



## I. PORTADA

Tema:	Proyecto Final-Desarrollo de aplicación distribuida hospitalaria con arquitectura MVC y base de datos distribuida.
Unidad de Organización Curricular:	PROFESIONAL
Nivel y Paralelo:	Quinto – “A”
Alumnos participantes:	Acosta Nuñez Edison Joel Díaz Mera Paulo Cesar Tonguino Nuñez Diego Alexander Ulloa Yáñez Xavier Alexander
Asignatura:	Sistemas de Bases de Datos Distribuidas
Docente:	Ing. Jose Caiza, Mg.

## II. INFORME DE GUÍA PRÁCTICA

### 2.1 Objetivos

#### General:

Desarrollar una aplicación distribuida que utilice la arquitectura MVC para gestionar la información de un sistema hospitalario integrado por múltiples centros médicos, aplicando conceptos de replicación, fragmentación y distribución de bases de datos en un entorno de contenedores.

#### Específicos:

- Implementar una arquitectura distribuida en la nube usando máquinas virtuales y contenedores Docker.
- Configurar una base de datos distribuida con replicación sincrónica y fragmentación horizontal.
- Desarrollar una API RESTful funcional con arquitectura MVC y una interfaz web para administración hospitalaria.

### 2.2 Modalidad

Práctica Asincrónica

### 2.3 Tiempo de duración

Presenciales: 1 hora

No presenciales: 6 horas

### 2.4 Instrucciones

- Configuración de VMs en la nube.
- Instalación y replicación de bases de datos distribuidas.
- Desarrollo de backend MVC con Node.js y Express.
- Diseño de interfaces web con Vue.js y React.
- Pruebas y despliegue en Azure DevOps.

### 2.5 Listado de equipos, materiales y recursos

- Computadoras personales con conexión a internet
- Servidores virtuales (máquinas virtuales)
- Entorno de desarrollo (Visual Studio, Azure Data Studio)



- Docker instalado en los nodos
- Navegador web (Chrome/Edge/Firefox)

TAC (Tecnologías para el Aprendizaje y Conocimiento) empleados en la guía práctica:

- Plataformas educativas
- Simuladores y laboratorios virtuales
- Aplicaciones educativas
- Recursos audiovisuales
- Gamificación
- Inteligencia Artificial

Otros (Específico): Azure DevOps, Node.js, Docker

## 2.6 Actividades por desarrollar

### 1. Creación de Entornos Virtuales

- Crear tres máquinas virtuales (VM) que representen distintos centros médicos: Quito, Guayaquil y Cuenca.
- Asignar nombres específicos a cada VM y configurar dos adaptadores de red: uno para conexión remota y otro para interconexión entre servidores.

### 2. Configuración de Bases de Datos Distribuidas

- Instalar y configurar SQL Server en cada máquina virtual.
- Verificar que los servicios de bases de datos estén activos y que el puerto de comunicación (1433) esté abierto.
- Probar la conexión utilizando el usuario sa ejecutando consultas básicas desde la consola.

### 3. Configuración de la Replicación de Datos

- Configurar el servidor Quito como publicador y distribuidor.
- Asignar al servidor Guayaquil como distribuidor.
- Configurar el servidor Cuenca también como publicador.
- Establecer y verificar la conectividad peer-to-peer entre los tres servidores, garantizando la replicación de datos.

### 4. Pruebas de Replicación

- Realizar consultas en la base de datos del servidor Quito para comprobar la disponibilidad de los datos.
- Validar que los datos se repliquen correctamente en los servidores Guayaquil y Cuenca.
- Insertar nuevos registros en Quito y verificar su replicación en los demás nodos.

### 5. Desarrollo del Backend en Arquitectura MVC

- Crear una API RESTful utilizando Node.js y Express, aplicando el patrón Modelo-Vista-Controlador (MVC).
- Implementar modelos, controladores y rutas necesarias para la gestión de las entidades: centros médicos, médicos, especialidades, empleados y consultas médicas.
- Documentar los endpoints usando Swagger.

### 6. Desarrollo del Frontend

- Desarrollar una interfaz administrativa en Vue.js para la gestión de usuarios, médicos y consultas.
- Crear una segunda interfaz en React destinada a la gestión de consultas médicas por parte de los hospitales.
- Integrar ambas interfaces con los servicios de la API RESTful.



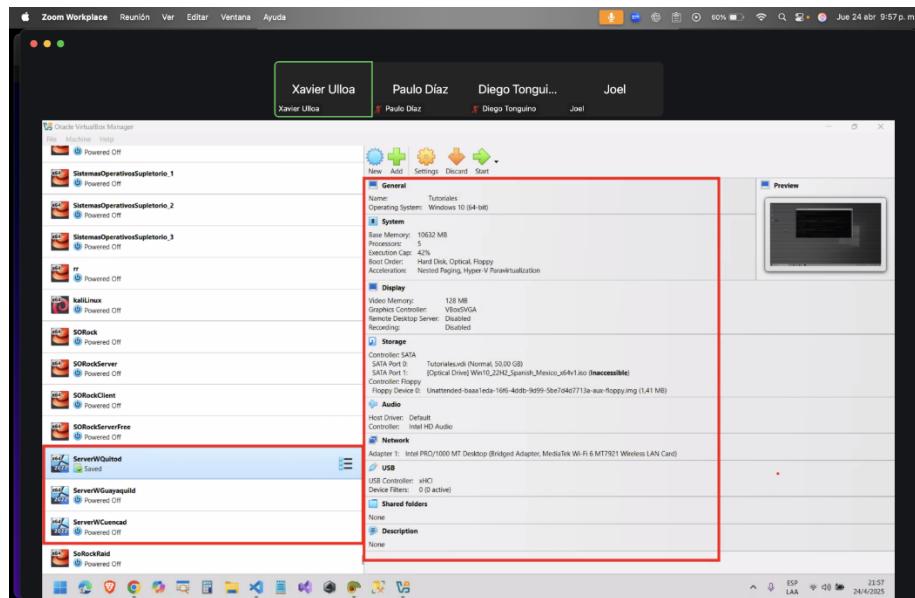
## 7. Despliegue y Automatización

- Utilizar Azure DevOps para gestionar el proyecto, configurar pipelines de integración continua (CI) y despliegue continuo (CD).
  - Desplegar las máquinas virtuales y contenedores Docker en la nube Azure.
  - Implementar平衡adores de carga para garantizar disponibilidad y escalabilidad.
8. Pruebas de Funcionalidad e Integración
- Realizar pruebas de replicación de datos entre bases distribuidas.
  - Verificar el funcionamiento correcto de todos los endpoints de la API RESTful.
  - Validar la funcionalidad y la experiencia de usuario de las interfaces web desarrolladas.

### 2.7 Resultados Obtenidos

#### 2.7.1 Creación de Entornos Virtuales

Se crearon tres entornos virtuales, representando las ciudades: Quito, Guayaquil y Cuenca. Para ello, se utilizó **Windows Server Core 2022** como sistema operativo en cada máquina virtual.

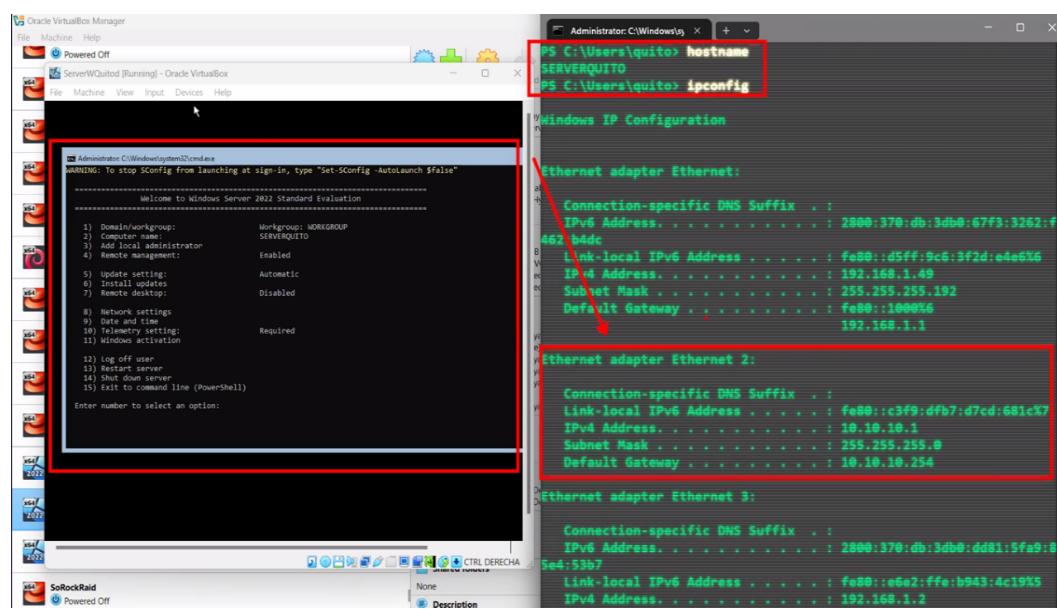


#### 2.7.2 Configuración de la MV Quito

- Se asignó el nombre “Quito” a la máquina virtual.
- Se integraron dos adaptadores de red: uno para permitir la conexión remota dentro de la misma red y otro para facilitar la interconexión con los demás servidores.



**UNIVERSIDAD TÉCNICA DE AMBATO**  
**FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL**  
**CARRERA DE Elige un elemento.**  
**CICLO ACADÉMICO: MARZO – JULIO 2025**



```
Administrator: C:\Windows\system32\cmd.exe
WARNING! To stop SConfig from launching at sign-in, type "Set-SConfig -Autolaunch $False"
Welcome to Windows Server 2022 Standard Evaluation
-----
1) Domain/Workgroup: WORKGROUP
2) Computer name: SERVERQUITO
3) Add local administrator
4) Remote management: Enabled
5) Update setting: Automatic
6) Install updates: Disabled
7) Remote desktop: Required
8) Network settings
9) Date and time
10) Telemetry setting: Windows activation
11) Windows activation
12) Log off user
13) Restart server
14) Shut down server
15) Exit to command line (PowerShell)

Enter number to select an option:
```

```
Administrator: C:\Windows> hostname
SERVERQUITO
Administrator: C:\Windows> ipconfig

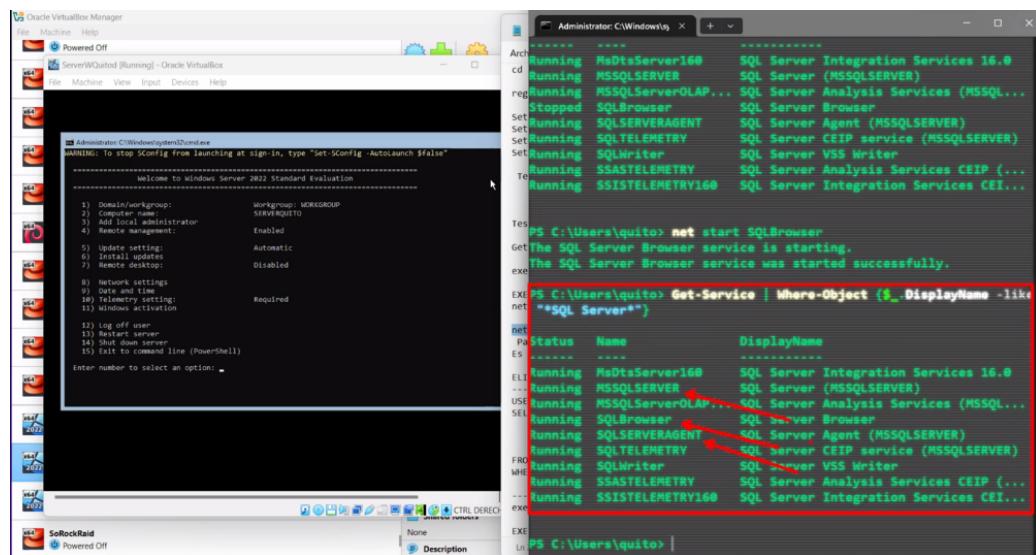
Windows IP Configuration

Ethernet adapter Ethernet:
   Connection-specific DNS Suffix . .
   IPv6 Address . . . . . 2800:370:db:3db0:67f9:3262:fe
   462 b4dc
   Link-local IPv6 Address . . . . : fe80::d5ff:9c6:3f2d:a4e6%6
   IP4 Address . . . . . 192.168.1.49
   Subnet Mask . . . . . 255.255.255.192
   Default Gateway . . . . . 192.168.1.1

Ethernet adapter Ethernet 2:
   Connection-specific DNS Suffix . .
   Link-local IPv6 Address . . . . : fe80::c3f9:dfb7:d7cd:681c%7
   IPv4 Address . . . . . 10.10.10.1
   Subnet Mask . . . . . 255.255.255.0
   Default Gateway . . . . . 10.10.10.254

Ethernet adapter Ethernet 3:
   Connection-specific DNS Suffix . .
   IPv6 Address . . . . . 2800:370:db:3db0:dd81:5fa9%8
   5e4:5b67
   Link-local IPv6 Address . . . . : fe80::e6e2:ffe:b943:4c19%5
   IPv4 Address . . . . . 192.168.1.2
```

