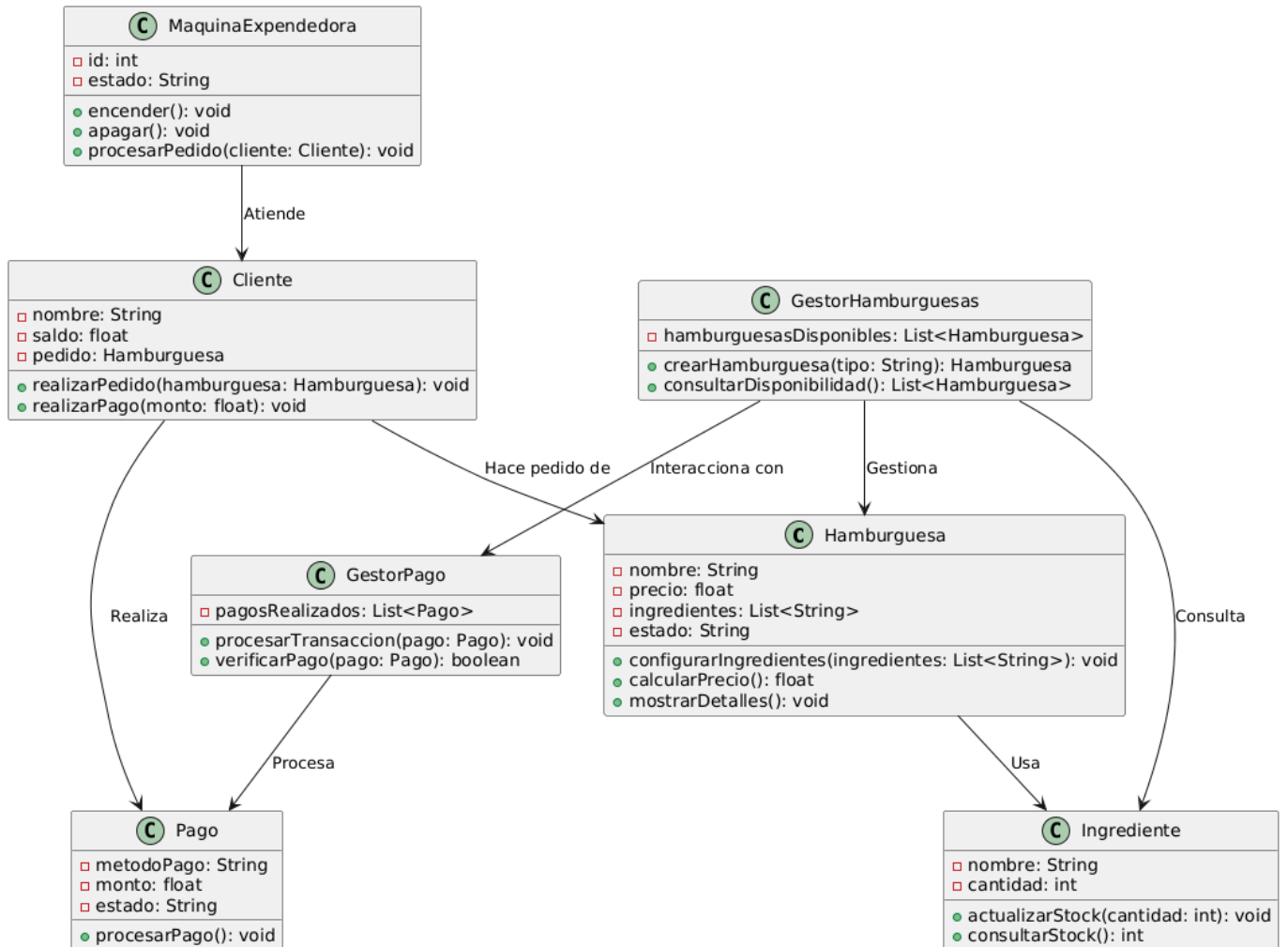


LÓGICO 5.4

DIAGRAMA DE CLASES



1. Clase "Hamburguesa"

- Atributos:
 - nombre: Representa el nombre de la hamburguesa (ejemplo: "Hamburguesa Doble", "Hamburguesa Vegetariana").
 - precio: El precio de la hamburguesa.
 - ingredientes: Una lista que contiene los ingredientes que lleva la hamburguesa (ejemplo: lechuga, carne, queso, etc.).
 - estado: El estado de la hamburguesa (ejemplo: "en proceso", "preparada", "entregada").
 - Métodos:
 - configurarIngredientes(ingredientes: List<String>): Permite configurar los ingredientes de la hamburguesa según las selecciones del cliente.
 - calcularPrecio(): Calcula el precio de la hamburguesa basado en los ingredientes seleccionados.
 - mostrarDetalles(): Muestra los detalles de la hamburguesa (nombre, precio y lista de ingredientes).
 - Relación:
 - La Hamburguesa es el objeto principal que el Cliente selecciona para pedir. Además, es utilizada en el GestorHamburguesas para ser creada y configurada.
-

2. Clase "Cliente"

- Atributos:
 - nombre: El nombre del cliente que realiza la compra.

- saldo: El saldo disponible del cliente para hacer el pago.
- pedido: El pedido realizado por el cliente, que es un objeto de tipo Hamburguesa.
- Métodos:
 - realizarPedido(hamburguesa: Hamburguesa): El cliente hace un pedido, el cual es una hamburguesa. Se guarda en el atributo pedido.
 - realizarPago(monto: float): El cliente realiza el pago por la hamburguesa. Esto reduce el saldo del cliente si el pago es exitoso.
- Relación:
 - El Cliente realiza un pedido de una Hamburguesa y luego efectúa el pago a través de la clase Pago.

3. Clase "Pago"

- Atributos:
 - metodoPago: El método de pago utilizado (por ejemplo, tarjeta de crédito, PayPal, efectivo, etc.).
 - monto: El monto del pago, que corresponde al precio de la Hamburguesa seleccionada.
 - estado: El estado del pago (puede ser "pendiente", "completado" o "fallido").
- Métodos:
