



2-6-2024

# Página WEB - YARA



Vicente Aguilar

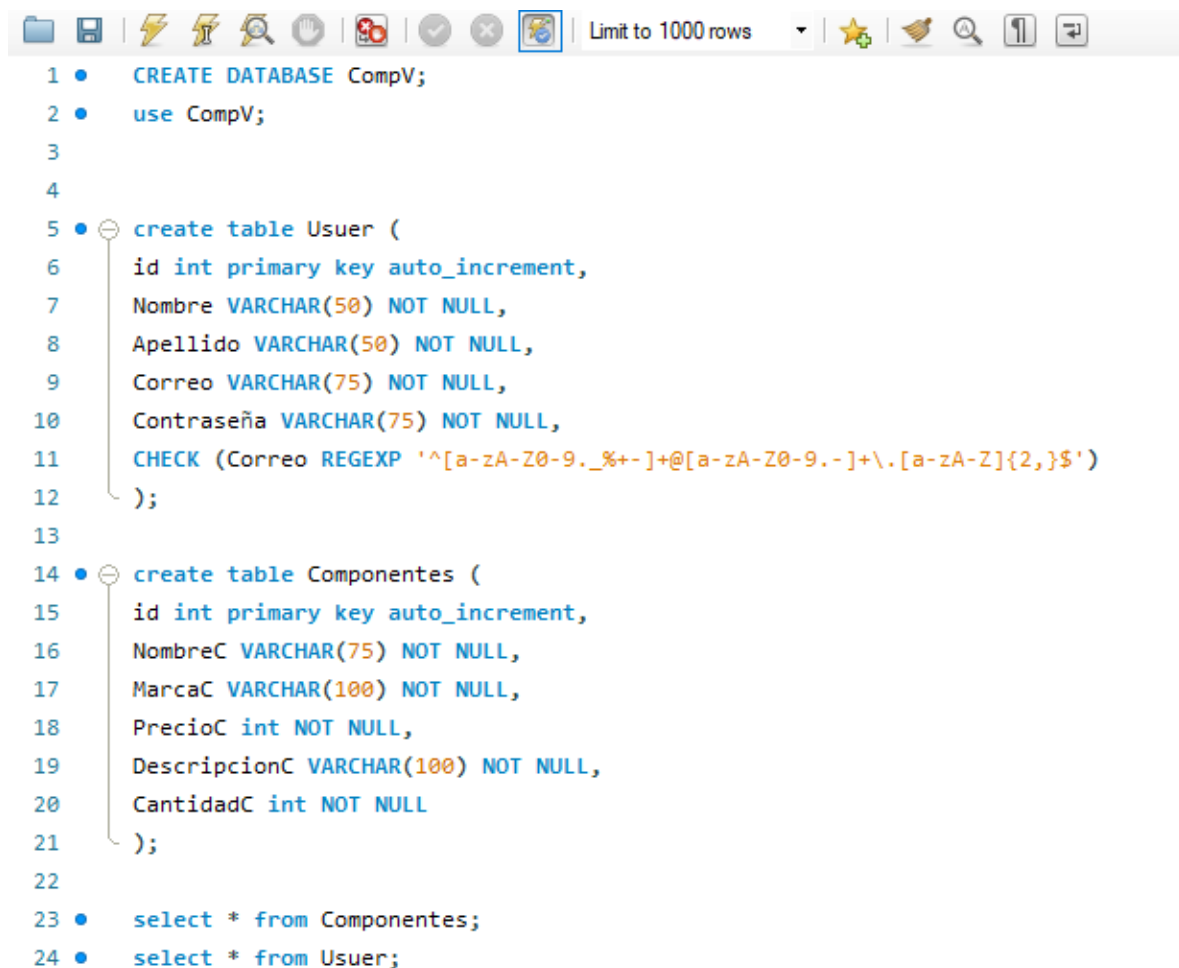
INSTITUTO PROFESIONAL SANTO TOMAS

## Pagina web – YARA:

Para la entrega de la evaluación “Evaluación Unidad 2” se realizo una pagina web enfocada primeramente como una pagina donde cada usuario pudiese registrarse para realizar diferentes compras de componentes para un computador, a lo largo del desarrollo esta idea cambio y paso a ser una pagina que funcionaria como parte administrativa de una empresa / Tienda que provee componentes para computadoras (YARA).

