hacer clic en el botón "Mostrar" se mostrarán las calificaciones de ese alumno. Inicialmente se ve así:

<div><div>---</div><div>Mostrar</div></div> <p>Tienes que elegir una evaluación</p>	<p><b>Nivel de la clase</b></p> <p>No se encontraron resultados</p>
---	---



Una vez seleccionada la evaluación deseada nos mostrará la lista de todas las asignaturas con sus respectivas notas, además de dos botones para exportar la tabla a formato **Excel** o **PDF** las cuales pueden utilizarse para enviar a el alumno o padre según corresponda.

Boletin\_Final curso.pdf

Boletin\_Final curso.xlsx

Mostrar todo

	A	B	C	D	E
1	ID	MATERIA	ALUMNO	NOTA	CURSO
2	1	BBDD	Marcos	5	ASIR1
3	2	SO	Marcos	5	ASIR1

ID	MATERIA	ALUMNO	NOTA	CURSO
1	BBDD	Marcos	5	ASIR1
2	SO	Marcos	5	ASIR1

Además de esto a la derecha nos saldrá una gráfica de medias, en la primera columna hace la media del alumno, suma todas las asignaturas y la segunda columna la media de la clase en este caso como los otros tres alumnos no tienen ninguna nota, la media es de 1.



Para la creación de la gráfica es necesaria la conexión a la base de datos, esto se ha logrado hacer de la siguiente manera:

Primero realizaremos la conexión a la base de datos y conseguiremos la nota media del alumno y de la clase.

```
1 $notaalumno = "(select COUNT(id_alumno) from calificaciones where id_alumno='$id_alumno')";
2 $notaclase = "(select COUNT(id_alumno) from calificaciones, materia where
3 calificaciones.id_materia=materia.id_materia and curso='$curso')";
4 $conn = mysqli_connect($servername, $username, $password, $dbname);
```

Hacemos un control previo de según qué evaluación ha sido elegida, que el query sea modificado.

```
1 // Consulta SQL
2 if ($evaluacion == "primer_ev") {
3     $sql = "SELECT SUM(primer_ev)/$notaalumno AS NotaAlumno, (SELECT sum(primer_ev)/$notaclase FROM
4 calificaciones, materia WHERE calificaciones.id_materia=materia.id_materia AND curso='$curso') AS
5 NotaClase FROM calificaciones WHERE id_alumno='$id_alumno'";
6 } elseif ($evaluacion == "segunda_ev") {
7     $sql = "SELECT SUM(segunda_ev)/$notaalumno AS NotaAlumno, (SELECT sum(segunda_ev)/$notaclase FROM
8 calificaciones, materia WHERE calificaciones.id_materia=materia.id_materia AND curso='$curso') AS
9 NotaClase FROM calificaciones WHERE id_alumno='$id_alumno'";
10 } elseif ($evaluacion == "tercer_ev") {
11     $sql = "SELECT SUM(tercer_ev)/$notaalumno AS NotaAlumno, (SELECT sum(tercer_ev)/$notaclase FROM
12 calificaciones, materia WHERE calificaciones.id_materia=materia.id_materia AND curso='$curso') AS
13 NotaClase FROM calificaciones WHERE id_alumno='$id_alumno'";
14 } elseif ($evaluacion == "nota_media") {
15     $sql = "SELECT SUM(nota_media)/$notaalumno AS NotaAlumno, (SELECT SUM(nota_media)/$notaclase FROM
16 calificaciones, materia WHERE calificaciones.id_materia=materia.id_materia AND curso='$curso') AS
17 NotaClase FROM calificaciones WHERE id_alumno='$id_alumno'";
18 }
```

El siguiente paso es mostrar estas notas.

```
1 // Ejecutar la consulta
2 $resultado = mysqli_query($conn, $sql);
3
4 if (mysqli_num_rows($resultado) > 0) {
5     $fila = mysqli_fetch_assoc($resultado);
6     $total_calificaciones = round($fila["NotaAlumno"]); // Redondear a valor entero
7     $nota_media = round($fila["NotaClase"]); // Redondear a valor entero
8
9     // Mostrar los resultados
10    echo "<br>Nota alumno: " . $total_calificaciones . "<br>";
11    echo "Media clase: " . $nota_media . "<br>";
12 } else {
13    echo "<br><br>No se encontraron resultados";
14 }
```