 - procesarPago(): Este método simula el proceso de validación y autorización del pago. Cambia el estado de pago según el resultado.
- Relación:
 - El Pago se vincula con el Cliente, que es quien realiza el pago por su pedido. Además, el GestorPago se encarga de procesar este pago.

4. Clase "GestorHamburguesas"

- Atributos:
 - hamburguesasDisponibles: Lista de hamburguesas que están disponibles para que el cliente las seleccione.
- Métodos:
 - crearHamburguesa(tipo: String): Crea una hamburguesa de acuerdo al tipo que seleccione el cliente (por ejemplo, "Hamburguesa Doble", "Hamburguesa BBQ", etc.).
 - consultarDisponibilidad(): Muestra la lista de hamburguesas disponibles que el cliente puede elegir.
- Relación:
 - El GestorHamburguesas es responsable de la creación de las hamburguesas y gestionarlas. Interactúa con Ingrediente para consultar los ingredientes disponibles y preparar la hamburguesa.

5. Clase "GestorPago"

- Atributos:
 - pagosRealizados: Lista de los pagos realizados por los clientes.
- Métodos:
 - procesarTransaccion(pago: Pago): Este método procesa el pago realizado por el cliente, cambiando el estado del pago y actualizando la base de datos de pagos.
 - verificarPago(pago: Pago): Verifica si el pago fue exitoso o fallido.
- Relación:

- El GestorPago interactúa con Pago para procesar y verificar los pagos realizados por el cliente. También está vinculado al Cliente, ya que procesa sus pagos.

6. Clase "Ingrediente"

- Atributos:
 - nombre: El nombre del ingrediente (por ejemplo, "Queso", "Carne", "Lechuga").
 - cantidad: La cantidad de ese ingrediente disponible en la base de datos.
- Métodos:
 - actualizarStock(cantidad: int): Actualiza la cantidad disponible de un ingrediente.
 - consultarStock(): Consulta la cantidad disponible de un ingrediente.
- Relación:
 - Ingrediente es utilizado por la Hamburguesa, ya que las hamburguesas se crean utilizando los ingredientes disponibles. Además, el GestorHamburguesas interactúa con Ingrediente para consultar y actualizar el stock de ingredientes.