La página funciona a través de “Node” con un “app.js” / express, y posee el apartado para la conexión a una base de datos (Script) donde se puede revalidar los cambios hechos en la página web.

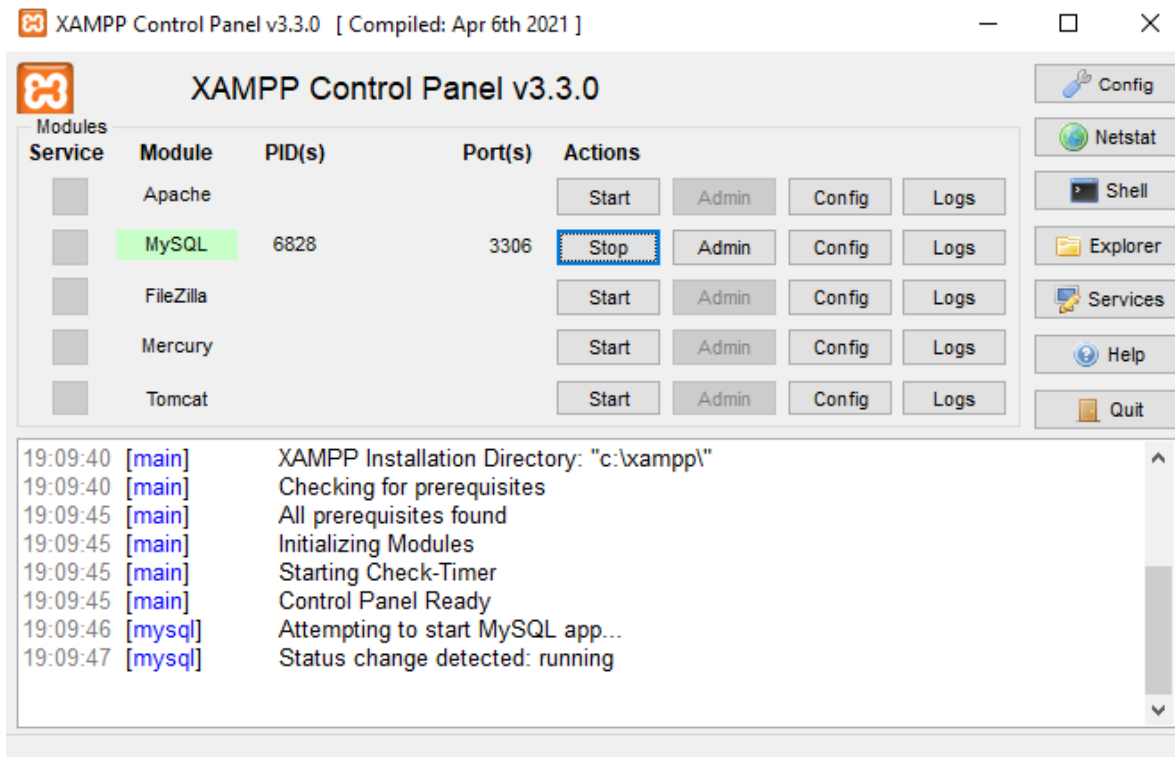
## Apartado Script BDD:



```
1 • CREATE DATABASE CompV;
2 • use CompV;
3
4
5 • create table User (
6     id int primary key auto_increment,
7     Nombre VARCHAR(50) NOT NULL,
8     Apellido VARCHAR(50) NOT NULL,
9     Correo VARCHAR(75) NOT NULL,
10    Contraseña VARCHAR(75) NOT NULL,
11    CHECK (Correo REGEXP '^[a-zA-Z0-9._%+-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}$')
12 );
13
14 • create table Componentes (
15     id int primary key auto_increment,
16     NombreC VARCHAR(75) NOT NULL,
17     MarcaC VARCHAR(100) NOT NULL,
18     PrecioC int NOT NULL,
19     DescripcionC VARCHAR(100) NOT NULL,
20     CantidadC int NOT NULL
21 );
22
23 • select * from Componentes;
24 • select * from User;
```

Este Script cuenta con las variables con las que trabajara la pagina web, en las cuales recibirán cambios en sus tablas (Inserción de datos o modificación).

