

ANEXO 2: ESPECIFICACIONES PARA EL SISTEMA DE TRAZABILIDAD EN EL SECTOR DE LAS ARTESANÍAS



Trabajo de Grado

**ELIZABETH MACA GOMEZ
YULY VIVIANA MONTENEGRO SALCEDO**

Universidad del Cauca
Facultad de Ingeniería Electrónica y Telecomunicaciones
Departamento de Telemática
Línea de Servicios Telemáticos y
Ambientes de Desarrollo en Gestión Tecnológica
Popayán, Diciembre de 2005

TABLA DE CONTENIDO

ESPECIFICACIONES PARA EL SISTEMA DE TRAZABILIDAD EN EL SECTOR DE LAS ARTESANIAS	2
ARQUITECTURA DE LA APLICACIÓN DE TRAZABILIDAD	2
STRUTS: IMPLEMENTACIÓN DEL PATRÓN MVC EN APLICACIONES WEB	4
STRUTS.....	4
ESTRUCTURA.....	5
SERVICIO WEB.....	8
ARQUITECTURA DE LOS SERVICIOS WEB.....	9
ESTANDARES DE LOS SERVICIOS WEB.....	9
RED EPC.....	12
DESCRIPCIÓN GENÉRICA DE COMPONENTES DE LA RED EPC	15
CÓDIGO EPC.....	15
TAGS.....	16
LECTORES.....	16
SOFTWARE MEDIADOR.....	17
ONS - OBJECT NAME SERVICE	17
PML – PHYSICAL MARKUP LANGUAGE	17
MODELO ESENCIAL DE CASOS DE USO DEL SISTEMA DE TRAZABILIDAD.....	19

ESPECIFICACIONES PARA EL SISTEMA DE TRAZABILIDAD EN EL SECTOR DE LAS ARTESANIAS

ARQUITECTURA DE LA APLICACIÓN DE TRAZABILIDAD

Para la aplicación en Corseda se ha utilizando Struts en el ámbito de desarrollo web, la arquitectura se puede observar en las Figuras 1 y 2, donde claramente se identifica el patrón Modelo Vista Control (MVC) para los actores administrador y operario respectivamente.

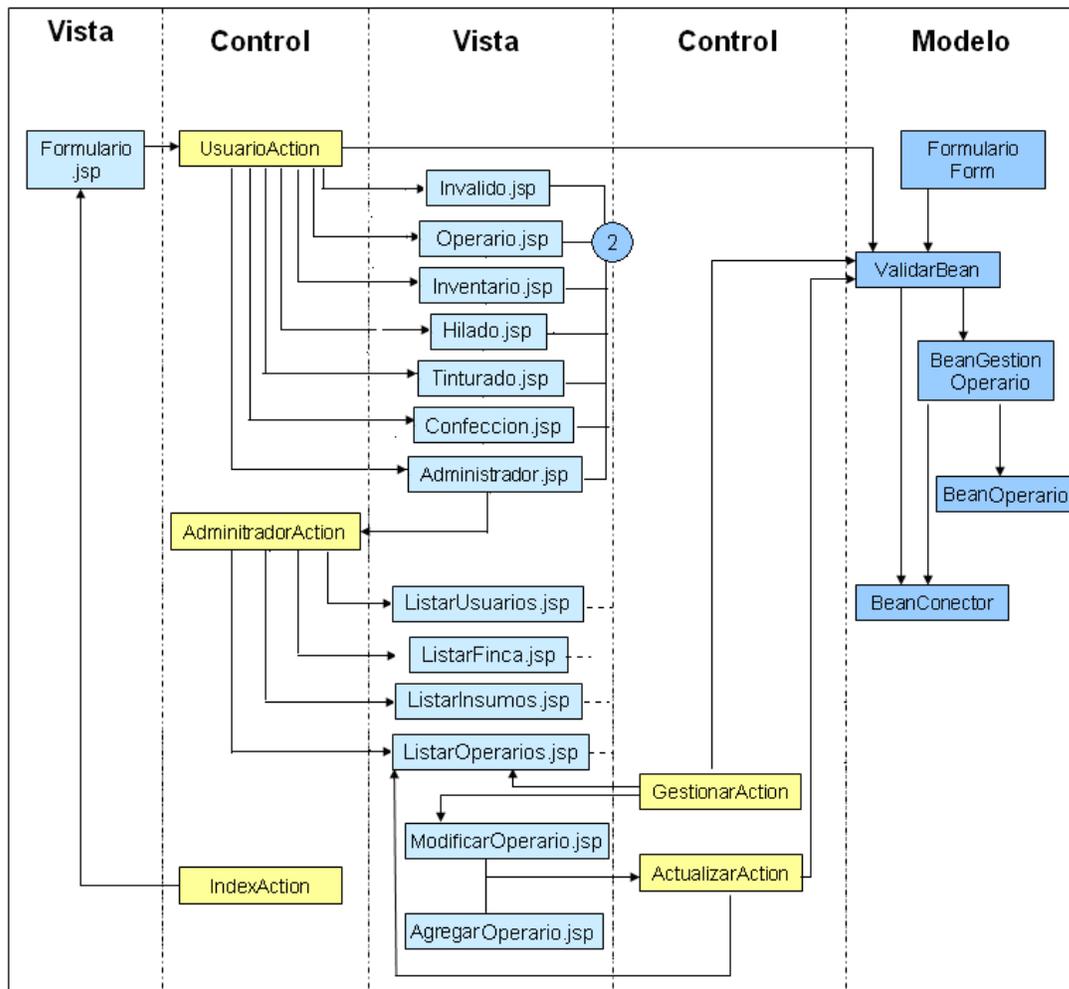


Figura 1. Estructura de la Aplicación de trazabilidad - Administrador

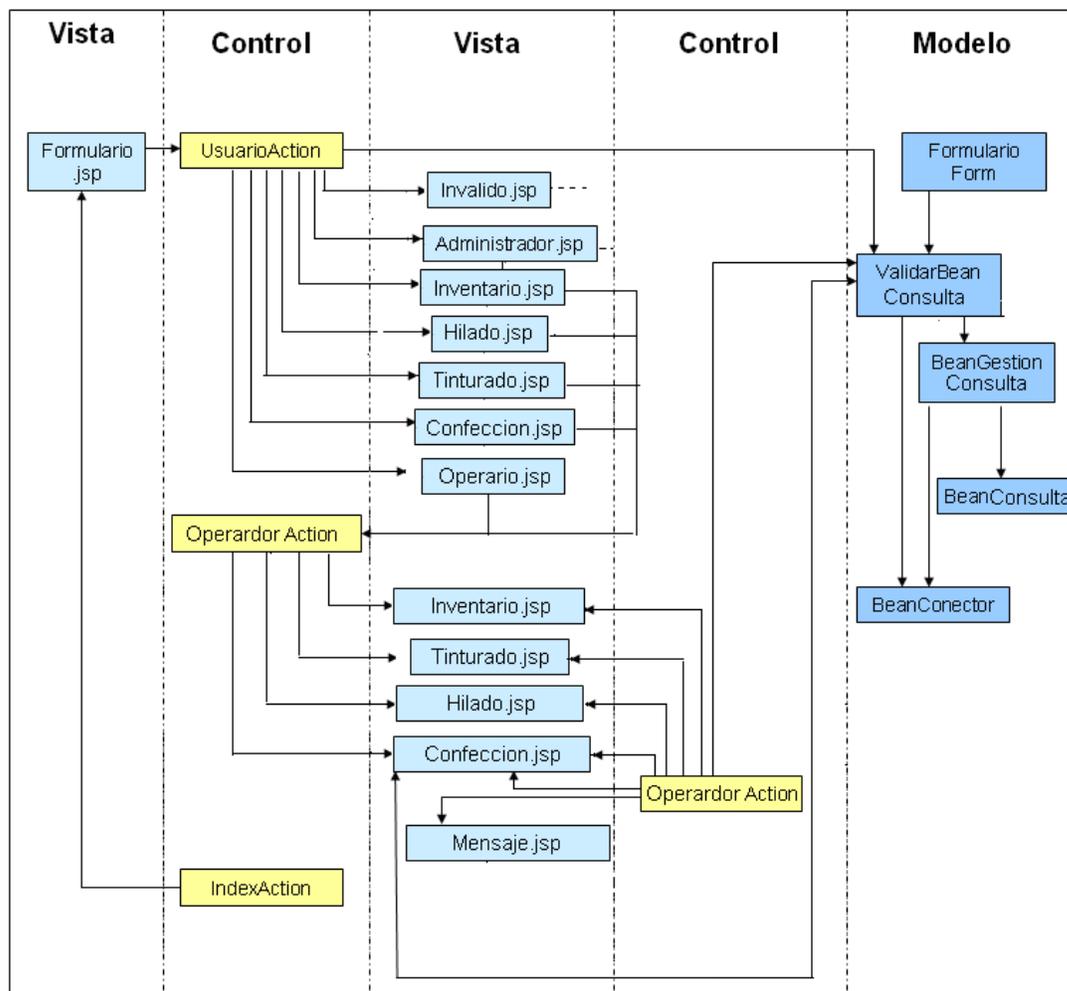


Figura 2. Estructura de la Aplicación de trazabilidad - Operario

Para las aplicaciones de Comercializadora y Transporte, la estructura con la que se trabajó es similar, sin embargo dichas aplicaciones se elaboraron únicamente con Servlets bajo el patrón Modelo Vista Control.