El siguiente paso es modificar la gráfica según nuestras necesidades.

```
1 <script type="text/javascript" src="https://www.gstatic.com/charts/loader.js"></script>
2 <script type="text/javascript">
3     google.charts.load('current', {'packages':['corechart']});
4     google.charts.setOnLoadCallback(drawChart);
5
6     function drawChart() {
7         var data = google.visualization.arrayToDataTable([
8             ['Calificaciones', 'Valor'],
9             ['Calificaciones del alumno', <?php echo $total_calificaciones; ?>],
10            ['Nota media', <?php echo $nota_media; ?>]
11        ]);
12
13        var options = {
14            title: 'Comparación de calificaciones con la nota media',
15            legend: { position: 'none' },
16            chartArea: { width: '50%' },
17            hAxis: {
18                title: 'Valor',
19                minValue: 0, // Establecer el valor mínimo en 0
20                maxValue: 10, // Establecer el valor máximo en 10
21                ticks: Array.from(Array(11).keys()) // Generar una matriz de 0 a 10 para las etiquetas del eje X
22            },
23            vAxis: {
24                title: 'Calificaciones'
25            },
26            colors: ['red'] // Establecer el color de las barras a rojo
27        };
28        var chart = new google.visualization.BarChart(document.getElementById('chart_div'));
29        chart.draw(data, options);
30    }
31 </script>
32 </head>
33 <body>
34     <div id="chart_div" style="width: 100%; height: 500px;"></div>
35 </body>
```

Este código utiliza **"Google Charts"** para dibujar gráficos de barras. En primer lugar, se carga la biblioteca y el archivo **"loader.js"**. El siguiente paso, es definir un script que se ejecutará una vez que la biblioteca se haya cargado correctamente, este carga el paquete **"corechart"**, necesario para dibujar el gráfico de barras. Dentro de la función **"drawChart"**, se establecen los datos del gráfico y las distintas opciones de configuración. **"Google.visualization.arrayToDataTable"** se utiliza para convertir los datos en una tabla. A continuación, se crea una instancia del gráfico de barras y se dibuja como un elemento HTML específico.

En resumen, el código carga la biblioteca **"Google Charts"**, define los datos y las opciones del gráfico de barras y lo dibuja en la página web.

## FUNCIONALIDAD

En cuanto a la funcionalidad de la página web se basa en código PHP, a excepción de un par de funciones que utilizan javascript.

Por ejemplo, en el caso de las notas del profesor, se utilizan funciones para abrir y cerrar el div en el que se introducen las notas de cada alumno, en el es necesario el almacenamiento del **id\_alumno** para su uso posterior.

