

MODELO TECNOLÓGICO PARA LA IMPLEMENTACIÓN DE UN SISTEMA MICRO
EMPRESARIAL PARA LA GESTIÓN DE FACTURACIÓN, INVENTARIO, COMPRAS Y
CONTABILIDAD

PABLO ANDRES HERNANDEZ MORALES

COD.1088299261

UNIVERSIDAD TECNOLOGICA DE PEREIRA
FACULTAD DE INGENIERIAS
PROGRAMA DE INGENIERIA DE SISTEMAS Y COMPUTACIÓN

2017

MODELO TECNOLÓGICO PARA LA IMPLEMENTACIÓN DE UN SISTEMA MICRO
EMPRESARIAL PARA LA GESTIÓN DE FACTURACIÓN, INVENTARIO, COMPRAS Y
CONTABILIDAD

PABLO ANDRES HERNANDEZ MORALES

INFORME PRESENTADO COMO REQUISITO PARA OPTAR AL TITULO DE
INGENIERO DE SISTEMAS Y COMPUTACIÓN

DIRECTOR

ING. CARLOS AUGUSTO MENESES ESCOBAR

UNIVERSIDAD TECNOLOGICA DE PEREIRA
FACULTAD DE INGENIERIAS
PROGRAMA DE INGENIERIA DE SISTEMAS Y COMPUTACION

2017

Tabla de contenido

Resumen	4
Abstract	5
Introducción	6
1.Generalidades	7
1.1. Problema.....	7
1.2. Justificación.....	8
1.3. Objetivos	9
1.4. Método	10
2.Marco teórico	11
2.1.UML	11
2.2.OSR	15
3.Desarrollo	19
3.1. Definiciones	19
3.2. Casos de uso	20
3.3. Diagramas de secuencia	25
3.3.1Módulo de facturación	25
3.3.2. Módulo de contabilidad.....	27
3.3.3Módulo de inventario	30
3.3.4Módulo de compras.....	35
4.Análisis de resultados	39
5.Conclusiones	40
6.Bibliografía.....	41

Resumen

En la actualidad se han desarrollado gran cantidad de micro empresas, cada día más y más personas han optado por emprender en busca de su triunfo personal, triunfo que a veces se ve truncado por la falta de conocimiento en el manejo de cuentas o se les hace más difícil el camino hacia una economía estable, por ello se ha decidido desarrollar una herramienta que facilite el manejo de las cuentas y puedan ser controladas de una manera más eficiente.

En esta práctica empresarial se elaboró el modelo del que será el software que podrán usar estos nuevos empresarios, desarrollando los módulos básicos necesarios para la administración y el manejo de una empresa como lo son facturación, compras, inventario y contabilidad, demostrando la importancia de la documentación y el diagramado previo al desarrollo de un aplicativo ya que sienta las bases de las buenas prácticas a la hora de iniciar un desarrollo, ya que entre más complejo llegue a ser un sistema más beneficios trae el uso de UML ("Unified Modeling Language"), una de las ventajas es que mediante la visión global que nos dan los diagramas del sistema resulta más fácil detectar las dependencias y dificultades implícitas del sistema, otra de las ventajas es que los cambios en la etapa de análisis (Etapa inicial)son más fáciles de realizar que en la etapa final de un sistema como lo fuese la fase de codificación.

De este modo con el trabajo que se realizó en esta práctica se muestra el interés de la empresa OBJETOS Y SISTEMAS RELACIONALES OSR S.A.S., de brindar el apoyo y las herramientas necesarias a los pequeños empresarios de la ciudad que recién estén empezando y a los antiguos para que puedan adaptarse más a este nuevo mundo tecnológico.

Abstract

Nowadays, a great number of micro-enterprises have been developed, more and more people have opted to start looking for their personal triumph, a triumph that is sometimes truncated by lack of knowledge in the management of accounts or made the more difficult the road to a stable economy, so it has been decided to develop a tool that facilitates the management of accounts and can be controlled in a more efficient way.

In this business practice, the model was developed which will be the software that these new entrepreneurs can use, developing the basic modules necessary for the administration and management of a company such as invoicing, purchasing, inventory and accounting, demonstrating the importance of documentation and the previous diagram to the development of an application since it lays the bases of the good practices when initiating a development, since the more complex the system becomes, the more benefits it brings the use of UML ("Unified Modeling Language"), one of the advantages is that through the global vision that the system diagrams give us, it is easier to detect the implied dependencies and difficulties of the system, another advantage is that the changes in the analysis stage (Initial stage) are more Easy to perform than in the final stage of a system such as the coding phase.

In this way, the work carried out in this practice shows the interest of the OBJETOS Y SISTEMAS RELACIONALES OSR SAS, to provide the support and the necessary tools to the small entrepreneurs of the city who are just beginning and the old ones so that can adapt more to this new technological world.

Introducción

Con el pasar de los años gran cantidad de personas ha optado por la creación de su propia empresa, ya sean las nuevas o antiguas generaciones, a veces yendo sin las herramientas necesarias para el desarrollo y la sostenibilidad de su empresa, lo cual hace que su porcentaje de posibilidad de estabilidad financiera disminuya, lo que se busca hacer en esta práctica es desarrollar el modelado y la documentación de lo que será un software con las herramientas básicas del área financiera.