7. Clase "MaquinaExpendedora"

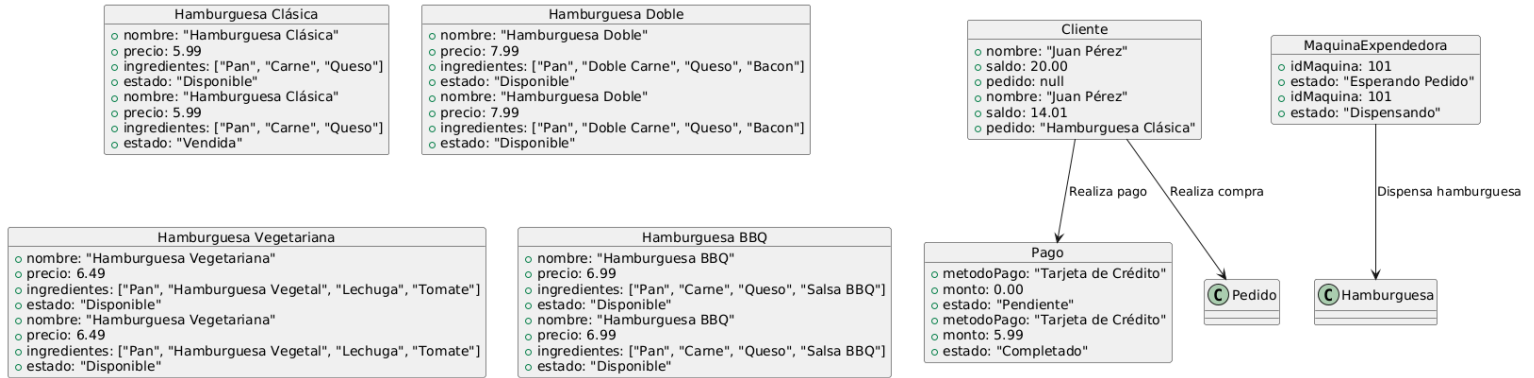
- Atributos:
 - id: Un identificador único para cada máquina expendedora.
 - estado: El estado de la máquina (por ejemplo, "encendida", "apagada").

- Métodos:
 - `encender()`: Enciende la máquina expendedora.
 - `apagar()`: Apaga la máquina expendedora.
 - `procesarPedido(cliente: Cliente)`: Procesa el pedido del cliente, entregando la hamburguesa seleccionada y gestionando la interacción con el cliente.
 - Relación:
 - La `MaquinaExpendedora` atiende al `Cliente`, entregando la hamburguesa una vez que el pedido ha sido procesado y el pago ha sido realizado.
-

Relaciones entre las clases:

- `Cliente` ↔ `Hamburguesa`: El cliente hace un pedido de una hamburguesa específica.
- `Cliente` ↔ `Pago`: El cliente realiza un pago para completar la compra de la hamburguesa.
- `GestorHamburguesas` ↔ `Hamburguesa`: El gestor de hamburguesas crea y administra las hamburguesas disponibles.
- `GestorPago` ↔ `Pago`: El gestor de pagos procesa las transacciones realizadas por los clientes.
- `Hamburguesa` ↔ `Ingrediente`: Las hamburguesas están compuestas por ingredientes, y la cantidad de ingredientes afecta a la creación de cada hamburguesa.
- `MaquinaExpendedora` ↔ `Cliente`: La máquina expendedora atiende al cliente entregando la hamburguesa después de completar la compra.

DIAGRAMA DE OBJETOS



1. Objeto "cliente1" (instancia de la clase Cliente):

- Este es un cliente que realiza una compra.
- Atributos:
 - nombre = "Juan Pérez": El nombre del cliente.
 - saldo = 50.0: El saldo disponible que tiene el cliente para realizar compras.
 - pedido = hamburguesa1: Este atributo contiene el objeto hamburguesa1 (una instancia de la clase Hamburguesa), que el cliente está pidiendo.

