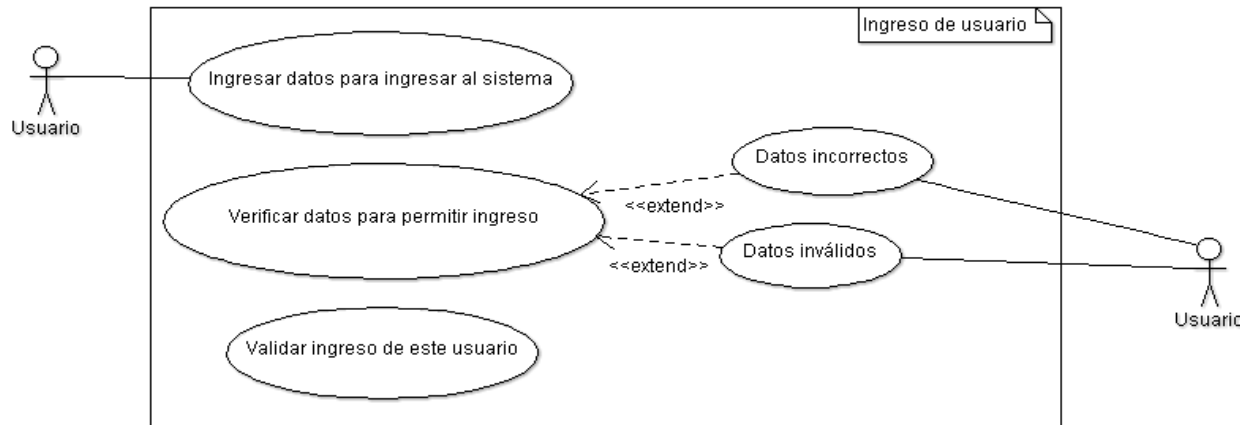


Universidad Rafael Landívar  
Facultad de ingeniería  
Análisis y diseño II  
Ingeniero Dhaby Xiloj

## Documentación del Proyecto

Danny Rocael Pérez Coroxón (21105-07)  
Josué Emmanuel Poz Choz (16466-14)  
Ana Gabriela Yoc De León (16135 15)

## Ingreso de de Usuario al Sistema



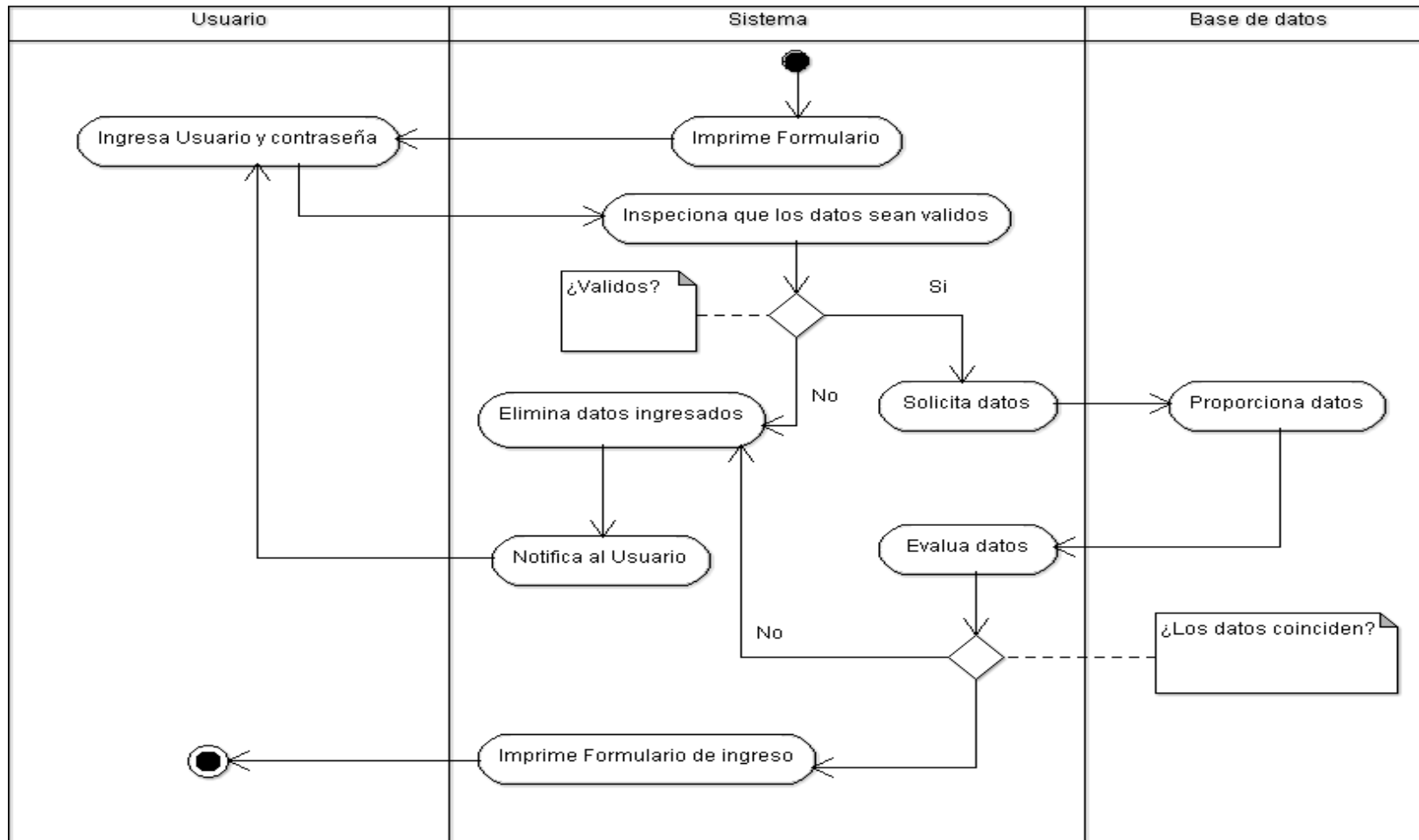
**Ingresar datos para ingresar al Sistema:** El usuario ingresa sus datos (Contraseña y nombre de usuario) en los campos correspondientes para que el sistema permita o deniegue el acceso a la información del Usuario correspondiente con el Nombre de usuario ingresado.

**Verificar datos para permitir el acceso:** El Sistema Verifica que el nombre de usuario y la contraseña sean los correctos para permitir el acceso a la información del usuario asignado.

