



**INSTITUTO SUPERIOR
TECNOLÓGICO QUITO**
Formamos tu PROPÓSITO DE VIDA

**YO
SOY
ITQ**



PROYECTO FINAL DE ASIGNATURA

Desarrollo de Software

WWW.ITQ.EDU.EC

2024

**CUADERNO DEL
ESTUDIANTE**



**INSTITUTO SUPERIOR
TECNOLÓGICO QUITO**
Formamos tu PROPÓSITO DE VIDA

PROYECTO FINAL DE ASIGNATURA

“Desarrollo de Software”

“Bryan Francisco Velasco Lloacana”

Marzo 2025 – Mayo 2025



CONTENIDO

OBJETIVO DE LA ASIGNATURA	4
INDICACIONES GENERALES	4
I INTRODUCCIÓN	5
1. Manual Técnico	5
1.1. NOMBRE DEL PROYECTO	5
1.2. PLANTEAMIENTO DEL PROBLEMA	5
1.3. OBJETIVOS GENERALES	5
1.4. OBJETIVOS ESPECÍFICOS	5
1.5. Tecnologías Utilizadas	6
1.6. Desarrollo – Módulos	6
1.7. Usuarios	8
1.8. Espacios	9
1.9. Reservas	9
1.10. Pagos	10
1.11. Membresías	12
1.12. Reportes	13
1.13. Estructura de Base de Datos	14
1.14. Vistas	15
1.15. Librerías	15
.....	16

OBJETIVO DE LA ASIGNATURA

Aprender a Crear sistemas a las necesidades de nuestros clientes aprende a utilizar las tecnologías de mejor manera eficiente para agilizar el trabajo,

INDICACIONES GENERALES

- Respetar el formato establecido por el docente.
- Las imágenes tienen que ir con descripciones y fuentes, con normas APA 7ma edición.
- El presente documento tiene por defecto configurado los diferentes recursos de normas APA, en la sección de estilos, usarlos para los diferentes, temas y subtemas desarrollados en clase, además de los márgenes y encabezados y pies de página.
- La fuente de letra es 12 y estilo Times News Román.

I INTRODUCCIÓN

El presente proyecto consiste en el desarrollo de un sistema de gestión para reservar espacios de coworking, que permita la administración eficiente de reservas, membresías, registros de pago y servicios adicionales. Este sistema ofrecerá una solución integral que incluye también la integración de calendarios, reportes y un sistema de control de acceso basado en códigos QR.

1. Manual Técnico

1.1. NOMBRE DEL PROYECTO

Desarrollar un Sistema de Gestión para Reservar Espacios de Coworking.

1.2. PLANTEAMIENTO DEL PROBLEMA

La administración manual y los sistemas dispersos en los espacios de coworking limitan la eficiencia y dificultan el seguimiento de reservas, pagos y membresías. Esto crea una experiencia de usuario fragmentada y aumenta los costos operativos.

1.3. OBJETIVOS GENERALES

- Desarrollar un sistema integral para la gestión de espacios de coworking que permita a los administradores y clientes realizar reservas, administrar membresías y generar reportes de uso.
- Llegar a tener un sistema total mente funcional que ayude a la empresa.