2. Objeto "hamburguesa1" (instancia de la clase Hamburguesa):

- Este es un ejemplo de una hamburguesa que ha sido seleccionada por el cliente.
- Atributos:

- nombre = "Hamburguesa Doble": El nombre de la hamburguesa seleccionada.
- precio = 20.0: El precio de la hamburguesa.
- ingredientes = [Carne, Queso, Lechuga, Pan]: La lista de ingredientes seleccionados para la hamburguesa.
- estado = "en proceso": El estado de la hamburguesa, indicando que está siendo preparada.

3. Objeto "pago1" (instancia de la clase Pago):

- Este objeto representa el pago que el cliente está realizando para su pedido.
- Atributos:
 - metodoPago = "Tarjeta de Crédito": El método de pago utilizado por el cliente.
 - monto = 20.0: El monto que el cliente paga.
 - estado = "pendiente": El estado del pago, indicando que aún está siendo procesado.

4. Objeto "gestorHamburguesas" (instancia de la clase GestorHamburguesas):

- El objeto gestorHamburguesas es responsable de gestionar las hamburguesas disponibles y de crear nuevas hamburguesas.
- Atributos:
 - hamburguesasDisponibles = [hamburguesa1, hamburguesa2, hamburguesa3]: Lista de hamburguesas que están disponibles para el cliente.

5. Objeto "gestorPago" (instancia de la clase GestorPago):

- El objeto gestorPago se encarga de procesar los pagos y verificar que el pago se haya realizado correctamente.

- Métodos:
 - procesarTransaccion(pago: Pago): Este método se utilizará para procesar el pago realizado por el cliente.
-

Relaciones y Procesos en el Diagrama de Objetos:

1. Cliente y Hamburguesa:

- El cliente1 hace un pedido de hamburguesa1. Esto está reflejado en el atributo pedido del objeto cliente1, que hace referencia a la instancia hamburguesa1.
- La hamburguesa1 tiene atributos como el nombre, precio e ingredientes. En este caso, es una hamburguesa con carne, queso, lechuga y pan.

2. Cliente y Pago:

- El cliente1 también tiene un objeto pago1 que representa el pago realizado. Este pago está asociado con el monto de la hamburguesa (20.0) y se realiza mediante el método "Tarjeta de Crédito".
- Inicialmente, el estado del pago está como "pendiente", lo que indica que el proceso de pago está en curso y aún no ha sido validado.

3. GestorHamburguesas y Hamburguesas:

- El gestorHamburguesas gestiona una lista de hamburguesas, de las cuales hamburguesa1 es una de ellas. Si el cliente selecciona una hamburguesa, el gestorHamburguesas se asegura de que esa hamburguesa esté disponible y de que sus ingredientes estén bien configurados.