- **Datos Incorrectos:** El sistema verifica que los datos ingresados no son los correctos debido a que el nombre del usuario o la contraseña no son los correctos.
- **Datos Inválidos:** El sistema verifica que los datos ingresados por el usuario, contienen caracteres no válidos.

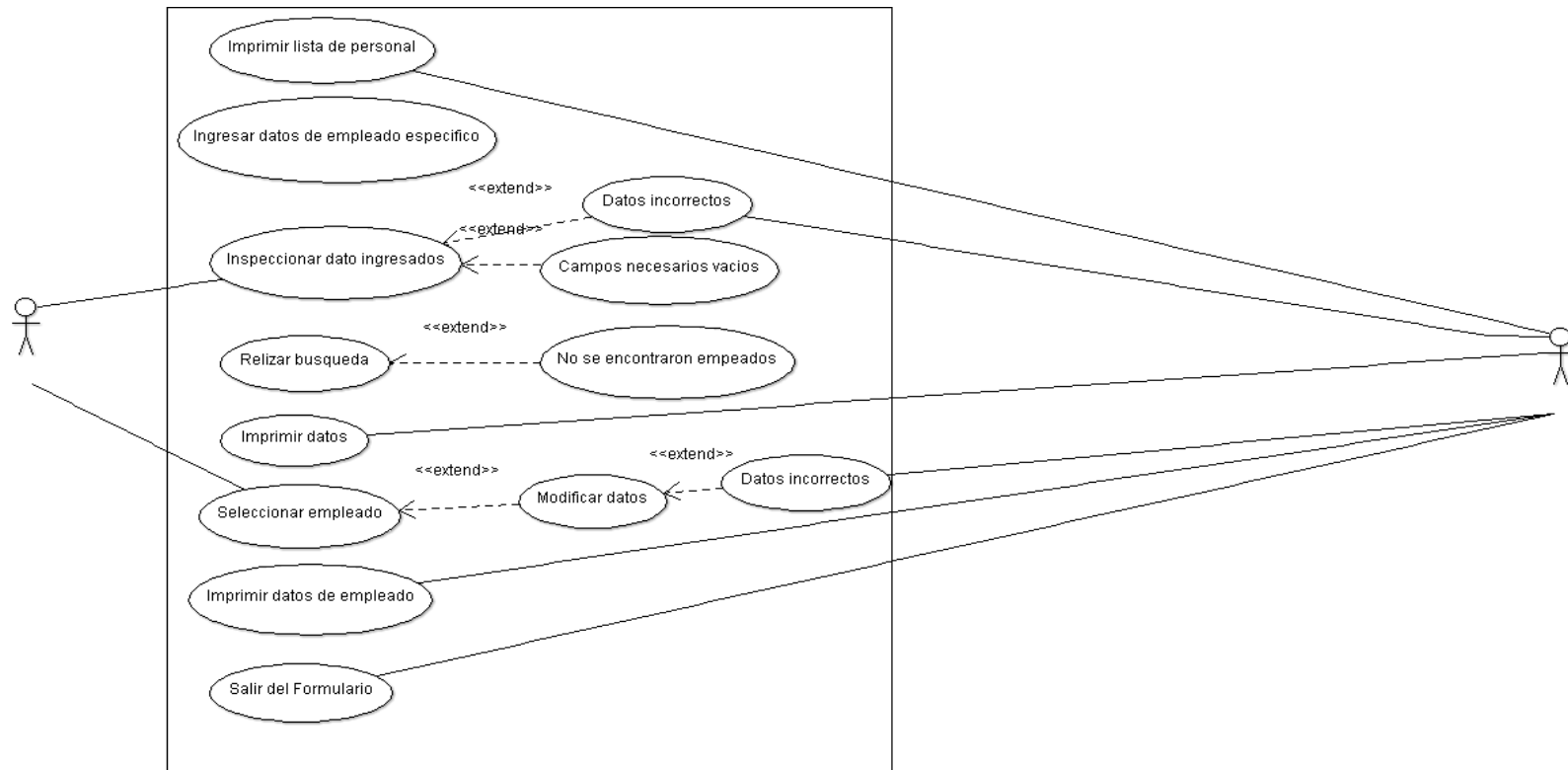
**Validar Ingreso de este Usuario:** El sistema informa que los datos ingresados son correctos y permite el acceso a la información del Usuario ingresado.

## Diagrama De Actividades Ingreso Al Sistema



## Modulo de Planillas

### Caso De Uso



Imprimir lista del personal: El sistema el imprime nombre de todos los empleados para mostrarlos al Usuario.

Ingresar datos del empleado específico: El Usuario ingresa los datos del usuario que desea ver para realizar la búsqueda de dicho Usuario.

Inspeccionar datos ingresados: El sistema verifica que los datos ingresados sean correctos para realizar la búsqueda.

- Datos incorrectos: Los datos ingresados, contienen caracteres no permitidos.
- Campos necesario vacíos: Los datos ingresados por el usuario no son los suficientes para realizar la búsqueda del empleado deseado.

Realizar Búsqueda: El sistema realiza una búsqueda de todos los empleados que encajan con la descripción, mostrarlos.

-El sistema informa que no existen empleados con la descripción dada, para que el usuario lo sepa.

Imprimir datos: El Sistema imprime los datos de la búsqueda, y muestra todos los datos de los empleados que encajen con los datos ingresados anteriormente.

Seleccionar datos: El usuario selecciona al empleado que busca, para ver sus datos.

