

INGENIERÍA DE SISTEMAS DE INFORMACIÓN  
 CURSO: PROGRAMACIÓN ORIENTADO A OBJETOS  
 SECCIÓN: 205   
DOCENTE: Juan Alfonso Ramírez Espinoza  
 **2025-10**  
  
**TRABAJO FINAL**  
 **SISTEMA DE GESTIÓN DE PEDIDOS PARA RESTAURANTES**

INTEGRANTES:

* Hidalgo Martel, Joseph Edward (U202421665)
* Huamán Flores, Alexis Miguel (U20241G114)
* Peña Roña, Antony Yomar (U202421102)
* Villavicencio Dávila, Ivette Lucero (U20241G010)

**Resumen del proyecto**

El proyecto trabajado es un sistema integral para la gestión eficiente de restaurante, orientado a optimizar tiempos y procesos en la gestión de registros y evaluación de mesas, mozos y pedidos. El sistema permitirá registrar y evaluar mesas, a partir de datos como número, zona, capacidad y estado. Asimismo, gestionara los mozos desde su registro y el cambio de mesa si uno no está disponible, validando que no superen su capacidad máxima de atención. En el caso de los pedidos, los clientes podrán seleccionar entre platos, postres y bebidas, registrándose la hora de inicio y entrega de cada uno. Además, el sistema permitirá la generación de reportes estadísticos relevantes, como los mozos con mayores pedidos atendidos, las mesas con mayor y menor actividad, el tiempo promedio de espera, los ingresos obtenidos por cada zona del restaurante y otros reportes adicionales. De esta manera, se mejorará la forma de tomar decisiones y una gestión más eficiente del establecimiento.

**Objetivo**

Como se mencionó anteriormente, el objetivo del proyecto es desarrollar e implementar un sistema digital de gestión para restaurantes que permita una atención y eficiencia óptima, mediante el registro y control digital de mesas, mozos y pedidos. Además, implementa la generación de reportes estadísticos que faciliten la obtención de información que ayuden a mejorar continuamente el servicio y la calidad de atención al cliente. (Instituto Le Cordon Bleu, s.f.)

**Alcance**

Este programa está diseñado específicamente para restaurantes en crecimiento que buscan organizar la gestión de sus recursos y mejorar la eficiencia operativa del negocio. Su uso puede estar orientado tanto para el personal operativo como los administradores del establecimiento, brindando herramientas de análisis que permitan la toma de decisiones relacionadas con un control correcto y eficiente de los recursos.

**INDICE**

1. **Objetivo del Estudiante** (Student Outcome)
2. **Capítulo 1:** Presentación y sustentación del problema a resolver

* Tema del trabajo
* Sustento teórico que el tema
* Motivación por la que escogió el tema

1. **Capítulo 2**: Estructuras de archivo

* Análisis de datos
* Plan de actividades

1. **Capítulo 3:** Listado de funcionalidades cumplidas y Manual de Usuario

* Tabla las funcionalidades
* Capturas de pantallas

1. **Conclusiones**
2. **Recomendaciones**
3. **Glosario de Términos**
4. **Bibliografía**
5. **Participación**

**OBJETIVOS DEL ESTUDIANTE**   
**(Student Outcome)**

**Student Outcome ABET – EAC – 2:**

* **Alexis Huamán Flores:**

Durante el desarrollo del proyecto de gestión de pedidos para restaurantes, apliqué conocimientos de programación orientada a objetos, lo que me permitió utilizar clases y sus propiedades para organizar y manejar de una manera eficiente los datos de la empresa. Asimismo, empleé las bibliotecas NumPy y Pandas para gestionar estructuras de datos de doble entrada de manera óptima. El objetivo principal fue ofrecer al restaurante una aplicación que automatice procesos y minimice errores humanos. Además, este proyecto contribuirá a la optimización de los recursos económicos de la empresa, al reducir el personal necesario para tareas repetitivas, mejorando así el trabajo en equipo gracias a la automatización funciones.

**CAPITULO 1**

**Presentación y sustentación del**   
**problema a resolver**

**A. Tema del trabajo:**

* Desarrollo de un sistema digital integral para la gestión operativa y administración de recursos en restaurantes

**B. Sustento teórico del tema:**

* Este proyecto se basa en el uso de un sistema digital para gestionar de manera más eficiente las operaciones de un restaurante, como la asignación de mesas, la atención de los mozos y el seguimiento de los pedidos. Al automatizar estas tareas, se mejora la organización y se reduce el error humano, lo que permite un servicio más rápido y preciso. Además, el programa genera reportes para ayudar a los encargados a tomar decisiones basadas en datos. Para realizar este proyecto utilizaremos el lenguaje Python, Programación Orientada Objetos, el cual nos permite representar las entidades del restaurante mediante clases como mesas, mozos, pedidos y carta, cada uno de estás cuentas con atributos y métodos que nos ayudarán con la organización, comprensibilidad y adaptabilidad del código frente a nuevas necesidades en la operación del código (Castro, 2018).

**C. Motivación por la que escogió el tema:**

* Nos motivó a desarrollar este proyecto la falta de automatización y eficiencia en la administración que aún se mantiene en muchos establecimientos de comida, ya sea debido al desconocimiento de estos programas o a la falta de recursos económicos. Nuestro objetivo es optimizar y mejorar el sistema de gestión de estos restaurantes, permitiendo un ahorro significativo de tiempo en tareas de gestión y administración del negocio. Así mismo, buscamos satisfacer tanto las necesidades operativas como la experiencia de los clientes.

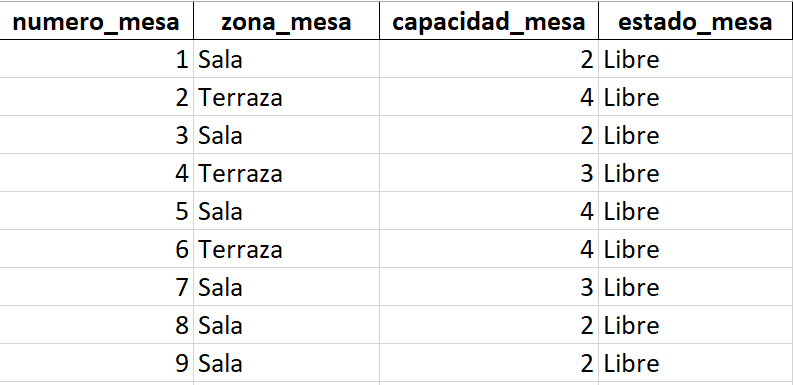
**CAPITULO 2**  
**Estructuras de archivos**

