

Modelo de Casos de Uso (Diagrama de Casos de Uso)



Caso de Uso: Venta Productos

Descripción: Realizar una venta de productos a un cliente

Actores: Encargado de Ventas (Principal), Cliente (Secundario).

Resumen: El Encargado de Ventas registra en el sistema una venta de productos a un cliente

Escenario principal:

1. El *Encargado de Ventas* comienza el alta de una venta introduciendo el número de la venta (*w_nº venta*).
2. El *Sistema* comprueba que no existe ya en el sistema una venta que tenga el *nºventa* introducido e inicia el registro de la venta.
3. El *Encargado de Ventas* introduce la fecha de la venta (*w_fecha*).
4. El *Encargado de Ventas* introduce el cliente de la venta (*w_cliente*).
5. El *Sistema* comprueba que el cliente existe y registra los datos generales de la venta (*w_fecha* y *w_cliente*).
6. El *Encargado de Ventas* introduce los datos de uno de los productos de la venta (*w_producto*, *w_cantidad*).
7. El *Sistema* comprueba que el producto existe y que hay stock suficiente del mismo.
8. El *Sistema* registra el producto de la venta (*w_producto*, *w_cantidad* y *precio* del producto cuando se realiza la venta)
9. El *Sistema* decrementa el *stock-disp* del producto

Los pasos 6, 7 y 8 se repiten hasta que el *Encargado de Ventas* indique que la venta ha finalizado.

Caso de Uso: Venta Productos

Escenarios alternativos:

2a. La venta ya existe.

1. El Sistema indica el error y solicita un nuevo n° de venta (paso 2).

5a. El cliente no existe.

1. El Sistema indica el error y solicita un nuevo cliente (paso 5).

7a. El producto vendido no existe.

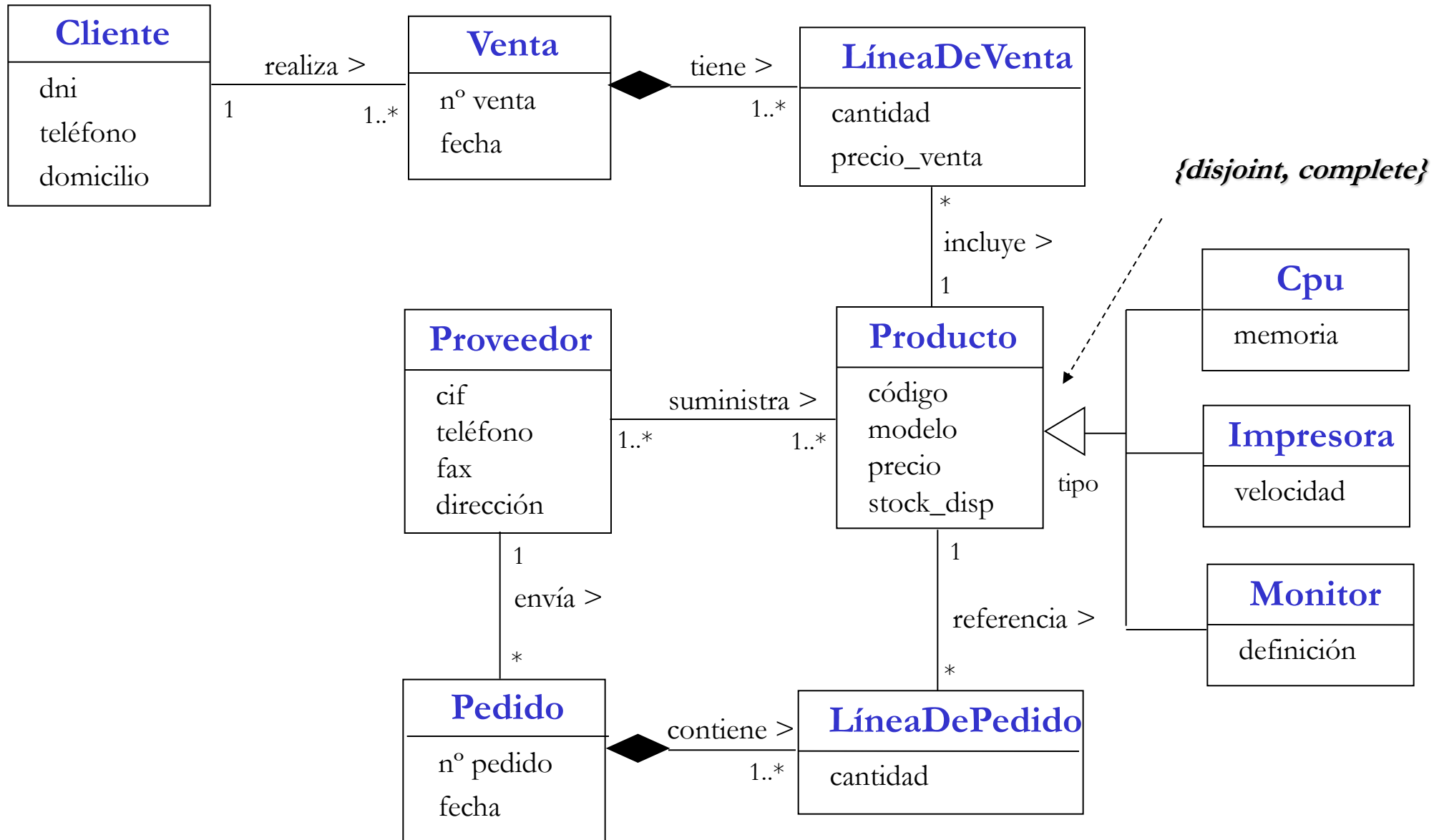
1. El Sistema indica el error y solicita un nuevo producto (paso 6).