La Figura 3 muestra la estructura de la aplicación Corseda cuando un cliente realiza la consulta sobre un producto y de esta forma obtener el reporte de trazabilidad del mismo.

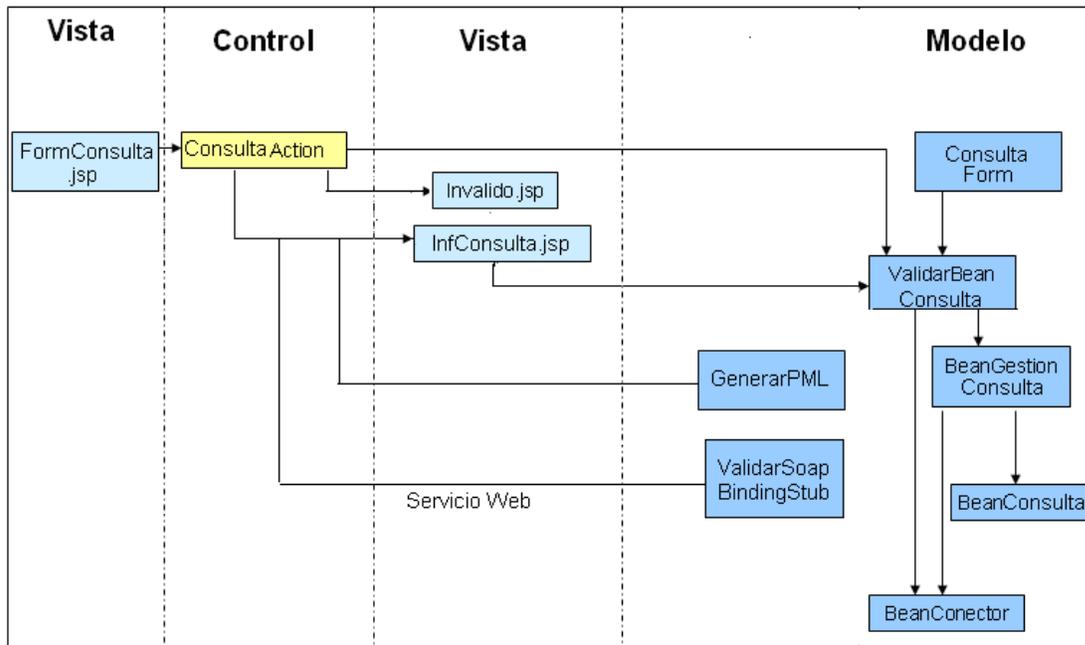


Figura 3. Estructura de la Aplicación de trazabilidad - Consulta

STRUTS: IMPLEMENTACIÓN DEL PATRÓN MVC EN APLICACIONES WEB

Se describe el patrón de diseño MVC (Modelo Vista Control), utilizado para la realización de la Aplicación a Corseda, donde se utiliza el framework Struts, que facilita el desarrollo de la aplicación web en Java bajo dicho patrón de diseño.

Utilizando Struts en el ámbito de desarrollo web se siguen unas pautas para conseguir un desarrollo completamente estructurado de las aplicaciones ya que se divide perfectamente la lógica de negocio (Modelo), la presentación (Vista) y control de flujo de aplicaciones (Control).

STRUTS

Es un framework para aplicaciones web java que implementa el modelo MVC. Realmente lo que provee es un conjunto de clases y TAG-LIBS que conforman el Controlador, la integración con el Modelo (o lógica de negocio) y facilitan la construcción de vistas.

Naturalmente, el Modelo o lógica de negocio es la parte que nos corresponde desarrollar. Por eso Struts es una plataforma sobre la que montamos la lógica de negocio, y esta plataforma nos permite dividir la lógica de la presentación entre otras cosas.

ESTRUCTURA

Utilizando Struts nunca se llega a una página de la capa de presentación directamente. Esto es, en la url nunca se llega a una página jsp o html a través de su nombre. De eso se trata el MVC, la presentación esta separada en otra capa.

En este entorno, se debe invocar una acción o aplicación que debe estar traducida a Struts: una acción se corresponderá con una clase Java (heredera de la clase Action de Struts). El mapeo de acciones y clases se especifica en un fichero de importancia vital: **strutsconfig.xml**. Ahí se especifican todas las relaciones entre acciones y clases, formularios y clases, acciones y jsps de presentación, que globalmente conforman el “mapa” de la aplicación.

Si se quisiera crear la aplicación Struts más simple posible, por ejemplo una página con un saludo, debiéramos hacer lo siguiente:

1. -Una página JSP (la presentación)
2. -Una clase Action (componente del controlador)
3. -La clase Action debe definirse en el struts-config.xml correctamente

Diagrama de clases

En el caso más simple este sería el conjunto de clases utilizadas y sus relaciones.

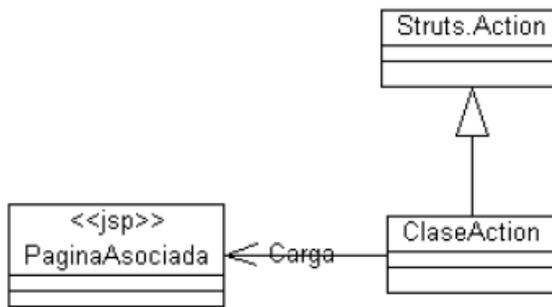


Figura 4. Diagrama de clases Struts

Como se puede ver, no tendríamos más que dos ficheros, una clase que hereda de struts.Action y una JSP con la presentación. La clase Action se configura en el struts-config.xml y se convierte en parte del controlador. Cuando la aplicación recibe una petición, Struts decidirá que debe cargar esa clase y través de ella cargará la JSP. Así se puede ver en el siguiente diagrama:

Diagrama de colaboración

Esta sería la manera de interactuar entre la clases y Struts

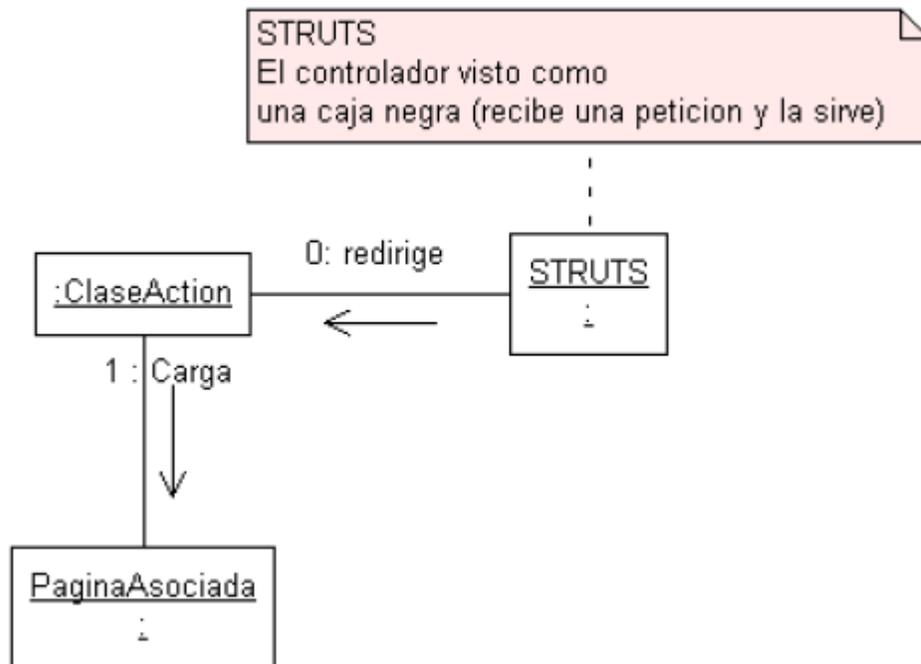


Figura 5. Diagrama de colaboración Struts

Una petición llega a struts (una URL); este mira en su “mapa” (el fichero struts-config.xml), y deduce que tiene que cargar la ClaseAction. Esta clase esta configurada para que cargue una pagina JSP.

