



**Instituto Tecnológico Colonia CTC**

**Analista programador  
Programación 1**

# **Obligatorio solitario, Programación 1**

Profesor: Carlos Rodriguez

**Julián Silvera**

**2024**

# Índice

<b>Abstract.....</b>	<b>2</b>
<b>Cronograma.....</b>	<b>3</b>
<b>Diccionario de clases.....</b>	<b>4</b>
Nombre de la Clase: Hotel.....	4
Nombre de la Clase: Habitacion.....	6
Nombre de la Clase: Huesped.....	7
Nombre de la Clase: Reserva.....	8
Nombre de la Clase: Pago.....	10
<b>Diagrama de la capa de dominio.....</b>	<b>11</b>
<b>Diagrama de las dos capas.....</b>	<b>12</b>
<b>Explicación de funciones extras:.....</b>	<b>15</b>

# Abstract

El proyecto "Hotel Estrella del Mar" representa un viaje personal de aprendizaje y desarrollo en la creación de un sistema de gestión hotelera que, aunque simple en su interfaz, resulta complejo en su funcionamiento. Esta travesía se ha llevado a cabo con un lenguaje de programación nuevo, con el que no estaba completamente familiarizado.

El sistema tiene como objetivo optimizar la experiencia de reserva para los clientes, permitiéndoles reservar habitaciones, editar sus reservas, administrar pagos y generar comprobantes, todo a través de una interfaz intuitiva y amigable.

Desde el principio, este proyecto ha sido un desafío constante. Cada día presentaba nuevos obstáculos, y aprendí a resolver problemas que surgieron, desde la implementación de la lógica de reservas hasta la gestión de la información del usuario. La perseverancia y la dedicación han sido esenciales para superar estos retos. Con cada iteración, la funcionalidad y usabilidad del sistema mejoraron, logrando un equilibrio entre eficiencia y facilidad de uso.

En conclusión, "Hotel Estrella del Mar" es una herramienta valiosa para la gestión hotelera, resultado del esfuerzo personal y la constancia en la búsqueda de una mejor experiencia para todos los usuarios.

# Cronograma

<b>Tarea</b>	<b>Tiempo estimado</b>
Análisis	5 hs
C#	50 hs
Documentación	6 hs
Total	61 horas

# Diccionario de clases

**Nombre de la Clase:** Hotel

**Descripción de la Clase:**

La clase Hotel representa un sistema de gestión hotelera que permite administrar las habitaciones, reservas, huéspedes y pagos. Incluye métodos para realizar reservas, modificar y cancelar reservas, generar comprobantes, y gestionar la información de los huéspedes.

Atributo	Tipo de Dato	Descripción
Habitaciones	List<Habitacion>	Lista de todas las habitaciones del hotel.
Huesped	List<Huesped>	Lista de todos los huéspedes registrados.
Reservas	List<Reserva>	Lista de reservas realizadas en el hotel.
Pagos	List<Pago>	Lista de todos los pagos registrados.
UsuarioActual	Huesped?	Huésped actualmente logueado.

**Tabla de Métodos de la Clase Hotel:**

Nombre	Devuelve	Recibe (Parámetros)	Descripción
CargarDatos	Nada	Ninguno	Carga habitaciones y huéspedes predeterminados.
IniciarSesion	bool	string email, string contraseña	Verifica credenciales y establece la sesión.
CerrarSesion	Nada	Ninguno	Cierra la sesión del usuario actual.
ListarHabitacionesDisponibles	Nada	Ninguno	Muestra todas las habitaciones disponibles.
RealizarReserva	Nada	int numeroHabitacion, DateTime fechaInicio, DateTime fechaFin	Realiza una nueva reserva si la habitación está disponible.
ModificarReserva	Nada	int numeroHabitacionReservada, int? nuevoNumeroHabitacion, DateTime? nuevaFechaInicio, DateTime? nuevaFechaFin	Modifica la reserva existente.
CancelarReserva	Nada	int numeroReserva	Cancela una reserva existente.
CheckIn	Nada	int numeroReserva	Realiza el check-in de una reserva.
RealizarPagoPorHabitacion	Nada	int numeroHabitacion	Procesa el pago de una reserva asociada a una habitación.
GenerarComprobante	Nada	int numeroReserva	Genera un comprobante de pago para una reserva.
RecuperarContraseña	Nada	string email	Permite recuperar la contraseña mediante una pregunta de seguridad.