4. GestorPago y Pago:

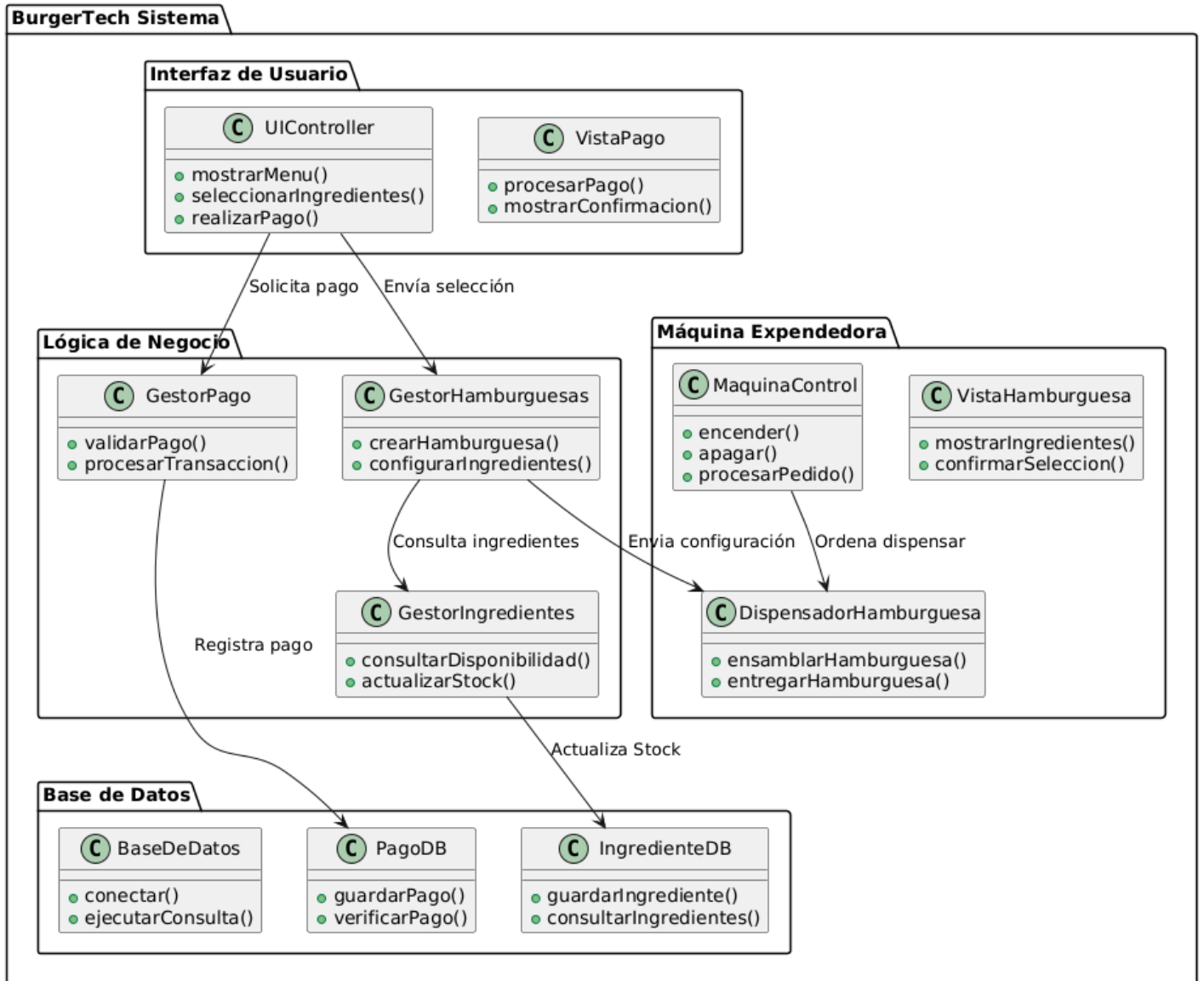
- El gestorPago es responsable de procesar pago1. Una vez que el cliente ha confirmado el pago, el gestorPago verifica si el pago es válido. Si el pago es validado correctamente, el estado de pago1 cambiará de "pendiente" a "completado".

Flujo del proceso en el diagrama de objetos:

1. El cliente1 realiza un pedido seleccionando hamburguesa1, la cual contiene ingredientes específicos como carne, queso, lechuga y pan.
2. El cliente1 realiza el pago utilizando pago1 (en este caso, con una tarjeta de crédito).
3. El gestorHamburguesas se asegura de que la hamburguesa1 esté disponible y configura los ingredientes seleccionados por el cliente.
4. El gestorPago procesa pago1. Si el pago es exitoso, el estado del pago cambia a "completado".
5. La hamburguesa1 pasa a estar en el estado de "preparada" cuando el pago se completa, y la maquina expendedora procesa el pedido del cliente.

DEPENDENCIA 5.5

DIAGRAMA PAQUETES



Paquetes en el Diagrama de Paquetes

1. Paquete "Cliente":

- Este paquete contiene todos los elementos relacionados con la interacción del cliente con el sistema. Aquí se encuentran las clases que gestionan las interacciones del cliente, como hacer un pedido, realizar un pago y ver los productos.
- Clases en este paquete:
 - Cliente: Representa al cliente que hace el pedido.
 - Pago: Representa el pago realizado por el cliente.
 - Pedido: (Opcional, si decides agregarla) Representa el pedido específico que el cliente realiza, incluyendo qué hamburguesa selecciona y si hay personalización de ingredientes.
- Responsabilidad: Este paquete se encarga de todo lo relacionado con las acciones del cliente: hacer el pedido, realizar el pago y consultar productos.

