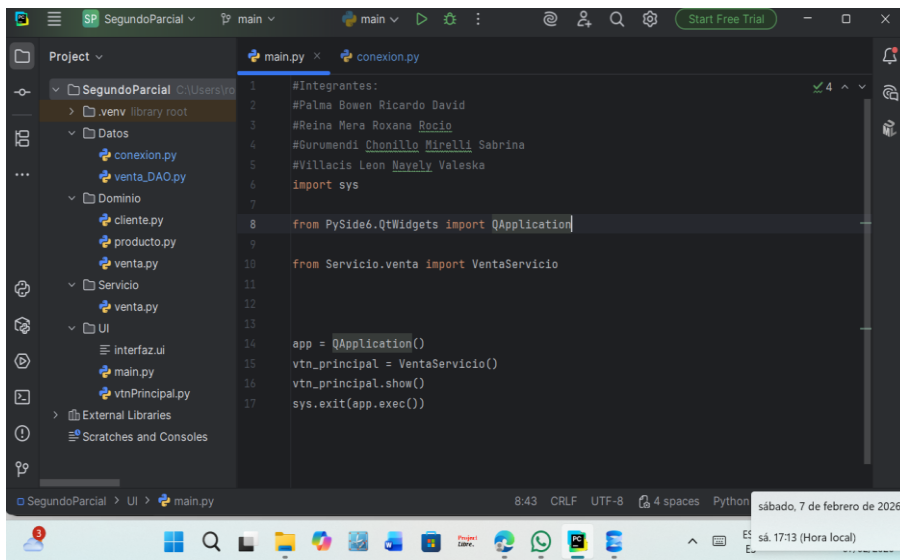
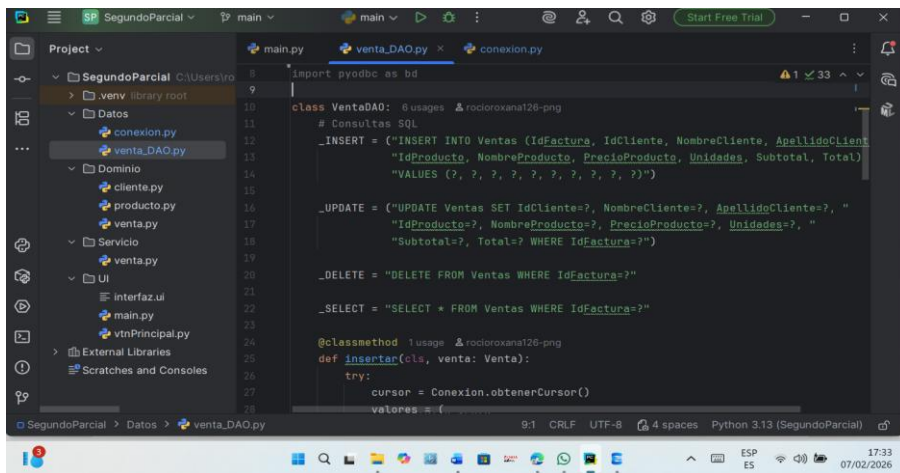


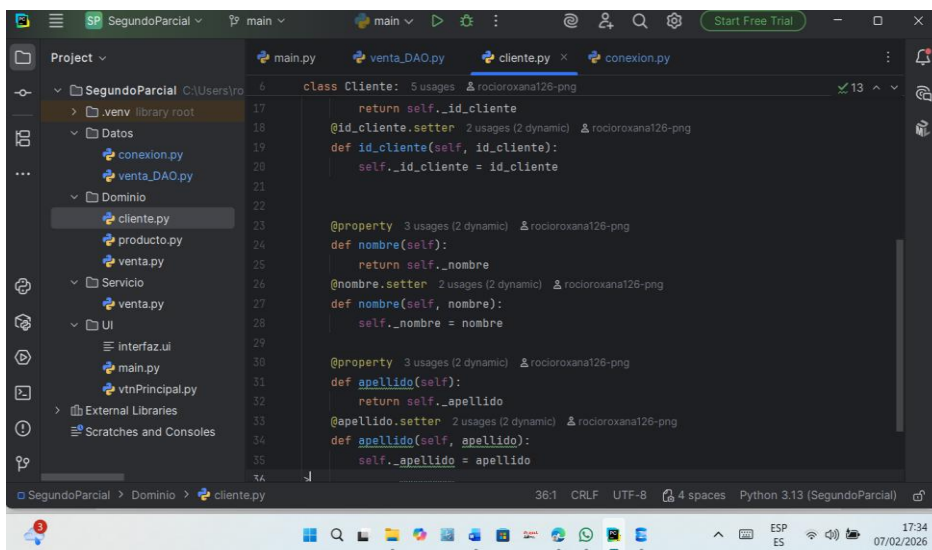
Capturas Proyecto: Venta Online



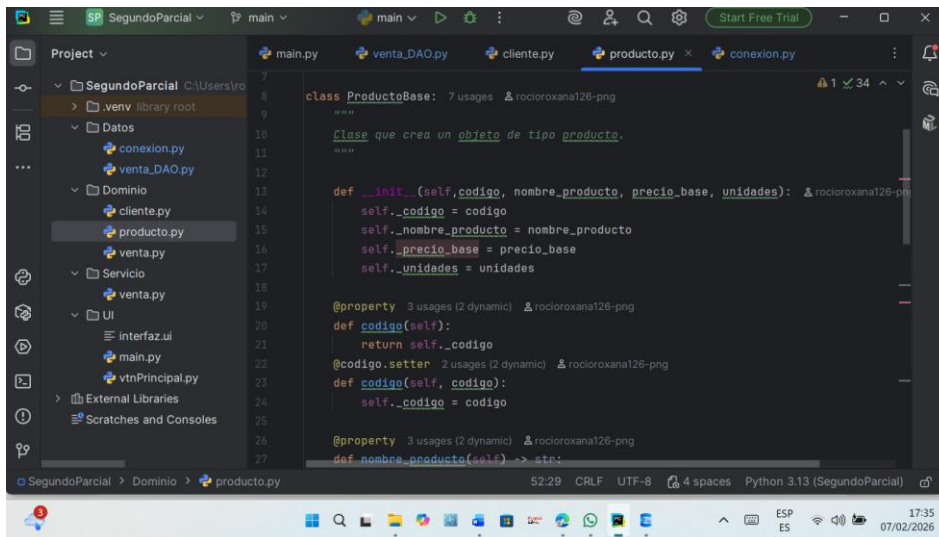
```
1 #Integrantes:
2 #Palma Bowen Ricardo David
3 #Reina Mera Roxana Rocio
4 #Gurumendi Chonillo Mirelli Sabrina
5 #Villacis Leon Nayely Valeska
6
7 import sys
8
9 from PySide6.QtWidgets import QApplication
10
11 from Servicio.venta import VentaServicio
12
13
14 app = QApplication()
15 vtn_principal = VentaServicio()
16 vtn_principal.show()
17 sys.exit(app.exec())
```



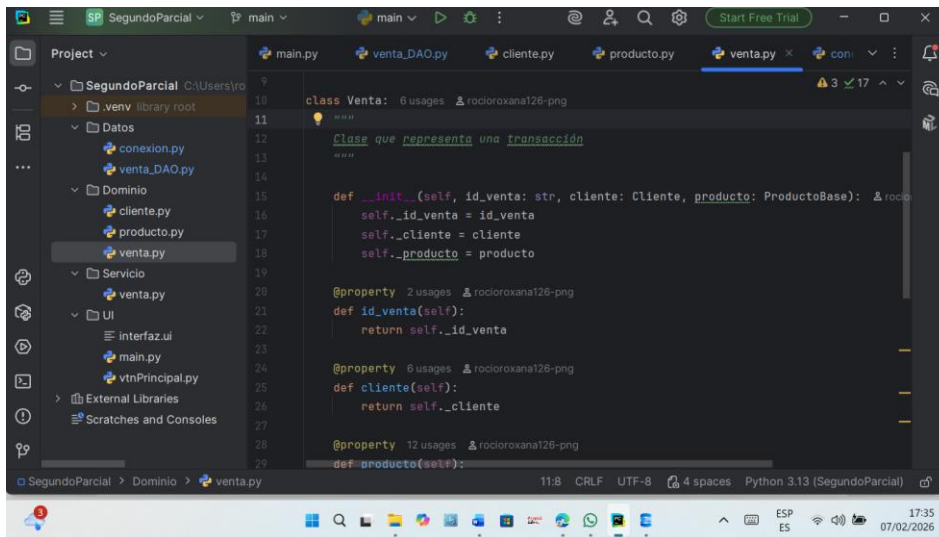
```
8 import pyodbc as bd
9
10 class VentaDAO:
11     # Consultas SQL
12     _INSERT = ("INSERT INTO Ventas (IdFactura, IdCliente, NombreCliente, ApellidoCliente, "
13               "IdProducto, NombreProducto, PrecioProducto, Unidades, Subtotal, Total) "
14               "VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?)")
15
16     _UPDATE = ("UPDATE Ventas SET IdClientes=?, NombreCliente=?, ApellidoCliente=?, "
17               "IdProducto=?, NombreProducto=?, PrecioProducto=?, Unidades=?, "
18               "Subtotal=?, Total=? WHERE IdFactura=?")
19
20     _DELETE = "DELETE FROM Ventas WHERE IdFactura=?"
21
22     _SELECT = "SELECT * FROM Ventas WHERE IdFactura=?"