**Nombre de la Clase:** Habitacion**Descripción de la Clase:**

La clase Habitacion representa una habitación del hotel, con atributos que incluyen el número, tipo, capacidad, precio diario y disponibilidad. Incluye métodos para cambiar la disponibilidad.

**Tabla de Atributos de la Clase Habitacion:**

Atributo	Tipo de Dato	Descripción
Numero	int	Número único que identifica la habitación.
Tipo	string	Tipo de habitación (Simple, Doble, Suite).
Capacidad	int	Número máximo de huéspedes que puede alojar.
PrecioDiario	decimal	Costo de la habitación por noche.
EstaDisponible	bool	Indica si la habitación está disponible.

**Tabla de Métodos de la Clase Habitacion:**

Nombre	Devuelve	Recibe (Parámetros)	Descripción
CambiarDisponibilidad	Nada	Número único que identifica la habitación.	Cambia el estado de disponibilidad de la habitación.

**Nombre de la Clase:** Huesped

**Descripción de la Clase:**

La clase Huesped representa a un huésped del hotel, con atributos como nombre, apellido, email y contraseña. Incluye métodos para verificar la respuesta de seguridad.

**Tabla de Atributos de la Clase Huesped:**

Atributo	Devuelve	Descripción
Id	int	Identificador único del huésped.
Nombre	string	Nombre del huésped.
Apellido	string	Apellido del huésped.
Email	string	Correo electrónico del huésped.
Contraseña	string	Contraseña del huésped.
PreguntaSeguridad	string	Pregunta de seguridad para recuperación de contraseña.
RespuestaSeguridad	string	Respuesta a la pregunta de seguridad.

**Tabla de Métodos de la Clase Huesped:**

Nombre	Devuelve	Recibe (Parámetros)	Descripción
VerificarRespuestaSeguridad	bool	string respuesta	Verifica si la respuesta dada coincide con la respuesta de seguridad.

**Nombre de la Clase:** Reserva



**Descripción de la Clase:**

La clase Reserva representa una reserva realizada por un huésped, incluyendo información sobre la habitación, fechas de estadía y estado de pago.

**Tabla de Atributos de la Clase Reserva:**

Atributo	Tipo de Dato	Descripción
NumeroReserva	int	Identificador único de la reserva.
NumeroHabitacion	int	Número de la habitación reservada.
Huesped	int	Huésped asociado a la reserva.
FechaInicio	DateTime	Fecha de inicio de la reserva.
FechaFin	DateTime	Fecha de fin de la reserva.
EstaPagada	bool	Indica si la reserva ha sido pagada.
MontoTotal	decimal	Monto total a pagar por la reserva.
CheckInRealizado	bool	Indica si se ha realizado el check-in.
TipoPago	string	Tipo de pago utilizado para la reserva.

**Tabla de Métodos de la Clase Reserva:**

Nombre	Devuelve	Recibe (Parámetros)	Descripción
CambiarHabitacion	Nada	Habitacion nuevaHabitacion	Cambia la habitación reservada.
CambiarFechas	Nada	DateTime nuevaFechaInicio, DateTime nuevaFechaFin	Cambia las fechas de la reserva.
GenerarComprobante	Nada	Ninguno	Genera un comprobante de la reserva.

**Nombre de la Clase:** Pago

**Descripción de la Clase:**

La clase Pago representa un pago realizado por un huésped por una reserva, incluyendo el monto, fecha y estado del pago.