1. **Estructura del archivo Excel**

**Mesas:**

**Encabezados:**

* numero\_mesa
* zona\_mesa
* capacidad\_mesa
* estado\_mesa



**Mozos:**

**Encabezados:**

* id\_mozo
* nombre
* telefono
* estado
* mesas\_reservadas



**Carta:**

**Encabezados:**

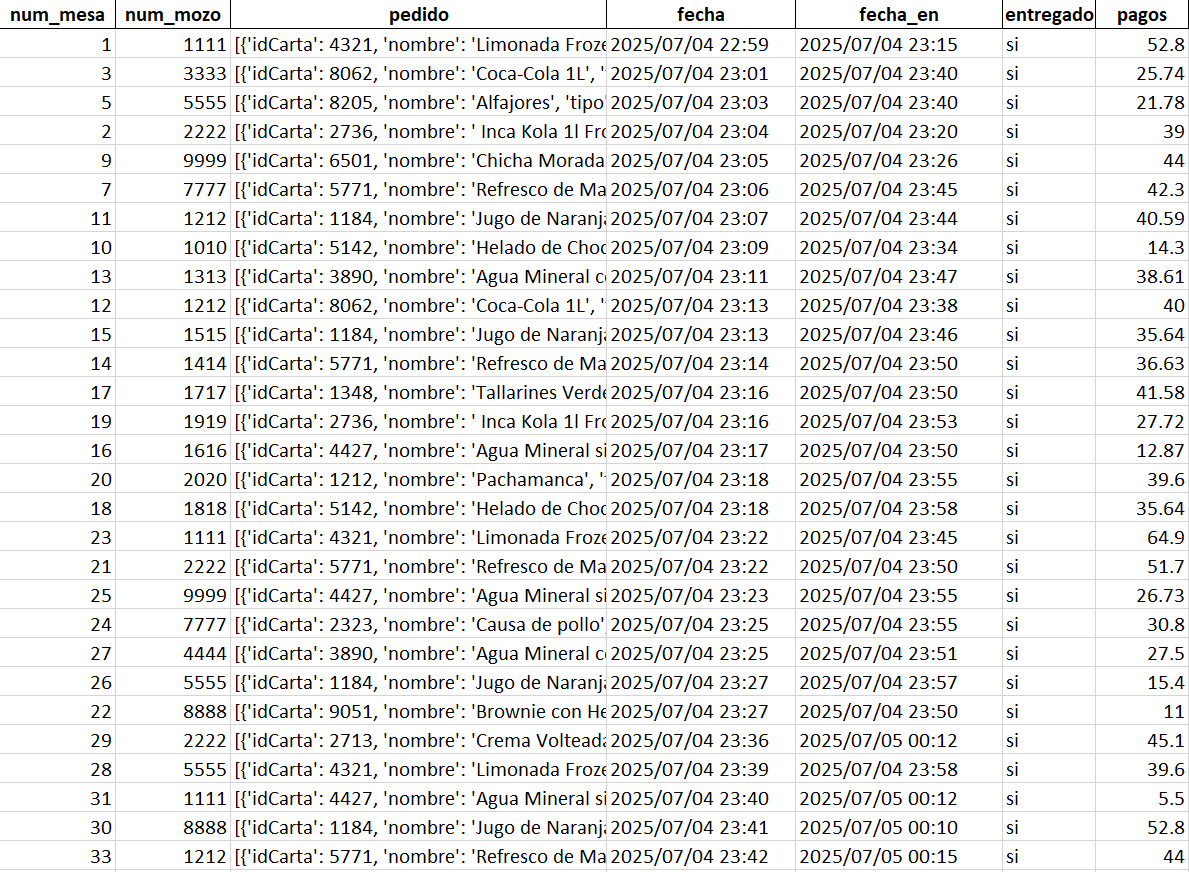
* idCarta
* nombre
* tipo
* precio



**Pedidos:**

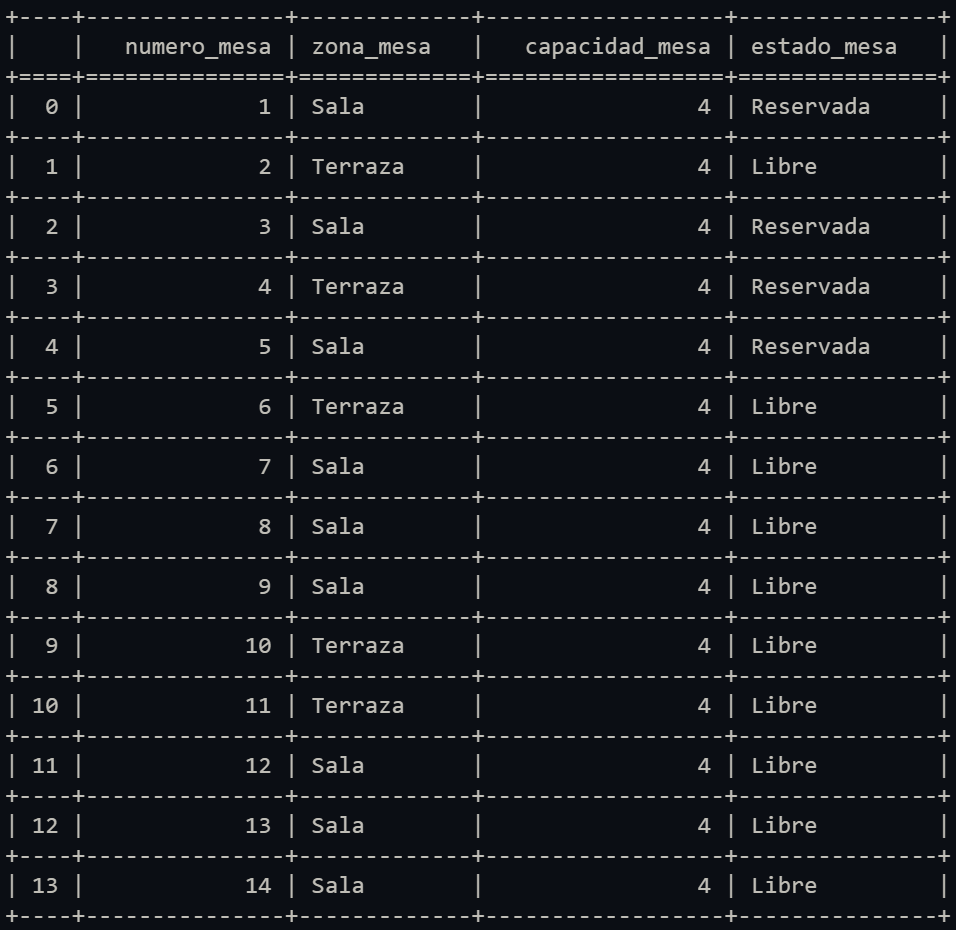
**Encabezados:**

* num\_mesa
* num\_mozo
* pedido
* fecha
* fecha\_en
* entregado
* pagos

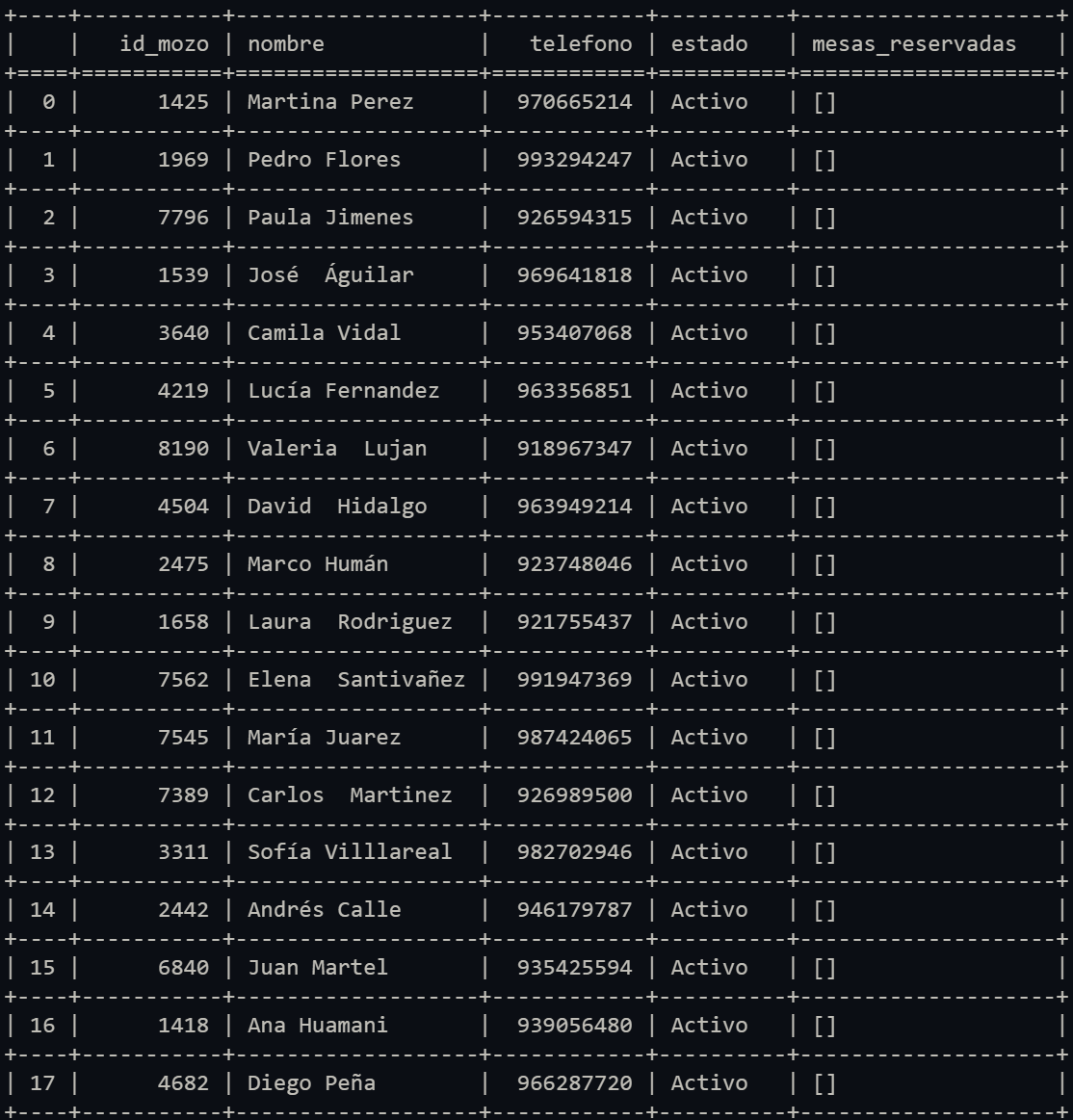


**Diseño de los resultados (Reportes y Gráficos)**

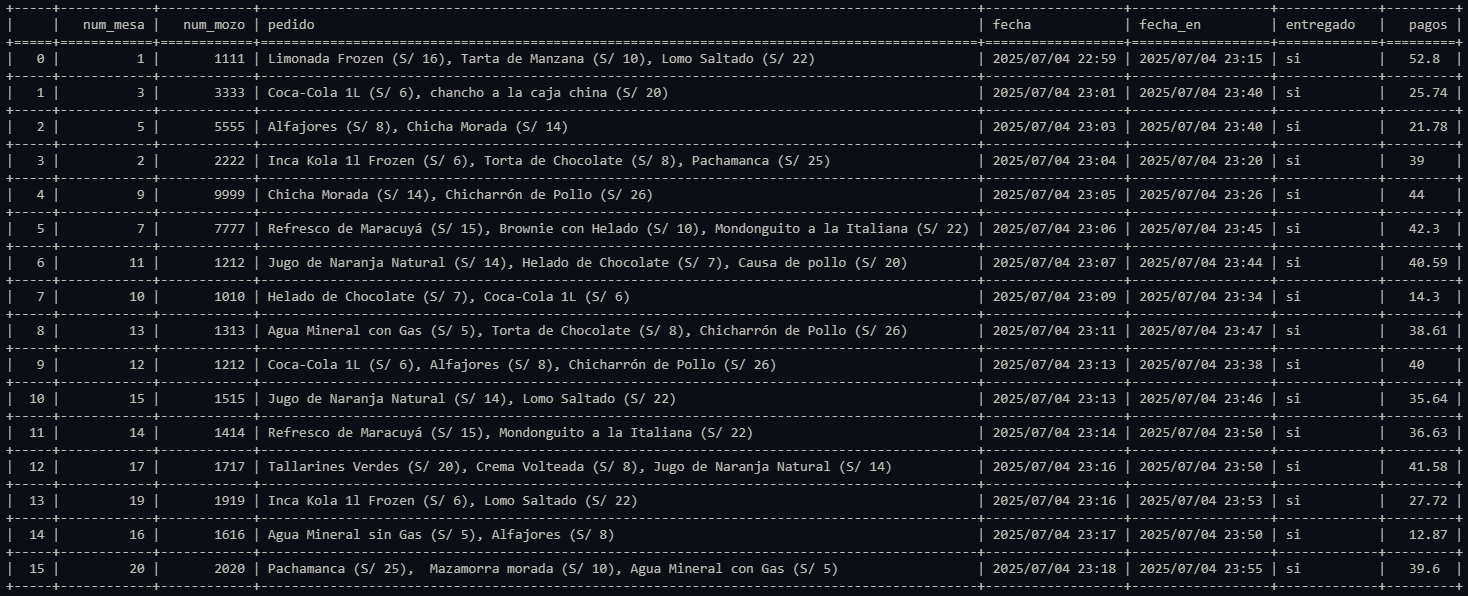
**Reporte de mesas:**



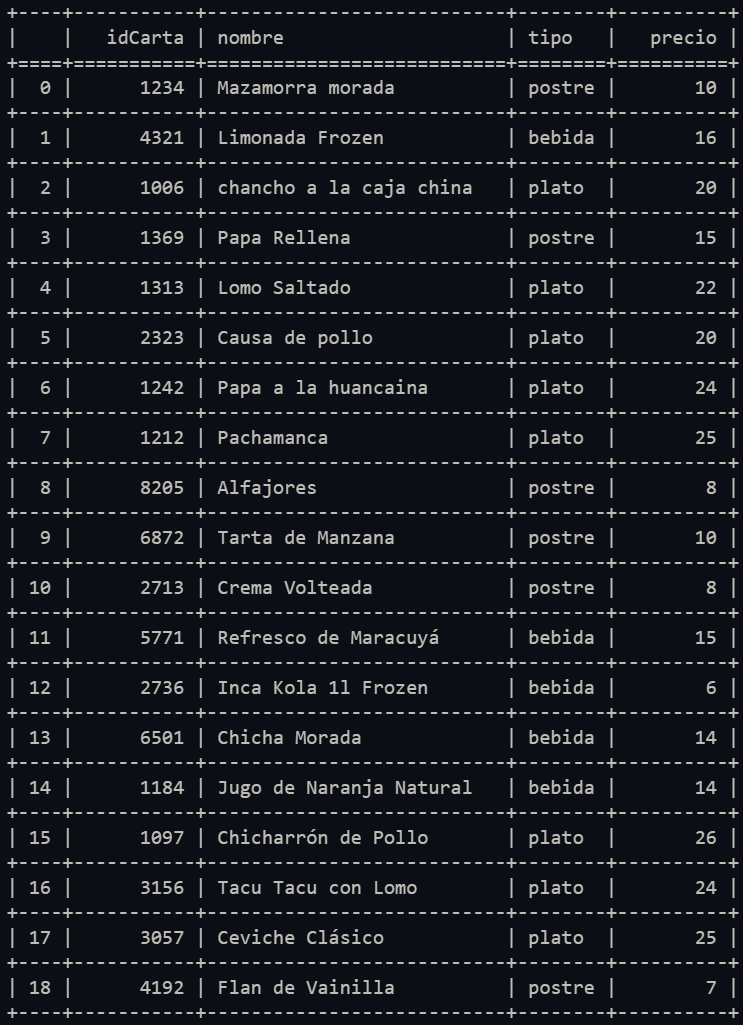