1.4. OBJETIVOS ESPECÍFICOS

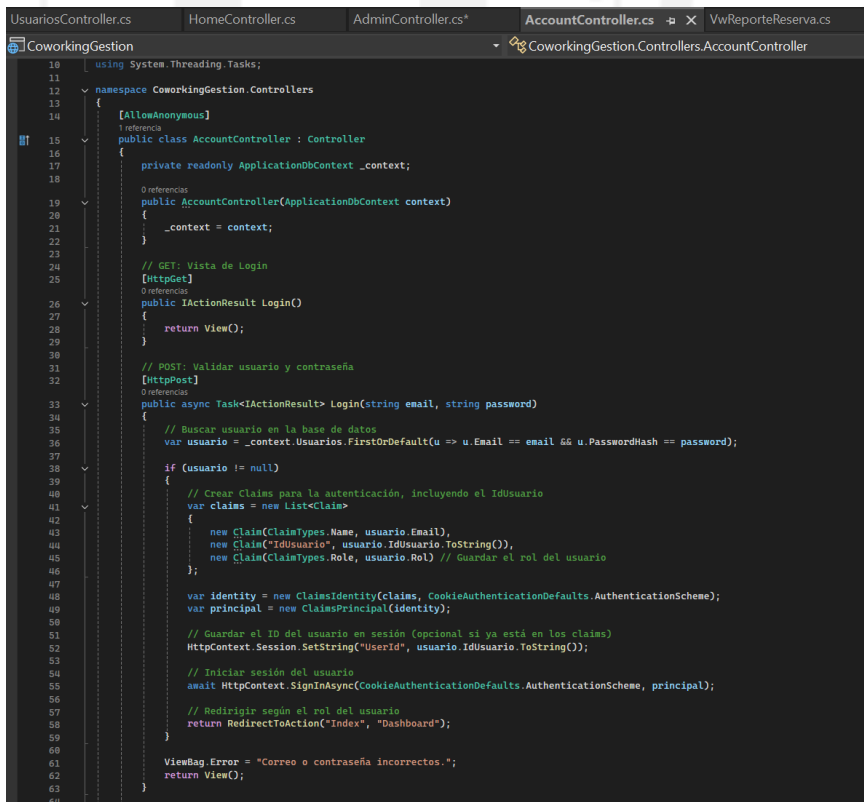
- Gestionar usuarios, espacios y reservas mediante formularios interactivos.
- Facilitar pagos para reservas y membresías.
- Generar códigos QR y archivos .ics para accesos y calendarios.
- Proveer reportes visuales basados en vistas SQL.

1.5. Tecnologías Utilizadas

- **Frontend:** Razor Pages, HTML5, CSS3, Bootstrap 4 (SB Admin 2)
- **Backend:** ASP.NET Core MVC
- **ORM:** Entity Framework Core
- **Base de Datos:** SQL Server
- **Lenguaje:** C#
- **Librerías externas:** SkiaSharp (para generar códigos QR), FullCalendar (calendario interactivo), DataTables (tablas avanzadas)

1.6. Desarrollo – Módulos

- **Autenticación y Autorización**
- Uso de cookies y claims para manejar sesiones.
- Login basado en roles (Administrador / Cliente).



```
10 using System.Threading.Tasks;
11
12 namespace CoworkingGestion.Controllers
13 {
14     [AllowAnonymous]
15     public class AccountController : Controller
16     {
17         private readonly ApplicationDbContext _context;
18
19         // GET: Vista de Login
20         [HttpGet]
21         public IActionResult Login()
22         {
23             return View();
24         }
25
26         // POST: Validar usuario y contraseña
27         [HttpPost]
28         public async Task<IActionResult> Login(string email, string password)
29         {
30             // Buscar usuario en la base de datos
31             var usuario = _context.Usuarios.FirstOrDefault(u => u.Email == email && u.PasswordHash == password);
32
33             if (usuario != null)
34             {
35                 // Crear Claims para la autenticación, incluyendo el IdUsuario
36                 var claims = new List<Claim>
37                 {
38                     new Claim(ClaimTypes.Name, usuario.Email),
39                     new Claim("IdUsuario", usuario.IdUsuario.ToString()),
40                     new Claim(ClaimTypes.Role, usuario.Rol) // Guardar el rol del usuario
41                 };
42
43                 var identity = new ClaimsIdentity(claims, CookieAuthenticationDefaults.AuthenticationScheme);
44                 var principal = new ClaimsPrincipal(identity);
45
46                 // Guardar el ID del usuario en sesión (opcional si ya está en los claims)