2. Paquete "Hamburguesa":

- Este paquete agrupa las clases que se encargan de la creación, configuración y gestión de las hamburguesas.
- Clases en este paquete:
 - Hamburguesa: Esta clase representa la hamburguesa en sí misma, con atributos como el nombre, los ingredientes y el precio.
 - Ingrediente: Clase que representa los ingredientes de las hamburguesas, como carne, queso, pan, etc.
 - GestorHamburguesas: Responsable de crear las hamburguesas, consultar su disponibilidad y administrar la configuración de los ingredientes.
- Responsabilidad: Este paquete maneja la lógica de creación de hamburguesas, agregando ingredientes y configurando el precio de

acuerdo a las selecciones del cliente.

3. Paquete "Pago":

- Este paquete agrupa las clases que manejan todo lo relacionado con el proceso de pago y la validación de los pagos del cliente.
- Clases en este paquete:
 - GestorPago: Gestiona el proceso de pago, verifica si el pago se ha completado y cambia el estado del pago según el resultado.
- Responsabilidad: Este paquete es responsable de procesar los pagos de los clientes, realizar las transacciones y asegurarse de que se haya completado correctamente el pago.

4. Paquete "MaquinaExpendedora":

- Este paquete representa el sistema de la máquina expendedora de hamburguesas. Contiene las clases que manejan la interacción con la máquina expendedora.
- Clases en este paquete:
 - MaquinaExpendedora: La clase que representa la máquina expendedora de hamburguesas. Es responsable de interactuar con el cliente, recibir pedidos y entregar el producto una vez que se ha completado el pago.
- Responsabilidad: Este paquete maneja la lógica de la máquina expendedora, desde la recepción del pedido hasta la entrega de la hamburguesa al cliente.

Relaciones entre los Paquetes

1. Cliente ↔ Hamburguesa:

- El paquete Cliente depende del paquete Hamburguesa porque el cliente realiza un pedido de hamburguesas. El cliente necesita acceder a la clase Hamburguesa y a los ingredientes para personalizar su

hamburguesa.

2. Cliente ↔ Pago:

- El paquete Cliente también depende del paquete Pago, ya que el cliente realiza un pago para completar su pedido. La clase Pago dentro del paquete Pago es crucial para que el cliente pueda hacer el pago.

3. Pago ↔ MaquinaExpendedora:

- El paquete Pago también depende de MaquinaExpendedora para que el proceso de pago se complete antes de que la máquina expendedora entregue la hamburguesa. La máquina espera la confirmación del pago para proceder con la entrega.

4. Hamburguesa ↔ MaquinaExpendedora:

- La máquina expendedora interactúa con el paquete Hamburguesa para asegurarse de que la hamburguesa está lista antes de entregarla al cliente. La máquina expendedora consulta las hamburguesas creadas y las envía una vez que se confirma el pago.

Flujo de Actividad en el Diagrama de Paquetes

El flujo de actividades en el diagrama de paquetes sería algo así:

1. El Cliente hace un pedido:

- El paquete Cliente interactúa con el paquete Hamburguesa para seleccionar una hamburguesa, configurarla con ingredientes y verificar el precio.