23
24     @classmethod
25     def insertar(cls, venta: Venta):
26         try:
27             cursor = Conexion.obtenerCursor()
28             valores = (
```



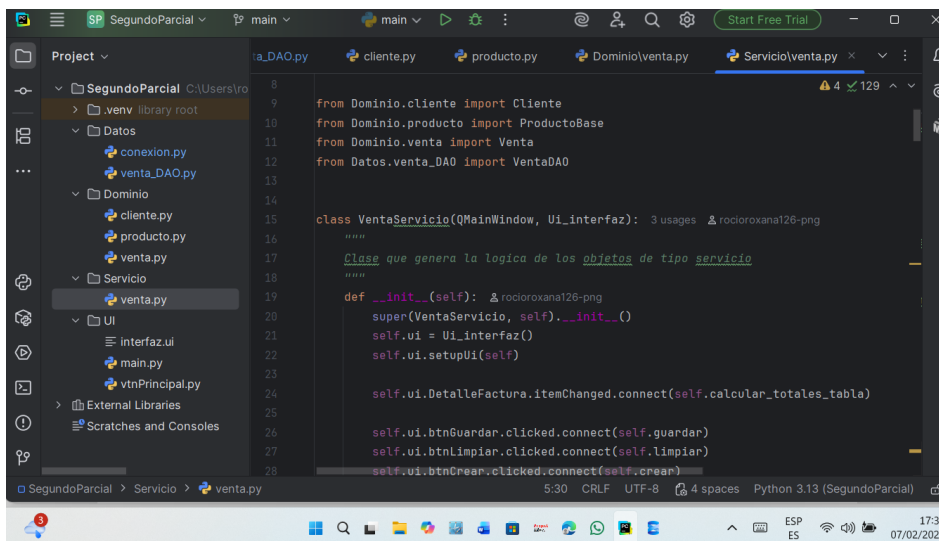
```
6 class Cliente:
7     return self._id_cliente
8
9     @id_cliente.setter
10     def id_cliente(self, id_cliente):
11         self._id_cliente = id_cliente
12
13
14     @property
15     def nombre(self):
16         return self._nombre
17
18     @nombre.setter
19     def nombre(self, nombre):
20         self._nombre = nombre
21
22
23     @property
24     def apellido(self):