Lo que se busca hacer con dicha herramienta es, disminuir el porcentaje de fracasos empresariales, aumentar el uso de herramientas tecnológicas en la sociedad, aumentar el conocimiento de las personas en las buenas prácticas del manejo económico de una empresa y evitar el uso innecesario de papel.

En este documento se encontrarán evidencias del trabajo realizado y de los conocimientos adquiridos dentro de la práctica empresarial realizada en OSR.SAS, en la cual se me contrato para apoyar los procesos de requerimientos, análisis, codificación y documentación de funcionalidades genéricas en la construcción de un modelo tecnológico para la implementación de un sistema micro empresarial.

1.Generalidades

1.1. Problema

Al llegar a la empresa OSR (Objetos y Sistemas Relacionales) se me asigno un proyecto en el cual debía elaborar la estructura de lo que sería un software contable para pequeñas y medianas empresas con lo cual busca entrar en todo tipo de mercado.

En este caso el problema a atacar es el gran porcentaje de fracasos empresariales debido al mal manejo de sus activos o de su inventario cuando se trata de una expendedora de productos ya que gran cantidad de empresas empiezan solo con sus ideas y suelen dejar el control de sus compras, gastos y ventas en segundo plano y como no suelen contratar un contador al comienzo del negocio o cuando se posee un negocio pequeño, no suelen tener una visión global de lo que ha sucedido en su negocio a lo largo que pasa el tiempo lo que suele terminar resumido en cuentas alcanzadas o ventas insuficientes añadiendo los vacíos en gastos que se pueden presentar.

Sin una herramienta que le sirva de apoyo a las empresas en el crecimiento este porcentaje de fracasos no disminuirá fácilmente o se mantendrán los métodos de papel y lápiz los cuales, aunque funcionen bien tomaran un gasto de tiempo y no da una visión amplia de todos los movimientos realizados, además de que será más difícil incluir un plan de crecimiento dentro de las empresas que busquen este camino.

1.2. Justificación

La importancia de plantear esta herramienta tecnológica para las empresas es que minimiza la utilización de distintos tipos de herramientas a la vez, se da una herramienta accesible y de fácil comprensión lo cual permite un monitoreo real de cómo se está enfocando el dinero de la empresa permitiendo así el crear planes de apoyo si se presenta algún problema económico disminuyendo así el porcentaje de fracasos que se pueden presentar en las empresas que recién nacen dentro del país asignando que herramientas que vayan disminuyendo el desconocimiento de los nuevos emprendedores y fortaleciendo las capacidades de los que ya llevan un buen tiempo establecidos, adquiriendo mayor conocimiento del área contable lo que les daría un respiro para concentrarse en las ideas de innovación.

1.3. Objetivos

Objetivo general

- Elaborar el documento de ingeniería de software con los diagramas UML de los modelos de facturación, inventario, compras y contabilidad.

Objetivos específicos

- Desarrollar e implementar el modelo de facturación.
- Desarrollar e implementar el modelo de inventario.
- Desarrollar e implementar el modelo de compras.
- Desarrollar e implementar el modelo de contabilidad.

1.4. Método

La metodología para la realización de esta práctica la cual consiste en el desarrollo de la fase de análisis de un sistema para la administración financiera de microempresas basado en los requerimientos entregados por la empresa, se estableció los puntos a realizar dentro del tiempo que dura la práctica

<i>Febrero-Marzo 2016</i>	Establecer cada uno de los perfiles que se usarán y sus características de acuerdo uso.
<i>Abril-Mayo 2016</i>	Realizar los diagramas de: Caso de uso Requerimientos Secuencia Clases
<i>Junio-Julio 2016</i>	Realizar los diagramas de: Componentes Despliegue Entidad relación

A partir de los requerimientos recibidos se establecieron los perfiles de usuario y se inició el desarrollo de los documentos base para la elaboración de los diagramas, los cuales fueron posteriormente realizados en la herramienta “Enterprise Architect” una de las aplicaciones más comúnmente usada para la elaboración de diagramas UML.

En la empresa se me fue otorgado un computador de mesa en el cual se desarrolló todo el trabajo en los 6 meses que duró la práctica, al momento de elaborar los diagramas se realizó un estudio previo de las buenas prácticas para desarrollar el modelado UML, el cual sería la base de la práctica a realizar, también se contó con el apoyo de los compañeros que ya llevaban un tiempo allí, para así verificar que el trabajo que se estaba realizando era el adecuado, ya con estas herramientas se inició lo que fue la documentación de un software administrativo enfocándome en realizar un diagramado completo y de fácil comprensión basado en diagramas de caso de uso, de secuencia y de clase como principales para que a la hora del desarrollo se le facilite al

programador el entendimiento de lo que se necesita realizar, aparte de que también se le puedan realizar modificaciones de más fácil manera.

2.Marco teórico

2.1.UML

2.1.1. Historia de UML (Lenguaje Unificado de Modelado)

El UML nació en 1996 cuando Rational Software Corporation concluyó que para cómo se estaban desarrollando los lenguajes de modelado debían encargar a los tres amigos (quienes eran llamados así por sus discusiones por las prácticas metodológicas), James Rumbaugh, Grady Booch e Ivar Jacobson la creación de un lenguaje unificado.

