

Tecnicatura Superior en Ciencias de Datos e Inteligencia Artificial

PROYECTO TECNOLÓGICO

Sistema de Gestión Hotelera Automatizado con Python y MySQL

Innovación en Gestión de Datos

Grupo 17

- Cáceres Giménez, Cesia Fiorella - **DNI:** 96320512
- Ferrere, Tomás - **DNI:** 44.762.329
- Juncos, David Eduardo - **DNI:** 29.686.787
- Mackinson, John - **DNI:** 45.837.754

Docentes:

Romina Cattaneo

Julián Conde

1 de Noviembre 2024

Descripción del Proyecto	3
Etapas del Proyecto	3
Justificación	4
Objetivos	5
Objetivo General	5
Objetivos Específicos	5
Relevancia del Proyecto	6
Metodología	7
CRONOGRAMA	8
Presentación del Proyecto	1
Conclusiones	1

Descripción del Proyecto

Este proyecto consiste en el desarrollo de una aplicación de consola en Python que facilite la gestión de un entorno hotelero. Utilizando Python como lenguaje de programación y MySQL como sistema de gestión de bases de datos (SGBD), se busca diseñar un sistema que optimice la administración de reservas, asignación de habitaciones, gestión de clientes y generación de informes de ocupación y rendimiento. La aplicación permitirá a los empleados del hotel realizar estas tareas de manera rápida y precisa, minimizando el margen de error humano y asegurando un flujo de trabajo ágil y seguro.

Etapas del Proyecto

1. **Análisis de Requerimientos:** Identificación de necesidades funcionales (gestión de reservas, clientes, habitaciones, informes) y no funcionales (seguridad, facilidad de uso).
2. **Diseño del Modelo de Datos:** Creación de un modelo relacional en MySQL que almacene y relacione la información de clientes, reservas y habitaciones.
3. **Implementación del Sistema:** Desarrollo de un programa en Python, estructurado con conceptos de Programación Orientada a Objetos (POO), que permita interactuar con la base de datos.
4. **Pruebas y Depuración:** Validación de todas las funcionalidades para asegurar el correcto flujo de datos entre la aplicación y la base de datos, y optimización del rendimiento.
5. **Documentación:** Redacción de la documentación técnica, incluyendo guías de instalación, manual de usuario, y comentarios en el código.

Justificación

Relevancia: La industria hotelera se enfrenta constantemente a la necesidad de mejorar la eficiencia en sus procesos de gestión. Implementar una aplicación que automaticice el manejo de datos de clientes y reservas no solo optimiza la administración diaria del hotel, sino que también facilita el acceso a datos estratégicos para la toma de decisiones. Este proyecto resulta relevante porque permite a los estudiantes aplicar conocimientos teóricos adquiridos en los cursos de Programación y Bases de Datos en un contexto realista, similar al de la industria hotelera.

Contexto y Necesidad: En muchos hoteles pequeños y medianos, la administración se realiza manualmente o con herramientas rudimentarias, lo que da lugar a posibles errores, demoras en la atención y una mayor carga administrativa. La necesidad de una aplicación como esta surge de la falta de herramientas accesibles y específicas que optimicen los procesos de reservas, asignación de habitaciones y consultas de disponibilidad.

Impacto: Este proyecto beneficiará a los usuarios potenciales del sistema hotelero al mejorar la eficiencia y precisión en el registro y consulta de datos, minimizando errores y asegurando una gestión más profesional. Además, el proyecto permite a los estudiantes desarrollar habilidades prácticas en programación y diseño de bases de datos.

Objetivos

Objetivo General

Desarrollar un sistema eficiente y seguro que automatice los procesos clave de la gestión hotelera, mejorando la precisión en el registro de datos y facilitando la generación de informes de ocupación y rendimiento.

Objetivos Específicos

1. Diseñar y crear una base de datos relacional en MySQL que gestione la información de clientes, reservas y habitaciones, garantizando integridad y consistencia.
2. Implementar en Python funcionalidades básicas y avanzadas como registro de clientes, consulta de disponibilidad de habitaciones, gestión de reservas y generación de informes.
3. Aplicar principios de Programación Orientada a Objetos (POO) para estructurar el código de manera modular y escalable.
4. Realizar pruebas exhaustivas del sistema, identificando y corrigiendo errores para asegurar un funcionamiento fluido y sin problemas.
5. Documentar el código y el proceso de desarrollo, siguiendo buenas prácticas de programación y garantizando que el sistema sea comprensible y mantenable.

Relevancia del Proyecto

Este proyecto es relevante tanto a nivel académico como profesional por varias razones:

1. **Aplicación de Conocimientos en Contextos Reales:** A través de este proyecto, los estudiantes pueden aplicar de forma práctica los conocimientos adquiridos en las asignaturas de Programación y Bases de Datos, enfrentándose a desafíos que imitan problemas y escenarios reales en la industria de la hospitalidad. Esto permite a los estudiantes consolidar sus habilidades en un entorno controlado y simulado, similar a situaciones profesionales.
2. **Desarrollo de Competencias Profesionales:** El proyecto fomenta habilidades técnicas esenciales, como la programación en Python y la gestión de bases de datos en MySQL, y competencias interpersonales y de trabajo en equipo, necesarias para resolver problemas complejos de manera colaborativa. Además, les permite desarrollar prácticas de documentación y gestión de código con Git, que son estándares en la industria del desarrollo de software.
3. **Beneficios para el Sector Hotelero:** La automatización de la gestión hotelera tiene un impacto positivo en la eficiencia operativa, ya que facilita la administración de reservas, habitaciones y clientes, y mejora la precisión de los datos. Esto puede reducir los costos operativos y mejorar la experiencia del cliente, haciendo que el hotel sea más competitivo y brindando un servicio de mayor calidad.
4. **Adaptabilidad a Diferentes Escenarios de Negocio:** Aunque está diseñado inicialmente para el sector hotelero, el sistema puede adaptarse fácilmente a otros tipos de negocios de la industria del turismo y la hospitalidad, como hostales y centros turísticos. Esto amplía el alcance del proyecto y su aplicabilidad en diferentes escenarios, aumentando el valor de la experiencia adquirida por los estudiantes.
5. **Contribución a la Formación Profesional de Técnicos Especializados:** Este proyecto contribuye a la formación de técnicos calificados que pueden responder a las demandas tecnológicas actuales del mercado laboral en el área de desarrollo de software y gestión de datos. Desarrollar un sistema práctico y funcional en un contexto hotelero les da a los estudiantes una ventaja competitiva y los prepara para enfrentar de manera competente desafíos en el ámbito profesional.

Metodología

- **Enfoque Metodológico:** Se seguirá una metodología ágil, enfocada en entregas frecuentes y ajustándose a los requerimientos del cliente ficticio (representado por un hotel típico de tamaño medio). Esta metodología permite iterar sobre el desarrollo y adaptar el sistema en función de las pruebas y de los comentarios recibidos durante el proceso.
- **Fases del Proyecto:**
 - **Análisis de Requerimientos:** Reunión y análisis de las necesidades del hotel en términos de funcionalidad (tareas de gestión específicas) y características técnicas (facilidad de uso, acceso a datos, velocidad).
 - **Diseño de Base de Datos:** Creación del modelo en MySQL, utilizando MySQL Workbench para definir tablas, relaciones, y restricciones que aseguren la integridad de los datos.
 - **Desarrollo del Sistema:** Desarrollo del sistema en Python, estructurando el código con clases y métodos bajo el paradigma de POO. Se emplearán librerías como `mysql-connector-python` para la conexión entre Python y MySQL.
 - **Pruebas y Depuración:** Realización de pruebas unitarias y de integración para cada módulo, depuración de errores y optimización de la eficiencia del sistema.
 - **Documentación:** Redacción de documentación detallada, incluyendo la guía de usuario, instrucciones de instalación y comentarios en el código para su comprensión y mantenimiento.
- **Herramientas Utilizadas:**
 - **Python:** Lenguaje de programación principal, estructurado con POO.
 - **MySQL:** Sistema de gestión de bases de datos relacional.
 - **MySQL Workbench:** Para el diseño y la gestión del modelo de datos.
 - **Git y GitHub:** Para el control de versiones y la gestión colaborativa del código.
- **Equipo Humano y Técnico:**
 - *Equipo Humano:* Estudiantes desarrolladores, bajo la supervisión de un instructor.
 - *Recursos Técnicos:* Computadoras, acceso a internet, software requerido (Python, MySQL Workbench, Git).

