**UNIVERSIDAD NACIONAL TECNOLÒGICA DE LIMA SUR**

**FACULTAD DE INGENIERÍA Y GESTIÓN**

**Escuela Académico Profesional de Ingeniería de Sistemas**

****

**SISTEMA DE VENTAS LICORERÍA ALVAREZ**

**ASIGNATURA:**

Testing y Calidad de Software

**DOCENTE:**

COBEÑAS SANCHEZ ROSA AMÉRICA

**ESTUDIANTES:**

ALVAREZ NUÑEZ, FEDERICO ANDRES

CHAMBI ROJAS, SAUL ALEXIS

IBARRA ZAVALA, DIEGO ALONSO

PALACIOS GUTIERREZ, ALEX DAVID

SANTOS BERNAL, DIANA LADY

TAIPE SILVA, DAVID ABRAHAM

**Villa el salvador  
2022**

LICORERÍA ALVAREZ

Documento de arquitectura de software

Versión <9.0>

Historia de revisiones

|  |  |  |  |
| --- | --- | --- | --- |
| **Fecha** | **Versión** | **Descripción** | **Autor** |
| 11/07/2021 | 1.0 | Versión preliminar como propuesta de desarrollo. | Sanchez Cardenas, Sergio |
| 13/07/2021 | 2.0 | Versión preliminar como propuesta de desarrollo. | Alvarez Nuñez, Federico |
| 15/07/2021 | 3.0 | Versión preliminar como propuesta de desarrollo. | Tupalaya Hernandez, Diego |
| 17/07/2021 | 4.0 | Versión preliminar como propuesta de desarrollo. | Bendezu Santisteban, Anthony |
| 20/07/2021 | 5.0 | Versión preliminar como propuesta de desarrollo. | Mansilla Menendez, Patrick |
| 22/07/2021 | 6.0 | Versión preliminar como propuesta de desarrollo. | Santos Bernal, Diana |
| 23/07/2021 | 7.0 | Versión preliminar como propuesta de desarrollo. | Taipe Silva, David |
| 26/07/2021 | 8.0 | Versión preliminar como propuesta de desarrollo. | Quispe Calixto, Javier |
| 28/07/2021 | 9.0 | Versión preliminar como propuesta de desarrollo. | Vite Cochachin, Sergio |

**Tabla de Contenido**

[**1. Introducción**](#_heading=h.ag1ndvdt4sk) **4**

[1.1 Propósito](#_heading=h.30j0zll) 4

[1.2 Alcance](#_heading=h.1fob9te) 4

[1.3 Definiciones, acrónimos y abreviaturas](#_heading=h.3znysh7) 5

[1.4 Referencias](#_heading=h.2et92p0) 6

[1.5 Resumen](#_heading=h.tyjcwt) 6

[2. Representación arquitectónica](#_heading=h.3dy6vkm) **6**

[3. Metas y limitaciones arquitectónicas](#_heading=h.1t3h5sf) **8**

[3.1 Herramientas de software utilizado](#_heading=h.4d34og8) 8

[3.1.1 Macromedia Dreamweaver 8](#_heading=h.2s8eyo1) 9

[3.1.2 Visual Studio Code](#_heading=h.g18akykh2m8o) 10

[3.1.3 MySQL](#_heading=h.17dp8vu) 11

[3.1.4 WampServer](#_heading=h.3rdcrjn) 11

[3.1.5 Rational Rose](#_heading=h.26in1rg) 12

[4. Vista de casos de uso](#_heading=h.35nkun2) **12**

[4.1 Realizaciones de casos de uso](#_heading=h.1ksv4uv) 14

[5. Vista lógica](#_heading=h.44sinio) **17**

[6. Vista de proceso](#_heading=h.2jxsxqh) **27**

[1. Introducción](#_heading=h.ag1ndvdt4sk) 4

[1.1 Propósito](#_heading=h.30j0zll) 4

[1.2 Alcance](#_heading=h.1fob9te) 4

[1.3 Definiciones, acrónimos y abreviaturas](#_heading=h.3znysh7) 5

[1.4 Referencias](#_heading=h.2et92p0) 6

[1.5 Resumen](#_heading=h.tyjcwt) 6

[2. Representación arquitectónica](#_heading=h.3dy6vkm) 6

[3. Metas y limitaciones arquitectónicas](#_heading=h.1t3h5sf) 8

[3.1 Herramientas de software utilizado](#_heading=h.4d34og8) 8

[3.1.1 Macromedia Dreamweaver 8](#_heading=h.2s8eyo1) 9

[3.1.2 Visual Studio Code](#_heading=h.g18akykh2m8o) 10

[3.1.3 MySQL](#_heading=h.17dp8vu) 11

[3.1.4 WampServer](#_heading=h.3rdcrjn) 11

[3.1.5 Rational Rose](#_heading=h.26in1rg) 12

[4. Vista de casos de uso](#_heading=h.35nkun2) 12

[4.1 Realizaciones de casos de uso](#_heading=h.1ksv4uv) 14

[5. Vista lógica](#_heading=h.44sinio) 17

[6. Vista de proceso](#_heading=h.2jxsxqh) 27

# 

# 

# 1. Introducción

Licorería Alvarez es un negocio dedicado a la venta de licores, se encuentra ubicado en el distrito de Lurín, el negocio se encuentra en constante innovación en ofertas; es por ello que el negocio ha ido aumentando sus ventas. Sin embargo, la atención al público se volvió lenta mayormente los fines de semanas y días festivos, llegando a formar una fila larga ocasionando que muchos de los clientes se vayan a la competencia; y todo esto se debe a que el negocio no cuenta con un software que agilice sus ventas.

Teniendo en cuenta lo mencionado anteriormente, el dueño del negocio está solicitando implementar un software que le permita agilizar sus ventas; para ello el sistema debe permitir generar una proforma, emitir una boleta, registrar despacho y gestionar los productos.

El objetivo principal del negocio es ofrecer a sus clientes un servicio eficiente, para que de esta forma el negocio pueda destacar en su distrito como el mejor comercializador de licores. Es por ello que a través de la implementación de este sistema se espera lograr las expectativas mencionadas, con el fin de que el negocio siga creciendo y sea reconocida por la calidad de servicio que brinda a sus clientes a través de sus productos.

## 1.1 Propósito

Proveer al dueño del negocio LICORERÍA ALVAREZ una herramienta de software que permita generar una proforma, emitir una boleta y gestionar productos, para que de esta manera agilice el proceso de ventas, lograr mayor cantidad de ventas y poder tener un mayor alcance con respecto a los clientes.

## 1.2 Alcance

* Diseñar las interfaces y crear la base de datos que permitan la interacción del usuario del sistema con el software de la forma más sencilla posible.
* Implementar proformas mediante un formulario en donde pueda ingresar el nombre del cliente, buscar el producto, agregar, editar o eliminar en la lista de productos de la proforma.
* Implementar boletas mediante la selección de una proforma emitida por el sistema
* Implementar la gestión de productos mediante un formulario en donde pueda ingresar el nombre del producto, precio y stock.

## 1.3 Definiciones, acrónimos y abreviaturas

Usuario: Utilizan una [interfaz de usuario](https://es.wikipedia.org/wiki/Interfaz_de_Usuario_Multiling%C3%BCe) para acceder a los sistemas, el proceso de identificación es conocido como identificación de usuario o acceso del usuario al sistema (login)

Controller(Controlador): El controlador es el encargado de conectar el modelo con las vistas, funciona como un puente entre la vista y el modelo, el controlador recibe eventos generados por el usuario desde las vistas y se encarga de direccionar al modelo la petición respectiva.

Shared: Es la carpeta compartida que tiene acceso a las vistas y clases que son usados por otros.

Boundary: Clase de interfaz o frontera que se utiliza para modelar la interacción entre el sistema y los actores.

Entity: Clase entidad, son usadas para modelar la información que tiene permanencia en el tiempo y es persistente.

Backup: Es una copia de los datos originales de un sistema de información o de un conjunto de [software](https://concepto.de/software/) ([archivos](https://concepto.de/archivo-informatico/), documentos, etc.) que se almacena en un lugar seguro o una región segura de la memoria del sistema, con el fin de poder volver a disponer de su [información](https://concepto.de/informacion/) en caso de que alguna eventualidad, accidente o desastre ocurra y ocasione su pérdida del [sistema](https://concepto.de/sistema-de-informacion/).

OD: Diagrama de objetos, representa una instancia de un diagrama de clase; donde se muestra los objetos, atributos y vínculos entre las distintas instancias.

SD: Diagrama de secuencia, donde se muestra a los objetos con su línea de vida. mensajes intercambiados entre los objetos ordenados secuencialmente y los scripts adicionales que aclaran el propósito de la interacción entre objetos.

## 1.4 Referencias

* *Rodríguez, H. (2014). Documento de arquitectura de software. Obtenido de* [*http://repositorio.cedia.org.ec/bitstream/123456789/952/1/Documento%20de%20Arquitectura%20CAD-CAM.pdf*](http://repositorio.cedia.org.ec/bitstream/123456789/952/1/Documento%20de%20Arquitectura%20CAD-CAM.pdf)
* *Vega, C. (s. f.). Documento de arquitectura de software. Academia. Recuperado 2 de agosto de 2021, de* <https://www.academia.edu/38585020/Documento_de_arquitectura_de_software>
* *Alvez P. , Foti F. , Scalone M. (2006). Documento de Arquitectura de Software. Obtenido de* [*http://materias.fi.uba.ar/7510/practica/zips/DocArquitecturaEj2.pdf*](http://materias.fi.uba.ar/7510/practica/zips/DocArquitecturaEj2.pdf)
* *Kendall, K., & Kendall, J. (2005). Análisis y diseño de sistemas. México. Obtenido de* [*https://ingenieriasoftware2011.files.wordpress.com/2011/07/analisis-y-disenio-de-sistemas-kendall-\_-kendall-6ta-edicion.pdf*](https://ingenieriasoftware2011.files.wordpress.com/2011/07/analisis-y-disenio-de-sistemas-kendall-_-kendall-6ta-edicion.pdf)

## 1.5 Resumen

El documento en un inicio está representado arquitectónicamente donde se hace la descripción metodológica de desarrollo en la que se trabaja, se detalla la estructura de las Vista de casos de uso, Vista lógica y Vista de componentes en donde cada vista cuenta con su paquete correspondiente.

Se describirán las metas y limitaciones arquitectónicas en donde se detalla las herramientas de software utilizado en el proyecto, el funcionamiento será mostrado con las vista lógica, el extracto se detalla en la descomposición general del modelo de diseño es dividido en jerarquía por paquetes, se muestra las paquetes de diseño de arquitectura, el detalle de cada paquete será con un diagrama de todas sus clases y sus paquetes.

Finalmente se mostrará las vistas de los procesos en donde se muestran los diagramas de diseño de objeto y secuencia de cada caso de uso.

# 2. Representación arquitectónica

La metodología de desarrollo es Rational Unified Process o RUP es una metodología de desarrollo que utiliza el enfoque orientado a objetos en su diseño, y está diseñado y documentado el uso de la notación Unified Modeling Language o UML para ilustrar los procesos en acción. Además, define un manojo entero de las actividades y de los elementos que se necesita elegir para construir el diagrama del proceso.

