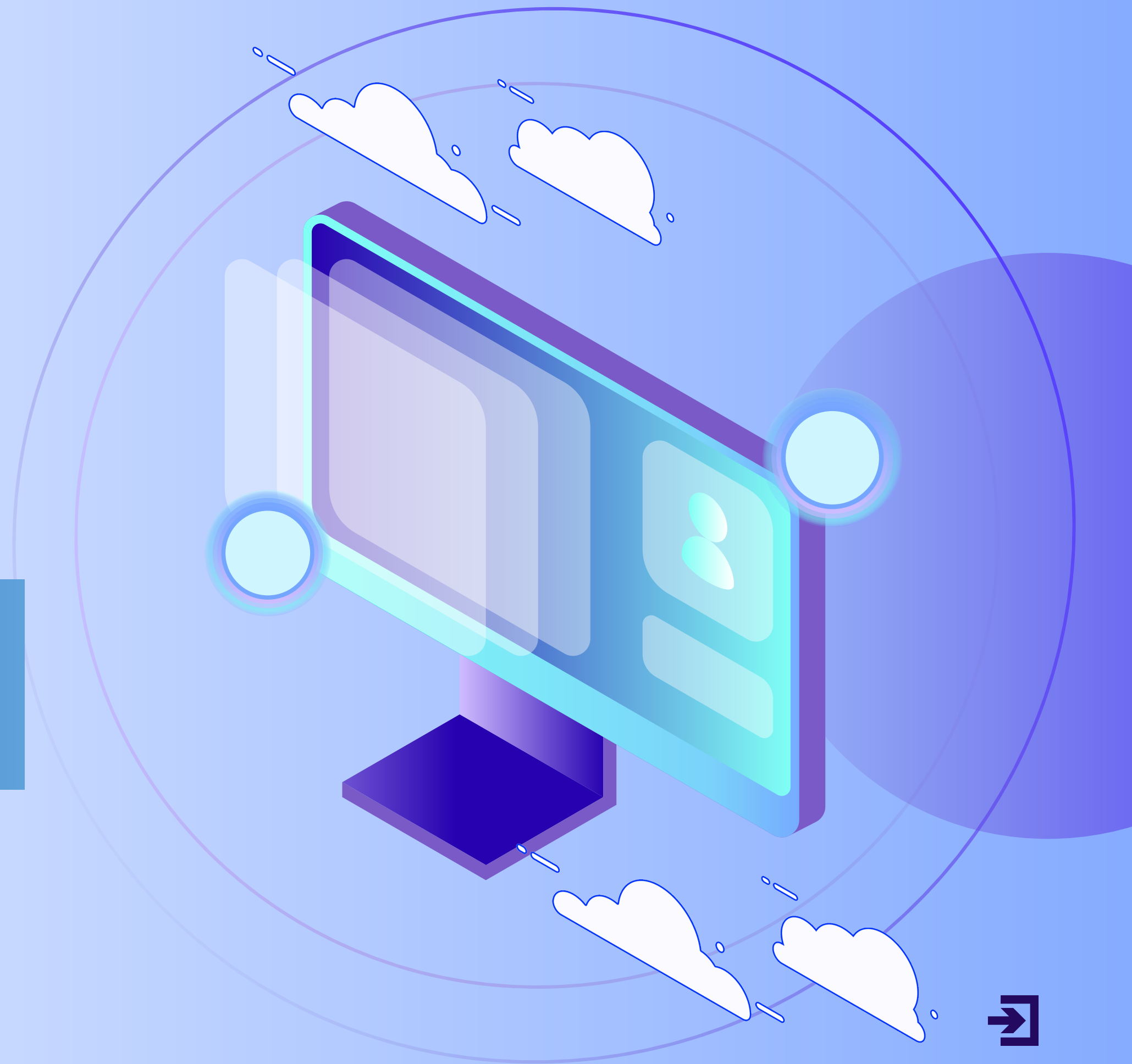




MEDICITAS

