**Trabajo Práctico**

## Caso Práctico – Pizzería

Presentación del Caso de Estudio

Una pizzería de la ciudad ofrece a sus clientes una amplia variedad de pizzas de fabricación propia, de varios tamaños (8, 10 y 12 porciones). Los clientes tienen a disposición un menú que describe para cada una de las variedades, el nombre, los ingredientes y el precio según el tamaño y el tipo (a la piedra, a la parrilla, de molde) de la pizza. Los clientes realizan sus pedidos en el mostrador.

El pedido debe contener el nombre del Cliente, para llamarlo cuando su pedido está listo; la cantidad de pizzas, el tamaño, la variedad, la fecha del pedido, la hora en la que el pedido debe entregarse y la demora estimada informada al cliente.

El pedido va a la cocina y cuando está preparado se informa al que lo tomó para que se genere la factura correspondiente y se le entregue el pedido al cliente.

El dueño de la pizzería ha manifestado la necesidad de acceder al menos a la siguiente información:

* Variedades y tipos de pizzas más pedidas por los clientes.
* Ingresos (recaudaciones) por períodos de tiempo.
* Pedidos (cantidad y monto) por períodos de tiempo.

Modelo de Dominio

***TipoPizza***

*-*

*descripcion*

*-*

*nombre*

***VariedadPizza***

*ingredientes*

*-*

*-*

*nombre*

***TamañoPizza***

*-*

*cantPorciones*

*nombre*

*-*

*«entity»*

***Pedido***

*fechaHoraCreacion*

*-*

*-*

*fechaHoraEntrega*

*-*

*nombreCliente*

*-*

*numero*

*+*

*calcTotalPedido*

*()*

*()*

*cancelar*

*+*

*()*

*confirmar*

*+*

*+*

*esPteFacturacion*

*()*

*()*

*+*

*facturar*

*()*

*getDetallePedido*

*+*

*()*

*getEstado*

*+*

*+*

*()*

*mostrarNombreCliente*

*+*

*mostrarNumero*

*()*

*new*

*()*

*+*

*()*

*+*

*obtenerDetallesPedido*

*()*

*+*

*setEstado*

*+*

*terminar*

*()*

*«entity»*

***DetallePedido***

*cantidad*

*-*

*-*

*precio*

*+*

*calcTotalItem*

*()*

*cancelar*

*()*

*+*

*+*

*()*

*getEstado*

*()*

*+*

*getPizza*

*+*

*mostrarDetallePedido*

*()*

*+*

*setEstado*

*()*

*«entity»*

***Pizza***

*-*

*nombre*

*-*

*precio*

*+*

*()*

*getNombre*

*()*

*getPizza*

*+*

*()*

*+*

*getPrecio*

*getTamañoPizza*

*()*

*+*

*+*

*()*

*getTipoPizza*

*+*

*getVariedadPizza*

*()*

*setNombre*

*+*

*()*

*()*

*setPrecio*

*+*

*«entity»*

***EstadoFactura***

*nombre*

*-*

*()*

*esGenerada*

*+*

*esPteFacturacion*

*+*

*()*

*«entity»*

***Factura***

*fechaHoraEmision*

*-*

*numero*

*-*

*()*

*buscarItemsAFacturar*

*+*

*()*

*calcTotalFactura*

*+*

*()*

*getDetalleFactura*

*+*

*getEstado*

*+*

*()*

*getNumero*

*()*

*+*

*new*

*+*

*()*

*()*

*setEstado*

*+*

*«entity»*

***EstadoPedido***

*nombre*

*-*

*esFacturado*

*+*

*()*

*()*

*esPteFacturacion*

*+*

*pizza*

*+*

*1*

*factura*

*+*

*0..1*

*detallesPedido*

*+*

*1..\**

*tamaño*

*+*

*1*

*+*

*estado*

*1*

*+*

*tipoPizza*

*1*

*estado*

*1*

*+*

*estado*

*1*

*+*

*detalleFactura*

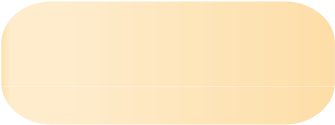
*1..\**

*+*

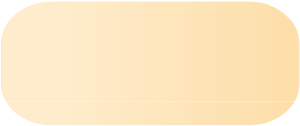
*variedad*

*1*

Diagrama de Máquina de Estado de la Clase Pedido



**PteConfirmación**



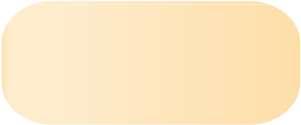
**Cancelado**



**EnPreparación**



**PteFacturación**



**Facturado**

. Cancelar Pedido /cancelar

()

15

10

Registrar Pedido /new

()

20

Informar Cierre

Pedido /terminar()

25

. Confirmar Pedido

/confirmar()

. Generar Factura /facturar

()

22

15

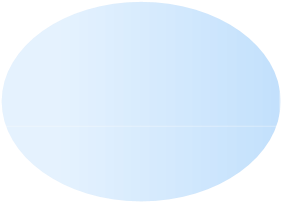
. Cancelar Pedido /cancelar

()