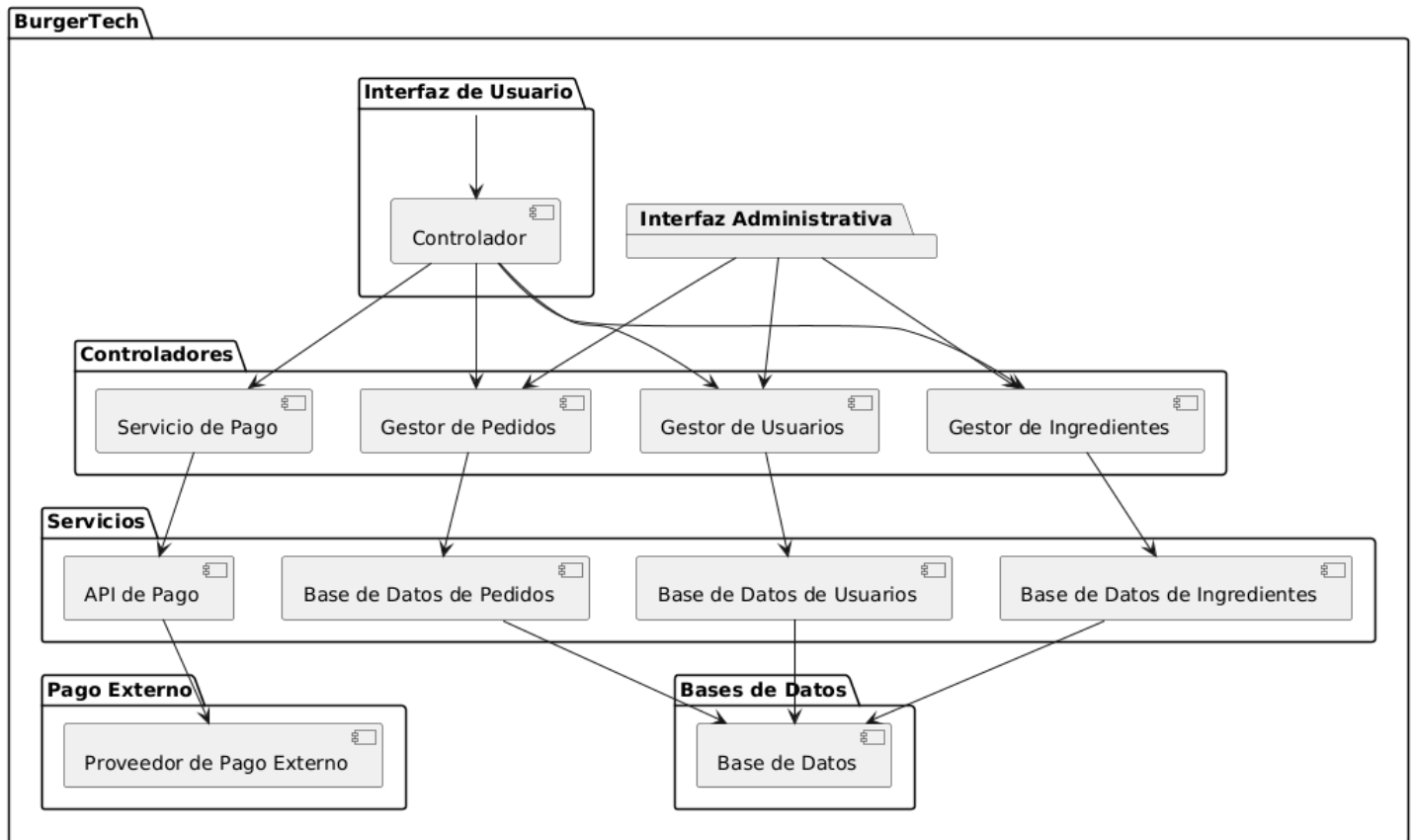
2. El Cliente realiza el pago:

- El cliente interactúa con el paquete Pago para procesar el pago a través de la clase GestorPago.

3. La MaquinaExpendedora entrega la hamburguesa:

- Una vez confirmado el pago, el paquete MaquinaExpendedora entrega la hamburguesa seleccionada al cliente.

DIAGRAMA DE COMPONENTES



1. Interfaz de Usuario (UI)

- Componente: "Interfaz de Usuario":
 - Representa la capa que interactúa directamente con el usuario. Esta capa permite a los clientes seleccionar hamburguesas, personalizarlas, realizar pedidos y pagos.
 - Interfaz: "Interfaz Gráfica": Es la interfaz de comunicación entre la Interfaz de Usuario y el sistema. La Interfaz Gráfica está diseñada para presentar la información al usuario (pantallas, botones, etc.) y tomar entradas del cliente (como seleccionar ingredientes o realizar un pago).