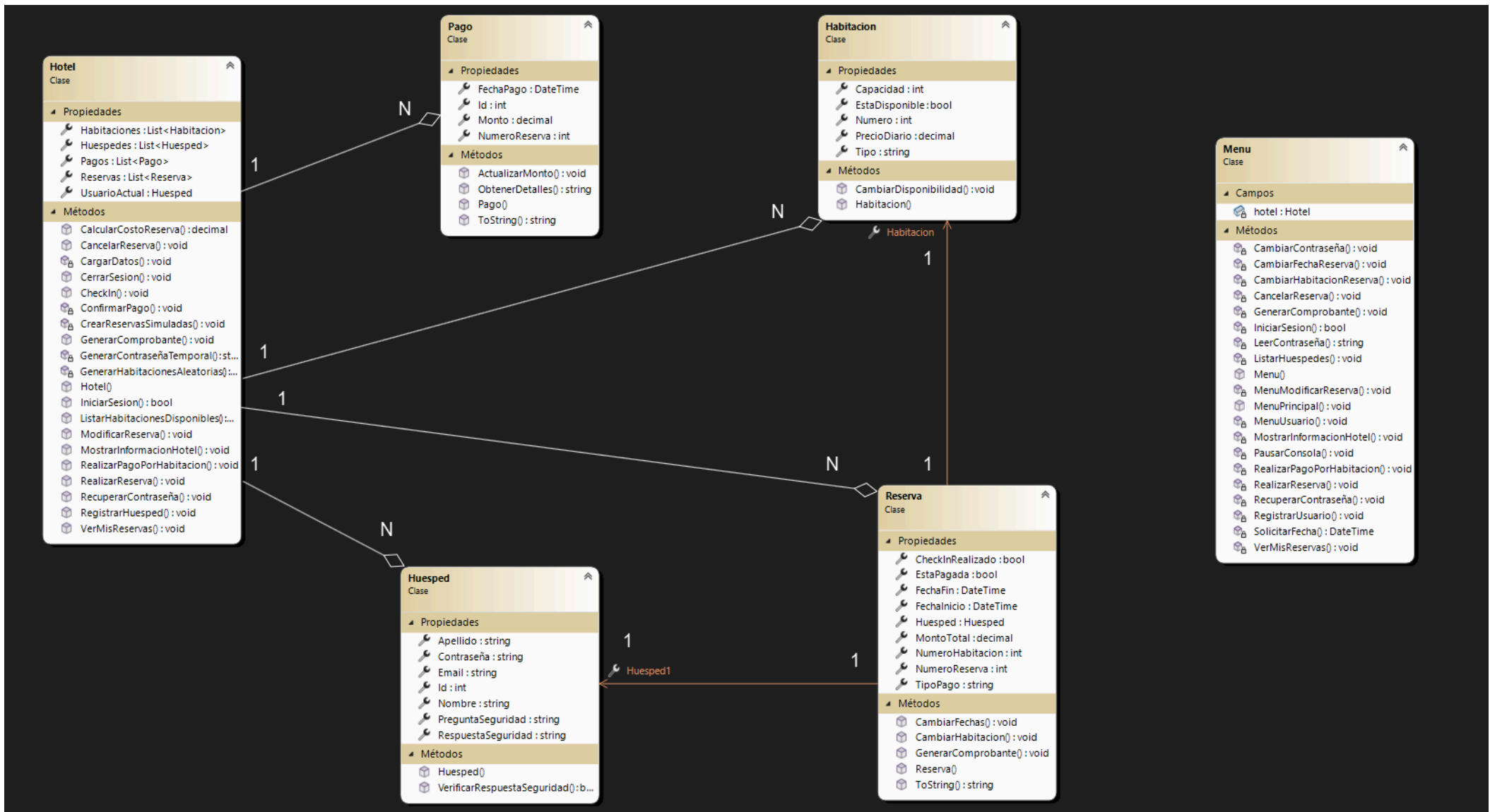
**Tabla de Atributos de la Clase Pago:**

Atributo	Tipo de Dato	Descripción
Id	int	Identificador único del pago.
NumeroReserva	int	Número de la reserva asociada al pago.
Monto	decimal	Monto total del pago realizado.
FechaPago	DateTime	Fecha en que se realizó el pago.