Vista parcial de los casos de uso de la Pizzería

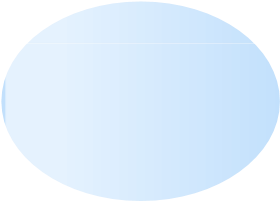


**Vendedor**



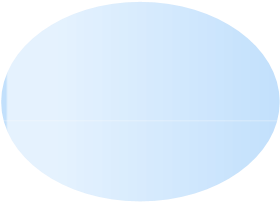
**. Generar Factura**

**22**



**. Registrar Pedido**

**10**

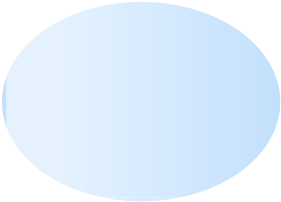


**. Cancelar Pedido**

**15**



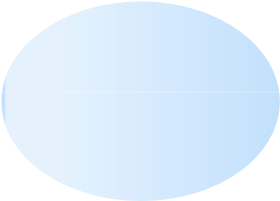
**Responsable de Cocina**



**. Confirmar**

**25**

**Pedido**



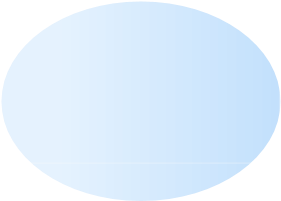
**. Informar Cierre**

**20**

**Pedido**

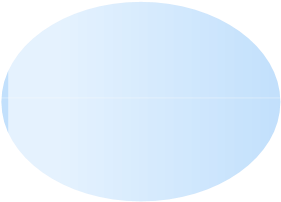