- Relación: La Interfaz de Usuario utiliza la Interfaz Gráfica para mostrar al usuario las opciones y recibir la entrada de usuario.

2. Lógica de Negocio

- Componentes:

- Gestor Hamburguesas: Es el componente que maneja la creación, personalización y gestión de las hamburguesas. Aquí se controla todo lo que tiene que ver con la adición de ingredientes, cálculo de precios y configuración de hamburguesas personalizadas.
- Gestor Pago: Este componente maneja el proceso de pago, incluyendo la validación del pago y la actualización del estado de la transacción. Se encarga de asegurarse de que el cliente ha realizado un pago exitoso.
- Controlador Pedido: Actúa como intermediario entre la Interfaz de Usuario y el Gestor Hamburguesas. Este componente es el encargado de coordinar las solicitudes de hamburguesas por parte del cliente, asegurándose de que las hamburguesas sean seleccionadas y personalizadas según las preferencias del cliente.
- Controlador Pago: Similar al Controlador Pedido, este componente interactúa con el Gestor Pago para manejar el proceso de pago, verificar que los pagos se procesen correctamente y confirmar la transacción.

- Relaciones:

- El Controlador Pedido consulta al Gestor Hamburguesas para obtener las opciones de hamburguesas disponibles o para personalizar las hamburguesas que el cliente ha elegido.
- El Controlador Pago consulta al Gestor Pago para verificar el estado del pago y asegurarse de que se ha procesado correctamente.
- Gestor Hamburguesas invoca al Gestor Pago para asegurar que el pago haya sido realizado antes de proceder con el pedido de la hamburguesa.

3. Máquina Expendedora

- Componente: "Máquina Expendedora":
 - Representa el hardware o el sistema físico que entrega las hamburguesas una vez que el cliente ha realizado el pago.
 - La Máquina Expendedora consulta al Gestor Hamburguesas para obtener la hamburguesa seleccionada por el cliente.
 - También interactúa con el Gestor Pago para confirmar que el pago ha sido completado antes de entregar la hamburguesa.
- Relaciones:
 - Máquina Expendedora consulta al Gestor Hamburguesas para obtener la información sobre la hamburguesa que el cliente seleccionó.
 - Máquina Expendedora verifica el pago con el Gestor Pago antes de entregar el producto.

4. Base de Datos

- Componente: "Base de Datos":
 - Este componente maneja toda la persistencia de datos en el sistema. Se encarga de almacenar los productos (hamburguesas, ingredientes, precios), los pagos y cualquier otra información necesaria.
 - Interfaces: Tablas de Productos y Tablas de Pagos son las interfaces que permiten a otros componentes consultar y almacenar información sobre los productos (hamburguesas) y los pagos realizados.
 - Relaciones:
 - Gestor Hamburguesas interactúa con la Base de Datos para obtener la lista de hamburguesas disponibles y sus respectivos ingredientes.
 - Gestor Pago consulta la Base de Datos para verificar si el pago se ha completado y para registrar la transacción de pago.
-

Resumen de las Relaciones de los Componentes:

1. Interfaz de Usuario ↔ Lógica de Negocio:

- La Interfaz de Usuario interactúa con la Lógica de Negocio a través de los Controladores. Los controladores son responsables de coordinar las interacciones entre la UI y la lógica del sistema.

2. Lógica de Negocio ↔ Base de Datos:

- Los componentes de la Lógica de Negocio (como el Gestor Hamburguesas y el Gestor Pago) interactúan con la Base de Datos para obtener o almacenar información relacionada con hamburguesas, pedidos y pagos.

3. Máquina Expendedora ↔ Lógica de Negocio:

- La Máquina Expendedora depende de la Lógica de Negocio para asegurarse de que el pedido esté confirmado antes de entregar la hamburguesa. Esto involucra consultas al Gestor Hamburguesas y al Gestor Pago.