- Se verificó la disponibilidad del servicio de SQL Server.



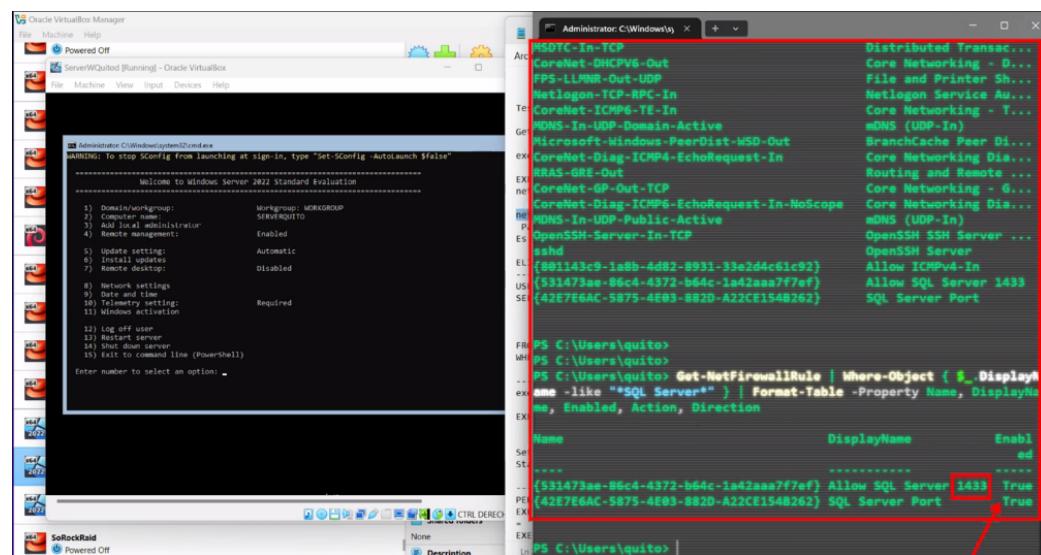
```
Administrator: C:\Windows\system32\cmd.exe
WARNING! To stop SConfig from launching at sign-in, type "Set-SConfig -Autolaunch $False"
Welcome to Windows Server 2022 Standard Evaluation
-----
1) Domain/Workgroup: WORKGROUP
2) Computer name: SERVERQUITO
3) Add local administrator
4) Remote management: Enabled
5) Update setting: Automatic
6) Install updates: Disabled
7) Remote desktop: Required
8) Network settings
9) Date and time
10) Telemetry setting: Windows activation
11) Windows activation
12) Log off user
13) Restart server
14) Shut down server
15) Exit to command line (PowerShell)

Enter number to select an option:
```

```
Administrator: C:\Windows> net start SQLBrowser
The SQL Server Browser service is starting.
The SQL Server Browser service was started successfully.

Administrator: C:\Windows> Get-Service | Where-Object { $_.DisplayName -like "*SQL Server*" }
Status Name DisplayName
---- -- -
Running MsDtsServer160 SQL Server Integration Services 16.0
Running MSSQLSERVER SQL Server (MSSQLSERVER)
Running MSSQLServerOLAP... SQL Server Analysis Services (MSSQL...
Stopped SQLBrowser SQL Server Browser
Running SQLSERVERAGENT SQL Server Agent (MSSQLSERVER)
Running SQLEMETRICS SQL Server CEIP service (MSSQLSERVER)
Running SSQWriter SQL Server VSS Writer
Running SSASTELEMETRY SQL Server Analysis Services CEIP (...
Running SSISTELEMETRY160 SQL Server Integration Services CEI...
```

- Se comprobó la apertura de los puertos necesarios, en especial el **puerto 1433**, esencial para la conexión con la base de datos.



```
Administrator: C:\Windows\system32\cmd.exe
WARNING! To stop SConfig from launching at sign-in, type "Set-SConfig -Autolaunch $False"
Welcome to Windows Server 2022 Standard Evaluation
-----
1) Domain/Workgroup: WORKGROUP
2) Computer name: SERVERQUITO
3) Add local administrator
4) Remote management: Enabled
5) Update setting: Automatic
6) Install updates: Disabled
7) Remote desktop: Required
8) Network settings
9) Date and time
10) Telemetry setting: Windows activation
11) Windows activation
12) Log off user
13) Restart server
14) Shut down server
15) Exit to command line (PowerShell)

Enter number to select an option:
```

```
Administrator: C:\Windows> netstat -an | find "1433"
  TCP    0.0.0.0:1433           0.0.0.0:0              LISTENING
Administrator: C:\Windows> Get-NetFirewallRule | Where-Object { $_.DisplayName -like "SQL Server*" } | Format-Table -Property Name, DisplayName, Enabled, Action, Direction
Name                DisplayName          Enabled
--                -----
{5b1473ae-86c4-4372-b64c-1a42aaa7f7ef} Allow SQL Server 1433 True
{42e7e6ac-5875-4e03-882d-a22ce154b262} SQL Server Port True
```



- Se realizó una conexión con el usuario sa y se ejecutaron consultas básicas desde consola para validar el funcionamiento del motor de base de datos.

The screenshot shows a split-screen interface. On the left is a web browser displaying the 'ZENTRUM MEDICAL' website, which features a photo of five medical professionals. On the right is a terminal window running on Windows. The terminal session is as follows:

```
PS C:\Users\quito> sqlcmd -U sa -P Serverquito
1> USE ZentrumMedicalDB
2> GO
3> Changed database context to 'ZentrumMedicalDB'.
4> SELECT*FROM roles;
5> go
roles_serial roles_name
1002 Enfermero
3 Guardia
1 Médico General
3 Paramédico
1003 Pasiente
(5 rows affected)
6> SELECT*FROM users;
7> go
users_serial users_id
users_firstName users_middleName users_lastName users_email
users_secondLastName users_email
users_userName
users_password
users_updatedAt
users_dateOfBirth users_createdAt
users_roleSerial users_photo
```

### 2.7.3 Configuración de las MV Guayaquil y Cuenca

- En Guayaquil, se aplicó una configuración similar, integrando la máquina a la red local y asegurando la disponibilidad del servicio

The screenshot shows a split-screen interface. On the left is a web browser displaying the 'ZENTRUM MEDICAL' website, which features a photo of a doctor interacting with a young patient. On the right is a terminal window running on Windows. The terminal session is as follows:

```
PS C:\Users\TEMP> hostname
SERVERGUAYAQUIL
PS C:\Users\TEMP> ipconfig
Windows IP Configuration

Ethernet adapter Ethernet:

Connection-specific DNS Suffix . :
IPv6 Address . . . . . : 2800:370:db:3db0:eba3:ce66:a
c89:6832
Link-local IPv6 Address . . . . . : fe80::8bda:7151:e0e8:fdax5
IPv4 Address . . . . . : 192.168.1.46
Subnet Mask . . . . . : 255.255.255.192
Default Gateway . . . . . : fe80::1000%5
192.168.1.1

Ethernet adapter Ethernet 2:

Connection-specific DNS Suffix . :
Link-local IPv6 Address . . . . . : fe80::7089:e633:ec85:a2e7%6
IPv4 Address . . . . . : 10.10.10.2
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . : 10.10.10.254
PS C:\Users\TEMP>
```

- Cuenca replicó exactamente el mismo procedimiento anterior.



The screenshot shows a web browser with a red border. On the left is a medical facility's website with a blue header and a photo of an operating room. On the right is a terminal window with the following text:

```
PS C:\Users\TEMP> hostname
SERVERCUENCA
PS C:\Users\luis> ipconfig

Windows IP Configuration

Ethernet adapter Ethernet:

Connection-specific DNS Suffix . :
IPv6 Address . . . . . : 2800:370:db:3db0:a99e:9fe7:b
Link-local IPv6 Address . . . . . : fe80::ee2f:aa7e:af07:63f1%5
IPv4 Address . . . . . : 192.168.1.47
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . : fe80::1000%5
                           192.168.1.1

Ethernet adapter Ethernet 2:

Connection-specific DNS Suffix . :
Link-local IPv6 Address . . . . . : fe80::d810:97a4:2aea:4c59%6
IPv4 Address . . . . . : 10.10.10.3
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . : 10.10.10.254

Ethernet adapter Ethernet 3:

Connection-specific DNS Suffix . :
IPv6 Address . . . . . : 2800:370:db:3db0:cec1:8588:7
Link-local IPv6 Address . . . . . : fe80::997:d1cf:30e7:9cf%8
IPv4 Address . . . . . : 192.168.1.12
```

#### 2.7.4 Verificación de Interconexión entre Servidores

- Se comprobó que los tres servidores podían comunicarse entre sí utilizando sus respectivos hostnames.

