



# Etapa de Análisis

## Casos de Uso

***Integrantes:***

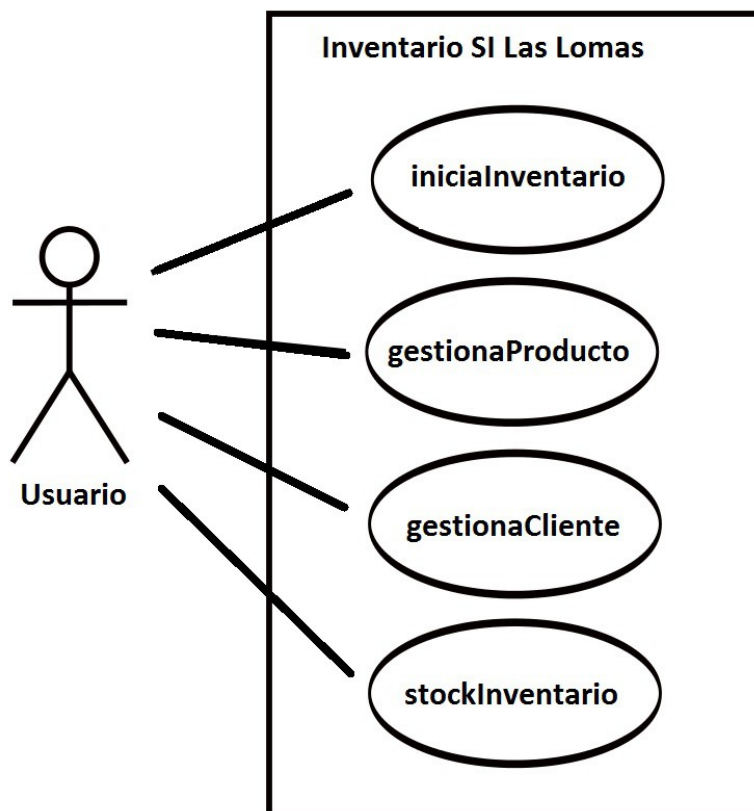
*Patricia Bello Parra.*

*Carolina Ramírez López.*

*Claudia Navarro Galindo.*

*Juan Maldonado Imilan.*

*Diagrama de Caso de Uso.*



Casos de Usos.

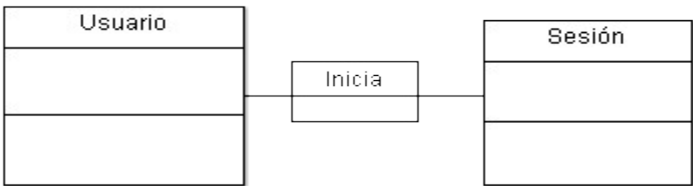
Caso de uso 1

**Nombre:** inicialInventario()  
**Actores:** usuario, sistema  
**Propósito:** el sistema permitirá al usuario iniciar sesión para utilizar la aplicación.  
**Precondición:** ya se ha creado el perfil de ingreso del usuario.  
**Resumen:** Éste caso de uso comienza cuando el usuario abre la aplicación y el sistema le solicita los datos de ingreso para validarlo, de ser así el sistema entrega el menú principal de la aplicación.

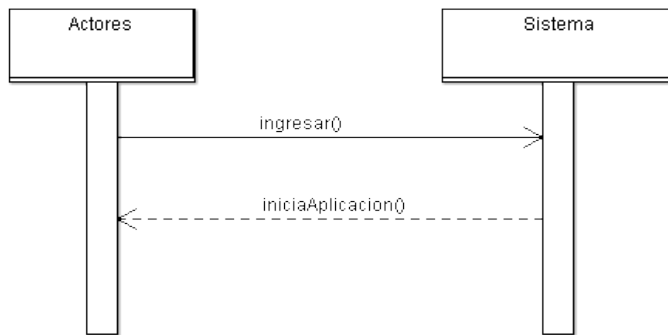
1.- Este caso de uso comienza cuando el usuario abre la aplicación para iniciar sesión.	
	2.- El sistema solicita el ingreso de los datos del usuario
3.- El usuario ingresa su nombre y contraseña.	
	4.- El sistema valida los datos ingresados y despliega el menú principal.
5.- El usuario ingresa al menú principal del sistema.	

Linea 3: El usuario o contraseña no son válidos. El sistema entrega un mensaje de error y solicita ingresar nuevamente los datos.