25         return self._apellido
26
27     @apellido.setter
28     def apellido(self, apellido):
29         self._apellido = apellido
```



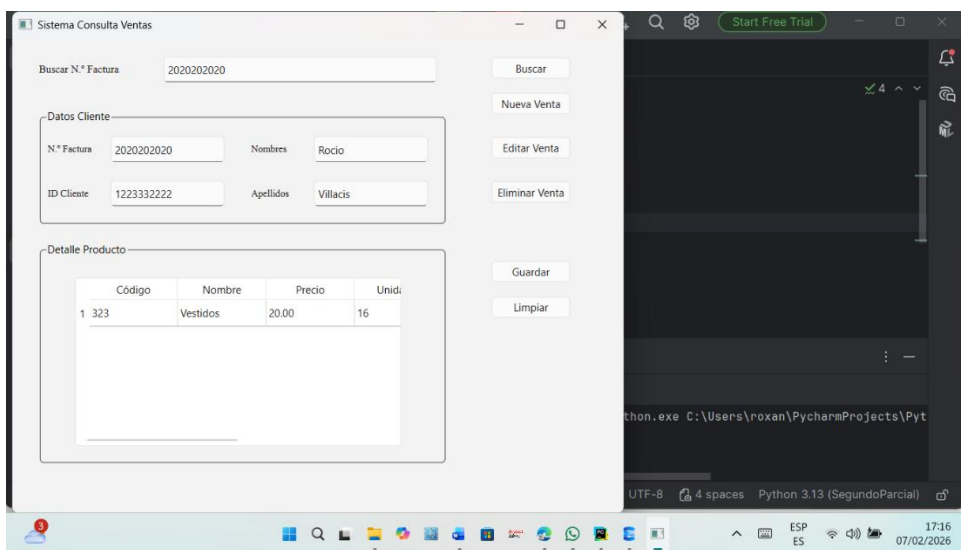
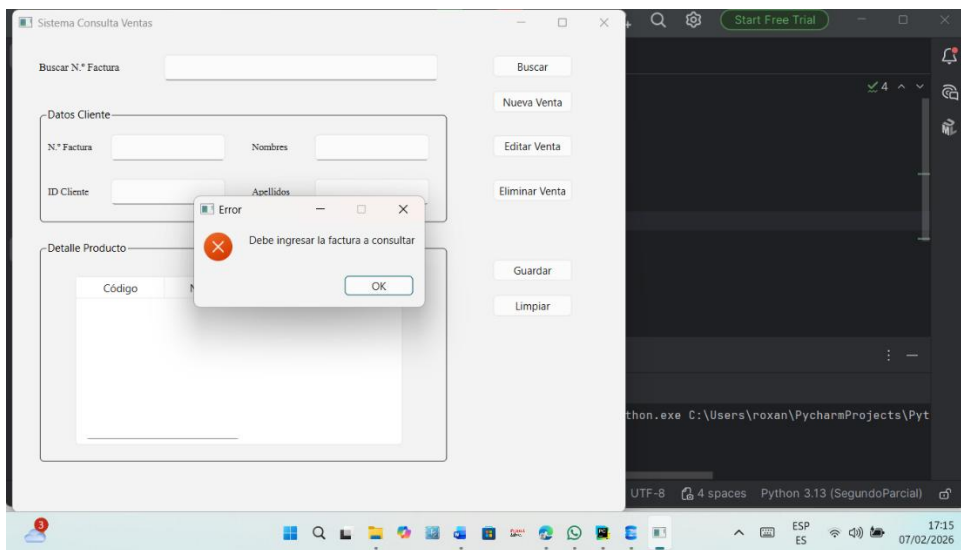
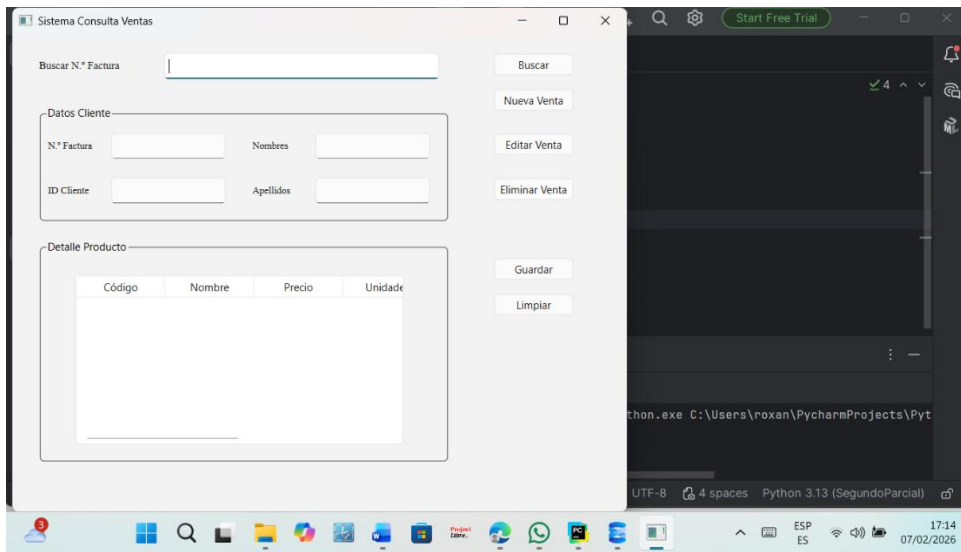
```
7 class ProductoBase:
8     """
9     Clase que crea un objeto de tipo producto.
10    """
11
12    def __init__(self, codigo, nombre_producto, precio_base, unidades):
13        self._codigo = codigo
14        self._nombre_producto = nombre_producto
15        self._precio_base = precio_base
16        self._unidades = unidades