**Reporte de mozos**



**Reporte de pedidos**

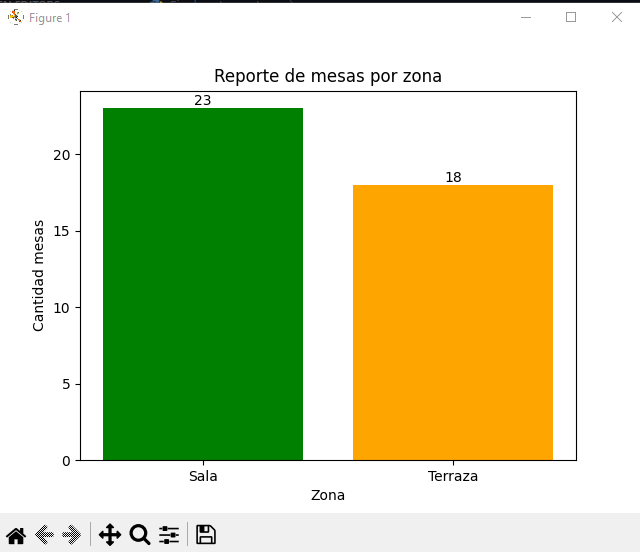


**CARTA:**

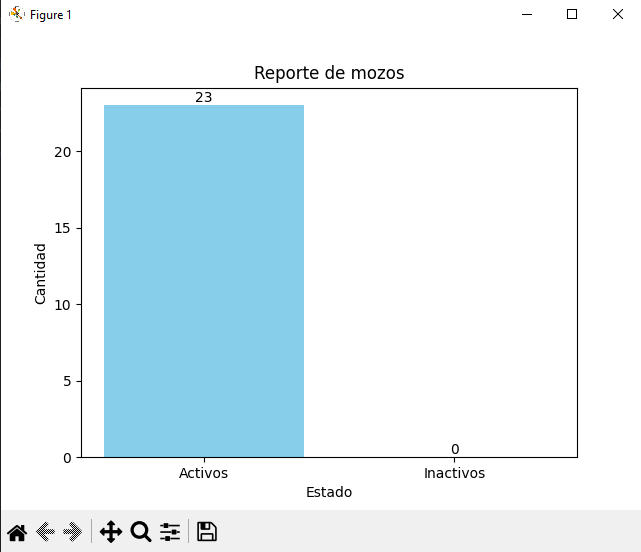


**GRÁFICOS**

* Gráfico de reportes de mesas registradas por zona:



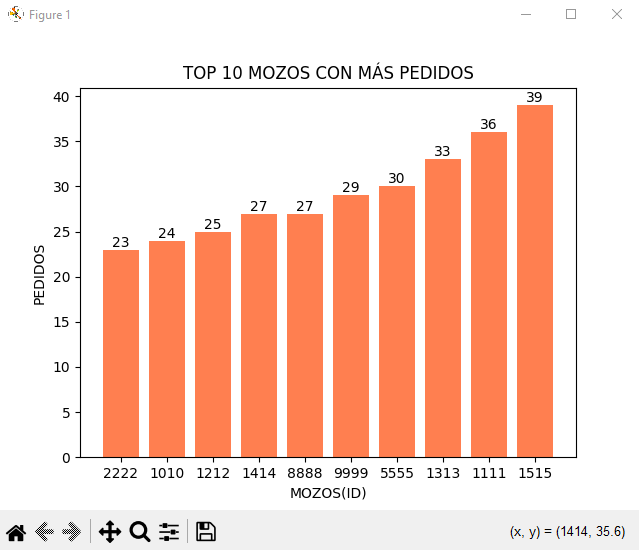
* Gráfico de reportes de mozos registrados y su estado:



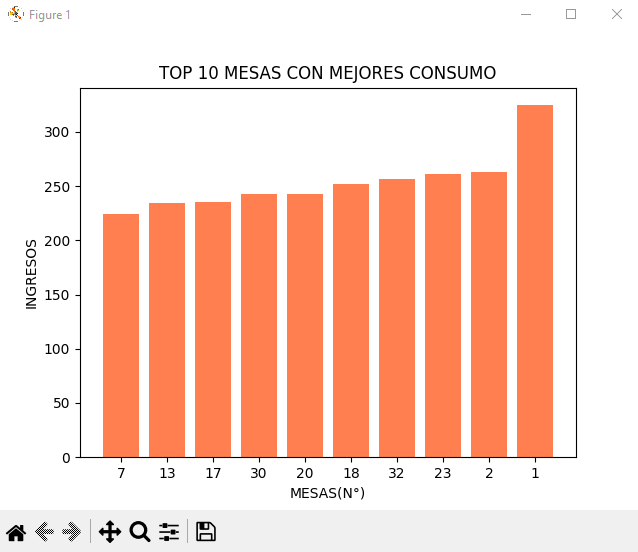
* Gráfico de reporte de pedidos registrados



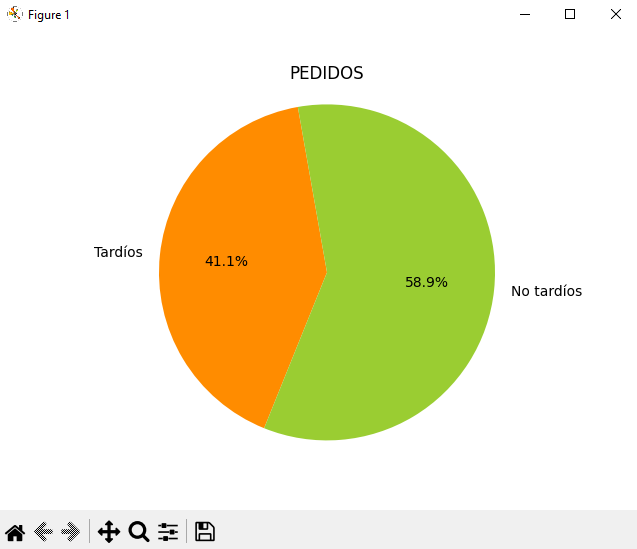
* Gráfico de reporte los 10 mozos con más pedidos



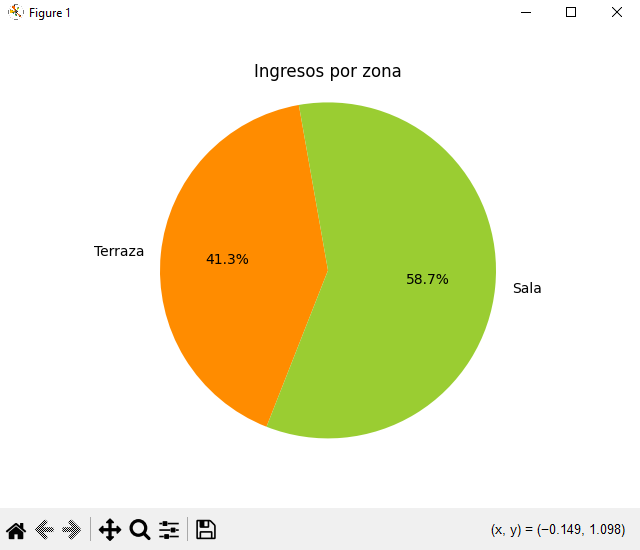
* Gráfico de reporte las 10 mesas con más pedidos



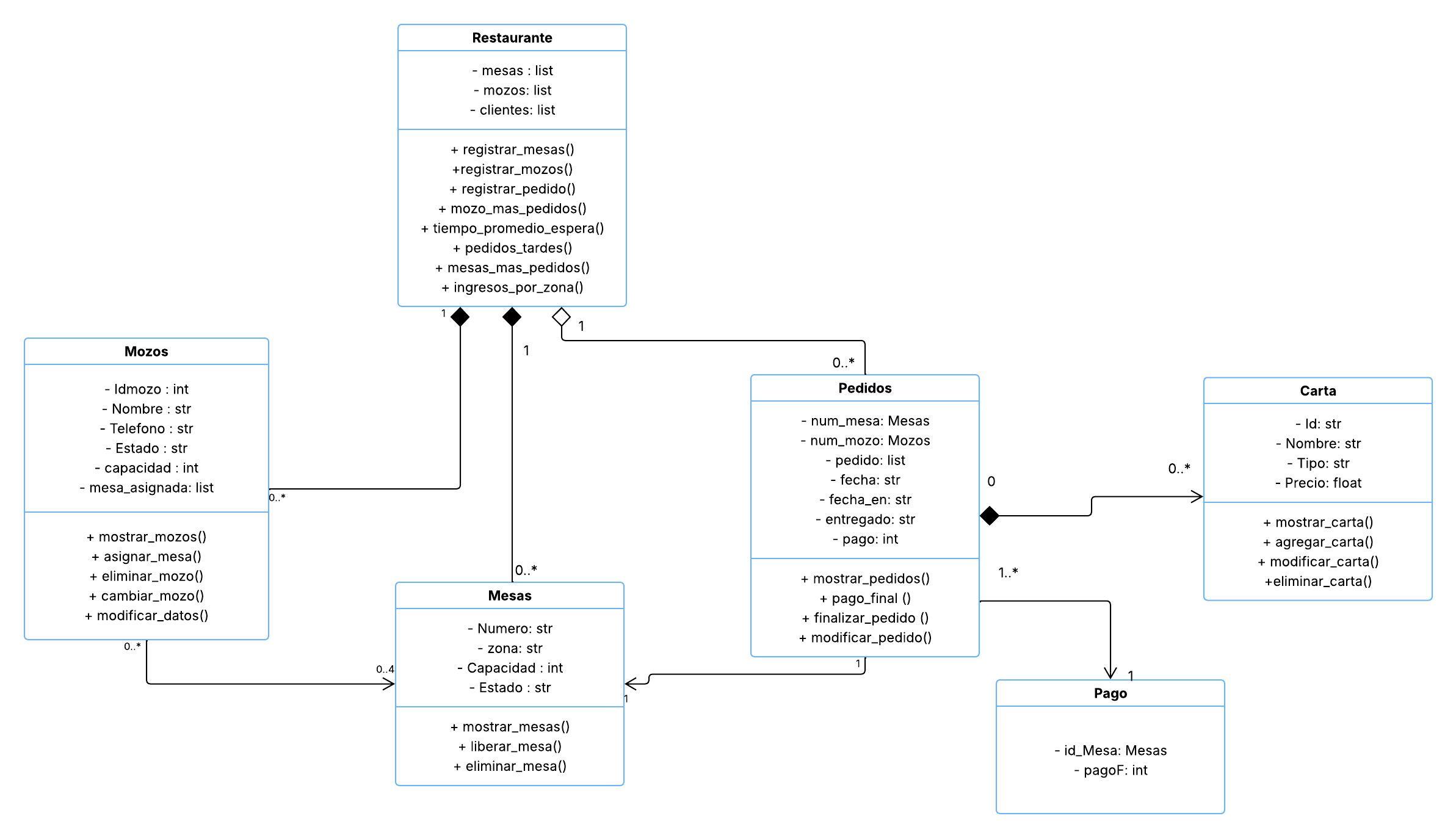
* Gráfico de reportes pedidos tardíos y no tardíos



