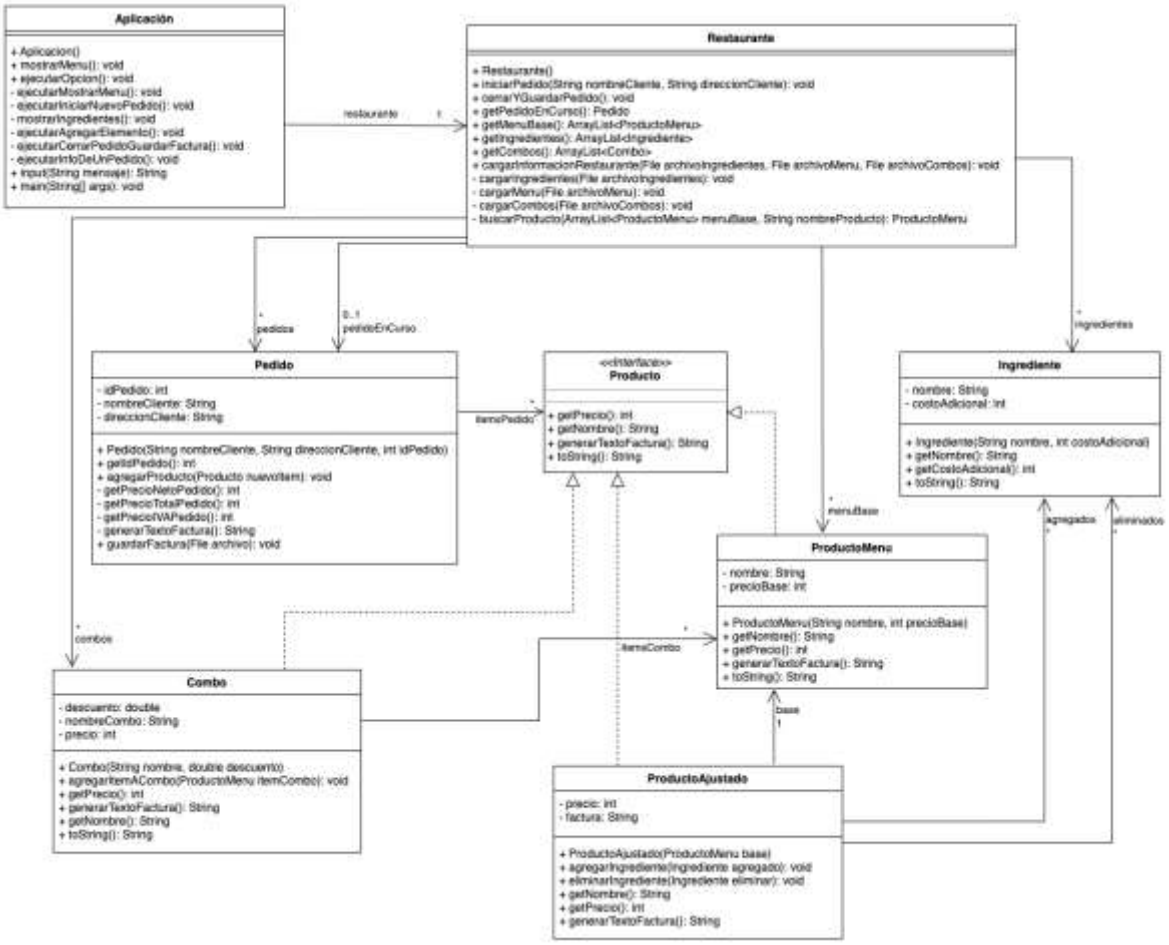
María Catalina Ibáñez Piñeres - 201922462 - m.ibanez@uniandes.edu.co

María Alejandra Pérez Petro - 201923972 - ma.perezp@uniandes.edu.co

DPOO-S04-202220-G04

Taller2-Hamburguesas

Diagrama UML



Modificaciones

1. Aplicación

En esta clase se implementaron 10 métodos. Dos de los cuales corresponden a los métodos de mostrarMenu() y ejecutar opción() del UML dado en las instrucciones, seis corresponden a los metodos encargados de ejecutar cada opcion del menú y por utlimo, dos corresponden a los métodos main(String[] args) e input(). Cabe resaltar que a diferencia del UML dado en las instrucciones, el método ejecutar opción no recibe por parametro un int con la opción seleccionada del menú, si no que en cambio, se pregunta al usuario por dicha opción dentro del método.

2. Restaurante

En está clase se implementaron dos métodos adicionales a los indicados en el UML de las instrucciones. El primero fue el método publico getCombos el cual permite consultar el ArrayList<Combo> combos. El segundo fue un método privado buscarProducto, el cual retorna el objeto productoMenu cuyo nombre corresponde al ingresado por parametro. Este ultimo método se utilizo al momento de cargar el archivo combos (método cargarCombos) con el objetivo de ir agregando los items del combo en la carga de datos.

3. Pedido

En está clase se implementaron exactamente los métodos indicados en el UML de las instrucciones. No obstante, cabe resaltar que no se creo el atributo numerosPedidos, puesto que no se consideró necesario para la implementación del programa.

4. Ingrediente

En está clase se implementaron los métodos indicados en el UML de las instrucciones y adicionalmente se implementó el método toString() con el fin de poder imprimir mas amigablemente la información de la clase Ingrediente (nombre y costo adicional).

5. Producto

Para esta clase interface se crearon los métodos indicados en el UML de las instrucciones y adicionalmente se creó el método toString().

6. ProductoMenu

En está clase se implementaron los métodos indicados en el UML de las instrucciones y adicionalmente se implementó el método toString() con el fin de poder imprimir mas amigablemente la información de la clase ProductoMenu(nombre y precioBase).

7. ProductoAjustado

En está clase se implementaron los métodos indicados en el UML de las instrucciones: getNombre(), getPrecio() y generarTextoFactura(). Adicionalmente se crearon los métodos agregarIngrediente(Ingrediente) y eliminarIngrediente(Ingrediente), los cuales añadian un ingrediente al arrayList agregados y elimiandos respectivamente. Lo anterior se realizó con el fin de registrar los ingredientes que se modificaban en el ProductoAjustado. Tambien se crearon dos atributos adicionales precio (int) y factura (String) con el fin de facilitar el cálculo del precio del ProductoAjustado y la generación del respectivo texto de la factura. Para ello, cada vez que se agregaba un ingrediente, en el método agregarIngrediente(Ingrediente) también se modificaba el precio del producto ajsutado, sumandole el costoAdicional de agregar el ingrediente.

8. Combo

En está clase se implementaron los métodos indicados en el UML de las instrucciones y adicionalmente se implementó el método toString() con el fin de poder imprimir mas amigablemente la información de la clase Combo(nombre, precio e items del combo). También

cabe resaltar que, se modificó el método agregarItemACombo() para recibir un objeto de clase ProductoMenu en vez de uno de clase Producto.

Mapa menú de opciones

Al ejecutar la aplicación se le muestra al usuario un mapa con seis opciones las cuales le permiten interactuar con el programa tal como se muestra en el diagrama de a continuación.