17
18    @property
19    def codigo(self):
20        return self._codigo
21
22    @codigo.setter
23    def codigo(self, codigo):
24        self._codigo = codigo
25
26    @property
27    def nombre_producto(self) -> str:
```

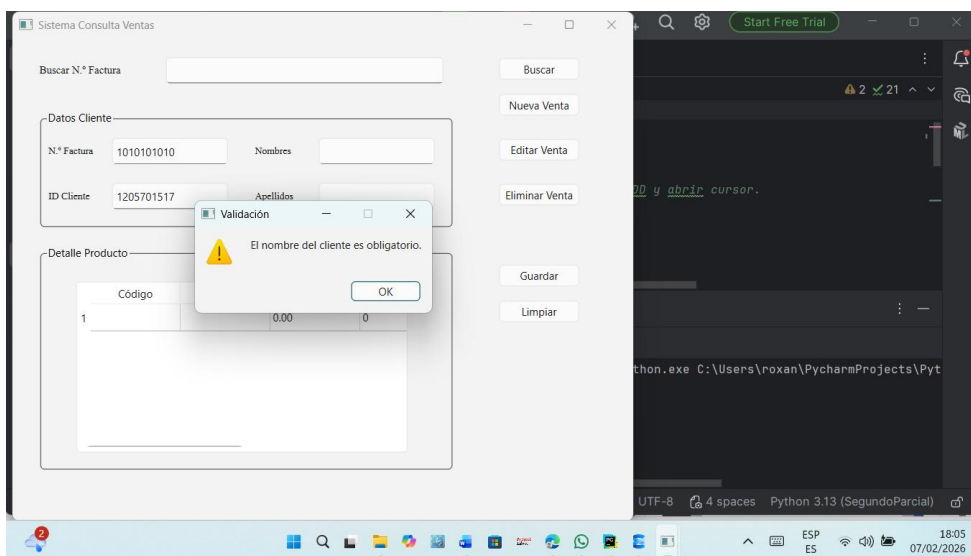
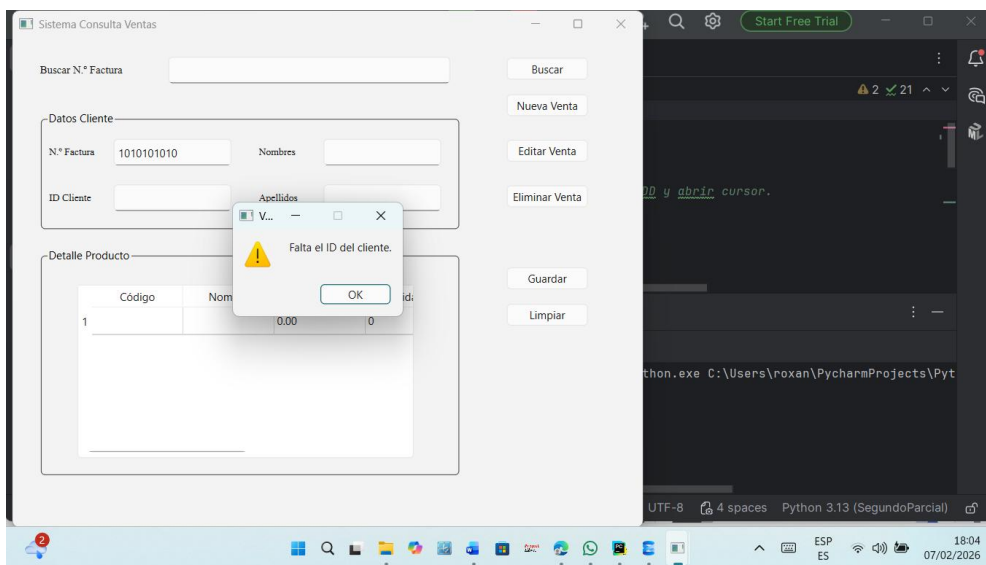
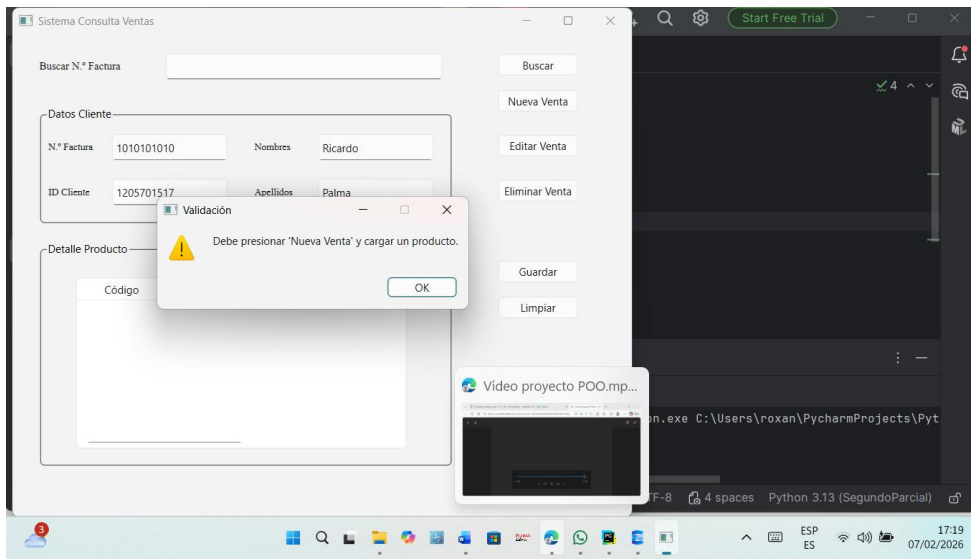


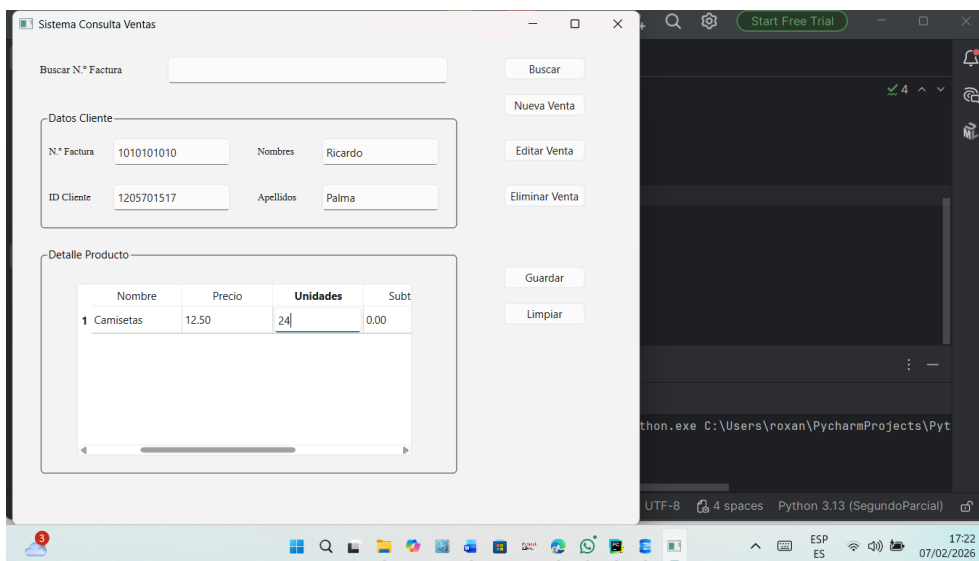
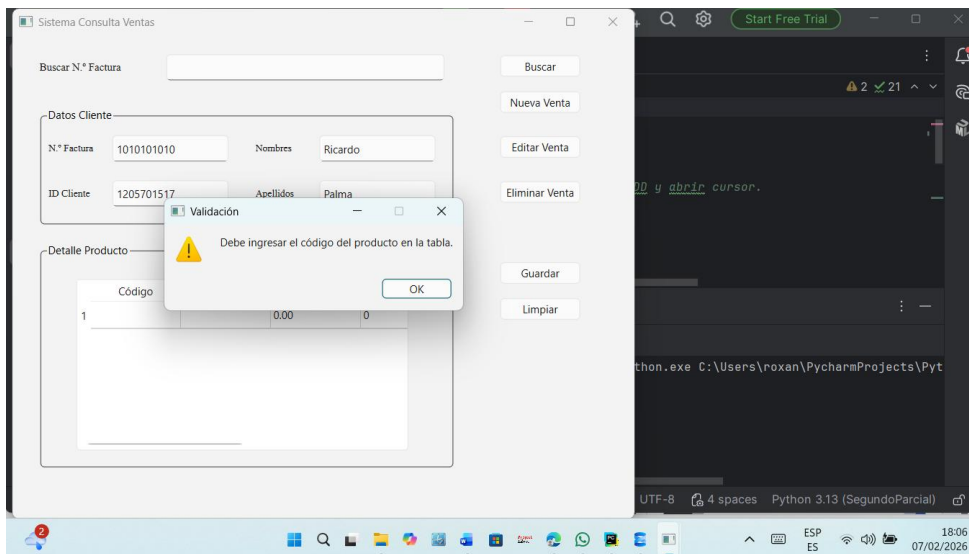
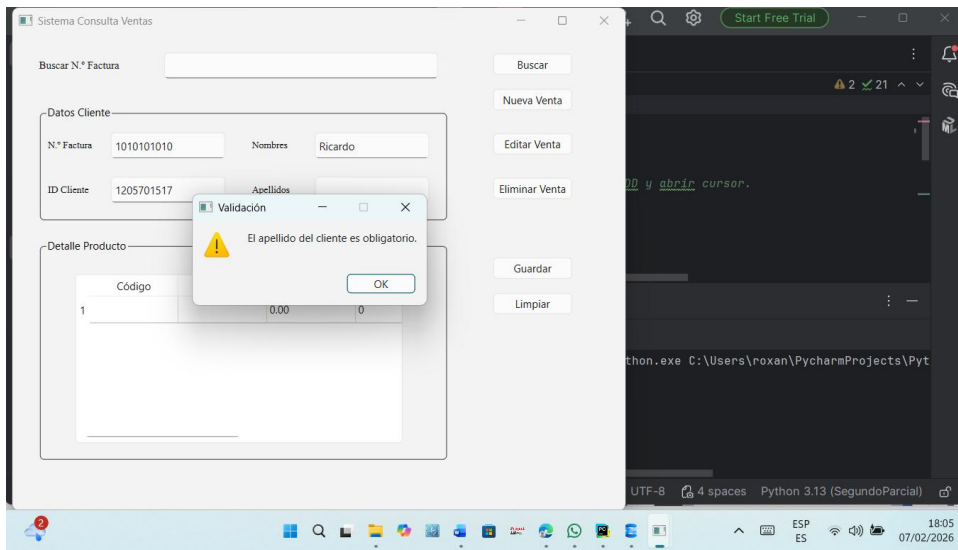
```
9 class Venta:
10     """
11     Clase que representa una transacción
12    """
13