**Responsable Pizzería**



**17**

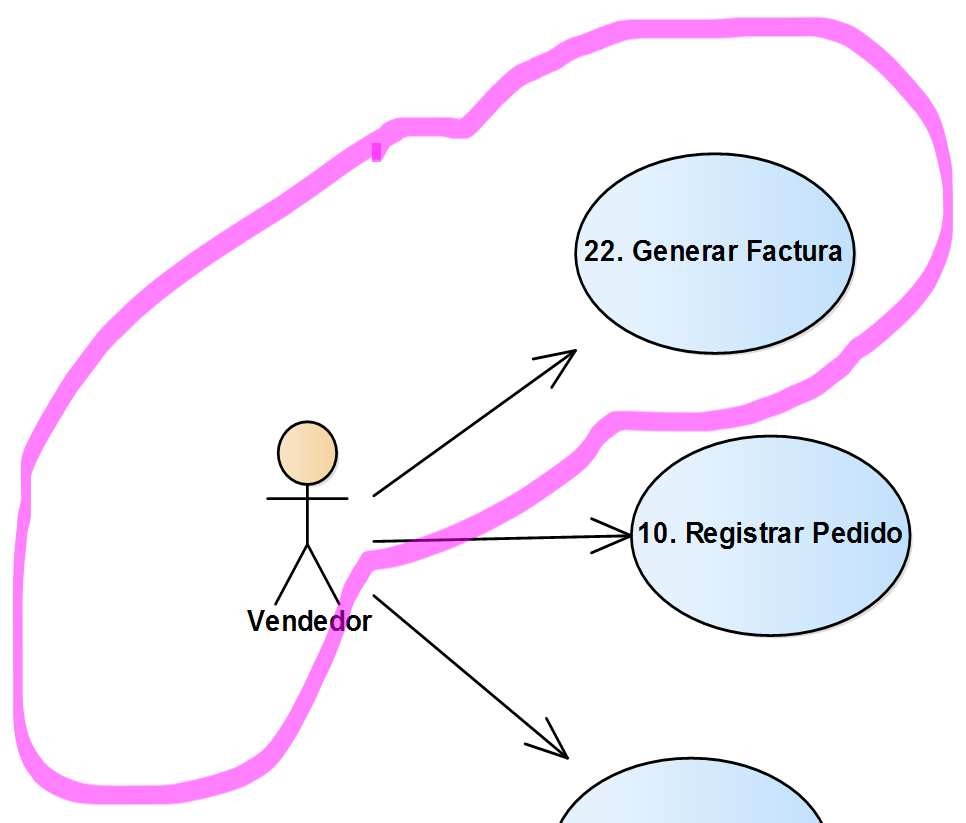
**. Registrar Menú**



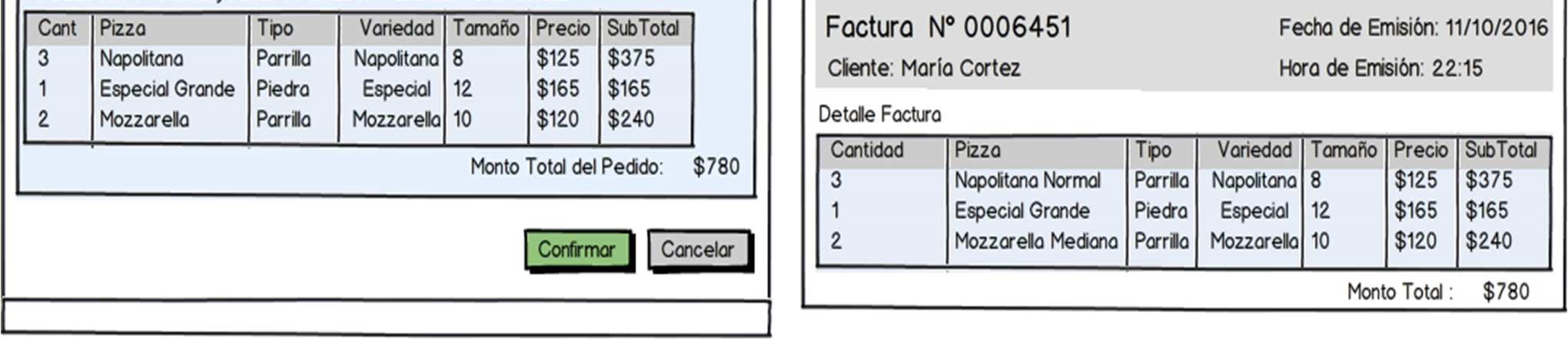
**14**

**. Actualizar**

**Precios**



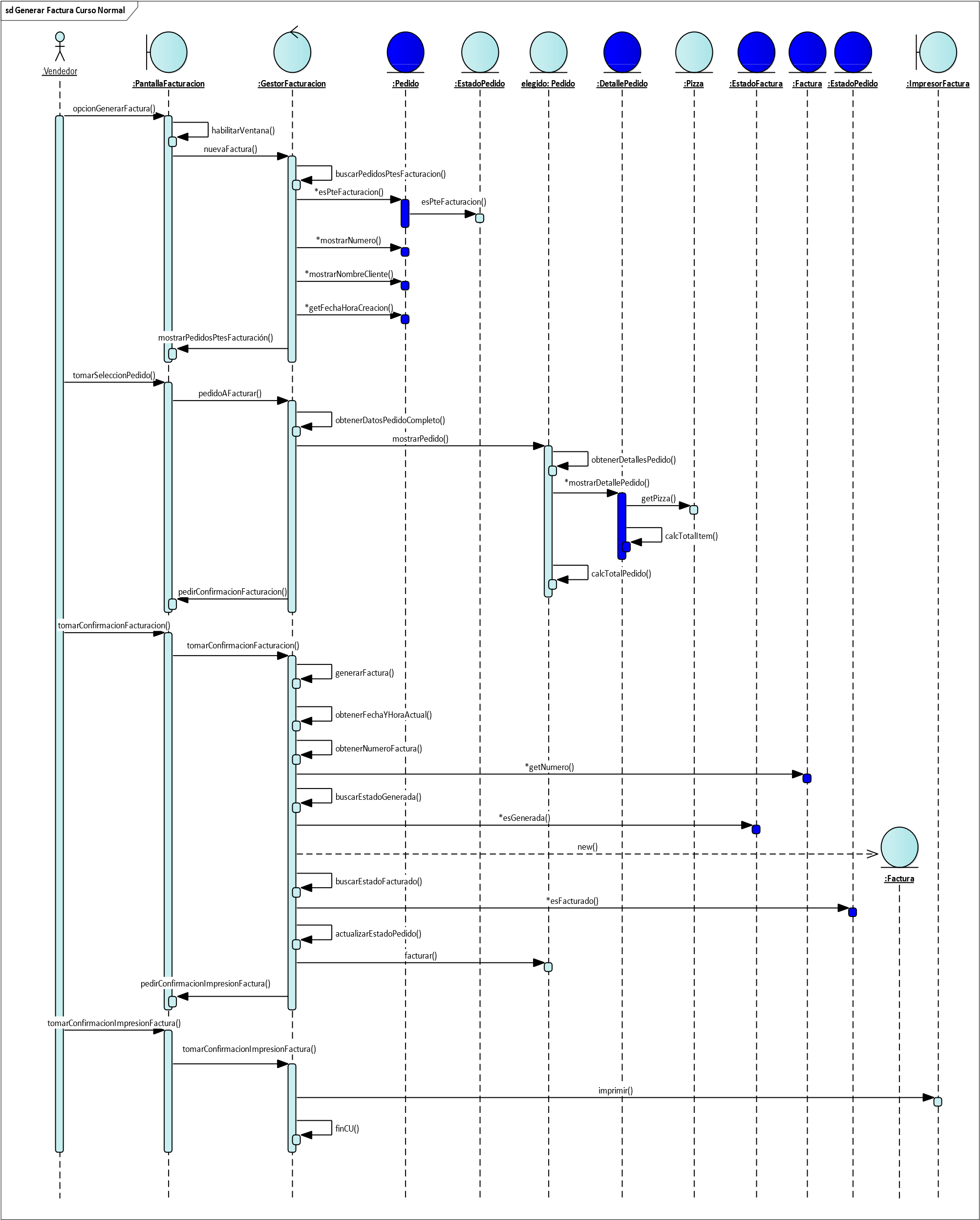
Prototipo de la Interfaz de Usuario para el Caso de Uso 22. Generar Factura