7b. No existe stock suficiente del producto vendido.

1. El Sistema indica el error y solicita un nuevo producto (paso 6).

*a En cualquier momento el *Encargado de Ventas* puede cancelar la venta de productos.

Modelo Conceptual de Datos



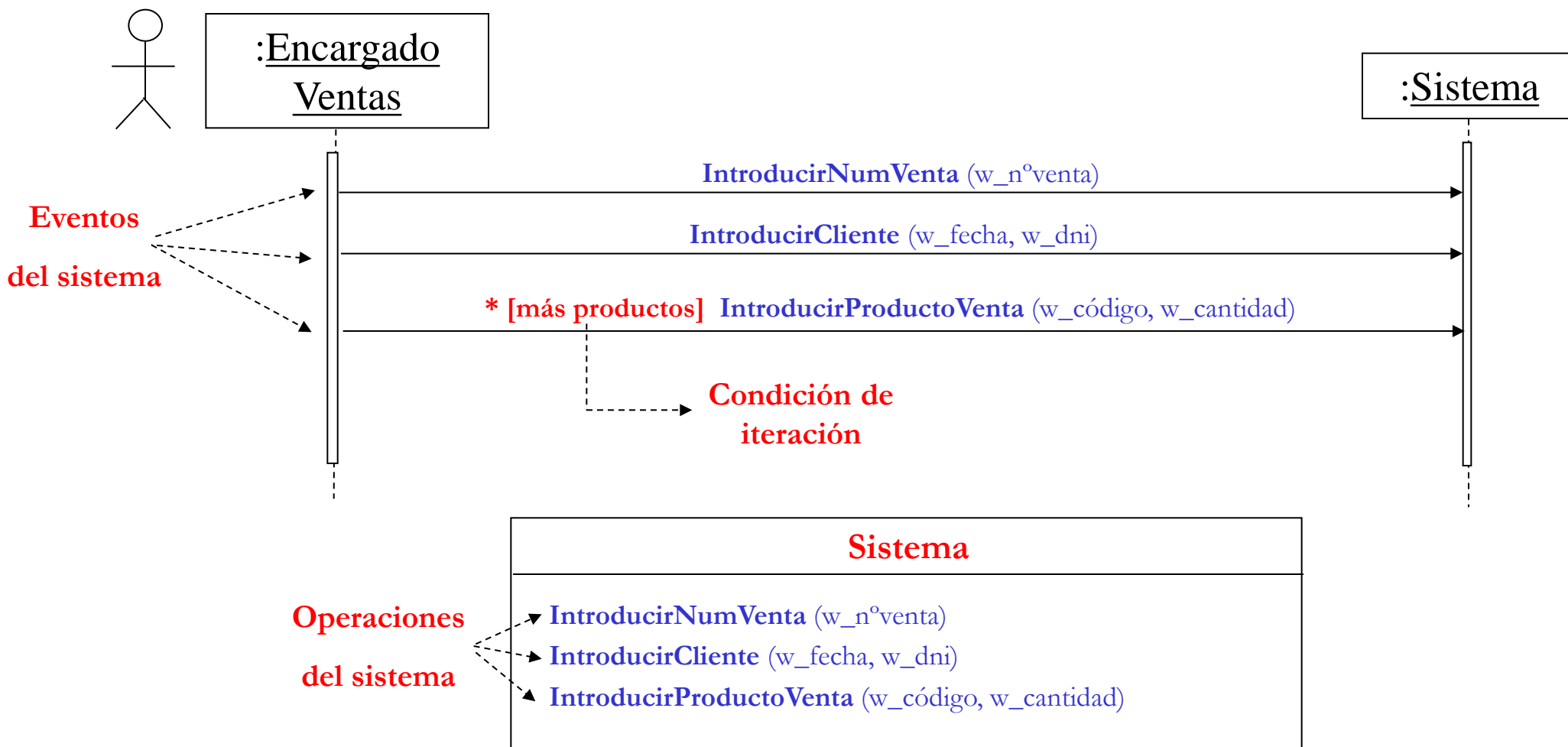
Modelo Conceptual de Datos

Restricciones de Integridad:

- **Claves de clases:** (Cliente, dni), (Venta, n° venta), (Proveedor, cif), (Pedido, n°pedido)
(Producto, código)
- En una misma Venta, no puede haber dos Líneas de Venta del mismo Producto
- En un mismo Pedido, no puede haber dos Líneas de Pedido del mismo Producto
- El proveedor de un pedido, ha de ser un proveedor de los productos del pedido

Modelo de Comportamiento del Sistema (DSS)

Caso de Uso: Venta Productos (Escenario Principal)



Modelo de Comportamiento del Sistema (Contratos)

Operación: IntroducirNumVenta ($w_n^{\circ}venta$)

Responsabilidades: Iniciar el registro de la venta en curso.

Referencias cruzadas: **Caso de Uso:** Venta Productos.

Precondiciones: No existe ninguna Venta con $n^{\circ}venta = w_n^{\circ}venta$.

Postcondiciones:

- Se creó una instancia V de Venta (creación de objeto).
- Se asignó $w_n^{\circ}venta$ a $V.n^{\circ}venta$

Modelo de Comportamiento del Sistema (Contratos)

Operación: IntroducirCliente (w_fecha , w_dni)

Responsabilidades: Registrar los datos generales de la venta.

Referencias cruzadas: Caso de Uso: Venta Productos.

Precondiciones:

- Existe una Venta V en curso.
- Existe un cliente C con $C.dni = w_dni$.