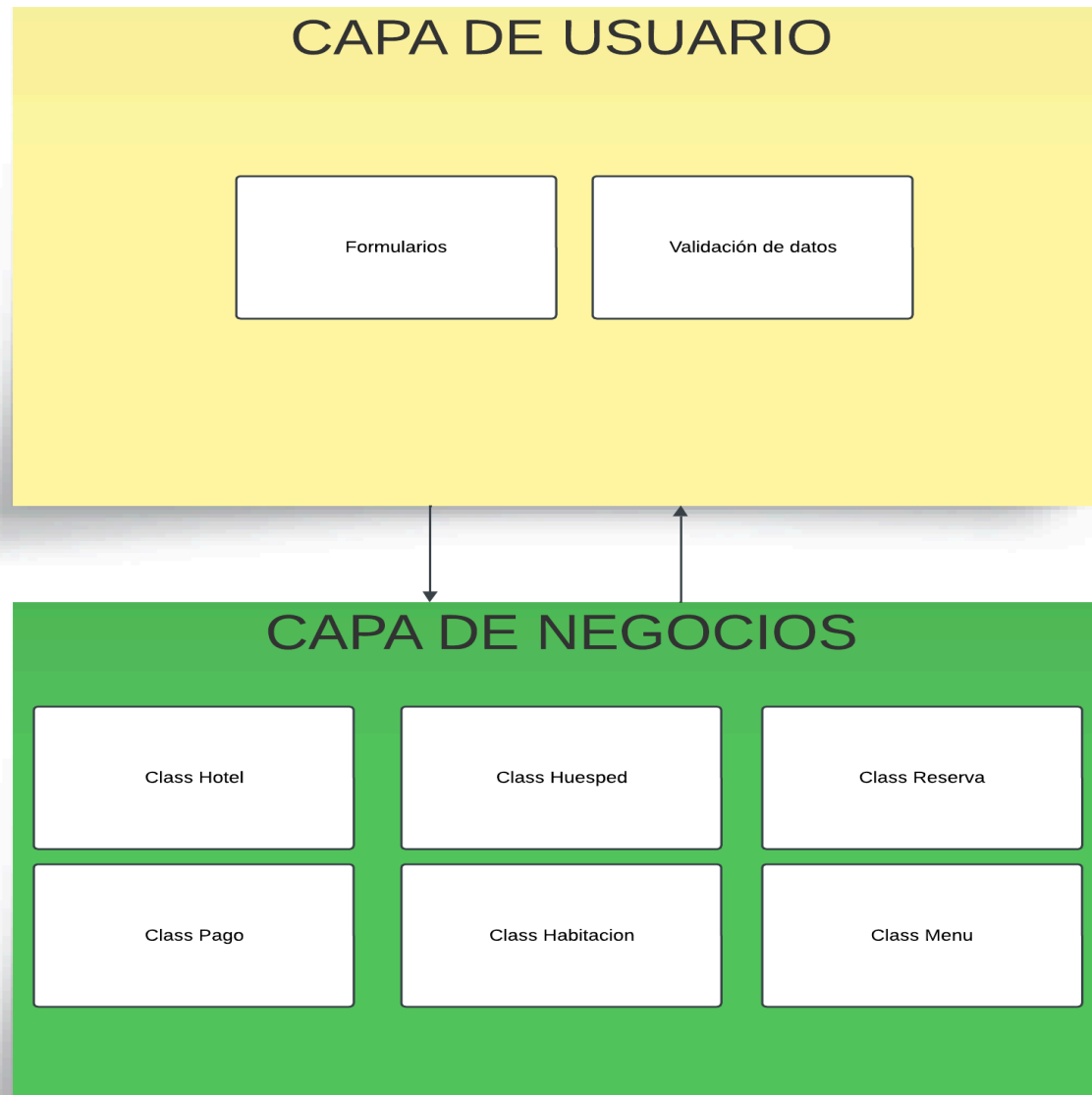
**Tabla de Métodos de la Clase Pago:**

Nombre	Devuelve	Recibe (Parámetros)	Descripción
ActualizarMonto	Nada	decimal nuevoMonto	Actualiza el monto del pago.
ObtenerDetalles	Nada	Ninguno	Devuelve una cadena con los detalles del pago.