Se organizó un consorcio internacional llamado UML Partners en 1996 bajo la dirección técnica de Rumbaugh, Jacobson y Booch para completar las especificaciones del UML. El borrador de la especificación UML 1.0 de UML Partners fue propuesto a la OMG en enero de 1997. Durante el mismo mes, la UML Partners formó una Fuerza de Tarea Semántica, encabezada por Cris Kobryn y administrada por Ed Eykholt, para finalizar las semánticas de la especificación y para integrarla con otros esfuerzos de estandarización. El resultado de este trabajo, el UML 1.1, fue presentado ante la OMG en agosto de 1997 y adoptado por la OMG en noviembre de 1997, a partir de esa fecha se han realizado muchas mejoras en las cuales se nota lo mucho que ha madurado y se han corregido los errores y defectos que se dieron en la primera versión de UML. Después de esto se adoptó por la OMG la versión UML 2.0 en el 2005, la cual ya va en su versión UML 2.5 a la fecha del día de hoy.

2.1.2. Importancia de UML

Hoy en día el uso de UML es usado como el lenguaje estándar a la hora del análisis y diseño de un sistema ya que gracias a este podemos plasmar los requerimientos y la estructura de un software de manera que sea entendible por todos y así disminuir los errores a la hora de escribir el código.

El uso de diagramas UML permite una mejor comunicación a la hora de desarrollar en equipos multidisciplinarios, hablamos de analistas, diseñadores, programadores, etc. Siendo estas solo las características básicas que hacen importante el UML.

Otra de las grandes características del uso de UML es que, para el desarrollo o mantenimiento de software de gran tamaño, ya que allí se pueden detectar las dependencias y dificultades que puede presentar el sistema, lo cual resultaría más sencillo aquí que una vez terminado el software.

Así que esta práctica se basó en el desarrollo de UML en la construcción de un modelo tecnológico para la implementación de un sistema micro empresarial, permitiéndome así ver la importancia de esto previo a desarrollarlo como código.

2.1.3. Tipos de diagramas en UML en la actualidad.

a. Estructurales

Muestran la estructura estática de los objetos en un sistema.

- **Diagrama de clases:** Los diagramas de clase son, sin duda, el tipo de diagrama UML más utilizado. Es el bloque de construcción principal de cualquier solución orientada a objetos. Muestra las clases en un sistema, atributos y operaciones de cada clase y la relación entre cada clase. En la mayoría de las herramientas de modelado, una clase tiene tres partes, nombre en la parte superior, atributos en el centro y operaciones o métodos en la parte inferior. En sistemas grandes con muchas clases relacionadas, las clases se agrupan para crear diagramas de clases. Las Diferentes relaciones entre las clases se muestran por diferentes tipos de flechas.
- **Diagrama de componentes:** Un diagrama de componentes muestra la relación estructural de los componentes de un sistema de software. Estos se utilizan principalmente cuando se trabaja con sistemas complejos que tienen muchos componentes. Los componentes se comunican entre sí mediante interfaces. Las interfaces se enlazan mediante conectores.

- **Diagrama de despliegue:** Un diagrama de despliegue muestra el hardware de su sistema y el software de ese hardware. Los diagramas de implementación son útiles cuando la solución de software se despliega en varios equipos, cada uno con una configuración única.
- **Diagrama de objetos:** Los diagramas de objetos, a veces denominados diagramas de instancia, son muy similares a los diagramas de clases. Al igual que los diagramas de clases, también muestran la relación entre los objetos, pero usan ejemplos del mundo real. Se utilizan para mostrar cómo se verá un sistema en un momento dado. Debido a que hay datos disponibles en los objetos, a menudo se utilizan para explicar relaciones complejas entre objetos.
- **Diagrama de paquetes:** Como su nombre indica, un diagrama de paquetes muestra las dependencias entre diferentes paquetes de un sistema.
- **Diagrama de perfiles:** El diagrama de perfil es un nuevo tipo de diagrama introducido en UML 2. Este es un tipo de diagrama que se utiliza muy raramente en cualquier especificación.
- **Diagrama de estructura compuesta:** Los diagramas de estructura compuesta se utilizan para mostrar la estructura interna de una clase.

b. De comportamiento

Muestran el comportamiento dinámico de los objetos en el sistema.

- **Diagrama de actividades:** Los diagramas de actividad representan los flujos de trabajo de forma gráfica. Pueden utilizarse para describir el flujo de trabajo empresarial o el flujo de trabajo operativo de cualquier componente de un sistema. A veces, los diagramas de actividad se utilizan como una alternativa a los diagramas de máquina del estado.
- **Diagrama de casos de uso:** Como el tipo de diagrama de diagramas UML más conocido, los diagramas de casos de uso ofrecen una visión general de los actores involucrados en un sistema, las diferentes funciones que necesitan esos actores y cómo interactúan estas diferentes funciones. Es un gran punto de partida para cualquier discusión del proyecto, ya que se pueden identificar fácilmente los principales actores involucrados y los principales procesos del sistema.