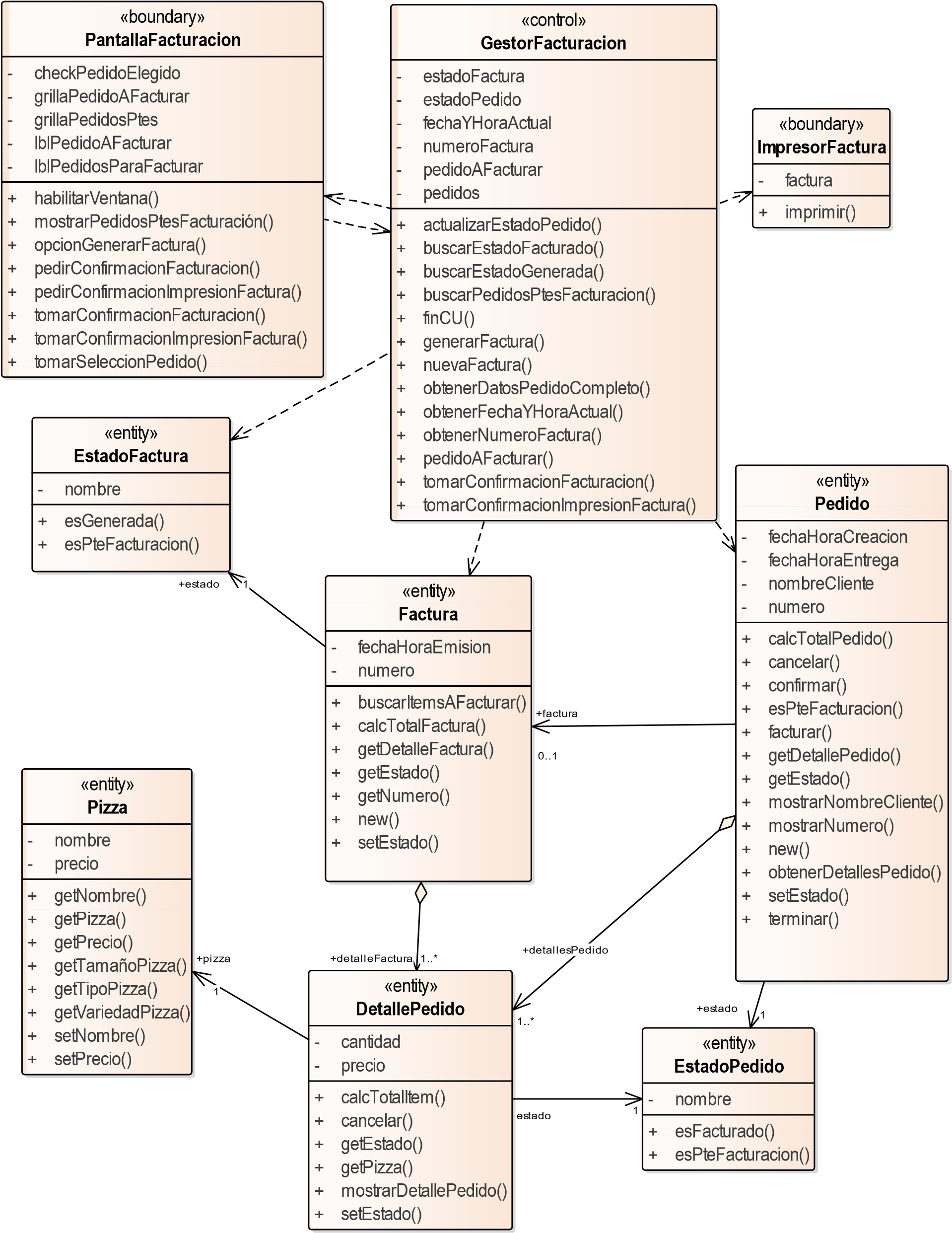
Descripción con resumen esencial del Caso de Uso 2. Generar Factura

|  |  |  |
| --- | --- | --- |
| **Nombre del Caso de uso**: Generar Factura | | **Nro. de Orden**: 2 |
| **Actor Principal**: Vendedor | **Actor Secundario**: no aplica | |
| **Objetivo**: generar la factura asociada a un pedido determinado. | | |
| **Flujo Básico** | | |
| 1.  **Vendedor**: selecciona la opción “**Generar Factura**”. | | |
| 2. **Sistema**: busca y muestra la lista de pedidos que estén listos y no hayan sido facturados aún y solicita se seleccione uno. | | |
| 3. **Vendedor**: selecciona el pedido que desea facturar. | | |
| 4. **Sistema**: muestra los datos completos del pedido y calcula el monto total a cobrar. | | |
| 5. **Sistema**: solicita la confirmación de la generación de la factura. | | |
| 6. **Vendedor:** confirma la generación de la factura. | | |
| 7. **Sistema**: genera la factura en estado “Generada” y actualiza el estado del pedido a “Facturado”. | | |
| 8. **Sistema**: solicita confirmación de la impresión de la factura. | | |
| 9. **Vendedor:** confirma la impresión. | | |
| 10. **Sistema**: imprime la factura. Fin del caso de uso. | | |
| **Flujos Alternativos** | | |
| **A1**: No hay pedidos pendientes de facturación.  **A2:** El vendedor no confirma la generación de la factura.  **A3**: El vendedor NO confirma la impresión. | | |
| **Observaciones**:  1. El Vendedor puede cancelar la ejecución del caso de uso en cualquier momento. | | |