The screenshot shows a web browser with a red border. On the left is a medical facility's website with a blue header and a photo of a doctor interacting with a child. On the right is a terminal window with the following text:

```
PS C:\Users\quito> ping SERVERGUAYAQUIL
Pinging serverguayaquil [10.10.10.2] with 32 bytes of data:
Reply from 10.10.10.2: bytes=32 time<1ms TTL=128
Reply from 10.10.10.2: bytes=32 time<1ms TTL=128
Reply from 10.10.10.2: bytes=32 time<1ms TTL=128
Reply from 10.10.10.2: bytes=32 time=2ms TTL=128

Ping statistics for 10.10.10.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 2ms, Average = 0ms
PS C:\Users\quito> ping SERVERCUENCA
Pinging servercuencia [10.10.10.3] with 32 bytes of data:
Reply from 10.10.10.3: bytes=32 time=2ms TTL=128
Reply from 10.10.10.3: bytes=32 time<1ms TTL=128
Reply from 10.10.10.3: bytes=32 time=1ms TTL=128
Reply from 10.10.10.3: bytes=32 time<1ms TTL=128

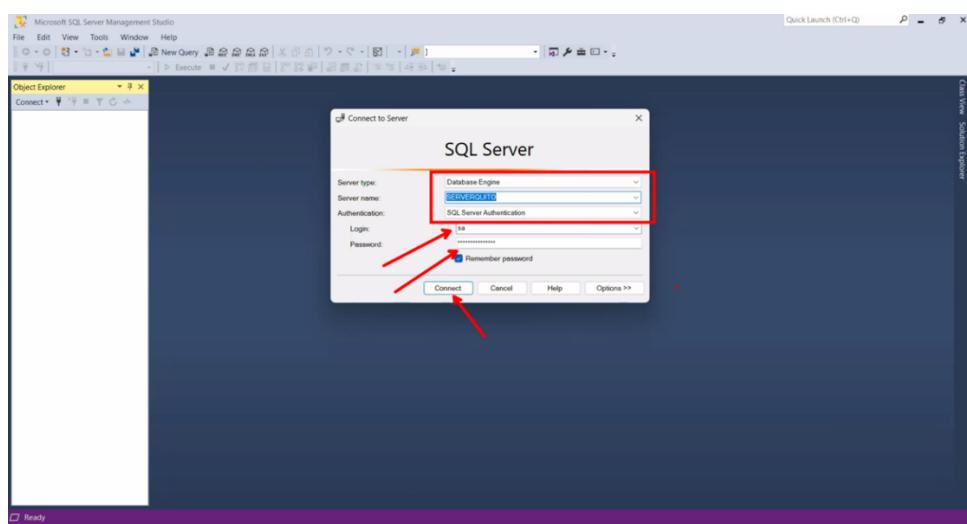
Ping statistics for 10.10.10.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 2ms, Average = 0ms
PS C:\Users\quito> ping SERVERQUITO
Pinging SERVERQUITO [fe80::e6e2:ffe:bf4c19%5] with 32 bytes of data:
Reply from fe80::e6e2:fe:b943:4c19%5: time<1ms
Reply from fe80::e6e2:fe:b943:4c19%5: time<1ms
Reply from fe80::e6e2:fe:b943:4c19%5: time<1ms
Reply from fe80::e6e2:fe:b943:4c19%5: time<1ms

Ping statistics for fe80::e6e2:ffe:b943:4c19%5:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
```

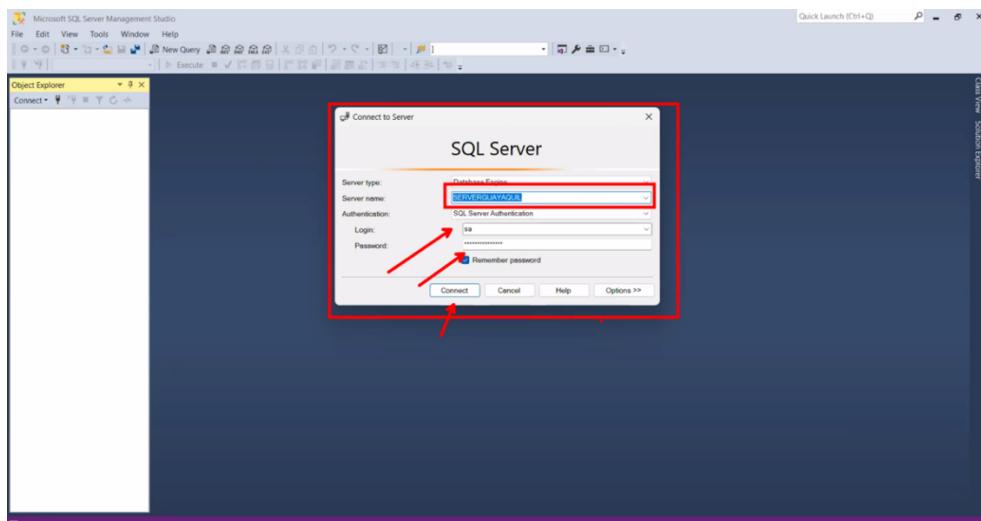
- Se accedió al servidor **Quito** desde la máquina anfitriona mediante su interfaz gráfica para verificar la correcta integración.



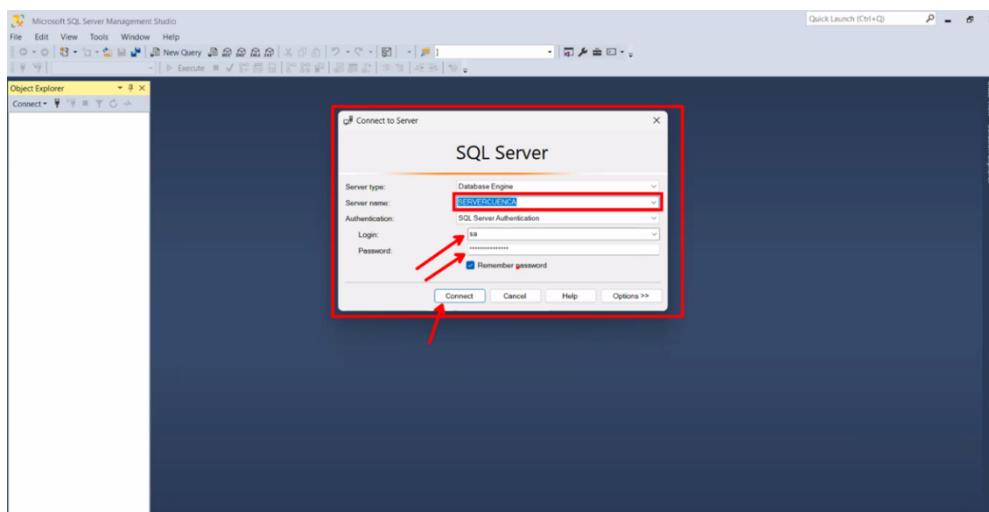
**UNIVERSIDAD TÉCNICA DE AMBATO**  
**FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL**  
**CARRERA DE Elige un elemento.**  
**CICLO ACADÉMICO: MARZO – JULIO 2025**



- Se accedió al servidor **Guayaquil** desde la máquina anfitriona mediante su interfaz gráfica para verificar la correcta integración.

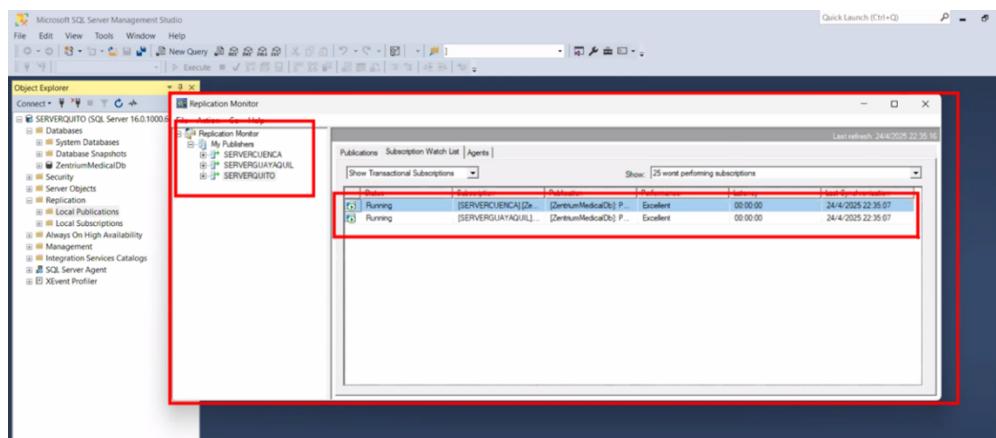


- Se accedió al servidor **Cuenca** desde la máquina anfitriona mediante su interfaz gráfica para verificar la correcta integración.



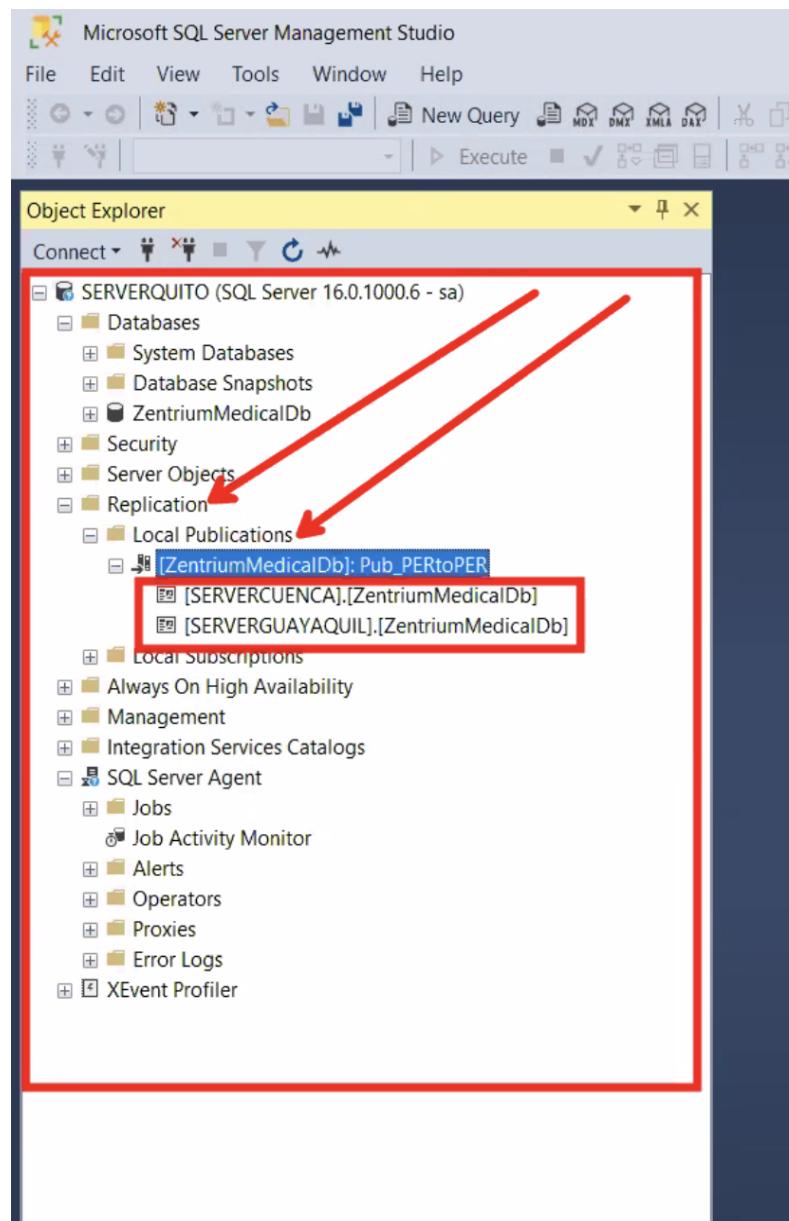


- Se validó la conectividad **peer-to-peer** entre los tres servidores, asegurando un entorno de replicación confiable.