Postcondiciones:

- Se asignó w_fecha a $V.fecha$ (modificación de atributo).
- Se asoció la Venta V con el Cliente C (creación de enlace).

Modelo de Comportamiento del Sistema (Contratos)

Operación: IntroducirProductoVenta ($w_código$, $w_cantidad$)

Responsabilidades: Registrar los productos de la venta

Referencias cruzadas: Caso de Uso: Venta Productos

Precondiciones:

- Existe una Venta V en curso.
- Existe un Producto P con $P.código = w_código$ y $P.stock_disp \geq w_cantidad$.

Postcondiciones:

- Se creó una instancia Ldv de LíneaDeVenta en la Venta V (creación de objeto).
- Se asignó $w_cantidad$ a $Ldv.cantidad$ (modificación de atributo)
- Se asignó $P.precio$ a $Ldv.precio_venta$ (modificación de atributo)
- Se asoció Ldv con P (creación de enlace)
- Se decrementó el $stock_disp$ de P ($P.stock_disp = P.stock_disp - w_cantidad$) (modificación de atributo)

- Al entrar en la aplicación se muestra el menú principal de la misma:

Gestión Almacén
Gestión Ventas

- Al seleccionar la opción Gestión Almacén, se muestra el menú correspondiente. Las opciones de este menú son los casos de uso de la parte de Gestión Almacén:

Gestión Inventario
Gestión Proveedores
Listado Proveedores-Producto
Pedidos a Proveedores

- Al seleccionar la opción Gestión Ventas, se muestra el menú correspondiente. Las opciones de este menú son los casos de uso de la parte de Gestión Ventas:

Gestión Clientes
Ventas a Clientes

Diagrama de Interacción Menú Principal

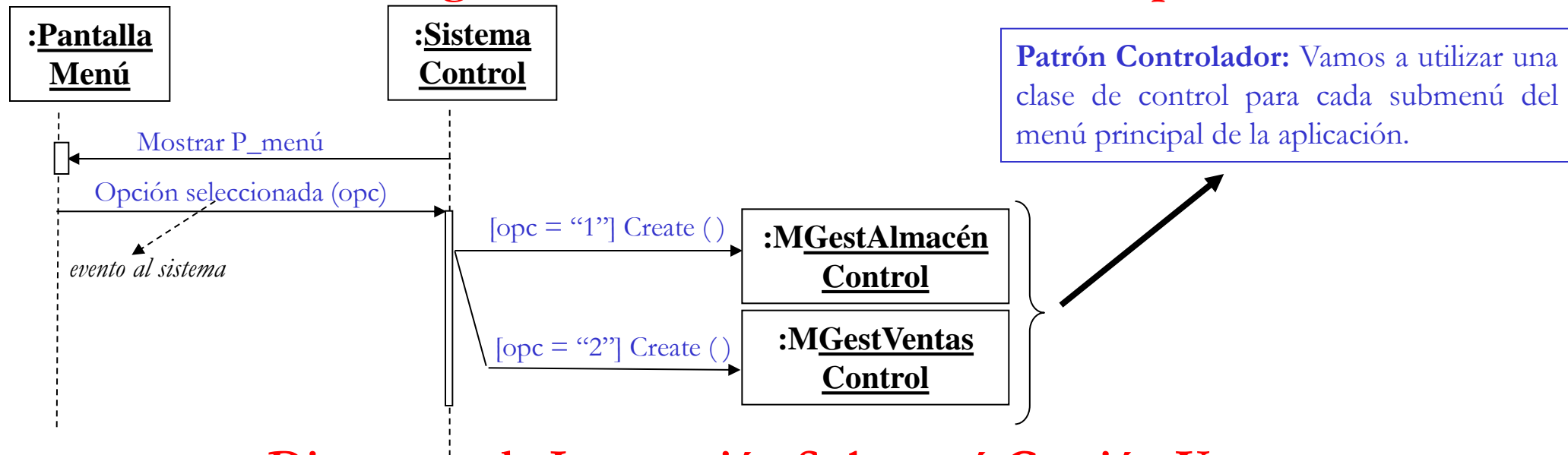


Diagrama de Interacción Submenú Gestión Ventas

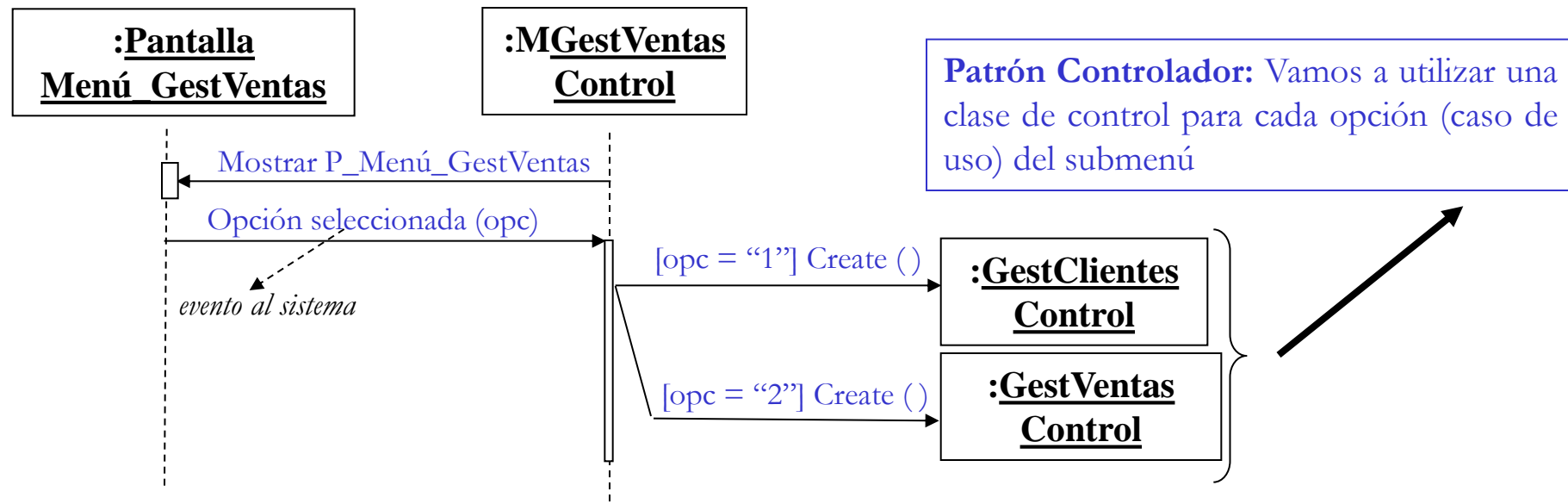
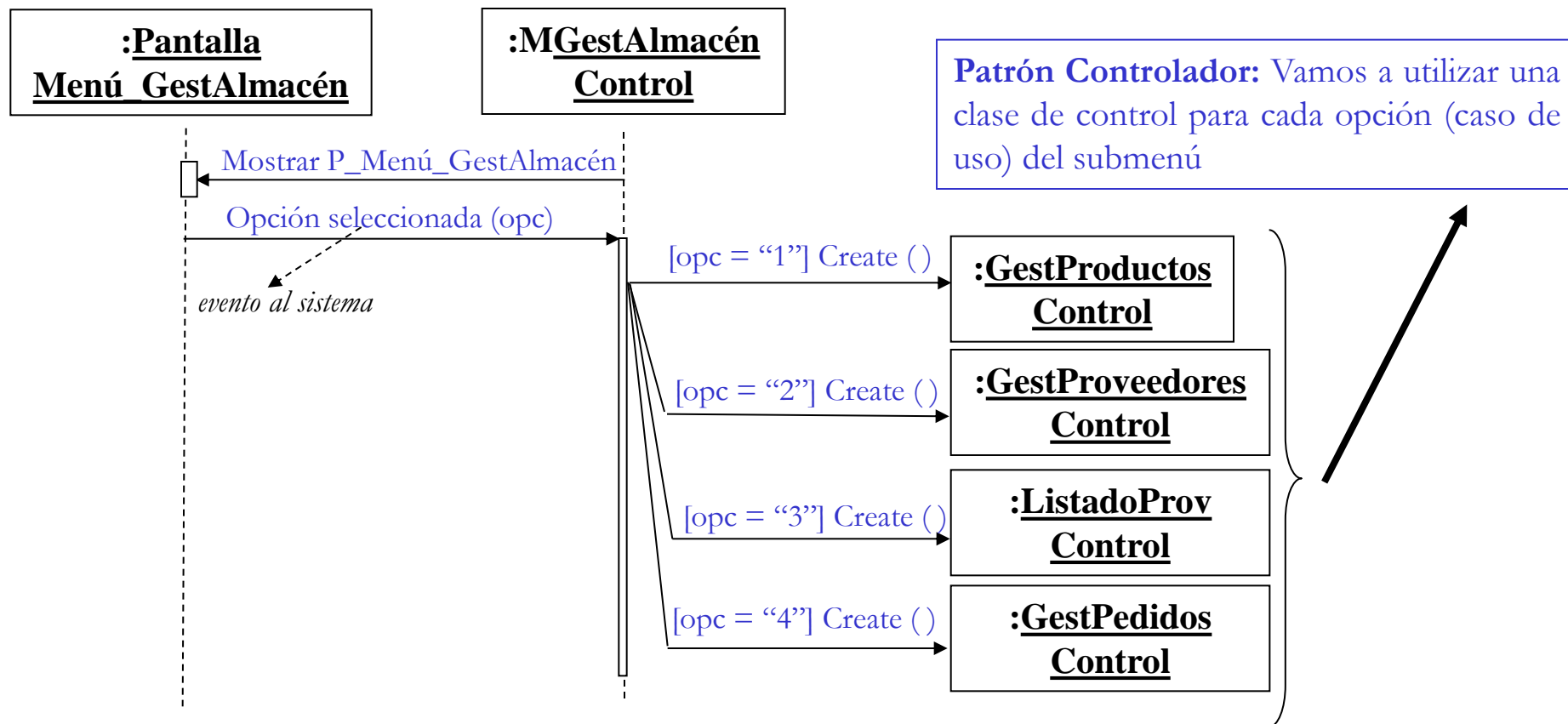