CRONOGRAMA

CRONOGRAMA								
ETAPA DEL PROYECTO	ACTIVIDAD	MES						
		Septiembre		Octubre				Noviembre
		Semana 3	Semana 4	Semana 1	Semana 2	Semana 3	Semana 4	Semana 1
Análisis de Requerimientos	Recopilación de requisitos y análisis	✓						
	Definición de alcances y funcionalidades		✓					
Diseño de la Base de Datos	Diseño del modelo de datos (DER)			✓				
	Creación de tablas y relaciones en MySQL			✓				
Desarrollo del Sistema	Configuración de entorno y dependencias				✓			
	Desarrollo del código en Python				✓	✓		
	Integración de funcionalidades.						✓	
Pruebas y Depuración	Depuración y Optimización							✓
Documentación y Entrega	Capturas y preparación para entrega final							✓

Presentación del Proyecto

Menú Principal.

```
--- Menú Principal ---
1. Menú de Usuarios
2. Ingresar al sistema con los datos de usuario
3. Análisis de datos
4. Salir
```

Menú de Usuarios

```
--- Menú de Usuarios ---
1. Crear Usuario
2. Modificar Usuario
3. Eliminar Usuario
4. Buscar Usuario
5. Mostrar Usuarios
6. Ordenar Usuarios
7. Cargar Accesos
8. Iniciar Sesión
9. Salir
Selecciona una opción:
```

Selección opción 1 (Menú de usuarios): Crear Usuario, se pide que ingrese ID, nombre, contraseña, email, DNI y lo ordena por DNI y crea y registra en usuarios.ispc

```
Selecciona una opción: 1
Ingresa el ID del usuario: 14
Ingresa el nombre de usuario: edgar
Ingresa la contraseña: e1234
Ingresa el email: edgar@gmail.com
Ingrrese el D.N.I.:45963121
Usuario edgar agregado y usuarios ordenados por DNI.
Usuario edgar creado y registrado en 'usuarios.ispc'.
```

Selección opción 2 (Menú de usuarios): Modifica Usuario, se pide que ingrese el nombre de usuario a modificar, si lo encuentra va pidiendo nuevo nombre, contraseña, email, DNI si desea modificar si los

datos quedan como están presiona tecla enter, continua si hacer modificación de este campo, estas modificaciones se registran en usuarios.ispc

```
Selecciona una opción: 2
Ingresa el username del usuario a modificar: edgar
Nuevo username (dejar vacío para no modificar): ernesto
Nueva contraseña (dejar vacío para no modificar):
Nuevo email (dejar vacío para no modificar):
Nuevo DNI (dejar vacío para no modificar):
Usuario edgar modificado exitosamente.
```

Selección opción 3 (Menú de usuarios): Elimina Usuario, te consulta username o email e ingresandolo lo borra del archivo usuarios.ispc

```
Selecciona una opción: 3
Ingresa el username o email del usuario a eliminar: guille
Usuario guille eliminado exitosamente.
```

Selección opción 4 (Menú de usuarios): Menu de búsqueda de Usuarios, podes elegir los distintos tipos de búsquedas .

```
--- Menú de Búsqueda de Usuarios ---
1) Buscar por Username
2) Buscar por DNI (búsqueda binaria)
3) Buscar por Email (búsqueda secuencial)
Seleccione una opción: |
```

Selección opción 1 (Menú búsqueda de usuarios). Buscas el usuario a través de su Username .

```
Seleccione una opción: 1
Ingresa el username a buscar: david
Búsqueda Secuencial por Username: buscando 'david'
Intento 1: david es distinto a ghf
Intento 2: david es distinto a fgh
Intento 3: david es distinto a pepito
No se encontró el username 'david' en 3 intentos.
```

Selección opción 2 (Menú búsqueda de usuarios). Buscas el usuario a través de su DNI.

```
--- Menú de Búsqueda de Usuarios ---
1) Buscar por Username
2) Buscar por DNI (búsqueda binaria)
3) Buscar por Email (búsqueda secuencial)
Seleccione una opción: 2
Ingrese el DNI a buscar: 29686787
Se utilizó la búsqueda por DNI y la técnica de búsqueda fue método binario.
Usuario encontrado: Usuario(ID: 7, Username: david, Email: david@gmail.com, dni: 29686787)

--- Menú de Usuarios ---
1. Crear Usuario
2. Modificar Usuario
3. Eliminar Usuario
4. Buscar Usuario
5. Mostrar Usuarios
6. Ordenar Usuarios
7. Cargar Accesos
8. Iniciar Sesión
9. Salir
Selecciona una opción: 1
```

Selección opción 3 (Menú búsqueda de usuarios). Buscas el usuario a través de su Email.

```
Búsqueda Secuencial por Email: buscando '123'
Intento 1: 123 es distinto a gfhfgh
Intento 2: 123 es distinto a fhghf
Intento 3: 123 es distinto a pepito@gmail.com
No se encontró el email '123' en 3 intentos.
```

Selección opción 5 (Menú de usuarios): Muestra todos los usuarios registrados en el archivo usuario.ispc y también muestra los usuarios ordenados por orden alfabético.

Selecciona una opción: 5

```
--- Usuarios del archivo 'usuarios.ispc' ---
Usuario(ID: 12, Username: pepe, Email: pepe@gmail.com, dni: 4698369)
Usuario(ID: 10, Username: quique, Email: quique@gmail.com, dni: 5396396)
Usuario(ID: 6, Username: elsa, Email: elsa@gmail.com, dni: 10463596)
Usuario(ID: 8, Username: clara, Email: clara@gmail.com, dni: 10596321)
Usuario(ID: 9, Username: edu, Email: edu@gmail.com, dni: 11632456)
Usuario(ID: 7, Username: david, Email: david@gmail.com, dni: 29686787)
Usuario(ID: 2, Username: ruben, Email: ruben@gmail.com, dni: 29686788)
Usuario(ID: 15, Username: nico, Email: nico@gmail.com, dni: 33665987)
Usuario(ID: 4, Username: john, Email: john@gmail.com, dni: 36987987)
Usuario(ID: 1, Username: tomas, Email: tomas@gmail.com, dni: 39654456)
Usuario(ID: 3, Username: cesia, Email: cesia@gmail.com, dni: 40890123)
Usuario(ID: 5, Username: Ana, Email: ana@gmail.com, dni: 50410400)
Usuario(ID: 11, Username: estela, Email: estela@gmail.com, dni: 60539874)
```

```
--- Usuarios del archivo 'usuariosOrdenadosPorUsername.ispc' ---
Usuario(ID: 5, Username: Ana, Email: ana@gmail.com, dni: 50410400)
Usuario(ID: 3, Username: cesia, Email: cesia@gmail.com, dni: 40890123)
Usuario(ID: 8, Username: clara, Email: clara@gmail.com, dni: 10596321)
Usuario(ID: 7, Username: david, Email: david@gmail.com, dni: 29686787)
Usuario(ID: 9, Username: edu, Email: edu@gmail.com, dni: 11632456)
Usuario(ID: 6, Username: elsa, Email: elsa@gmail.com, dni: 10463596)
Usuario(ID: 11, Username: estela, Email: estela@gmail.com, dni: 60539874)
Usuario(ID: 4, Username: john, Email: john@gmail.com, dni: 36987987)
Usuario(ID: 12, Username: pepe, Email: pepe@gmail.com, dni: 4698369)
Usuario(ID: 10, Username: quique, Email: quique@gmail.com, dni: 5396396)
Usuario(ID: 2, Username: ruben, Email: ruben@gmail.com, dni: 29686788)
Usuario(ID: 1, Username: tomas, Email: tomas@gmail.com, dni: 39654456)
```

Selección opción 6 (Menú de usuarios): Ordena los usuarios por el método burbuja o python según cuál selecciones .

Selecciona una opción: 6

Selecciona el método de ordenación (burbuja o python): burbuja
 Usuarios ordenados usando Burbuja.