47                 HttpContext.Session.SetString("UserId", usuario.IdUsuario.ToString());
48
49                 // Iniciar sesión del usuario
50                 await HttpContext.SignInAsync(CookieAuthenticationDefaults.AuthenticationScheme, principal);
51
52                 // Redirigir según el rol del usuario
53                 return RedirectToAction("Index", "Dashboard");
54             }
55
56             ViewBag.Error = "Correo o contraseña incorrectos.";
57             return View();
58         }
59     }
60 }
```



```
namespace CoworkingGestion.Controllers
{
    [Authorize(Roles = "Administrador")] // Solo los administradores pueden acceder a este controlador
    <reference>
    public class AdminController : Controller
    {
        private readonly ApplicationDbContext _context;

        <reference>
        public AdminController(ApplicationDbContext context)
        {
            _context = context;
        }

        // Página principal del panel de administración
        <reference>
        public IActionResult Index()
        {
            return View();
        }

        // Listar todos los usuarios registrados
        <reference>
        public IActionResult Usuarios()
        {
            var usuarios = _context.Usuarios.ToList();
            return View(usuarios);
        }

        // GET: Vista para crear un nuevo usuario
        [HttpGet]
        <reference>
        public IActionResult CrearUsuario()
        {
            return View();
        }

        // POST: Registrar un nuevo usuario con rol
        [HttpPost]
        <reference>
        public IActionResult CrearUsuario(string ncedula, string nombre, string apellido, string telefono, string email, string password, string rol)
        {
            // Verificar si el usuario ya existe
            var usuarioExistente = _context.Usuarios.FirstOrDefault(u => u.Email == email);

            if (usuarioExistente != null)
            {
                ViewBag.Error = "El correo ya está registrado.";
                return View();
            }

            // Crear nuevo usuario
            var nuevoUsuario = new Usuario
            {
                Ncedula = ncedula,
                Nombre = nombre,
                Apellido = apellido,
            }
        }
    }
}
```