# Diagrama de la capa de dominio



# Diagrama de las dos capas



# Resultados de las pruebas realizadas:

A lo largo del desarrollo del sistema, se llevaron a cabo diversas pruebas para garantizar el correcto funcionamiento de todas las funcionalidades implementadas.

## Pruebas de Registro de Usuario:

Se verificó que los usuarios pudieran registrarse correctamente proporcionando un nombre, apellido, email y contraseña, junto con una pregunta y respuesta de seguridad para poder modificar la contraseña a futuro.

```
--- Registro de Usuario ---
Nombre: Julian
Apellido: Silvera
Email: juliandsilvera@gmail.com
Contraseña: *****
Pregunta de seguridad (por ejemplo, ¿Cuál es el nombre de tu primera mascota?): Primera Mascota
Respuesta de seguridad: Celeste
Huésped registrado exitosamente.
Registro exitoso. Ahora puedes iniciar sesión.
Presiona cualquier tecla para continuar...
```

Como resultado todas las pruebas de registro fueron exitosas. Los usuarios se lograron registrar y se almacenaron correctamente.

## Pruebas de Inicio de Sesión:

El sistema permitió el inicio de sesión exitoso con credenciales válidas y bloqueó el acceso con credenciales inválidas, mostrando mensajes de error apropiados.

```
--- Inicio de Sesión ---
Email: juliandsilvera@gmail.com
Contraseña: *****
Bienvenido/a, Julian.
Presiona cualquier tecla para continuar...
```

```
--- Inicio de Sesión ---
Email: juliandsilvera@gmail.com
Contraseña: *****
Error: Las credenciales no coinciden. Inténtalo de nuevo.
Presiona cualquier tecla para continuar...
```

### Pruebas de Modificación de Reserva:

Se llevaron a cabo varios testeos y verificación de la capacidad del sistema para modificar tanto las fechas como las habitaciones de reservas existentes. Esta fase del testing fue particularmente desafiante, ya que implicaba asegurar que los cambios no afectarán la integridad de los datos y que se mantuviera la disponibilidad correcta de las habitaciones.

Tras realizar los test, se logró que todas las modificaciones se procesarán correctamente, con el sistema actualizando las reservas de acuerdo con los cambios solicitados.

```
--- Modificar Reserva ---
1. Cambiar Fecha
2. Cambiar Habitación
3. Volver al Menú Principal
Seleccione una opción: _
```

### Pruebas de Pagos:

Se validó el proceso de pago, tanto en efectivo como con tarjeta, asegurando que se registrara correctamente el estado de pago y se generará el comprobante.

Se lograron generar comprobantes en formato .txt y se actualizó el estado de las reservas a "pagada".

```
Ingrese el número de la habitación para realizar el pago: 1
Seleccione el método de pago:
1. Efectivo
2. Tarjeta
Opción:
```

 \*Comprobante\_Reserva\_6.txt: Bloc de notas

```
Archivo  Edición  Formato  Ver  Ayuda
Comprobante de Pago:
Reserva ID: 1
Habitación: 1
Desde: 9/11/2024 Hasta: 11/11/2024
Estado: Pagada
Monto Total: $238
Tipo de Pago: Tarjeta
```

### Conclusión de las Pruebas

Las pruebas realizadas evidencian el correcto funcionamiento del sistema. Todas las funcionalidades clave operan según lo previsto, proporcionando una experiencia de usuario fluida y eficiente.

# Explicación de funciones extras:

La función “**LeerContraseña**” es responsable de capturar la entrada de la contraseña del usuario de manera segura. Se muestra un mensaje en la consola pidiendo al usuario que ingrese su contraseña, sin revelar los caracteres que está introduciendo.