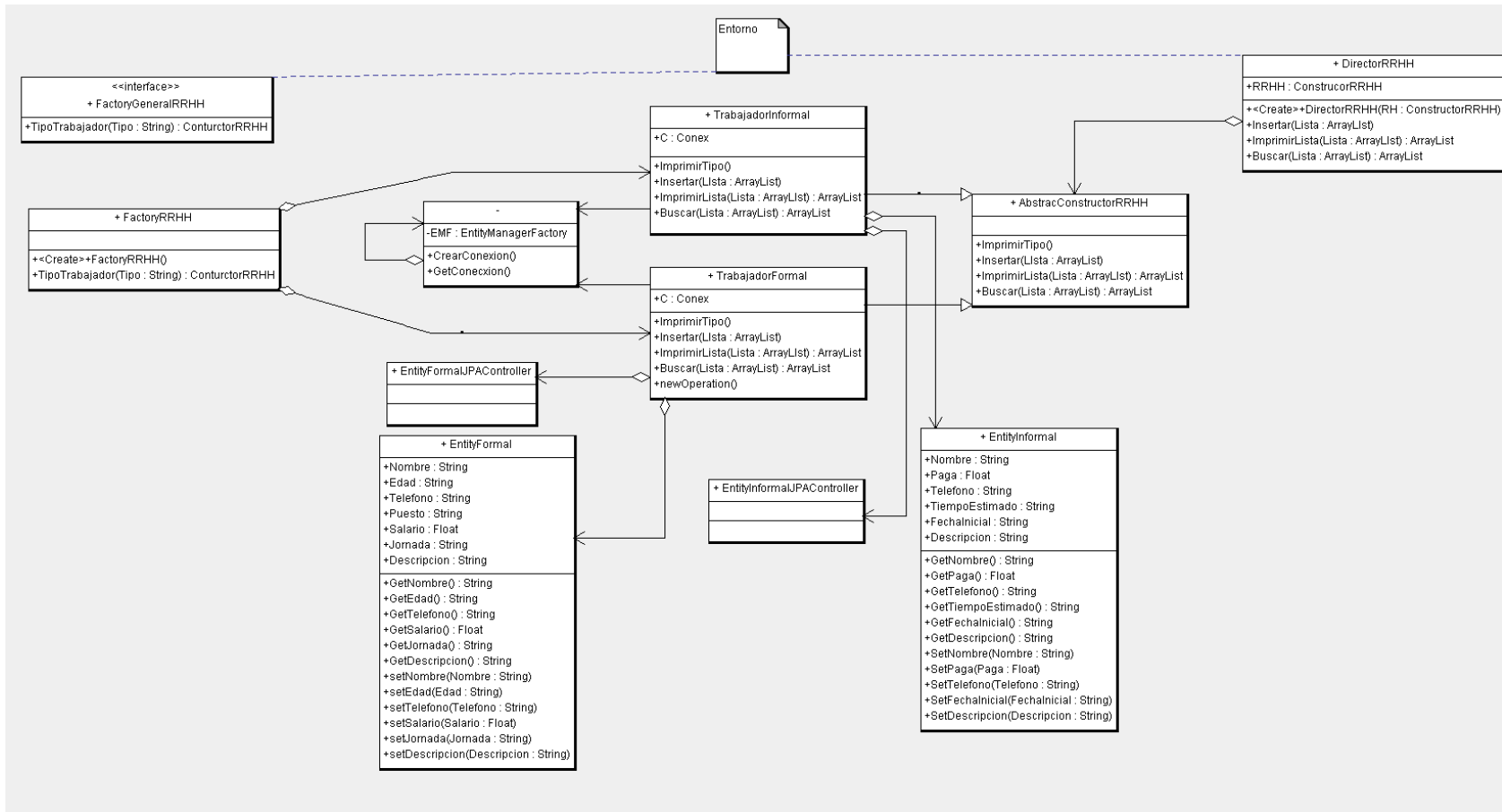
- Modificar datos: El Usuario modifica los datos del empleado, para ser guardados.

- Datos incorrectos: El usuario introduce datos no permitidos.

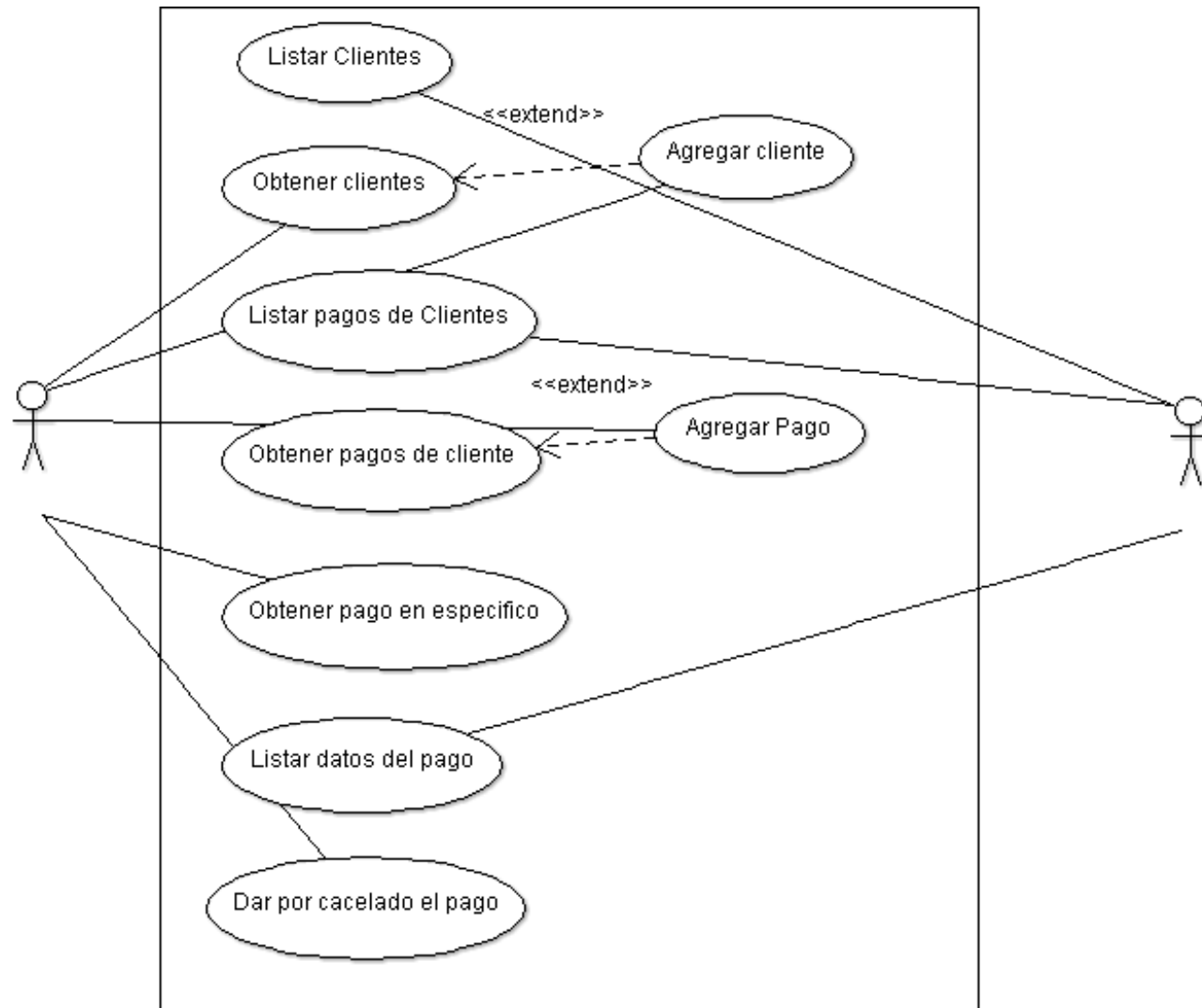
Imprimir datos de empleado: El sistema imprime los datos del empleado seleccionado para que el Usuario los vea.