- **Diagrama de máquina de estados:** Los diagramas de máquina de estado son similares a los diagramas de actividad, aunque las anotaciones y el uso cambian un poco. En algún momento se conocen como diagramas de estados o diagramas de diagramas de estado también. Estos son muy útiles para describir el comportamiento de los objetos que actúan de manera diferente de acuerdo con el estado en que se encuentran en el momento.

c. De interacción

- **Diagrama global de interacciones:** Los diagramas generales o globales de interacción son muy similares a los diagramas de actividad. Mientras que los diagramas de actividad muestran una secuencia de procesos, los diagramas de interacción muestran una secuencia de diagramas de interacción. En términos simples, pueden llamarse una colección de diagramas de interacción y el orden en que suceden. Como se mencionó anteriormente, hay siete tipos de diagramas de interacción, por lo que cualquiera de ellos puede ser un nodo en un diagrama de vista general de interacción.
- **Diagrama de comunicación:** El diagrama de comunicación se llamó diagrama de colaboración en UML 1. Es similar a los diagramas de secuencia, pero el foco está en los mensajes pasados entre objetos.
- **Diagrama de secuencia:** Los diagramas de secuencia en UML muestran cómo los objetos interactúan entre sí y el orden en que se producen esas interacciones. Es importante tener en cuenta que muestran las interacciones para un escenario en particular. Los procesos se representan verticalmente y las interacciones se muestran como flechas.
- **Diagrama de tiempos:** Los diagramas de sincronización son muy similares a los diagramas de secuencia. Representan el comportamiento de los objetos en un marco de tiempo dado. Si es solo un objeto, el diagrama es directo, pero si hay más de un objeto involucrado, también se pueden usar para mostrar interacciones de objetos durante ese período de tiempo.

2.2.OSR

2.2.1. Introducción

Es una es una organización de derecho privado, vinculada al sector de las tecnologías de la información y la comunicación (TIC), orientada a satisfacer necesidades de los sectores público y privado a nivel nacional e internacional.

Ubicada en la ciudad de Pereira en el departamento de Risaralda.

Cuenta con un portafolio de productos y servicios en desarrollo de software y soporte de primer nivel:

- PSP2A.
- Back office.
- SIOT Comercial.
- SIOT Operativo.
- Recuperación y Cartera.
- ERP de servicios públicos, privados y de producción.
- Impuestos.
- iAleph.
- WMS.
- Recurso Humano.
- Portal turístico.

2.2.2. Reseña histórica

Desde 1999, año de su fundación, la organización se ha preocupado, además del desarrollo de software especializado para empresas públicas o privadas del tipo industrial, comercial y de Servicios Públicos Domiciliarios, del orden nacional e internacional, por situar el problema o la necesidad de cada uno de los clientes como una situación propia, convirtiéndose en un aliado y no en simple proveedor; esto ha permitido establecer una política de precios competitivos y un amplio conocimiento del mercado tecnológico.

A pesar de que la empresa no es de reciente creación y cuenta en la actualidad con unidades de negocio variados, podría considerarse una empresa inicial, en mérito a que algunos de sus productos aún se encuentran en la etapa de investigación y desarrollo y otros aún no han sido implementados para sus respectivos ajustes.

Las alianzas estratégicas y las uniones temporales, han permitido la participación de proyectos de significativo monto, y han llevado al crecimiento mediático de la organización, evitando socios inversionistas que aliviaran la iliquidez de la empresa, pero que restarían seguramente el manejo y control de la misma.

Una de las ventajas competitivas, y podría decirse, la de mayor relevancia, que ofrece OSR, es la oportunidad de establecer alianza con la Universidad Tecnológica de Pereira, para garantizar altos niveles de contratación inmediatos.

2.2.3. Misión

Agregamos valor a nuestros clientes a través de productos y servicios TI de alta competitividad, integrando tecnología de punta, ambientes web y georreferenciación, soportados en principios de desarrollo sustentable y responsabilidad social empresarial.

2.2.4. Visión

Alcanzaremos reconocimiento y liderazgo nacional en proyectos SAAS en servicios públicos domiciliarios, a través de un portafolio de soluciones en Cloud Computing, facilitando la exportación de productos o servicios al mercado hispanoamericano, soportados en criterios de calidad del servicio.

2.2.5. Políticas

OSR se compromete en la protección de la inversión de nuestros clientes, sustentándola en:

- Cumplimiento de los requisitos demandados.
- Actualización permanente de nuestros productos manteniéndolos a la vanguardia tecnológica.
- Procesos eficaces, eficientes, soportados en la mejora continua.
- Excelente servicio postventa.
- Profesionales que cuentan con alta capacitación tecnológica.
- Solidez de nuestra organización.

2.2.6. Objetivos

- Acreditar nuestros procesos, productos y servicios, mediante certificaciones de calidad con reconocimiento internacional, haciéndolos eficaces y eficientes mejorando en forma continua.
- Posicionar un portafolio de productos SAAS y de computación en la nube (Cloud Computing). Mínimo un producto en el área de servicios públicos domiciliarios. Alinear y adecuar la infraestructura, las operaciones y la gestión para integrar este nuevo modelo comercial.
- Implantar el área de innovación, investigación y desarrollo de nuevos productos y tecnologías.
- Garantizar unas relaciones benéficas de Costo / Beneficio y Tiempo proyectado / tiempo ejecutado, en proyectos de clientes.
- Mantener altos niveles de capacitación tecnológica para todos los niveles organizacionales.
- Acompañar a nuestros clientes a lo largo de la vida útil del producto.

