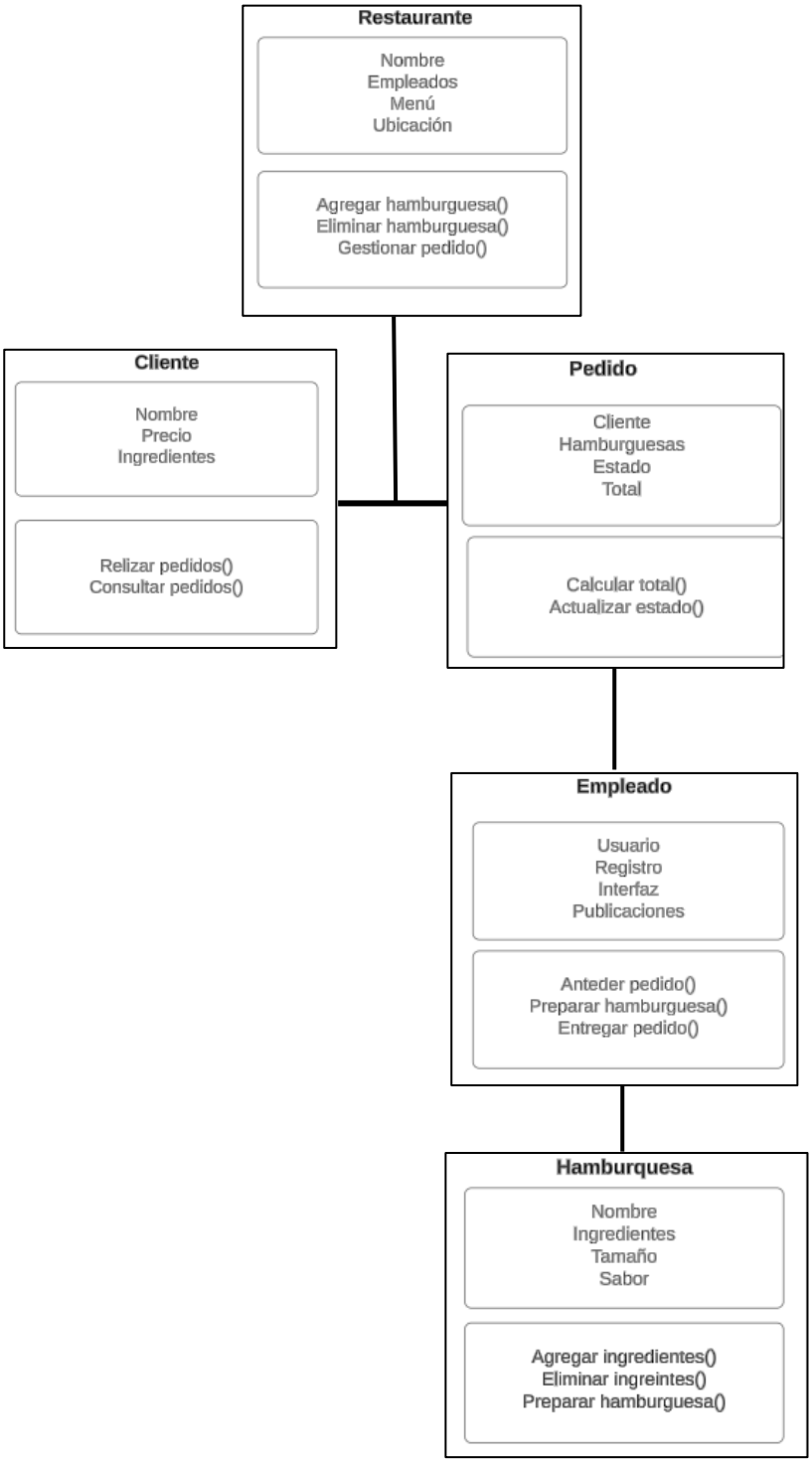
**Universidad de las Fuerzas Armadas ESPE**

Nombre: Dylan López

NRC: 1322



Relaciones clave:

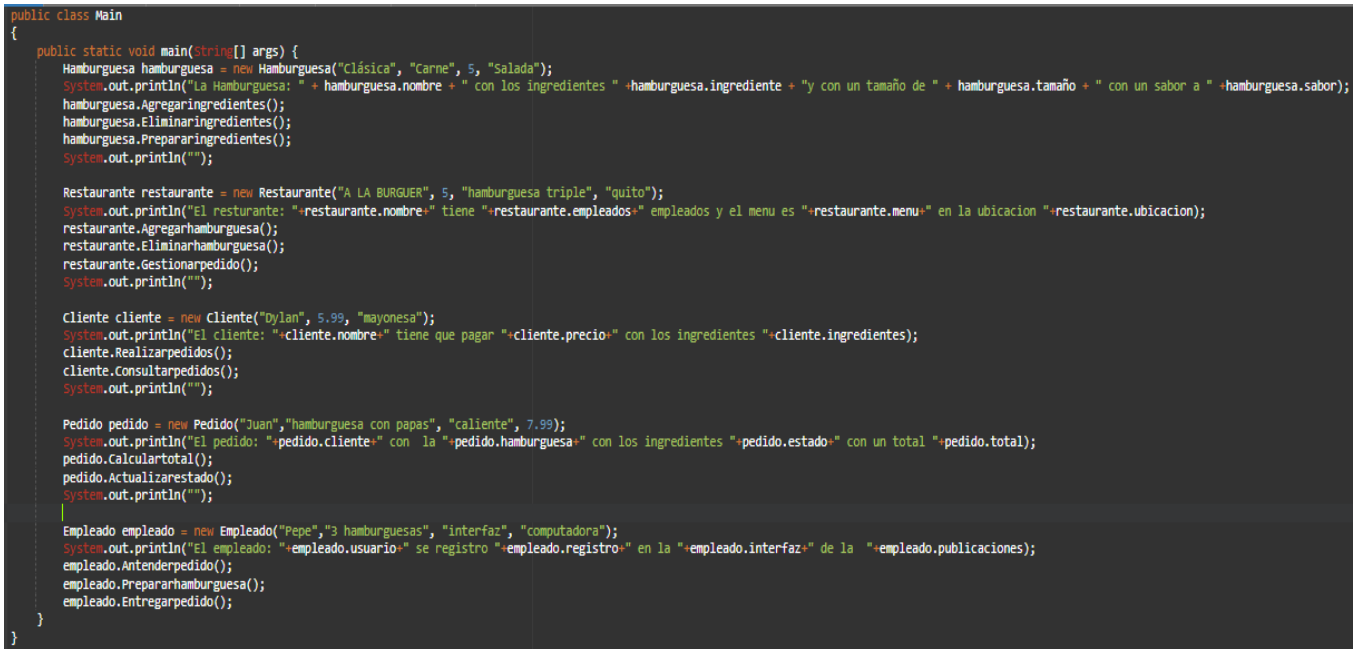
Cliente ↔ Pedido: Asociación (un cliente puede realizar varios pedidos).

Pedido ↔ Hamburguesa: Agregación (un pedido está compuesto por varias hamburguesas).

Empleado ↔ Pedido y Empleado ↔ Hamburguesa: Asociación (el empleado atiende pedidos y prepara hamburguesas).

Restaurante ↔ Hamburguesa, Restaurante ↔ Pedido, Restaurante ↔ Empleado: Asociación (el restaurante gestiona el menú, los pedidos y los empleados).

1. **Main del código:**



```
public class Main
{
    public static void main(String[] args) {
        Hamburguesa hamburguesa = new Hamburguesa("Clásica", "Carne", 5, "Salada");
        System.out.println("La Hamburguesa: " + hamburguesa.nombre + " con los ingredientes " +hamburguesa.ingrediente + "y con un tamaño de " + hamburguesa.tamaño + " con un sabor a " +hamburguesa.sabor);
        hamburguesa.Agregaringredientes();
        hamburguesa.Eliminaringredientes();
        hamburguesa.Prepararingredientes();
        System.out.println("");

        Restaurante restaurante = new Restaurante("A LA BURGUER", 5, "hamburguesa triple", "quito");
        System.out.println("El resturante: "+restaurante.nombre+" tiene "+restaurante.empleados+" empleados y el menu es "+restaurante.menu+" en la ubicacion "+restaurante.ubicacion);
        restaurante.Agregarhamburguesa();
        restaurante.Eliminarhamburguesa();
        restaurante.Gestionarpedido();
        System.out.println("");

        Cliente cliente = new Cliente("Dylan", 5.99, "mayonesa");
        System.out.println("El cliente: "+cliente.nombre+" tiene que pagar "+cliente.precio+" con los ingredientes "+cliente.ingredientes);
        cliente.Realizarpedidos();
        cliente.Consultarpedidos();
        System.out.println("");

        Pedido pedido = new Pedido("Juan","hamburguesa con papas", "caliente", 7.99);
        System.out.println("El pedido: "+pedido.cliente+" con  la "+pedido.hamburguesa+" con los ingredientes "+pedido.estado+" con un total "+pedido.total);
        pedido.Calculartotal();
        pedido.Actualizarestado();
        System.out.println("");

        Empleado empleado = new Empleado("Pepe","3 hamburguesas", "interfaz", "computadora");
        System.out.println("El empleado: "+empleado.usuario+" se registro "+empleado.registro+" en la "+empleado.interfaz+" de la  "+empleado.publicaciones);
        empleado.Antenderpedido();
        empleado.Prepararhamburguesa();
        empleado.Entregarpedido();
    }
}
```