* Use case view
* Business use case
* Business Actor: BA cliente, BA gerente, BA izipay.
* Area atención al cliente
* BUC comprar producto : BAD comprar producto
* Vista general de los business use case del negocio LICORERÍA ALVAREZ
* Use case model
* Actors
* módulo gerencial
* módulo gestión de sistema
* módulo reclamo
* módulo seguridad
* módulo ventas
* paquetes
* Use Case Diagram
* Logical view
* AnalysisModel
* bundary
* controller
* entities
* Realización modulo gerencial
* Realización módulo gestión del sistema
* Realización módulo seguridad
* Realización módulo ventas
* Business Object Model
* RateancionCliente
* RcerrarCaja
* Business entities
* Business worker
* Design model
* model
  + - * + conexion\_singleton
        + conexion\_singleton\_Client
        + EntityDetalleUsuario
        + EntityUsuario
        + myiterator
        + strategicModels
* moduloGerencial
* moduloGestionSistema
* moduloReclamo
* moduloSeguridad
* moduloVentas
* shared
* cabeza\_singleton
* cabeza\_singleton\_Client
* formMensajeSistema
* formularioGeneral

# 3. Metas y limitaciones arquitectónicas

## 3.1 Herramientas de software utilizado

### 3.1.1 Macromedia Dreamweaver 8

Macromedia® Dreamweaver® 8 de Adobe, herramienta de desarrollo Web líder del sector, le permite diseñar, desarrollar y mantener sitios y aplicaciones Web basadas en las normas del sector de forma eficiente.

Algunas otras características que puedo mencionar de Dreamweaver son las siguientes:

1 Compatibilidad con CMS integrada

Ahora puedes tener una mayor compatibilidad de pruebas y creaciones de diversos marcos de trabajo de sistemas de gestión de contenido como WordPress, Joomla! y Drupal, interesante para aquellos blogueros que tienen que usar otros servicios y Dreamweaver a la vez.

2 Inspección de CSS

Ahora podemos ver en detalle el modelo de cuadro de CSS y cambie las propiedades de CSS sin necesidad de leer el código ni utilizar otra utilidad, ya era hora de que el programa tuviera más participación en cuanto a CSS ya que por mucho que les cueste a muchos CSS ahora es lo que le da estilo a los sitios web, HTML la estructura solamente.

3 Integración con Adobe BrowserLab

También contamos con la opción de poder utilizar Adobe BrowserLab que nos permitirá testear, de manera rápida y sencilla, nuestros sitios en diferentes versiones de navegadores. De esta manera, podremos ver en tiempo real cómo se representaría nuestro sitio en diversos navegadores web.

4 Sugerencias de código de clase personalizada de PHP

Otro cambio muy importante al menos para los usuarios de páginas dinámicas unidas a bases de datos como PHP y Mysql porque hay sugerencias de código personalizadas de código PHP

5 HTML5

En lo que se refiere a HTML5 y CSS3, Dreamweaver no los soporta en el modo de diseño, pero sí en vistas previas en vivo. Vale la pena recordar, que estos dos estándares están aún en definición, razón por la cual la gente de Adobe no se ha decidido a implementar esta función en la vista de diseño, lo que no significa que sirva en la vista código ya que en esa parte si lo reconoce.

### 3.1.2 Visual Studio Code

Es un editor de código fuente ligero pero potente que se ejecuta en su escritorio y está disponible para Windows, macOS y Linux. Viene con soporte incorporado para JavaScript, TypeScript y Node.js y tiene un rico ecosistema de extensiones para otros lenguajes (como C ++, C #, Java, Python, PHP, Go) y tiempos de ejecución (como .NET y Unity) . También es personalizable, de modo que los usuarios pueden cambiar el tema del editor, los métodos abreviados de teclado y las preferencias. Es gratuito y de código abierto.

* Se puede utilizar como lenguajes de programación.
* Visual Studio Code es una herramienta que tiene soporte nativo para gran variedad de lenguajes, entre ellos podemos destacar los principales del desarrollo [Web](https://www.ecured.cu/Web): [HTML](https://www.ecured.cu/HTML), [CSS](https://www.ecured.cu/CSS), y [JavaScript](https://www.ecured.cu/JavaScript), entre otros.
* Posibilidad de configurar la interfaz a nuestro gusto. De esta forma, podremos tener más de un código visible al mismo tiempo, las carpetas de nuestro proyecto y también acceso a la terminal o un detalle de problemas, entre otras posibilidades.
* Existencia de una amplísima gama de temas o estilos visuales para Visual Studio Code, que hacen el trabajo con el software más agradable a la vista.
* Goza de un soporte técnico formidable pues debido a su frecuente uso por la comunidad de desarrolladores, se puede encontrar fácilmente documentación y ayuda en foros y sitios relacionados.

### 3.1.3 MySQL

Sistema de gestión de base de datos relacional que está relacionado bajo la licencia dual (Licencia general Pública/ Licencia comercial por Oracle Corporation)

* Base de datos

La palabra “relacional” quiere decir que los datos se almacenan en forma de tablas, cada tabla se relaciona de alguna manera.

* Flexible y fácil de usar

Puedes modificar el código fuente sin costo, incluidas las opciones de actualización a la versión comercial avanzada. Su instalación es simple y no debería durar más de 30 min.

* Alto rendimiento

Un amplio compendio de servidores de clúster respalda a MySQL, ya sea que se estén almacenando ingentes cantidades de datos, MySQL cuenta con una velocidad óptima.

* Multiplataforma

Se puede instalar para SO tales como Windows, Linux y Mac.

### 3.1.4 WampServer

WampServer es uno de los WAMP más sencillos de utilizar, pues dispone de un panel de control muy completo. WAMP es un acrónimo que significa Windows, Apache, MySQL y PHP. Es un stack o conjunto de soluciones de software que significa que cuando instalas WAMP, estás instalando [Apache](https://www.hostinger.es/tutoriales/que-es-apache/), [MySQL](https://www.hostinger.es/tutoriales/que-es-mysql/) y PHP en tu sistema operativo (Windows en el caso de WAMP).

### 3.1.5 Rational Rose

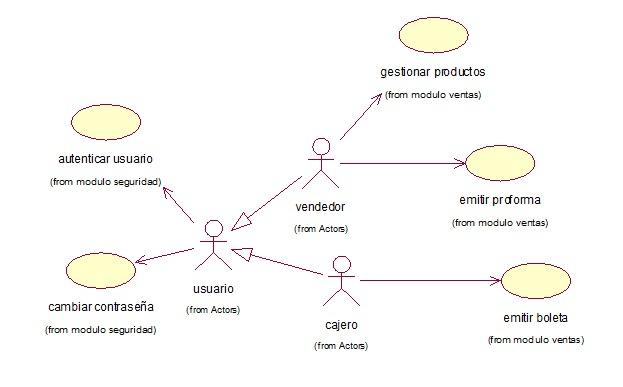
Es una [herramienta CASE](https://www.ecured.cu/Herramienta_CASE) (Computer – Arded [Software](https://www.ecured.cu/Software) Engineering), traducido al español como Ingeniería Asistida por Computadora, desarrollada por Rational Corporation basada en el [Lenguaje Unificado de Modelación](https://www.ecured.cu/Lenguaje_de_Modelaje_Unificado) ([UML](https://www.ecured.cu/Lenguaje_de_Modelaje_Unificado)), que permite crear los diagramas que se van generando durante el proceso de [Ingeniería en el Desarrollo del Software](https://www.ecured.cu/Ingenier%C3%ADa_de_Software).

* Soporte para análisis de patrones ANSI C++, Rose J y Visual C++ basado en "Design Patterns: Elements of Reusable Object-Oriented Software"
* La generación de código Ada, ANSI C ++, C++, CORBA, Java y Visual Basic, con capacidad de sincronización modelo- código configurables
* Soporte Enterprise Java Beans™ 2.0
* El Add-In para modelado Web provee visualización, modelado y las herramientas para desarrollar aplicaciones de Web
* Modelado UML para trabajar en diseños de base de datos, con capacidad de representar la integración de los datos y los requerimientos de aplicación a través de diseños lógicos y físicos

# 4. Vista de casos de uso

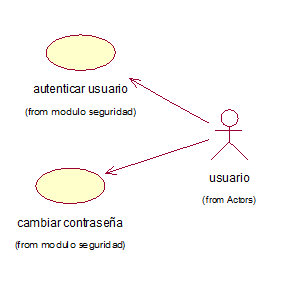
**Figura 1.**

**Diagrama de Casos de uso del Sistema**

****

**Figura 2.**

**Relación de actor usuario con el use case: Autenticar usuario y Cambiar contraseña**

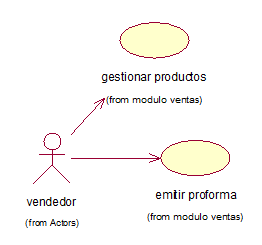


Autenticar usuario: Permite al usuario autenticarse usando el sistema.

Cambiar contraseña: Permite a un usuario registrado, cambiar la contraseña.

**Figura 3.**

**Relación de actor Vendedor con el use case: Gestionar productos y Emitir proforma**

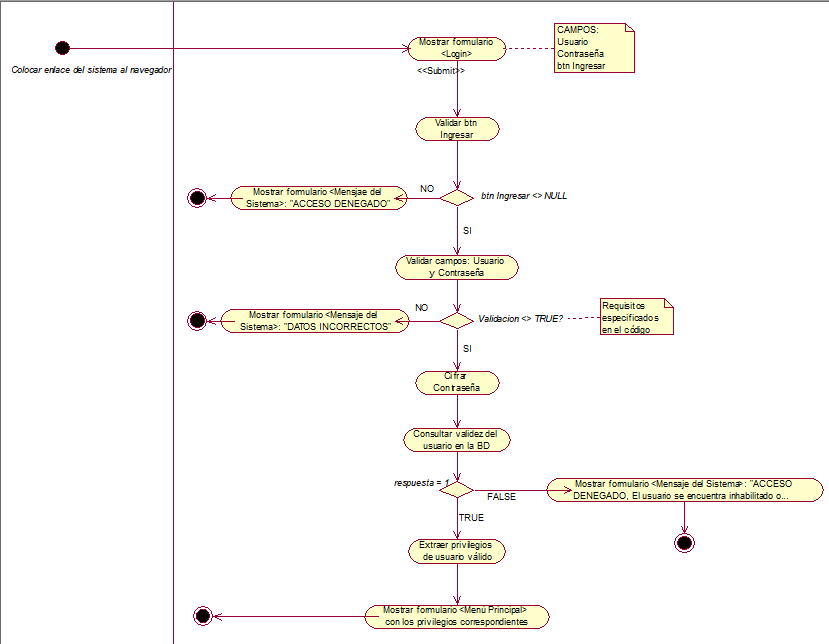
****

Gestionar productos: El sistema permite gestionar el stock de productos.