2.2.7. Principios

La organización fomentará los siguientes principios como base del accionar empresarial, práctica permanente de sus empleados y características de sus productos o servicios:

- Innovación permanente.
- Construcción de confianza, respeto, compromiso y experiencia.
- Desarrollo de productos sobre la base de principios de calidad, eficacia y eficiencia.
- Liderazgo, proactividad, responsabilidad social y vanguardia tecnológica.
- Protección de la inversión de nuestros clientes.

3.Desarrollo

Como evidencia del trabajo realizado se presentarán los diagramas de caso de uso y de secuencia, dando una idea general de lo que se realizó en la práctica empresarial.

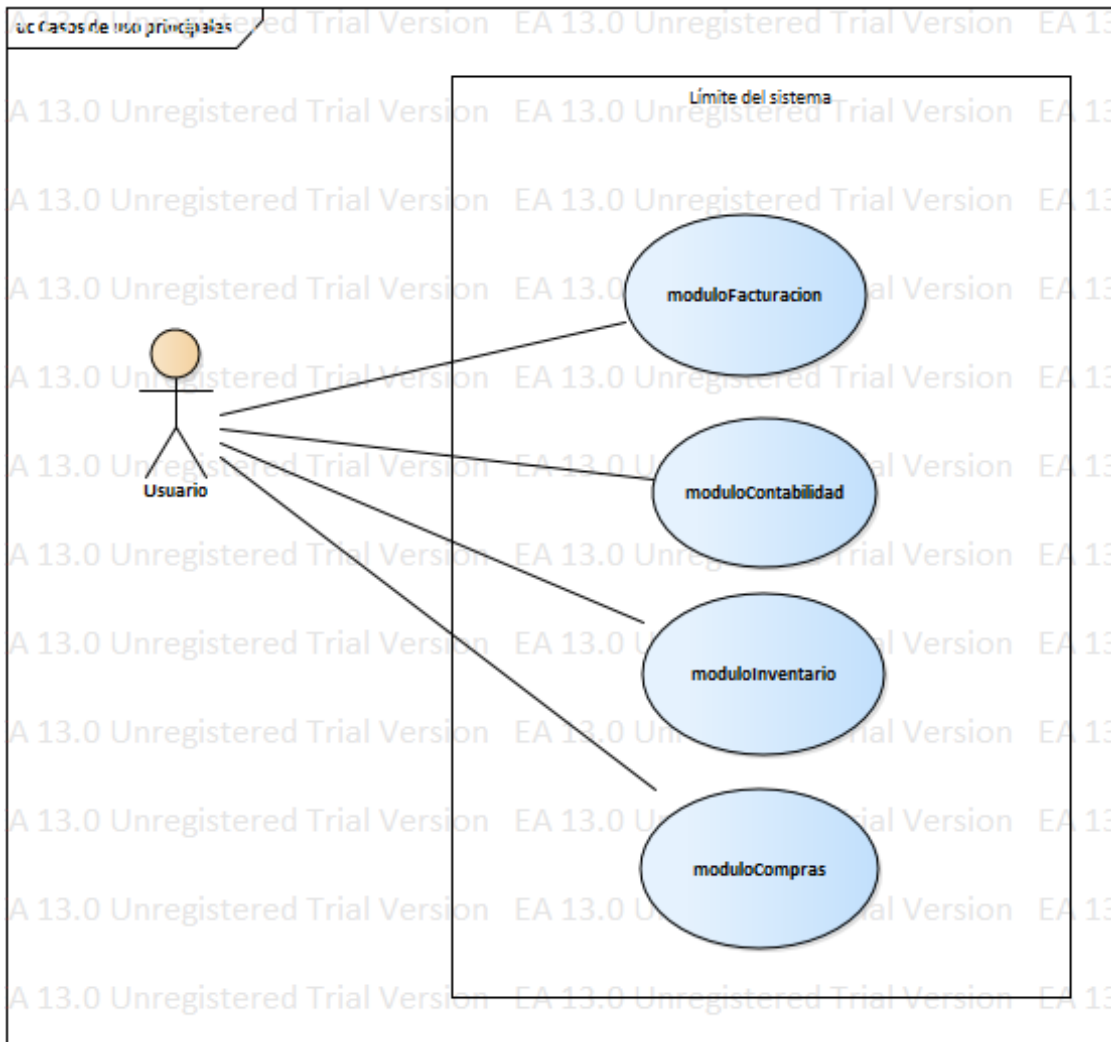
3.1. Definiciones

Administrador: Es el encargado de la configuración general del sistema y delegar permisos a los perfiles de los empleados agregados, es quien puede cancelar facturas si el producto es devuelto o cambiado.

Usuario Empleado: Puede efectuar todas las acciones permitidas por el administrador dentro de los módulos de facturación compras inventario contabilidad.