Selección opción 7 (Menú de usuarios): Muestra los registros de accesos de los usuarios.

Selecciona una opción: 7

```
Usuario: j, Fecha de acceso: 2024-10-31 19:36:14.986419
Usuario: j, Fecha de acceso: 2024-10-31 19:36:52.162930
Usuario: j, Fecha de acceso: 2024-10-31 19:39:32.995543
```

Selección opción 8 (Menú de usuarios): Inicio de sesión introduciendo los datos requeridos que son nombre de usuario y contraseña.

```
Selecciona una opción: 8
Ingresa tu nombre de usuario: pepe
Ingresa tu contraseña: p1234
Búsqueda realizada por técnica binaria.
Usuario no encontrado.
Se registró el intento fallido de pepe en 'logs.txt'.
```

Selección opción 9 (Menú de usuarios): Para volver al menú principal

```
Selecciona una opción: 9
Saliendo del menú de usuarios.
```

Selección opción 2 (Menú de usuarios): Ingresa al menú del sistema de gestión hotelera con los datos del usuario

```
--- Menú Principal ---
1. Menú de Usuarios
2. Ingresar al sistema con los datos de usuario
3. Análisis de datos
4. Salir
Seleccione una opción: 2

--- Sistema de Gestión del Hotel ---
1. Registrar cliente
2. Hacer reserva
3. Consultar disponibilidad
4. Generar informe de mantenimientos
5. Generar informe de reservas activas
6. Buscar Cliente por DNI
7. Salir
Seleccione una opción: 1
```

Selección opción 1 (Menú gestión del hotel): Registrar cliente introduciendo su DNI, nombre, apellido, dirección, email, número de teléfono.

```
Seleccione una opción: 1
Ingrese DNI: 29686788
Ingrese nombre: ruben
Ingrese apellido: juncos
Ingrese dirección: dir2
Ingrese email: ruben@gmail.com
Ingrese número de teléfono: 123456
```

Selección opción 2 (Menú gestión del hotel): Generar una reserva ingresando ID de la persona que hace la reserva, de la habitación, del cliente y se debe ingresar las fechas de entrada y salida.

```
Seleccione una opción: 2
Ingrese ID del personal que hace la reserva: 1
Ingrese ID de la habitación: 1
Ingrese ID del cliente: 29686787
Ingrese fecha de entrada (YYYY-MM-DD HH:MM): 2024-11-05
Ingrese fecha de salida (YYYY-MM-DD HH:MM): 2024-11-10
```

Selección opción 3 (Menú gestión del hotel): Ingresando las fechas de entrada y salida sabrás que habitaciones estarán disponibles.

Seleccione una opción: 3

Ingresar fecha de entrada (YYYY-MM-DD HH:MM): 2024-11-01

Ingresar fecha de salida (YYYY-MM-DD HH:MM): 2024-11-30

Habitación 101 - Tipo: Suite - Precio: 100000.00

Habitación 104 - Tipo: Doble Estándar - Precio: 25000.00

Habitación 107 - Tipo: Apartamento - Precio: 50000.00

Habitación 111 - Tipo: Doble Estándar - Precio: 25000.00

Habitación 113 - Tipo: Individual - Precio: 18000.00

Habitación 116 - Tipo: Apartamento - Precio: 50000.00

Habitación 119 - Tipo: Individual - Precio: 18000.00

Habitación 123 - Tipo: Suite - Precio: 100000.00

Habitación 126 - Tipo: Individual - Precio: 18000.00

Habitación 127 - Tipo: Apartamento - Precio: 50000.00

Habitación 130 - Tipo: Suite - Precio: 100000.00

Habitación 132 - Tipo: Doble Estándar - Precio: 25000.00

Habitación 136 - Tipo: Suite - Precio: 100000.00

Habitación 139 - Tipo: Individual - Precio: 18000.00

Habitación 142 - Tipo: Suite - Precio: 100000.00

Habitación 146 - Tipo: Individual - Precio: 18000.00

Habitación 149 - Tipo: Suite - Precio: 100000.00

Selección opción 4 (Menú gestión del hotel): Ingresando esta opción te da el informe de mantenimiento

Informe de Mantenimientos						
Habitación	Estado	Fecha Reporte	Fecha Resolución	Descripción Problema	Costo	Personal a Cargo
113	En Proceso	2024-10-05	No resuelto	Reparar tapa del inodoro	N/A	Ana Martínez
106	Pendiente	2024-10-05	2024-10-09	Revisar cerradura de la puerta	N/A	María González
117	Completado	2024-10-04	No resuelto	Reparar silla rota	N/A	María González
105	En Proceso	2024-10-04	2024-10-08	Pintura de pared desgastada	N/A	Pedro López
110	Pendiente	2024-10-04	2024-10-08	Ajustar ventana que no cierra bien	N/A	Pedro López
130	Completado	2024-10-03	2024-10-07	Reemplazar bombilla del plafón	N/A	Ana Martínez
120	Completado	2024-10-03	No resuelto	Cambiar cerradura de la puerta principal	N/A	María González
109	En Proceso	2024-10-03	2024-10-07	Revisar sistema eléctrico	N/A	Ana Martínez
112	Pendiente	2024-10-03	No resuelto	Desinfectar moho en la pared	N/A	Ana Martínez
116	Pendiente	2024-10-03	No resuelto	Cambiar bombilla de la lámpara de lectura	N/A	Ana Martínez
111	Completado	2024-10-02	No resuelto	Inspeccionar fuga en el aire acondicionado	N/A	Ana Martínez
120	En Proceso	2024-10-02	2024-10-06	Ajustar calefacción que no enciende	N/A	Ana Martínez
108	Pendiente	2024-10-02	2024-10-06	Reemplazar colchón dañado	N/A	Ana Martínez
115	Pendiente	2024-10-02	No resuelto	Ajustar radiador que no calienta	N/A	Ana Martínez
119	Pendiente	2024-10-02	No resuelto	Inspeccionar fugas en la ducha	N/A	Ana Martínez
107	Completado	2024-10-01	2024-10-05	Reparar grifo que gotea	N/A	Ana Martínez
114	Completado	2024-10-01	No resuelto	Reemplazar cortinas rotas	N/A	Pedro López
118	En Proceso	2024-10-01	No resuelto	Desinfectar moho en el baño	N/A	Pedro López
110	Pendiente	2024-10-01	2024-10-05	Reparar fuga de agua en el baño	N/A	Ana Martínez
140	Pendiente	2024-10-01	2024-10-05	Limpieza de aire acondicionado	N/A	Ana Martínez

Selección opción 5 (Menú gestión del hotel): Ingresando esta opción te da el informe de todas las reservas activas

Informe de Reservas Activas					
ID Reserva	Nombre Cliente	Apellido Cliente	Habitación	Fecha Entrada	Fecha Salida
1	Juan	Pérez	101	2024-10-10	2024-10-15
2	Ana	García	102	2024-10-12	2024-10-14
3	Luis	Rodríguez	103	2024-10-11	2024-10-16
4	Juan	Pérez	104	2024-10-05	2024-10-09
5	María	López	120	2024-10-07	2024-10-10
6	Ana	García	125	2024-10-09	2024-10-13
7	Carlos	Martínez	132	2024-10-01	2024-10-05
8	Laura	González	135	2024-10-04	2024-10-06

Selección opción 6 (Menú gestión del hotel): Buscador de clientes dando su información ingresando su DNI.

--- Sistema de Gestión del Hotel ---						
1. Registrar cliente						
2. Hacer reserva						
3. Consultar disponibilidad						
4. Generar informe de mantenimientos						
5. Generar informe de reservas activas						
6. Buscar Cliente por DNI						
7. Salir						
Seleccione una opción: 6						
Ingrese el DNI del cliente: 29686787						
<i>Información del Cliente</i>						
ID	DNI	Nombre	Apellido	Dirección	Email	Teléfono
51	29686787	david	juncos	dir1	mail@gmail.com	123456

Selección opción 7 (Menú gestión del hotel): Para volver al menú principal.