Como se puede observar, no se carga la JSP directamente, hay que pasar por el controlador

Diagrama de secuencia

Estos serían los pasos seguidos por la aplicación en el plano temporal.

Obviamente, este no es el ejemplo más clásico de una acción de Struts. Por lo general, en cualquier aplicación web siempre se sigue un mismo esquema:

1. se muestra un formulario
2. se rellena
3. se validase
4. se manda a una página que realiza la lógica de negocio
5. se muestra el resultado.

En Struts necesitaríamos lo siguiente:

1. -Una clase Action encargada de cargar el formulario.
2. -Un Bean tipo Form cuyos campos u atributos de instancia coinciden con los campos del formulario. (de tipo JavaBean y tendrá los correspondientes métodos set y get para los atributos) . Este form y sus atributos son un requisito imprescindible.
3. -Una JSP que contiene el formulario citado. Los campos deben coincidir con los definidos en el Bean de formulario
4. -Una clase Action encargada de pasar la instancia del formulario al Bean de Logica de Negocio, esperar su resultado y redirigir el flujo a una JSP de error o de éxito.
5. -Un Bean de lógica de negocio, procesa los datos del formulario

6. -Una JSP para mostrar un mensaje de resultado en caso de éxito en la operación

7. -Una JSP para mostrar los errores en caso de error

SERVICIO WEB

Los servicios Web son módulos de software que realizan tareas o conjuntos de tareas discretas, a los que se accede y se llama a través de una red, sobre todo la World Wide Web. El desarrollador puede crear una aplicación cliente que llama a una serie de servicios Web mediante llamadas a procedimientos remotos (RPC) o un servicio de mensajes en los que se proporciona un fragmento o la mayor parte de la lógica de la aplicación.

Los servicios web publicados se describen de forma que los desarrolladores pueden localizarlos y determinar si se ajustan a sus necesidades. Por ejemplo, para nuestro caso objeto de estudio, comercializadora y transporte proporcionan a sus clientes un servicio Web por el cual se realiza un seguimiento del producto a través de la cadena de abastecimiento, mediante el cual los clientes pueden obtener información respecto a la historia del producto y su paso por dichas empresas.

Los servicios web utilizan SOAP (Simple Object Access Protocol) para la carga XML, así como un sistema de transporte de tipo HTTP para transmitir los mensajes SOAP de un sitio a otro. En realidad, los mensajes SOAP son los documentos XML que se envían entre el servicio web y la aplicación que efectúa la llamada.

Los servicios web se pueden escribir en cualquier lenguaje, y se ejecutan en todas las plataformas. Los clientes de servicios web también se pueden escribir en cualquier lenguaje, y se ejecutan en todas las plataformas. Así, por ejemplo, un cliente escrito en Delphi que se ejecuta en Windows puede llamar a un servicio web escrito en Java que se ejecuta en Linux.

ARQUITECTURA DE LOS SERVICIOS WEB

La arquitectura de servicios web tiene tres cometidos: proporciona datos, los solicita y hace de intermediario. Como proveedor, crea el servicio web y lo pone a disposición de los clientes que desean utilizarlo, quienes realizan la solicitud para consumir el servicio web. Este servicio solicitado también puede ser un cliente de otros servicios web. El intermediario, como un registro de servicio, permite la interacción entre la aplicación cliente y al proveedor.

Estos tres cometidos interactúan por medio de las operaciones de publicación, búsqueda y enlace. El proveedor utiliza la interfaz de publicación del intermediario para comunicarle la existencia del servicio web, con el fin de ponerlo a disposición de los clientes. La información publicada describe el servicio e indica dónde se encuentra. La aplicación que hace la solicitud consulta al intermediario para que busque los servicios web publicados. Con la información obtenida del intermediario, la aplicación cliente puede enlazarse al servicio web (llamarlo). En el diagrama 6 se resume cómo se realiza la interacción entre proveedor, solicitante y gestor.

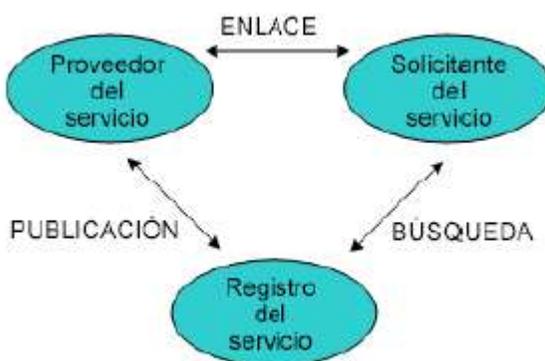


Figura 6. Operaciones y cometidos de los servicios web

ESTANDARES DE LOS SERVICIOS WEB

Las normas en las que se basa el desarrollo de servicios web son tecnologías en proceso de evolución. Las principales son SOAP (Simple Access Protocol, Protocolo de acceso a objetos sencillos), WSDL (Web Services Description Language, Lenguaje de descripción de servicios web), UDDI (Universal Description, Discovery, and Integration, Descripción, detección e

integración universales) y WSIL (Web Services Inspection Language, Lenguaje de inspección de servicios web).

SOAP:

SOAP es un protocolo de mensajes independiente del transporte. Los mensajes SOAP son documentos XML. SOAP utiliza mensajes unidireccionales, aunque es posible combinar mensajes en secuencias de solicitud y respuesta. La especificación SOAP define el formato del mensaje XML, pero no su contenido ni la forma en que se envía. No obstante, SOAP especifica la forma en que los mensajes SOAP se encaminan por medio de HTTP.

WSDL:

Los servicios web sólo resultan útiles si otras aplicaciones pueden reconocer qué hacen y la forma de llamarlos. El equipo de desarrollo debe estar suficientemente informado sobre un servicio web para poder generar un programa cliente que lo llame.

WSDL es un lenguaje basado en XML que se utiliza para definir servicios web y describe la forma de acceder a ellos. Concretamente, describe los contratos de datos y mensajes que ofrece un servicio web. De esta forma, para averiguar que métodos hay disponibles y como se realizan las llamadas con los parámetros adecuados se debe examinar este documento.

UDDI:

UDDI es una norma en proceso de evolución, que trata sobre la descripción, la publicación y la localización de los servicios web que ofrecen las empresas. Se trata de una especificación para el registro distribuido de información sobre servicios web. Cuando se ha desarrollado un servicio web y se ha creado un documento WSDL que lo describe, es necesario que exista una forma de proporcionar la información WSDL a los usuarios del servicio web.

Cuando un servicio web se publica en un registro UDDI, los posibles usuarios pueden buscar e informarse de la existencia de los servicios web. El contenido de un registro UDDI se puede comparar con las guías telefónicas.

WSIL:

WSIL, al igual que UDDI, proporciona un método de descubrimiento de servicios para servicios web. A diferencia de UDDI, WSIL utiliza un modelo descentralizado y distribuido, en lugar de un modelo centralizado.

Los documentos WSIL son, básicamente, punteros de listas de servicios que permiten a los clientes de servicios web examinar los servicios disponibles. La norma WSIL establece la forma de utilizar documentos con formato XML estándar para inspeccionar sitios web en busca de servicios y series de reglas que determinan la forma en que se proporciona la información. Los documentos WSIL contienen varias referencias a documentos de descripción de servicios anteriores. El proveedor del servicio aloja el documento WSIL de forma que los consumidores pueden encontrar los servicios disponibles. (Ver figura 7)

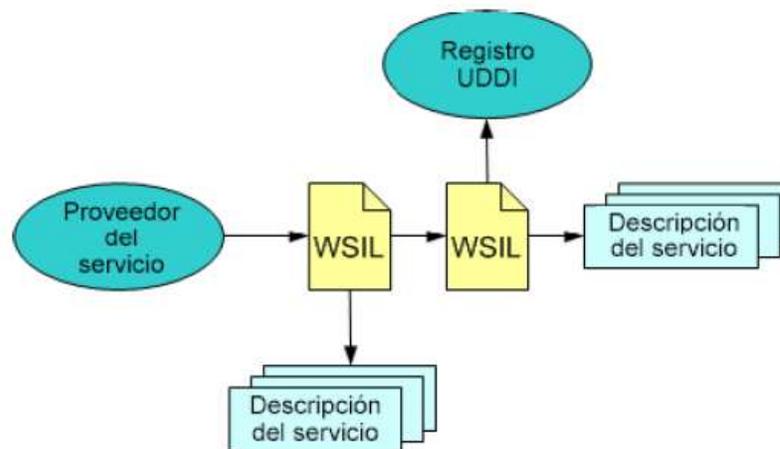


Figura 7. Documento WSIL

RED EPC

El código electrónico de producto – EPC [1] fue concebido como medio para identificar todos los objetos físicos. Es lo suficientemente largo para enumerar todos los objetos y para acomodar todos los métodos, actuales y futuros, de nominación. Este código es proveído por la industria de normalización de códigos, como son la UCC Uniform Code Council (El Concilio de código uniforme) y la EAN European Article Numbering (La enumeración Europea del artículo).