Se utilizó un bucle do-while para permitir que el usuario ingrese caracteres uno a uno hasta que presione la tecla Enter. Dentro del bucle, se usa `Console.ReadKey` para capturar la tecla presionada sin mostrarla en la consola. Esto permite que el sistema mantenga la privacidad de la contraseña.

Esta función permite a los usuarios ingresar sus contraseñas sin que sean visibles, protegiendo así su información sensible, brindándole una sensación de seguridad al introducir su contraseña.

```
private string LeerContraseña()
{
    string contraseña = "";
    ConsoleKeyInfo tecla;

    Console.Write("Contraseña: ");
    do
    {
        tecla = Console.ReadKey(intercept: true);
        if (tecla.Key == ConsoleKey.Backspace && contraseña.Length > 0)
        {
            contraseña = contraseña[..^1];
            Console.Write("\b \b");
        }
        else if (!char.IsControl(tecla.KeyChar))
        {
            contraseña += tecla.KeyChar;
            Console.Write("*");
        }
    } while (tecla.Key != ConsoleKey.Enter);

    Console.WriteLine();
    return contraseña;
}
```



## Paso a Paso para Generar un Comprobante en Formato .txt

### Iniciar Sesión en el Sistema

Abre la aplicación y en el menú principal, selecciona la opción para iniciar sesión. Si no tienes una cuenta, puedes registrarte fácilmente. Ingresa tu correo electrónico y contraseña. Si las credenciales son correctas, recibirás un mensaje de bienvenida.

```
--- Inicio de Sesión ---  
Email: crodriquez@ctc.edu.uy  
Contraseña: *****  
Bienvenido/a, Carlos.  
Presiona cualquier tecla para continuar...  
_
```

### Acceder al Menú de Usuario

Una vez dentro del sistema, accede al menú de usuario donde puedes realizar diversas acciones como reservas, pagos, etc.

```
--- Menú de Usuario ---  
1. Listar habitaciones disponibles  
2. Realizar reserva  
3. Modificar reserva  
4. Cancelar reserva  
5. Realizar pago  
6. Generar Comprobante  
7. Cambiar contraseña  
8. Listar huéspedes  
9. Ver mis reservas  
10. Habitaciones más reservadas  
11. Cerrar sesión  
12. Salir  
Selecciona una opción: _
```

### Realizar el Pago de la Reserva

Antes de generar un comprobante, asegúrate de que la reserva esté pagada. En el menú de usuario, selecciona la opción "Realizar pago" (Opción 5) y sigue las instrucciones para completar el pago de la habitación.

```
Selecciona una opción: 5  
Ingresa el número de la habitación para realizar el pago: 2  
Seleccione el método de pago:  
1. Efectivo  
2. Tarjeta  
Opción: 2  
Confirme el pago con tarjeta (S/N): s  
Pago realizado con éxito.  
Comprobante de Pago:  
Reserva ID: 5  
Habitación: 2  
Desde: 9/11/2024 Hasta: 11/11/2024  
Estado: Pagada  
Monto Total: $230  
Tipo de Pago: Tarjeta  
Presiona cualquier tecla para continuar...  
_
```

## Seleccionar la Opción de Generar Comprobante

En el menú de usuario, selecciona la opción para **"Generar Comprobante"**.

```
--- Menú de Usuario ---  
1. Listar habitaciones disponibles  
2. Realizar reserva  
3. Modificar reserva  
4. Cancelar reserva  
5. Realizar pago  
6. Generar Comprobante
```

## Ingresa el Número de la Habitación

Cuando se te solicita, ingresa el número de la habitación para la cual deseas generar el comprobante. Asegúrate de que la habitación tenga una reserva activa asociada.

## Confirmación de Generación del Comprobante

El sistema generará el comprobante en formato .txt. Si el proceso es exitoso, recibirás un mensaje de confirmación como en la imagen.