Este Script fue utilizado en Mysql Workbench, si se utiliza otro programa, solo hay que actualizar el apartado en “app.js” para realizar la conexión a la BDD, además la conexión fue posible a “Xampp” para levantar un LocalHost en mysql Workbench.



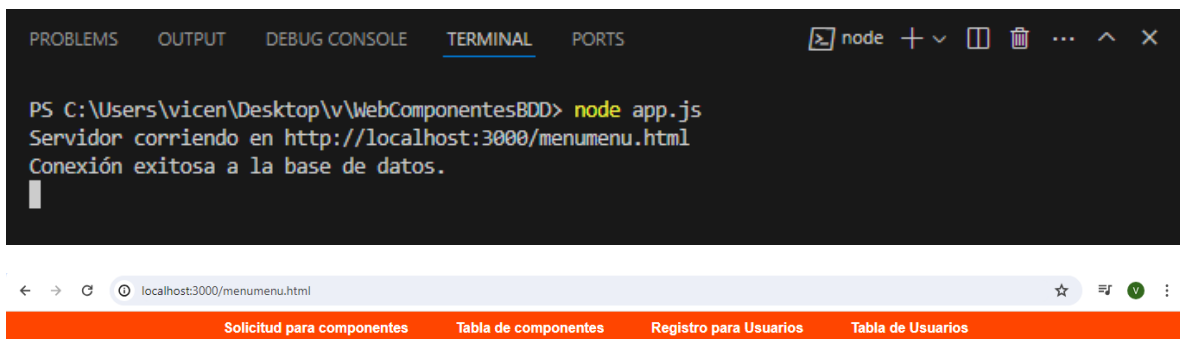
## Página Web – YARA:

Se presenta documentación de la página web a continuación.



Al ejecutar el comando “node app.js” en la Terminal de Visual Studio Code, se nos proporcionara el link que nos llevara a este menú principal, el cual cuenta con una breve descripción del uso de esta página web:

*¡Uso de pagina solo para empleados autorizados!*



Como se visualiza en la parte superior del menú, se encuentran las diferentes pestañas disponibles

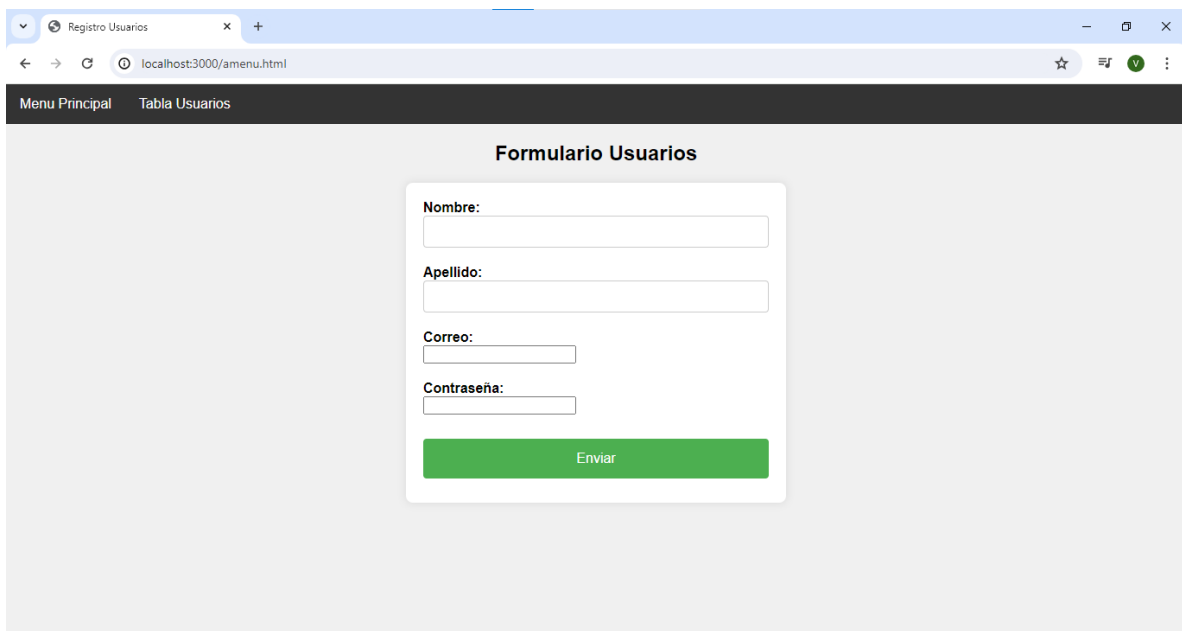
## Usuarios:

El apartado de usuarios cuenta con dos páginas, una para realizar un registro y otro para visualizar los usuarios registrados

**Registro para Usuarios**

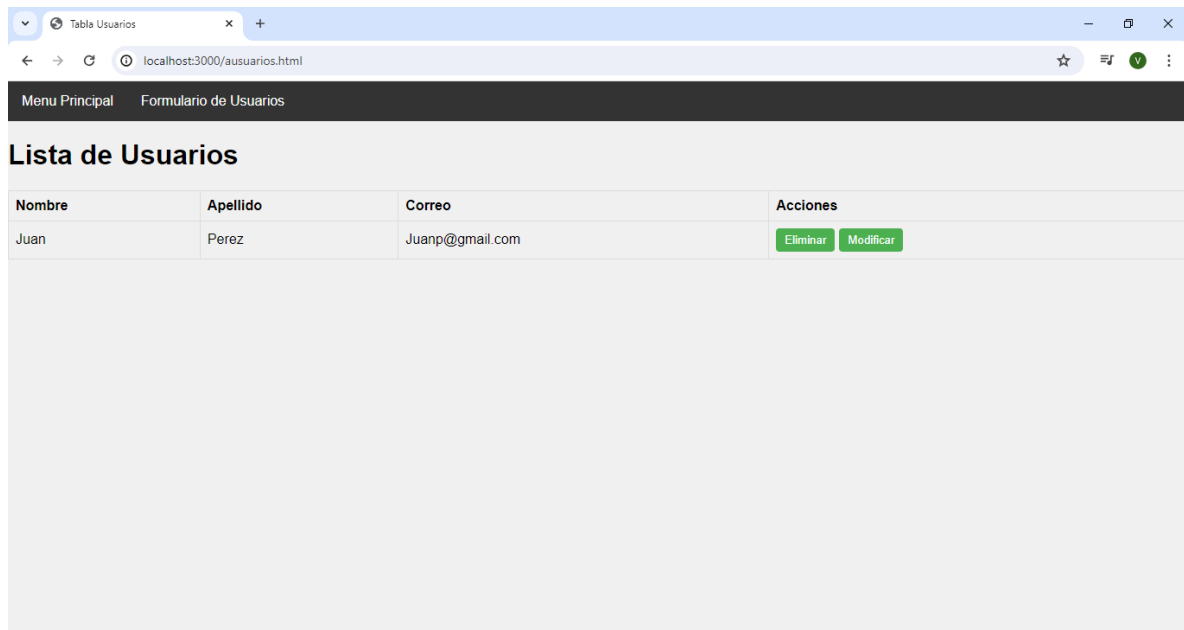
**Tabla de Usuarios**

## Registro de usuarios:

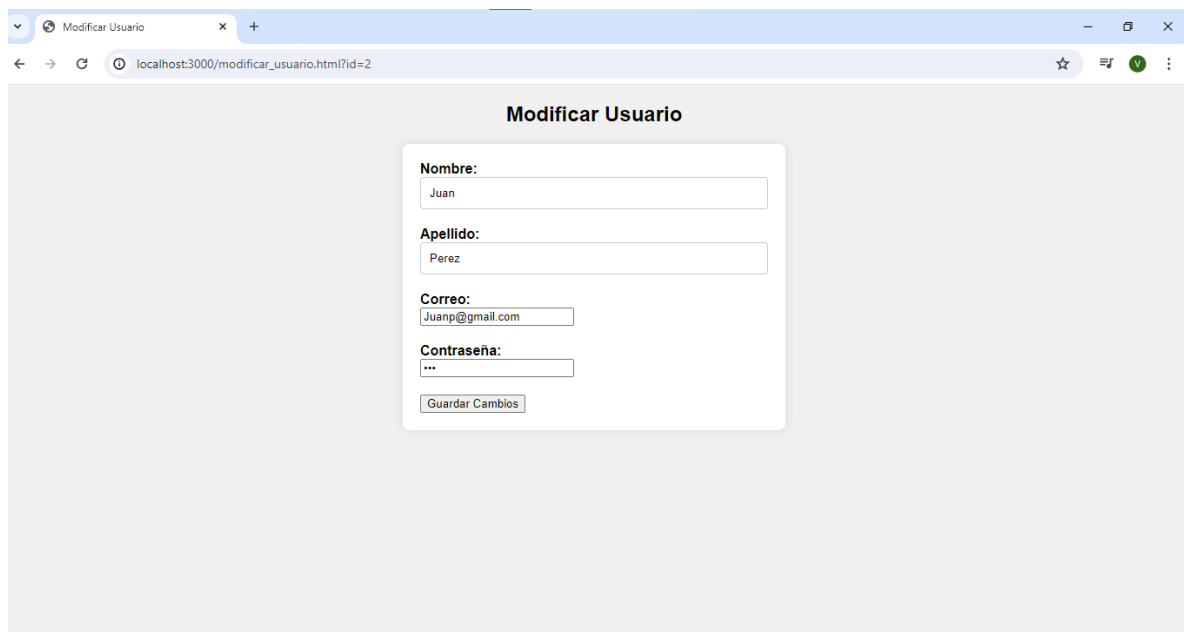


The screenshot shows a web browser window with the title 'Registro Usuarios'. The address bar displays 'localhost:3000/amenu.html'. The browser's navigation bar includes 'Menu Principal' and 'Tabla Usuarios'. The main content area features a form titled 'Formulario Usuarios' with the following fields: 'Nombre:', 'Apellido:', 'Correo:', and 'Contraseña:'. Each field is represented by a text input box. Below these fields is a green button labeled 'Enviar'.

Un registro algo simple que cuenta con el Nombre – Apellido – Correo electrónico – Contraseña.



La tabla representa los usuarios de esta forma, dando la posibilidad de eliminar o modificar a este usuario, si decidimos modificarlo saldrá esta otra pestaña:



Al modificarlo se vera reflejado en el apartado anterior si le damos click en el botón “guardar cambios”

Tabla Usuarios

localhost:3000/usuarios.html