El código EPC es usado principalmente para enlazar los objetos físicos a la red y podría servir como referencia eficiente de información. Adicionalmente el código puede ser extensible permitiendo su expansión en tamaño y diseño.

El código de producto electrónico (EPC) y la red (o sistema) de EPC son propuestas creadas para ayudar a los negocios para mejorar la visibilidad de los productos y ayudar a mantener la seguridad e integridad del producto a través de la cadena de suministro. Para las compañías no es necesario solamente el saber donde están sus productos, sino también necesitar compartir esa información con sus socios de negocios para procesos más seguros y eficientes.

El código por si sólo no significa nada, se pretende la creación de toda una infraestructura inteligente que contiene [2]: (ver figura 8.)

- ✓ Número EPC
- ✓ Tags y lectores
- ✓ Software mediador
- ✓ ONS (Object Name Service)
- ✓ PML (Physical Markup Language)

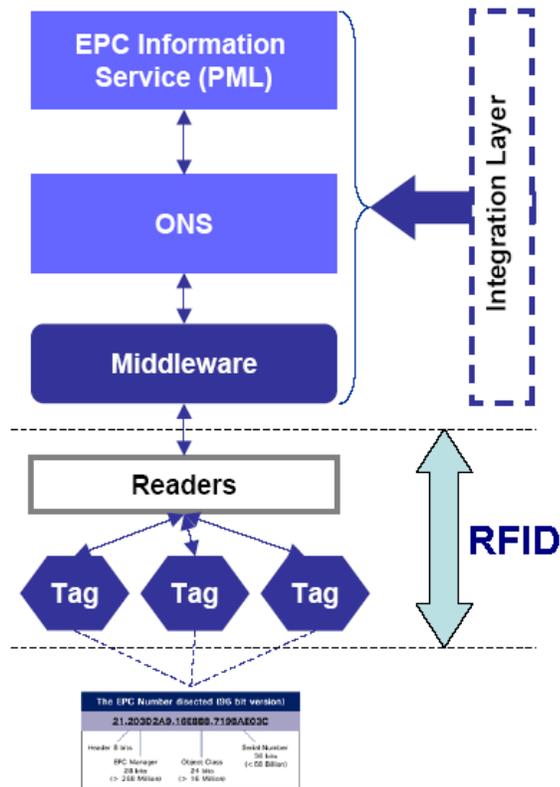


Figura 8. Red EPC

La arquitectura esta específicamente diseñada para su puesta en práctica en grande, en las empresas que necesitan integrar datos en tiempo real que fluyen dentro de procesos existentes del negocio y de usar el back-end de los sistemas de la empresa usando tecnologías automáticas de identificación (Auto - ID) tales como RFID Radio frequency Identification (Identificación por Radiofrecuencia).

TECNOLOGÍA RFID

RFID utiliza una etiqueta que permite reflejar ondas de manera que las ondas retornadas cargan datos. RFID es clasificado como una tecnología de Captura Automática de Datos ADC, al igual que el código de barras, el reconocimiento óptico de caracteres, el reconocimiento de formas y las tarjetas de cinta magnética, etc.

Un sistema de identificación por radiofrecuencia que básicamente consta de tres componentes [3]:

- Sensor inteligente o etiqueta.
- Lector
- Software especializado.

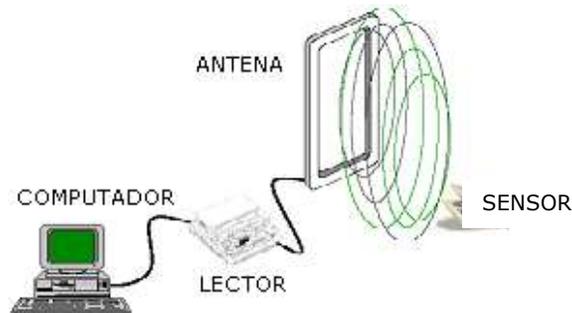


Figura 9. Componentes de RFID [3].

El sensor inteligente es el elemento que se inserta o mimetiza en cualquier producto o artículo que produzca o tenga una empresa. El sensor, más conocido como etiqueta, puede estar disponible en variedad de formas y tamaños, puede almacenar en memoria información variada y que signifique un valor importante para los procesos de una organización o negocio. Este mismo no requiere línea de vista como el código de barras y puede convivir con otras tecnologías.

El lector permite la captura de la información que se encuentra en el sensor inteligente mediante el proceso de lectura, además tiene la capacidad de modificar la información “en el aire” si es necesario.

El software especializado es el que le permite dar un valor agregado a su producto ya que a través de él se ven implementadas las políticas de su organización. Es el que determina como se debe modificar, o en que momento leer y verificar la información contenida en el sensor inteligente.

Esta tecnología está basada en estándares para lograr integración y adición a futuro.

El rango de distancia entre el lector y el sensor puede variar según conveniencia y aplicaciones, permitiendo lograr distancia desde unos pocos centímetros hasta metros mediante el empleo de antenas.

DESCRIPCIÓN GENÉRICA DE COMPONENTES DE LA RED EPC

CÓDIGO EPC

El código EPC es análogo a la dirección IP dada a un nodo en Internet. Es también similar al código del sistema UPC/ EAN (Código de producto universal/ número internacional de artículo), el EPC identificaría cada objeto de manera individual.

Dada la necesidad de tener un único número serial, se propone un esquema de codificación de 96 bits con un encabezado de 8 bits y tres particiones de datos. Ver figura 10.

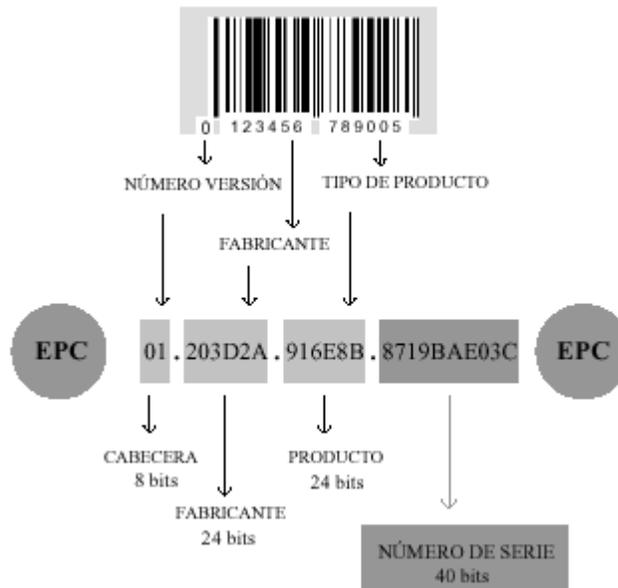


Figura 10. Ejemplo de numeración EPC

Este esquema de 96 bits permite solventar la necesidad de numerar cada objeto en el mundo contra las limitaciones de la mismo esquema, ya que permite la generación de una estructura jerárquica que a su vez permite definir múltiples opciones de código.

TAGS

Son sensores inteligentes que consisten de un microchip conectado a una antena revestidos de un montaje que actúa como una forma de protección. La antena dentro del sensor es el encargado de alimentar de manera eléctrica al microchip, al convertir el campo magnético proveniente del lector [4].



Figura 11. Sensores RFID [4]

Se puede realizar clasificación de estos sensores de diversas maneras:

- ✓ Sensores activos y sensores pasivos
- ✓ Según las frecuencias en las que trabajan
- ✓ Sensores de sólo lectura, Lectura/Escritura o WORM (write-once, Read-many)
- ✓ Sensores sólidos y tipo etiqueta

Es así como existen diversos tipos de sensores y su uso depende del ambiente operativo, el activo a identificar, duración, temperatura, alcance que se desee lograr, entre otros.