--- Sistema de Gestión del Hotel ---						
1. Registrar cliente						
2. Hacer reserva						
3. Consultar disponibilidad						
4. Generar informe de mantenimientos						
5. Generar informe de reservas activas						
6. Buscar Cliente por DNI						
7. Salir						
Seleccione una opción: 7						
Saliendo del sistema.						
<i>Menú Principal</i>						

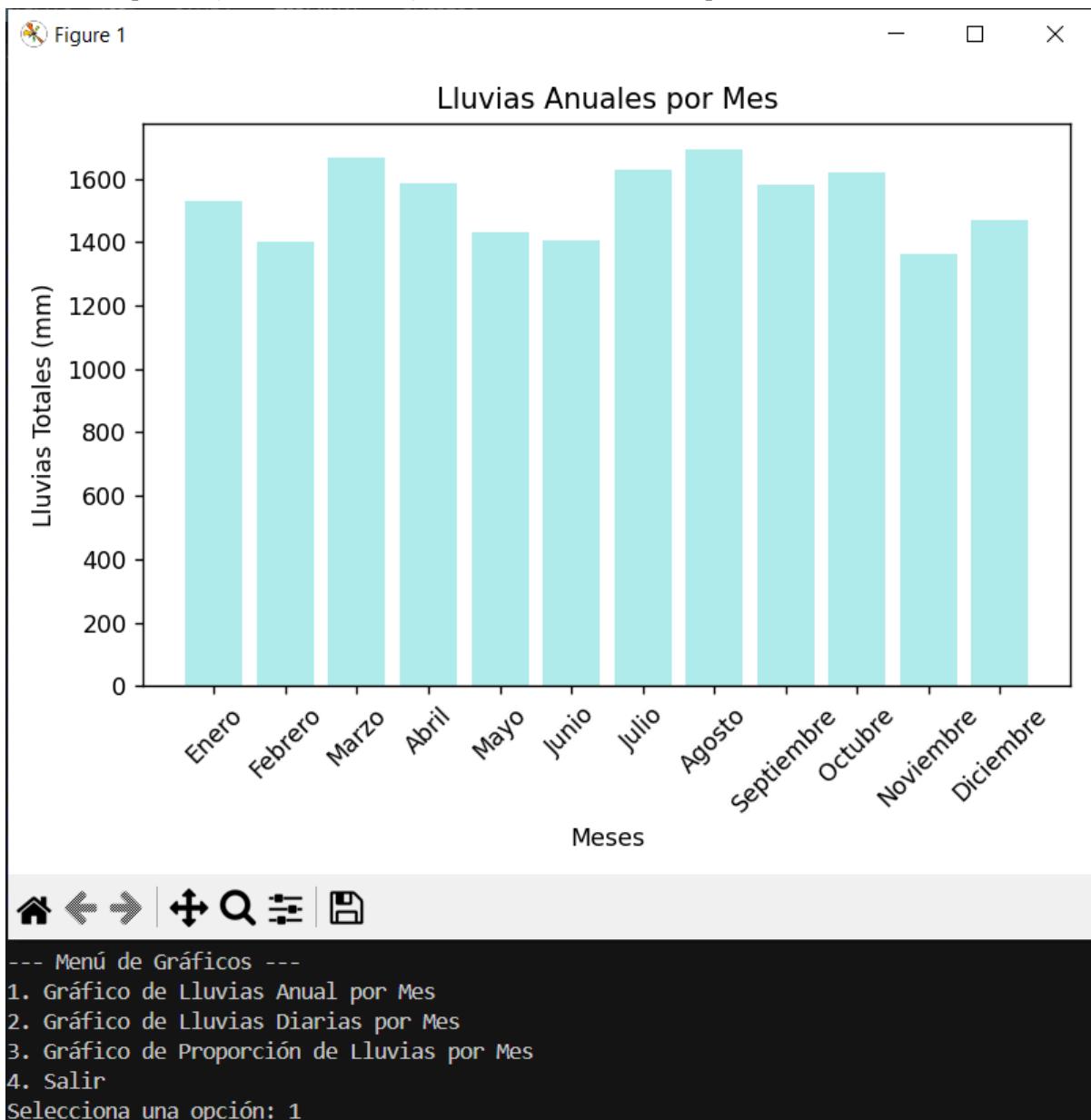
Selección opción 3 (Menú de usuarios): Ingresando el año y el mes de registros pluviales vas adquirir los registros de lluvia de todos los días de ese mes.

```
Selecciona una opción: 3
Ingresa el año para cargar registros pluviales: 2024
Registros guardados en c:\Users\Usuario\Desktop\Evidencia3\Proyecto-integrador-grupo17\Proyecto_Final\Programacion\datosAnalizados\registroPluvial2024.csv.
Ingresa el mes para mostrar los registros (1-12): 1
Registros de lluvia para el mes 1:
Día 1: 90 mm
Día 2: 39 mm
Día 3: 40 mm
Día 4: 61 mm
Día 5: 34 mm
Día 6: 22 mm
Día 7: 97 mm
Día 8: 28 mm
Día 9: 1 mm
Día 10: 19 mm
Día 11: 92 mm
Día 12: 85 mm
Día 13: 30 mm
Día 14: 67 mm
Día 15: 34 mm
Día 16: 88 mm
Día 17: 70 mm
Día 18: 43 mm
Día 19: 22 mm
Día 20: 80 mm
Día 21: 55 mm
Día 22: 81 mm
Día 23: 94 mm
Día 24: 12 mm
Día 25: 50 mm
Día 26: 39 mm
Día 27: 20 mm
Día 28: 1 mm
Día 29: 9 mm
```

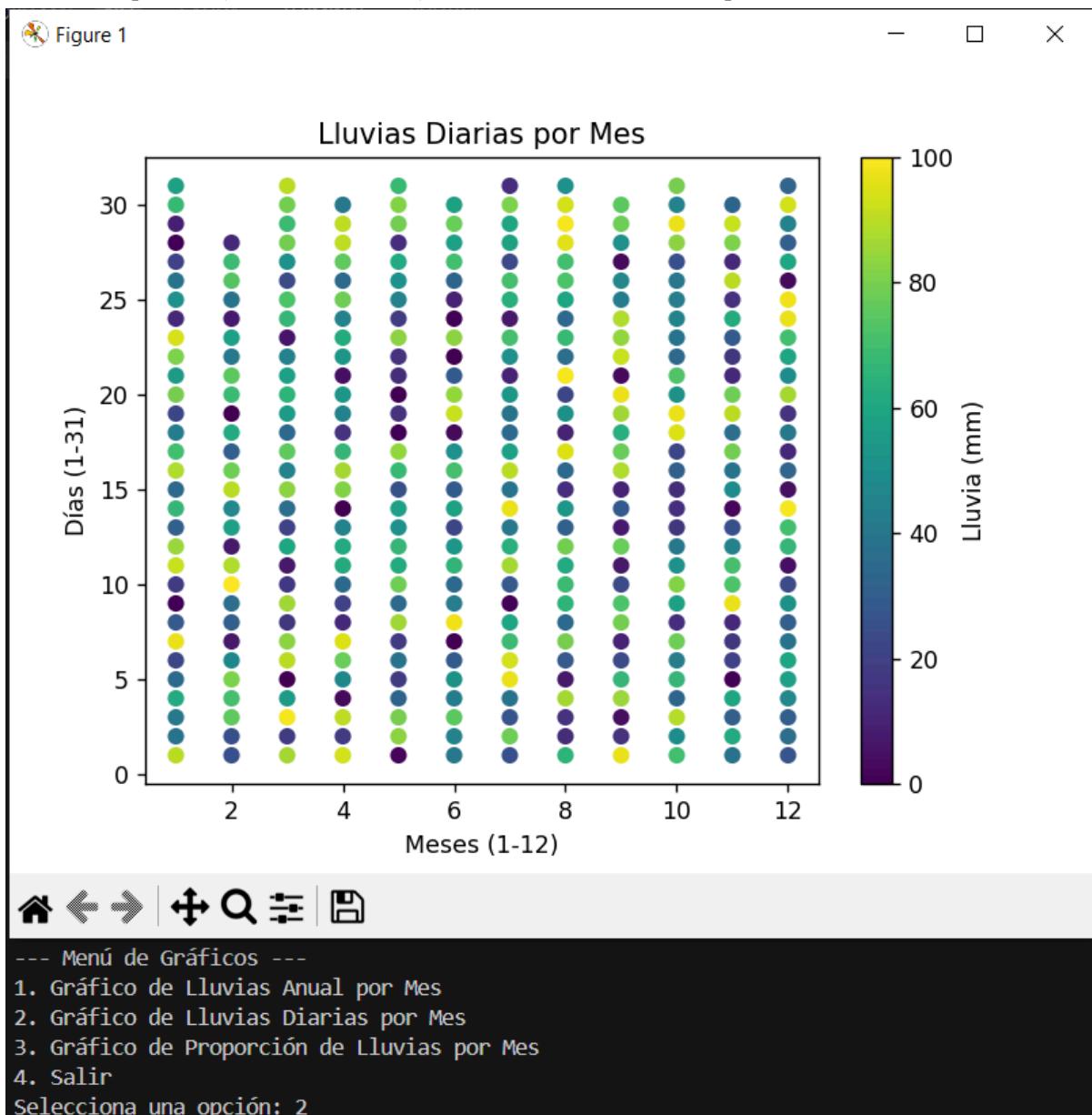
Menú de Gráficos.

- ```
--- Menú de Gráficos ---
1. Gráfico de Lluvias Anual por Mes
2. Gráfico de Lluvias Diarias por Mes
3. Gráfico de Proporción de Lluvias por Mes
4. Salir
```