Diagrama de Interacción Submenú Gestión Almacén



Explicación Diagramas de Interacción para gestionar los menús

- Para ejecutar la aplicación, se crea un objeto de la clase *Sistema Control* (*new.SistemaControl*) y se ejecuta una operación de la clase *Sistema Control* que invoca a la operación *Mostrar P-menú* de la clase *Pantalla Menú*. La operación *Mostar P-menú* se encarga de mostrar la pantalla del menú de la aplicación (en realidad esta operación invocará a otras operaciones de las clases de la parte de interfaz de usuario para mostrar la pantalla con el menú de la aplicación)
- Una vez mostrada la pantalla del menú de opciones, se está a la espera de que se produzca algún evento. En este caso el evento que se puede producir es que el usuario seleccione una determinada opción del menú (*Opción seleccionada (opc)*).
- En función de la opción que el usuario seleccione, se crea un objeto de la clase de control que controla el submenú seleccionado. **Ejemplo:** Si se selecciona la opción *Gestión Ventas*, se crea un objeto de la clase *MGestVentasControl* (*new.MGestVentasControl*) y se ejecuta una operación definida en esta clase que le manda un mensaje a la Interfaz de Usuario (simplificada en la clase *Pantalla Menú_GestVentas*) para que muestre la pantalla del *Menú de Gestión Ventas*.
- En función de la opción del submenú que el usuario seleccione, se crea un objeto de la clase de control que controla la acción asociada a la opción (caso de uso) seleccionada. **Ejemplo:** si se selecciona la opción *Ventas a Clientes*, se crea un objeto de la clase de control *GestVentasControl* (*new.GestVentasControl*) y se ejecuta una operación definida en esta clase que le manda un mensaje a la Interfaz de Usuario (simplificada en la clase *Pantalla GestVentas*) para que muestre la pantalla de Ventas de Productos a Clientes.

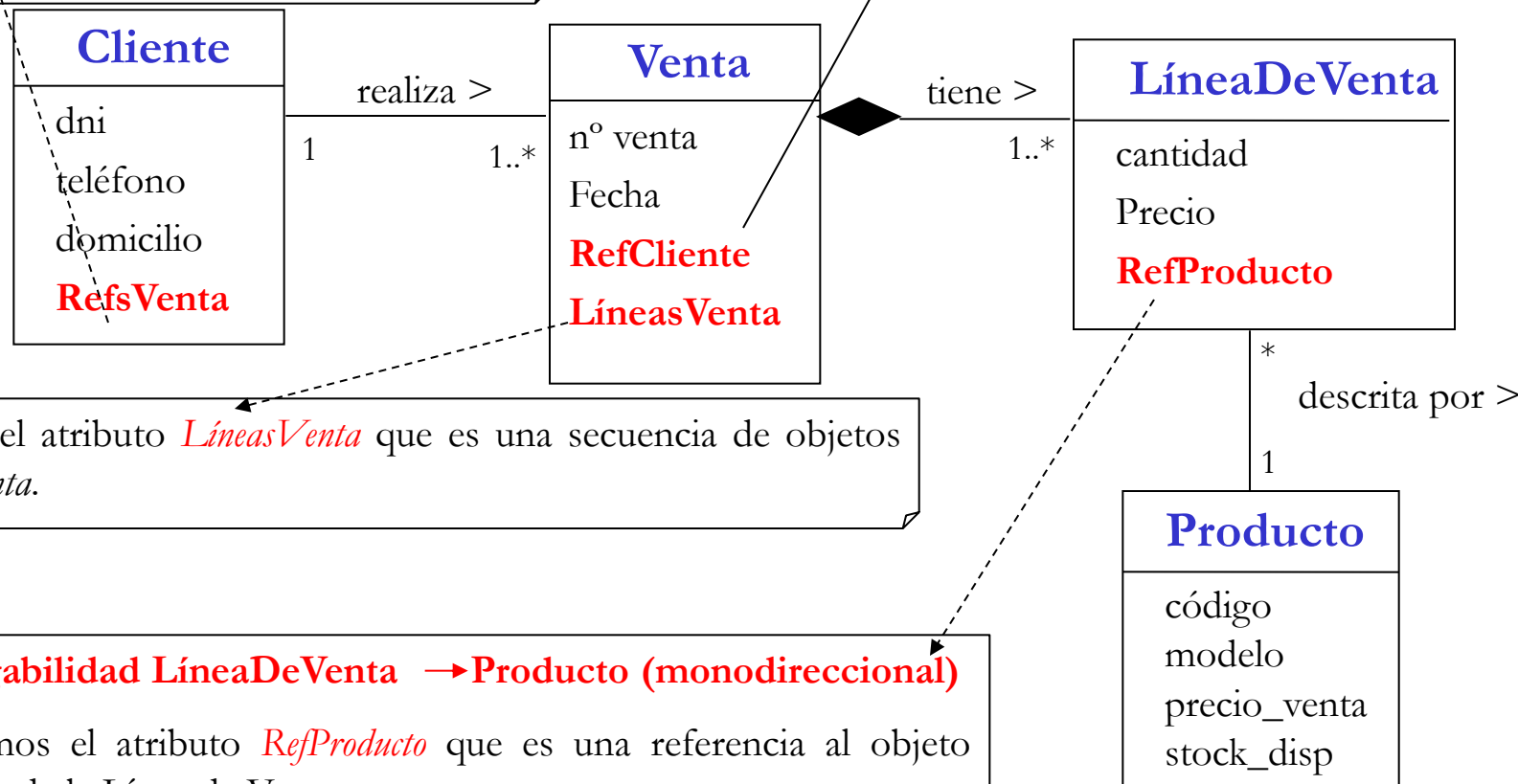
Diseño de las relaciones entre las clases

Navegabilidad Cliente → Venta

Añadimos el atributo *RefsVenta* que es un conjunto de referencias a los objetos *Venta* que realiza el Cliente.

Navegabilidad Venta → Cliente

Añadimos el atributo *RefCliente* que es una referencia al objeto *Cliente* que realiza la venta.



Navegabilidad LíneaDeVenta → Producto (monodireccional)

Añadimos el atributo *RefProducto* que es una referencia al objeto *Producto* de la Línea de Venta

Cambios Contratos de las Operaciones del Sistema

Operación: IntroducirNumVenta ($w_n^{\circ}venta$)

Sin cambios

Operación: IntroducirCliente (w_fecha , w_dni)

Precondiciones: Sin cambios

- Existe una Venta V en curso.
- Existe un cliente C con $C.dni = w_dni$.