* Gráfico de reportes ingresos por zona (sala / terraza)



**B. DIAGRAMA DE CLASES**



**C. Plan de Actividad:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Código** | **Alumno** | **Actividad** | **Fecha de entrega** |
| **U202421102** | Peña Roña, Antony Yomar | * Gestión de Mesas y Mozos * Registrar mesas con zona (sala/terraza). * Asignar mozos con límite de 4 mesas por mozo (regla de negocio). * Añadir/eliminar mesas y mozos * Actualizar datos del mozo | **28/06/2025** |
| **U20241G114** | Huamán Flores, Alexis Miguel | **Tomar pedidos por mesa**   * Añadir/eliminar pedidos * Registrar hora de inicio y hora de final * 1 reporte adicional | **02/07/2025** |
| **U20241G010** | Villavicencio Dávila, Ivette Lucero | **Cálculo de pago**   * Descuento automático del 10% si el tiempo de espera supera 30 minutos. * Propina sugerida del 10% del total. * 1 reporte adicional | **05/07/2025** |
| **U202421665** | Hidalgo Martel, Joseph Edward | **Mostrar reportes básicos**   * Mesa con Mayor Consumo (ventas totales por mesa). * Mozo con Más Pedidos Atendidos (eficiencia del personal). * Tiempo Promedio de Espera por Pedido (optimización de cocina). * Ingresos por Zona (sala vs. terraza) (análisis de demanda). * +2 reportes * Clase Carta | **05/07/2025** |
| **Observaciones:** | | | |
| * **Ninguna** | | | |

**B. Funcionalidades a cargo de cada estudiante:**

|  |  |
| --- | --- |
| Nombre | Responsabilidad |
| Antony Yomar Peña Roña | Gestión de mesas y mozos |
| Alexis Miguel Huamán Flores | Tomar pedidos por mesa |
| Ivette Lucero Villavicencio Dávila | Cálculo de pago |
| Joseph Edward Hidalgo Martel | Mostrar reportes básicos |

**CAPITULO 3**  
**Listado de funcionalidades cumplidas y Manual de Usuario**

**A. Tabla de funcionalidades:**

|  |  |  |
| --- | --- | --- |
| **#FR** | **FUNCIÓN** | **Cumplido** |
| **FR1** | El sistema debe presentar una portada con el logo de la empresa y los datos del proyecto. | **Se cumplió totalmente** |
| **FR2** | El sistema debe presentar un menú interactivo hacia todas las opciones del programa (registros y reportes). | **Se cumplió totalmente** |
| **FR3** | El sistema registra correctamente mesas y mozos. | **Se cumplió totalmente** |
| **FR4** | El sistema valida el ingreso de datos para el mozo y la mesa:   * Mesas: numero, estado, capacidad, zona * Mozos: numero, estado, nombre, teléfono, capacidad | **Se cumplió totalmente** |
| **FR5** | El sistema verifica que la capacidad máxima de atención del mozo sea de 4 mesas. | **Se cumplió totalmente** |
| **FR6** | El sistema muestra un reporte de mesas y mozos registrados con sus respectivas mesas asignadas. | **Se cumplió totalmente** |
| **FR7** | El sistema permite cambiar el mozo actual por uno nuevo respecto a una mesa asignada en caso de incidencias. | **Se cumplió totalmente** |
| **FR8** | El sistema valida el estado de las mesas para cuando esta una sea reservada su estado cambie automáticamente. | **Se cumplió**  **totalmente** |
| **FR9** | El sistema permite liberar mesas asignadas, eliminar mozos y mesas registrados. | **Se cumplió**  **totalmente** |
| **FR10** | El sistema permite modificar datos del mozo registrado | **Se cumplió**  **totalmente** |
| **FR11** | El sistema muestra correctamente el menú de platos, postres y bebidas. | **Se cumplió totalmente** |
| **FR12** | El sistema valida que la opción ingresada por el usuario sea correcta con respecto a los pedidos disponibles. | **Se cumplió totalmente** |
| **FR13** | El sistema permite la modificación y eliminación de algún pedido realizado por el cliente. | **Se cumplió totalmente** |
| **FR14** | El sistema valida que, para modificar el pedido, este no debió ya haber sido entregado ni pagado. | **Se cumplió totalmente** |
| **FR15** | El sistema muestra un reporte de los clientes y sus pedidos. | **Se cumplió totalmente** |
| **FR16** | El sistema permite finalizar un pedido al solicitar la fecha en la que se entregó el pedido y validar que esta sea una hora correcta. | **Se cumplió totalmente** |
| **FR17** | El sistema permite calcular el pago total de cada mesa. | **Se cumplió totalmente** |
| **FR18** | El sistema permite realizar un descuento automático dependiendo del tiempo de espera de un pedido. | **Se cumplió totalmente** |
| **FR19** | El sistema realiza pregunta al cliente si desea abonar propina, 10%, del pago total. | **Se cumplió totalmente** |
| **FR20** | El sistema permite el ingreso del número de mesa a querer cancelar. | **Se cumplió totalmente** |
| **FR21** | El sistema valida que el número de mesa ingresado este anteriormente registrado y exista. | **Se cumplió totalmente** |
| **FR22** | El sistema compara el tiempo de entrega de pedido con el tiempo de espera máximo. | **Se cumplió totalmente** |
| **FR23** | El sistema ejecuta bien todos los reportes sin ninguna falla. | **Se cumplió totalmente** |
| **FR24** | El sistema debe mostrar un reporte de la mesa con mayor consumo (ventas totales por mesa). | **Se cumplió totalmente** |
| **FR25** | El sistema debe generar un reporte del mozo con más pedidos atendidos (eficiencia del personal). | **Se cumplió totalmente** |
| **FR26** | El sistema debe mostrar el tiempo promedio de espera de todos los pedidos (optimización de cocina). | **Se cumplió totalmente** |
| **FR27** | El sistema debe mostrar un reporte de si el pedido llegó tarde o llegó a tiempo. | **Se cumplió totalmente** |
| **FR28** | El sistema debe presentar un reporte de ingresos por zona (sala vs. terraza) con un análisis de demanda. | **Se cumplió totalmente** |
|  | Referencia : Van Rossum, et al., (2025) |  |
|  |  |  |

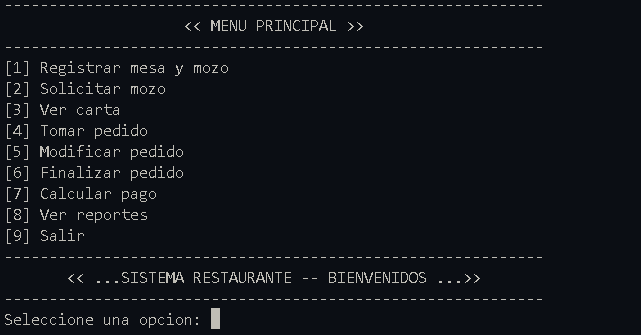
**B. Capturas de pantallas:**

**Caratula inicial:**



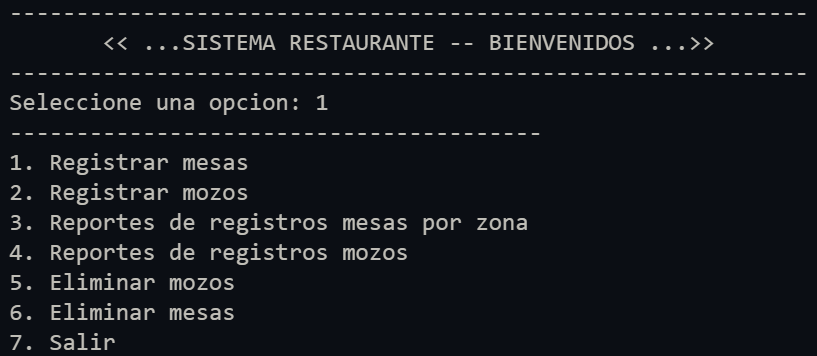
Caratula inicial donde se muestra información específica del curso, docente e integrantes a cargo del proyecto.

**Menú Principal:**



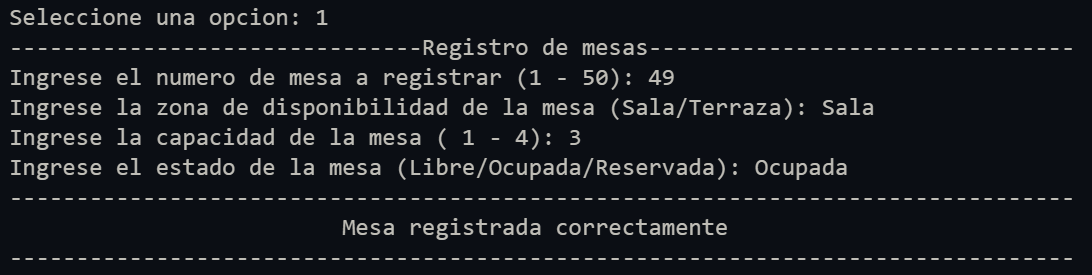
Menú principal en la que se muestran las opciones generales del sistema.

**1. REGISTRAR MESA Y MOZO**



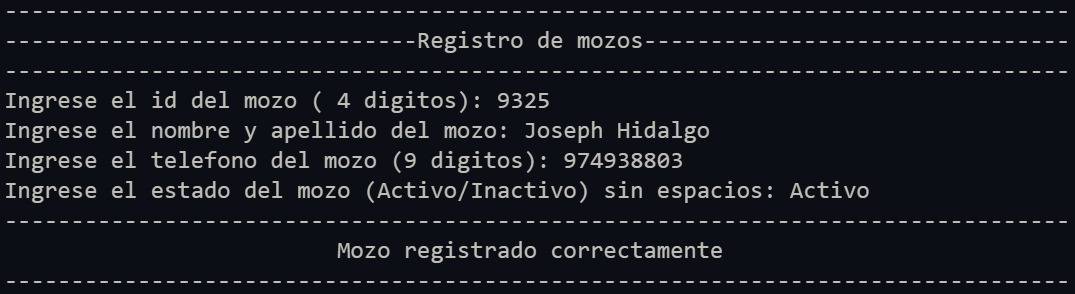
El usuario al seleccionar la **opción 1** del menú principal **“Registrar mesas y mozos”** será dirigido a un submenú de registros y visualización de los datos actuales, además podrá eliminar las mesas y mozos guardados en el archivo base.

**Registrar mesas:**



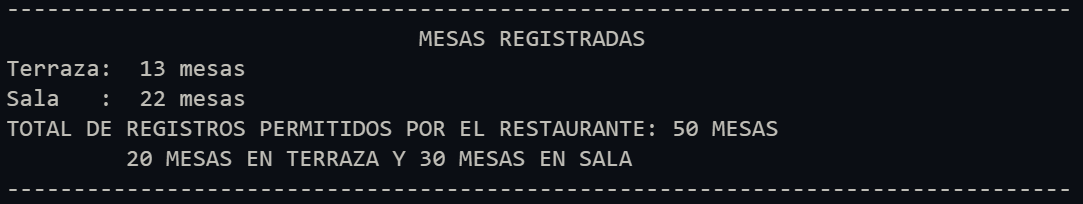
El usuario al escoger la **opción 1 “Registrar mesas”** el sistema le pedirá los datos los necesarios de la mesa tales como número, zona de disponibilidad, capacidad y su estado respectivo.