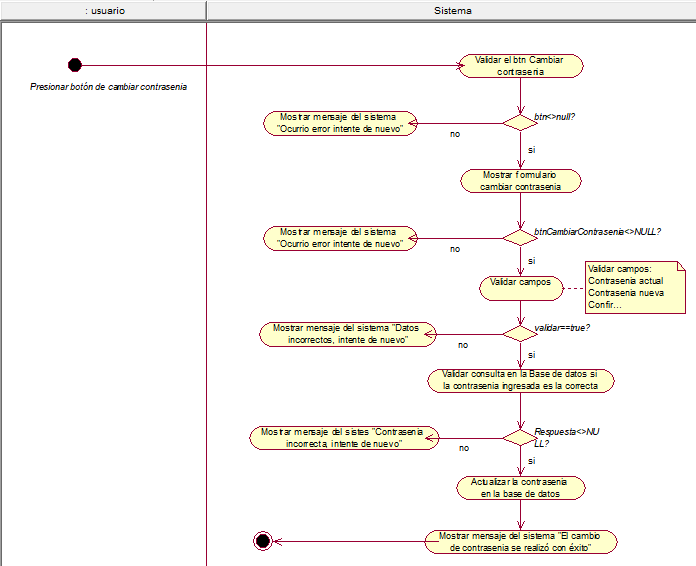
Emitir proforma: El sistema permite emitir proforma.

## 4.1 Realizaciones de casos de uso

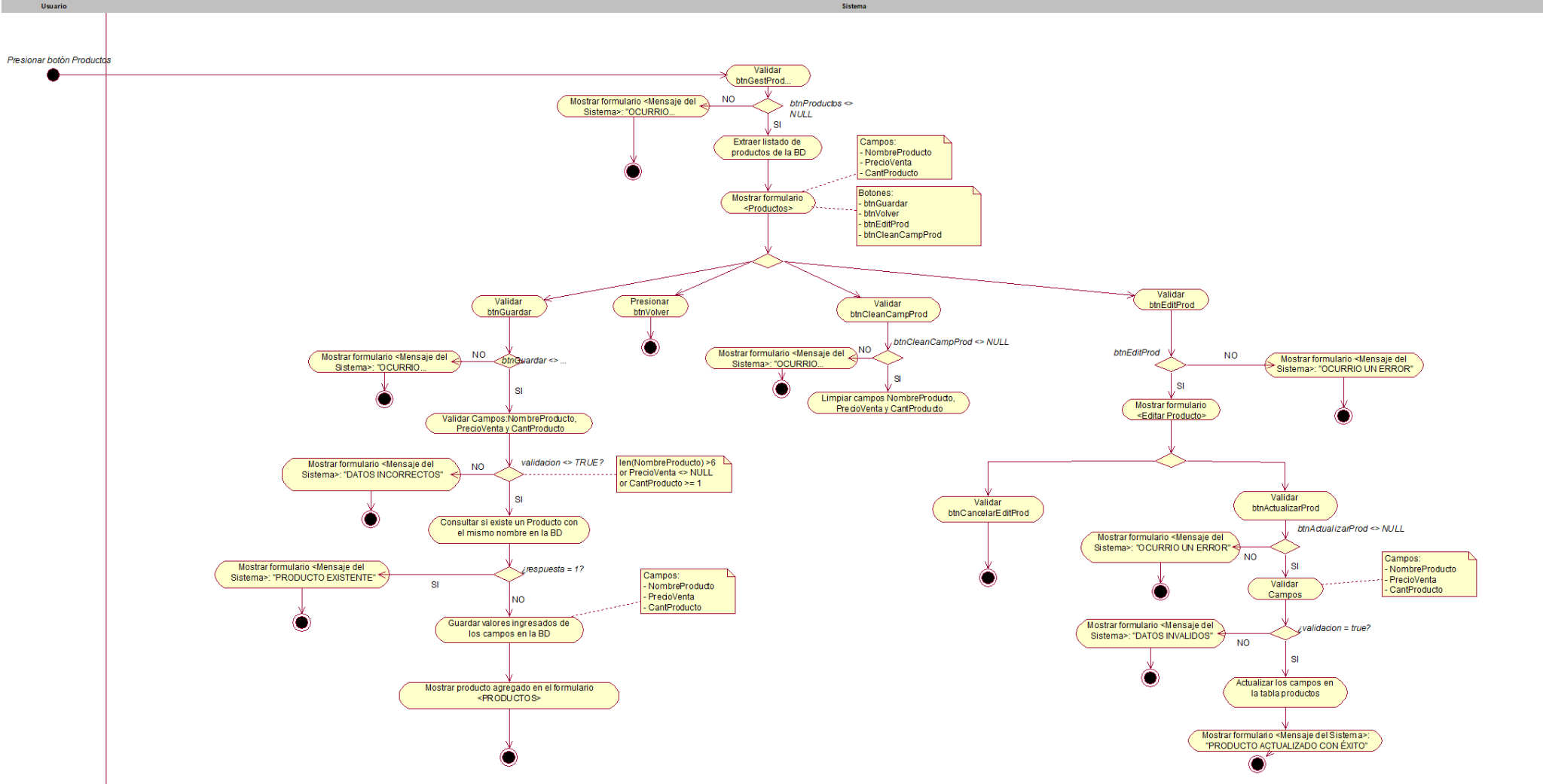
**Figura 5. Diagrama de actividades del UC Autenticar usuario**

****

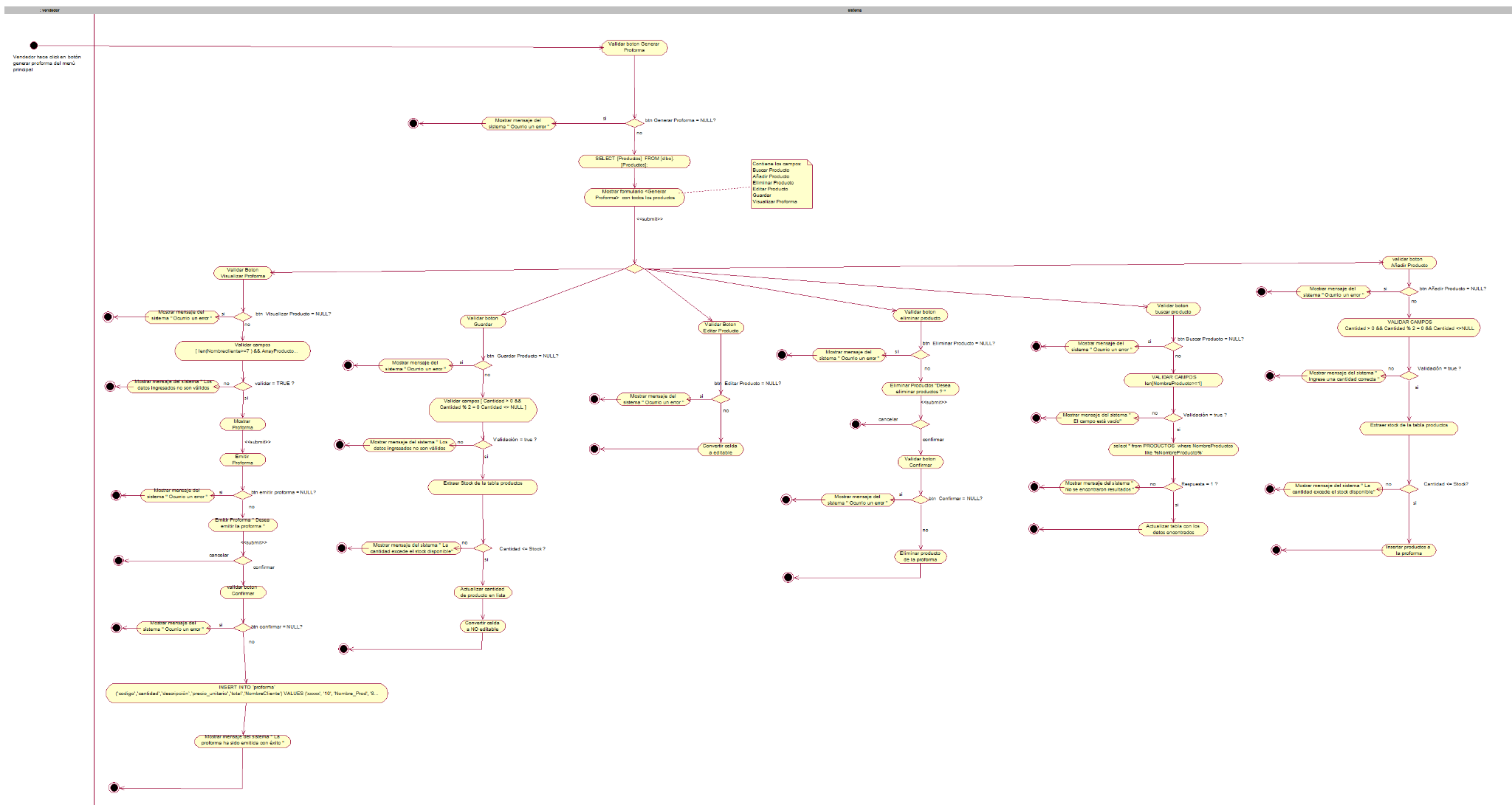
**Figura 6. Diagrama de actividades del UC Cambiar contraseña**



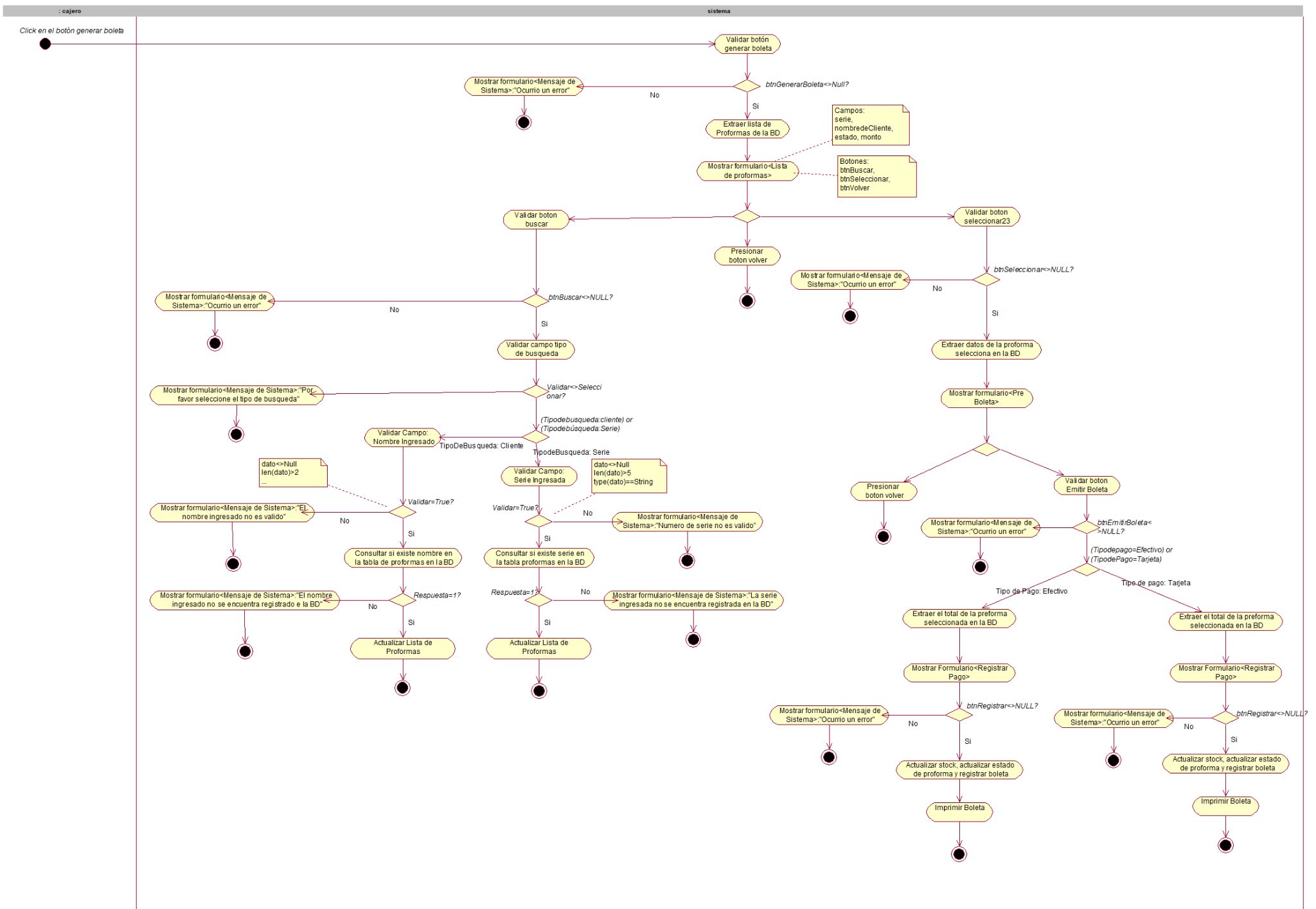
**Figura 7. Diagrama de actividades del UC Gestionar productos**