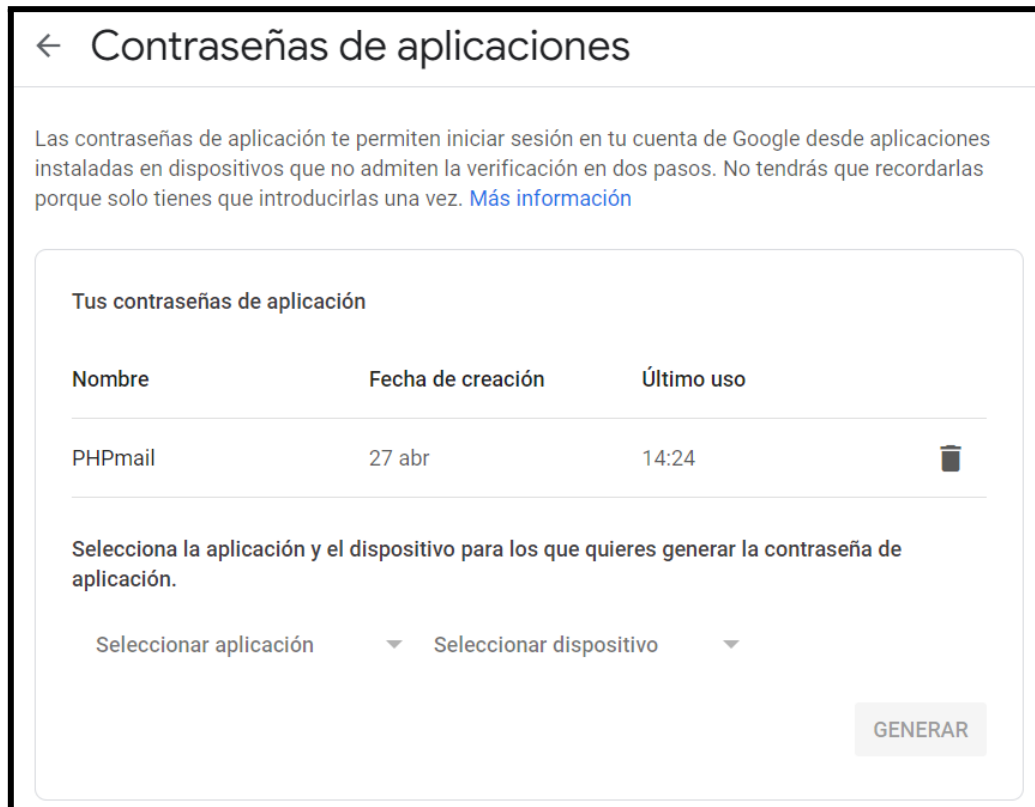
```
1  function abrir(id_alumno) {
2      var divId = "pitola_" + id_alumno;
3      var div = document.getElementById(divId);
4      if (div) {
5          div.style.display = "block";
6      }
7      var notaInputId = "nota_" + id_alumno;
8      var notaInput = document.getElementById(notaInputId);
9      if (notaInput) {
10         notaInput.focus();
11     }
12 }
13
14 function cerrar(id_alumno) {
15     var divId = "pitola_" + id_alumno;
16     var div = document.getElementById(divId);
17     if (div) {
18         div.style.display = "none";
19     }
20 }
```

También se utilizan funciones para exportar las tablas a excel y a pdf uso otras dos funciones.

Primero se recoge la evaluación seleccionada y después recoge los datos que se muestran en la tabla y los transforma, para su funcionamiento es necesario una librería **html2pdf** para generar el archivo PDF y para Excel la biblioteca **xlsx**.

```
1  var evaluacionSelect = document.getElementById("evaluacion");
2  var evaluacionValue = evaluacionSelect.value;
3
4  function generatePDF() {
5      var element = document.querySelector('.table');
6      var select = document.getElementById('evaluacion');
7      var evaluacion = select.options[select.selectedIndex].text;
8      var fileName = 'Boletin_' + evaluacion + '.pdf';
9
10     var opt = {
11         margin: 0.5,
12         filename: fileName,
13         image: { type: 'jpeg', quality: 0.98 },
14         html2canvas: { scale: 2 },
15         jsPDF: { unit: 'in', format: 'letter', orientation: 'landscape' },
16     };
17
18     html2pdf().from(element).set(opt).save();
19 }
20
21 function generateExcel() {
22     var table = document.querySelector('.table');
23     var select = document.getElementById('evaluacion');
24     var evaluacion = select.options[select.selectedIndex].text;
25     var fileName = 'Boletin_' + evaluacion + '.xlsx';
26
27     var workbook = XLSX.utils.table_to_book(table);
28     var excelBuffer = XLSX.write(workbook, { bookType: 'xlsx', type: 'array' });
29
30     saveAsExcelFile(excelBuffer, fileName);
31 }
32
33 function saveAsExcelFile(buffer, fileName) {
34     var blob = new Blob([buffer], { type: 'application/octet-stream' });
35     saveAs(blob, fileName);
36 }
37
38 function saveAs(blob, fileName) {
39     var url = window.URL.createObjectURL(blob);
40     var anchorElement = document.createElement('a');
41     anchorElement.href = url;
42     anchorElement.download = fileName;
43     anchorElement.click();
44     window.URL.revokeObjectURL(url);
45 }
```


El servicio de correo funciona mediante la biblioteca PHPMailer, la dificultad extra añadida ha sido lograr su funcionamiento con gmail. Para que funcione con gmail, primero se crea una cuenta nueva en la cual hay que activar la verificación en dos pasos y generar una clave de aplicaciones que nos genera gmail.