Iniciar Sesión

Correo Electrónico

Contraseña

Ingresar

[¿No tienes cuenta? Regístrate](#)

- Loworkingus
 - Database Diagrams
 - Tables
 - System Tables
 - FileTables
 - External Tables
 - Graph Tables
 - dbo.Espacios
 - dbo.Membresias
 - dbo.Pagos
 - dbo.PlanesMembresias
 - dbo.Reservas
 - dbo.Usuarios
 - Views
 - External Resources
 - Synonyms
 - Programmability
 - Query Store
 - Service Broker
 - Storage
 - Security
 - DBCARNITO
 - DBLoginT
 - DDCruoAngular
 - DistribuidoraRepuestosMolina
 - Documento

90 %

Results Messages

Id	Ncedula	Nombre	Apellido	Telefono	Email	PasswordIash	Rol
1	1723456789	Admin	Super	0991234567	admin@coworking.com	Admin123	Administrador
2	1729876543	Juan	Cliente	0997654321	juan@coworking.com	Cliente123	Cliente



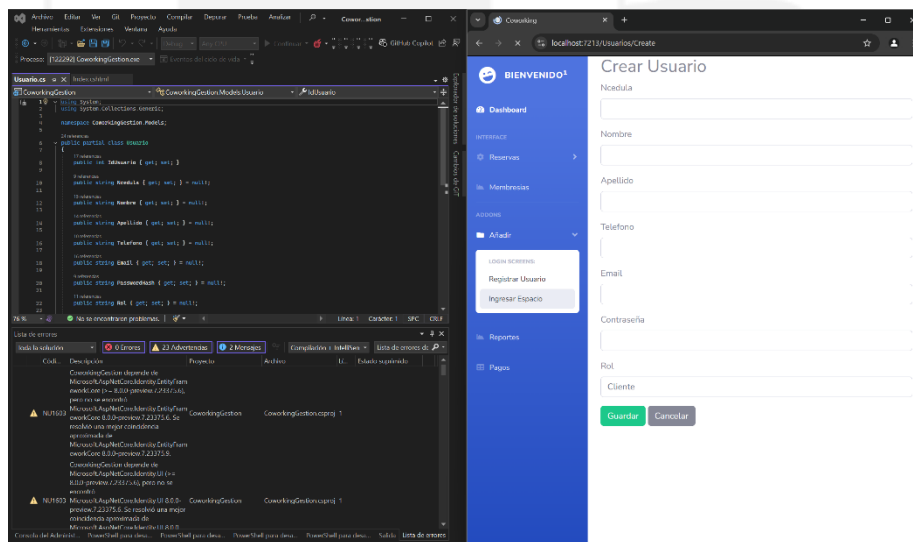
1.7. Usuarios

- CRUD completo (Crear, Listar, Editar, Eliminar).
- Validaciones por email y sesión.

```

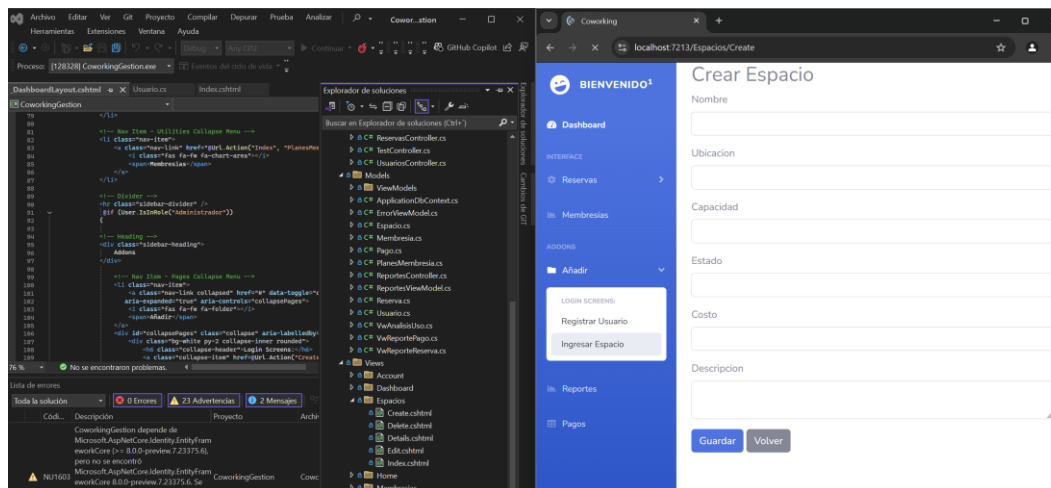
10 private readonly ApplicationDbContext _context;
11 public UsersController(ApplicationDbContext context)
12 {
13     _context = context;
14 }
15 // GET: Usuarios
16 public async Task

```



1.8. Espacios

- Registro de espacios disponibles con capacidad, ubicación, estado y costo.
- CRUD completo.



Espacios

Crear Nuevo Espacio

Nombre	Ubicación	Capacidad	Estado	Costo	Acciones
Sala B	Edificio Central	15	Disponible	\$75,00	Editar Detalles Eliminar

1.9. Reservas

- Creación de reservas vinculadas a espacios y usuarios.
- Estado inicial: "Pendiente de Pago".
- Visualización en calendario (FullCalendar).

Reservas

DataTables Example

Mostrar 10 registros Buscar:

ID	Usuario	Espacio	Fecha Reserva	Fecha Inicio	Fecha Fin	Estado	Costo	Acciones
1	Juan Cliente	Sala B	26/3/2025 20:09	26/3/2025 23:11	30/3/2025 23:11	Pagado	\$75.00	Eliminar Descargar iCS Generar QR
2	Juan Cliente	Sala B	26/3/2025 21:09	29/3/2025 0:12	30/3/2025 1:13	Pagado	\$75.00	Eliminar Descargar iCS Generar QR
3	Juan Cliente	Sala B	26/3/2025 21:15	29/3/2025 0:18	30/3/2025 1:19	Pagado	\$75.00	Eliminar Descargar iCS Generar QR