****

**Figura 8. Diagrama de actividades del UC Generar proforma**



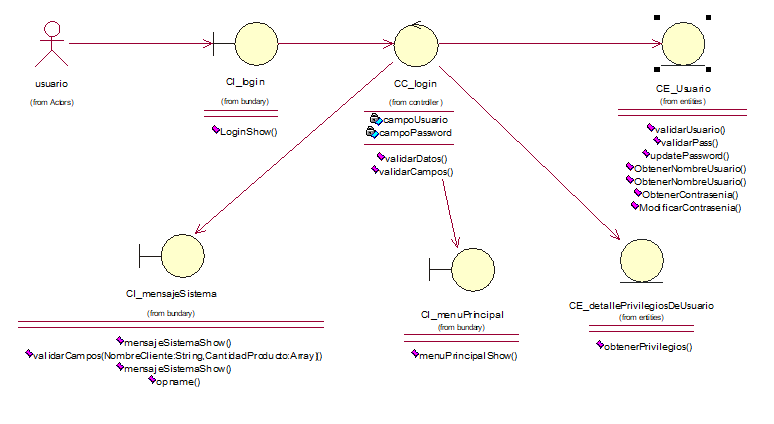
**Figura 9. Diagrama de actividades del UC Generar Boleta**

****

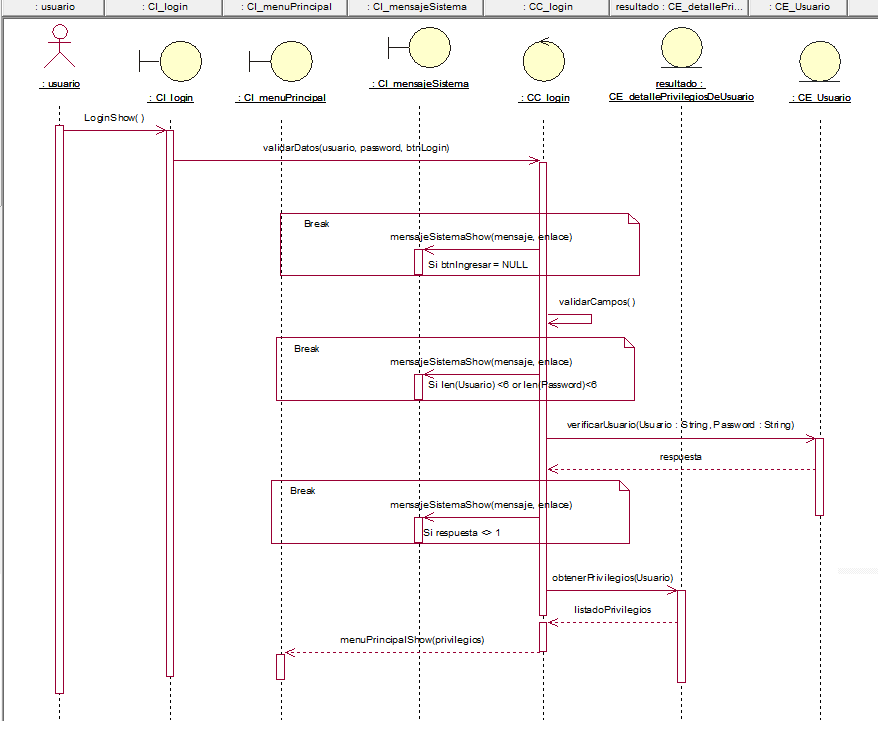
# 5. Vista lógica

Autenticar usuario

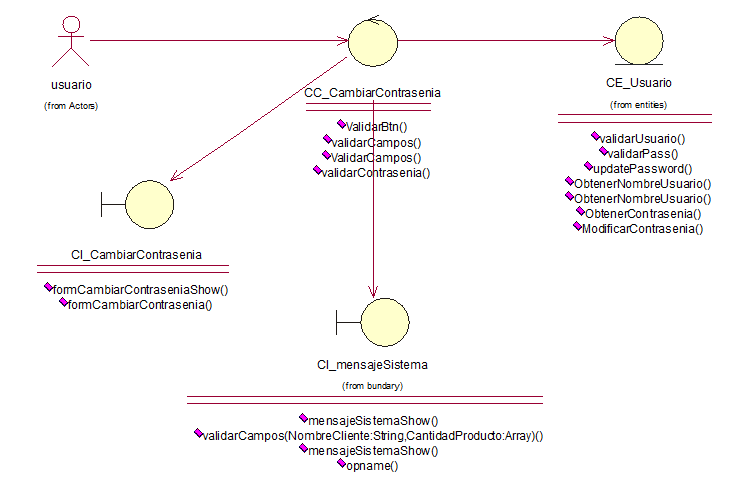
**Figura 10. Diagrama de objeto del UC Autenticar usuario**

****

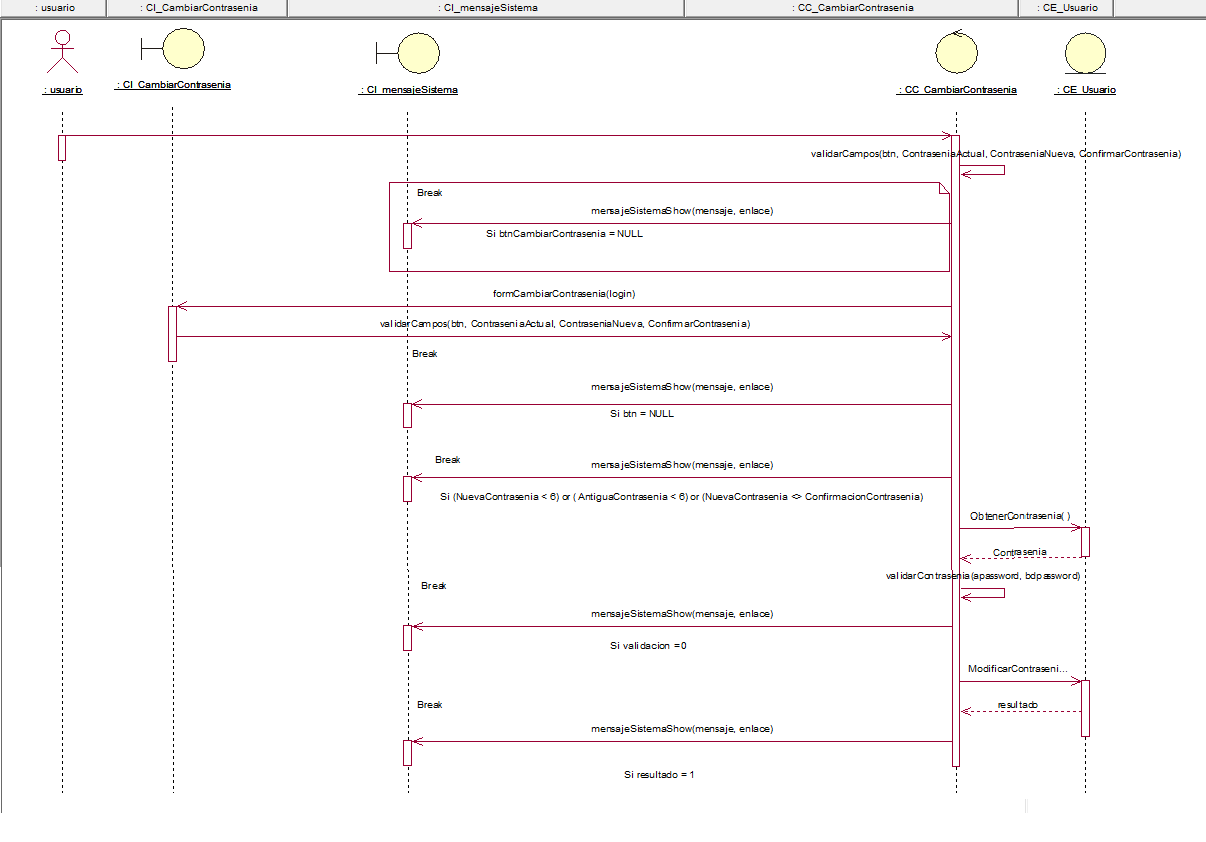
**Figura 11. Diagrama de secuencia del UC Autenticar usuario**



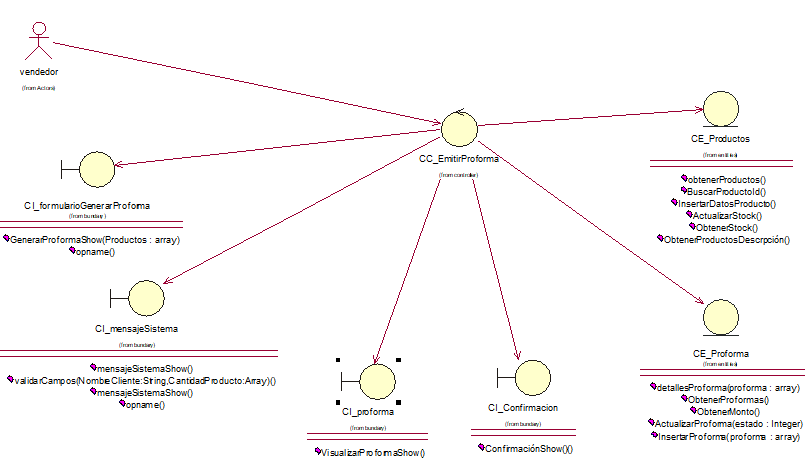
**Figura 12. Diagrama de objeto del UC Cambiar contraseña**