14    def __init__(self, id_venta: str, cliente: Cliente, producto: ProductoBase):
15        self._id_venta = id_venta
16        self._cliente = cliente
17        self._producto = producto
18
19    @property
20    def id_venta(self):
21        return self._id_venta
22
23    @property
24    def cliente(self):
25        return self._cliente
26
27    @property
28    def producto(self):
```

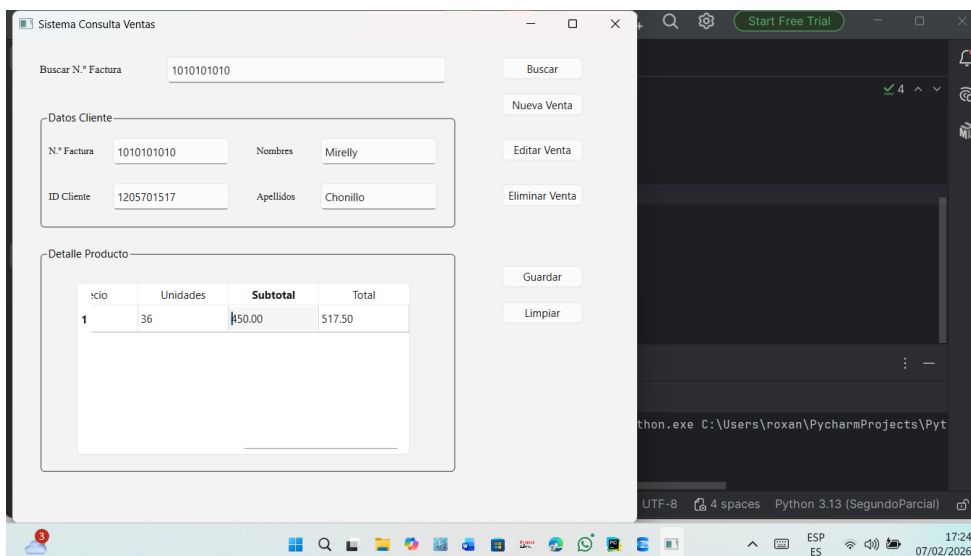
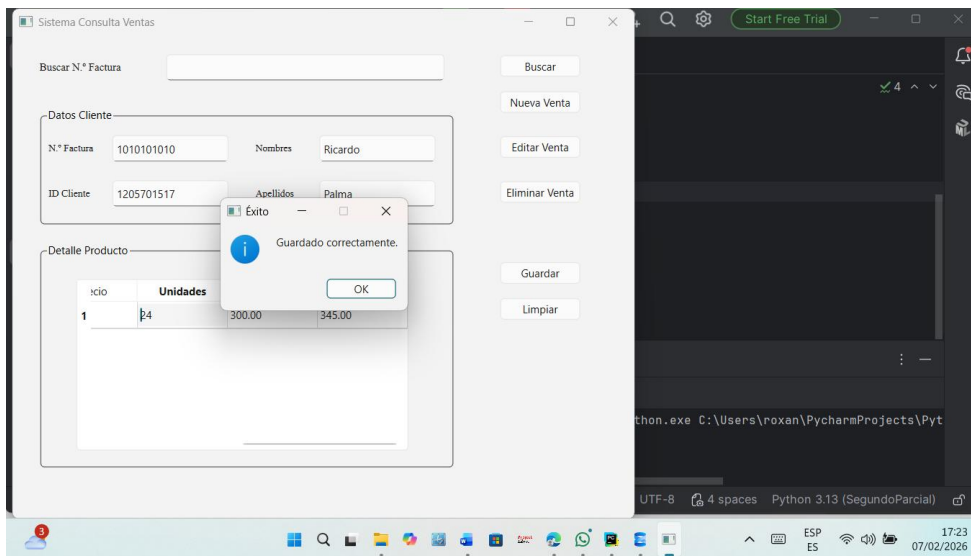
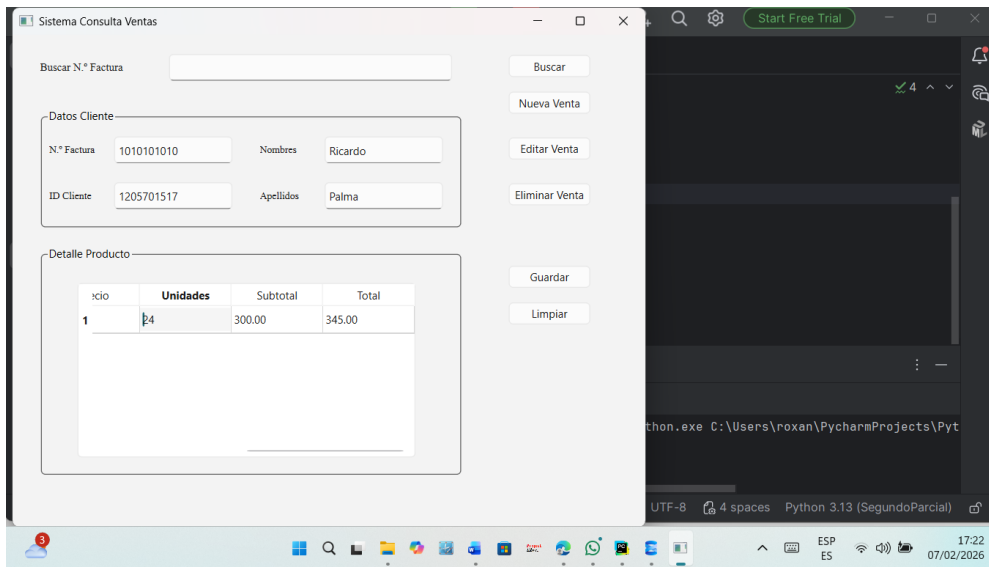


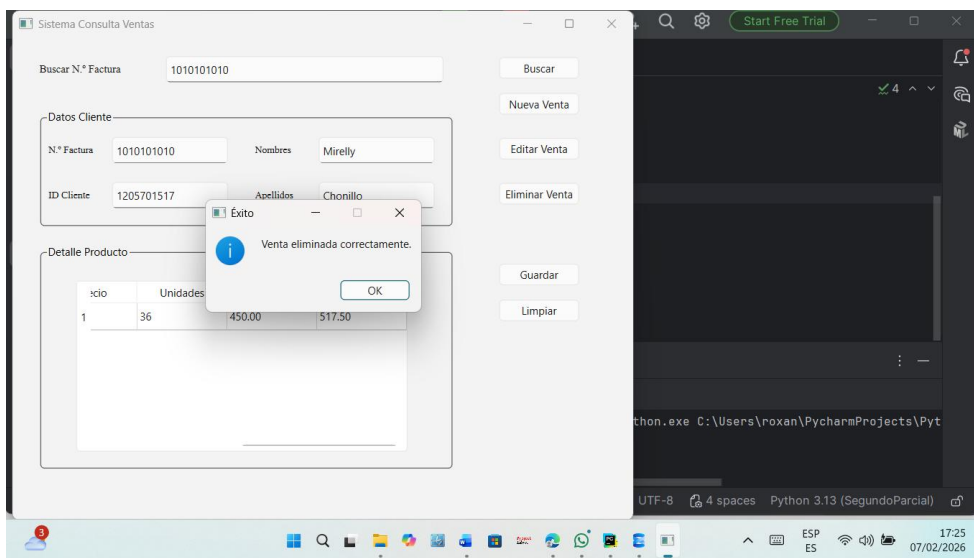
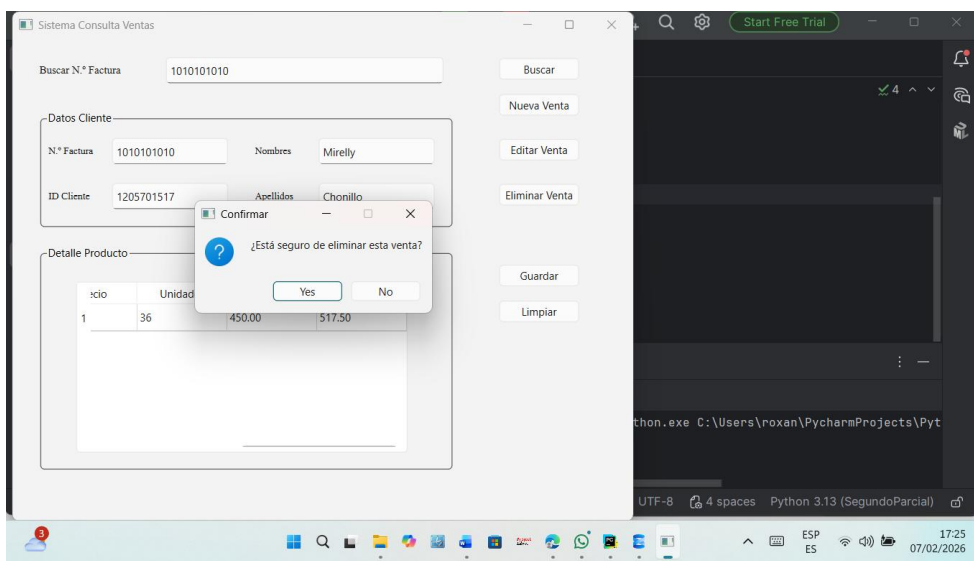