← Contraseñas de aplicaciones

Las contraseñas de aplicación te permiten iniciar sesión en tu cuenta de Google desde aplicaciones instaladas en dispositivos que no admiten la verificación en dos pasos. No tendrás que recordarlas porque solo tienes que introducirlas una vez. [Más información](#)

Tus contraseñas de aplicación

Nombre	Fecha de creación	Último uso	
PHPmail	27 abr	14:24	

Selecciona la aplicación y el dispositivo para los que quieres generar la contraseña de aplicación.

Seleccionar aplicación ▼ Seleccionar dispositivo ▼

GENERAR

```
1 $mail->Host      = 'smtp.gmail.com';
2 $mail->SMTPAuth   = true;
3 $mail->Username   = 'educadosinfo@gmail.com';
4 $mail->Password   = 'griitqiiiawfsyjq';
5 $mail->SMTPSecure = 'tls';
6 $mail->Port       = 587;
```

La clave sirve como autenticación de la cuenta en la biblioteca y en este caso en vez de enviar un texto se adjunta el boletín.

```
1 $mail->isHTML(true);
2 $mail->CharSet = 'UTF-8';
3 $mail->Subject = 'Publicación boletín';
4 $mail->Body    = $body;
5 $mail->AddAttachment($boletin['tmp_name'], $boletin['name']);
```

## DISEÑO

Para el diseño de Educados se utilizan dos tipos de hojas de estilo CSS: una para la página principal y otra para la interfaz. La paleta de colores principal está compuesta por azul oscuro y naranja. La hoja de estilo secundaria varía según el tipo de usuario, es decir el administrador tendrá el verde mientras que el profesor tiene la roja. Todo el diseño en ambas está basado en “**Responsive design**”, para que en caso de disminuir el tamaño de la página, el texto se adapte a la página. Esto se consigue usando las medidas en %, y haciendo uso de diseños flexibles como “**display: flex**”, o de reglas especiales como **@**:

```
1 @media screen and (min-width: 48rem) {  
2   .dashboard {  
3     --column-count: 4;  
4   }  
5 }  
6 @supports (display: grid) {  
7   .dashboard {  
8     margin: 0;  
9   }
```

En este caso **@media** aplica estilos específicos cuando la pantalla tiene un ancho mínimo de **48 rem**. Y **@supports** para aplicar estilos si el navegador lo admite.

El diseño busca evitar la monotonía y ser más interactivo, creando animaciones en el menú o en los botones.

```
1 input[type="button"] {  
2   text-align: justify;  
3   align-items: center;  
4   align-content: center;  
5   font-size: 30px;  
6   font-weight: 600;  
7   color: white;  
8   padding: 6px 25px 0 20px;  
9   margin: 10px 8px 20px 0;  
10  display: inline-block;  
11  text-decoration: none;  
12  border-radius: 5px;  
13  background-color: #023877;  
14  box-shadow: 0 3px #023877;  
15  transition: all 0.1s linear 0s;  
16  position: relative;  
17  width: auto;  
18 }
```

En este caso a el type="button", se le agrega una sombra del mismo color, lo que simula ser 3D

```
1 input[type="button"]:hover {  
2   top: 3px;  
3   background-color: #0074C7;  
4   box-shadow: none;  
5 }
```

Y en este se refiere a los estilos que se aplicarán cuando se encuentre en “**hover**”, es decir cuando se pase el ratón, en este caso lo que hacemos es quitarle ese sombreado y simula ser pulsado.

## ARQUITECTURA DE DOCKER

### EXPLICACIÓN DE DOCKER

La contenerización de Educados es fundamental para el desarrollo del proyecto, nos proporciona un entorno de contenedores virtualizados que facilita la implementación y gestión de las aplicaciones. Docker es una plataforma que empaqueta y distribuye aplicaciones con todas sus dependencias, garantizando una ejecución consistente en diferentes entornos, asegura un funcionamiento correcto y además de que usa menos recursos para las aplicaciones. En el caso de Educados se basa en dos servicios, una pila lemp, que funciona como servidor y otra pila elk para el registro y control de los logs.