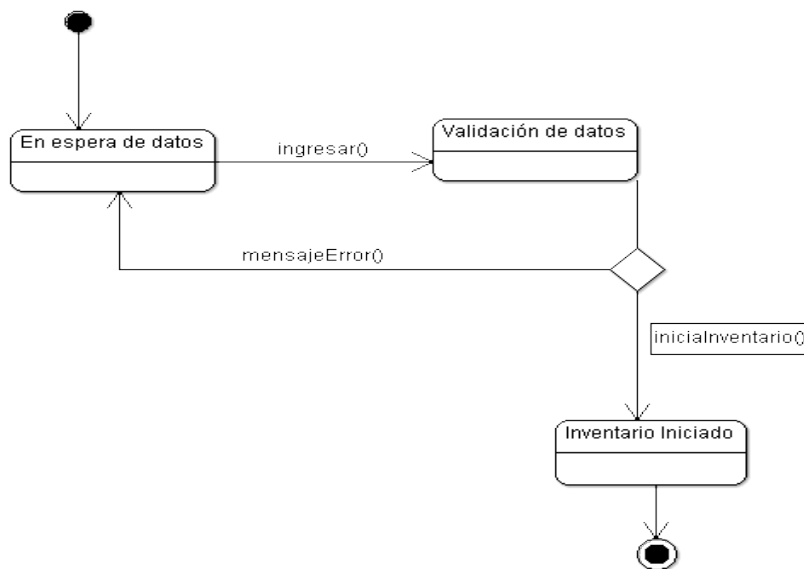
Modelo conceptual: inicialInventario()



### Diagrama de secuencia: inicialInventario()



### Diagrama de estado: inicialInventario()



### Contratos: inicialInventario()

**Contrato:** ingresar()

**Responsabilidad:** Permitir que el usuario ingrese al sistema.

**Precondición:** El usuario y contraseña existen.

**Postcondición:** Se inicia sesión.

## Caso de uso 2

**Nombre:** gestionaProducto()

**Actores:** usuario, sistema

**Propósito:** el sistema permitirá agregar, eliminar o modificar la información de un producto.

**Precondición:** la base de datos ya ha sido creada.

**Resumen:** Éste caso de uso comienza cuando el usuario selecciona la opción “Producto” del menú principal, selecciona un producto (si existe) y realiza los cambios, o lo agrega en caso de que el producto no exista. El sistema guardará las modificaciones en la base de datos.

1.- Este caso de uso comienza cuando el usuario selecciona la opción “Producto” del menú principal del Sistema.	
	2.- El sistema despliega la información de cada producto almacenado en la base de datos. Y ofrece al usuario la opción de editar un producto, agregar un nuevo producto, o eliminar un producto existente.
3.- El usuario selecciona un producto y la opción “Editar”.	
	4.- El sistema desplegará el formulario para modificar la información de un producto existente.
5.- El usuario ingresa la información necesaria y finaliza.	
	6.- El sistema solicita al usuario que confirme los cambios realizados.
7.- El usuario confirma la acción realizada.	
	8.- El sistema almacena los cambios en la base de datos y regresa a la lista de productos.
9.- El usuario selecciona la opción “Agregar”.	
	10.- El sistema desplegará el formulario para agregar la información de un nuevo producto.
11.- El usuario ingresa la información necesaria y finaliza.	
	12.- El sistema solicita al usuario que confirme los cambios realizados.
13.- El usuario confirma la acción realizada.	
	14.- El sistema almacena el nuevo producto en la base de datos y regresa a la lista de productos.

15.- El usuario selecciona un producto y la opción “Eliminar”.	
	16.- El sistema solicita al usuario que confirme la acción.
17.- El usuario confirma la acción realizada.	
	18.- El sistema elimina el producto seleccionado de la base de datos y regresa a la lista de productos.

Linea 5: El usuario no ingresa alguno de los datos solicitados, o el ingreso no es válido. El sistema envía mensaje de error y solicita el ingreso de los datos nuevamente.

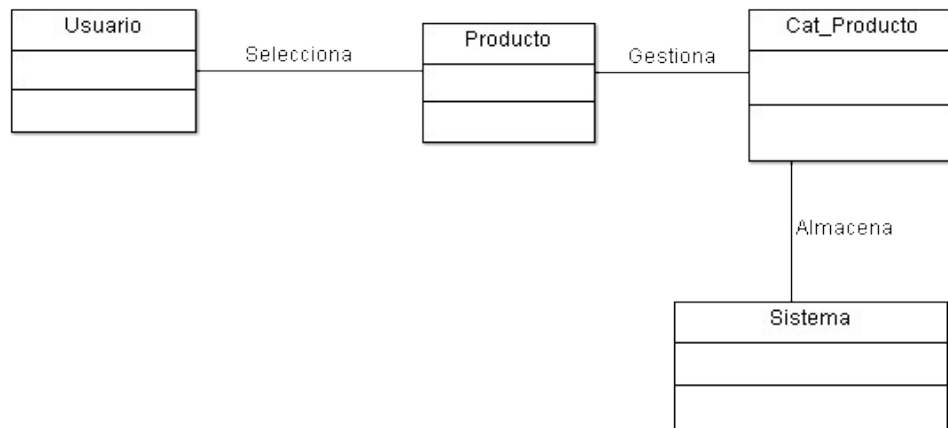
Linea 7: El usuario no confirma el cambio de los datos. El sistema advierte al usuario que no se almacenarán los datos y no realiza cambios en la base de datos.