LECTORES

El lector se compone de una antena y un controlador de comunicaciones [5]. El controlador tiene tres funciones principales:

- ✓ Codificar y decodificar los datos, revisa los datos, almacena los datos y se comunica con el servidor.
- ✓ Administrador de comunicaciones, que es responsable por activar la etiqueta de RFID, iniciar una sesión, leer y escribir los datos, autorización, revisión integral, etc.
- ✓ Comunicaciones físicas, la frecuencia, la tasa de datos, modulación y potencia.

Los atributos claves para un lector son [6]: el rango de lectura y la capacidad de lectura de tags sin que haya colisión. Estos atributos dependen de la frecuencia de la onda de radio, donde los rangos usados más comúnmente son 125 – 134 KHz, 13.5 MHz, 415 MHz, 868 – 925 MHz, y 2.4 GHz.

Se espera que en el futuro los lectores [6] puedan operar en mas de una banda, entender los protocolos nativos de la Internet, ser parte de un sistema distribuido o cliente-servidor, incorporar un agente conductor para gestionar una población en un nivel alto de ellas y no requerir la intervención humana para solución de problemas.

SOFTWARE MEDIADOR

Software basado en una arquitectura abierta que provee la "inteligencia" a nivel local para controlar e integrar la infraestructura de tags y lectores [2].

El software mediador cumple las siguientes funciones:

- ✓ Recibir y depurar la información enviada por los lectores.
- ✓ Enviar datos a aplicaciones internas o redes externas.
- ✓ Administrar tareas.

ONS - OBJECT NAME SERVICE

El ONS o Servidor de nombres de objetos [2] actúa como un directorio que convierte un EPC en un URL en Internet. Los URL son utilizados para localizar direcciones IP, en las que se encontrará información más detallada sobre el producto, caja o estiba identificada.

PML – PHYSICAL MARKUP LANGUAGE

El lenguaje de marcación físico PML [2] es un método basado en XML para describir objetos de tal forma que pueda ser entendido por personas y sistemas de cómputo.

La meta principal del PML es proveer un lenguaje estándar para representar y distribuir información relacionada con objetos identificados con RFID. Esto incluye administración del inventario, transacciones automatizadas y seguimiento en la cadena de abastecimiento.

MODELO ESENCIAL DE CASOS DE USO DEL SISTEMA DE TRAZABILIDAD

Diagrama de Casos de Uso del Proceso de Trazabilidad en Corseda

El diagrama de casos de uso representa la relación de actores y casos de uso del sistema de trazabilidad de productos en Corseda.

Para dar mayor claridad a la representación del diagrama, se ha dividido en dos partes: la perteneciente al sistema interno de Corseda y otro diagrama que representa la relación con su entorno (usuario, comercializadora y transporte).

Diagrama de casos de uso sistema Interno Trazabilidad Corseda

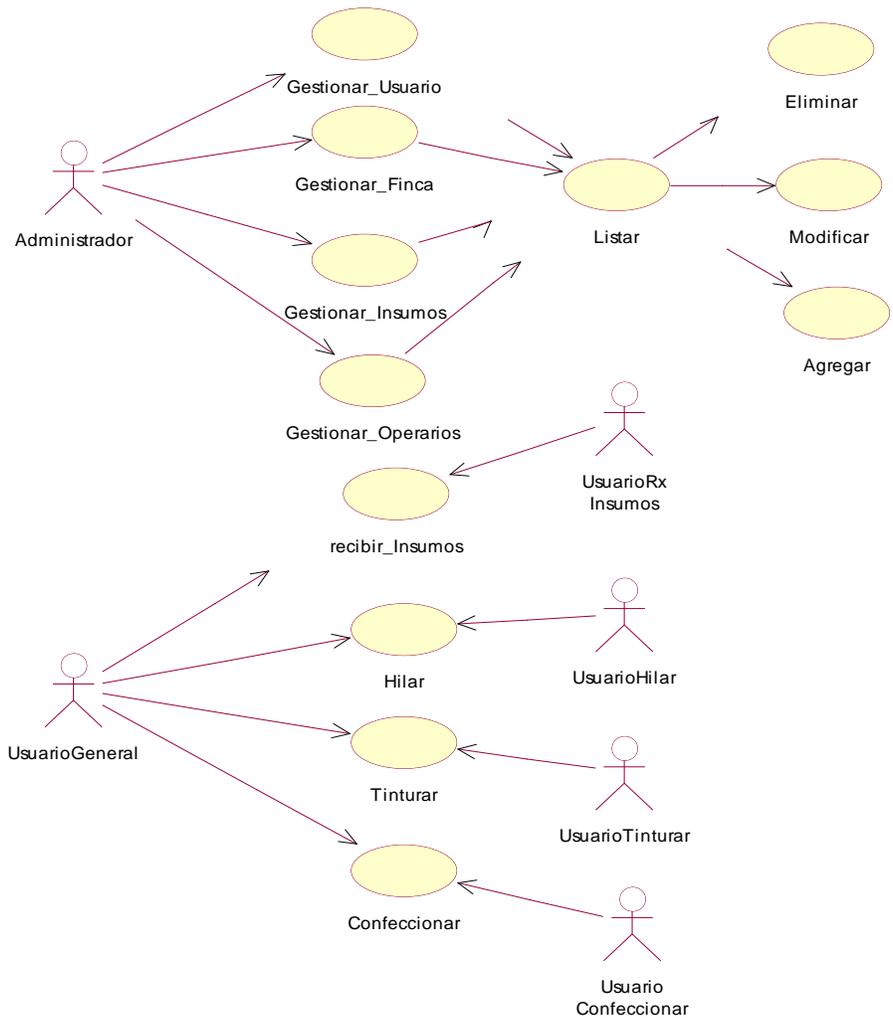
El sistema interno de Corseda cuenta con tres tipos de actores:

- Administrador
- Usuario General
- Usuario Especializado

Administrador: Persona encargada de la gestión de la información de los recursos de corseda como es Gestión de fincas, de insumos, de usuarios y operarios.

Usuario General: Persona encargada de manejar el sistema en los procesos de recepción de insumos, tinturado, hilado y confeccionado. Este tipo de usuario es considerado para aquellas organizaciones pequeñas que no cuenten con el personal suficiente para especializarse en cada proceso, sino que tengan que suplir varias funciones.

Usuario especializado: Existen cuatro tipos de usuarios especializados para cada proceso que se lleva a cabo en Corseda: el usuario de recepción de insumos, usuario de hilado, usuario de tinturado y usuario de confeccionado. Para este caso los usuarios sólo tienen permiso para ingresar exclusivamente a las funciones que realizan.



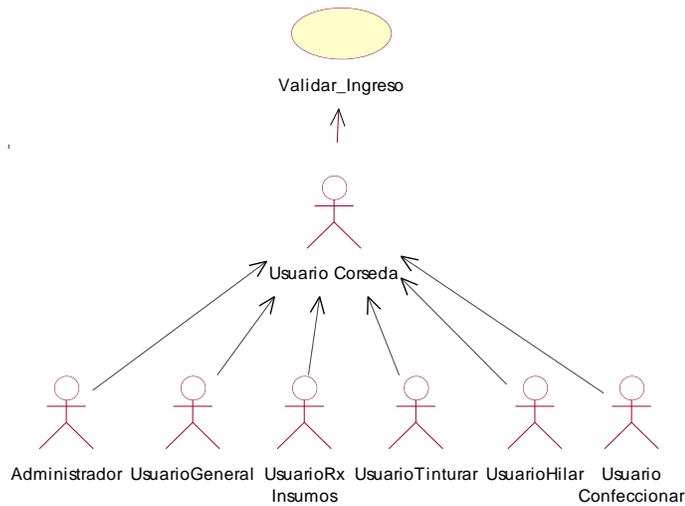


Figura 12. Diagramas de casos de uso sistema Interno trazabilidad Corseda

Descripción de los Escenarios de los Casos de Uso Esenciales

CASO DE USO 1:	Listar	
ACTOR:	Administrador	
PROPÓSITO:	Interfaz que permite acceder a las funciones de gestión como son eliminar, modificar o adicionar.	
PRECONDICIONES:	El administrador se debe haber validado en el sistema. Haber ingresado a un sistema de gestión (de operarios, de insumos, de fincas).	
E S C E N A R I O	ADMINISTRADOR	SISTEMA CORSEDA
	1. ingresa a alguna de las aplicaciones de gestión 3. Escoge cualquiera de las opciones	2. muestra las funciones de gestión: Adicionar, Modificar o Eliminar 4. Realiza la acción escogida en la base de datos.
POSCONDICIONES:	Opción de escoger cualquiera de las funciones de gestión.	
FLUJOS ALTERNATIVOS:	Ninguno	
NOTAS:	Este caso de uso siempre se utiliza para inicializar los casos de uso2: Adicionar, caso de uso 3: Modificar y caso de uso 4: Eliminar.	
EXCEPCIONES:	Ninguna	

CASO DE USO 2:	Adicionar
ACTOR:	Administrador
PROPÓSITO:	Permite adicionar un elemento (Insumo, finca, operario) en las bases de datos correspondientes, dependiendo de la aplicación de gestión que haya seleccionado.
PRECONDICIONES:	El administrador se debe haber validado en el sistema. Haber ingresado a un sistema de gestión (de operarios, de insumos, de fincas).