La pila **LEMP** (Linux, Nginx, Mysql y Php) es una combinación de tecnologías donde vamos a ejecutar Educados, es decir, su página web y su base de datos, por otro lado la pila **ELK** (Elasticsearch, Logstash, Kibana) que se utiliza para la gestión y análisis de registros y logs, Elasticsearch se encarga del almacenamiento y la búsqueda de registros, Logstash se utiliza para la recopilación, el procesamiento y el enriquecimiento de registros, y Kibana se encarga de la visualización y el análisis de datos.

Estas dos pilas al formarse de varias aplicaciones van a ser contenerizadas y desplegadas con **DOCKER COMPOSE**, el objetivo principal de la contenerización de servicios se basa en ocupar el menor espacio posible en disco.

[MANUAL DOCKER](#) - [GITHUB](#)

## FUNCIONAMIENTO

Para su funcionamiento Educados ha sido creado en ambas pilas en docker-compose, en la pila **LEMP**, se necesitan dos Dockerfiles, uno para Nginx y otro para PHP ya que Linux no ejecuta PHP de por sí y necesita ser instalado. A partir de estos Dockerfiles, se crean las imágenes necesarias para contenerizar los servicios. En el docker-compose, se crean 4 contenedores, siendo el cuarto PHPMyAdmin. PHPMyAdmin ha sido diseñado para los administradores de la escuela, proporcionándoles una interfaz más amigable en caso de que no tengan conocimientos avanzados en el uso de MySQL. Los puertos **8080** y **8081** se utilizan para diferentes aplicaciones: el puerto 8080 para la página web y el puerto 8081 para PHPMyAdmin.

Por otro lado, la pila **ELK** se crea utilizando imágenes ya existentes en Docker, por lo que no es necesario crear Dockerfiles. En el docker-compose de la pila **ELK**, se incluyen 5 contenedores, donde Elasticsearch se mapea al puerto **9200** y Kibana al puerto **5601**.

Para la unión de ambos docker-compose es necesaria la creación de una red Docker, la cual asigna una dirección ip (172.X.X.X) a cada uno de sus contenedores, su funcionalidad se basa en establecer una comunicación y conectividad. La red Docker permite crear y administrar redes virtuales para los contenedores facilitando la comunicación entre ellos.