Linea 11: El usuario no ingresa alguno de los datos solicitados, el ingreso no es válido. El sistema envía mensaje de error y solicita el ingreso de los datos nuevamente.

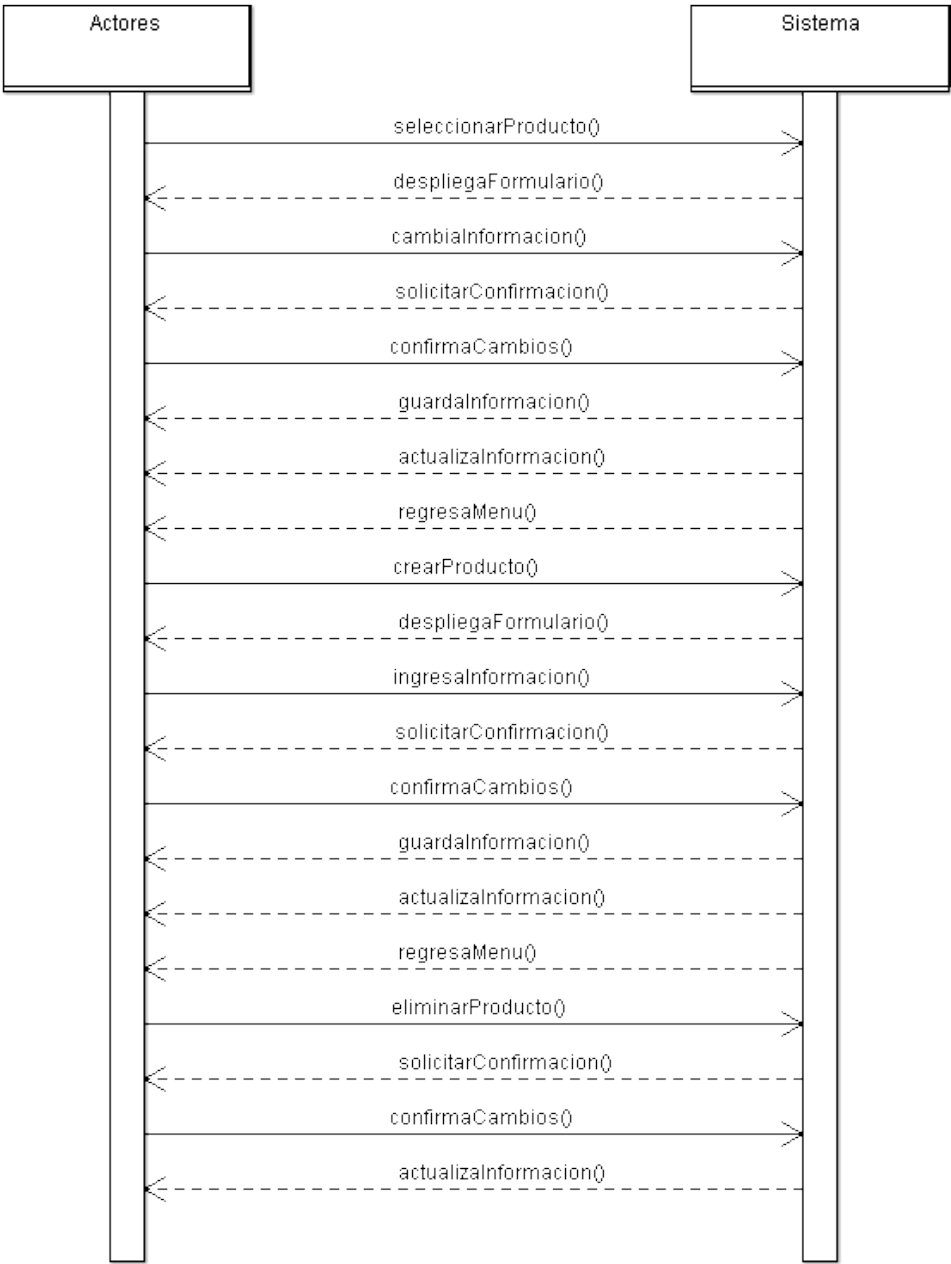
Linea 13: El usuario no confirma el cambio de los datos. El sistema advierte al usuario que no se almacenarán los datos y no realiza cambios en la base de datos.