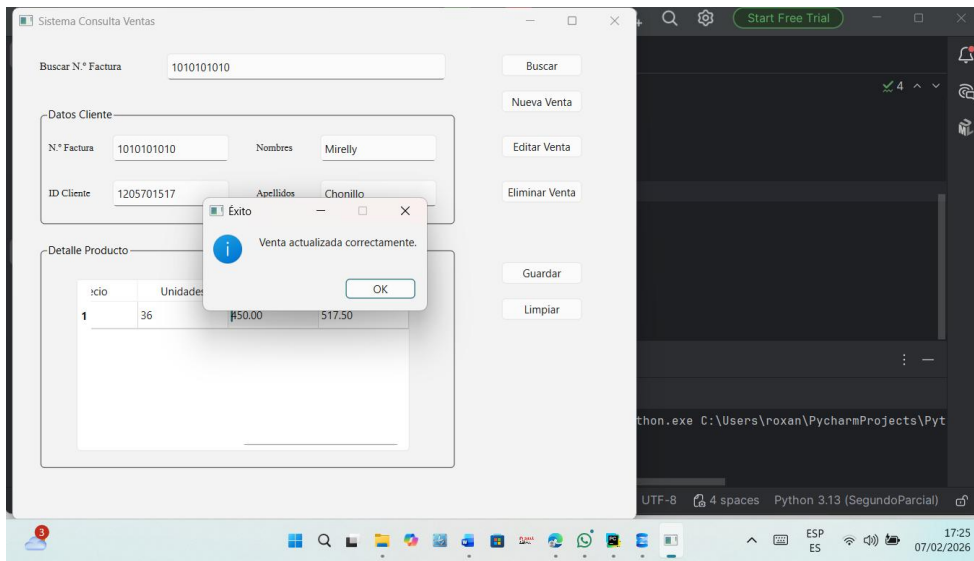
```
8 from Dominio.cliente import Cliente
9 from Dominio.producto import ProductoBase
10 from Dominio.venta import Venta
11 from Datos.venta_DAO import VentaDAO
12
13 class VentaServicio(QMainWindow, Ui_interfaz):
14     """
15     Clase que genera la logica de los objetos de tipo servicio
16    """
17
18    def __init__(self):
19        super(VentaServicio, self).__init__()
20        self.ui = Ui_interfaz()
21        self.ui.setupUi(self)
22
23        self.ui.DetalleFactura.itemChanged.connect(self.calcular_totales_tabla)
24
25        self.ui.btnGuardar.clicked.connect(self.guardar)
26        self.ui.btnLimpiar.clicked.connect(self.limpiar)
27        self.ui.btnCrear.clicked.connect(self.crear)
```











Link del video con la explicación de las clases:

<https://1drv.ms/v/c/420e760fd39caeac/IQDmYnk9Ef38SKo8wT1HKw8CAZ3XpyAh3z72q1FhAOm2DbY?e=4c8iem>

Nota:

En algunas capturas coincide la hora porque me estaba olvidando de esas capturas y tuve que volver a ejecutar para sacar las capturas que me hacían falta.

Debido a que el video solo podía tener 2 minutos tuve que sacar mas capturas para completar con el requerimiento