**Registrar mozos:**



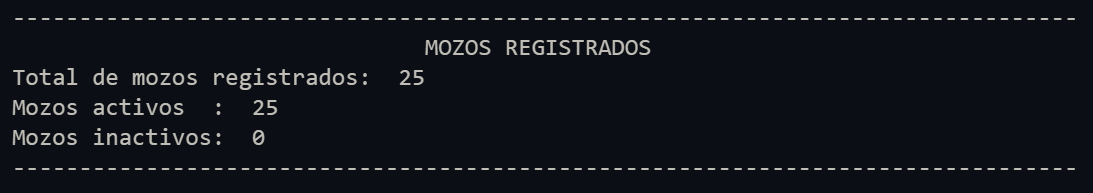
Si el usuario ingresa la **opción 2 “Registrar mozos”** el sistema les pedirá nuevamente los datos necesarios para el registro del mozo, entre ellos **id** de 4 dígitos numéricos, **nombre y apellido**, **teléfono** y su **estado** actual.

**Reportes de registros mesas por zona:**



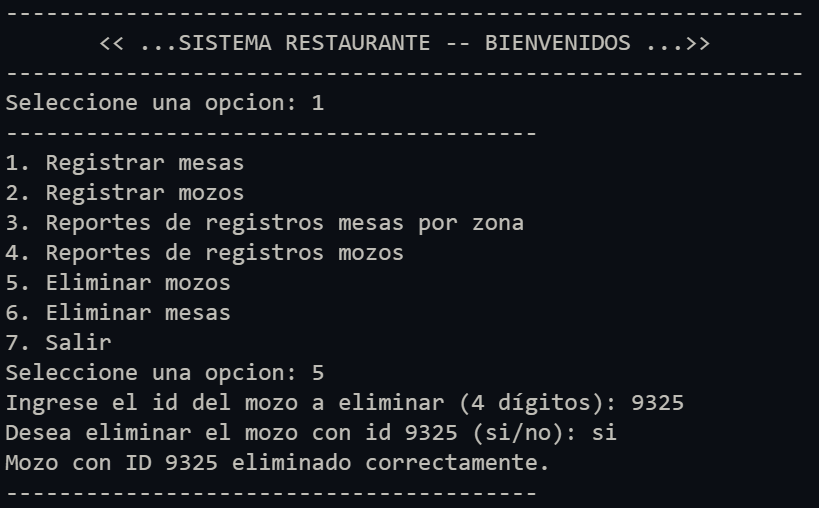
Al seleccionar la **opción 3**, **“Reporte de registros de mesas por zona”**, se mostrará en pantalla la cantidad de mesas registradas hasta ese momento, diferenciadas por su zona de disponibilidad en el restaurante (sala y terraza). Además, se indicará la capacidad máxima de registros permitidos tanto por zona como de forma general.

**Reportes de registros mozos:**



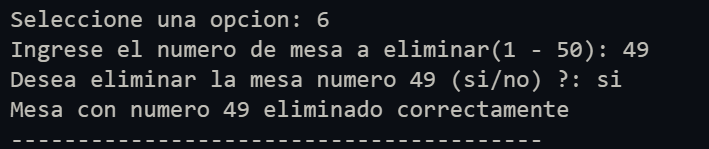
Si el usuario selecciona la **opción 4**, **“Reporte de registros de mozos”**, el programa mostrará en pantalla la cantidad total de mozos registrados. Adicionalmente, se detallará la cantidad de mozos según su estado (activos e inactivos).

**Eliminar mozos:**



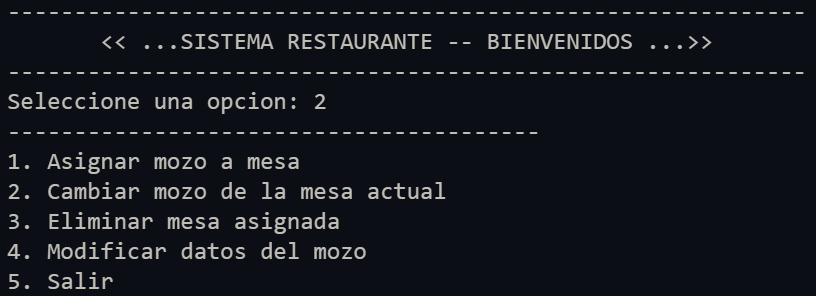
Si se selecciona la **opción 5**, **“Eliminar mozos”**, el sistema solicitará el **ID** del mozo que se desea eliminar y, posteriormente, pedirá una confirmación **(si/no)** para finalizar el proceso. Al concluir, mostrará un mensaje de la acción realizada. Cabe resaltar que esta operación elimina completamente del archivo base los registros para el mozo indicado.

**Eliminar mesas:**



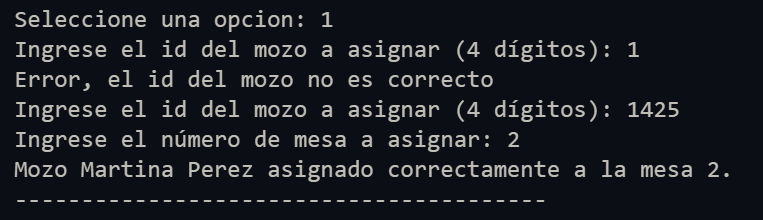
Al elegir la **opción 6**, **“Eliminar mesas”**, el sistema solicitará el número de la mesa que se desea eliminar y, a continuación, pedirá una confirmación para continuar con el proceso. Una vez finalizado, mostrará un mensaje indicando que la operación se completó exitosamente.

**2. SOLICITAR MOZO**



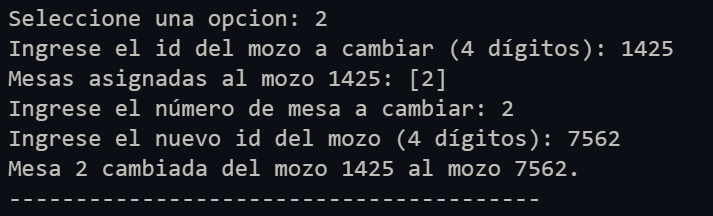
Si el usuario selecciona la **opción 2** del menú principal, **“Solicitar mozo”**, el sistema le solicitará los datos necesarios para completar este proceso.

**Asignar mozo a mesa:**



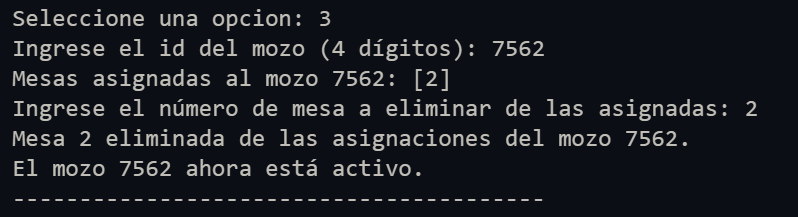
Si el usuario selecciona la **opción 1** del submenú, **“Asignar mozo a mesa”**, el sistema pedirá el ID del mozo, el cual debe estar previamente registrado en la base de datos y contar con un estado **“Activo”** para poder ser asignado. A continuación, solicitará el número de la mesa correspondiente la cual también debe estar registrada y contar con estado **“Libre”**. Al finalizar, se mostrará un mensaje indicando que el proceso se completó exitosamente.

**Cambiar mozo de la mesa actual:**



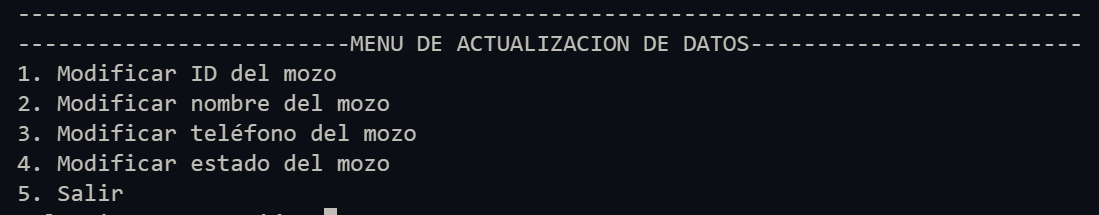
Al seleccionar la **opción 2 “Cambiar mozo de la mesa actual”,** el usuario podrá cambiar el mozo asignado a una mesa. Para ello, el sistema solicitará el ID del mozo actual, el número de la mesa que se desea cambiar y el ID del nuevo mozo. De este modo, se realizará la reasignación de la mesa a otro mozo disponible.

**Eliminar mesa asignada:**



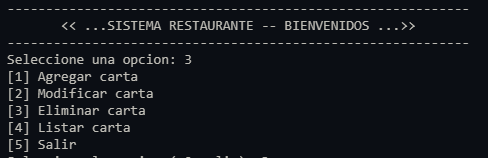
Al seleccionar la **opción 3**, **“cambiar mozo de la mesa actual “**, se le pedirá al usuario los datos necesarios (ID del mozo, N° mesa) para proceder con la eliminación de una mesa asignada.

**Modificar datos del mozo:**



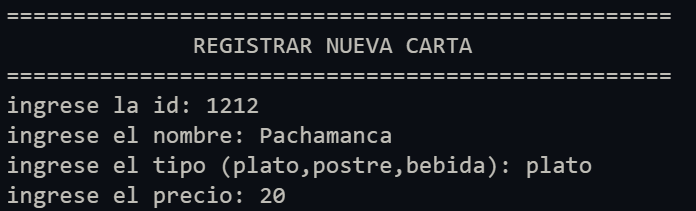
Al seleccionar la **opción 4**, **“Modificar datos del mozo”**, se mostrará en pantalla un menú con las opciones disponibles para actualizar la información del mozo. El usuario podrá elegir qué dato desea modificar e ingresará el nuevo valor para reemplazar el existente.

**3.VER CARTA**



Si el usuario elige la **opción 3** del menú principal, **“Ver carta”**, podrá acceder a un submenú con opciones disponibles para realizar nuevas operaciones, como agregar, modificar o listar los platos disponibles en la carta del restaurante.

**Agregar carta:**



Si se selecciona la **opción 1**, **“Agregar carta”**, del submenú, el sistema solicitará el ID, el nombre, el tipo (plato, postre o bebida) y el precio del nuevo elemento que se desea registrar en la carta.

**Modificar carta:**



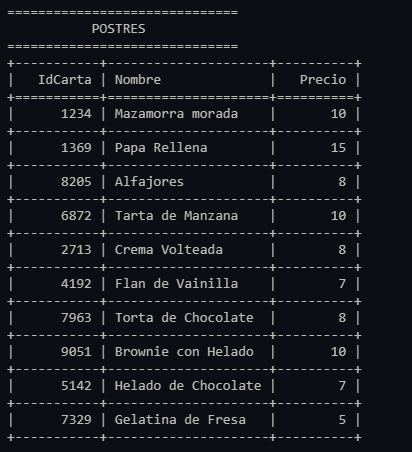
Al seleccionar la **opción 2**, **“Modificar carta”**, del submenú, el sistema solicitará el ID del elemento registrado (plato, bebida o postre) para poder actualizar su precio

**Eliminar carta:**



Al seleccionar la **opción 3, “Eliminar carta”,** del submenú, el sistema solicitara el ID del pedido, para posteriormente eliminarlo de la carta.