	ADMINISTRADOR	SISTEMA CORSEDA
E S C E N A R I O	1. ingresa a alguna de las aplicaciones de gestión	2. caso de uso1: listar
	3. Escoge la opción Adicionar	4. Muestra formulario con la información que se va a adicionar.
	5. Ingresa los datos en los campos vacíos de la información que va a ser adicionada	6. Adiciona un elemento (Insumo, finca, operario) y la información relacionada a la base de datos específica de cada aplicación de gestión

POSCONDICIONES:	Datos adicionados en alguna de las bases de datos
FLUJOS ALTERNATIVOS:	Ninguno
NOTAS:	Este caso de uso siempre será usado en la función Adicionar de los casos de uso de gestión de fincas, de insumos y de usuarios.
EXCEPCIONES:	Ninguna

CASO DE USO 3:	Modificar
ACTOR:	Administrador
PROPÓSITO:	Permite modificar un elemento (Insumo, finca, operario) en las bases de datos correspondientes, dependiendo de la aplicación de gestión que haya seleccionado.
PRECONDICIONES:	El administrador se debe haber validado en el sistema. Haber ingresado a un sistema de gestión (de operarios, de insumos, de fincas).

	ADMINISTRADOR	SISTEMA CORSEDA
E S C E N A R I O	1. ingresa a alguna de las aplicaciones de gestión	2 caso de uso1: listar
	3. Escoge la opción Modificar	4. Muestra formulario con la información que se va a modificar.
	5. Cambia los datos en los campos que van a ser modificados por parte del usuario.	6. Modifica un elemento (Insumo, finca, operario) y la información relacionada con él en la base de datos específica de cada aplicación de gestión
POSCONDICIONES:	Datos modificados en alguna de las bases de datos	
FLUJOS ALTERNATIVOS:	Ninguno	
NOTAS:	Este caso de uso siempre será llamado en la función Modificar de los casos de uso de gestión de fincas, de insumos y de usuarios.	
EXCEPCIONES:	Ninguna	

CASO DE USO 4:	Eliminar
ACTOR:	Administrador
PROPÓSITO:	Permite eliminar un elemento (Insumo, finca, operario) en las bases de datos correspondientes, dependiendo de la aplicación de gestión que haya seleccionado.
PRECONDICIONES:	El administrador se debe haber validado en el sistema. Haber ingresado a un sistema de gestión (de operarios, de insumos, de fincas).

	ADMINISTRADOR	SISTEMA CORSEDA
E S C E N A R I O	1. ingresa a alguna de las aplicaciones de gestión	2. caso de uso1: listar
	3. Escoge la opción Eliminar	4. Muestra formulario con el dato que va a ser eliminado
	5. Escoge el elemento que va a ser eliminado.	6. Elimina un elemento (Insumo, finca, operario) y la información relacionada con él en la base de datos específica de cada aplicación de gestión
POSCONDICIONES:	Datos eliminados en alguna de las bases de datos	
FLUJOS ALTERNATIVOS:	Ninguno	
NOTAS:	Este caso de uso siempre será llamado en la función Eliminar de los casos de uso de gestión de fincas, de insumos y de	

	usuarios.
EXCEPCIONES:	Ninguna

CASO DE USO 5:	Gestionar_Operario
ACTOR:	Administrador
PROPÓSITO:	Mostrar la interfaz en que se puede gestionar (Adicionar, eliminar o modificar) un operario.
PRECONDICIONES:	El administrador se debe haber validado en el sistema.

E S C E N A R I O	ADMINISTRADOR	SISTEMA CORSEDA
	1. Oprime el botón Gestionar Usuarios 3. Escoge cualquiera de las opciones de gestión (eliminar, modificar o adicionar)	2. muestra interfaz de gestión de usuarios 4. Caso de uso 1: listar.

POSCONDICIONES:	Operario adicionado, eliminado o modificado en la base de datos de operarios
FLUJOS ALTERNATIVOS:	Ninguno
NOTAS:	Este caso de uso usa el caso de 1: Listar y alguno de los casos de uso 2, 3 o 4 dependiendo de la opción de gestión escogida.
EXCEPCIONES:	Ninguna

CASO DE USO 6:	Gestionar_Finca
ACTOR:	Administrador
PROPÓSITO:	Mostrar la interfaz en que se puede gestionar (Adicionar, eliminar o modificar) una finca.
PRECONDICIONES:	El administrador se debe haber validado en el sistema.

E S C E N A R I O	ADMINISTRADOR	SISTEMA CORSEDA
	1. Oprime el botón Gestionar Fincas 3. Escoge cualquiera de las opciones de gestión (eliminar, modificar o adicionar)	2. muestra interfaz de gestión de Fincas 4. Caso de uso 1: listar.

POSCONDICIONES:	Finca adicionada, eliminada o modificada en la base de datos de fincas
-----------------	--

FLUJOS ALTERNATIVOS:	Ninguno
NOTAS:	Este caso de uso usa el caso de 1: Listar y alguno de los casos de uso 2, 3 o 4 dependiendo de la opción de gestión escogida.
EXCEPCIONES:	Ninguna

CASO DE USO 7:	Gestionar_Insumos
ACTOR:	Administrador
PROPÓSITO:	Mostrar la interfaz en que se puede gestionar (Adicionar, eliminar o modificar) un insumo.
PRECONDICIONES:	El administrador se debe haber validado en el sistema.

E S C E N A R I O	ADMINISTRADOR	SISTEMA CORSEDA
	1. Oprime el botón Gestionar Insumos 3. Escoge cualquiera de las opciones de gestión (eliminar, modificar o adicionar)	2. muestra interfaz de gestión de Insumos 4. Caso de uso 1: listar.

POSCONDICIONES:	Insumo adicionado, eliminado o modificado en la base de datos de insumos.
FLUJOS ALTERNATIVOS:	Ninguno
NOTAS:	Este caso de uso usa el caso de 1: Listar y alguno de los casos de uso 2, 3 o 4 dependiendo de la opción de gestión escogida.
EXCEPCIONES:	Ninguna

CASO DE USO 8:	Validar_Ingreso
ACTOR:	Usuario_Corseda
PROPÓSITO:	Validar el ingreso de un operario u administrador al sistema.
PRECONDICIONES:	Ninguna

E S C E N A R I O	ADMINISTRADOR	SISTEMA CORSEDA
		1. Ingresa el nombre de usuario y la contraseña en los campos de la página de inicio del sistema. 2. Oprime el botón Aceptar
POSCONDICIONES:		Usuario validado para la utilización de las aplicaciones del sistema.
FLUJOS ALTERNATIVOS:		Ninguno
NOTAS:		Este caso de uso usa el caso de 1: Listar y alguno de los casos de uso 2, 3 o 4 dependiendo de la opción de gestión escogida.
EXCEPCIONES:		E1 : No corresponda el nombre de usuario con la contraseña

CASO DE USO 9:		Recibir_Insumos
ACTOR:		Usuario_General (Usuario Rx insumos)
PROPÓSITO:		Registrar la recepción de insumos a Corseda
PRECONDICIONES:		Usuario validado en el sistema
E S C E N A R I O	USUARIO_GENERAL (Usuario Rx insumos)	SISTEMA CORSEDA
	1. Oprime el botón de Recepción de insumos 3. Escribe el código de insumo en el campo vacío correspondiente a programar. 5. Verifica si el código ya existe en sistema. 6. Ingresa la información de los campos vacíos como son referencia de insumo, cantidad y referencia de finca. 7. Oprime Guardar registro	2. Muestra interfaz de recepción de insumos 4. Programa el código en la tag E1 8. Almacena información en base de datos de insumos perteneciente a los datos ingresados por el usuario, la fecha, el operario, la hora.
POSCONDICIONES:		Nuevo insumo registrado en base de datos.