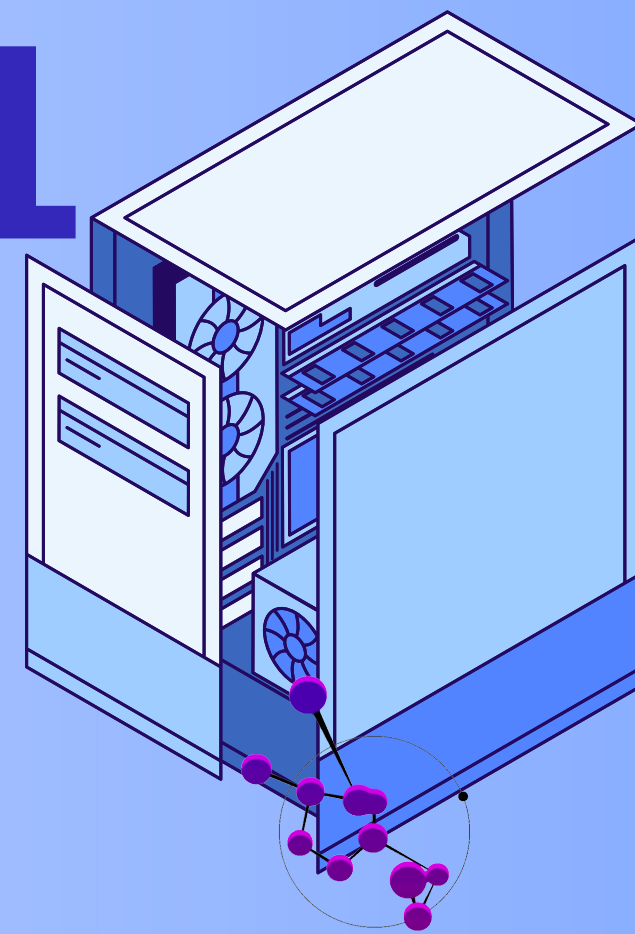
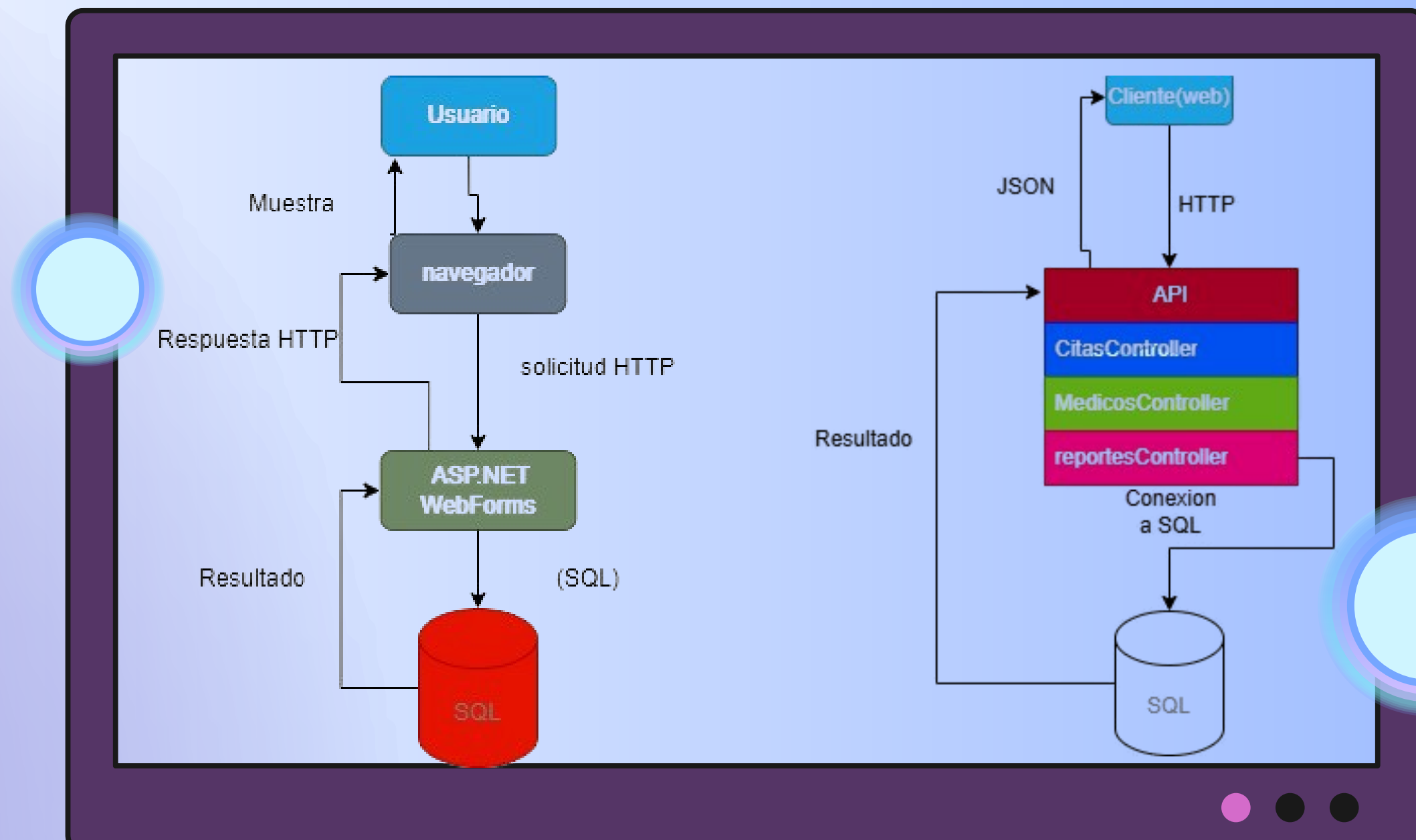