Mostrando 1 a 3 de 3 registros Anterior 1 Siguiente

Calendario de Reservas

marzo de 2025

today < >

dom	lun	mar	mié	jue	vie	sáb
23	24	25	26	27	28	1
2	3	4	5	6	7	8

Calendario de Reservas

marzo de 2025

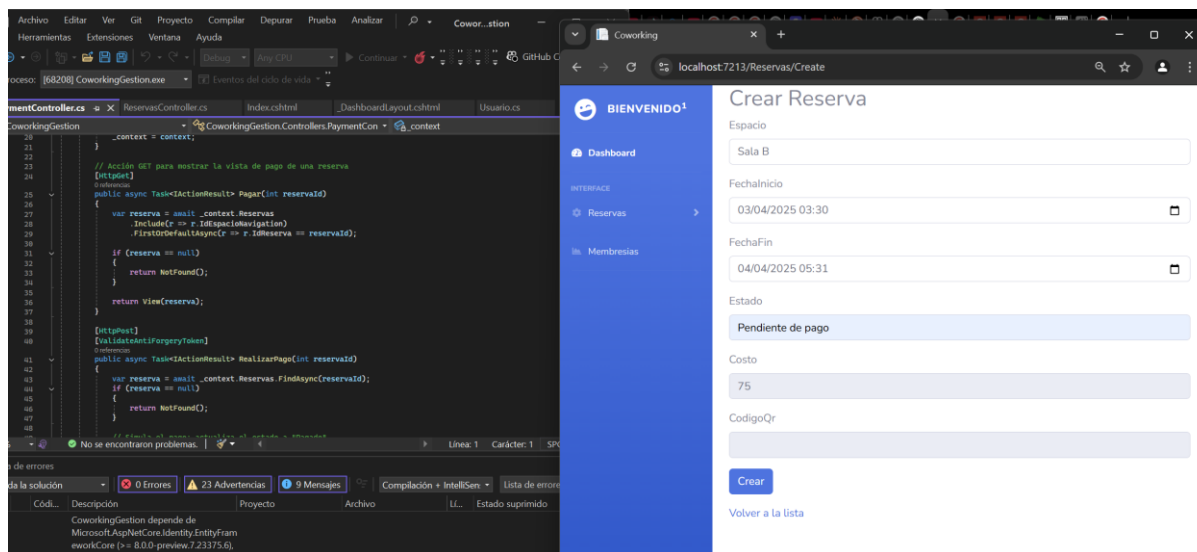
today < >

dom	lun	mar	mié	jue	vie	sáb
23	24	25	26	27	28	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

23:11 Sala B
0:12 Sala B
0:18 Sala B

1.10. Pagos

- Actualización de estado a "Pagado".
- Registro automático en tabla Pagos.
- Generación de código QR con SkiaSharp.



3	Juan Cliente	Sala B	26/3/2025 21:15	29/3/2025 0:18	30/3/2025 1:19	Pagado	\$75.00	Eliminar	Descargar ICS	Generar QR
4	Juan Cliente	Sala B	27/3/2025 1:29	3/4/2025 3:30	4/4/2025 5:31	Pendiente de Pago	\$75.00	Eliminar	Pagar	

Mostrando 1 a 4 de 4 registros

Anterior 1 Siguiente

Pago para Reserva #4

Detalles de la Reserva

Espacio: Sala B

27/3/2025 1:29

Fecha de Inicio: 3/4/2025 3:30

Fecha de Fin: 4/4/2025 5:31

Costo: \$75.00

Estado Actual: Pendiente de Pago

Pagar



Reservas

GRACIAS POR SU PAGO, su reservación se realizó con éxito. Aquí está su QR de acceso.

Código QR generado



DataTables Example

Mostrar 10 registros

Buscar:

ID	Usuario	Espacio	Fecha Reserva	Fecha Inicio	Fecha Fin	Estado	Costo	Acciones
1	Juan Cliente	Sala B	26/3/2025 20:09	26/3/2025 23:11	30/3/2025 23:11	Pagado	\$75.00	Eliminar Descargar ICS Generar QR
2	Juan Cliente	Sala B	26/3/2025 21:09	29/3/2025 0:12	30/3/2025 1:13	Pagado	\$75.00	Eliminar Descargar ICS Generar QR
3	Juan Cliente	Sala B	26/3/2025 21:15	29/3/2025 0:18	30/3/2025 1:19	Pagado	\$75.00	Eliminar Descargar ICS Generar QR
4	Juan Cliente	Sala B	27/3/2025 1:29	3/4/2025 3:30	4/4/2025 5:31	Pagado	\$75.00	Eliminar Descargar ICS Generar QR

Mostrando 1 a 4 de 4 registros

Anterior 1 Siguiente

1.11. Membresías

- Listado de planes disponibles.
- Pago artificial de membresías.
- Registro del plan en tabla Membresías y el pago en Pagos.

Planes de Membresía

Tu Membresía Activa

Plan: Plan Básico

Duración: 1 Mes

Precio: \$9,99

Descripción: Acceso limitado a salas compartidas.

Plan Básico

Acceso limitado a salas compartidas.

Duración: 1 Mes

Precio: \$9,99

Adquirir

Plan Pro

Acceso completo a salas compartidas y preferencia en reservas.

Duración: 3 Meses

Precio: \$24,99

Adquirir

Plan Empresarial

Acceso VIP a salas privadas, prioridad total y beneficios empresariales.