FLUJOS ALTERNATIVOS:	Ninguno
NOTAS:	Ninguna
EXCEPCIONES:	E1: Error en la programación de la tag, no permite registro del nuevo insumo.

CASO DE USO 10:	Tinturar
ACTOR:	Usuario_General (Usuario Tinturar)
PROPÓSITO:	Registrar la información del proceso de Tinturado
PRECONDICIONES:	Usuario validado en el sistema

	USUARIO_GENERAL (Usuario Tinturar)	SISTEMA CORSEDA
E S C E N A R I O	<p>1. Oprime el botón de Proceso de tinturado</p> <p>3. Escribe el código de producto tinturado en el campo vacío correspondiente a programar.</p> <p>5. Ingresa la información de la descripción</p> <p>7. Aproxima el insumo utilizado para este proceso al lector y oprime el botón Insumo 1. N1</p> <p>10. Ingresa la cantidad usada del insumo.</p> <p>11. Oprime el botón Guardar registro</p>	<p>2. Muestra interfaz de proceso de tinturado</p> <p>4. Programa el código en la tag E1</p> <p>8. Lee la tag del insumo utilizado.</p> <p>9. Proporciona información de referencia del insumo.</p> <p>12. Almacena información en base de datos de insumos perteneciente a los datos ingresados por el usuario, la fecha, el operario, la hora.</p>

POSCONDICIONES:	Información de proceso de tinturado registrado en base de datos.
FLUJOS ALTERNATIVOS:	Ninguno
NOTAS:	N1: Se hace lo mismo para los otros insumos utilizando los otros botones de insumos. Puede haber un máximo de 4 insumos.
EXCEPCIONES:	E1: Error en la programación de la tag, no permite registro del nuevo insumo.

CASO DE USO 11:	Hiladura
------------------------	-----------------

ACTOR:	Usuario_General (Usuario Hilar)
PROPÓSITO:	Registrar la información del proceso de hiladura
PRECONDICIONES:	Usuario validado en el sistema

E S C E N A R I O	USUARIO_GENERAL (Usuario Hilar)	SISTEMA CORSEDA
		1. Oprime el botón de Proceso de hiladura 3. Escribe el código de producto de hilado en el campo vacío correspondiente a programar. 5. Ingresa la información de la descripción 7. Aproxima el insumo utilizado para este proceso al lector y oprime el botón Insumo 1. N1 10. Ingresa la cantidad usada del insumo. 11. Oprime el botón Guardar registro

POSCONDICIONES:	Información de proceso de tinturado registrado en base de datos.
FLUJOS ALTERNATIVOS:	Ninguno
NOTAS:	N1: Se hace lo mismo para los otros insumos utilizando los otros botones de insumos. Puede haber un máximo de 4 insumos.
EXCEPCIONES:	E1: Error en la programación de la tag, no permite registro del nuevo insumo.

CASO DE USO 12:	Confecionar	
ACTOR:	Usuario_General (Usuario Confecionar)	
PROPÓSITO:	Registrar la información del proceso de confección	
PRECONDICIONES:	Usuario validado en el sistema	
E S	USUARIO_GENERAL (Usuario Hilar)	SISTEMA CORSEDA

C E N A R I O	<p>1. Oprime el botón de Proceso de confección</p> <p>3. Escribe el código de producto de confección perteneciente al código EPC en el campo vacío correspondiente a programar.</p> <p>5. Ingresa la información de la descripción</p> <p>7. Aproxima el insumo utilizado para este proceso al lector y oprime el botón Insumo 1. N1</p> <p>10. Ingresa la cantidad usada del insumo.</p> <p>11. Oprime el botón Guardar registro</p>	<p>2. Muestra interfaz de proceso de confección</p> <p>4. Programa el código en la tag E1</p> <p>8. Lee la tag del insumo utilizado.</p> <p>9. Proporciona información de referencia del insumo.</p> <p>12. Almacena información en base de datos de insumos perteneciente a los datos ingresados por el usuario, la fecha, el operario, la hora.</p>
---------------------------------	--	--

POSCONDICIONES:	Información de proceso de confección registrado en base de datos.
FLUJOS ALTERNATIVOS:	Ninguno
NOTAS:	N1 : Se hace lo mismo para los otros insumos utilizando los otros botones de insumos. Puede haber un máximo de 4 insumos.
EXCEPCIONES:	E1 : Error en la programación de la tag, no permite registro del nuevo insumo.

Diagrama de casos de uso sistema Externo Trazabilidad Corseda

El sistema de corseda interactúa con distintos actores externos, en este caso:

- Cliente
- Transporte
- Comercializadora

Para este proyecto se han escogido dos entes externos además del cliente final, pero en general, el sistema debe ser capaz de interactuar con los distintos eslabones de la cadena de abastecimiento de una manera sencilla.

Adicionalmente cada ente externo tiene funcionalidades semejantes a las realizadas al interior de Corseda, por lo que se prioriza la descripción de casos de uso internos de Corseda.

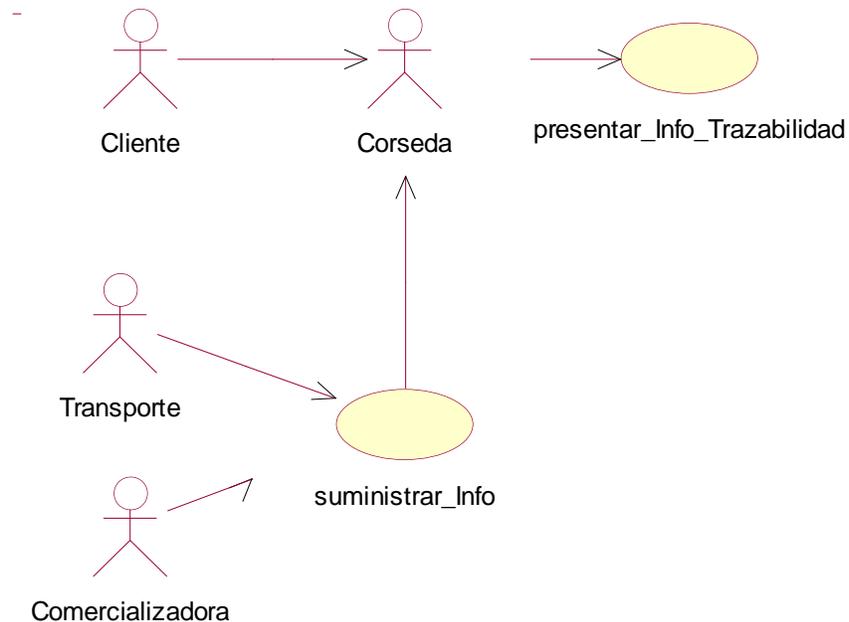


Figura 13. Diagrama de casos de uso sistema Externo Trazabilidad Corseda

Cliente: Persona que adquiere el producto final y tiene la posibilidad de ver la información de trazabilidad en el sistema de Corseda.

Comercializadora: Organización encargada de comercializar los productos de Corseda. Entrega la información de trazabilidad de los productos relacionadas con los procesos de recepción de productos, pedidos y entrega a transporte.

Transporte: Se encarga de transportar los productos a los sitios donde se han hecho pedidos y dar la información de trazabilidad de los productos en los procesos de recepción y entrega.