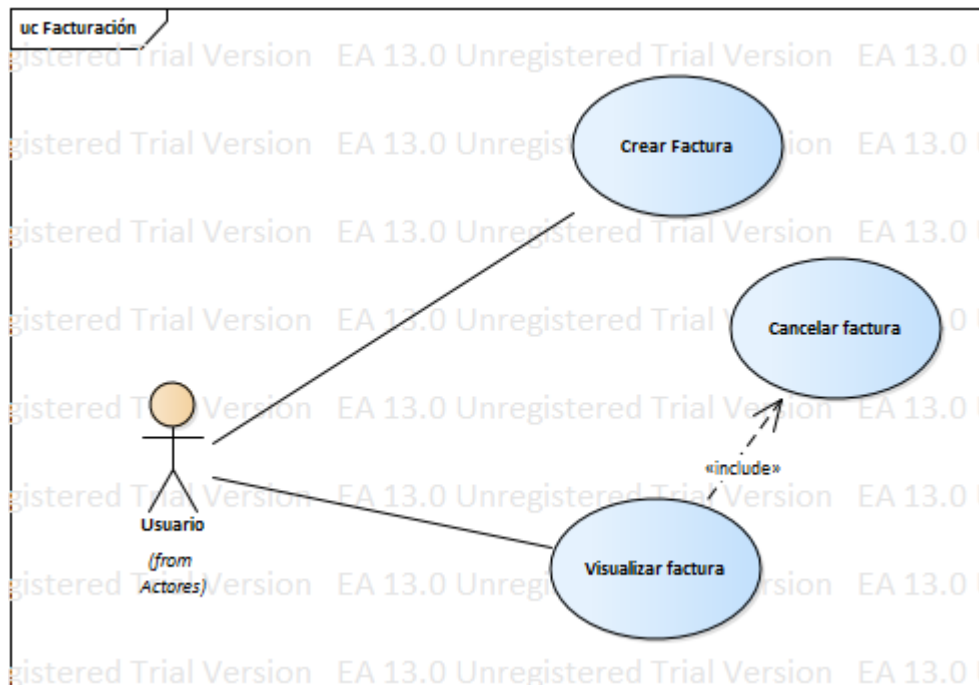
3.2. Casos de uso

A continuación, se encontrarán los diagramas de caso de uso:



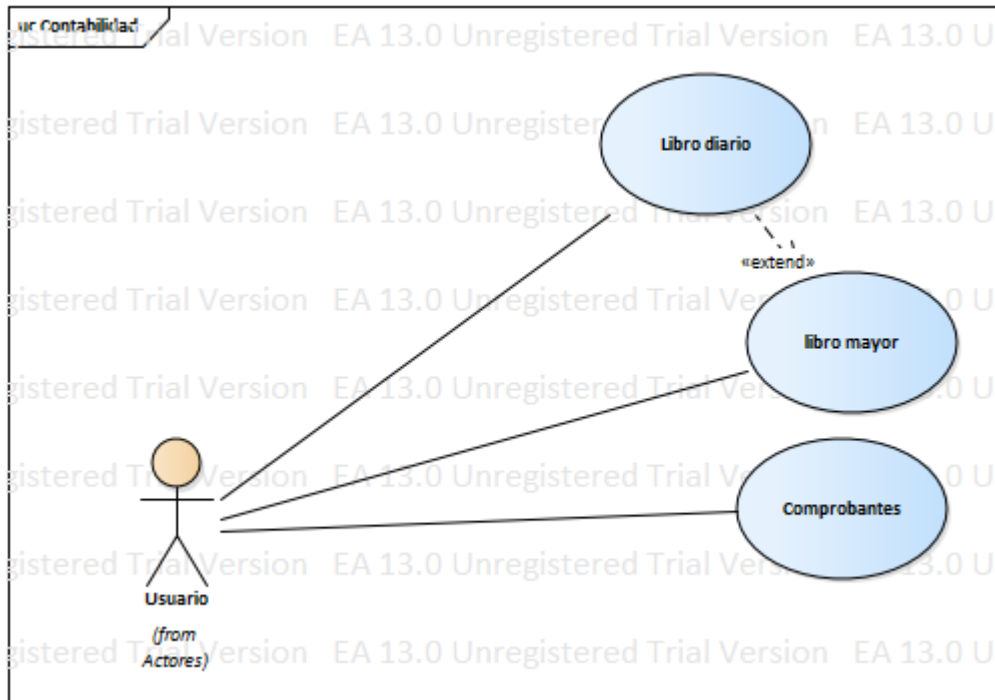
1. Diagrama general

Aquí se muestra el diagrama general, aquí se muestran los módulos de forma general, a continuación, se muestra como están distribuidos cada uno.