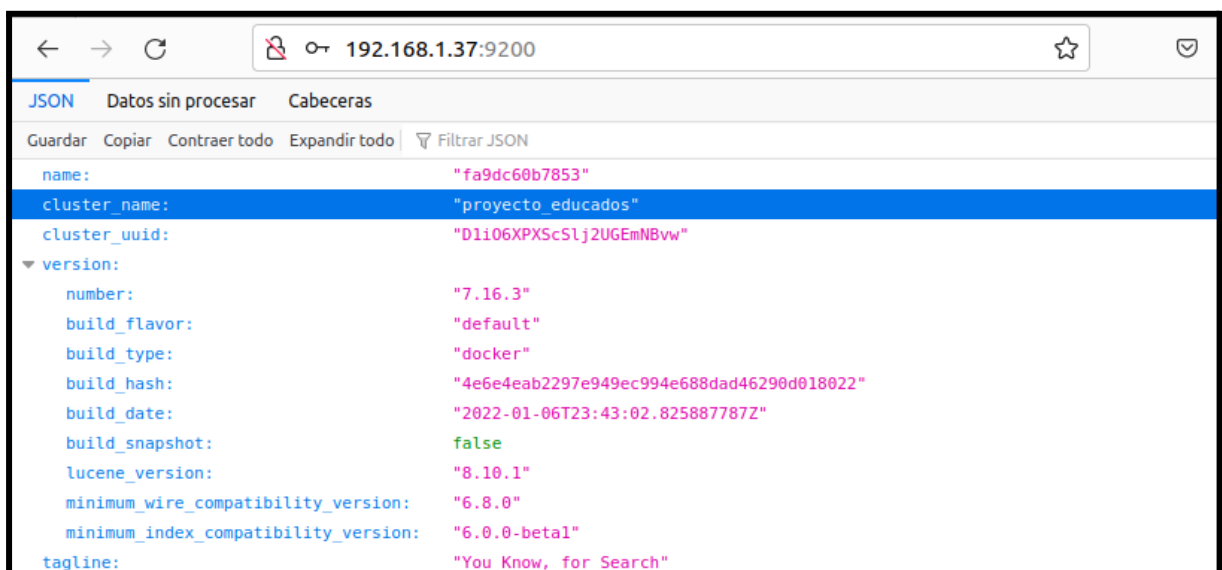
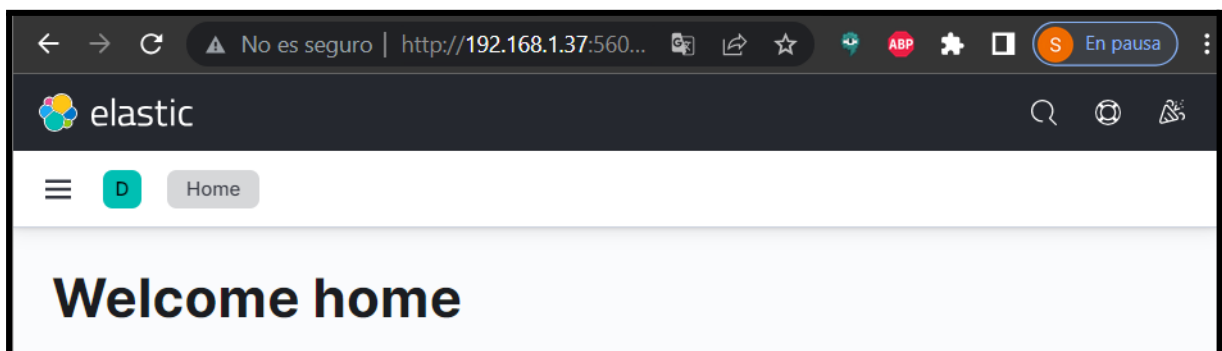
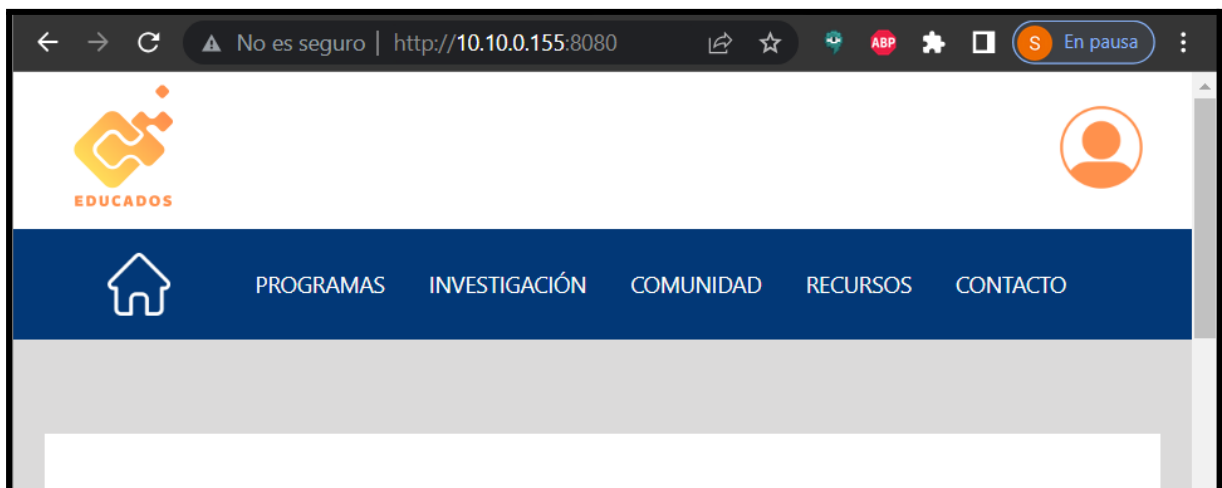
Aunque ambas pilas podrían crearse en un único docker-compose, tenerlas como servicios separados ofrece ventajas en términos de modularidad, flexibilidad y escalabilidad, ya que si uno de los servicios falla, el otro puede seguir funcionando y viceversa.