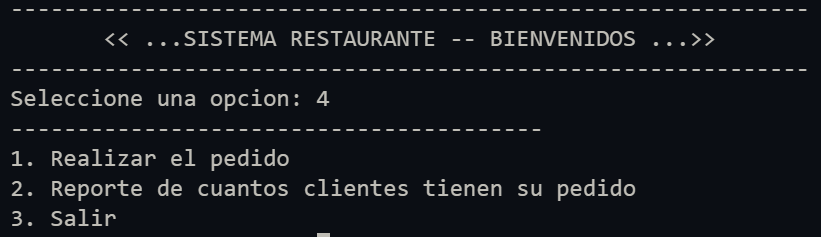
**Listar carta:**





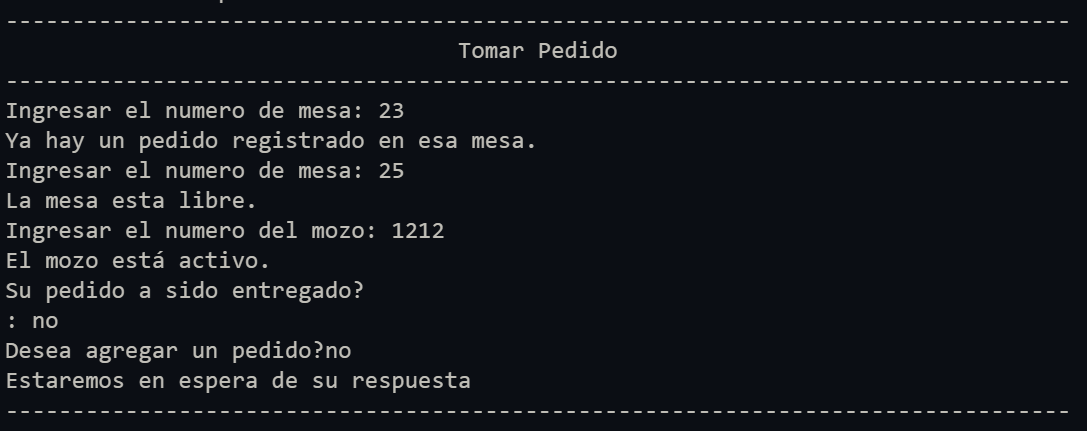
Si se selecciona la **opción 3**, **“Listar carta",** del submenú se mostrarán en pantalla todos los elementos registrados previamente (platos, bebidas y postres), los cuales estarán disponibles para realizar los pedidos de las mesas del restaurante.

**4.TOMAR PEDIDO**



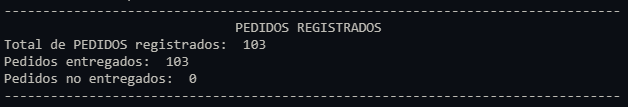
Si el usuario elige la **opción 4** del menú principal, **“Tomar pedido”**, podrá acceder a un submenú con opciones para registrar los pedidos correspondientes a las mesas ya reservadas. Además, podrá visualizar qué clientes ya recibieron su pedido y cuáles aún están pendientes.

**Realizar el Pedido:**



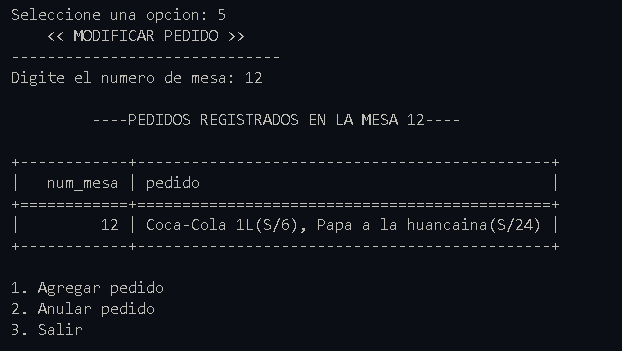
Al seleccionar la **opción1** del submenú, **“Realizar pedido”**, el programa solicitará el número de mesa y el ID del mozo previamente registrados. Además, preguntará si el pedido ya ha sido entregado para decidir si continuará con el proceso o cancelará la operación. Asimismo, brindará la opción de realizar un nuevo pedido o cancelar la acción en caso el usuario no desee continuar

**Reporte de cuantos clientes tienen su pedido:**



Al seleccionar la **opción 2** del submenú, **“Reporte de cuántos clientes tienen su pedido”**, el sistema mostrará en pantalla la cantidad total de pedidos realizados, indicando cuántos de ellos han sido entregados y cuántos aún están pendientes.

**5.MODIFICAR PEDIDO**

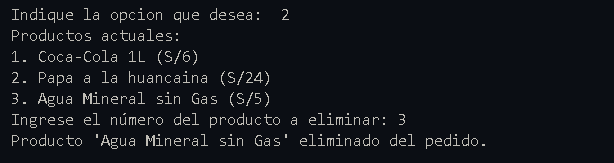


Al seleccionar la **opción 5** del menú principal **“Modificar Pedido”**, el sistema te pedirá el número de mesa a la cual deseas agregar o anular el pedido realizado anteriormente.

**1.Agregar pedido**

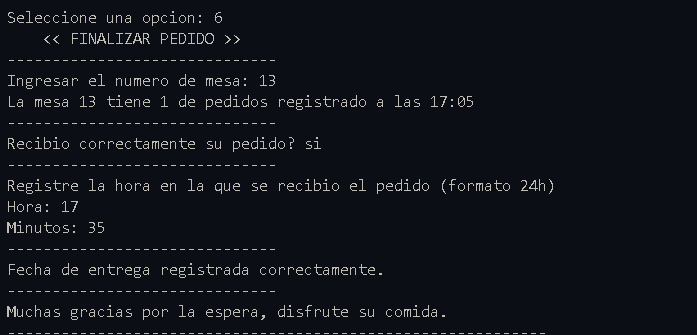
Al seleccionar la **opción 1** del submenú **'Agregar pedido'**, el sistema validara que el pedido no haya sido entregado ni pagado, para posteriormente mostrar la carta completa del restaurante, permitiendo elegir entre platos, bebidas y postres para registrar el pedido correspondiente.

**2. Anular Pedido**



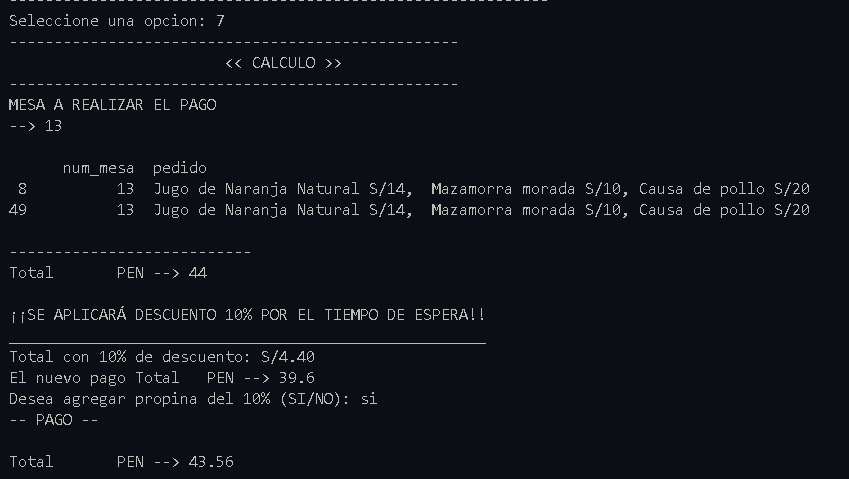
Si se escoge la **opción 2** del submenú **“Anular pedido”**, el programa mostrará todos los pedidos registrados para una mesa determinada. Luego, el usuario podrá seleccionar uno de esos pedidos para eliminarlo y posteriormente se mostrará un mensaje de confirmación del proceso.

**6.FINALIZAR PEDIDO**



Si se selecciona la **opción 6** del menú principal **“Finalizar pedido”**, el sistema solicitará primero el número de mesa. Luego, mostrará los pedidos realizados por esa mesa y permitirá registrar la hora y los minutos en que se entregó el pedido.

**7.CALCULAR PAGO**

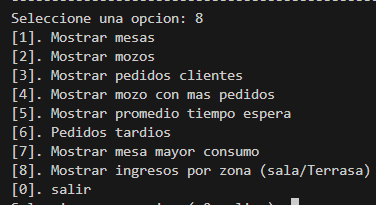


Si el usuario selecciona la **opción 7** del menú principal, **'Realizar pago'**, el sistema solicitará primero el número de mesa. A continuación, mostrará todos los pedidos realizados y calculará el total a pagar.

Esta opción también evalúa si el pedido fue entregado dentro del tiempo establecido (30 minutos después de haber sido registrado); de no ser así, se aplicará un descuento del 10 % sobre el total.

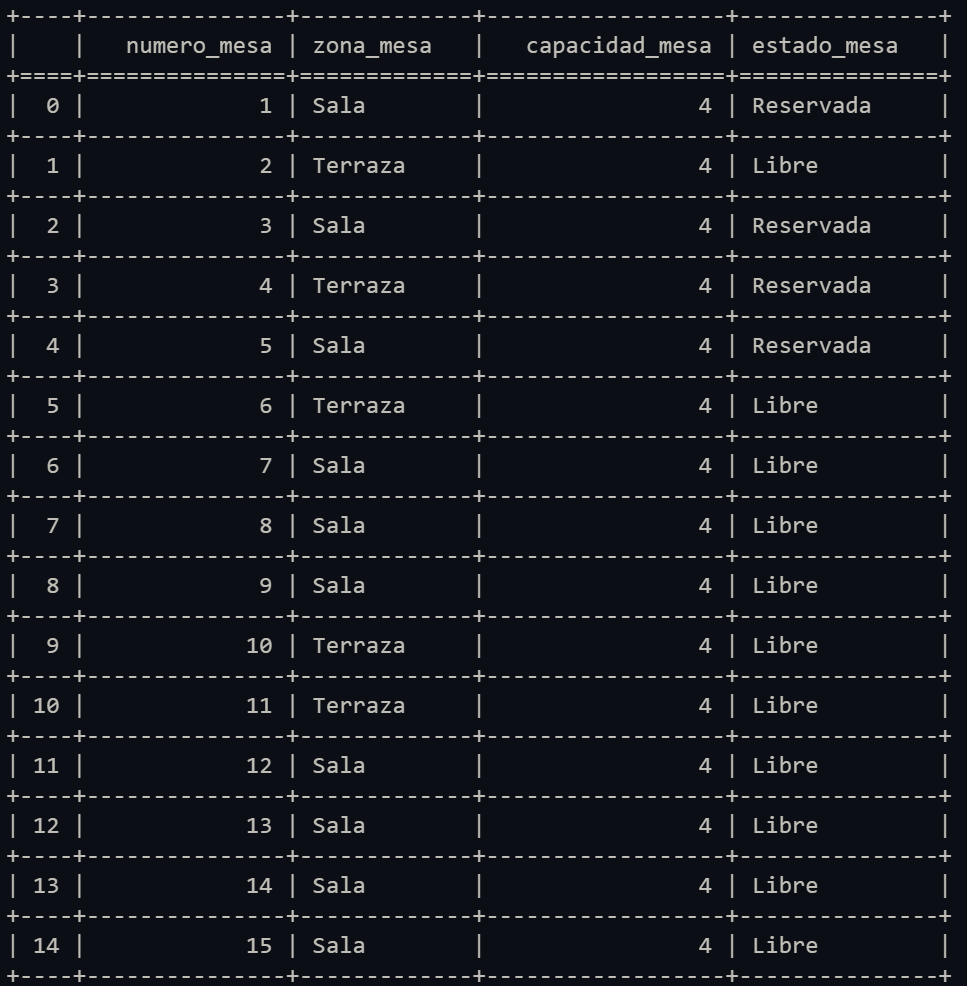
Además, el sistema ofrecerá la opción de agregar propina. Finalmente, se mostrará un mensaje que confirma la finalización del proceso de pago