Salir Formulario: El Usuario decide salir del formulario para dar por finalizadas las acciones.

## Diagrama De Clases



## Modulo De Pagos



Listar clientes: El Sistema imprime de manera ordenada, los nombres de todos los clientes para que el Usuario pueda verlos.

Obtener Cliente: El Usuario selecciona a el nombre de un cliente, para que el sistema pueda mostrar los datos de dicho cliente.

Agregar Cliente: El Usuario puede ingresar los datos de un cliente nuevo, para que el sistema pueda guardarlos en la base de datos.

Listar Pagos de cliente: El sistema imprime de manera ordenada, todos los pagos de un cliente en específico, para que el usuario pueda verlos.

Agregar Pago: El usuario ingresa un nuevo pago, para que el sistema, lo guarde en la base de datos.

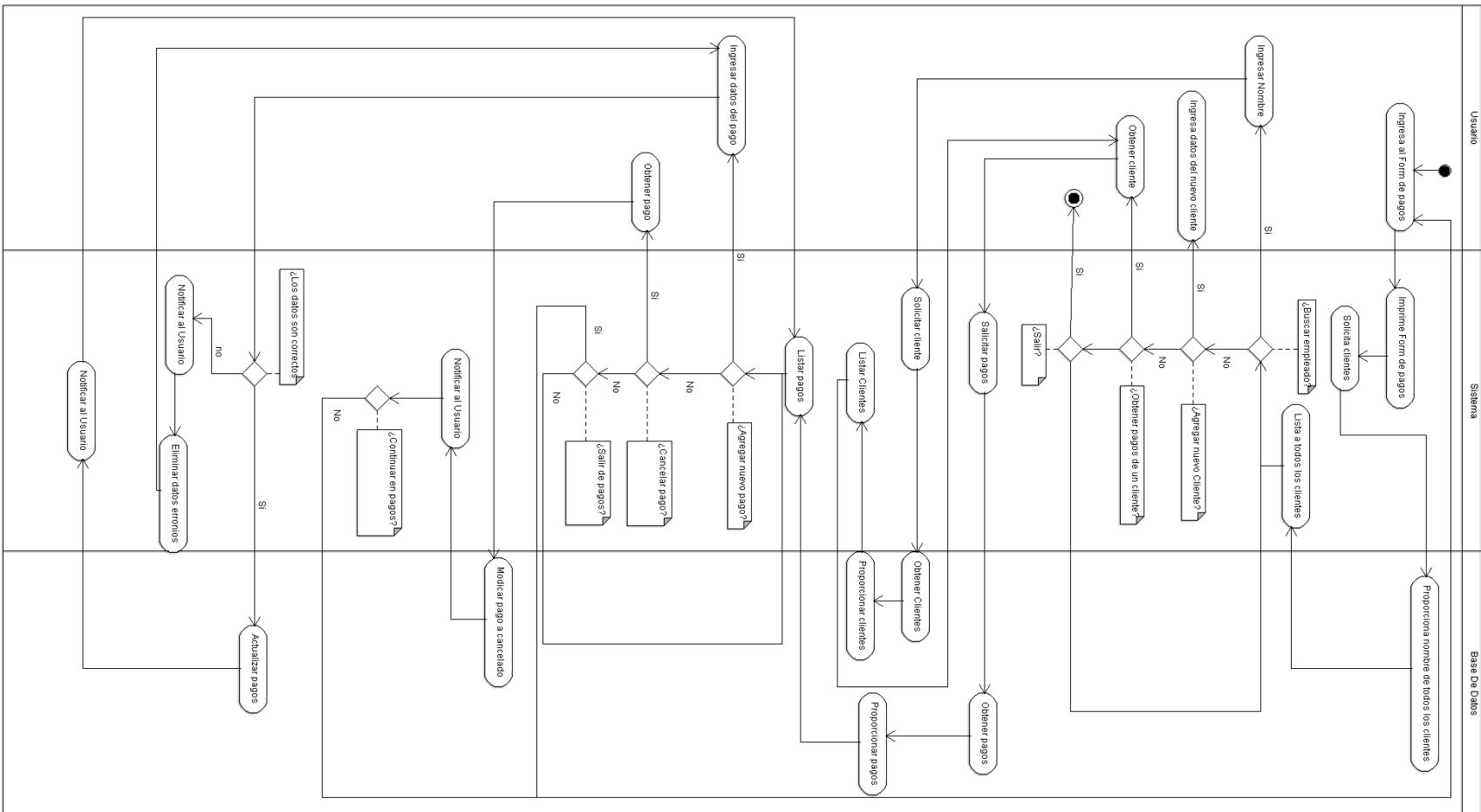
Obtener pago en específico: El Usuario, selecciona un pago de cliente, para que el sistema muestre los datos de dicho pago.

Listar datos del pago: El sistema muestra los datos de un pago, para que el usuario pueda verlos