Duración: 6 Meses

Precio: \$49,99

Adquirir



BIENVENIDO¹

Dashboard

INTERFACE

Reservas

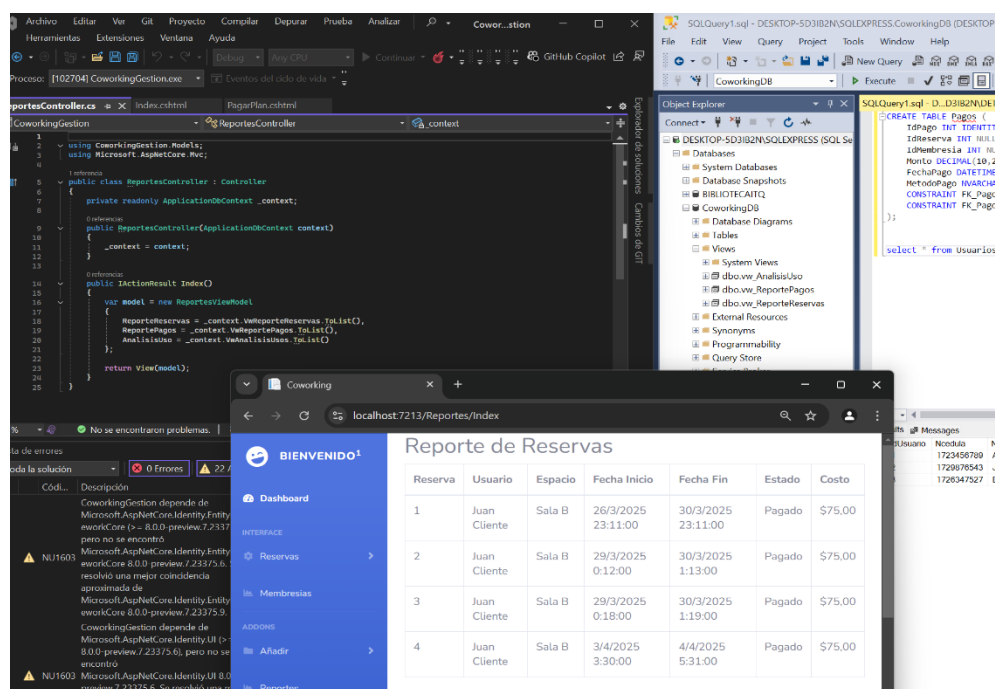
Membresías

Mis Membresías

Nombre del Plan	Duración	Precio	Descripción
Plan Básico	1 Mes	\$9,99	Acceso limitado a salas compartidas.
Plan Básico	1 Mes	\$9,99	Acceso limitado a salas compartidas.
Plan Pro	3 Meses	\$24,99	Acceso completo a salas compartidas y preferencia en reservas.
Plan Empresarial	6 Meses	\$49,99	Acceso VIP a salas privadas, prioridad total y beneficios empresariales.

1.12. Reportes

- Uso de vistas SQL para generar reportes:
- vw_ReporteReservas
- vw_ReportePagos
- vw_AnalisisUso
- Visualización combinada en una única vista.



Reserva	Usuario	Espacio	Fecha Inicio	Fecha Fin	Estado	Costo
1	Juan Cliente	Sala B	26/3/2025 23:11:00	30/3/2025 23:11:00	Pagado	\$75.00
2	Juan Cliente	Sala B	29/3/2025 0:12:00	30/3/2025 1:13:00	Pagado	\$75.00
3	Juan Cliente	Sala B	29/3/2025 0:18:00	30/3/2025 1:19:00	Pagado	\$75.00
4	Juan Cliente	Sala B	3/4/2025 3:30:00	4/4/2025 5:31:00	Pagado	\$75.00

Pago	Usuario	Monto	Fecha	Método	Estado Reserva
1	Juan Cliente	\$75.00	26/3/2025 21:01:59	Pago Artificial	Pagado
2	Juan Cliente	\$75.00	26/3/2025 21:10:08	Pago Artificial	Pagado
3	Juan Cliente	\$75.00	26/3/2025 21:17:51	Pago Artificial	Pagado
5	Juan Cliente	\$75.00	27/3/2025 1:30:10	Pago Artificial	Pagado

Espacio	Total Reservas	Ingresos	Primer Uso	Último Uso
Sala B	4	\$300.00	26/3/2025 23:11:00	4/4/2025 5:31:00