Para que Educados funcione correctamente, es necesario tener los contenedores en estado Up y saber que están trabajando correctamente.

La idea detrás de la pila **ELK** es automatizar procesos, como por ejemplo leer los logs de MySQL y mostrar los procesos en gráficos, o enviar un correo electrónico al estudiante cuando se publican las notas. Esta idea surgió de la observación de que al crear la pila LEMP, los logs de cada contenedor se mostraban en el terminal.

Para utilizar Educados, basta con abrir un navegador web e introducir la dirección IP del servidor. Además, es necesario mapear los puertos correspondientes. Si todo ha ido bien, se puede acceder y utilizar Educados a través de cualquier navegador, permitiendo disfrutar de todas las funcionalidades y características que ofrece.





## RESULTADOS

Con el resultado de Educados, he obtenido gran cantidad de conocimientos, como el uso de JS para la exportación de las tablas a PDF y Excel o para la creación de la gráfica. He obtenido conocimientos en bases de datos al automatizar una base de datos tan extensa, así como de Docker con la creación de las redes de los contenedores.

## CRONOLOGÍA

El proyecto comenzó en el verano del 2022, intenté desarrollar una idea para el proyecto, comencé a ver posibilidades y probar ideas, intenté hacer un bot en telegram para que sirviera de asistente por ejemplo. Durante ese verano no avancé en el proyecto pero saqué en conclusión que quería automatizar una base de datos.

Al comienzo del curso se me ocurrió hacer la idea de un servicio donde con una base de datos hacer gráficas de resultados, como un gráfico circular que mostrará el inicio de sesión de los usuarios.

Cuando propuse la solicitud del proyecto, consideré la opción de hacer una automatización de una base de datos, mi idea era hacer eso hasta que me di cuenta que era muy simple, nada complicado con lo que aprendiera. Como la idea ya estaba elegida decidí hacer la página web principal durante las vacaciones de Navidad mientras pensaba en una idea para mi proyecto.

Se me ocurrió la idea de crear la página web de una aerolínea, simulando a Iberia, hasta que me di cuenta que se requiere de muchas aplicaciones que automatizan el proceso de la compra de vuelos, que según el mes era más barato y demás, por lo que simplemente acabó como idea.

Finalizado el mes de febrero y los exámenes decidí comenzar el proyecto de nuevo, la idea surgió un día que estaba mirando educamos. Se me ocurrió hacer un ecosistema educativo digital sin conocer cómo funciona, el resultado ha sido Educados, que es la mezcla entre educamos y dos, sin conocer las funciones que puede hacer un profesor o un administrador desarrollé mi página web dándome una idea de cómo funciona. Antes del desarrollo de mi página web realice la base de datos, lo más complicado de la base de datos es la gran extensión de tablas y el proceso de automatizar esas tablas sin errores.

Según iba avanzando y haciendo el desarrollo de la página web no paraba de intentar desarrollar idea innovadoras, como **primer\_login** que me resolvía dudas de por ejemplo como hacer la relación entre padre e hijo, o de hacer un trigger que hasta que el alumno no elija su curso, no se le agrega a la tabla calificaciones.

Durante el desarrollo de la GUI del profesor, se me ocurrió la idea de un calendario, en este caso se me ocurrió la idea de añadir el calendario de Google, este es fácil de administrar y así poder aprovecharme de las actividades que vais añadiendo.

Como mi idea principal era realizar gráficas no se me olvidó la idea y lo implemente para realizar entre los alumnos las medias correspondientes.

Una vez hice Educados, con el deseo de seguir avanzando con el proyecto para que sea cada vez más desarrollado, se me ocurrió la idea de convertirlo en aplicación, ¿Cómo?, con Docker, debido a que en las prácticas trabajé con Docker, se me ocurrió Contenerizar este servicio, realicé la contenerización de la pila lemp, donde conseguí hacer que funcionara una máquina virtual de linux como servidor.