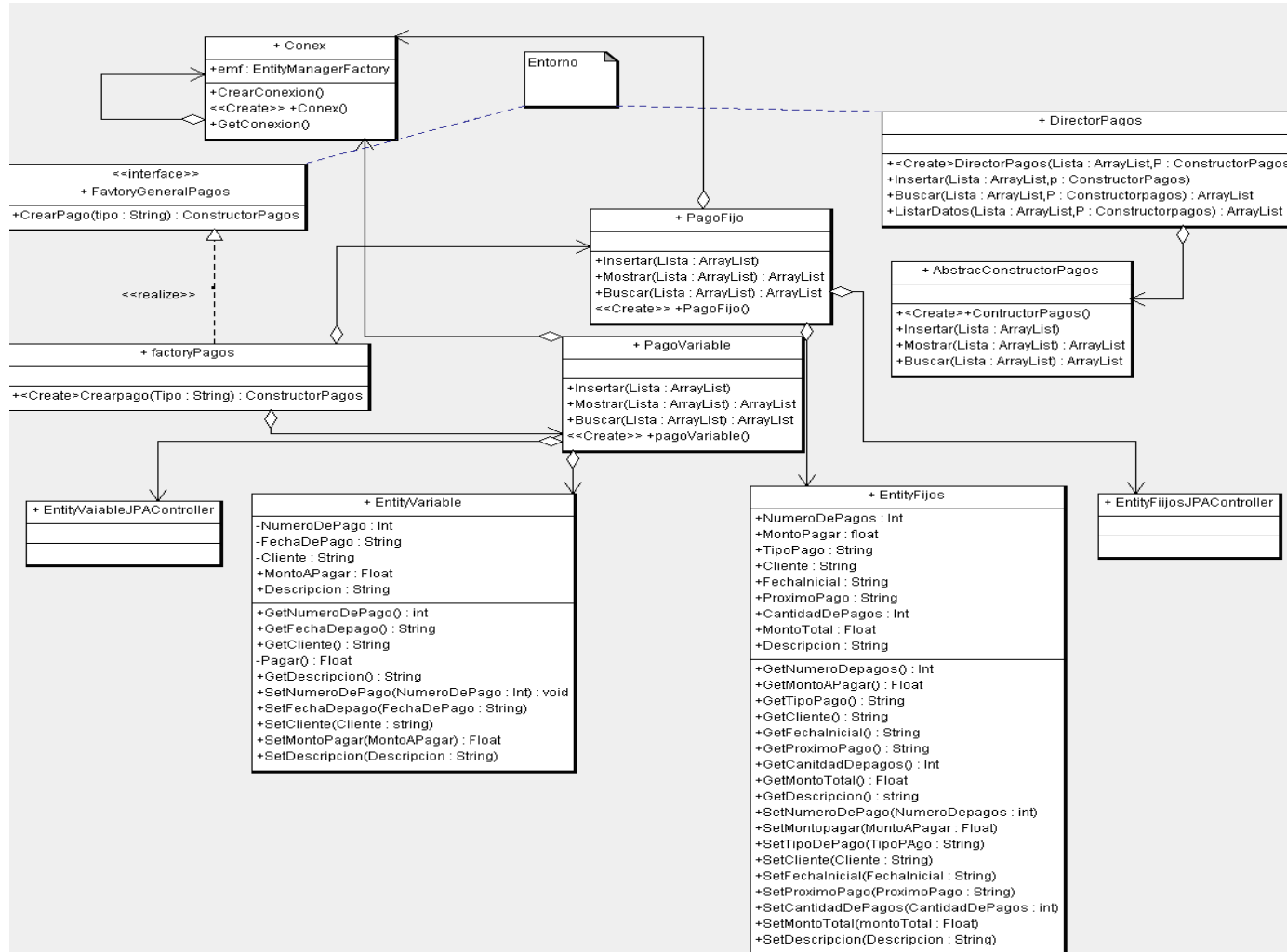
Dar por cancelado un pago: El Usuario da por cancelado un pago, para que el sistema, lo modifique en la base de datos.



### Diagrama De actividades



## Diagrama De Clases



## DESCRIPCIÓN DE CASO DE USO

<b>Nombre:</b>	Inventario
<b>Actores:</b>	Usuario
<b>Función:</b>	Control del inventario.
<b>Descripción:</b>	Ingresa al inventario

Se muestran todos los datos de los productos registrados.

Ingresa los datos de un producto para buscarlo.

Si el producto no existe puede crearlo.

Muestra los datos del producto

Puede modificar o discontinuar el producto.