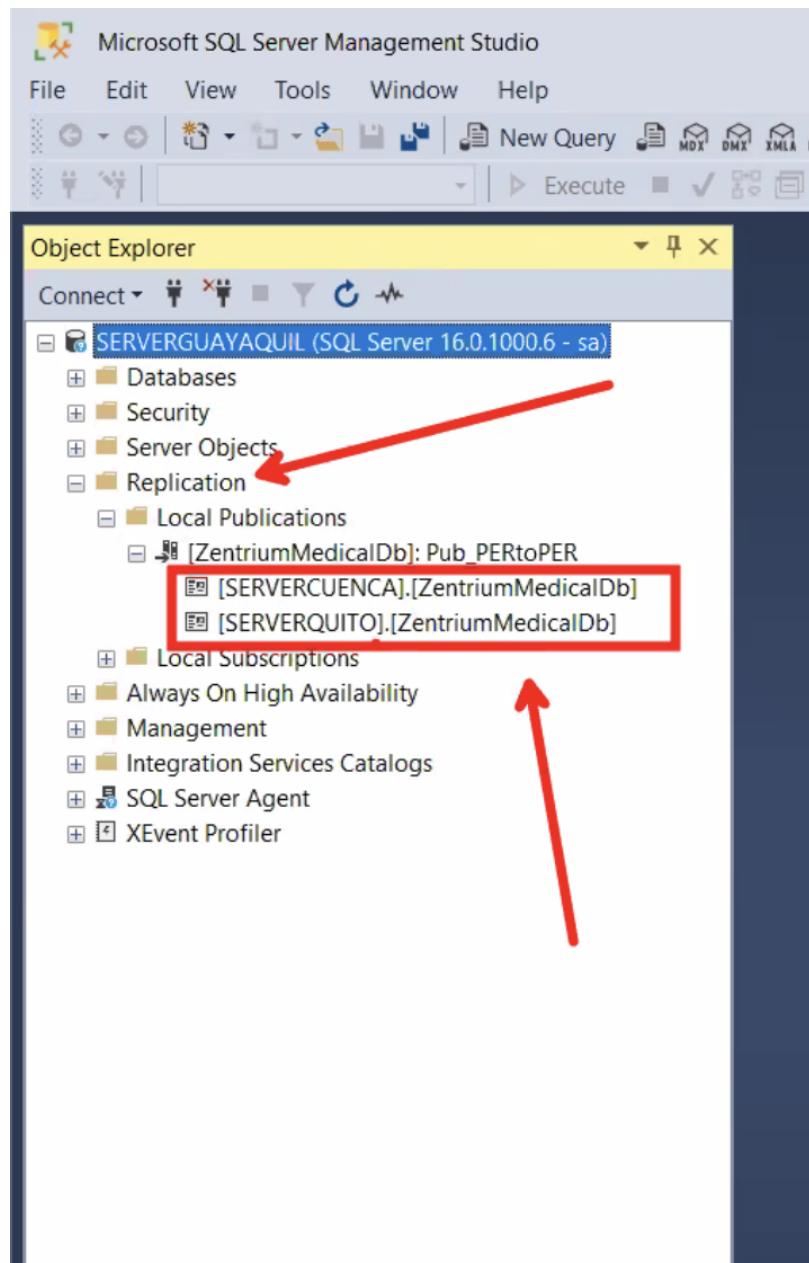
### 2.7.5 Configuración de la Replicación de Datos

- Quito fue configurado como publicador y distribuidor.





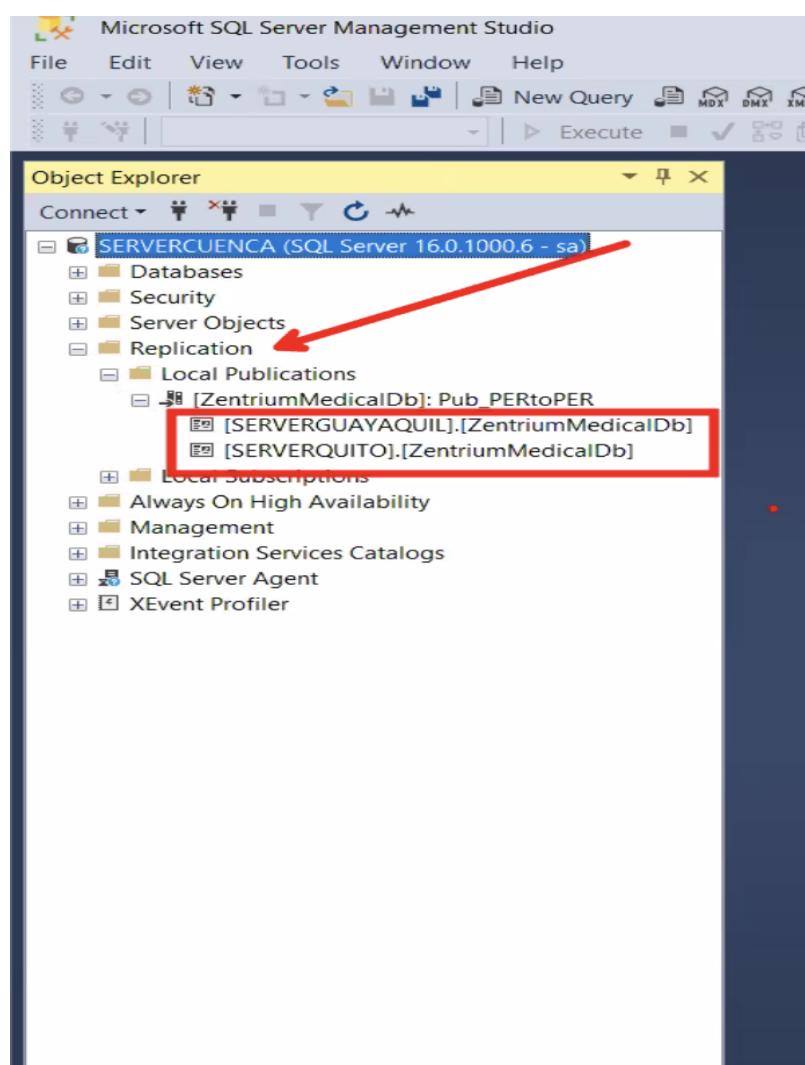
- Guayaquil fue asignado como distribuidor y se verificó su enlace con Quito.



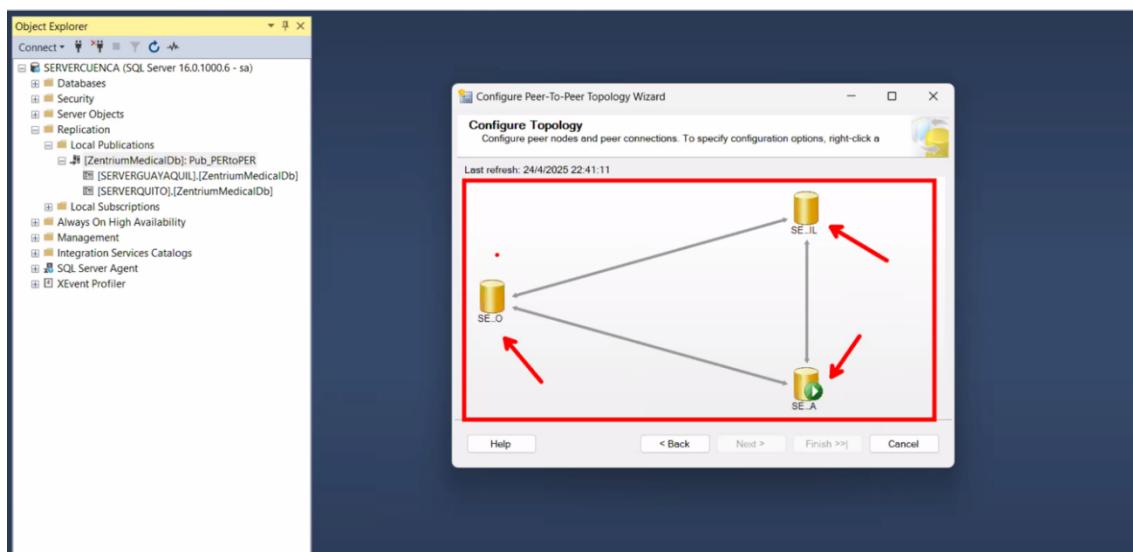
- Cuenca fue también configurado como publicador.



**UNIVERSIDAD TÉCNICA DE AMBATO**  
**FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL**  
**CARRERA DE Elige un elemento.**  
**CICLO ACADÉMICO: MARZO – JULIO 2025**



- Se validó la conectividad **peer-to-peer** entre los tres servidores, asegurando un entorno de replicación confiable.





## 2.7.6 Pruebas de Replicación

- Se realizaron consultas generales en el servidor Quito para verificar la presencia de datos

The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left shows the connection to SERVERQUITO (SQL Server 16.0.1000.6 - sa). The central pane displays a query window with the following code:

```
USE ZentriumMedicalDb;
SELECT * FROM roles;
```

The results pane shows a table with the following data:

roles_serial	roles_name
1002	Enfermero
3	Guardia
1	Médico General
2	Paramédico
1003	Paciente

- Posteriormente, se comprobó la correcta replicación de dichos datos en el servidor Guayaquil.

The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left shows the connection to SERVERGUAYAQUIL (SQL Server 16.0.1000.6 - sa). The central pane displays a query window with the following code:

```
USE ZentriumMedicalDb;
SELECT * FROM roles;
```

The results pane shows a table with the same data as the previous screenshot:

roles_serial	roles_name
1002	Enfermero
3	Guardia
1	Médico General
2	Paramédico
1003	Paciente



**UNIVERSIDAD TÉCNICA DE AMBATO**  
**FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL**  
**CARRERA DE Elige un elemento.**  
**CICLO ACADÉMICO: MARZO – JULIO 2025**



- Se comprobó la correcta replicación de dichos datos en el servidor Guayaquil.

The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left shows the connection to SERVERCUENCA (SQL Server 16.0.1000.6 - sa). The Results pane on the right displays the output of the following SQL query:

```
USE ZentriumMedicalDb;
SELECT * FROM roles;
```

The results show the following data:

roles_serial	roles_name
1	Enfermero
2	Guardia
3	Médico General
4	Paramédico
5	Paciente

- Se ejecutó una inserción de datos (registro de una secretaria) en el servidor Quito, validando su replicación automática en los otros dos servidores.