**8.REPORTES**



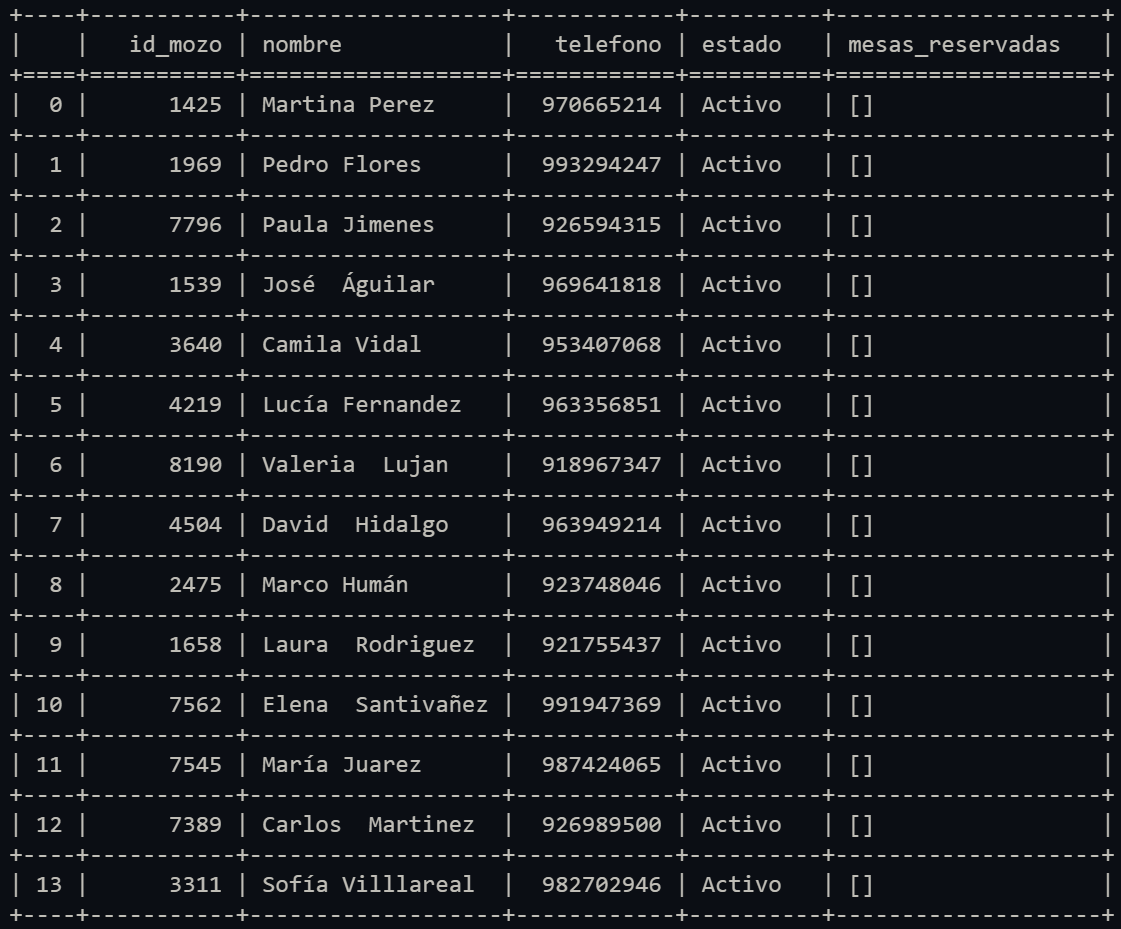
Al seleccionar la **opción 8 del menú principal, “Ver reportes”**, el usuario podrá acceder a un submenú que ofrece la visualización de los datos registrados en las diferentes funcionalidades del sistema.

**Mostrar mesas:**



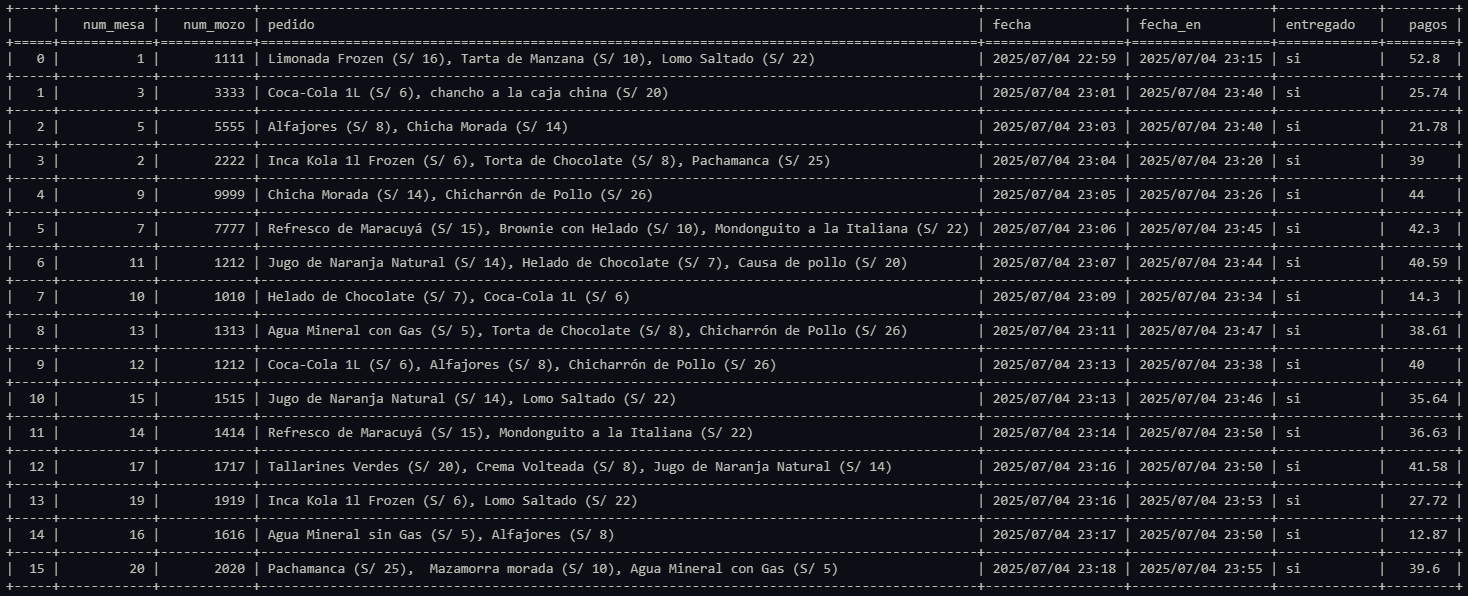
Si se selecciona la **opción 1,** **“Mostrar mesas”**, se podrá acceder a toda la información de las mesas registradas y almacenadas en el base de datos del restaurante.

**Mostrar mozos:**



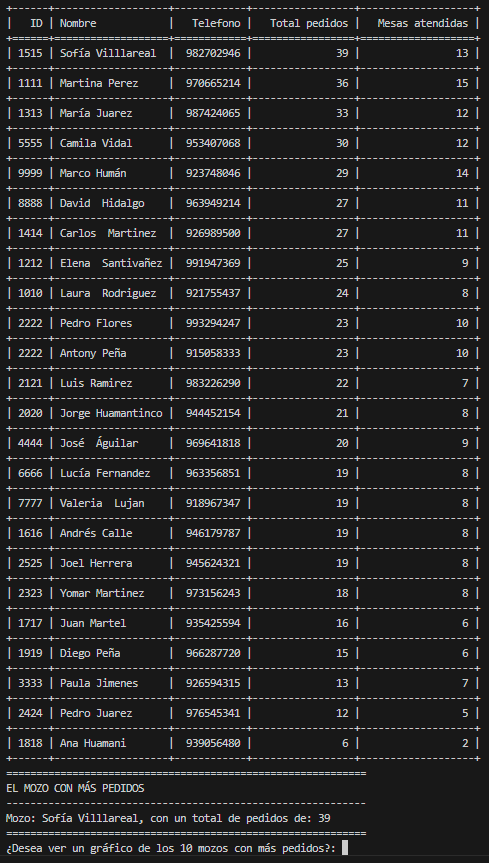
Si se selecciona la **opción 2,** **“Mostrar mozos”**, se podrá acceder a toda la información de los mozos registrados y almacenados en el base de datos del restaurante.

**Mostrar pedidos de clientes:**



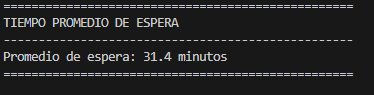
Si se elige la **opción 3 “Mostrar pedidos de clientes “**, se podrá visualizar toda la información de los pedidos registrados y almacenados en la base de datos.

**Mostrar mozo con más pedidos:**



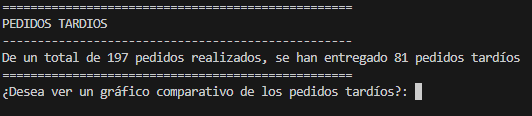
Si se selecciona la **opción 4,** **“Mostrar mozo con más pedidos”**, se podrá acceder a toda la información de los mozos con sus respectivos pedidos, además están ordenados de mayor a menor, y al final un mensaje de cuál fue el mozo con más pedidos.

**Mostrar promedio tiempo espera:**



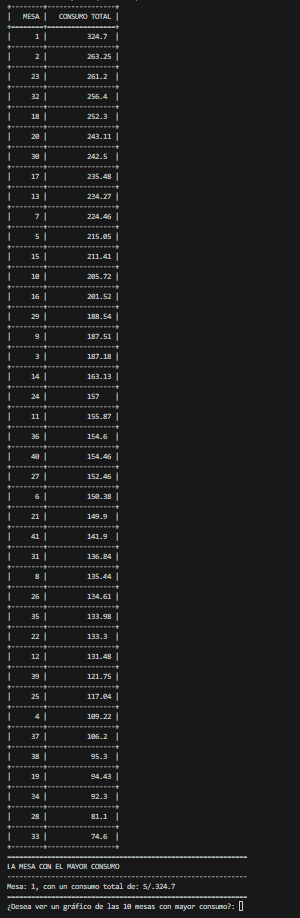
Si se selecciona la **opción 5,** **“Mostrar promedio tiempo espera”**, se le mostrará el promedio de espera de todos los pedidos realizados.