Postcondiciones:

- Se asignó w_fecha a $V.fecha$ (modificación de atributo). Sin cambio
- Se crea un enlace de la asociación “realiza” de la Venta V con el Cliente C
- **Cambia:**
 - Se añade la Referencia de V a $C.RefsVenta$
 - $V.RefCliente = Referencia de C$.

Cambios Contratos de las Operaciones del Sistema

Operación: IntroducirProductoVenta ($w_código$, $w_cantidad$)

Precondiciones: Sin cambios

- Existe una Venta V en curso.
- Existe un Producto P con $P.código = w_código$ y $P.stock_disp \geq w_cantidad$.

Postcondiciones: Cambia

- Se creó una instancia Ldv de LíneaDeVenta (creación de objeto).
- Se asignó $w_cantidad$ a $Ldv.cantidad$ (modificación de atributo)
- Se asignó $P.precio$ a $Ldv.precio_venta$ (modificación de atributo)
- Se asignó Referencia de P a $Ldv.RefProducto$ (creación de enlace)
- Se añade Ldv a $V.LineasVenta$ (se guarda el objeto línea de venta Ldv en la venta V)
- Se decrementó el $stock_disp$ de P ($P.stock_disp = P.stock_disp - w_cantidad$) (modificación de atributo)

Diagrama de Interacción CU: Escenario Principal Venta Productos

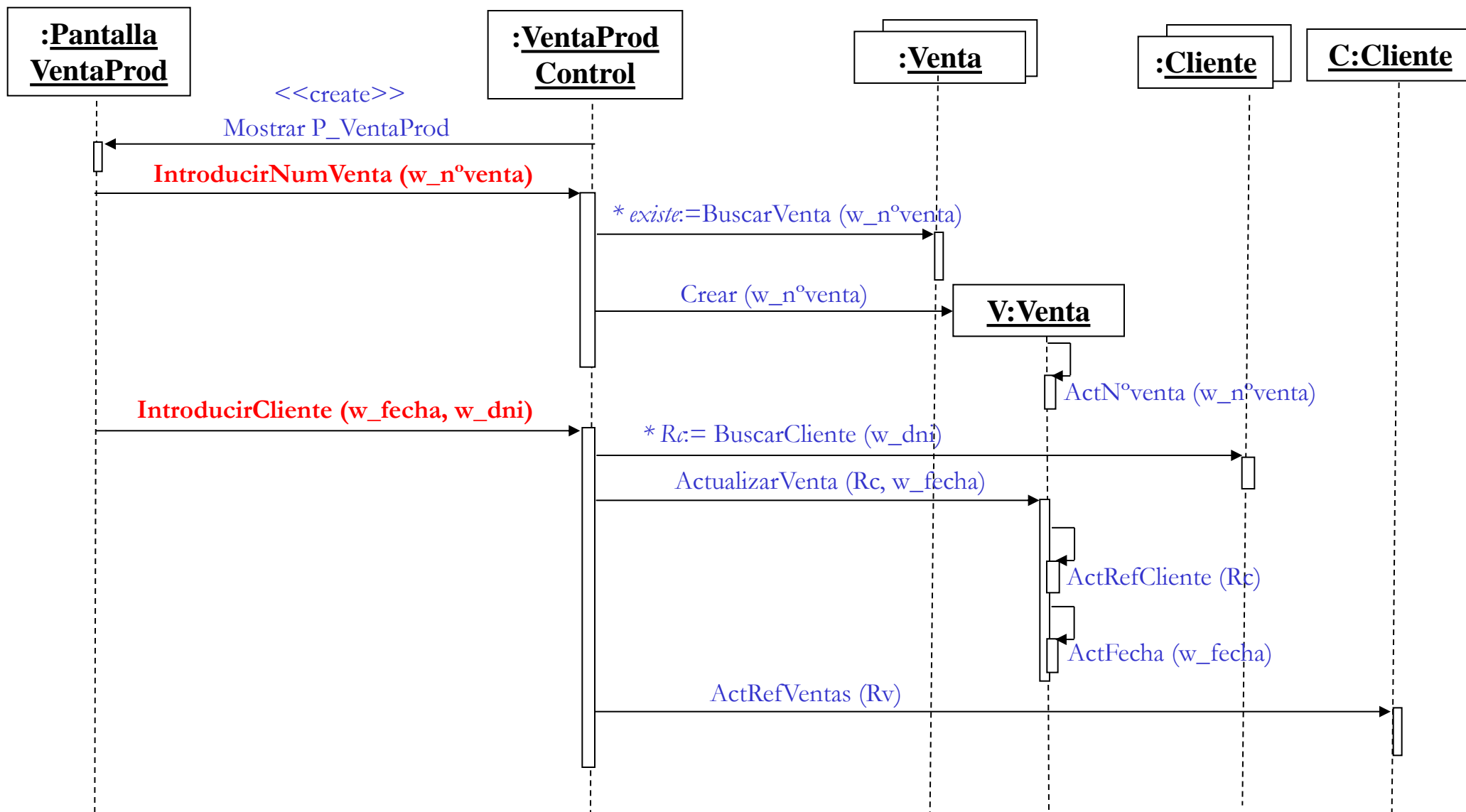


Diagrama de Interacción CU: Venta Productos

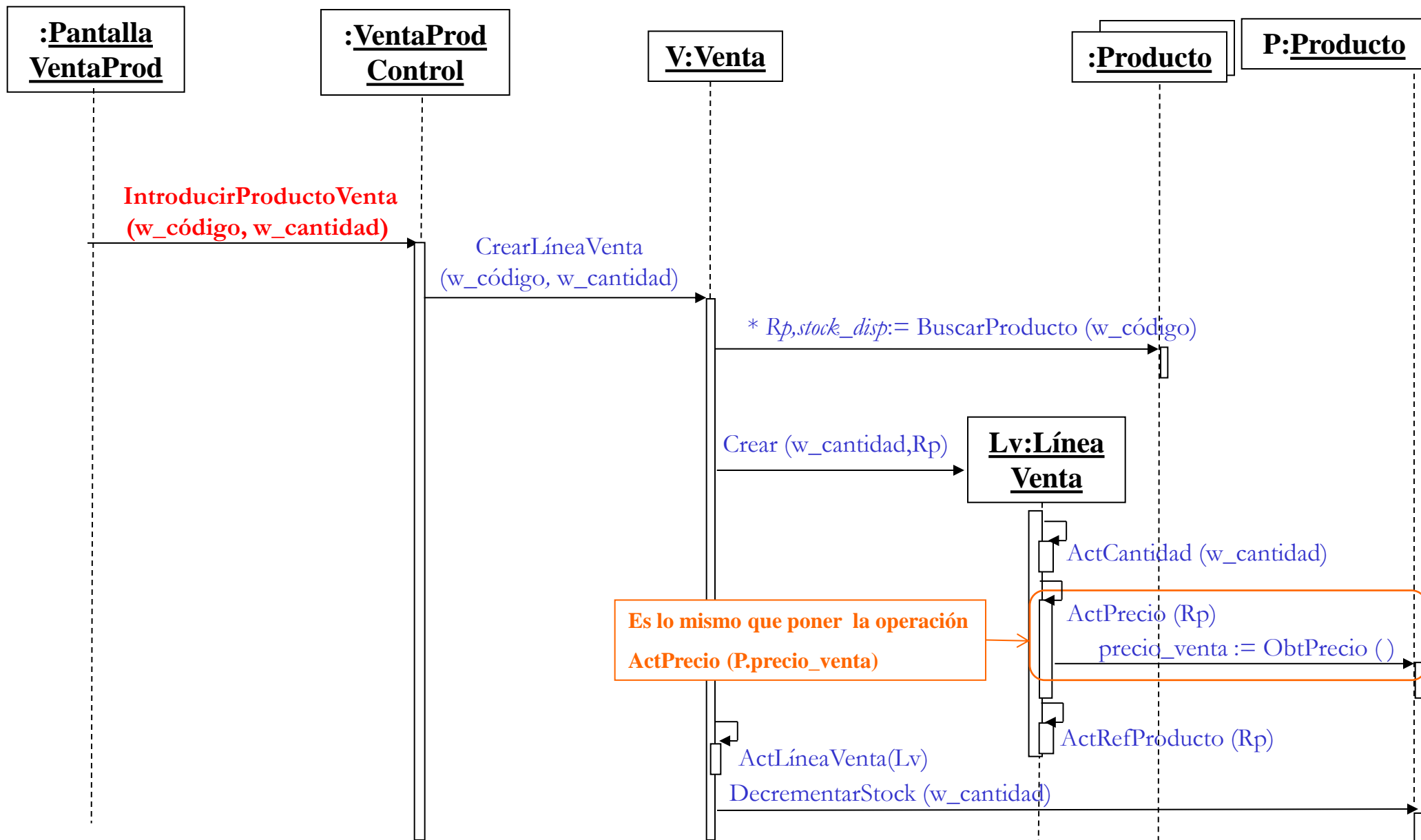


Diagrama de Clases de Diseño (Clases)

- De los diagramas de interacción que hemos hecho anteriormente, se obtienen nuevas clases, nuevas asociaciones y operaciones a añadir al diagrama de clases conceptual que hemos hecho en el análisis del sistema.