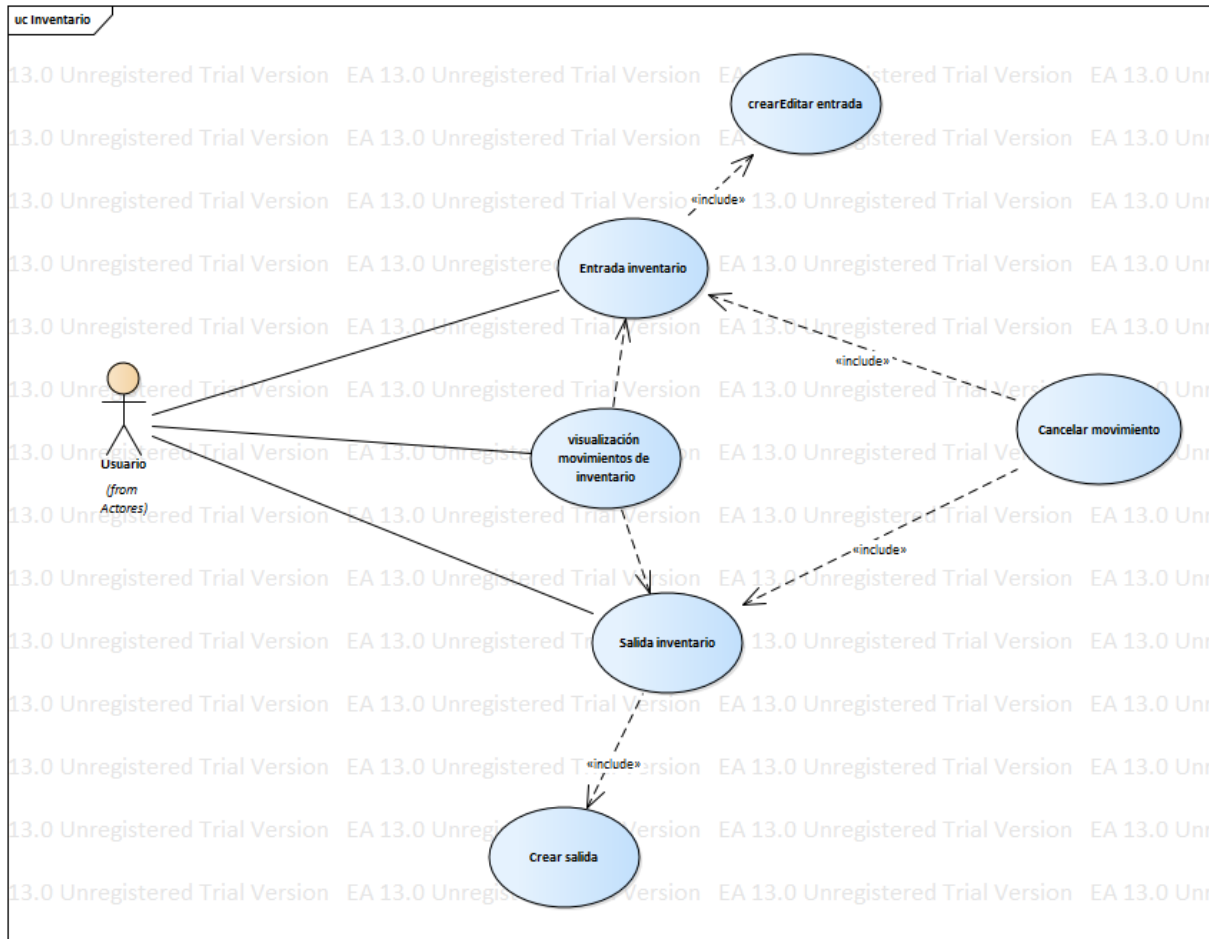
1.1. Módulo facturación

Este módulo está compuesto de sencillas funciones principales, como visualizar factura, en la cual el usuario podrá visualizar las facturas creadas y si posee los permisos necesarios podría cancelarla, además de crear nuevas facturas, lo cual se especificará más adelante con los diagramas de secuencia.