- Public class Main: Define la clase principal del programa, donde se encuentra el punto de entrada
- public static void main(String[] args): Define el método principal (main), que es el punto de inicio de cualquier programa Java.
- Creación de Objetos:
  Se crean instancias de las clases Hamburguesa, Restaurante, Cliente, Pedido y Empleado.
  Cada objeto representa un elemento o entidad del sistema:
- Una hamburguesa con atributos como su nombre, ingredientes, tamaño, y sabor.
- Un restaurante con su menú, ubicación, y número de empleados.
- Un cliente que realiza pedidos con un precio asociado y preferencias de ingredientes.
- Un pedido que incluye detalles del cliente, estado y costo.
- Un empleado encargado de atender, preparar, y entregar pedidos.

Con System.out.println: se imprimen detalles en consola sobre las acciones realizadas y las características de los objetos.

- hamburguesa.Agregaringredientes();
- hamburguesa.Eliminaringredientes();

- hamburguesa.Prepararingredientes();
  Llama a los métodos de la clase Hamburguesa para realizar acciones relacionadas con los ingredientes: Agregar ingredientes, Eliminar ingredientes, Preparar ingredientes.

**2. Hamburguesa.java**

```java
public class Hamburguesa
{
    public String nombre;
    public String ingrediente;
    public int tamaño;
    public String sabor;

    public Hamburguesa(String nombre, String ingrediente, int tamaño, String sabor){
        this.nombre=nombre;
        this.ingrediente=ingrediente;
        this.tamaño=tamaño;
        this.sabor=sabor;
    }

    public void Agregaringredientes() {
        System.out.println("Se agregaron los ingredientes a la hamburguesa  " + nombre);
    }
    public void Eliminaringredientes() {
        System.out.println("Se eliminaron algunos ingredientes de la hamburguesa " + nombre);
    }
    public void Prepararingredientes() {
        System.out.println("Se prepararon los ingredientes de la hamburguesa " + nombre);
    }



}
```

- public class Hamburguesa

  Define una clase pública llamada Hamburguesa.

  Representa una hamburguesa con atributos y comportamientos específicos.

- Atributos de la clase

public String nombre;

public String ingrediente;

public int tamaño;

public String sabor;

nombre: Cadena de texto que almacena el nombre de la hamburguesa (ejemplo: "Clásica").

ingrediente: Cadena que describe el ingrediente principal de la hamburguesa (ejemplo: "Carne").

tamaño: Entero que representa el tamaño de la hamburguesa (ejemplo: 5).

sabor: Cadena que describe el sabor característico de la hamburguesa (ejemplo: "Salada").

- Constructor de la clase

public Hamburguesa(String nombre, String ingrediente, int tamaño, String sabor) {

this.nombre = nombre;

this.ingrediente = ingrediente;

this.tamaño = tamaño;

this.sabor = sabor;

}

Es un método especial que se ejecuta automáticamente cuando se crea un objeto de esta clase.

this: Se refiere al atributo de la instancia actual, diferenciándolo de los parámetros.

- Metodo

public void Agregaringredientes() {

System.out.println("Se agregaron los ingredientes a la hamburguesa " + nombre);

}

Imprime un mensaje indicando que se han agregado ingredientes a la hamburguesa.

Usa el atributo nombre para personalizar el mensaje.