Selección opción 1(Menú de Gráficos) Gráfico de Lluvias Anual por Mes.



Selección opción 2 (Menú de Gráficos) Gráfico de Lluvias Diarias por Mes.

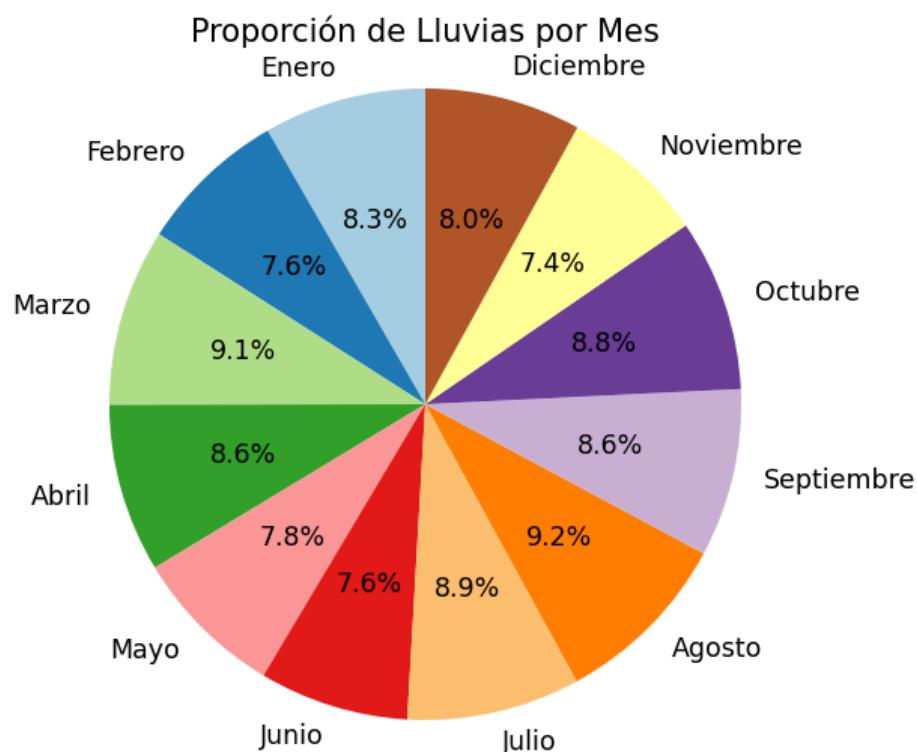


## Selección opción 3 (Menú de Gráficos) Gráfico de Proporción de Lluvias por Mes



Figure 1

-



--- Menú de Gráficos ---

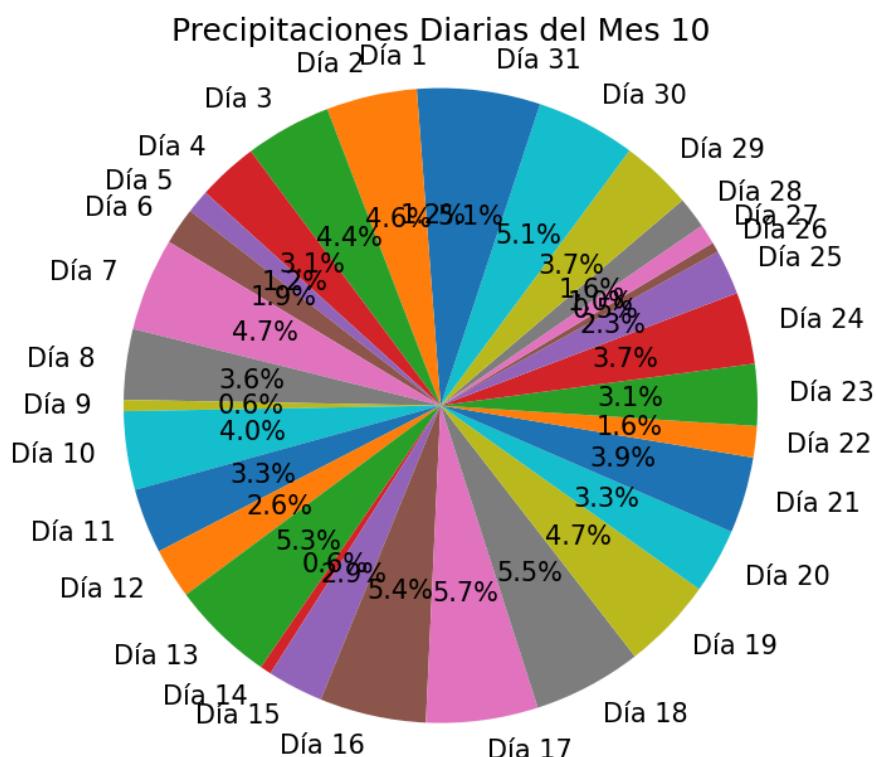
1. Gráfico de Lluvias Anual por Mes
2. Gráfico de Lluvias Diarias por Mes
3. Gráfico de Proporción de Lluvias por Mes
4. Salir

Selecciona una opción: 3



Selección opción 4 (Menú de Gráficos): Gráfico de Precipitaciones diarias en un mes:

---



Selección opción 5 (Menú de Gráficos): Datos Estadísticos

Estadísticas de Precipitaciones para el Mes 1:  
 Máxima precipitación: 97 mm  
 Mínima precipitación: 1 mm  
 Promedio de precipitación: 49.32 mm

Selección opción 5 (Menú de Gráficos): Para volver al menú principal

--- Menú de Gráficos ---  
 1. Gráfico de Lluvias Anual por Mes  
 2. Gráfico de Lluvias Diarias por Mes  
 3. Gráfico de Proporción de Lluvias por Mes  
 4. Salir  
 Selecciona una opción: 4

--- Menú Principal ---  
 1. Menú de Usuarios  
 2. Ingresar al sistema con los datos de usuario  
 3. Análisis de datos  
 4. Salir  
 Selecciona una opción: 4

Todos los Scripts de la base de datos

Base de datos:

```

CREATE SCHEMA IF NOT EXISTS `hoteles`;
USE `hoteles`;

-- -----
-- Tabla `cliente`

CREATE TABLE IF NOT EXISTS `cliente` (
 id INT NOT NULL PRIMARY KEY AUTO_INCREMENT,
 dni VARCHAR(10) NOT NULL UNIQUE,
 nombre VARCHAR(100) NOT NULL,
 apellido VARCHAR(100) NOT NULL,
 direccion VARCHAR(100) NOT NULL,
 email VARCHAR(50) NOT NULL UNIQUE,
 numeroTelefono VARCHAR(15) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb3;
You, la semana pasada • Agg scrip corregido
-- -----
-- Tabla `disponibilidad`

CREATE TABLE IF NOT EXISTS `disponibilidad` (
 id INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
 disponibilidad VARCHAR(100) NOT NULL UNIQUE
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb3;

-- Valores iniciales para `disponibilidad`
INSERT INTO `disponibilidad` (disponibilidad) VALUES
('Disponible'), ('Reservada'), ('Ocupada'), ('Mantenimiento');

-- -----
-- Tabla `tipoHabitacion`

CREATE TABLE IF NOT EXISTS `tipoHabitacion` (
 id INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
 tipo VARCHAR(30) NOT NULL,
 precioNoche DECIMAL(10,2) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb3;

-- Valores iniciales para `tipoHabitacion`
INSERT INTO `tipoHabitacion` (tipo, precioNoche) VALUES
('Suite', 100000), ('Doble Estándar', 25000), ('Individual', 18000), ('Apartamento', 50000);

-- -----
-- Tabla `rol`