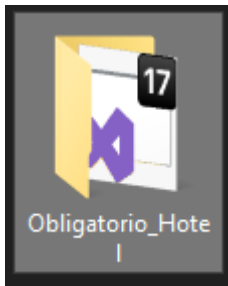
```
Selecciona una opción: 6  
Ingresa el número de la habitación para generar el comprobante: 2  
Comprobante de Pago:  
Reserva ID: 5  
Habitación: 2  
Desde: 9/11/2024 Hasta: 11/11/2024  
Estado: Pagada  
Monto Total: $230  
Tipo de Pago: Tarjeta  
  
Presiona cualquier tecla para continuar...  
_
```

## Ubicación del Comprobante

El archivo de texto con el comprobante se guardará en el directorio de la aplicación con un nombre que incluye el número de la reserva.

A continuación te adjunto imágenes para saber donde está ubicado:

Paso 1: Abrir la carpeta donde está el proyecto:



Paso 2: Seleccionar la carpeta:

.vs	6/11/2024 02:13	Carpeta de archivos	
Obligatorio_Hotel	6/11/2024 02:23	Carpeta de archivos	
Obligatorio_Hotel.sln	6/11/2024 02:13	Microsoft Visual S...	2 KB

Paso 3: Entrar a carpeta "bin":

bin	6/11/2024 02:13	Carpeta de archivos	
obj	6/11/2024 02:13	Carpeta de archivos	
Habitacion.cs	6/11/2024 02:17	C# Source File	1 KB
Hotel.cs	6/11/2024 02:23	C# Source File	16 KB
Huesped.cs	6/11/2024 02:17	C# Source File	2 KB
Menu.cs	6/11/2024 02:23	C# Source File	17 KB
Obligatorio_Hotel.csproj	6/11/2024 02:13	C# Project file	1 KB
Pago.cs	6/11/2024 02:17	C# Source File	2 KB
Program.cs	6/11/2024 02:17	C# Source File	1 KB
Reserva.cs	6/11/2024 02:17	C# Source File	4 KB

Paso 4: Carpeta "Debug":

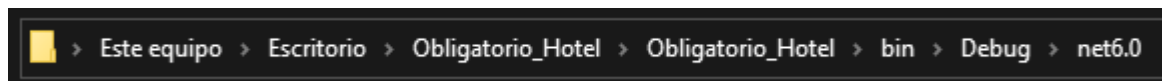
Debug	6/11/2024 02:13	Carpeta de archivos	
-------	-----------------	---------------------	--







Paso 5: Carpeta "net6.0":

net6.0	6/11/2024 02:45	Carpeta de archivos	
--------	-----------------	---------------------	--

## Ruta de comprobante:

C:\Users\NombreDeEquipo\Desktop\Obligatorio\_Hotel\Obligatorio\_Hotel\bin\Debug\net6.0



 Comprobante_Reserva_5.txt	6/11/2024 02:42	Documento de te...	1 KB
 Obligatorio_Hotel.deps.json	6/11/2024 02:17	JetBrains WebStorm	1 KB
 Obligatorio_Hotel.dll	6/11/2024 02:23	Extensión de la ap...	38 KB
 Obligatorio_Hotel.exe	6/11/2024 02:23	Aplicación	148 KB
 Obligatorio_Hotel.pdb	6/11/2024 02:23	Program Debug D...	21 KB
 Obligatorio_Hotel.runtimeconfig.json	6/11/2024 02:17	JetBrains WebStorm	1 KB

## Notas Adicionales:

- Asegúrate de que la habitación esté reservada antes de intentar generar un comprobante, ya que el sistema solo generará comprobantes para reservas válidas.
- En caso de que no haya reservas asociadas a la habitación, el sistema te mostrará un mensaje de error y no generará el comprobante.
- Se utilizó el método "System.IO.File.WriteAllText" para guardar el contenido de la variable comprobante en un archivo de texto. El nombre del archivo se genera dinámicamente utilizando el número de reserva, lo que permite identificar cada comprobante de manera única.
- Para manejar posibles errores durante el proceso de escritura en el archivo, se implementa un bloque try y catch. Si ocurre algún error, el programa captura la excepción y muestra un mensaje de error en la consola, lo que mejora la robustez y usabilidad del sistema.