### 3. Restaurante.java

```java
public class Restaurante{
    public String nombre;
    public int empleados;
    public String menu;
    public String ubicacion;

    public Restaurante(String nombre, int empleados, String menu, String ubicacion){
        this.nombre=nombre;
        this.empleados=empleados;
        this.menu=menu;
        this.ubicacion=ubicacion;
    }

    public void Agregarhamburguesa() {
        System.out.println("Se agregaron nuevos ingredientes al menú del restaurante "+ nombre );
    }
    public void Eliminarhamburguesa() {
        System.out.println("Se eliminaron algunos ingredientes del menú del restaurante "+ nombre);
    }
    public void Gestionarpedido() {
        System.out.println("Se preparó un nuevo plato en el restaurante " + nombre);
    }

}
```

- Definición de la Clase

public class Restaurante {

Define una clase pública llamada Restaurante.

- Atributos de la Clase

public String nombre;

public int empleados;

public String menu;

public String ubicacion;

nombre: Cadena que almacena el nombre del restaurante (por ejemplo: "A LA BURGUER").

empleados: Entero que representa la cantidad de empleados que trabajan en el restaurante.

menu: Cadena que describe el menú o platillo destacado del restaurante (por ejemplo: "hamburguesa triple").

ubicacion: Cadena que indica la ubicación del restaurante (por ejemplo: "Quito").

- Constructor de la Clase

```
public Restaurante(String nombre, int empleados, String menu, String ubicacion) {
   this.nombre = nombre;
   this.empleados = empleados;
   this.menu = menu;
   this.ubicacion = ubicacion;
}
```

nombre: Nombre del restaurante.

empleados: Número de empleados en el restaurante.

menu: Menú destacado del restaurante.

ubicacion: Ubicación física del restaurante.

this: Se refiere a los atributos de la instancia actual, diferenciándolos de los parámetros.

- Metodo

```
public void Agregarhamburguesa() {
   System.out.println("Se agregaron nuevos ingredientes al menú del restaurante " + nombre);
}
```

Imprime un mensaje indicando que se han agregado ingredientes al menú del restaurante.

4. **Cliente.java**

```java
public class Cliente{
    public String nombre;
    public Double precio;
    public String ingredientes;

    public Cliente(String nombre, Double precio, String ingredientes){
        this.nombre=nombre;
        this.precio=precio;
        this.ingredientes=ingredientes;
    }

    public void Realizarpedidos() {
        System.out.println("Se realizaron los siguientes pedidos " + nombre );
    }
    public void Consultarpedidos() {
        System.out.println("Se consultaron los siguientes pedidos "+ nombre);
    }

}
```

- Definición de la Clase

public class Cliente {

Define una clase pública llamada Cliente.

Representa a un cliente del restaurante, con información sobre sus pedidos y preferencias.

- Atributos de la Clase

public String nombre;

public Double precio;

public String ingredientes;

nombre: Cadena que almacena el nombre del cliente (por ejemplo: "Dylan").

precio: Un número de tipo Double que representa el total a pagar por el cliente.

ingredientes: Cadena que describe los ingredientes preferidos del cliente (por ejemplo: "mayonesa").

- Constructor de la Clase

public Cliente(String nombre, Double precio, String ingredientes) {

this.nombre = nombre;

this.precio = precio;

this.ingredientes = ingredientes;

}

nombre: Nombre del cliente.

precio: Monto que debe pagar el cliente.

ingredientes: Ingredientes seleccionados o preferidos del cliente.

this: Se refiere a los atributos de la instancia actual, diferenciándolos de los parámetros.

- Método

public void Realizarpedidos() {

System.out.println("Se realizaron los siguientes pedidos " + nombre);

}

Simula que el cliente realiza un pedido.

Personaliza el mensaje en consola utilizando el atributo nombre del cliente.

### 5. Pedido.java

```java
public class Pedido{
    public String cliente;
    public String hamburguesa;
    public String estado;
    public Double total;

    public Pedido(String cliente, String hamburguesa, String estado, Double total){
        this.cliente=cliente;
        this.hamburguesa=hamburguesa;
        this.estado=estado;
        this.total=total;
    }

    public void Calculartotal() {
        System.out.println("Se calcula el siguiente pedido " + cliente );
    }
    public void Actualizarestado() {
        System.out.println("Se acualizo el pedido "+ cliente);
    }

}
```

- Definición de la Clase

public class Pedido {

Declara una clase pública llamada Pedido.

Representa un pedido realizado en el restaurante, incluyendo información sobre el cliente, la hamburguesa, su estado y el precio total.