```

```
-- Tabla `rol`

--

CREATE TABLE IF NOT EXISTS `rol` (
| id INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
| rol VARCHAR(100) NOT NULL UNIQUE
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb3;

--

-- Tabla `habitacion`

--

CREATE TABLE IF NOT EXISTS `habitacion` (
| id INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
| numeroHabitacion INT NOT NULL UNIQUE,
| idTipoHabitacion INT NOT NULL,
| idDisponibilidad INT NOT NULL,
| FOREIGN KEY (idDisponibilidad) REFERENCES `disponibilidad` (id),
| FOREIGN KEY (idTipoHabitacion) REFERENCES `tipoHabitacion` (id)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb3;

--

-- Tabla `personal`

--

CREATE TABLE IF NOT EXISTS `personal` (
| id INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
| nombre VARCHAR(100) NOT NULL,
| apellido VARCHAR(100) NOT NULL,
| correo VARCHAR(50) UNIQUE,
| numeroTelefono VARCHAR(15),
| horaEntrada TIME NOT NULL,
| horaSalida TIME NOT NULL,
| idRol INT NOT NULL,
| FOREIGN KEY (idRol) REFERENCES `rol` (id)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb3;

CREATE TABLE IF NOT EXISTS `estadoMantenimiento` (
| id INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
| estado VARCHAR(30) NOT NULL UNIQUE
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb3;

-- Valores iniciales para `estadoMantenimiento`
INSERT INTO `estadoMantenimiento` (estado) VALUES
('Pendiente'), ('En Proceso'), ('Completado');
```

```
-- Tabla `reserva`
--
CREATE TABLE IF NOT EXISTS `reserva` (
 id INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
 idPersonal INT NOT NULL,
 idHabitacion INT NOT NULL,
 idCliente INT NOT NULL,
 fechaEntrada DATETIME NOT NULL,
 fechaSalida DATETIME NOT NULL,
 estadoReserva ENUM('activa', 'completada', 'cancelada') DEFAULT 'activa',
 FOREIGN KEY (idCliente) REFERENCES `cliente`(id) ON DELETE CASCADE,
 FOREIGN KEY (idHabitacion) REFERENCES `habitacion`(id),
 FOREIGN KEY (idPersonal) REFERENCES `personal`(id)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb3;

--
-- Tabla `servicio`
--
CREATE TABLE IF NOT EXISTS `servicio` (
 id INT AUTO_INCREMENT PRIMARY KEY,
 nombreServicio VARCHAR(45),
 descripcion VARCHAR(85),
 precio DECIMAL(10,2) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb3;

--
-- Tabla `servicioConsumido`
--
CREATE TABLE IF NOT EXISTS `servicioConsumido` (
 id INT AUTO_INCREMENT PRIMARY KEY,
 idReserva INT NOT NULL,
 idServicio INT NOT NULL,
 cantidad INT NOT NULL,
 FOREIGN KEY (idReserva) REFERENCES `reserva`(id),
 FOREIGN KEY (idServicio) REFERENCES `servicio`(id)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb3;
```

```
-- -----
-- Tabla `mantenimiento` --
-- -----
CREATE TABLE IF NOT EXISTS `mantenimiento` (
 id INT AUTO_INCREMENT PRIMARY KEY,
 idHabitacion INT NOT NULL,
 idPersonal INT NOT NULL,
 descripcionProblema TEXT NOT NULL,
 fechaReporte DATE NOT NULL,
 fechaResolucion DATE,
 idEstadoMantenimiento INT NOT NULL,
 costo DECIMAL(10,2),
 FOREIGN KEY (idHabitacion) REFERENCES `habitacion`(id),
 FOREIGN KEY (idPersonal) REFERENCES `personal`(id),
 FOREIGN KEY (idEstadoMantenimiento) REFERENCES `estadoMantenimiento`(id)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb3;
```