Menu Principal Formulario de Usuarios

## Lista de Usuarios

Nombre	Apellido	Correo	Acciones
Juuuuan	Perez	Juanp@gmail.com	<a href="#">Eliminar</a> <a href="#">Modificar</a>

De igual forma, recordando que esta conectada la pagina con una base de datos, en MySql se ve de la siguiente manera:

Query 1

```

16 NombreC VARCHAR(75) NOT NULL,
17 MarcaC VARCHAR(100) NOT NULL,
18 PrecioC int NOT NULL,
19 DescripcionC VARCHAR(100) NOT NULL,
20 CantidadC int NOT NULL
21 );
22
23 • select * from Componentes;
24 • select * from Usuer;

```

Result Grid

id	Nombre	Apellido	Correo	Contraseña
2	Juuuuan	Perez	Juanp@gmail.com	avv

Usuer 1

Output

#	Time	Action	Message	Duration / Fetch
1	19:30:27	use Comp V	0 row(s) affected	0.000 sec
2	19:30:35	select * from Usuer LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec

Y si le damos en Eliminar aparecerá lo siguiente:

Tabla Usuarios

localhost:3000/usuarios.html

Menu Principal Formulario de Usuarios

## Lista de Usuarios

Nombre	Apellido	Correo	Acciones
Juuuuan	Perez	Juanp@gmail.com	<a href="#">Eliminar</a> <a href="#">Modificar</a>

localhost:3000 dice  
¿Estás seguro de que deseas eliminar este usuario?