1.13. Estructura de Base de Datos

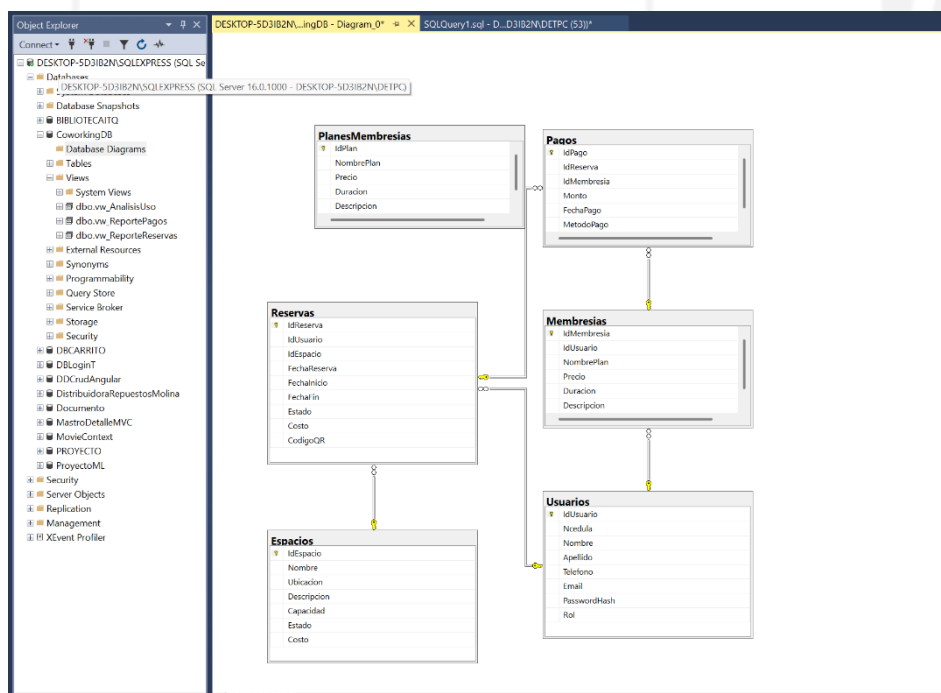
- Usuarios
- Espacios
- Reservas
- Pagos
- Membresías
- PlanesMembresia

```

-- Tabla de Membresías
CREATE TABLE Membresías (
    IdMembresia INT IDENTITY(1,1) PRIMARY KEY,
    IdUsuario INT NOT NULL,
    NombrePlan NVARCHAR(100) NOT NULL,
    Precio DECIMAL(10,2) NOT NULL,
    Duracion NVARCHAR(50) NOT NULL,
    Descripcion NVARCHAR(500),
    CONSTRAINT FK_Membresias_Usuarios FOREIGN KEY (IdUsuario) REFERENCES Usuarios(IdUsuario)
);

-- Tabla de Planes de Membresías
CREATE TABLE PlanesMembresias (
    IdPlan INT IDENTITY(1,1) PRIMARY KEY,
    NombrePlan NVARCHAR(100) NOT NULL,
    Precio DECIMAL(10,2) NOT NULL,
    Duracion NVARCHAR(50) NOT NULL,
    Descripcion NVARCHAR(500)
);

-- Tabla de Pagos
CREATE TABLE Pagos (
    IdPago INT IDENTITY(1,1) PRIMARY KEY,
    IdReserva INT NOT NULL,
    IdMembresia INT NOT NULL,
    Monto DECIMAL(10,2) NOT NULL,
    FechaPago DATETIME DEFAULT GETDATE(),
    MetodoPago NVARCHAR(100) NOT NULL,
    CONSTRAINT FK_Pagos_Reservas FOREIGN KEY (IdReserva) REFERENCES Reservas(IdReserva),
    CONSTRAINT FK_Pagos_Membresias FOREIGN KEY (IdMembresia) REFERENCES Membresias(IdMembresia)
);
    
```