## Inyecciones:

```
-- INYECCION DE DATOS EN TABLA CLIENTES
USE hoteles;
INSERT INTO cliente (dni, nombre, apellido, direccion, email, numeroTelefono)
VALUES
(12345678, 'Juan', 'Pérez', 'Calle Falsa 123', 'juan.perez@example.com', '111-2345'),
(23456789, 'Ana', 'García', 'Av. Libertador 456', 'ana.garcia@example.com', '111-6789'),
(34567890, 'Luis', 'Rodríguez', 'Calle Colón 789', 'luis.rodriguez@example.com', '111-1122'),
(45678901, 'María', 'López', 'Calle San Martín 321', 'maria.lopez@example.com', '111-3344'),
(56789012, 'Carlos', 'Martínez', 'Av. Rivadavia 654', 'carlos.martinez@example.com', '111-5566'),
(67890123, 'Laura', 'González', 'Calle Sarmiento 987', 'laura.gonzalez@example.com', '111-7788'),
(78901234, 'Pedro', 'Gómez', 'Av. Belgrano 159', 'pedro.gomez@example.com', '111-9900'),] Cesiaf,
(89012345, 'Sofía', 'Fernández', 'Calle Mitre 753', 'sofia.fernandez@example.com', '111-2233'),
(90123456, 'Diego', 'Ramírez', 'Calle Moreno 456', 'diego.ramirez@example.com', '111-4455'),
(12345067, 'Valeria', 'Sosa', 'Calle Güemes 789', 'valeria.sosa@example.com', '111-6677'),
(23456178, 'Jorge', 'Silva', 'Av. Mitre 963', 'jorge.silva@example.com', '111-8899'),
(34567289, 'Carmen', 'Torres', 'Calle Belgrano 123', 'carmen.torres@example.com', '111-1111'),
(45678390, 'Ricardo', 'Flores', 'Av. San Juan 456', 'ricardo.flores@example.com', '111-3333'),
(56789401, 'Paula', 'Ríos', 'Calle Corrientes 789', 'paula.rios@example.com', '111-5555'),
(67890512, 'Esteban', 'Díaz', 'Calle Callao 159', 'esteban.diaz@example.com', '111-7777'),
(78901623, 'Marta', 'Cabrera', 'Av. Santa Fe 753', 'marta.cabrera@example.com', '111-9999'),
(89012734, 'Roberto', 'Medina', 'Calle Lavalle 456', 'roberto.medina@example.com', '111-2222'),
(90123845, 'Isabel', 'Vega', 'Calle Córdoba 789', 'isabel.vega@example.com', '111-4444'),
(12345956, 'Daniel', 'Castro', 'Av. Las Heras 963', 'daniel.castro@example.com', '111-6666'),
(23456167, 'Carolina', 'Suárez', 'Calle Roca 123', 'carolina.suarez@example.com', '111-8888'),
(34567278, 'Gonzalo', 'Molina', 'Av. Alberdi 456', 'gonzalo.molina@example.com', '111-0000'),
(45678389, 'Lucía', 'Reyes', 'Calle Pueyrredón 789', 'lucia.reyes@example.com', '111-1212'),
(56789490, 'Federico', 'Aguilar', 'Av. Jujuy 159', 'federico.aguilar@example.com', '111-3434'),
(67890501, 'Elena', 'Ortiz', 'Calle Urquiza 753', 'elena.ortiz@example.com', '111-5656'),
(78901612, 'Alberto', 'Paz', 'Av. Salta 456', 'alberto.paz@example.com', '111-7878'),
(89012723, 'Gabriela', 'Campos', 'Calle Mendoza 789', 'gabriela.campos@example.com', '111-9090'),
(90123834, 'Francisco', 'Núñez', 'Calle Tucumán 963', 'francisco.nunez@example.com', '111-1414'),
(12345945, 'Adriana', 'Romero', 'Av. Entre Ríos 123', 'adriana.romero@example.com', '111-3636'),
(23456156, 'Matías', 'Vargas', 'Calle San Juan 456', 'matias.vargas@example.com', '111-5858'),
(34567267, 'Silvina', 'Herrera', 'Av. Libertador 789', 'silvina.herrera@example.com', '111-7070'),
(45678378, 'Mariano', 'Méndez', 'Calle Alem 159', 'mariano.mendez@example.com', '111-9292'),
(56789489, 'Alejandra', 'Morales', 'Calle Moreno 753', 'alejandra.morales@example.com', '111-1313'),
(67890500, 'Hernán', 'Figueroa', 'Av. San Lorenzo 456', 'hernan.figueroa@example.com', '111-3535'),
(78901611, 'Rosa', 'Navarro', 'Calle Callao 789', 'rosa.navarro@example.com', '111-5757'),
(89012722, 'Emanuel', 'Peña', 'Av. 9 de Julio 963', 'emanuel.pena@example.com', '111-7979'),
(90123833, 'Claudia', 'Bustos', 'Calle Belgrano 123', 'claudia.bustos@example.com', '111-0101'),
(12345944, 'Andrés', 'Cortés', 'Av. Corrientes 456', 'andres.cortes@example.com', '111-2323'),
(23456155, 'Liliana', 'Juárez', 'Calle Sarmiento 789', 'liliana.juarez@example.com', '111-4545'),
(34567266, 'Pablo', 'Leiva', 'Av. Mitre 159', 'pablo.leiva@example.com', '111-6767'),
(45678377, 'Victoria', 'Correa', 'Calle Moreno 753', 'victoria.correa@example.com', '111-8989'),
```

```
-- INYECCION DE DATOS EN TABLA HABITACIÓN
INSERT INTO habitacion (numeroHabitacion, idTipoHabitacion, idDisponibilidad) VALUES
(101, 1, 1), -- Suite, disponible
(102, 1, 2), -- Suite, reservada
(103, 2, 3), -- Doble Estándar, ocupada
(104, 2, 1), -- Doble Estándar, disponible
(105, 3, 4), -- Individual, mantenimiento
(106, 3, 2), -- Individual, reservada
(107, 4, 1), -- Apartamento, disponible
(108, 4, 3), -- Apartamento, ocupada
(109, 1, 2), -- Suite, reservada
(110, 1, 3), -- Suite, ocupada
(111, 2, 1), -- Doble Estándar, disponible
(112, 2, 4), -- Doble Estándar, mantenimiento
(113, 3, 1), -- Individual, disponible
(114, 3, 3), -- Individual, ocupada
(115, 4, 2), -- Apartamento, reservada
(116, 4, 1), -- Apartamento, disponible
(117, 1, 3), -- Suite, ocupada
(118, 2, 4), -- Doble Estándar, mantenimiento
(119, 3, 1), -- Individual, disponible
(120, 3, 2), -- Individual, reservada
(121, 4, 3), -- Apartamento, ocupada
(122, 1, 2), -- Suite, reservada
(123, 1, 1), -- Suite, disponible
(124, 2, 3), -- Doble Estándar, ocupada
(125, 2, 2), -- Doble Estándar, reservada
(126, 3, 1), -- Individual, disponible
(127, 4, 1), -- Apartamento, disponible
(128, 4, 2), -- Apartamento, reservada
(129, 1, 4), -- Suite, mantenimiento
(130, 1, 1), -- Suite, disponible
(131, 2, 3), -- Doble Estándar, ocupada
(132, 2, 1), -- Doble Estándar, disponible
(133, 3, 2), -- Individual, reservada
(134, 4, 3), -- Apartamento, ocupada
(135, 4, 2), -- Apartamento, reservada
(136, 1, 1), -- Suite, disponible
(137, 1, 3), -- Suite, ocupada
(138, 2, 4), -- Doble Estándar, mantenimiento
```

```
-- INYECCIÓN DE DATOS EN TABLA ROL
INSERT INTO rol (rol) VALUES
('Repcionista'),
('Gerente'),
('Personal de limpieza'),
('Mantenimiento'),
('Cocinero');

-- INYECCIÓN DE DATOS EN TABLA PERSONAL
INSERT INTO personal (nombre, apellido, horaEntrada, horaSalida, idRol) VALUES
('Juan', 'Pérez', '08:00:00', '16:00:00', 1),
('María', 'González', '09:00:00', '17:00:00', 1),
('Pedro', 'López', '08:00:00', '16:00:00', 2),
('Ana', 'Martínez', '10:00:00', '18:00:00', 3),
('Luis', 'Fernández', '07:00:00', '15:00:00', 4),
('Sofía', 'Ramírez', '11:00:00', '19:00:00', 5),
('Diego', 'Torres', '08:00:00', '16:00:00', 2),
('Claudia', 'Díaz', '09:00:00', '17:00:00', 3);

-- INYECCIÓN DE DATOS EN TABLA MANTENIMIENTO
INSERT INTO mantenimiento (idHabitacion, idPersonal, descripcionProblema, fechaReporte, fechaResolucion, idEstadoMantenimiento) VALUES
(10, 4, 'Reparar fuga de agua en el baño', '2024-10-01', '2024-10-05', 1),
(20, 4, 'Ajustar calefacción que no enciende', '2024-10-02', '2024-10-06', 2),
(30, 4, 'Reemplazar bombilla del plafón', '2024-10-03', '2024-10-07', 3),
(40, 4, 'Limpieza de aire acondicionado', '2024-10-01', '2024-10-05', 1),
(5, 3, 'Pintura de pared desgastada', '2024-10-04', '2024-10-08', 2),
(6, 2, 'Revisar cerradura de la puerta', '2024-10-05', '2024-10-09', 1),
(7, 4, 'Reparar grifo que gotea', '2024-10-01', '2024-10-05', 3),
(8, 4, 'Reemplazar colchón dañado', '2024-10-02', '2024-10-06', 1),
(9, 4, 'Revisar sistema eléctrico', '2024-10-03', '2024-10-07', 2),
(10, 3, 'Ajustar ventana que no cierra bien', '2024-10-04', '2024-10-08', 1),
(11, 4, 'Inspeccionar fuga en el aire acondicionado', '2024-10-02', NULL, 3),
(12, 4, 'Desinfectar moho en la pared', '2024-10-03', NULL, 1),
(13, 4, 'Reparar tapa del inodoro', '2024-10-05', NULL, 2),
(14, 3, 'Reemplazar cortinas rotas', '2024-10-01', NULL, 3),
(15, 4, 'Ajustar radiador que no calienta', '2024-10-02', NULL, 1),
(16, 4, 'Cambiar bombilla de la lámpara de lectura', '2024-10-03', NULL, 1),
(17, 2, 'Reparar silla rota', '2024-10-04', NULL, 3),
(18, 3, 'Desinfectar moho en el baño', '2024-10-01', NULL, 2),
(19, 4, 'Inspeccionar fugas en la ducha', '2024-10-02', NULL, 1),
(20, 2, 'Cambiar cerradura de la puerta principal', '2024-10-03', NULL, 3);
```

```
-- INYECCIÓN DE DATOS EN TABLA RESERVA
INSERT INTO reserva (idPersonal, idHabitacion, idCliente, fechaEntrada, fechaSalida) VALUES
(1, 1, 1, '2024-10-10', '2024-10-15'), -- Juan, habitación 101, cliente 1
(2, 2, 2, '2024-10-12', '2024-10-14'), -- María, habitación 102, cliente 2
(3, 3, 3, '2024-10-11', '2024-10-16'), -- Pedro, habitación 103, cliente 3
(4, 4, 1, '2024-10-05', '2024-10-09'), -- Ana, habitación 104, cliente 1
(1, 20, 4, '2024-10-07', '2024-10-10'), -- Juan, habitación 120, cliente 4
(5, 25, 2, '2024-10-09', '2024-10-13'), -- Sofía, habitación 125, cliente 2
(6, 32, 5, '2024-10-01', '2024-10-05'), -- Luis, habitación 132, cliente 5
(2, 35, 6, '2024-10-04', '2024-10-06'); -- María, habitación 135, cliente 6

-- INYECCIÓN DE DATOS EN TABLA SERVICIO
INSERT INTO servicio (nombreServicio, descripcion, precio) VALUES
('Desayuno', 'Desayuno buffet con opciones variadas', 15000),
('Almuerzo', 'Almuerzo a la carta en el restaurante', 25000),
('Cena', 'Cena gourmet con menú especial', 30000),
('Servicio de lavandería', 'Lavado y planchado de ropa', 10000),
('Servicio de limpieza', 'Limpieza diaria de la habitación', 8000);

-- INYECCIÓN DE DATOS EN TABLA SERVICIOCONSUMIDO
INSERT INTO servicioconsumido (idReserva, idServicio, cantidad)
VALUES
(1, 1, 2), -- Reserva 1, 2 Desayunos
(1, 2, 1), -- Reserva 1, 1 Almuerzo
(2, 3, 1), -- Reserva 2, 1 Cena
(3, 1, 1), -- Reserva 3, 1 Desayuno
(4, 4, 3), -- Reserva 4, 3 Lavandería
(5, 5, 1), -- Reserva 5, 1 Limpieza
(6, 1, 1), -- Reserva 6, 1 Desayuno
(7, 2, 2); -- Reserva 7, 2 Almuerzos
```