[Aceptar](#) [Cancelar](#)

y el usuario será eliminado:

Tabla Usuarios

localhost:3000/usuarios.html

Menu Principal

Formulario de Usuarios

## Lista de Usuarios

Nombre	Apellido	Correo	Acciones

Query 1

Limit to 1000 rows

16

NombreC VARCHAR(75) NOT NULL,

17

Ma

Execute the selected portion of the script or everything, if there is no selection

18

PreC int NOT NULL,

19

DescripcionC VARCHAR(100) NOT NULL,

20

CantidadC int NOT NULL

21

);

22

23

• select \* from Componentes;

24

• select \* from Usuarios;

Result Grid

Filter Rows:

Edit

Export/Import

Wrap Cell Content

id	Nombre	Apellido	Correo	Contraseña

Usuer 2

Apply

Output

Action Output

## Componentes:

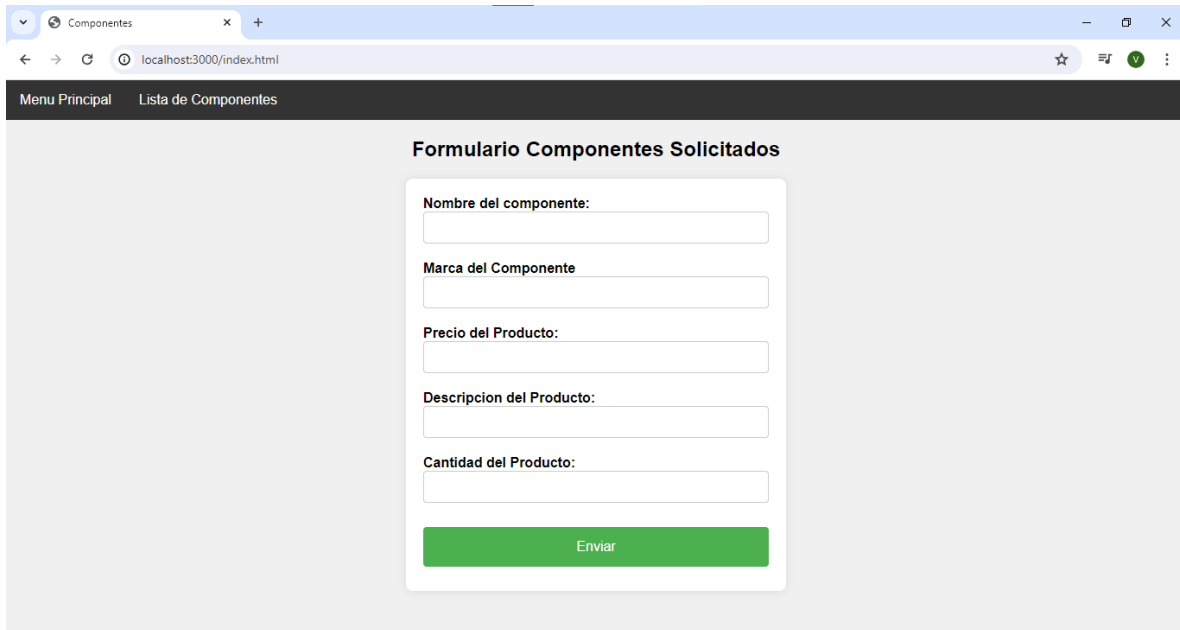
Al igual que el apartado para los usuarios, la solicitud de componentes cuenta con dos apartados:

**Solicitud para componentes**

**Tabla de componentes**



En el primer apartado de “Solicitud para componentes”, se nos despliega una pestaña con un Forms:



The screenshot shows a web browser window with the address bar displaying 'localhost:3000/index.html'. The browser has a tab titled 'Componentes'. The page has a dark header with 'Menu Principal' and 'Lista de Componentes'. The main content area is titled 'Formulario Componentes Solicitados' and contains a form with the following fields:

- Nombre del componente:
- Marca del Componente
- Precio del Producto:
- Descripcion del Producto:
- Cantidad del Producto:

At the bottom of the form is a green button labeled 'Enviar'.

En el cual podremos solicitar un producto – Agregarle la marca deseada – Colocarle un precio de venta – Agregar una descripción del producto – Agregarle la cantidad a solicitar.

Así mismo podemos visualizar los componentes solicitados en una tabla en el apartado de “Lista de Componentes” donde podremos visualizar lo siguiente:



The screenshot shows a web browser window with the address bar displaying 'localhost:3000/listardatos.html'. The browser has a tab titled 'Tabla Componentes'. The page has a dark header with 'Solicitud de Componentes' and 'Menu Principal'. The main content area is titled 'Componentes' and contains a table titled 'Lista de Componentes'.