**Mostrar Pedidos tardíos:**



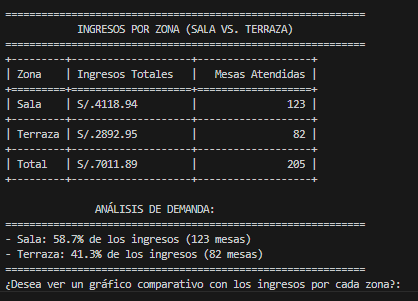
Si se selecciona la **opción 6,** **“Mostrar pedidos tardíos”**, se le mostrará el total de pedidos y el total de pedidos tardíos.

**Mostrar mesa mayor consumo:**



Si se selecciona la **opción 7,** **“Mostrar mesa con mayor consumo”**, se le mostrará un cuadro ordenado de mayor a menor, donde estarán las mesas con sus respectivos ingresos, además de un mensaje que mostrará la mesa con su mayor consumo.

**Mostrar ingresos por zona (sala/Terraza):**



Si se selecciona la **opción 8,** **“Mostrar ingresos por zona (Sala/Terraza)”**, se le mostrará un cuadro de ingresos por cada zona y su ingreso en totalidad, cada uno de estos con sus respectivas mesas, además se le mostrará un análisis de demanda, que este mostrará el porcentaje de ingresos de cada zona.

**CONCLUSIONES**

* **MEJORA EN MANEJO DE ESTRUCTURAS**

Durante el desarrollo del proyecto, logre mejorar mis habilidades en programación orientada a objetos con la creación de clases y dataframe, entrelazándolas entre sí para guardar adecuadamente los pedidos de cada usuario. Esto me permitió comprender y analizar de una mejor manera el manejo de datos y la estructura que pueden formarse a través de estos tipos de elementos.

* **BUENA ORGANIZACIÓN DE CODIGO**

Comprendí, durante el proyecto, que una buena organización es fundamental para escribir un código en Python. Por ejemplo, separar las funciones en bloques facilita notablemente la lectura del código y logra identificar errores de una manera más sencilla y rápida.

* **LIBRERIAS TABULATE, DATETIME, AST, PANDAS Y MATPLOTLIB**

Durante la elaboración del proyecto, mi grupo y yo logramos importar y utilizar diferentes librerías, como tabulate, para dar un formato mucho más presentable a los datos mostrados para el cliente. También usamos la librería datetime para generar variables con la fecha y hora actual, lo cual fue útil para registrar el momento exacto en la que se realizó un pedido. Adicionalmente, empleamos AST para convertir cadenas de texto en un objeto de Python, lo que nos permitió editarlo como listas de diccionarios. Con Pandas trabajamos estructuras tipo dataframe para una mejor gestión de los datos, y finalmente, utilizamos Matplotlib para mostrar gráficos estadísticos sobre los reportes del restaurante.

**RECOMENDACIONES**

* **VALIDACIONES DE VARIABLES:**

Una de las partes fundamentales al escribir código en cualquier lenguaje de programación es realizar las validaciones correspondientes a las variables. Esto permite que el usuario no pueda ingresar datos incoherentes, ya que podrían generar errores al mostrar las diferentes funciones de un proyecto y perjudicar la experiencia del usuario.

* **INTERFAZ GRAFICA:**

Una interfaz con un diseño simple y ordenado en cualquier proyecto logra que el usuario se sienta más cómodo y motivado a usar el programa. Esto mejora su experiencia de uso y reduce la probabilidad de que el cliente se confunda o se pierda entre demasiada información.

* **COPIA DE SEGURIDAD DE TU PROYECTO:**

Uno de los problemas más comunes al programar es que, al modificar alguna línea de código, se generen nuevos errores inesperados. Esto puede ocasionar que el problema se vuelva mucho más grande, afectando a más funciones del proyecto. Por ello, recomiendo crear siempre una copia de seguridad del código antes de realizar cualquier cambio, lo que permitirá volver fácilmente a una versión anterior en caso de emergencia.

**GLOSARIO DE TÉRMINOS**

|  |  |
| --- | --- |
| **TÉRMINOS:** | **CONCEPTO:** |
| **Class Mesas** | Es la clase que representa un objeto de tipo mesa en el restaurante y tiene atributos como número, zona (Sala / Terraza), capacidad y estado (Libre/Reservada) |
| **Class Mozos** | Es la clase que representa un objeto de tipo mozo en el restaurante y tiene atributos como ID, nombre, teléfono, estado (Activo/Inactivo) y mesas asignadas. |
| **Class Carta** | Es la clase que contiene la carta general del restaurante esta incluye Platos, bebidas postres disponibles |
| ***Class Pedido*** | Es la clase que hace la solicitud de productos de la carta hecha por una mea y atendida por un mozo. Incluye productos, fecha de pedido, fecha entrega y estado (entregado/no entregado). |
| ***Class Pago*** | Es la clase que maneja el proceso de cobro por los productos consumidos en la mesa, considerando descuentos y propinas. |
| **C*lass Restaurante*** | Gestiona la lógica principal del sistema, registro de mesas/mozos, asignación, pedidos, pagos, reportes y operaciones sobre los datos. |
| ***registrar\_mesas*** | Método que Permite agregar nuevas mesas al sistema, indicando la zona, capacidad y estado respectivo. |
| **registrar\_mozos** | Método que permite agregar nuevos mozos al sistema con valuación de los datos (ID, nombre, estado, teléfono). |
| **Registrar\_pedido** | Permite tomar el pedido de una mesa, asignando productos y mozo. |
| **Cambiar\_mozo\_mesa** | Permite transferir la atención de una mesa de un mozo a otro en caso de incidencias. |
| **Modificar\_pedido** | Permite agregar, cambiar o anular productos de un pedido existente. |
| **Finalizar\_pedido** | Marca el pedido como entregado, registra la hora y libera la mesa y la asignación del mozo. |
| **Eliminar\_mesa**  **Eliminar\_mozo** | Permite eliminar mesas o mozos del sistema. |
| **Liberar\_mesa** | Quita una mesa de las asignadas a un mozo y la pone como "Libre". |
| **Modificar\_datos\_mozo** | Permite actualizar ID, nombre, teléfono o estado de un mozo |
| **Agregar\_carta**  **Modificar\_carta** **Listar\_carta** | Permite gestionar los platos, bebidas y postres disponibles en la carta general del restaurante. |
| **Realizar\_pago** | Calcula el monto a pagar por una mesa, aplicando descuentos por demora y sumando propina si corresponde. |
| **Reporte\_mesas**  **Mostrar\_mesas** | Muestra cuántas mesas hay en cada zona (Sala/Terraza)  Muestra el reporte general de las mesas registradas en la base de datos del restaurante |
| **Reporte\_mozos**  **Mostrar\_mozos** | Muestra el total de mozos, cuántos están activos e inactivos  Muestra el reporte total de mozos existentes en la base de datos del restaurante. |
| **Reportes\_pedidos** | Muestra el total de pedidos, cuántos han sido entregados y cuántos no |
| **Mostrar\_pedidos** | Lista en pantalla todos los pedidos registrados por el sistema con sus respectivos datos y estados. |
| **Ast (Abstract Syntax Trees)** | Librería que permite leer y manipular el código fuente como si fuera una estructura de datos. |
| **Datetime** | Librería que nos permite acceder a la hora actual en la que el programa se ejecuta y así controlar los tiempos de una manera más fácil. |
| **Tabulate** | Es una librería de Python que permite convertir diversas estructuras de datos en tablas organizadas y legibles, facilitando la visualización de grandes volúmenes de información generada por el programa. |
| **main()** | Es la función principal del programa, donde se llaman a las demás funciones de forma organizada para lograr su ejecución. |
| **data\_mozos** | Data Frame donde se carga el archivo excel de mozos. |
| **data\_pedidos** | Data Frame donde se carga el archivo excel de pedidos. |
| **data\_carta** | Data Frame donde se carga el archivo excel de carta. |
| **data\_mesas** | Data Frame donde se carga el archivo excel de mesas. |
| **pandas** | Librería que proporciona estructuras de datos eficientes como DataFrame y Series, que permiten trabajar fácilmente con datos en formato tabular, similar a Excel o bases de datos. |
| **mozo\_mas\_pedido** | Muestra el reporte de mozos con más pedidos. |
| **prom\_tiempo\_e** | Muestra el reporte de promedio de tiempo de espera de pedidos. |
| **pedidos\_tardes** | Muestra el reporte de los pedidos tardíos. |
| **mesa\_mas\_cons** | Muestra el reporte de mesas con mayor consumo. |
| **ingreso-zona** | Muestra el reporte de ingreso por zona (Sala / Terraza). |

**Bibliografía:**

Castro Escobar, Ó. (2018). *Desarrollo de un sistema de información para un restaurante* [Trabajo Fin de Grado, Universidad de Málaga]. RIUMA. <https://hdl.handle.net/10630/17114>

Instituto Le Cordon Bleu. (s.f.). *Ingeniería del menú*. Instituto LeCordon Bleu Perú. Recuperado el 8 de mayo de 2025, de <https://www.ilcb.edu.pe/categoria/ingenieria-del-menu>

Moreno, A. (s.f.). *Diccionarios en Python*. El Libro de Python. Recuperado el 11 de mayo de 2025, de <https://ellibrodepython.com/diccionarios-en-python>

Moreno, A. (s.f.). *Listas en Python*. El Libro de Python. Recuperado el 11 de mayo de 2025, de <https://ellibrodepython.com/listas-en-python>

Python Central. (2025, abril 1). *Tabulación de Python: Creación de hermosas tablas a partir de sus datos*. Python Central. Recuperado el 8 de mayo de 2025, de <https://www.pythoncentral.io/python-tabulate-creating-beautiful-tables-from-your-data/>

Python Software Foundation. (2025). *Datetime — Tipos básicos de fecha y hora*. Documentación de Python. Recuperado el 8 de mayo de 2025, de <https://docs.python.org/3/library/datetime.html>

Python Software Foundation. (2024, 21 de abril). *Estructuras de datos*. Documentación de Python 3.13. Recuperado el 9 de mayo de 2025, de <https://docs.python.org/es/3.13/tutorial/datastructures.html>

Ramírez, O. (2025). *Diccionarios en Python (Mapas) - dicts*. El Pythonista. <https://elpythonista.com/diccionarios-en-python-dict>

Van Rossum, G., Warsaw, B., & Coghlan, A. (2025). *PEP 8 – Guía de estilo para código Python*. Python Enhancement Proposals. <https://peps.python.org/pep-0008/>

**Participación:**

|  |  |
| --- | --- |
| **Integrantes:** | **Porcentaje** |
| Hidalgo Martel, Joseph Edward | 100 % |
| Huamán Flores, Alexis Miguel | 100 % |
| Peña Roña, Antony Yomar | 100 % |
| Villavicencio Dávila, Ivette Lucero | 100 % |