The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left shows the connection to SERVERQUITO (SQL Server 16.0.1000.6 - sa). The Results pane on the right displays the output of the following SQL query:

```
USE ZentriumMedicalDb;
INSERT INTO roles
VALUES('Secretaria');
SELECT * FROM roles;
```

The results show the following data:

roles_serial	roles_name
1	Enfermero
2	Guardia
3	Médico General
4	Paramédico
5	Paciente
6	Secretaria



**UNIVERSIDAD TÉCNICA DE AMBATO**  
**FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL**  
**CARRERA DE Elige un elemento.**  
**CICLO ACADÉMICO: MARZO – JULIO 2025**



• Verificación GUAYAQUIL

The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left shows the connection to SERVERGUAYAQUIL (SQL Server 16.0.1000.6 - sa). The Query Editor window contains the following SQL code:

```
USE ZentriumMedicalDb;
SELECT * FROM roles;
```

The Results pane displays the following table of roles:

roles_serial	roles_name
1002	Enfermero
3	Guardia
1	Médico General
2	Paramédico
1003	Paciente
1004	Secretaria

• Verificación CUENCA:

The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left shows the connection to SERVERCUENCA (SQL Server 16.0.1000.6 - sa). The Query Editor window contains the following SQL code:

```
USE ZentriumMedicalDb;
SELECT * FROM roles;
```

The Results pane displays the following table of roles:

roles_serial	roles_name
1002	Enfermero
3	Guardia
1	Médico General
2	Paramédico
1003	Paciente
1004	Secretaria



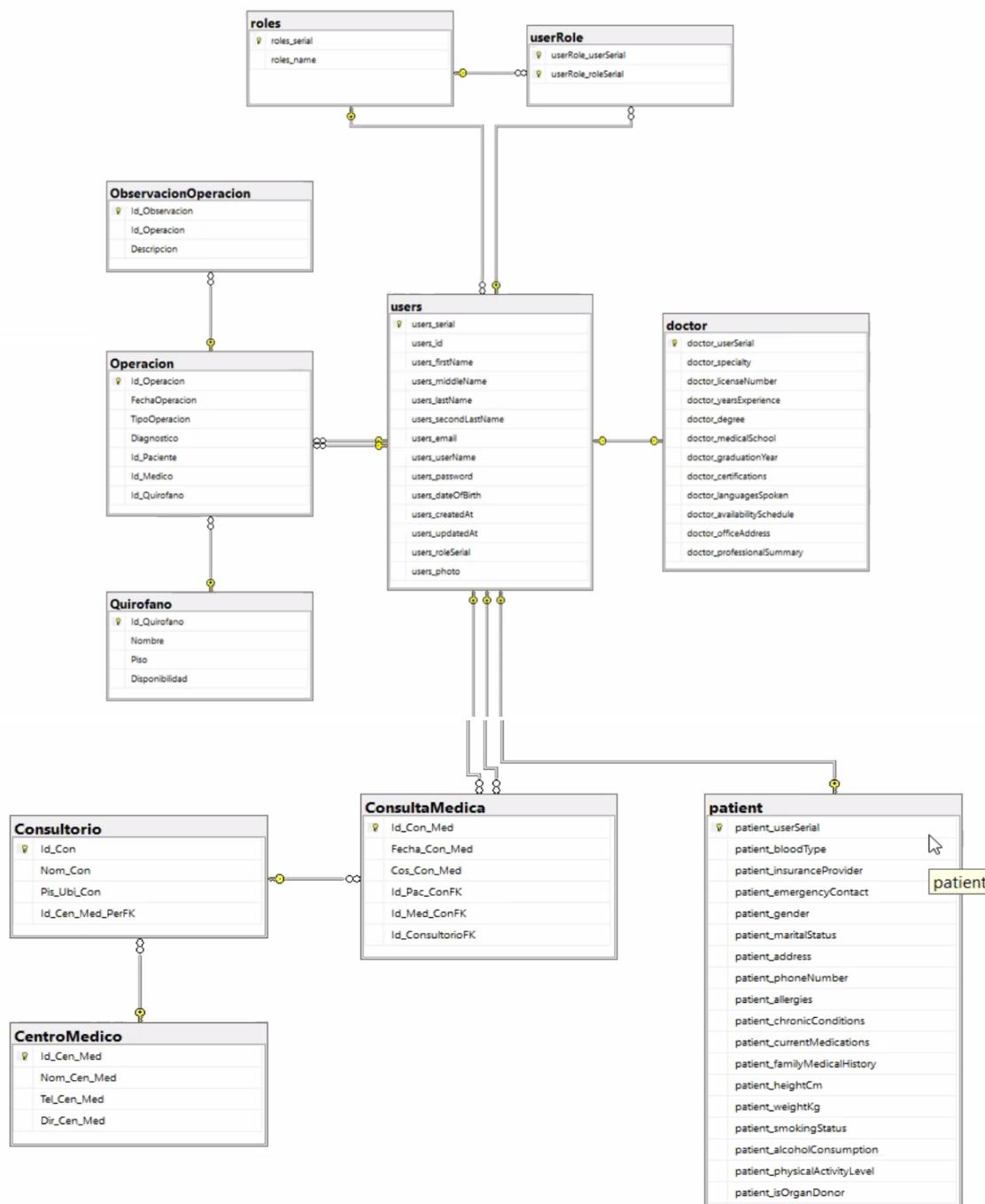
### 2.7.7 Modelo de Base de Datos Hospitalaria Distribuida

Para el desarrollo de la aplicación hospitalaria, se diseñó una base de datos relacional distribuida que representa la estructura funcional de un sistema hospitalario con múltiples centros médicos. Esta base de datos fue replicada y fragmentada horizontalmente entre los nodos virtuales (Quito, Guayaquil y Cuenca), garantizando consistencia y disponibilidad de la información.

El modelo entidad-relación contempla las siguientes entidades principales:

- **users:** Tabla que almacena la información de los usuarios del sistema, incluyendo credenciales, tipo de usuario y estado.
- **roles y userRole:** Gestionan los distintos perfiles de usuario (como administrador, médico, secretaria), permitiendo asignaciones flexibles mediante relaciones muchos-a-muchos.
- **doctor:** Contiene detalles profesionales de los médicos registrados, tales como especialidades, experiencia y biografía.
- **patient:** Almacena información detallada de los pacientes, incluyendo datos personales, historial clínico y características médicas específicas.
- **CentroMedico y Consultorio:** Representan la infraestructura médica disponible, permitiendo registrar centros médicos y los consultorios disponibles en cada uno.
- **ConsultaMedica:** Tabla central que registra las consultas realizadas, incluyendo fecha, hora, profesional responsable y consultorio asignado.
- **Operacion y ObservationOperation:** Controlan las operaciones quirúrgicas realizadas, con sus respectivas observaciones médicas.
- **Quirofano:** Define la disponibilidad y características de los quirófanos en los distintos centros médicos.

Este diseño asegura la escalabilidad del sistema y permite una distribución eficiente de la carga entre las réplicas. Gracias a la replicación sincrónica, cualquier inserción o modificación realizada en un nodo se propaga automáticamente a los demás, como se evidenció en la verificación de los datos replicados en el servidor Cuenca, donde se observa correctamente la tabla roles con sus respectivos registros.



### 2.7.8 Desarrollo del Backend y API

- Se implementaron las API necesarias utilizando Node.js y Express, incluyendo las clases fundamentales como Home, Login, y Register.



**UNIVERSIDAD TÉCNICA DE AMBATO**  
**FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL**  
**CARRERA DE Elige un elemento.**  
**CICLO ACADÉMICO: MARZO – JULIO 2025**



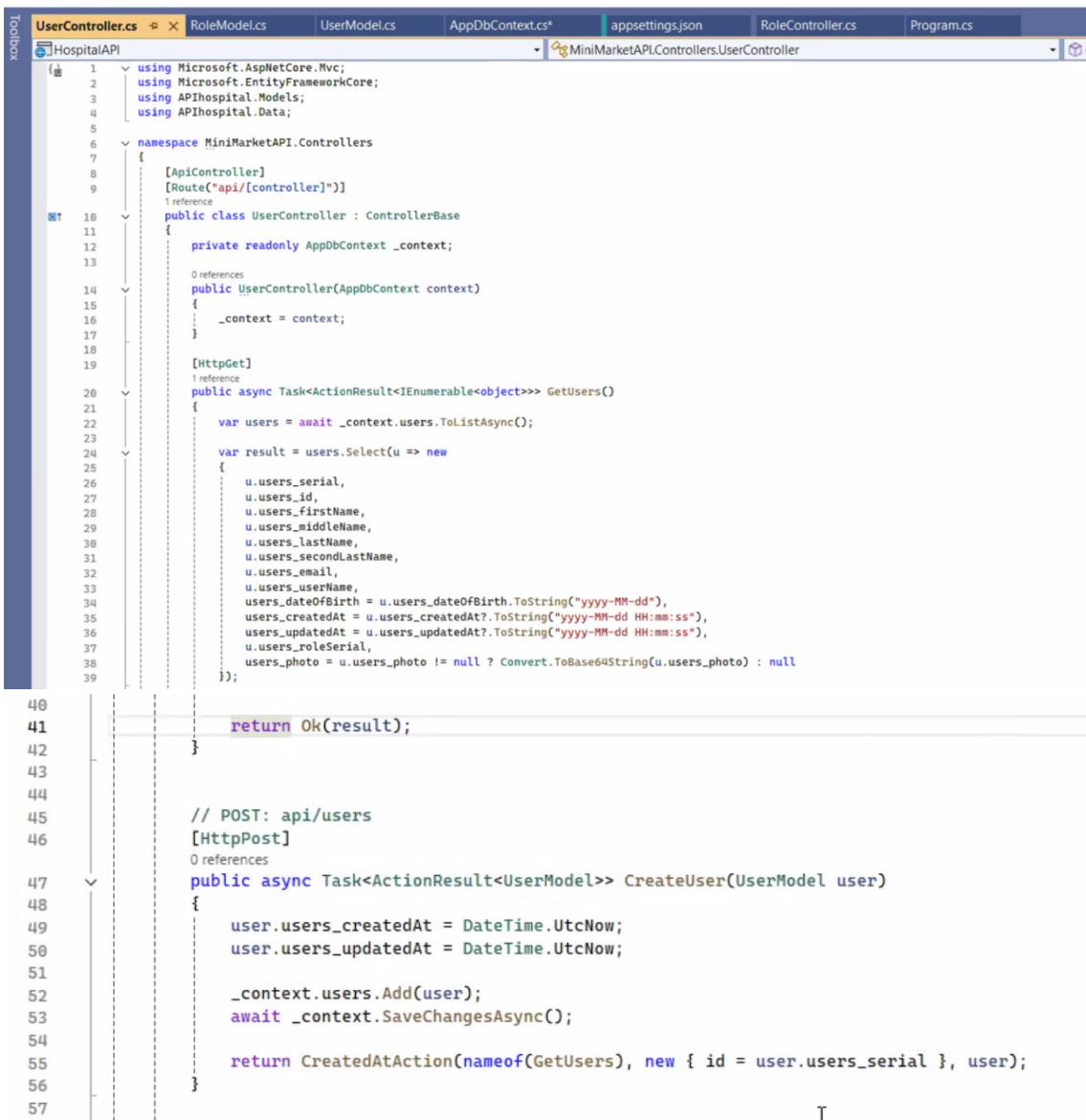
```
1 "Logging": {
2     "LogLevel": {
3         "Default": "Information",
4         "Microsoft.AspNetCore": "Warning"
5     }
6 }
7
8 "AllowedHosts": "*",
9
10 "ConnectionStrings": {
11     "QuitoServer": "Server=SERVERQUITO;Database=ZentriumMedicalDb;User Id=sa;Password=Serverquito;$TrustServerCertificate=True;Connection Timeout=30;"
12 }
13
14 }
```

```
1 using System.ComponentModel.DataAnnotations;
2
3 namespace APIHospital.Models
4 {
5     public class UserModel
6     {
7         [Key]
8         public int users_serial { get; set; }
9         public string users_id { get; set; }
10        public string users_firstname { get; set; }
11        public string? users_middleName { get; set; } // puede ser NULL
12        public string? users_lastname { get; set; }
13        public string? users_secondLastName { get; set; } // puede ser NULL
14        public string? users_email { get; set; }
15        public string? users_userName { get; set; }
16        public byte[]? users_password { get; set; }
17        public DateTime? users_dateOfBirth { get; set; }
18        public DateTime? users_createdAt { get; set; } // puede ser NULL
19        public DateTime? users_updatedAt { get; set; } // puede ser NULL
20        public int? users_roleSerial { get; set; }
21        public byte[]? users_photo { get; set; }
22
23        public UserModel()
24        {
25
26        }
27
28        public UserModel(int users_serial, string users_id, string users_firstname, string? users_middleName, string? users_lastname, string? users_secondLastName, string users_email, string users_userName, byte[]? users_password)
29        {
30            this.users_serial = users_serial;
31            this.users_id = users_id;
32            this.users_firstname = users_firstname;
33            this.users_middleName = users_middleName;
34            this.users_lastname = users_lastname;
35            this.users_secondLastName = users_secondLastName;
36            this.users_email = users_email;
37            this.users_userName = users_userName;
38            this.users_password = users_password;
39            this.users_dateOfBirth = users_dateOfBirth;
40            this.users_createdAt = users_createdAt;
41            this.users_updatedAt = users_updatedAt;
42            this.users_roleSerial = users_roleSerial;
43            this.users_photo = users_photo;
44        }
45
46    }
47
48 }
```

```
1 using Microsoft.EntityFrameworkCore;
2 using Microsoft.EntityFrameworkCore.SqlServer.Query.Internal;
3 using APIHospital.Models;
4 using System.Collections.Generic;
5
6 namespace APIHospital.Data
7 {
8     public class AppDbContext : DbContext
9     {
10        public AppDbContext(DbContextOptions<AppDbContext> options) : base(options) { }
11
12        //Tablas
13        public DbSet<RoleModel> roles { get; set; }
14        public DbSet<UserModel> users { get; set; }
15    }
16
17 }
```