## Inventario

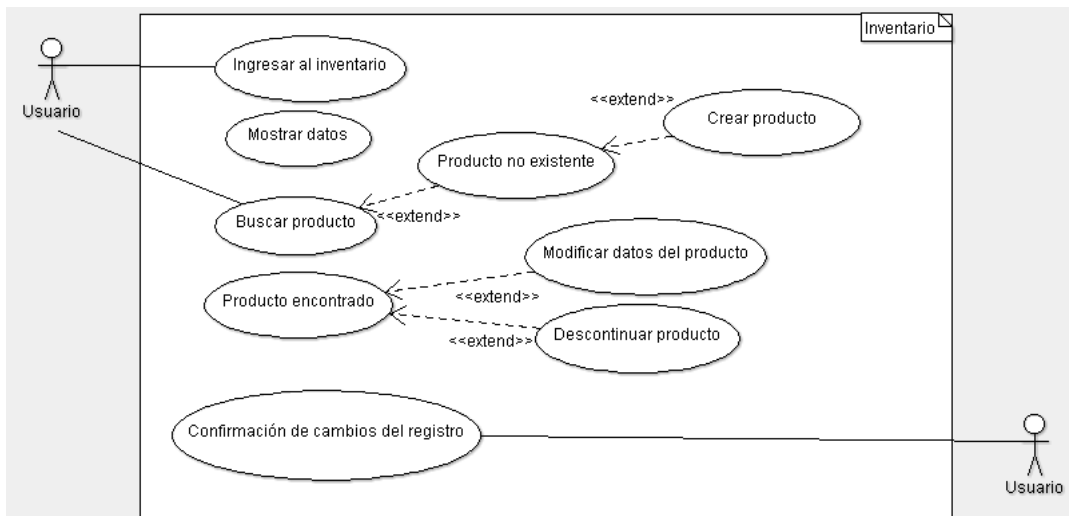


Diagrama de actividades

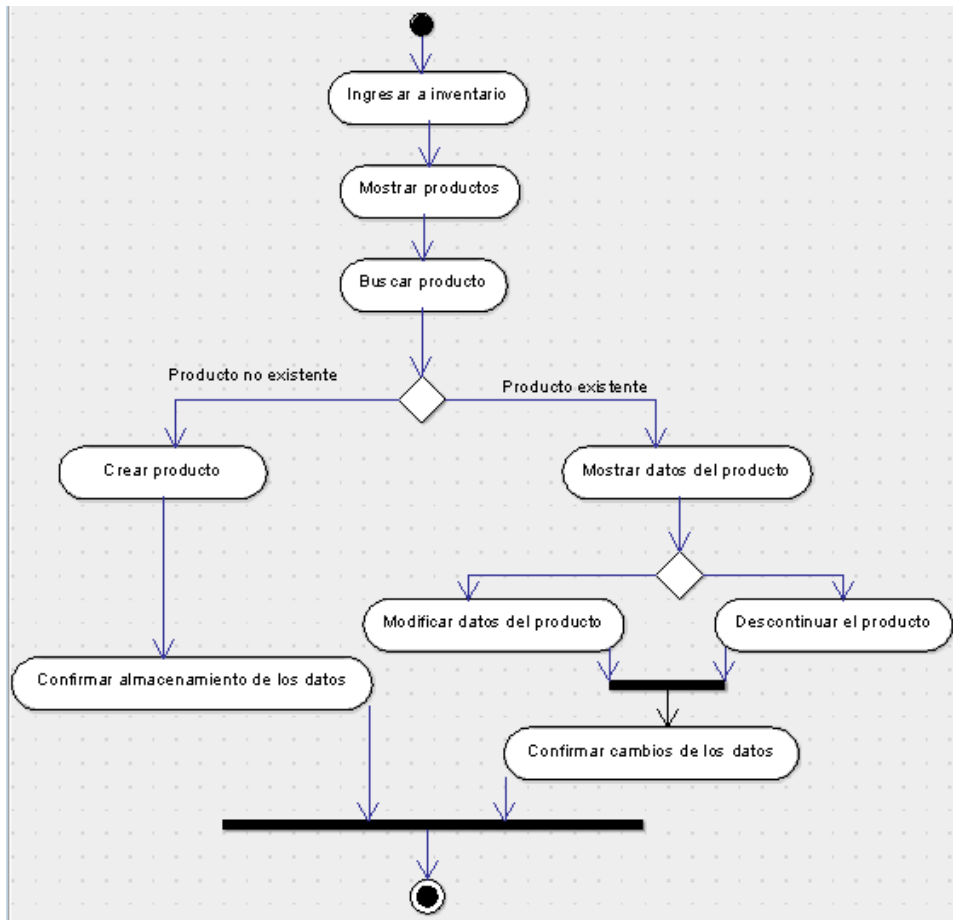
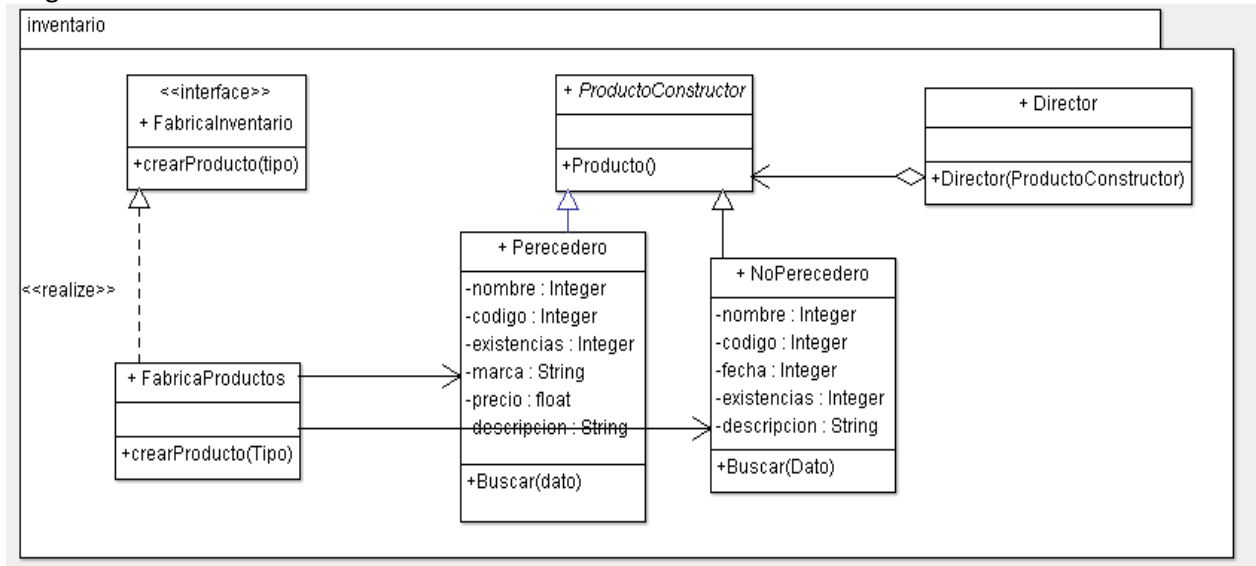
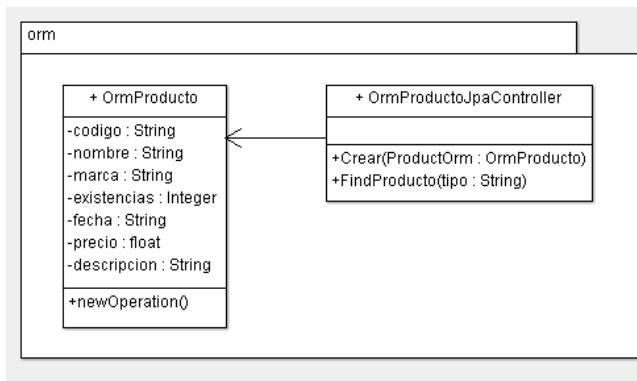