Nombre:	Marca	Precio de venta	Descripcion	Stock	Acciones
CPU	INTEL	50000	Cpu	20	<button>Eliminar</button> <button>Modificar</button>

Si clickeamos en modificar se nos desplegara lo siguiente:

Modificar Pelicula

localhost:3000/modificar\_componente.html?id=3

### Modificar Componente

**Nombre:**  
CPU

**Marca:**  
INTEL

**Precio:**  
50000

**Descripción:**  
Cpu

**Cantidad:**  
20

Guardar Cambios

Y al igual que con usuarios, se vera reflejado algún cambio en la base de datos en MySQL:

Tabla Componentes

localhost:3000/listardatos.html

Solicitud de Componentes    Menu Principal

### Componentes

### Lista de Componentes

Nombre:	Marca	Precio de venta	Descripcion	Stock	Acciones
CPUUUUU	INTEL	50000	Cpu	20	<a href="#">Eliminar</a> <a href="#">Modificar</a>

Query 1

```

15 id int primary key auto_increment,
16 NombreC VARCHAR(75) NOT NULL,
17 MarcaC VARCHAR(100) NOT NULL,
18 PrecioC int NOT NULL,
19 DescripcionC VARCHAR(100) NOT NULL,
20 CantidadC int NOT NULL
21 );
22
23 • select * from Componentes;

```

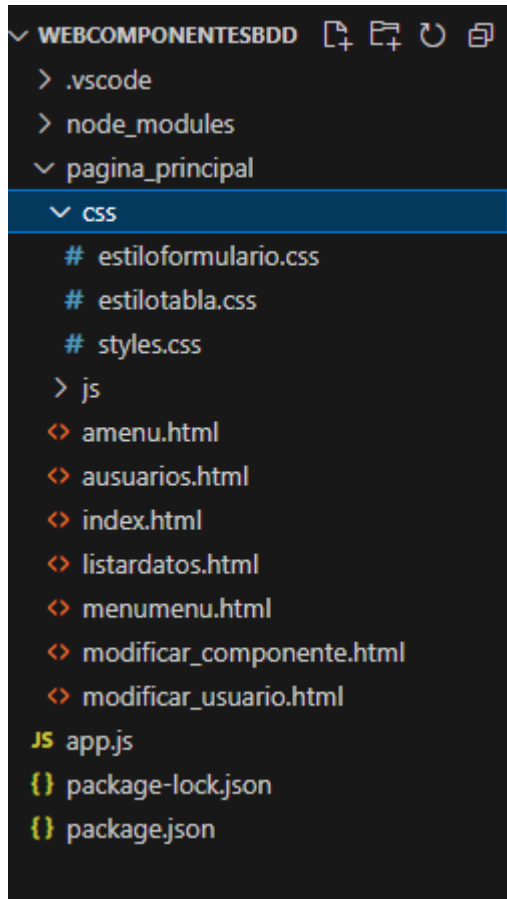
Result Grid

	id	NombreC	MarcaC	PrecioC	DescripcionC	CantidadC
▶	3	CPUUUUU	INTEL	50000	Cpu	20

Componentes 3

## Código:

Como se menciona en primera instancia la página funciona gracias a “Node” y un app.js:



Cada HTML está conectado a su correspondiente CSS para brindarle el estilo correspondiente.

El “app.js” permite las acciones realizadas en los HTML para que se reflejen como se desea en MySQL, como primera instancia permite la conexión:

```

JS app.js > ...
1  //Libreria de Express
2  const express = require('express');
3  //Libreria Path
4  const path = require('path');
5  //Libreria Mysql
6  const mysql = require('mysql');
7
8  const app = express();
9  const port = 3000;
10
11 // Configurar la conexión a la base de datos
12 const connection = mysql.createConnection({
13   host: '127.0.0.1',
14   user: 'root',
15   password: '',
16   database: 'CompV'
17 });
18
19 //Verificacion de errores para validar si la conexion es correcta
20 connection.connect((err) => {
21   if (err) {
22     console.error('Error de conexión a la base de datos: ' + err.stack);
23     return;
24   }
25   console.log('Conexión exitosa a la base de datos.');
```

Además, como otras acciones como Eliminar – Agregar – Modificar.

Se adjunta conjunto a este archivo PDF, la carpeta para su revisión en Visual Studio Code y el SCRIPT para la Base de datos (Recordar conectar en apartado app.js).