**UNIVERSIDAD TÉCNICA DE AMBATO**  
**FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL**  
**CARRERA DE Elige un elemento.**  
**CICLO ACADÉMICO: MARZO – JULIO 2025**



```
UserController.cs  RoleModel.cs  UserModel.cs  ApplicationDbContext.cs*  appsettings.json  RoleController.cs  Program.cs
[ApiController]
[Route("api/[controller]")]
public class UserController : ControllerBase
{
    private readonly AppDbContext _context;

    public UserController(AppDbContext context)
    {
        _context = context;
    }

    [HttpGet]
    public async Task<ActionResult<IEnumerable<object>>> GetUsers()
    {
        var users = await _context.users.ToListAsync();

        var result = users.Select(u => new
        {
            u.users_serial,
            u.users_id,
            u.users(firstName,
            u.users_middleName,
            u.users_lastName,
            u.users_secondLastName,
            u.users_email,
            u.users_userName,
            users_dateOfBirth = u.users_dateOfBirth.ToString("yyyy-MM-dd"),
            users_createdAt = u.users_createdAt?.ToString("yyyy-MM-dd HH:mm:ss"),
            users_updatedAt = u.users_updatedAt?.ToString("yyyy-MM-dd HH:mm:ss"),
            u.users_roleSerial,
            users_photo = u.users_photo != null ? Convert.ToString(u.users_photo) : null
        });
    }

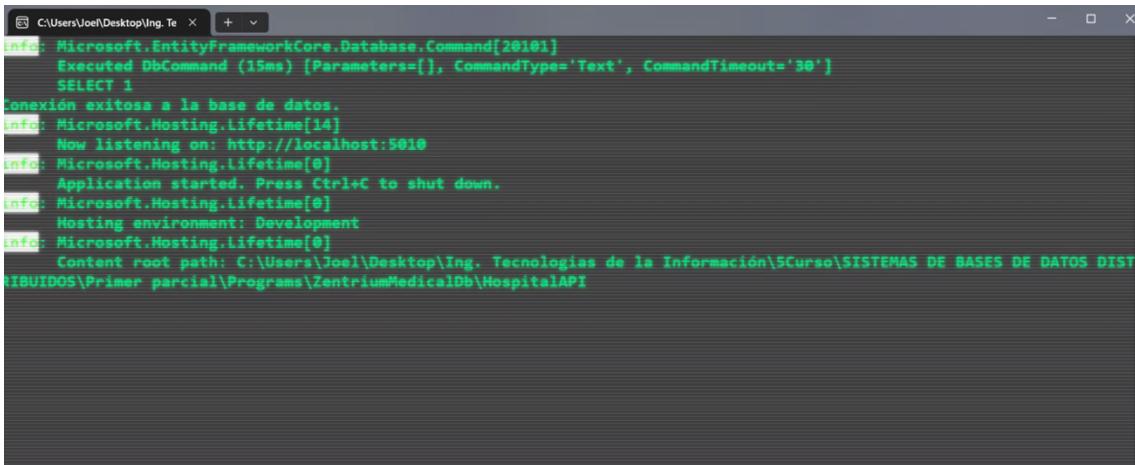
    return Ok(result);
}

// POST: api/users
[HttpPost]
public async Task<ActionResult<UserModel>> CreateUser(UserModel user)
{
    user.users_createdAt = DateTime.UtcNow;
    user.users_updatedAt = DateTime.UtcNow;

    _context.users.Add(user);
    await _context.SaveChangesAsync();

    return CreatedAtAction(nameof(GetUsers), new { id = user.users_serial }, user);
}
```

- Se verificó que todos los servicios de hosting se encontraban funcionando correctamente.



```
C:\Users\Joel\Desktop\Ing. Te
Info: Microsoft.EntityFrameworkCore.Database.Command[20101]
      Executed DbCommand (15ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
      SELECT 1
Conexión exitosa a la base de datos.
info: Microsoft.Hosting.Lifetime[14]
      Now listening on: http://localhost:5010
info: Microsoft.Hosting.Lifetime[0]
      Application started. Press Ctrl+C to shut down.
info: Microsoft.Hosting.Lifetime[0]
      Hosting environment: Development
info: Microsoft.Hosting.Lifetime[0]
      Content root path: C:\Users\Joel\Desktop\Ing. Tecnologías de la Información\5Curso\SISTEMAS DE BASES DE DATOS DISTRIBUIDOS\Primer parcial\Programs\ZentriumMedicalDb\HospitalAPI
```



**UNIVERSIDAD TÉCNICA DE AMBATO**  
**FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL**  
**CARRERA DE Elige un elemento.**  
**CICLO ACADÉMICO: MARZO – JULIO 2025**



- Se revisó y probó la documentación generada en **Swagger**



## 2.7.9 Configuración del Entorno de Desarrollo

- Se trabajó en **Visual Studio Code**, donde se realizó la implementación del backend bajo el patrón MVC.
- Se desarrolló una interfaz de usuario funcional que interactúa con los servicios REST expuestos.

Clase **home** su código y su interfaz:

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows the project structure under "app-ZentrumMedicalDb". The "src" folder contains "pages" (with "Home.jsx" highlighted), "components", "assets" (with "imagePrincipal.jpg" and "logohospital.png"), and "index.css".
- Code Editor:** Displays the code for "Home.jsx". The code includes imports for "react", "react-dom", "styled-components", and "navigation.map". It defines a "const navigation = [...]" array with items like {name: "Pacientes y Visitantes", href: "#"}, {name: "Especialidades", href: "#"}, etc. It also defines a "const Slideshow = () => [...]" function and a "export default function Example() {...}" function.
- Terminal:** Shows the command "npm start" being run.
- Status Bar:** Shows the current file is "Home.jsx" and the status "Cursoreact".

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows the project structure under "app-ZentrumMedicalDb". The "src" folder contains "pages" (with "Home.jsx" highlighted), "components", "assets" (with "imagePrincipal.jpg" and "logohospital.png"), and "index.css".
- Code Editor:** Displays the code for "Home.jsx". A red arrow points to the "navigation" array in the code. The array contains items such as {name: "Pacientes y Visitantes", href: "#"}, {name: "Especialidades", href: "#"}, {name: "Servicios", href: "#"}, {name: "Equipo de Salud", href: "#"}, {name: "Académico", href: "#"}, {name: "Internado", href: "#"}, and {name: "Agencias", href: "#"}.
- Terminal:** Shows the command "npm start" being run.
- Status Bar:** Shows the current file is "Home.jsx" and the status "Cursoreact".



**UNIVERSIDAD TÉCNICA DE AMBATO**  
FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL  
CARRERA DE Elige un elemento.  
CICLO ACADÉMICO: MARZO – JULIO 2025



The screenshot shows a web browser window for 'localhost:5173/#'. At the top, there's a header with links for 'Pacientes y Visitantes', 'Especialidades', 'Servicios', 'Consejos de Salud', 'Academia', 'Internado', 'Agencias', and a red 'Agendar Cita' button. Below the header is a banner for 'ZENTRUM MEDICAL' featuring a photo of medical staff performing surgery. To the right of the banner is text about the new app: 'Conoce la nueva app de Hospital ZentrumMedcial.' It says 'HZentrum APP es más que una simple aplicación. Es tu aliado integral para cuidar tu salud y la de todos tus seres queridos.' Buttons for 'Google Play' and 'App Store' are shown.

- Clase login su código y su interfaz:

A screenshot of a code editor (VS Code) showing the 'Login.js' file. The file contains React component code for a login form. An arrow points to the 'Login.js' file in the left sidebar. The code includes imports for 'useState' and 'useEffect', and defines a state variable 'example' with an initial value of an object containing 'email' and 'password' fields. It uses 'useFormik' from 'formik' to handle form validation and submission. The UI part of the code uses Tailwind CSS classes for styling the input fields and buttons.

A screenshot of a browser window for 'localhost:5173/login'. The page has a dark blue header with the 'Zentrum Medical' logo on the left and the text 'Zentrum Medical' in white on the right. The main content area has a background image of five healthcare professionals in scrubs standing in a hallway. On the right side, there are two input fields: 'Cédula o Pasaporte' and 'Contraseña', each with a placeholder and a 'Forgot password?' link. Below the inputs are two buttons: a blue 'Iniciar sesión' button and a green 'Registrar' button.



Cédula o Pasaporte

Contraseña

Olvidaste tu contraseña?

Iniciar sesión

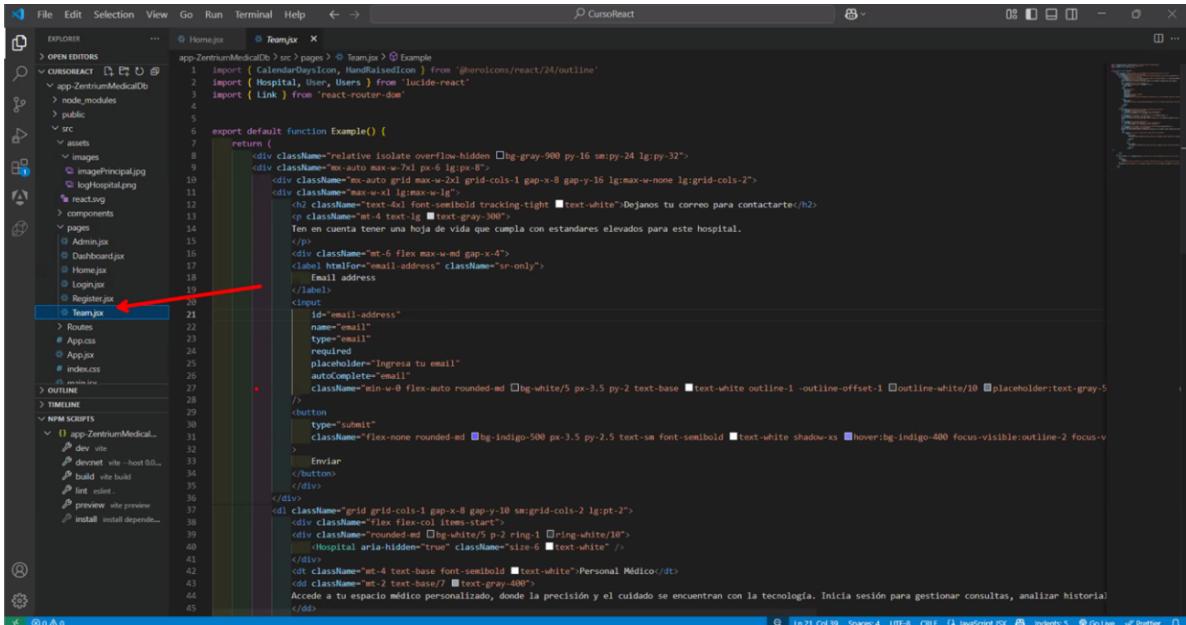
Registrar



**UNIVERSIDAD TÉCNICA DE AMBATO**  
**FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL**  
**CARRERA DE Elige un elemento.**  
**CICLO ACADÉMICO: MARZO – JULIO 2025**