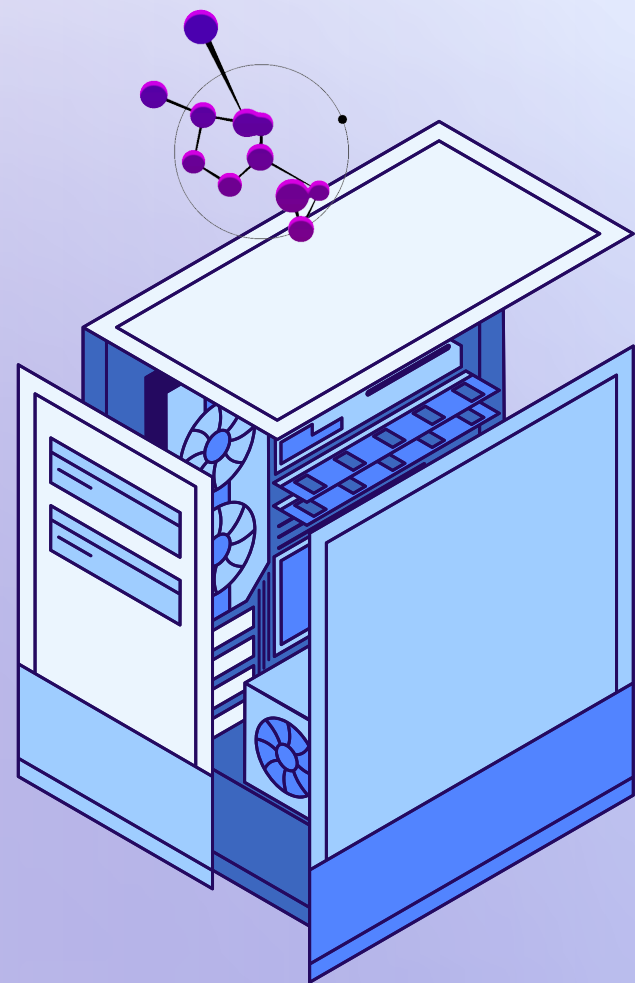
PROYECTO DS4