Linea 17: El usuario no confirma la acción. El sistema no eliminará el producto seleccionado.

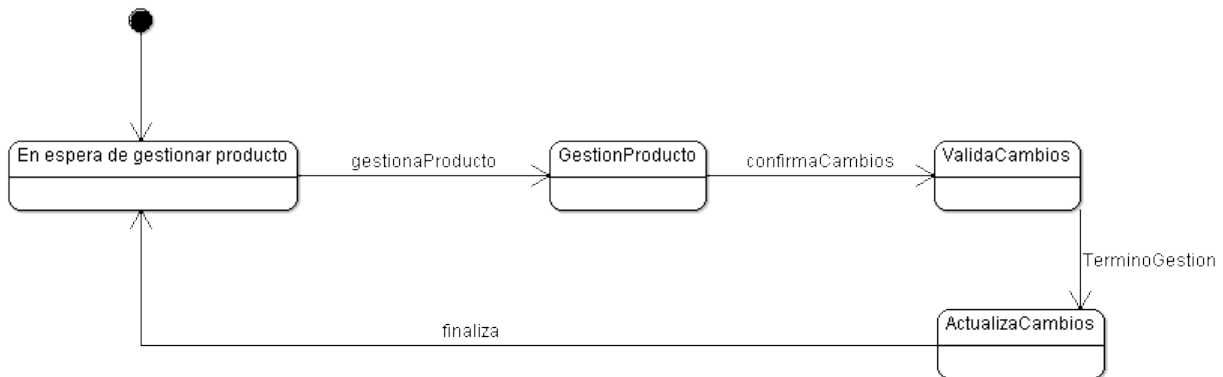
### **Modelo conceptual: gestionaProducto()**



**Diagrama de secuencia: gestionaProducto()**



## Diagrama de estado: gestionaProducto()



## Contratos: gestionaProducto()

**Contrato:** `seleccionarProducto()`

**Responsabilidad:** Seleccionar un producto en el sistema para editar su información.

**Precondición:** El producto a editar existe en la Base de Datos.

Se crea instancia `seleccionarProducto()`.

Producto crea instancia `seleccionarProducto()`.

**Postcondición:** El producto es seleccionado.

**Contrato:** `cambiarInformación()`.

**Responsabilidad:** editar la información del producto seleccionado.

**Precondición:** Se conoce toda la información del producto.

Se ha seleccionado previamente un producto.

**Postcondición:** Se cambia la información del producto.

Se crea instancia `cambiarInformación()`.

Producto crea instancia `cambiarInformación()`

Producto crea instancia `solicitarConfirmación()`.

**Excepción:** No se puede cambiar el código del producto.

**Contrato:** `confirmaCambios()`.

**Responsabilidad:** Confirmar cambios realizados en la información del producto.

**Precondición:** El usuario ha finalizado de modificar información del producto.

**Postcondición:** Se almacenan los cambios de la información del producto confirmado.

Se crea instancia `confirmaCambios()`.

Producto crea instancia `confirmaCambios()`

Producto crea instancia `guardarInformación()`.

**Contrato:** `crearProducto()`.

**Responsabilidad:** desplegar un formulario para agregar un nuevo producto en el sistema.



**Precondición:** La Base de Datos de los productos esta creada.

**Postcondición:** Se despliega el formulario para ingresar información del nuevo producto.  
Se crea instancia crearProducto().

**Excepción:** Si el código no es válido, indicar el error cometido.

**Contrato:** ingresaInformación().

**Responsabilidad:** Registrar los datos introducidos por el usuario y almacenarlos en la Base de Datos.

**Precondición:** El usuario conoce toda la información del producto que se solicita.

**Postcondición:** El usuario ingresa toda la información del producto solicitada.

Se crea la instancia ingresaInformación().

Producto crea instancia ingresaInformación()

Producto crea instancia solicitarConfirmación().

**Excepción:** No se puede dejar en blanco ningún campo de información.

**Contrato:** confirmaCambios().

**Responsabilidad:** Confirmar información ingresada sobre el nuevo producto.

**Precondición:** El usuario ingresó toda la información solicitada del producto.

**Postcondición:** Se almacenan los cambios en la información del producto confirmado.

Se crea la instancia confirmaCambios().

Producto crea instancia confirmaCambios()

Producto crea instancia guardarInformación()

Producto crea instancia actualizarInformación().

**Contrato:** eliminarProducto().

**Responsabilidad:** Eliminar un producto de la base de datos del sistema.

**Precondición:** El producto existe en la Base de Datos.

El producto ha sido seleccionado previamente.