Consultas:

```
-- SELECT
-- Nombres y apellidos de clientes con reservas
SELECT c.nombre, c.apellido
FROM cliente c
JOIN reserva r ON c.id = r.idCliente;

-- todas las reservas que tienen fecha de entrada después del 10 de octubre de 2024
SELECT id
FROM reserva
WHERE fechaEntrada > '2024-10-10';

-- id, nombre y precio de todos los servicios
SELECT id, nombreServicio, precio
FROM servicio
ORDER BY id DESC;

-- Reservas con más de 2 noches de estancia
SELECT id, idCliente, fechaEntrada, fechaSalida
FROM reserva
WHERE DATEDIFF(fechaSalida, fechaEntrada) > 2;

-- Detalles de mantenimiento por estado "Pendiente":
SELECT m.id, m.idHabitacion, m.descripcionProblema, em.estado
FROM mantenimiento m
JOIN estadoMantenimiento em ON m.idEstadoMantenimiento = em.id
WHERE em.estado = 'Pendiente'; You, la semana pasada • Agg de consultas

-- Nombre del personal encargado de las reservas de un cliente específico
SELECT p.nombre
FROM personal p
JOIN reserva r ON p.id = r.idPersonal
WHERE r.idCliente = 1;

-- INSERT

-- Nuevo cliente
INSERT INTO cliente (dni, nombre, apellido, direccion, email, numeroTelefono)
VALUES (12345678, 'Carlos', 'Mendoza', 'Calle Ejemplo 123', 'carlos.mendoza@example.com', '555-1234');

-- Nuevo personal
INSERT INTO personal (nombre, apellido, horaEntrada, horaSalida, idRol)
VALUES ('Laura', 'Sánchez', '08:00:00', '16:00:00', 1);
```

```

-- Nuevo servicio
INSERT INTO servicio (nombreServicio, descripcion, precio)
VALUES ('Masaje Relajante', 'Masaje de 60 minutos para relajación', 10000);

-- Inserciones múltiples de clientes
INSERT INTO cliente (dni, nombre, apellido, direccion, email, numeroTelefono)
VALUES
 (23456789, 'Ana', 'García', 'Calle Falsa 456', 'ana.garcia@example.com', '555-5678'),
 (34567890, 'Luis', 'Rodríguez', 'Calle Colón 789', 'luis.rodriguez@example.com', '555-9012'),
 (45678901, 'María', 'López', 'Calle San Martín 321', 'maria.lopez@example.com', '555-3456');

-- UPDATE

-- Actualizar dirección de clientes por DNI
UPDATE cliente
SET
 direccion = CASE
 WHEN dni = 12345678 THEN 'Calle Nueva 456'
 WHEN dni = 23456789 THEN 'Calle Vieja 789'
 ELSE direccion
 END
WHERE dni IN (12345678, 23456789);

-- Cambiar disponibilidad de habitaciones tipo Suite
UPDATE habitacion
SET idDisponibilidad = 2 -- Reservada
WHERE idTipoHabitacion = 1 AND idDisponibilidad = 1;

-- Aumentar precio de servicios consumidos más de dos veces
UPDATE servicio s
SET s.precio = s.precio * 1.10
WHERE EXISTS (
 SELECT 1
 FROM servicioConsumido sc
 WHERE sc.idServicio = s.id AND sc.cantidad > 2
);

```

```
-- Eliminar un cliente por ID
DELETE FROM cliente WHERE id = 33;

-- Eliminar servicios con IDs específicos
DELETE FROM servicio WHERE id IN (1, 2, 3);

-- Eliminar habitaciones ocupadas de tipo Suite
DELETE FROM habitacion
WHERE idDisponibilidad = 3 AND idTipoHabitacion = 1;

-- Eliminar servicios consumidos sin reservas asociadas
DELETE FROM servicioConsumido
WHERE idReserva NOT IN (SELECT id FROM reserva);

-- Eliminar todos los registros de mantenimiento
DELETE FROM mantenimiento;

-- Eliminar reservas de un cliente específico por DNI
DELETE r
FROM reserva r
JOIN cliente c ON r.idCliente = c.id
WHERE c.dni = 12345678;

-- JOIN

-- Obtener todas las reservas con detalles del cliente
SELECT
 r.id AS idReserva,
 c.nombre AS nombreCliente,
 c.apellido AS apellidoCliente,
 r.fechaEntrada,
 r.fechaSalida
FROM reserva r
JOIN cliente c ON r.idCliente = c.id;

-- Obtener servicios consumidos junto con la reserva
SELECT
 sc.idReserva,
 s.nombreServicio,
 sc.cantidad
FROM servicioConsumido sc
JOIN servicio s ON sc.idServicio = s.id
JOIN reserva r ON sc.idReserva = r.id;
```

```
-- Obtener habitaciones ocupadas con información del cliente
SELECT
 h.numeroHabitacion,
 c.nombre AS nombreCliente,
 c.apellido AS apellidoCliente,
 r.fechaEntrada,
 r.fechaSalida
FROM habitacion h
JOIN reserva r ON h.id = r.idHabitacion
JOIN cliente c ON r.idCliente = c.id
WHERE h.idDisponibilidad = 3;
```

## Conclusiones

El desarrollo del Sistema de Gestión Hotelera Automatizado ha sido un proceso enriquecedor que no solo ha permitido la aplicación práctica de conocimientos en programación y bases de datos, sino que también ha contribuido al fortalecimiento de competencias técnicas y profesionales en el equipo de trabajo. A lo largo de las distintas etapas del proyecto, desde el análisis de requerimientos hasta la implementación y pruebas del sistema, se ha evidenciado la importancia de una planificación meticulosa y una metodología ágil que favorece la adaptabilidad y el aprendizaje continuo.

La automatización de los procesos de gestión hotelera representa una solución eficaz ante los desafíos que enfrenta la industria, como la mejora de la eficiencia operativa y la reducción de errores en la administración de reservas y clientes. La implementación de un sistema robusto y seguro en Python, junto con el uso de MySQL para la gestión de datos, no solo optimiza el flujo de trabajo en un entorno hotelero, sino que también proporciona a los empleados las herramientas necesarias para realizar su labor con mayor agilidad y precisión.

Además, este proyecto destaca la relevancia de la formación práctica en el ámbito académico, preparando a los estudiantes para enfrentar los desafíos del mercado laboral actual. La experiencia adquirida en el desarrollo del sistema, la colaboración en equipo y el manejo de herramientas modernas de programación y gestión de bases de datos son aspectos que sin duda enriquecerán el perfil profesional de cada uno de los integrantes del grupo.

En resumen, el Sistema de Gestión Hotelera Automatizado no solo responde a una necesidad real, sino que también representa un proceso formativo, evidenciando cómo la teoría y la práctica pueden integrarse de manera efectiva para generar soluciones innovadoras.