- **Clases nuevas:** clases de control y clases de la interfaz de usuario (simplificadas en clases Pantalla-X):
 - *Pantalla Menú*
 - *Sistema Control*
 - *MGestAlmacén Control*
 - *MGestVentas Control*
 - *Pantalla Menú_GestVentas*
 - *Pantalla Menú_GestAlmacén*
 - *GestProductos Control*
 - *GestProveedores Control*
 - *ListadoProv Control*
 - *GestPedidos Control*
 - *Pantalla VentaProd*
 - *VentaProd Control*

Diagrama de Clases de Diseño (Asociaciones)

- Colaboraciones entre objetos:

- *Pantalla Menú y Sistema Control*
- *Sistema Control y MGestAlmacén Control*
- *Sistema Control y MGestVentas Control*
- *Pantalla Menú_GestVentas y MGestVentas Control*
- *MGestVentas Control y GestClientes Control*
- *MGestVentas Control y GestVentas Control*
- *Pantalla Menú_GestAlmacén y MGestAlmacén Control*
- *MGestAlmacén Control y GestProductos Control*
- *MGestAlmacén Control y GestProveedores Control*
- *MGestAlmacén Control y ListadoProv*
- *MGestAlmacén Control y GestPedidos Control*
- *Pantalla VentaProd y VentaProd Control*
- *VentaProd Control y Venta*
- *VentaProd Control y Producto*
- *Venta y Cliente*
- *Venta y Producto*
- *Venta y Línea de Venta*
- *Línea de Venta y Producto*

Diagrama de Clases de Diseño (Operaciones)

- **Clase Sistema Control**

- Operación que se ejecuta al hacer *new.SistemaControl*. Esta operación invoca a la operación *Mostrar P-menú* de la clase *Pantalla Menú*.
- Operación *Opción seleccionada (opc)*. Invoca a la operación *Create ()* de la clase de control asociada a la opción seleccionada.