**Postcondición:** Se solicita eliminar el producto.

Se crea instancia eliminarProducto().

Producto crea instancia eliminarProducto()

Producto crea instancia actualizarInformación().

**Contrato:** confirmaCambios().

**Responsabilidad:** Confirmar eliminación de un producto.

**Precondición:** Se ha solicitado la eliminación de un producto.

**Postcondición:** Se elimina el producto de la base de datos del sistema.

Se crea la instancia confirmaCambios().

Producto crea instancia confirmaCambios()

Producto crea instancia guardarInformación().

### **Caso de uso 3**

**Nombre:** gestionaCliente()

**Actores:** usuario, sistema

**Propósito:** el sistema permitirá agregar, eliminar o modificar la información de un cliente.

**Precondición:** la base de datos ya ha sido creada.

**Resumen:** Éste caso de uso comienza cuando el usuario selecciona la opción “clientes” del menú principal, selecciona un cliente (si existe) y realiza los cambios, o lo agrega en caso de que el cliente no exista. El sistema guardará las modificaciones en la base de datos.

1.- Este caso de uso comienza cuando el usuario selecciona la opción “Cliente” del menú principal del Sistema.	
	2.- El sistema despliega la información de cada cliente almacenado en la base de datos. Y ofrece al usuario la opción de editar un cliente, agregar un nuevo cliente, o eliminar un cliente existente.
3.- El usuario selecciona un cliente y la opción “Editar”.	
	4.- El sistema desplegará el formulario para modificar la información de un cliente existente.
5.- El usuario ingresa la información necesaria y finaliza.	
	6.- El sistema solicita al usuario que confirme los cambios realizados.
7.- El usuario confirma la acción realizada.	
	8.- El sistema almacena los cambios en la base de datos y regresa a la lista de clientes.
9.- El usuario selecciona la opción “Agregar”.	
	10.- El sistema desplegará el formulario para agregar la información de un nuevo cliente.
11.- El usuario ingresa la información necesaria y finaliza.	
	12.- El sistema solicita al usuario que confirme los cambios realizados.
13.- El usuario confirma la acción realizada.	
	14.- El sistema almacena el nuevo cliente en la base de datos y regresa a la lista de clientes.
15.- El usuario selecciona un cliente y la opción	

“Eliminar”.	
	16.- El sistema solicita al usuario que confirme la acción.
17.- El usuario confirma la acción realizada.	
	18.- El sistema elimina el cliente seleccionado de la base de datos y regresa a la lista de clientes.

Linea 5: El usuario no ingresa alguno de los datos solicitados, o el ingreso no es válido. El sistema envía mensaje de error y solicita el ingreso de los datos nuevamente.

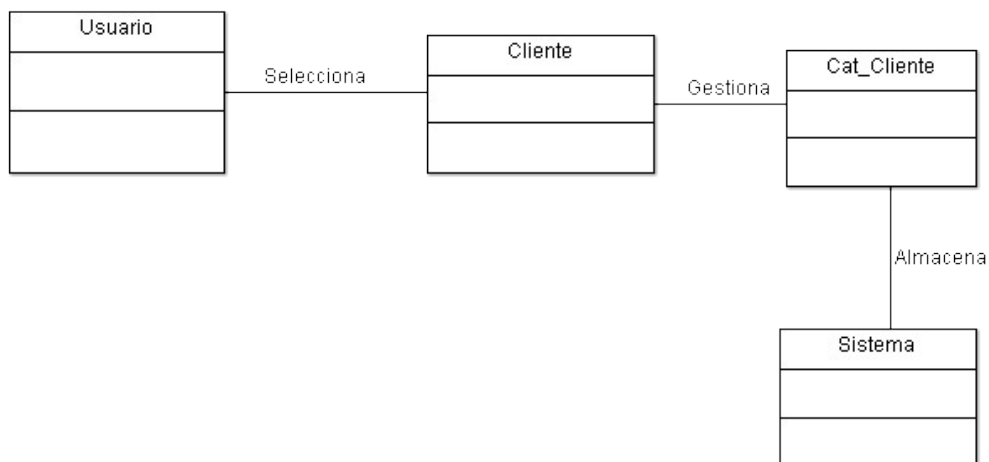
Linea 7: El usuario no confirma el cambio de los datos. El sistema advierte al usuario que no se almacenarán los datos y no realiza cambios en la base de datos.

Linea 11: El usuario no ingresa alguno de los datos solicitados, el ingreso no es válido. El sistema envía mensaje de error y solicita el ingreso de los datos nuevamente.