- Clase **team** código y su interfaz:



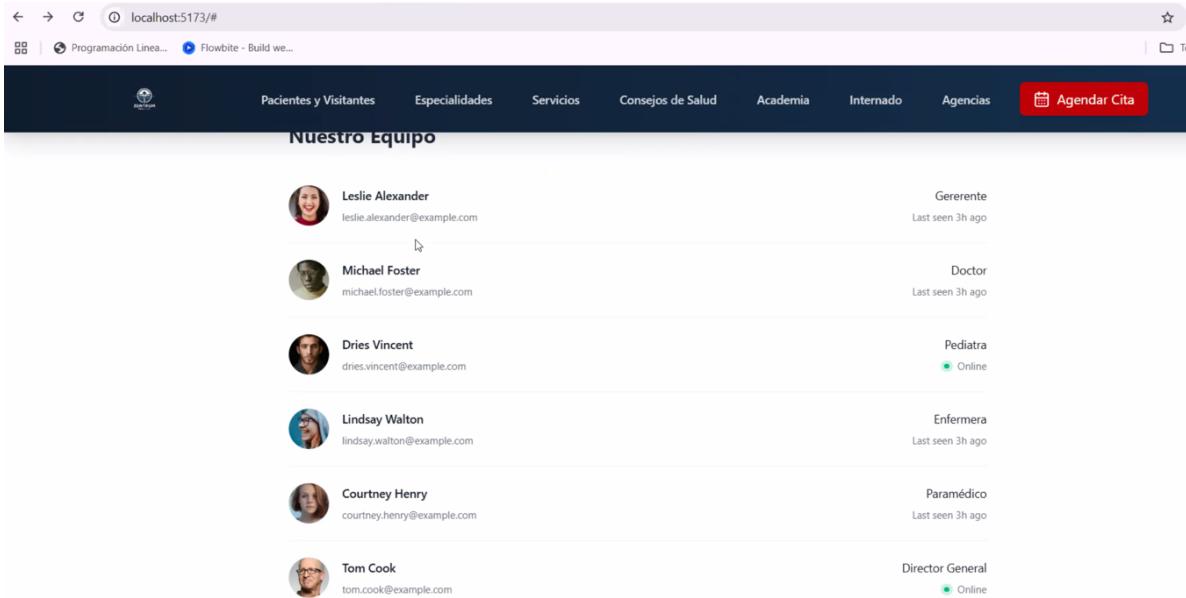
```
File Edit Selection View Go Run Terminal Help < - > Cursoreact
```

```
app-ZenithumMedicalDb>src>pages>Team.jsx>example
1 import { CalendarDayIcon, HandshakeIcon } from '@heroicons/react/24/outline'
2 import { Hospital, User, Users } from 'lucide-react'
3 import { Link } from 'react-router-dom'
4
5
6 export default function Example() {
7   return (
8     <div className="relative isolate overflow-hidden bg-gray-900 py-16 sm:py-24 lg:py-32">
9       <div className="mx-auto max-w-2xl grid max-w-2xl grid-cols-1 gap-x-8 gap-y-16 lg:max-w-none lg:grid-cols-2">
10      <div className="max-w-xl lg:mx-0 lg:grid-cols-1">
11        <h2 className="text-4xl font-semiBold tracking-tight text-white">Déjanos tu correo para contactarte</h2>
12        <p>Ten en cuenta tener una hoja de vida que cumpla con estándares elevados para este hospital.</p>
13      </div>
14      <div className="mt-4 flex max-w-md gap-x-4">
15        <label htmlFor="emailAddress" className="sr-only">Email address</label>
16        <input id="emailAddress" name="email" type="email" placeholder="Ingresa tu email" autoComplete="email" className="min-w-0 flex-auto rounded-md bg-white/5 px-3.5 py-2 text-base text-white outline-1 outline-offset-1 outline-white/10 placeholder:text-gray-500" />
17        <button type="submit" className="flex-none rounded-md bg-indigo-500 px-3.5 py-2.5 text-sm font-semibold text-white shadow-xs hover:bg-indigo-400 focus-visible:outline-2 focus-visible:outline-white/10 placeholder:text-gray-400">Enviar</button>
18      </div>
19    </div>
20  </div>
21
```

Ten en cuenta tener una hoja de vida que cumpla con estándares elevados para este hospital.

Enviado

Accede a tu espacio médico personalizado, donde la precisión y el cuidado se encuentran con la tecnología. Inicia sesión para gestionar consultas, analizar historia clínica y más.



localhost:5173/#

Programación Linea... Flowbite - Build we...

Pacientes y Visitantes Especialidades Servicios Consejos de Salud Academia Internado Agencias Agendar Cita

Nuestro Equipo

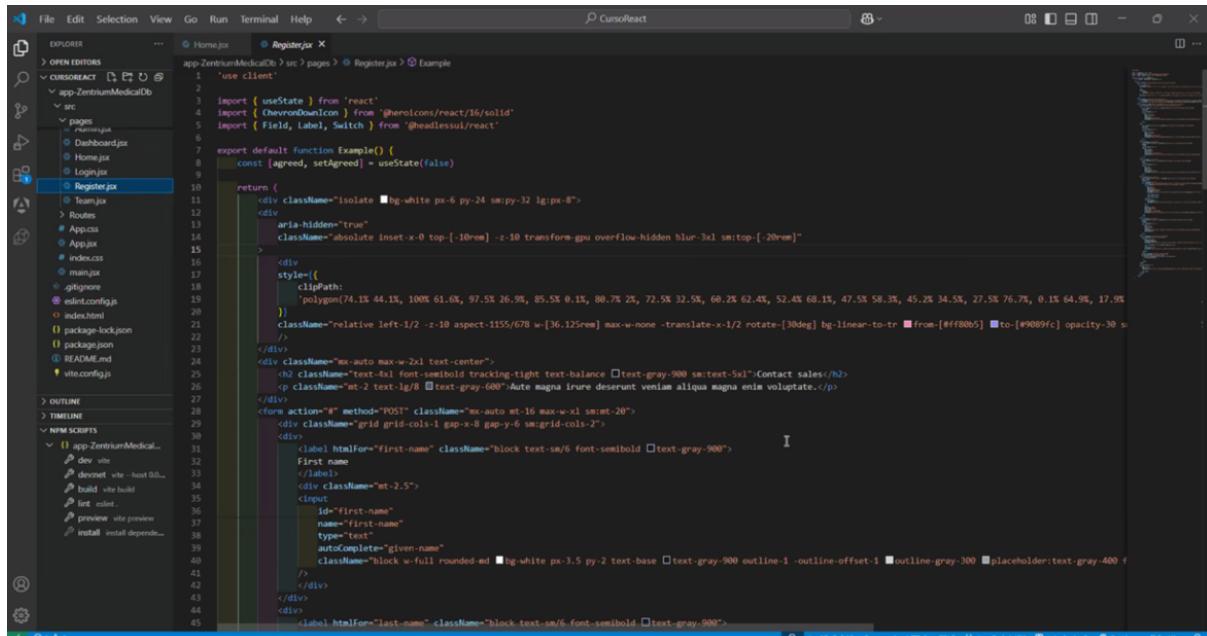
Nombre	Email	Role	Última actividad
Leslie Alexander	leslie.alexander@example.com	Gerente	Last seen 3h ago
Michael Foster	michael.foster@example.com	Doctor	Last seen 3h ago
Dries Vincent	dries.vincent@example.com	Pediatra	Online
Lindsay Walton	lindsay.walton@example.com	Enfermera	Last seen 3h ago
Courtney Henry	courtney.henry@example.com	Paramédico	Last seen 3h ago
Tom Cook	tom.cook@example.com	Director General	Online



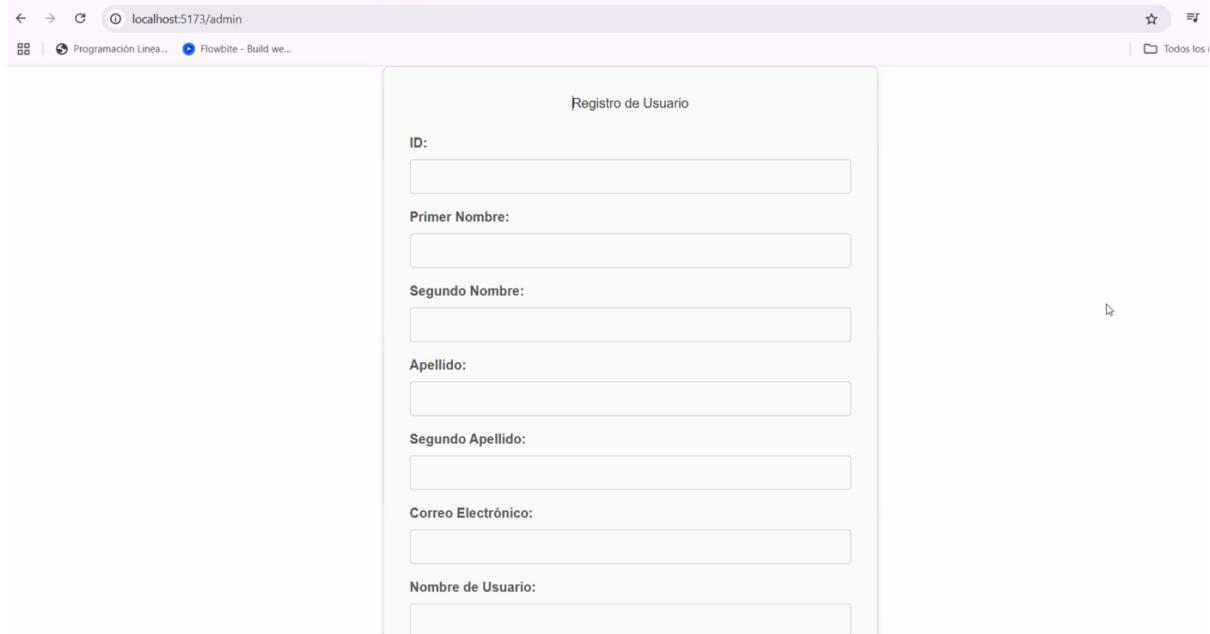
**UNIVERSIDAD TÉCNICA DE AMBATO**  
**FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL**  
**CARRERA DE Elige un elemento.**  
**CICLO ACADÉMICO: MARZO – JULIO 2025**