CASO DE USO 13:	Suministrar_Info	
ACTOR:	Sistema Corseda	
PROPÓSITO:	Suministrar a Corseda la información de trazabilidad de transporte o comercializadora	
PRECONDICIONES:	Existencia de información de trazabilidad en servidores de comercializadora y transporte Comunicación WEB disponible Consulta de información de trazabilidad por parte de un cliente que adquirió un producto.	
E S C E N A R I O	SISTEMA CORSEDA	TRANSPORTE COMERCIALIZADORA
	1 Realiza consulta de código EPC. 4. Despliega información de trazabilidad para el cliente que realiza la consulta. 5. Genera archivo PML	2. Suministra un WEB Service para comunicar la información de trazabilidad. 3. Envía información de trazabilidad del producto consultado. F1
POSCONDICIONES:	Información de trazabilidad suministrada al cliente	
FLUJOS ALTERNATIVOS:	F1: Si los servidores de comercializadora y transporte se encuentran fuera de servicio, Corseda usa el archivo PML para brindar la información de trazabilidad del producto al cliente.	
NOTAS:	Ninguna	
EXCEPCIONES:	Ninguna	

CASO DE USO 14:	Presentar_Info_Trazabilidad
------------------------	------------------------------------

ACTOR:	Cliente	
PROPÓSITO:	Suministrar la información de trazabilidad del producto adquirido por un cliente.	
PRECONDICIONES:	Consulta de información de trazabilidad por parte de un cliente que adquirió un producto. Información de trazabilidad del producto de toda la cadena de abastecimiento en archivo PML o servidores de comercializadora y transporte en uso.	
E S C E N A R I O	Cliente	SISTEMA CORSEDA
	1 Realiza consulta de código EPC.	2. Valida en bases de datos de código EPC el código registrado por el cliente. E1 3. Suministra información de trazabilidad
POSCONDICIONES:	Información de trazabilidad suministrada al cliente	
FLUJOS ALTERNATIVOS:	Ninguno	
NOTAS:	Ninguna	
EXCEPCIONES:	E1: si el código en la base de datos no se puede entregar la información de trazabilidad.	

DESCRIPCIÓN DE PAQUETES Y CLASES DE ANÁLISIS ESENCIALES DEL SISTEMA

Para mayor entendimiento, el sistema de trazabilidad se ha dividido en tres paquetes principalmente: Corseda, Comercializadora y Transporte (ver figura 14). Cada uno de ellos sigue el patrón MVC (Modelo-Vista-Control). Este patrón pretende conseguir un desarrollo completamente estructurado de las aplicaciones ya que divide perfectamente la lógica del negocio (modelo), la presentación (vista) y el control de flujo de aplicaciones (Control).

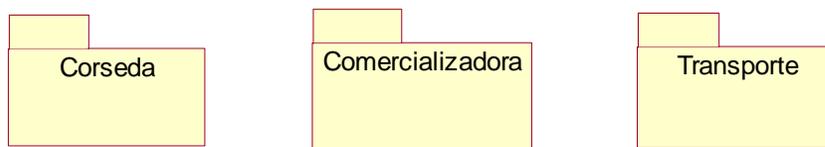


Figura 14. Paquetes de análisis del sistema de trazabilidad de Corseda

La forma en como se ha desarrollado el sistema interno de Corseda es el mismo para la comercializadora y transporte. Por este motivo se explicará en detalle los componentes del

sistema interno de Corseda y se asume que es similar para los otros dos sistemas pero con sus propios procesos que intervienen en la trazabilidad de los productos (almacenamiento, embalaje, recepción, entrega, etc.).

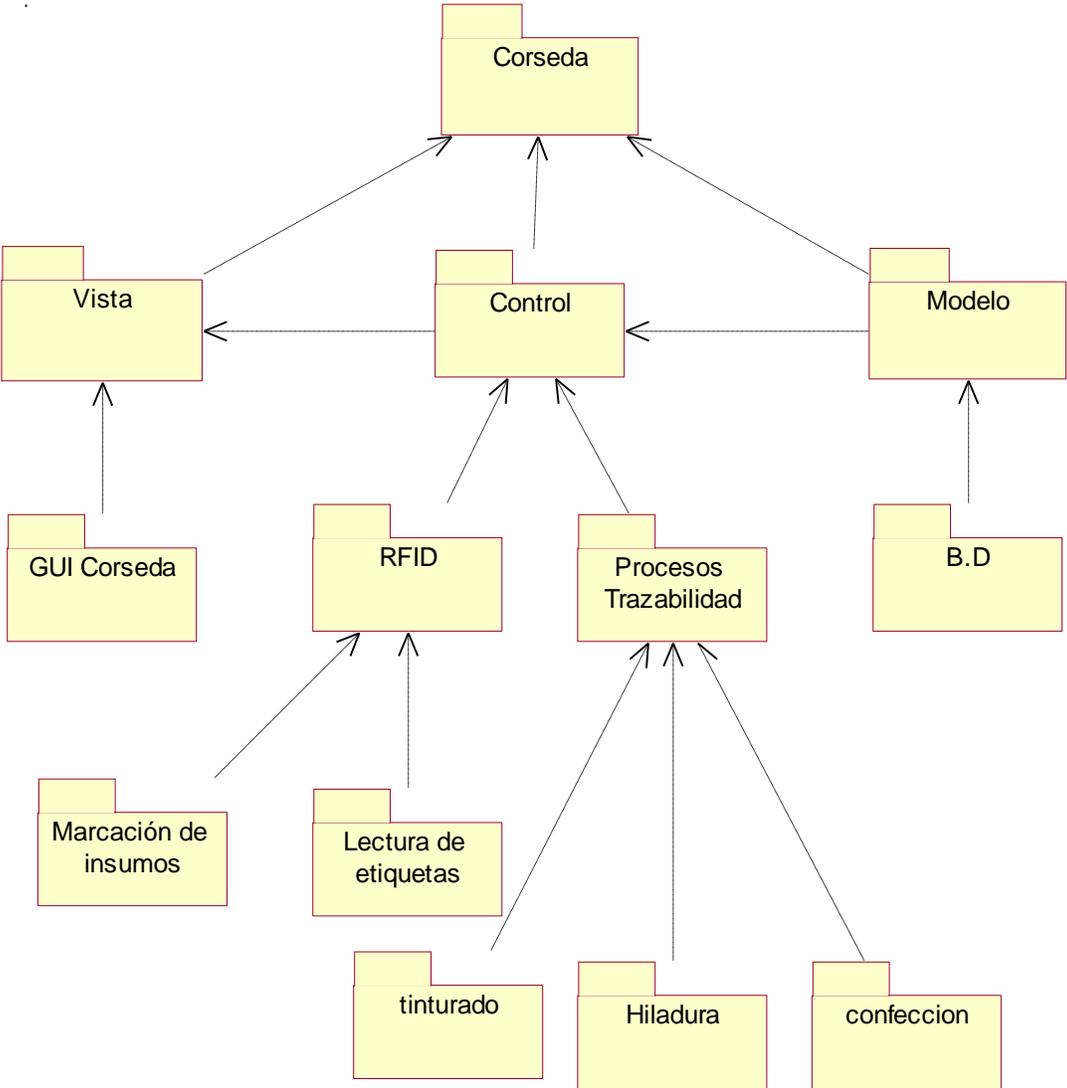


Figura 15. Diagrama de paquetes del sistema de trazabilidad interno de Corseda

CLASES PERTENECIENTES A LOS PAQUETES

CONTROL

AdministradorAction.java

IndexAction.java

OperadorAction.java

- Consulta
AdministrarAction.java
ConsultaAction.java
- Fincas

ActualizarAction.java
GestionarAction.java
- Insumos
ActualizarAction.java
GestionarAction.java
- Operarios
ActualizarAction.java

GestionarAction.java
- Usuarios
ActualizarAction.java
GestionarAction.java
UsuarioAction.java

MODELO

BeanConector.java

ConsultaForm.java

FormularioForm.java

- Consulta
BeanConsulta.java
BeanGestionConsulta.java
PMLConsulta.java
ValidarBeanConsulta.java
- Fincas
BeanFinca.java

BeanGestionFinca.java
PMLFinca.java
ValidarBeanFinca.java

- Insumos
BeanGestionInsumos.java
BeanInsumos.java
PMLInsumo.java
ValidarBeanInsumos.java
- Operarios
BeanGestionOperario.java
BeanOperario.java
PMLOperario.java
ValidarBeanOperario.java
- Usuarios
BeanGestionUsuario.java
BeanUsuario.java
ValidarBeanUsuario.java
- PML
generarPML.java

- localhost.Servicio.services.Validar

BIBLIOGRAFÍA

- [1] D. Brock, "The Electronic Product Code (EPC). A Naming Scheme for Physical Objects", 2001.
- [2] G. Biffi, "EPC - Electronic Product Code. Identifique cualquier objeto esté donde esté", IAC Colombia, 2005.
- [3] ID Plus. Información comercial de ID Plus, Popayán, 2004.
- [4] J. Vello, "RFID, una tecnología madura en un sector dispar", e-business Center PwC&IESE, 2004.
- [5] EAN INTERNATIONAL., "Sistema EAN·UCC - RFID - Identificación Por Radiofrecuencia", 1999.
- [6] M. Reynolds, "Agile Reader Reference Plataforma", Auto ID Center. 2004