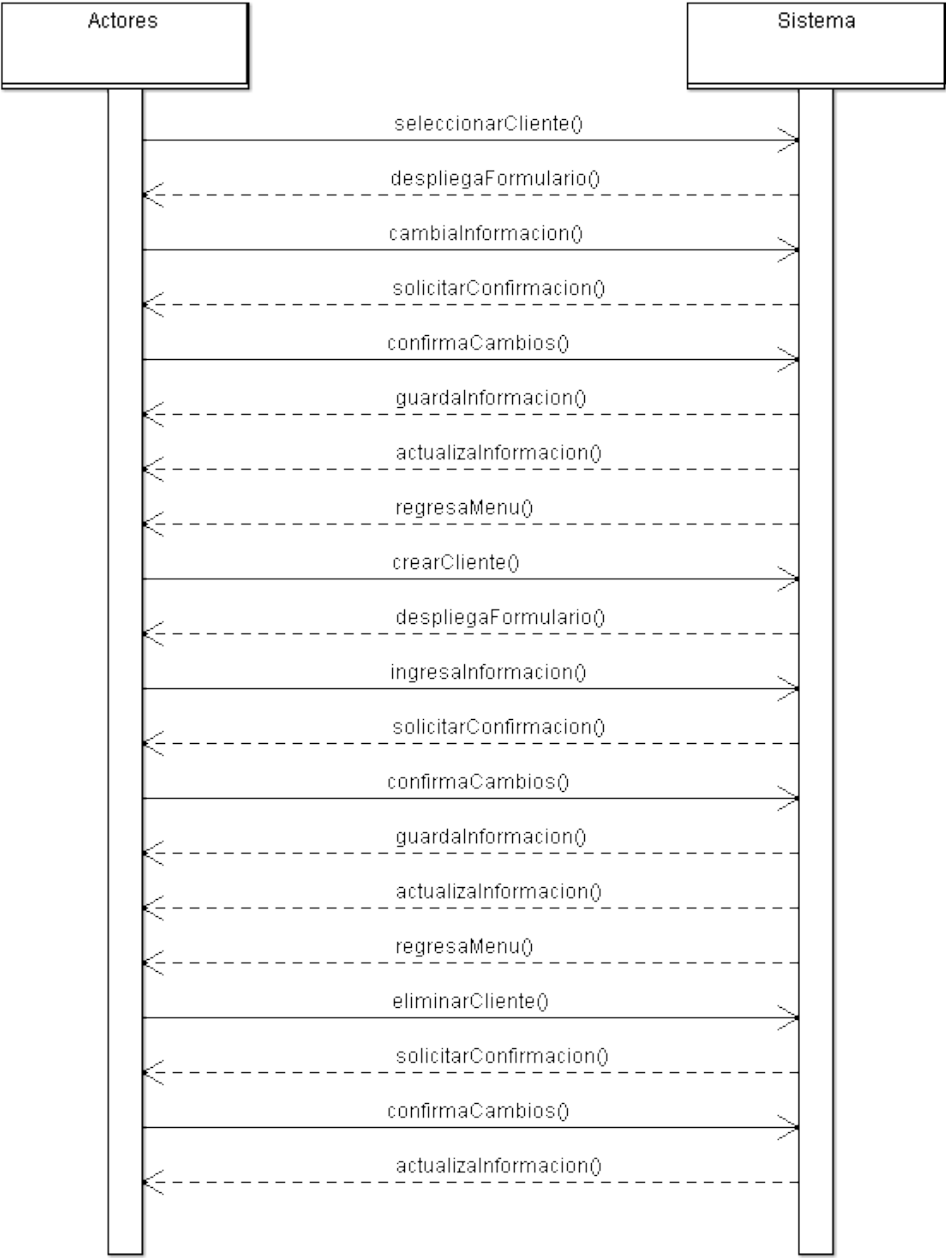
Linea 13: El usuario no confirma el cambio de los datos. El sistema advierte al usuario que no se almacenarán los datos y no realiza cambios en la base de datos.

Linea 17: El usuario no confirma la acción. El sistema no eliminará el cliente seleccionado.