- Atributos de la Clase

public String cliente;

public String hamburguesa;

public String estado;

public Double total;

cliente: Cadena que almacena el nombre del cliente asociado al pedido (por ejemplo: "Juan").

hamburguesa: Cadena que describe el tipo de hamburguesa solicitada (por ejemplo: "hamburguesa con papas").

estado: Cadena que indica el estado actual del pedido (por ejemplo: "caliente", "preparando", "listo").

total: Número de tipo Double que representa el precio total del pedido (por ejemplo: 7.99).

- Constructor de la Clase

public Pedido(String cliente, String hamburguesa, String estado, Double total) {

this.cliente = cliente;

this.hamburguesa = hamburguesa;

this.estado = estado;

this.total = total;

}

cliente: Nombre del cliente.

hamburguesa: Descripción de la hamburguesa pedida.

estado: Estado actual del pedido.

total: Monto total del pedido.

this: Diferencia entre los atributos del objeto actual y los parámetros recibidos

- Método

public void Calculartotal() {

System.out.println("Se calcula el siguiente pedido " + cliente);

}

Imprime un mensaje personalizado con el atributo cliente, indicando que su pedido está siendo calculado.

### 6. Empleado.java

```java
public class Empleado{
    public String usuario;
    public String registro;
    public String interfaz;
    public String publicaciones;

    public Empleado(String usuario, String registro, String interfaz, String publicaciones){
        this.usuario=usuario;
        this.registro=registro;
        this.interfaz=interfaz;
        this.publicaciones=publicaciones;
    }

    public void Antenderpedido() {
        System.out.println("Se atendio el siguiente pedido " + usuario );
    }
    public void Prepararhamburguesa() {
        System.out.println("Se preparo el siguiente pedido "+ usuario);
    }
    public void Entregarpedido() {
        System.out.println("Se entrego el siguiente pedido "+ usuario);
    }

}
```

- Definición de la Clase

public class Empleado {

Declara una clase pública llamada Empleado.

Representa a un empleado del restaurante, con información personal y las acciones que puede realizar.

- Atributos de la Clase

public String usuario;

public String registro;

public String interfaz;

public String publicaciones;

usuario: Cadena que almacena el nombre o identificador del empleado (por ejemplo: "Pepe").

registro: Cadena que describe el tipo de registro o actividad asociada al empleado (por ejemplo: "3 hamburguesas").

interfaz: Cadena que indica la interfaz o área de trabajo donde se realiza la actividad (por ejemplo: "interfaz").

publicaciones: Cadena que describe el medio o herramienta utilizada (por ejemplo: "computadora").

- Constructor de la Clase

```
public Empleado(String usuario, String registro, String interfaz, String publicaciones) {
   this.usuario = usuario;
   this.registro = registro;
   this.interfaz = interfaz;
   this.publicaciones = publicaciones;
}
```

usuario: Nombre o identificador del empleado.

registro: Registro asociado al trabajo del empleado.

interfaz: Área o entorno donde trabaja.

publicaciones: Herramienta o medio utilizado.

this: Diferencia entre los atributos del objeto actual y los parámetros del constructor.

- Método

```
public void Antenderpedido() {
   System.out.println("Se atendió el siguiente pedido " + usuario);
```

}

Simula que el empleado atiende un pedido.

Imprime un mensaje personalizado utilizando el atributo usuario para identificar al empleado que realiza la acción.

**Ejecución del código:**

```
La Hamburguesa: Clásica con los ingredientes Carney con un tamaño de 5 con un sabor a Salada
Se agregaron los ingredientes a la hamburguesa  Clásica
Se eliminaron algunos ingredientes de la hamburguesa Clásica
Se prepararon los ingredientes de la hamburguesa Clásica

El resturante: A LA BURGUER tiene 5 empleados y el menu es hamburguesa triple en la ubicacion quito
Se agregaron nuevos ingredientes al menú del restaurante A LA BURGUER
Se eliminaron algunos ingredientes del menú del restaurante A LA BURGUER
Se preparó un nuevo plato en el restaurante A LA BURGUER

El cliente: Dylan tiene que pagar 5.99 con los ingredientes mayonesa
Se realizaron los siguientes pedidos Dylan
Se consultaron los siguientes pedidos Dylan

El pedido: Juan con  la hamburguesa con papas con los ingredientes caliente con un total 7.99
Se calcula el siguiente pedido Juan
Se acualizo el pedido Juan

El empleado: Pepe se registro 3 hamburguesas en la interfaz de la  computadora
Se atendio el siguiente pedido Pepe
Se preparo el siguiente pedido Pepe
Se entrego el siguiente pedido Pepe
```