- **Clase Pantalla Menú**

- Operación *Mostrar P-menú*. Esta operación muestra la pantalla del menú principal de la aplicación.

- **Clase MGestAlmacén Control**

- Operación que se ejecuta al hacer *new.MGestAlmacénControl*. Esta operación invoca a la operación *Mostrar P_MGestAlmacén* de la clase *Pantalla Menú_GestAlmacén*.
- Operación *Opción seleccionada (opc)*. Invoca a la operación *Create ()* de la clase de control asociada a la opción seleccionada.

- **Clase MGestVentas Control**

- Operación que se ejecuta al hacer *new.MGestVentasControl*: Invoca a la operación *Mostrar P_MGestVentas* de la clase *Pantalla Menú_GestVentas*.
- Operación *Opción seleccionada (opc)*: Invoca a la operación *Create ()* de la clase de control asociada a la opción seleccionada.

- **Clase Pantalla Menú_GestAlmacén**

- Operación *Mostrar P-menú*: Muestra la pantalla del menú de gestión almacenes.

- **Clase Pantalla Menú_GestVentas**

- Operación *Mostrar P-menú*: Muestra la pantalla del menú de gestión ventas.

- **Clase VentaProd Control**

- Operación que se ejecuta al hacer *new.VentaProdControl*: Invoca a la operación *Mostrar P_VentaProd* de la clase *Pantalla VentaProd*.
- Operación *Introducir NumVenta (w_nºventa)*:
 - Invoca a la operación ** existe:= BuscarVenta (w_nºventa)* de Venta.
 - Invoca a la operación *Crear (w_nºventa)* de Venta.

- **Clase VentaProd Control (cont)**

- Operación *IntroducirCliente* (w_fecha , w_dni):
 - Invoca a la operación $Rc := \text{BuscarCliente}(w_dni)$ de Cliente.
 - Invoca a la operación *ActualizarVenta* (Rc , w_fecha) del objeto Venta V .
 - Invoca a la operación *ActRefVentas* (Rv) del objeto Cliente C .
- Operación *IntroducirProductoVenta* ($w_código$, $w_cantidad$):
 - Invoca a la operación *CrearLíneaVenta* ($w_código$, $w_cantidad$) del objeto venta V .

- **Clase Pantalla VentaProd**

- Operación *Mostrar P_VentaProd*. Muestra la pantalla de venta de productos.

- **Clase Venta**

- Operación * *existe:=BuscarVenta (w_nºventa)*: Comprueba que no existe una venta con nº venta = w_nºventa.
- Operación *Crear (w_nºventa)*:
 - Crea un objeto *V* de la clase Venta (constructor).
 - Invoca a la operación *ActNºventa (w_nºventa)* del objeto venta *V*.
- Operación *ActualizarVenta (Rc, w_fecha)*:
 - Invoca a la operación *ActRefCliente (Rc)* del objeto venta *V*.
 - Invoca a la operación *ActFecha (w_fecha)* del objeto venta *V*.

- **Clase Venta (cont)**
 - Operación ActRefCliente (Rc)
 - Asigna Rc al atributo RefCliente del objeto venta V (V.RefCliente = Rc)
 - Operación ActFecha (w_fecha)
 - Asigna w_fecha al atributo Fecha del objeto venta V (V.Fecha = w_fecha)
 - Operación CrearLíneaVenta (w_código, w_cantidad)
 - Invoca a la operación ** Rp,stock:= BuscarProducto (w_codigo)* de Producto.
 - Comprueba que *w_cantidad* ≤ *stock* (escenario principal del caso de uso).
 - Invoca a la operación *Crear (w_cantidad, Rp)* del objeto *Lv* de Línea de venta.
 - Invoca a la operación *ActLíneaVenta (Lv)* del objeto *V* de Venta.
 - Invoca a la operación *DecrementarStock (w_cantidad)* del objeto producto P.

– Clase Línea de Venta

- Operación *Crear* ($w_cantidad$, Rp)
 - Crea un objeto Lv de la clase Línea de Venta (constructor).
 - Invoca a la operación *ActCantidad* ($w_cantidad$) del objeto Línea de Venta Lv .
 - Invoca a la operación *ActPrecio* (Rp) del objeto Línea de Venta Lv .
 - Invoca a la operación *ActRefProducto* (Rp) del objeto Línea de Venta Lv .
- Operación *ActCantidad* ($w_cantidad$)
 - Asigna $w_cantidad$ a $Lv.cantidad$
- Operación *ActPrecio* (Rp)
 - Invoca a la operación $precio_venta := ObtPrecio$ () del objeto Producto P.
 - Asigna $precio_venta$ a $Lv.precio$.
- Operación *ActRefProducto* (Rp)
 - Asigna Rp a $Lv.RefProducto$

– Clase Producto

- Operación $*Rp := \text{BuscarRef } (w_código)$
 - Busca la referencia del objeto producto con $código = w_código$.
- Operación $precio_venta := \text{Obtprecio } ()$
 - Obtiene el valor del atributo $precio$ del objeto producto con referencia Rp (P).
- Operación $\text{DecrementarStock } (w_cantidad)$
 - Decrementa el valor del atributo $stock_disp$ del objeto producto con referencia Rp (P)
($P.stock = stock_disp - w_cantidad$)

– Clase Cliente

- Operación $*Rc := \text{BuscarRef } (w_dni)$: Busca la referencia del objeto cliente cuyo $dni = w_dni$.
- Operación $\text{ActRefVentas } (Rv)$
 - Añade la referencia del objeto Venta V (Rv) al conjunto de referencias a ventas del objeto cliente C ($C.\text{RefVentas}$).