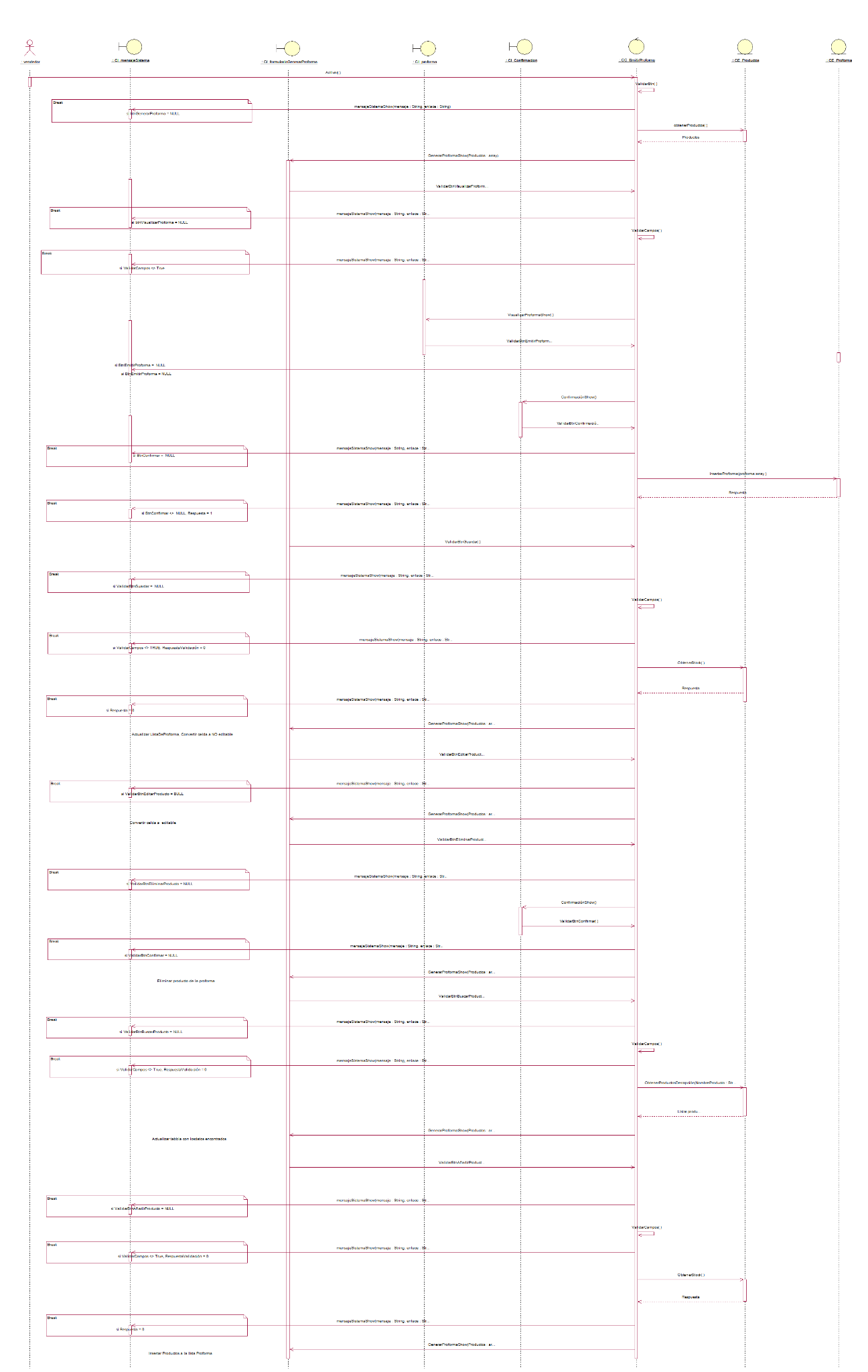


**Figura 13. Diagrama de secuencia del UC Cambiar contraseña**

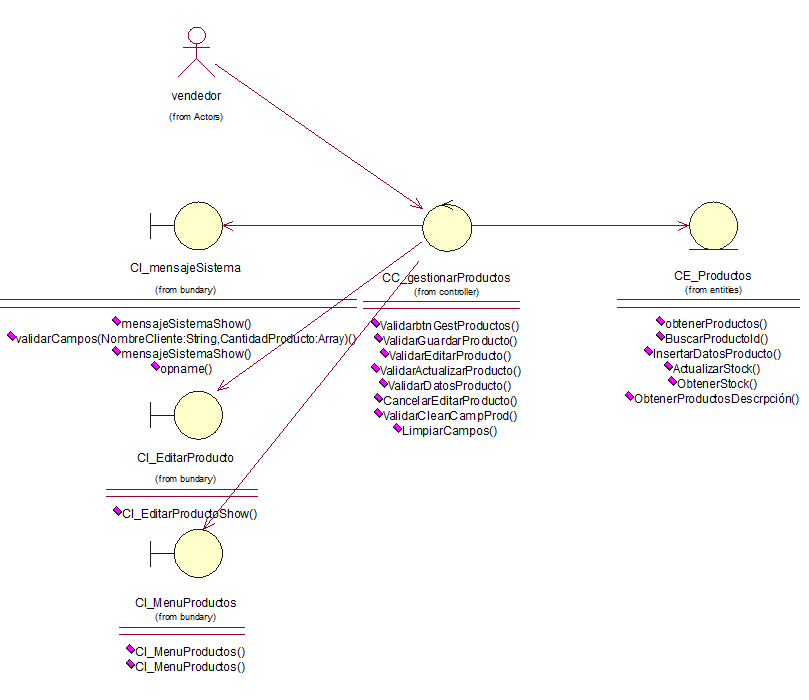


**Figura 14. Diagrama de objeto del UC Generar proforma**

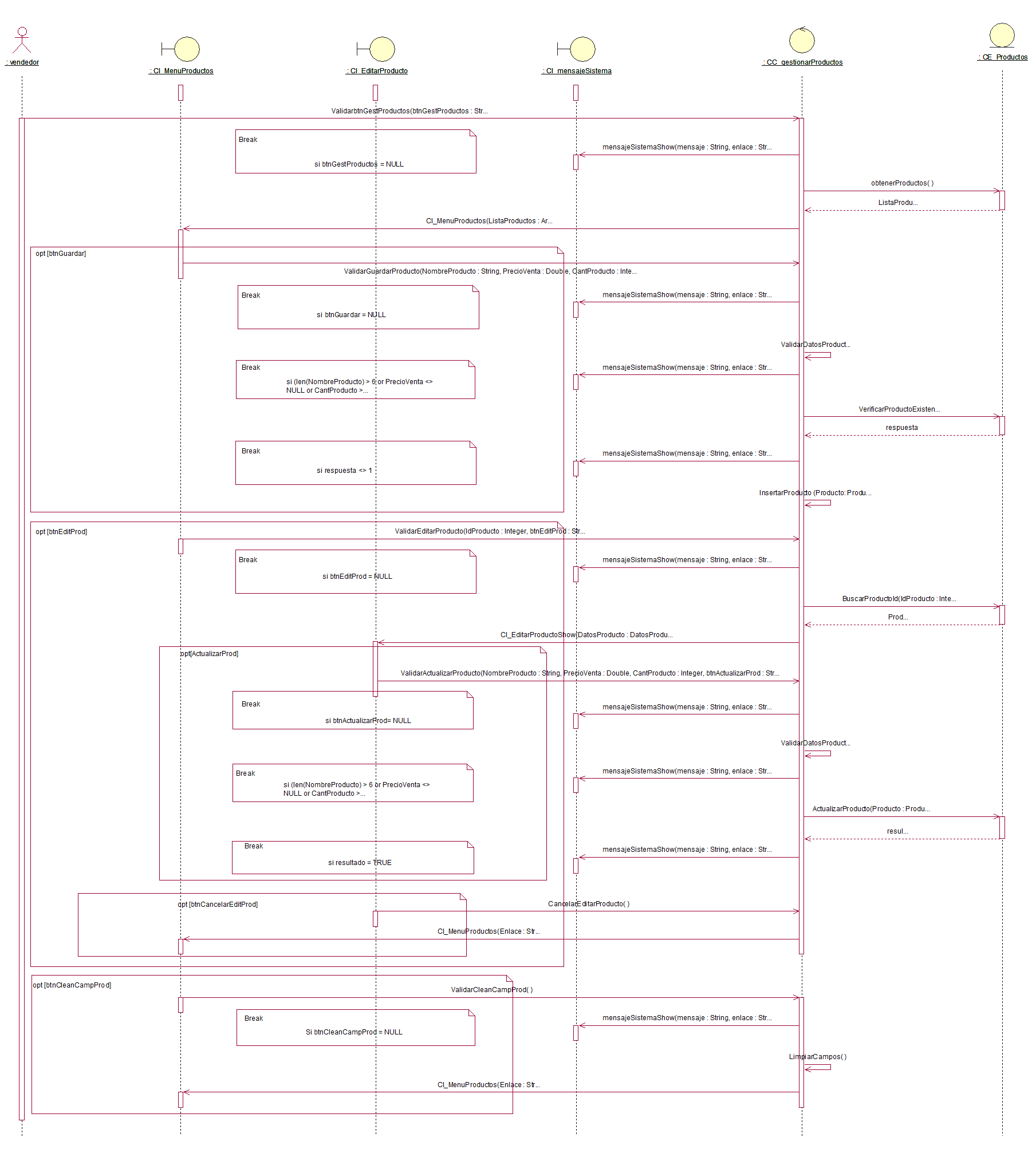


**Figura 15. Diagrama de secuencia del UC Generar proforma**

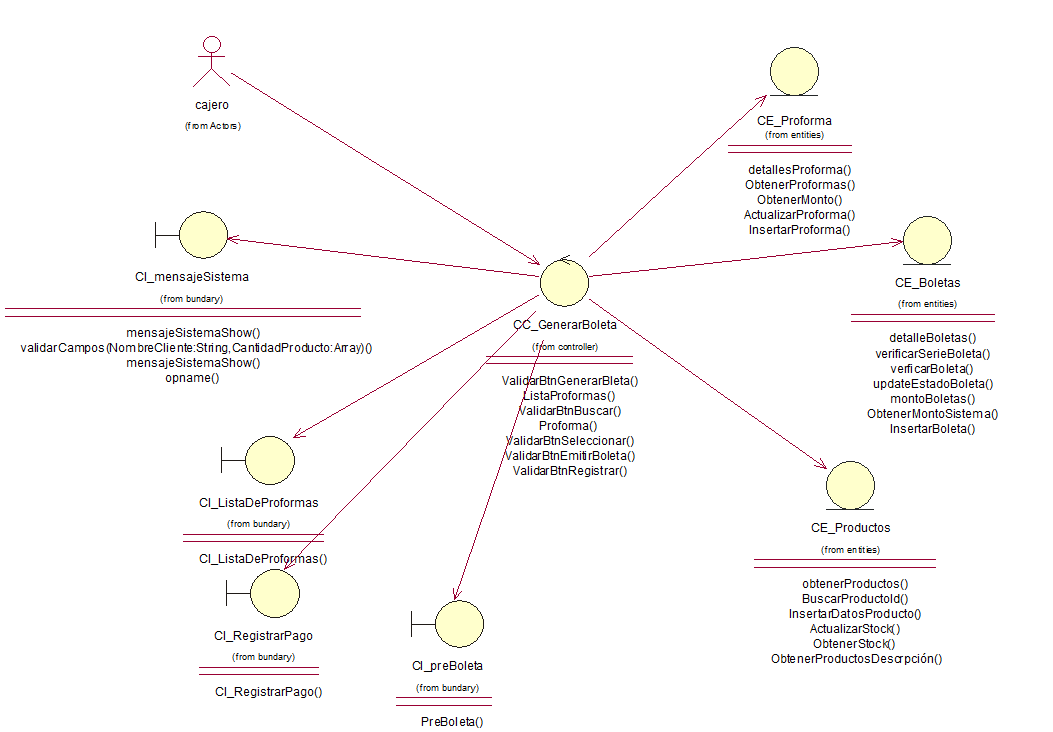
**Figura 16. Diagrama de objeto del UC Gestionar Productos**