Diagrama de clases de inventario





## Patrón Factory y Builder

**FactoryCompras:** Interfaz que declara el método que devuelve el producto.

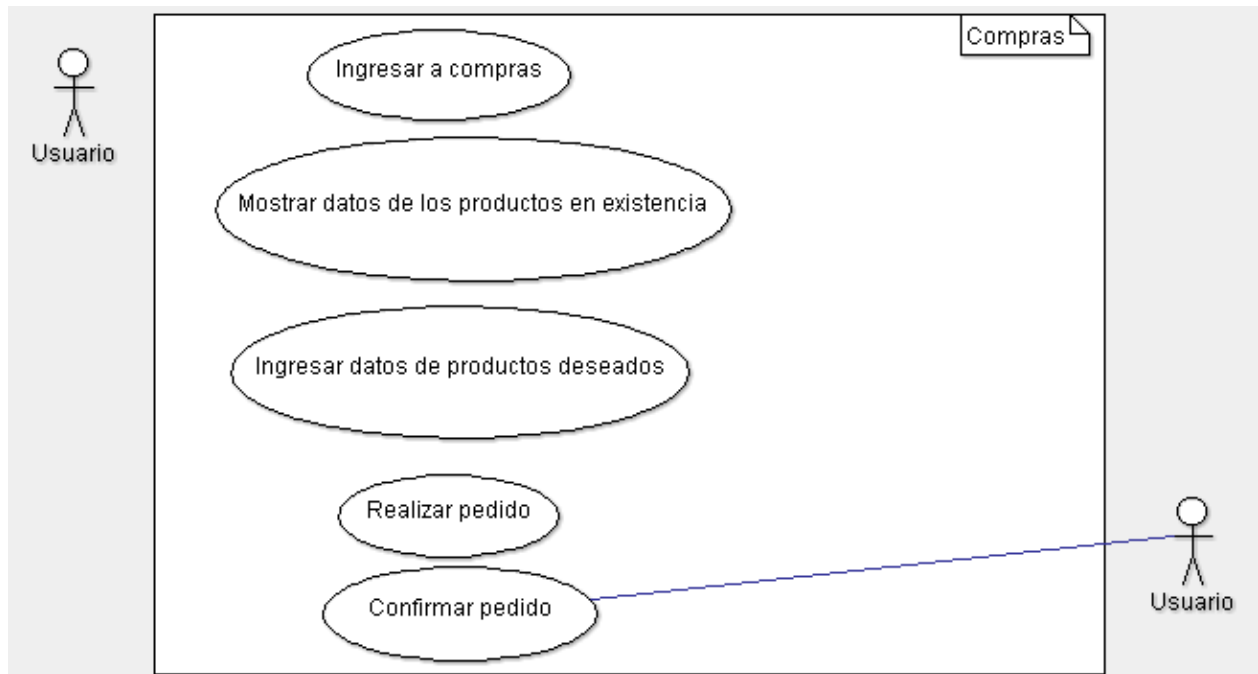
**FactoryComprasConcreto:** Clase que implementa la interfaz FactoryCompras. Implementa el método de FactoryCompras que crea y devuelve el producto concreto según el tipo de producto que se haya enviado como parámetro.

**ProductoConstructor:** Declara los pasos para construir el producto.

**Perecedero y NoPerecedero:** Clases que implementan ProductoConstructor y los metodos para la construcción del producto y para la inserción de los datos que correspondan al tipo de producto. Recibe como parámetro el ArrayList con los datos del producto, que añadirá mediante el JpaController.

**Director:** Construye los productos usando el objeto ProductoConstructor que recibe como parámetro, cuando la fábrica lo instancia. Otro parámetro que recibe es un ArrayList de cadenas, que contiene los datos recopilados por la interfaz. Usa este objeto para generar la construcción, además de los datos del array que son los datos del producto.

## Módulo de compras



#### DESCRIPCIÓN DE CASO DE USO

**Nombre:** Compras

**Actores:** Usuario

**Función:** Revisar las existencias disponibles para realizar los pedidos de los productos cuya cantidad de existencias está en límite mínimo del inventario.

**Descripción:** Ingresa al modulo de compras  
Muestra los productos en existencia  
Ingresa los datos del producto según el tipo de producto  
Aplica los cambios

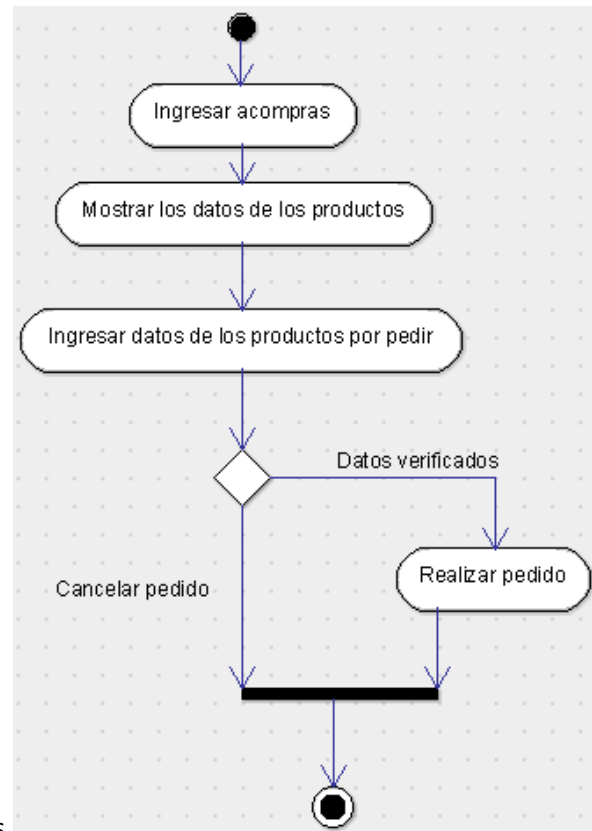
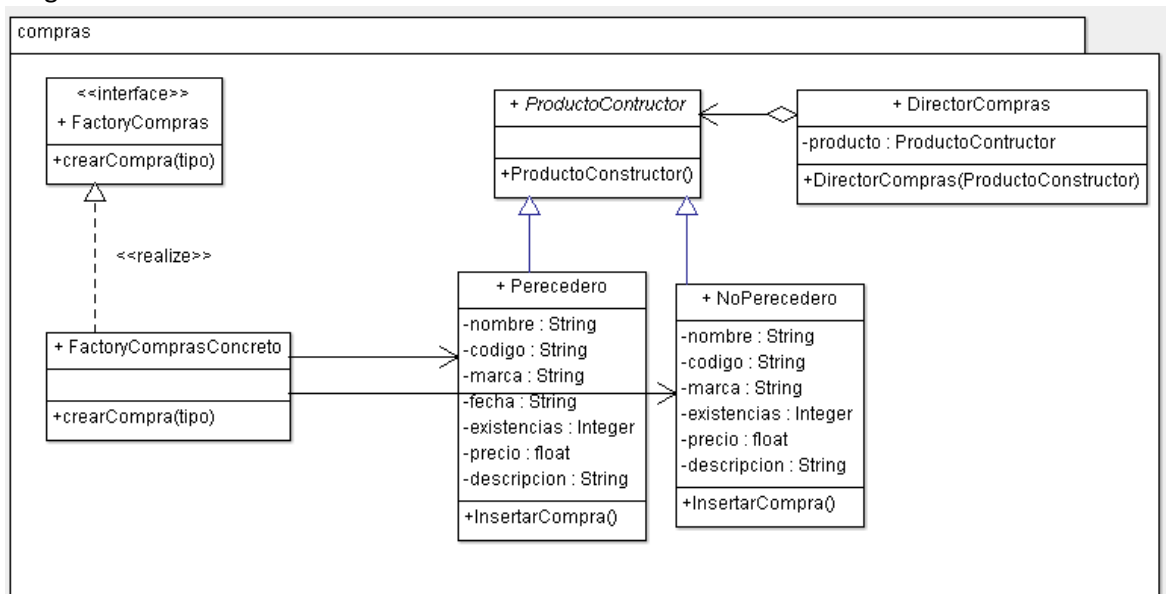
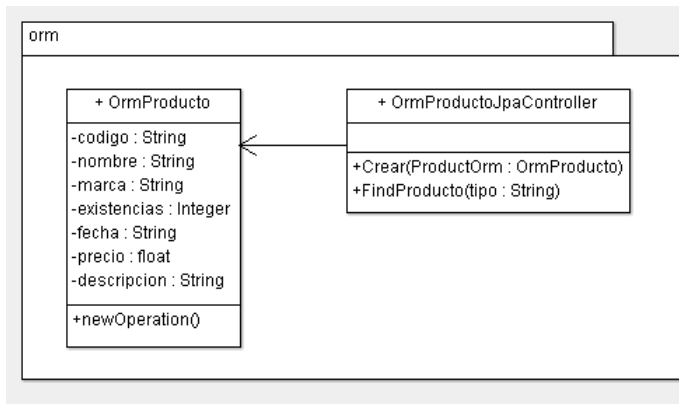