- Clase Register código y su interfaz:



```
File Edit Selection View Go Run Terminal Help ← → ⌘ Cursoreact 08 □ □ □ ...  
OPEN EDITORS  
CURSOREACT app-ZentrumMedicalDb  
app-ZentrumMedicalDb  
src  
pages  
Dashboard.jsx  
Home.jsx  
Login.jsx  
Register.jsx  
Tareas.jsx  
Routes  
App.jsx  
index.jsx  
main.jsx  
gitignore  
eslint.config.js  
package-lock.json  
package.json  
README.md  
vite.config.js  
OUTLINE  
TIMELINE  
NPM SCRIPTS  
app-ZentrumMedical...  
dev vite  
devnet vite --host 0.0.0.0  
build vite build  
first eslint...  
preview vite preview  
install install dependencies  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57  
58  
59  
60  
61  
62  
63  
64  
65  
66  
67  
68  
69  
70  
71  
72  
73  
74  
75  
76  
77  
78  
79  
80  
81  
82  
83  
84  
85  
86  
87  
88  
89  
90  
91  
92  
93  
94  
95  
96  
97  
98  
99  
100  
101  
102  
103  
104  
105  
106  
107  
108  
109  
110  
111  
112  
113  
114  
115  
116  
117  
118  
119  
120  
121  
122  
123  
124  
125  
126  
127  
128  
129  
130  
131  
132  
133  
134  
135  
136  
137  
138  
139  
140  
141  
142  
143  
144  
145  
use client';  
import { useState } from 'react';  
import { ChevronDownIcon } from '@heroicons/react/16/solid';  
import { Field, Label, Switch } from '@headlessui/react';  
  
export default function Example() {  
  const [agreed, setAgreed] = useState(false);  
  
  return (  
    <div className="isolate w-[36.125rem] max-w-none transform-gpu overflow-hidden blur-3xl sm:top-[20rem] sm:z-[10]">  
      <div style={{  
        clipPath:  
          "polygon(74.1% 44.1%, 100% 61.6%, 97.5% 26.9%, 85.5% 0.1%, 80.7% 2%, 72.5% 32.5%, 60.2% 62.4%, 52.4% 68.1%, 47.5% 58.3%, 45.2% 34.5%, 27.5% 76.7%, 0.1% 64.9%, 17.5%"  
      }}>  
        <div className="relative left-1/2 -z-10 aspect-1155/678 w-[36.125rem] max-w-none -translate-x-1/2 rotate-[30deg] bg-linear-to-tr from-[#FFB0b5] to-[#9089Fc] opacity-30">  
          <div className="mx-auto max-w-2xl text-center">  
            <h2>Contact sales</h2>  
            <p>Autem magna irure deserunt veniam aliqua magna enim voluptate.</p>  
            <form action="#" method="POST" className="w-full max-w-2xl sm:w-1/2 sm:mt-20">  
              <div className="grid grid-cols-1 gap-x-3 gap-y-6 sm:grid-cols-2">  
                <div>  
                  <label htmlFor="first-name" className="block text-sm font-semibold text-gray-900">  
                    First name  
                  </label>  
                  <div className="mt-2.5">  
                    <input id="first-name" name="first-name" type="text" autoComplete="given-name" className="block w-full rounded-md bg-white px-2 text-base text-gray-900 outline-1 outline-offset-1 outline-gray-300 placeholder:text-gray-400" />  
                  </div>  
                </div>  
                <div>  
                  <label htmlFor="last-name" className="block text-sm font-semibold text-gray-900">  
                    Last name  
                  </label>  
                  <input id="last-name" name="last-name" type="text" autoComplete="family-name" className="block w-full rounded-md bg-white px-2 text-base text-gray-900 outline-1 outline-offset-1 outline-gray-300 placeholder:text-gray-400" />  
                </div>  
              </div>  
            </form>  
          </div>  
        </div>  
      </div>  
    </div>
```



localhost:5173/admin

Programación Linea... Flowbite - Build we...

Todos los i

Registro de Usuario

ID:

Primer Nombre:

Segundo Nombre:

Apellido:

Segundo Apellido:

Correo Electrónico:

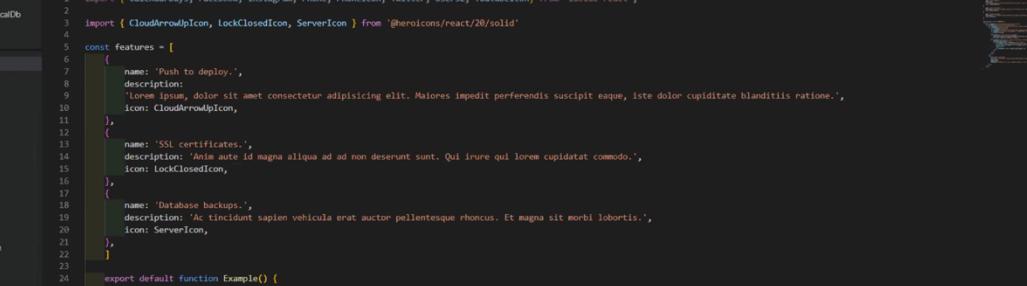
Nombre de Usuario:



**UNIVERSIDAD TÉCNICA DE AMBATO**  
**FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL**  
**CARRERA DE Elige un elemento.**  
**CICLO ACADÉMICO: MARZO – JULIO 2025**



- Clase dashboard su código y su interfaz:



```
File Edit Selection View Go Run Terminal Help <--> CursoReact

EXPLORER ... Dashboard.jsx x
OPEN EDITORS
CURSOREACT
app-ZentrumMedicalDb > src > pages > Dashboard.jsx > Example
1 import { CalendarDays, Facebook, Instagram, Phone, PhoneIcon, Twitter, Users2, YouthsBeacon } from 'lucide-react';
2 import { CloudArrowUpIcon, LockClosedIcon, ServerIcon } from '@heroicons/react/20/solid';
3
4 const features = [
5   {
6     name: 'Push to deploy',
7     description: `lorem ipsum, dolor sit amet consectetur adipisicing elit. Maiores impedit perferendis suscipit eaque, iste dolor cupiditate blanditiis ratione.`,
8     icon: CloudArrowUpIcon,
9   },
10   {
11     name: 'SSL certificates',
12     description: `Anax autem id magna aliqua ad non deserunt sunt. Qui irure qui lorem cupidatat commodo.`,
13     icon: LockClosedIcon,
14   },
15   {
16     name: 'Database backups',
17     description: `Ac tincidunt sapien vehicula erat auctor pellentesque rhoncus. Et magna sit morbi lobortis.`,
18     icon: ServerIcon,
19   },
20 }
21
22
23
24 export default function Example() {
25   return (
26     <div className="overflow-hidden w-[white-space: nowrap] py-2 sm:py-32">
27       <div className="mx-auto max-w-7xl px-8 lg:px-0">
28         <div className="ex-auto grid max-w-[grid-cols-1 gap-x-8 gap-y-16 sm:gap-x-20 lg:gap-y-20 lg:max-w-none lg:grid-cols-2">
29           <div className="lg:pt-4 lg:pr-0 lg:w-[lg]>
30             <div className="lg:max-w-[lg]>
31               <h2 className="text-base-7 font-base-sembold text-indigo-600">Deploy faster</h2>
32               <p className="mt-2 text-4xl font-sembold tracking-tight text-pretty text-gray-900 sm:text-5xl">
33                 A better workflow
34               </p>
35               <p className="mt-6 text-lg/8 text-gray-600">
36                 lorem ipsum, dolor sit amet consectetur adipisicing elit. Maiores impedit perferendis suscipit eaque,
37                 iste dolor cupiditate blanditiis ratione.
38               </p>
39             </div>
40           <div className="mt-10 max-w-[1 space-y-8 text-base-7] text-gray-600 lg:max-w-none">
41             <feature.name> <feature.name> <feature.name>
42             <div key={feature.name} className="relative pt-9">
43               <div className="absolute top-0 left-0 w-full h-full bg-white rounded-[10px] z-10">
44                 <feature.icon aria-hidden="true" className="absolute top-1 left-1 size-5 text-indigo-600" />
45                 <feature.name>
46               </div>
47             </div>
48           </div>
49         </div>
50       </div>
51     </div>
52   </div>
53 
```

The screenshot shows a dark-themed web application interface. At the top, there's a navigation bar with icons for back, forward, search, and refresh, followed by the URL "localhost:5173/team". Below the URL, there are tabs for "Programación Linea..." and "Flowbite - Build we...".

The main content area has a dark background with a central white box containing a contact form. The form includes a large input field for an email address and a blue "Enviar" button.

On the right side, there are two sections: "Personal Médico" and "Personal Administrativo", each with an icon and a brief description.

At the bottom, there are two green "Ingresar" buttons, one for each section.

**Dejanos tu correo para contactarte**

Ten en cuenta tener una hoja de vida que cumpla con estandares elevados para este hospital.

Ingresa tu email

Enviar

**Personal Médico**

Accede a tu espacio médico personalizado, donde la precisión y el cuidado se encuentran con la tecnología. Inicia sesión para gestionar consultas, analizar historiales y optimizar la atención, todo en un entorno intuitivo y seguro.

**Personal Administrativo**

Accede a tu portal administrativo y optimiza la gestión con facilidad. Inicia sesión para supervisar procesos, coordinar tareas y mantener el flujo de trabajo en un entorno seguro y eficiente.

Ingresar

Ingresar

### **6.55 Habilidades blandas empleadas en la práctica**

- Liderazgo
  - Trabajo en equipo
  - Comunicación asertiva
  - La empatía
  - Pensamiento crítico
  - Flexibilidad
  - La resolución de conflictos
  - Adaptabilidad
  - Responsabilidad



### **6.56 Conclusiones**

- Se logró desplegar satisfactoriamente un entorno distribuido compuesto por varias máquinas virtuales interconectadas y nodos Docker, esta implementación permitió simular centros médicos distribuidos geográficamente, asegurando una infraestructura escalable y flexible para el manejo de datos hospitalarios.
- Se implementó una replicación adecuada entre los nodos utilizando un modelo homogéneo de bases de datos, con esquemas de replicación sincrónica y fragmentación horizontal, esto aseguró la disponibilidad de los datos en todos los nodos y mejoró el rendimiento en la consulta y actualización de información médica.
- Se construyó una API RESTful siguiendo el patrón Modelo-Vista-Controlador (MVC) utilizando Node.js y Express, lo cual facilitó la separación de responsabilidades y el mantenimiento del código, además, se integró un frontend funcional que permitió a los usuarios interactuar de forma intuitiva con el sistema, gestionar entidades clave y consumir los servicios expuestos por la API.

### **6.57 Recomendaciones**

- Optimizar la configuración de los contenedores para reducir el uso de recursos.
- Documentar los endpoints RESTful en Swagger desde el inicio del desarrollo.
- Realizar pruebas de carga para validar la escalabilidad del sistema.

### **6.58 Referencias bibliográficas**

- [1] R. Elmasri and S. B. Navathe, *Fundamentals of Database Systems*, 7th ed. Boston, MA, USA: Pearson, 2015.
- [2] M. Fowler, *Patterns of Enterprise Application Architecture*. Boston, MA, USA: Addison-Wesley, 2002.
- [3] Node.js, “Node.js Documentation,” [Online]. Available: <https://nodejs.org/>
- [4] Docker, “Docker Documentation,” [Online]. Available: <https://docs.docker.com/>
- [5] IEEE Standards Association, “IEEE Std 1063-2001 - IEEE Standard for Software and System Documentation,” 2001. [Online]. Available: <https://standards.ieee.org/standard/1063-2001.html>
- [6] Microsoft, “SQL Server Always On availability groups,” [Online]. Available: <https://learn.microsoft.com/en-us/sql/sql-server/always-on/>
- [7] Microsoft, “ASP.NET Core Web API Documentation,” [Online]. Available: <https://learn.microsoft.com/en-us/aspnet/core/web-api/>