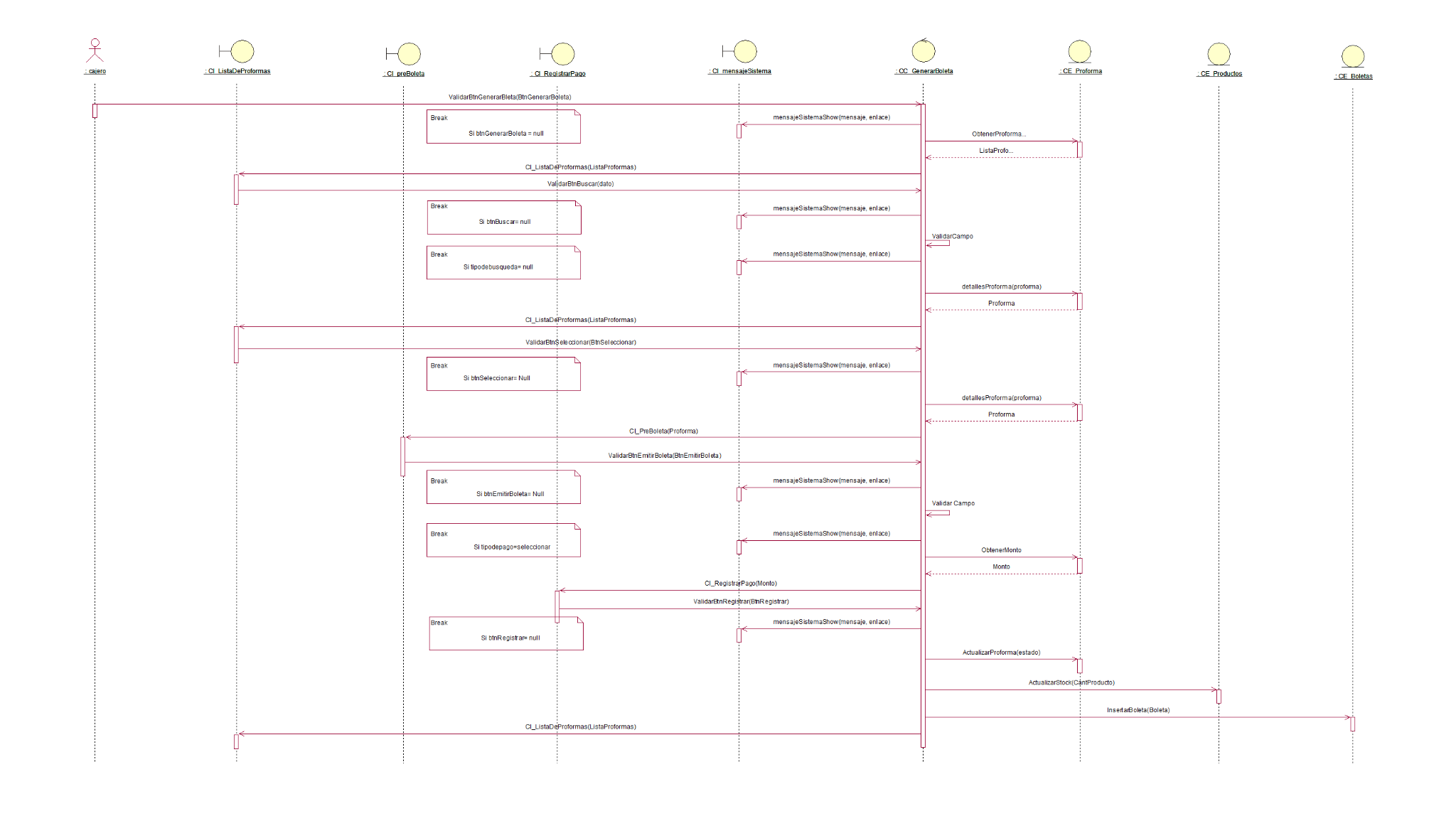
**Figura 17. Diagrama de secuencia del UC Gestionar productos**



**Figura 18. Diagrama de secuencia del UC Generar Boleta**

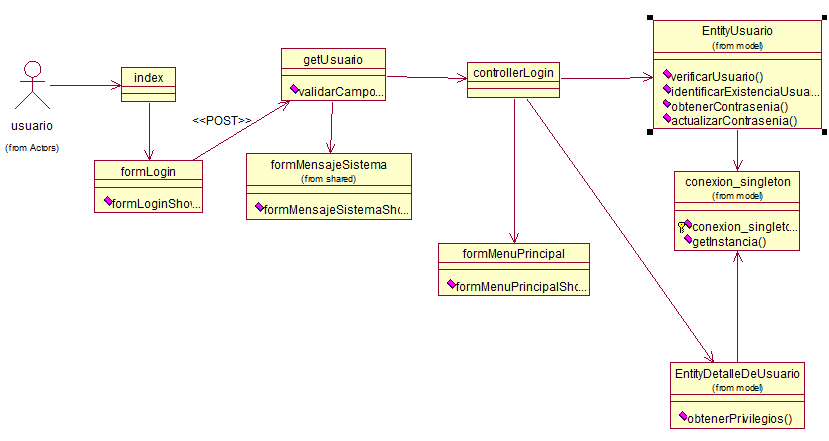
****

**Figura 19. Diagrama de secuencia del UC Generar Boleta**

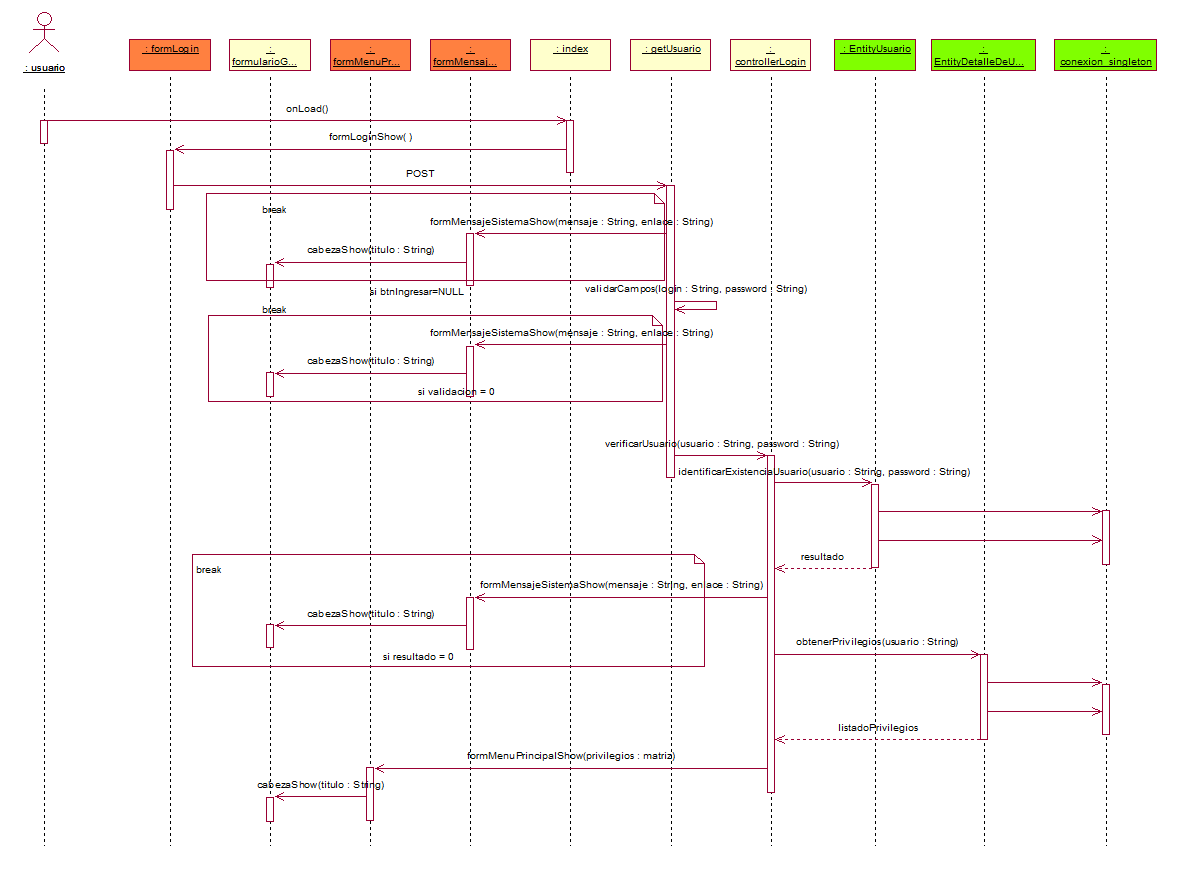


# 6. Vista de proceso

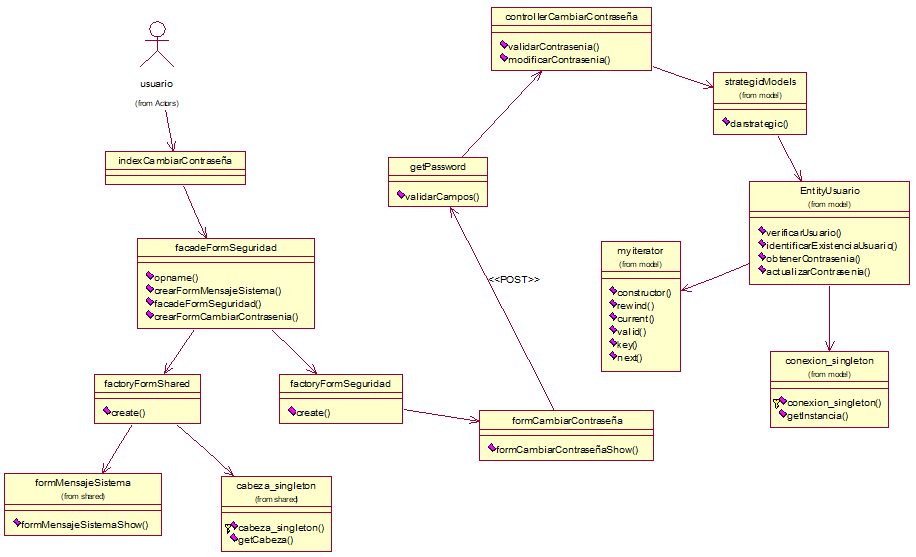
**Figura 21. Diagrama de objeto de diseño del UC Autenticar usuario**