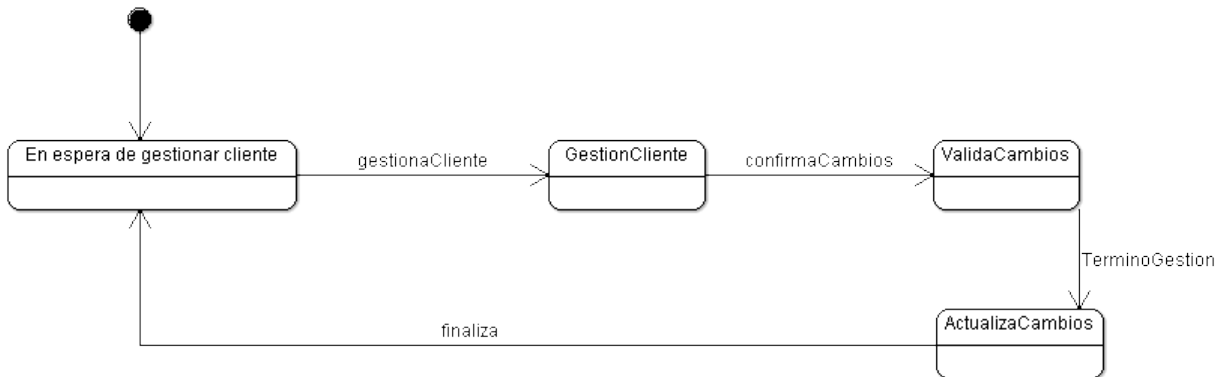
### **Modelo conceptual: gestionaCliente()**



**Diagrama de secuencia: gestionaCliente()**



### Diagrama de estado: gestionaCliente()



### Contratos: gestionaCliente()

**Contrato:** seleccionarCliente()

**Responsabilidad:** Seleccionar un cliente en el sistema para editar su información.

**Precondición:** El cliente a editar existe en la Base de Datos.

Se crea instancia seleccionarCliente().

Cliente crea instancia seleccionarCliente().

**Postcondición:** El cliente es seleccionado.

**Contrato:** cambiarInformación().

**Responsabilidad:** editar la información del cliente seleccionado.

**Precondición:** Se conoce toda la información del cliente.

Se ha seleccionado previamente un cliente.

**Postcondición:** Se cambia la información del cliente.

Se crea instancia cambiarInformación().

Cliente crea instancia cambiarInformación()

Cliente crea instancia solicitarConfirmación().

**Excepción:** No se puede cambiar el código del cliente.

**Contrato:** confirmaCambios().

**Responsabilidad:** Confirmar cambios realizados en la información del cliente.

**Precondición:** El usuario ha finalizado de modificar información del cliente.

**Postcondición:** Se almacenan los cambios de la información del cliente confirmado.

Se crea instancia confirmaCambios().

Cliente crea instancia confirmaCambios()

Cliente crea instancia guardarInformación().

**Contrato:** crearCliente().

**Responsabilidad:** Desplegar un formulario para agregar un nuevo cliente en el sistema.

**Precondición:** La Base de Datos de los clientes esta creada.

**Postcondición:** Se despliega el formulario para ingresar información del nuevo cliente.  
Se crea instancia crearCliente().

**Excepción:** Si el código no es válido, indicar el error cometido.

**Contrato:** ingresaInformación().

**Responsabilidad:** Registrar los datos introducidos por el usuario y almacenarlos en la Base de Datos.

**Precondición:** El usuario conoce toda la información del cliente que se solicita.

**Postcondición:** El usuario ingresa toda la información del cliente solicitada.

Se crea la instancia ingresaInformación().

Cliente crea instancia ingresaInformación()

Cliente crea instancia solicitarConfirmación().

**Excepción:** No se puede dejar en blanco ningún campo de información.

**Contrato:** confirmaCambios().

**Responsabilidad:** Confirmar información ingresaada sobre el nuevo cliente.

**Precondición:** El usuario ingresó toda la información solicitada del cliente.

**Postcondición:** Se almacenan los cambios en la información del cliente confirmado.

Se crea la instancia confirmaCambios().

Cliente crea instancia confirmaCambios()

Cliente crea instancia guardarInformación()

Cliente crea instancia actualizarInformación().

**Contrato:** eliminarCliente().

**Responsabilidad:** Eliminar un cliente de la base de datos del sistema.

**Precondición:** El cliente existe en la Base de Datos.

El cliente ha sido seleccionado previamente.

**Postcondición:** Se solicita eliminar el cliente.

Se crea instancia eliminarCliente().

Cliente crea instancia eliminarcliente()

Cliente crea instancia actualizarInformación().

**Contrato:** confirmaCambios().

**Responsabilidad:** Confirmar eliminación de un cliente.

**Precondición:** Se ha solicitado la eliminación de un cliente.

**Postcondición:** Se elimina el cliente de la base de datos del sistema.

Se crea la instancia confirmaCambios().

Cliente crea instancia confirmaCambios()

Cliente crea instancia guardarInformación().

## **Caso de uso 4**

**Nombre:** resumenVentas()

**Actores:** usuario, sistema

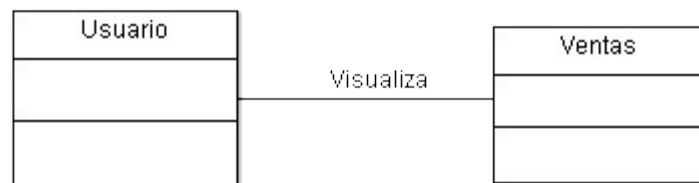
**Propósito:** gestionar un resumen de las ventas.