David Pérez



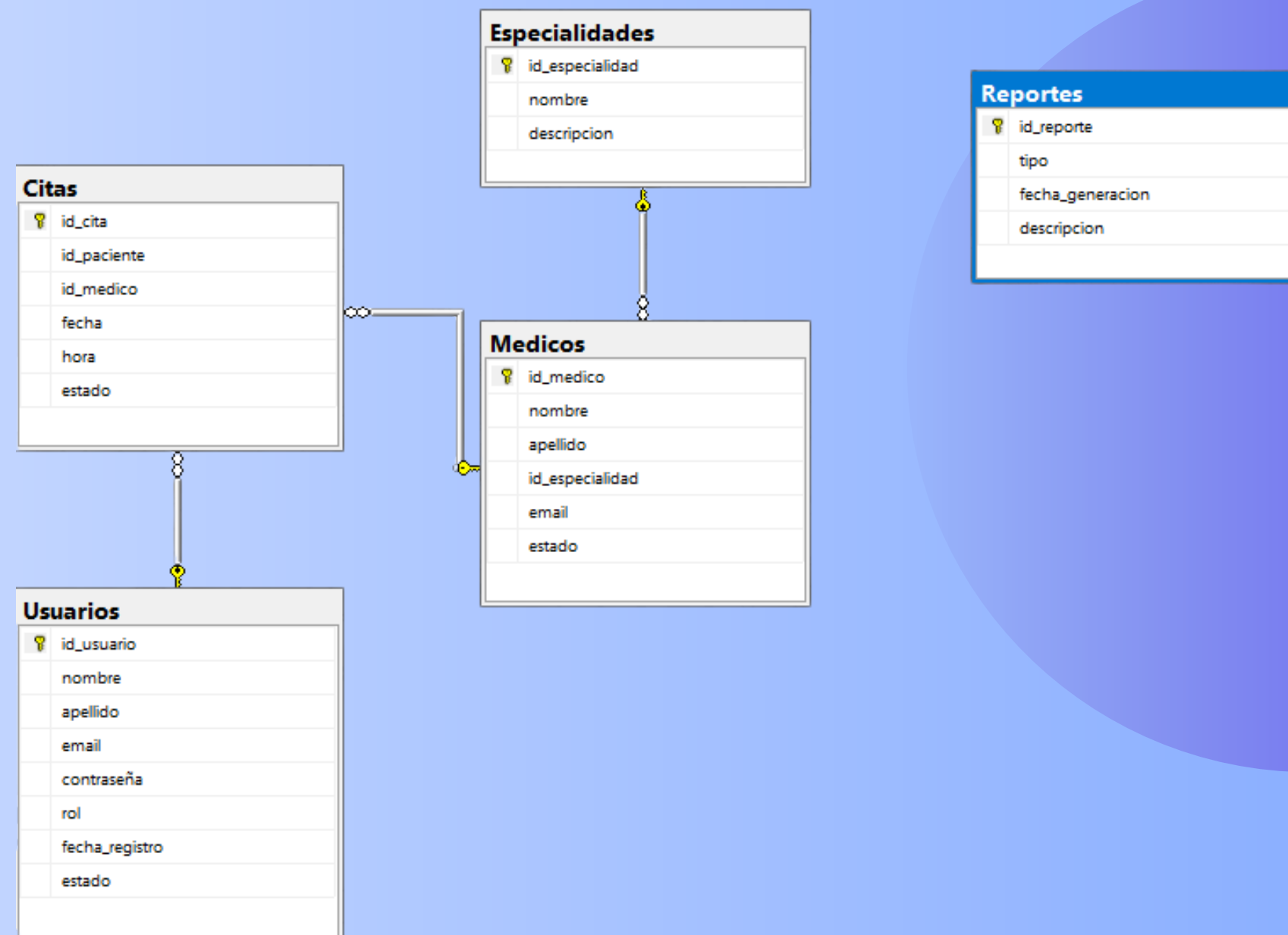


DIAGRAMAS DE ALTO NIVEL



```
SQLQuery1.sql...songo (52))* X DAVID\SQLEXP... - Diagram_0*
1 CREATE DATABASE MediCitasBD;
2 GO
3 USE MediCitas;
4 GO
5
6
7 CREATE TABLE Usuarios (
8     id_usuario INT IDENTITY(1,1) PRIMARY KEY,
9     nombre NVARCHAR(50) NOT NULL,
    apellido NVARCHAR(50) NOT NULL,
    email NVARCHAR(100) NOT NULL UNIQUE,
    contraseña NVARCHAR(100) NOT NULL,
    rol NVARCHAR(20) NOT NULL, -- 'PACIENTE' / 'ADMIN'
    fecha_registro DATETIME DEFAULT GETDATE(),
    estado BIT DEFAULT 1
);
10
11 CREATE TABLE Especialidades (
12     id_especialidad INT IDENTITY(1,1) PRIMARY KEY,
13     nombre NVARCHAR(50) NOT NULL,
14     descripcion NVARCHAR(200) NULL
);
```

BASE DE DATOS



```

[HttpGet]
[Route("")]
0 referencias
public IActionResult GetCitas()
{
    using (SqlConnection con = new SqlConnection(conexion))
    {
        string sql = @"
SELECT
    c.id_cita,
    u.nombre + ' ' + u.apellido AS paciente,
    m.nombre + ' ' + m.apellido AS medico,
    c.fecha,
    CONVERT(VARCHAR(5), c.hora, 108) AS hora,
    c.estado
FROM Citas c
INNER JOIN Usuarios u ON c.id_paciente = u.id_usuario
INNER JOIN Medicos m ON c.id_medico = m.id_medico";

        SqlCommand cmd = new SqlCommand(sql, con);

        con.Open();
        SqlDataReader dr = cmd.ExecuteReader();

        var lista = new List<object>();

        while (dr.Read())
        {
            lista.Add(new
            {
                id_cita = dr["id_cita"],
                paciente = dr["paciente"],
                medico = dr["medico"],
                fecha = dr["fecha"],
            });
        }
    }
}

```

// POST api/citas

```

[HttpPost]
[Route("")]
0 referencias