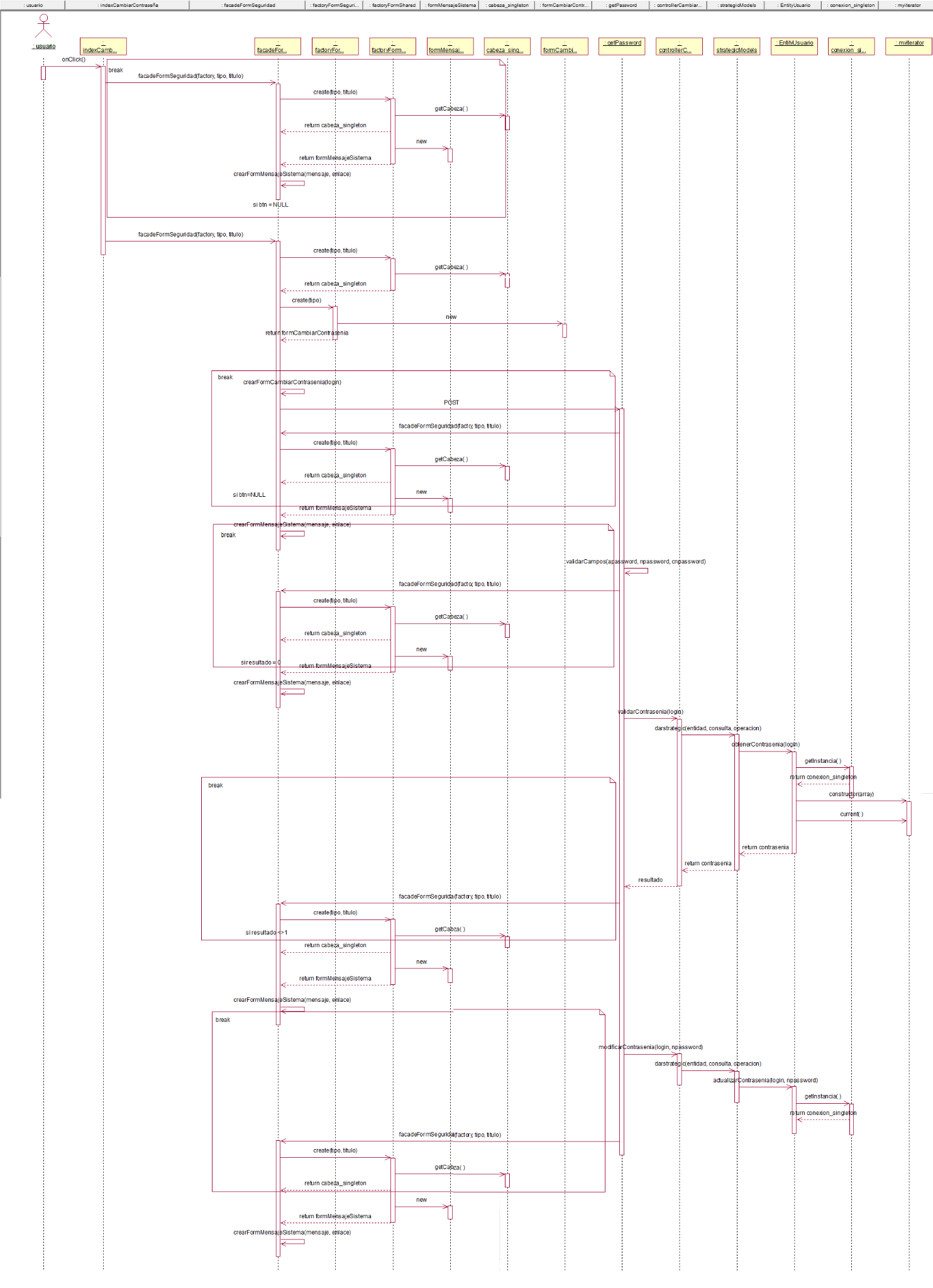
**Figura 22. Diagrama de secuencia de diseño del UC Autenticar usuario**



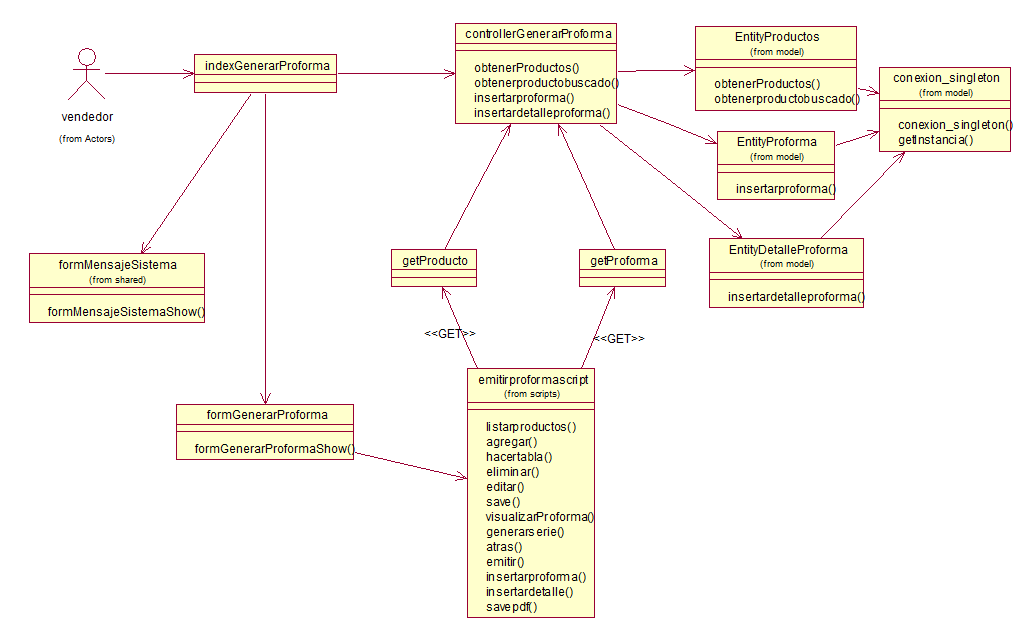
**Figura 23. Diagrama de objeto de diseño del UC Cambiar contraseña**



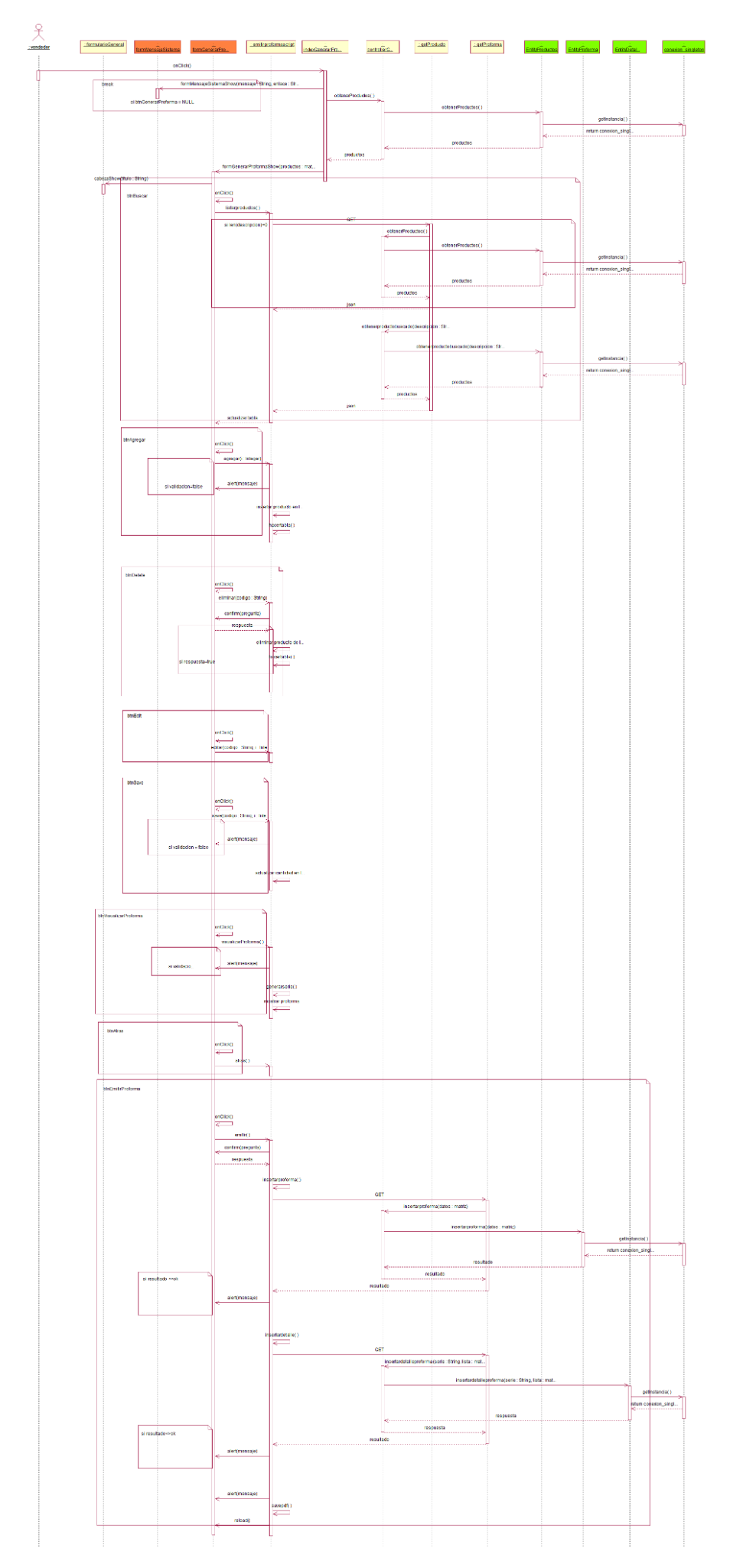
**Figura 24. Diagrama de secuencia de diseño del UC Cambiar contraseña**

****

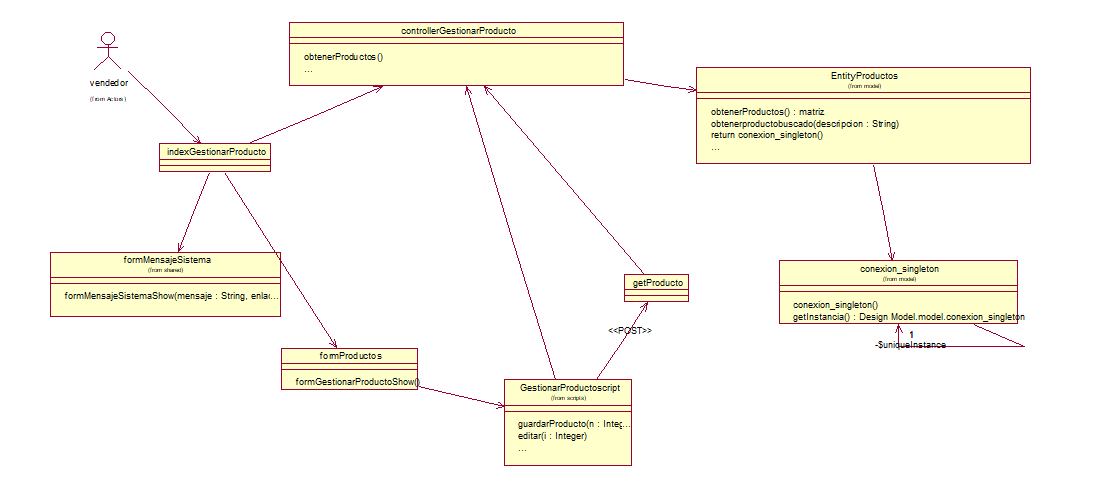
**Figura 25. Diagrama de objeto de diseño del UC Emitir Proforma**