**Precondición:** existe una venta en el sistema.

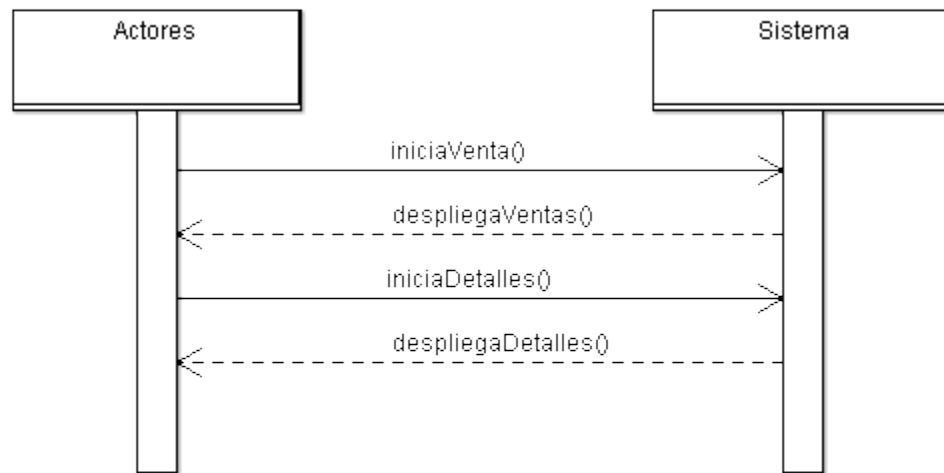
**Resumen:** Éste caso de uso comienza cuando el usuario selecciona la opción “Ventas” del menú principal. El sistema desplegará un resumen de las ventas existentes asociadas a clientes y productos.

1.- Éste caso de uso comienza cuando el usuario selecciona la opción “Ventas” del menú principal.	
	2.- El sistema despliega el resumen de las ventas realizadas, asociadas a cada cliente.
3.- El usuario selecciona un cliente y la opción “Detalles”.	
	3.- El sistema despliega en pantalla el detalle de las ventas realizadas a ese cliente.
4.- El usuario cierra el menú “Ventas”.	

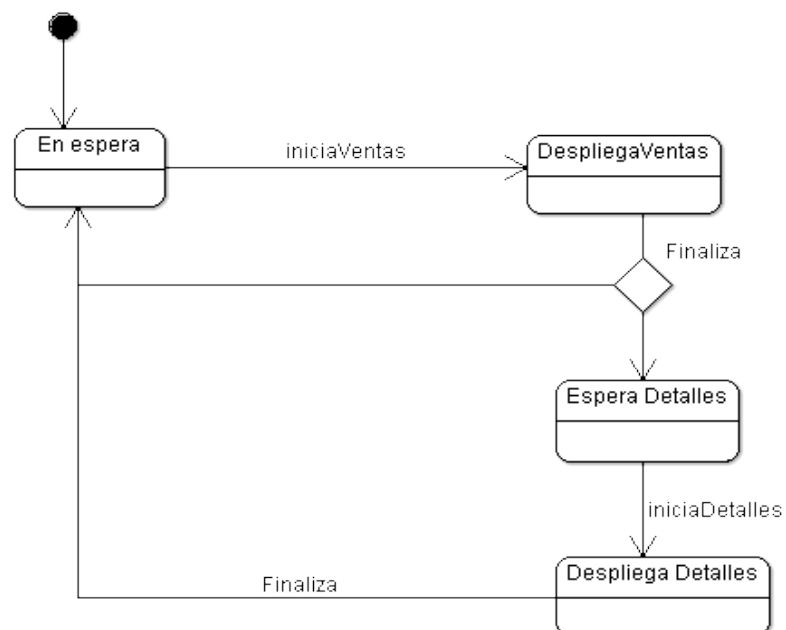
## **Modelo conceptual: resumenVentas()**



**Diagrama de secuencia: resumenVentas()**



**Diagrama de estado: resumenVentas()**





### **Contratos: resumenVentas()**

**Contrato:** iniciaVenta()

**Responsabilidad:** Mostrar información de ventas asociadas a los clientes en el sistema.

**Precondición:** Existen ventas en el sistema.

**Postcondición:** Se muestran/despliegan las ventas.

**Contrato:** iniciaDetalles()

**Responsabilidad:** Mostrar los detalles de una venta en particular.

**Precondición:** Existe la venta.

**Postcondición:** Se muestran/despliegan los detalles de la venta seleccionada.