Diagrama de Secuencia del escenario del curso normal del Caso de Uso 22 Generar Factura.



Vista Generar Factura – Diagrama de Clases de Análisis



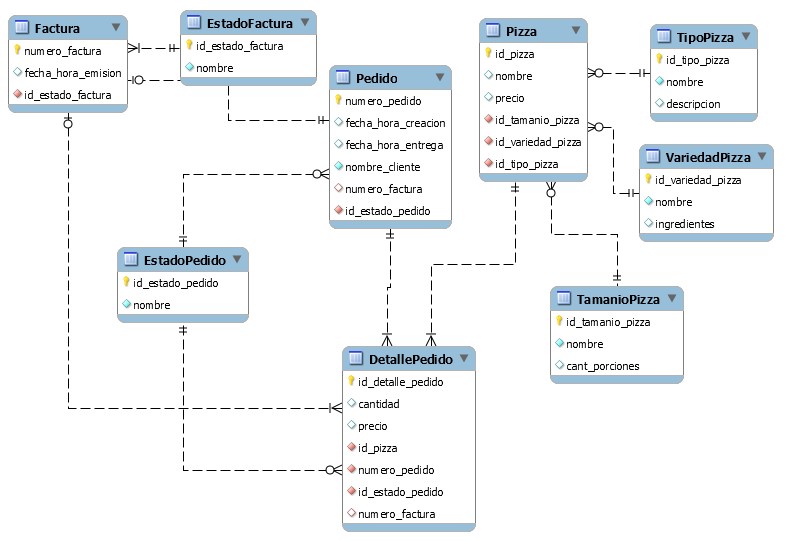
Programa Nacional 111 Mil - Analistas del Conocimie

nto

Guía Práctica del Módulo Base de Datos

Versión 1.1 – Liberada el 16/02/2017

24



Consignas para el Diseño e Implementación:

* Crear un nuevo proyecto en NetBeans. Si aún no está instalado, las instrucciones de instalación y configuración están en el *“Instructivo de instalación de Java JDK y NetBeans IDE.”*
* Si se requieren instrucciones acerca de la creación de un nuevo proyecto, consultar el *“Instructivo de creación de un nuevo proyecto en NetBeans IDE”.*
* Crear cada una de clases según lo especificado en el Modelo de Dominio planteado. Para cada clase deberás:
  + Implementar sus atributos con su tipo de datos correspondiente.
  + Definir un constructor para cada clase.
  + Crear los métodos de acceso [método get() y método set()], para poder obtener y/o asignar el valor de los atributos de una clase.
* Considerando la funcionalidad del caso de uso especificada en su descripción, la máquina de estado que modela el ciclo de vida de los objetos de la clase con sus estados y transiciones permitidas y el diagrama de secuencia que modela el escenario descripto, implementar los métodos necesarios para que los objetos de las clases puedan realizar el escenario modelado. Considerar la *signatura completa* del método, es decir: el modificador de acceso (public o private), el valor de retorno y los parámetros necesarios.
* Implementar las clases de *fabricación pura* que se consideren necesarias, para permitir que el usuario pueda ejecutar la funcionalidad del escenario modelado del caso de uso.
* Diseñar la interfaz gráfica, utilizando la vista de diseño que brinda NetBeans.