****

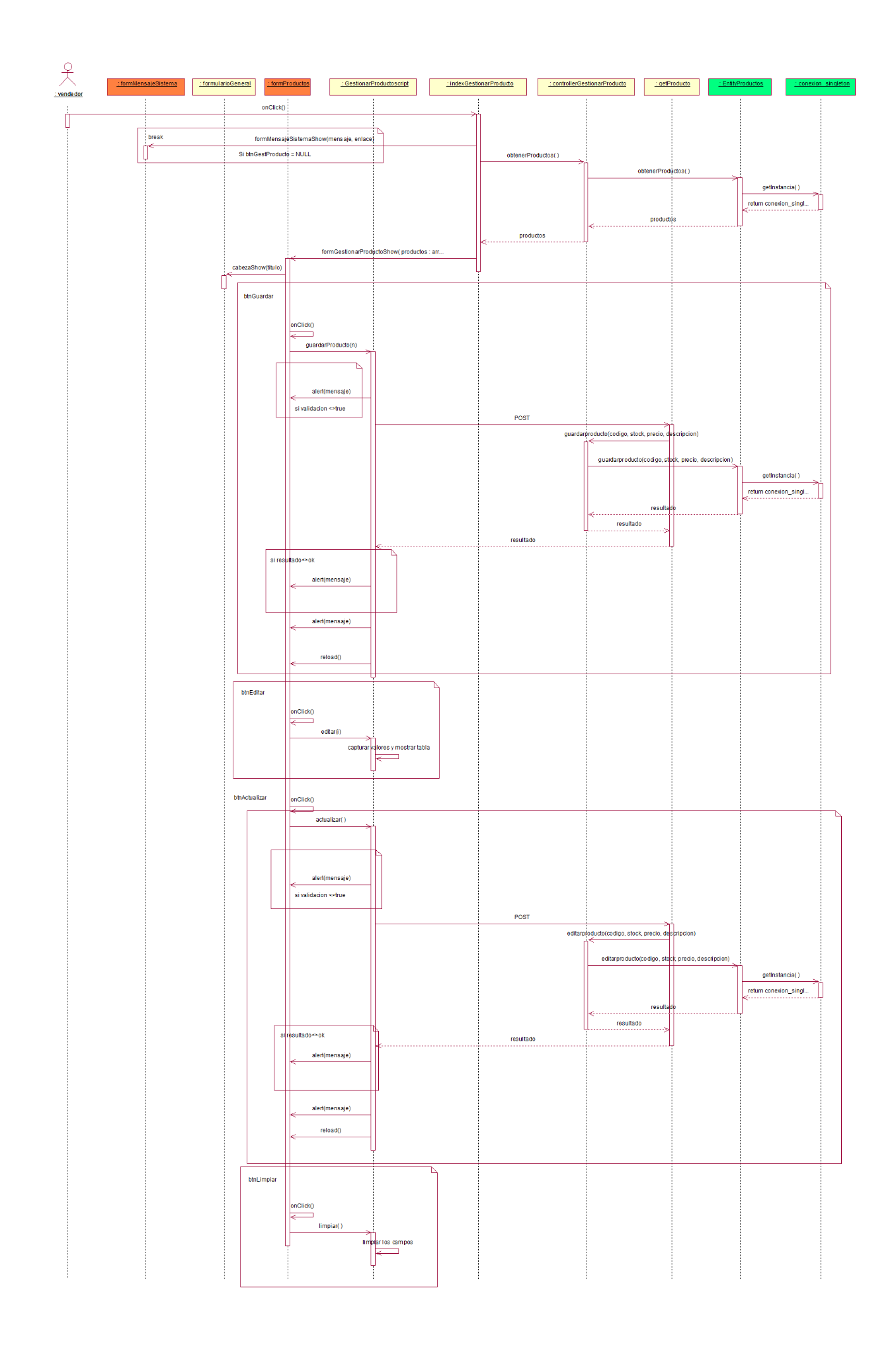
**Figura 26. Diagrama de diseño de secuencias del UC Emitir Proforma**



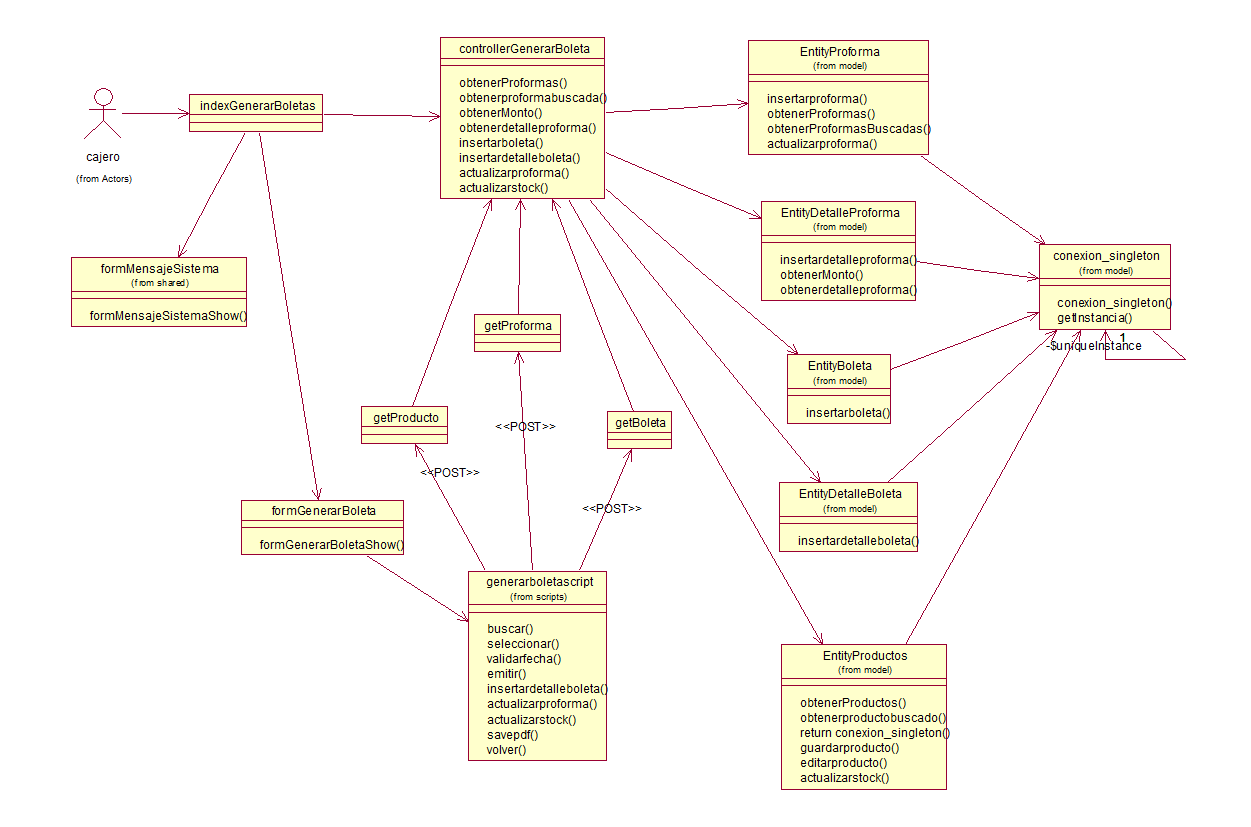
**Figura 27. Diagrama de objeto de diseño del UC Gestionar Productos**

****

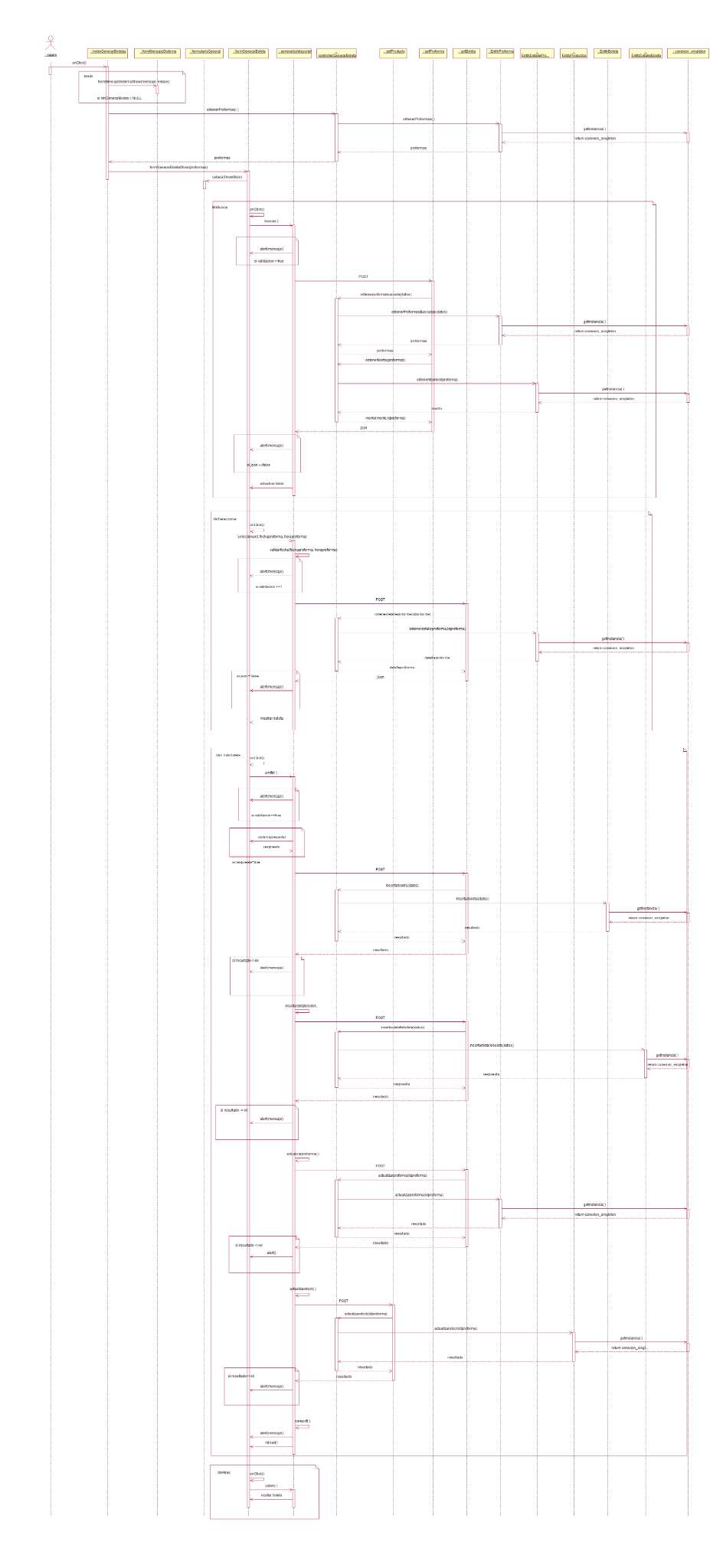
**Figura 28. Diagrama de diseño de secuencias del UC Gestionar Productos**

****

**Figura 29. Diagrama de objeto de diseño del UC Generar Boleta**

****

**Figura 30. Diagrama de diseño de secuencias del UC Generar Boleta**

****