public IActionResult GuardarCita(Cita c)
{
    using (SqlConnection con = new SqlConnection(conexion))
    {
        string sql = @"INSERT INTO Citas
            (id_paciente, id_medico, fecha, hora)
            VALUES
            (@id_paciente, @id_medico, @fecha, @hora)";

        SqlCommand cmd = new SqlCommand(sql, con);
        cmd.Parameters.AddWithValue("@id_paciente", c.id_paciente);
        cmd.Parameters.AddWithValue("@id_medico", c.id_medico);
        cmd.Parameters.AddWithValue("@fecha", c.fecha.Date);
        cmd.Parameters.AddWithValue("@hora", TimeSpan.Parse(c.hora.ToString()));

        con.Open();
        cmd.ExecuteNonQuery();

        return Ok("Cita guardada correctamente");
    }
}

```

```

[HttpGet]
[Route("disponibles/{id}")]
0 referencias
public IActionResult ObtenerHorariosDisponibles(int id, [FromUri] string fecha)
{
    if (!DateTime.TryParseExact(fecha, "yyyy-MM-dd", null,
        System.Globalization.DateTimeStyles.None, out DateTime fechaCita))
        return BadRequest("Fecha inválida");

    using (SqlConnection con = new SqlConnection(conexion))
    {
        string sql = @"
DECLARE @Horarios TABLE (Hora TIME(0));
INSERT INTO @Horarios VALUES
    ('09:00'),('10:00'),('11:00'),('12:00'),
    ('13:00'),('14:00'),('15:00'),('16:00');

SELECT CONVERT(VARCHAR(5), h.Hora, 108) AS Hora
FROM @Horarios h
WHERE NOT EXISTS (
    SELECT 1 FROM Citas c
    WHERE c.id_medico = @id
    AND c.fecha = @fecha
    AND CAST(c.hora AS TIME(0)) = h.Hora
);";

        SqlCommand cmd = new SqlCommand(sql, con);
        cmd.Parameters.AddWithValue("@id", id);
        cmd.Parameters.AddWithValue("@fecha", fechaCita.Date);

        con.Open();
        SqlDataReader dr = cmd.ExecuteReader();
    }
}

```