Diagrama de actividades del módulo de compras.

Diagrama de clases





### Patrón Factory y Builder

FabricaInventario: Interfaz que declara el método que devuelve el producto.

FactoryInventarioConcreto: Clase que implementa la interfaz FabricaInventario. Implementa el método de FabricaInventario que crea y devuelve el producto concreto según el tipo de producto que se haya enviado como parámetro.

ProductoConstructor: Declara los pasos para construir el producto.

Perecedero y NoPerecedero: Clases que implementan ProductoConstructor y los metodos para la construcción del producto y para la inserción de los datos que correspondan al tipo de producto. Recibe como parámetro el ArrayList con los datos del producto, que añadirá mediante el JpaController.

Director: Construye los productos usando el objeto ProductoConstructor que recibe como parámetro, cuando la fábrica lo instancia. Otro parámetro que recibe es un ArrayList de cadenas, que contiene los datos recopilados por la interfaz. Usa este objeto para generar la construcción, además de los datos del array que son los datos del producto.

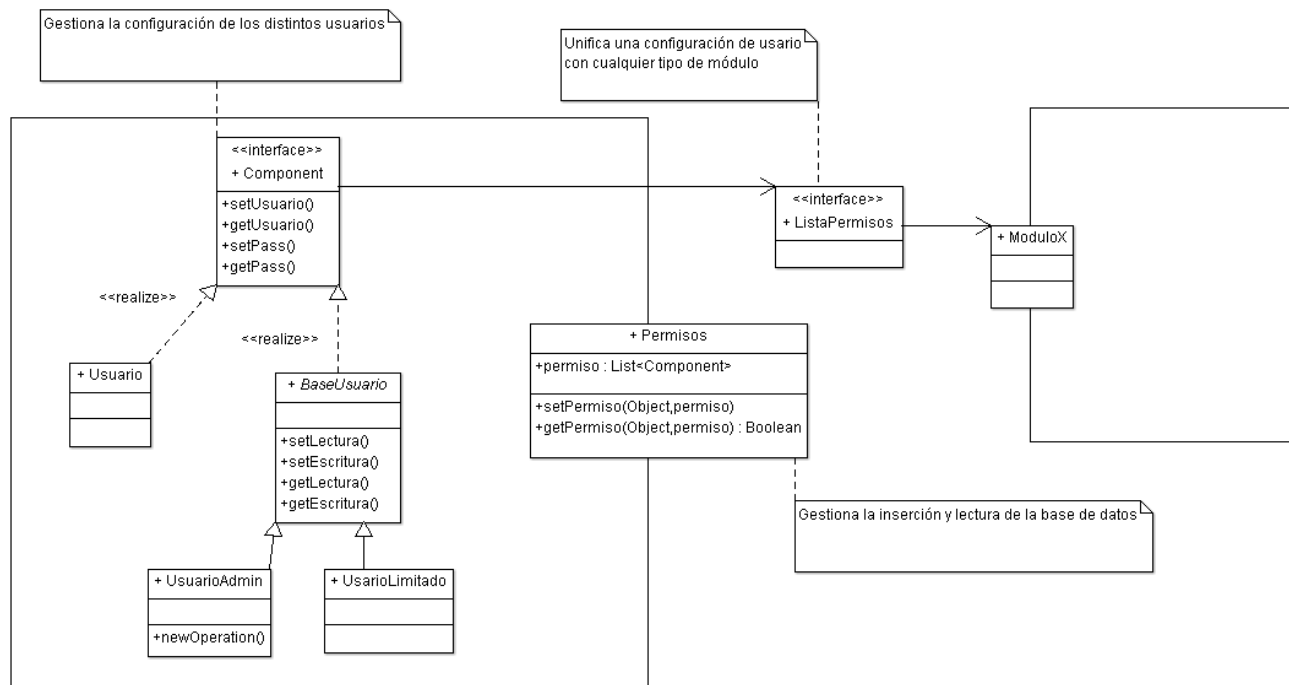


# Módulo Usuario:

DESCRIPCIÓN  
DE CASO DE  
USO

Nombre:	Usuario
Actores:	Usuario
Función:	Crear usuarios, con sus permisos de acceso
Descripción:	<div>Ingresar módulos usuarios</div> <div>Muestra los usuarios</div> <div>Muestra permisos de usuario</div> <div>Administra permisos de usuario</div> <div>Ingresar datos de usuario: usuario contraseña</div> <div>Valida inserción de usuarios</div> <div>Aplica los cambios</div>

Diagrama de clases



Patron delegate: Permite crear diversos tipos de permisos de usuario cuando sea necesario, estos ya están programados de manera definida con tres tipos de permisos, lectura, escritura y control total, para crear los permisos se tiene una fachada que se comunica a distintos módulos, cuando se le asigna un permiso al módulo se obtiene el módulo específico, el usuario, y utilizando el patrón delegate el permiso que se le asignara a dicho modulo almacenando en la base de datos la configuración deseada