1.14. Vistas

- vw_ReporteReservas
- vw_ReportePagos
- vw_AnalisisUso

Reporte de Reservas

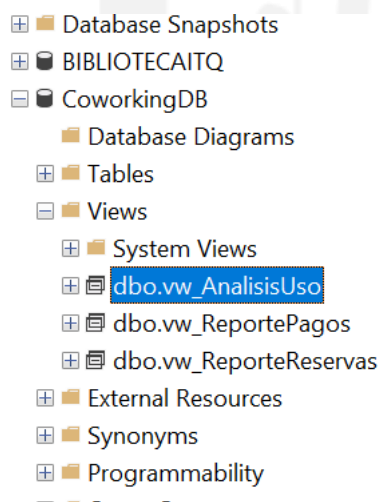
Reserva	Usuario	Espacio	Fecha Inicio	Fecha Fin	Estado	Costo
1	Juan Cliente	Sala B	26/3/2025 23:11:00	30/3/2025 23:11:00	Pagado	\$75.00
2	Juan Cliente	Sala B	29/3/2025 0:12:00	30/3/2025 1:13:00	Pagado	\$75.00
3	Juan Cliente	Sala B	29/3/2025 0:18:00	30/3/2025 1:19:00	Pagado	\$75.00
4	Juan Cliente	Sala B	3/4/2025 3:30:00	4/4/2025 5:31:00	Pagado	\$75.00

Reporte de Pagos

Pago	Usuario	Monto	Fecha	Método	Estado Reserva
1	Juan Cliente	\$75.00	26/3/2025 21:01:59	Pago Artificial	Pagado
2	Juan Cliente	\$75.00	26/3/2025 21:10:08	Pago Artificial	Pagado
3	Juan Cliente	\$75.00	26/3/2025 21:17:51	Pago Artificial	Pagado
5	Juan Cliente	\$75.00	27/3/2025 1:30:10	Pago Artificial	Pagado

Análisis de Uso

Espacio	Total Reservas	Ingresos	Primer Uso	Último Uso
Sala B	4	\$300.00	26/3/2025 23:11:00	4/4/2025 5:31:00



1.15. Librerías

- Generación de archivos .ics para calendario externo.
- Validación de acceso vía QR.
- Notificaciones via TempData para éxito/error.

Reservas

QR generado correctamente.

Código QR generado



Crear Reserva

DataTables Example

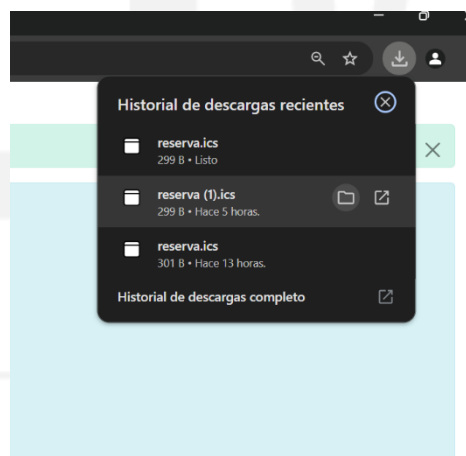
Mostrar 10 registros

Buscar:

ID	Usuario	Espacio	Fecha Reserva	Fecha Inicio	Fecha Fin	Estado	Costo	Acciones
1	Juan Cliente	Sala B	26/3/2025 20:09	26/3/2025 23:11	30/3/2025 23:11	Pagado	\$75,00	Editar Eliminar Descargar ICS Generar QR
2	Juan Cliente	Sala B	26/3/2025 21:09	29/3/2025 0:12	30/3/2025 1:13	Pagado	\$75,00	Editar Eliminar Descargar ICS Generar QR
3	Juan Cliente	Sala B	26/3/2025 21:15	29/3/2025 0:18	30/3/2025 1:19	Pagado	\$75,00	Editar Eliminar Descargar ICS Generar QR
4	Juan Cliente	Sala B	27/3/2025 1:29	3/4/2025 3:30	4/4/2025 5:31	Pagado	\$75,00	Editar Eliminar Descargar ICS Generar QR

Mostrando 1 a 4 de 4 registros

Anterior 1 Siguiente



CONCLUSIONES

El desarrollo del sistema de gestión para la empresa de coworking ha permitido implementar un entorno digital completo y funcional, que facilita la administración de reservas, usuarios, espacios, membresías y pagos.

Durante el proceso, se aplicaron buenas prácticas de desarrollo con ASP.NET Core MVC y herramientas modernas como Entity Framework Core y librerías externas especializadas. La integración de autenticación, protección de rutas, reportes SQL y generación de códigos QR enriquecen la funcionalidad del sistema.

RECOMENDACIONES FINALES

- El sistema debe ser alojado en un servidor web o nube para que esté disponible las 24h
- Dar soporte al sistema para que no tengo ningún fallo.
- Actualizar el modo de autenticación de los roles.
- Encriptar Contraseñas para no mostrarlos en textos planos.

Este sistema está listo para ser utilizado, escalado y adaptado a nuevos requerimientos de la empresa.