A escasas semanas de la entrega del proyecto desarrolle la idea la pila elk, como quería hacer un bot de telegram para automatizar procesos y actividades, se me ocurrió la idea de usar las **SESSION**, para hacer un control de los usuarios activos y poder cerrar sesiones o controlar el tiempo activo de estos, pero al no tener conocimientos y poco tiempo realicé la pila elk, con el objetivo de que se pudiera administrar los servicios a base de los logs, esta idea se desarrolló al conocer que en el Docker de la pila lemp se mostraban los logs de los distintos servicios, cada cambio o movimientos en la web era un log, por lo que decidí aprovecharlo.

## CONCLUSIÓN

Educados ha sido un total desafío, especialmente porque al comienzo del proyecto tenía un conocimiento limitado sobre las tecnologías que he utilizado. Sin embargo, esta experiencia me ha brindado la oportunidad de aprender y crecer en el camino.

Todo el proceso desde el inicio al final ha sido muy entretenido, a la vez de agobiante, me ha mantenido muy preocupado en realizar un proyecto en condiciones a la vez que me ha mantenido interesado en la realización de este.

Además Educados, me ha servido para el desarrollo de mis conocimientos en las distintas materias utilizadas, ya sea en Base de datos, donde he aprendido a automatizar una base de datos extensa, a la vez de Implementación de desarrollo web donde he extendido mis conocimientos sobre PHP y JS, y también en Sistemas Operativos, donde he aprendido a desarrollar una aplicación web en Docker.

La parte más elevada en cuanto a su complejidad sin duda ha sido en la creación de la página web, el desarrollo de esta al ser tan extensa y compleja para mí ha supuesto la mayor parte del tiempo del proyecto.

En cuanto a la finalización del proyecto ha supuesto una gran satisfacción para mí debido al aprendizaje y desarrollo de nuevos y ya conocidos conocimientos reforzando estos.

## **POSIBLES AMPLIACIONES**

- ❖ Integración de un chat entre profesor y los padres o alumnos.
- ❖ Integración de un foro para la clase.
- ❖ Integración de una plataforma de recursos educativos.
- ❖ Sistema de notificaciones.
- ❖ Integración de plataforma de video para reuniones o la realización de clases en línea.
- ❖ Sistema de cambio de contraseña.
- ❖ Registro y seguimiento de asistencia.
- ❖ Biblioteca virtual.
- ❖ Personalización del perfil del usuario.
- ❖ Aplicación móvil.
- ❖ Ubuntu server.
- ❖ Dominio

## WEBGRAFÍA

- *Stack Overflow en español*. (s. f.). Stack Overflow en español.  
<https://es.stackoverflow.com/>
- *Web Server*. (s. f.). NGINX Documentation.  
<https://docs.nginx.com/nginx/admin-guide/web-server/>
- *El ELK Stack: de los creadores de Elasticsearch*. (s. f.). Elastic.  
<https://www.elastic.co/es/what-is/elk-stack>
- *Docker*. (s. f.). <https://hub.docker.com/>
- CodePen. (s. f.). <https://codepen.io/Wcobra9>
- SergioAcostaGuijarro. (s. f.). *GitHub - SergioAcostaGuijarro/ProyectoFinalCiclo*. GitHub. <https://github.com/SergioAcostaGuijarro/ProyectoFinalCiclo>
- Youtube:
  - Joe. (2020, 13 noviembre). *Docker - NGINX, PHP, MYSQL* [Video]. YouTube.  
<https://www.youtube.com/watch?v=tXg24aFcp-0>
  - Lian. (2022, 6 febrero). *Install Elasticsearch, Kibana, Logstash, and Filebeat using Docker Compose* [Video]. YouTube.  
<https://www.youtube.com/watch?v=urUQGoT6d4Y>
- Heidi, E., & Horcasitas, J. (2022). How To Install Linux, Nginx, MySQL, PHP (LEMP stack) on Ubuntu 22.04. *DigitalOcean*.  
<https://www.digitalocean.com/community/tutorials/how-to-install-linux-nginx-mysql-php-lemp-stack-on-ubuntu-22-04>
- *W3Schools Free Online Web Tutorials*. (s. f.). <https://www.w3schools.com/>
- *OpenWebinars, Aceleradora de Profesionales Tecnológicos*. (s. f.). OpenWebinars.net. <https://openwebinars.net/>