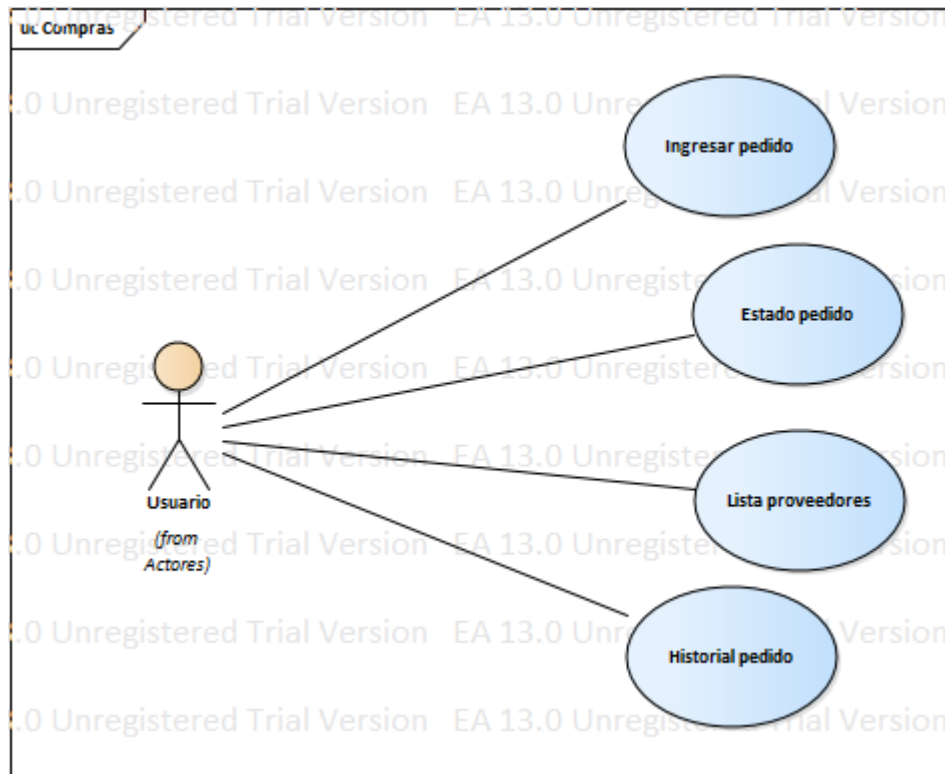
1.2. Módulo de contabilidad

En este módulo se encuentran las herramientas básicas necesarias para mantener un balance en cuanto al capital que está manejando la empresa, teniendo el libro mayor y el libro diario sincronizados se podrá tener un vistazo general y otro específico de lo que como se maneja el patrimonio empresarial.



1.3. Módulo de inventario

Uno de los módulos más importantes si se trata de una empresa que maneje gran variedad de productos, aquí se manejan por separados los procesos para mayor entendimiento a la hora del desarrollo y para que el producto final termine siendo igualmente entendible para el usuario.

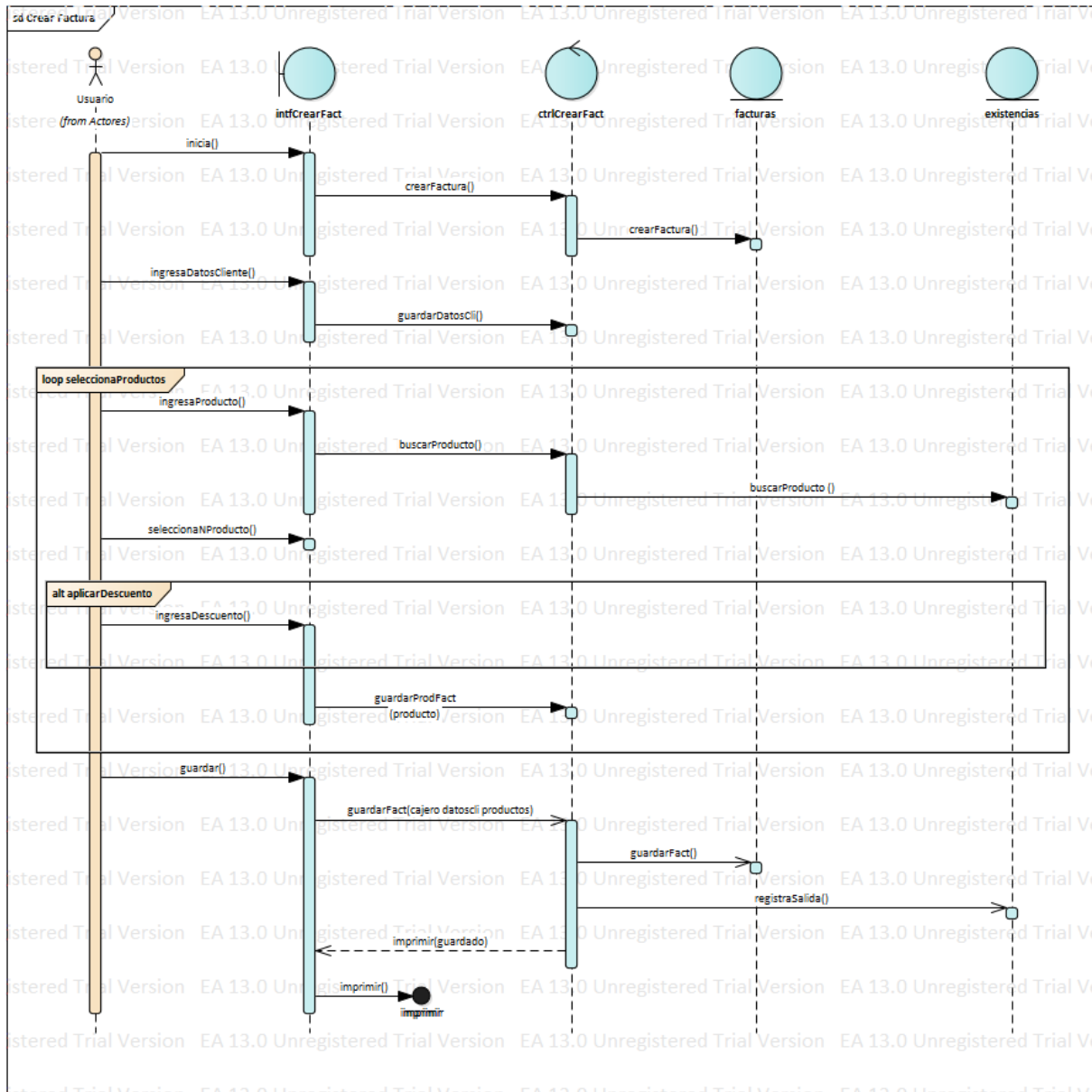


1.4. Módulo de compras

En este módulo el usuario podrá llevar el control de los productos solicitados y sus respectivos proveedores, llevando un historial general de las compras realizadas.

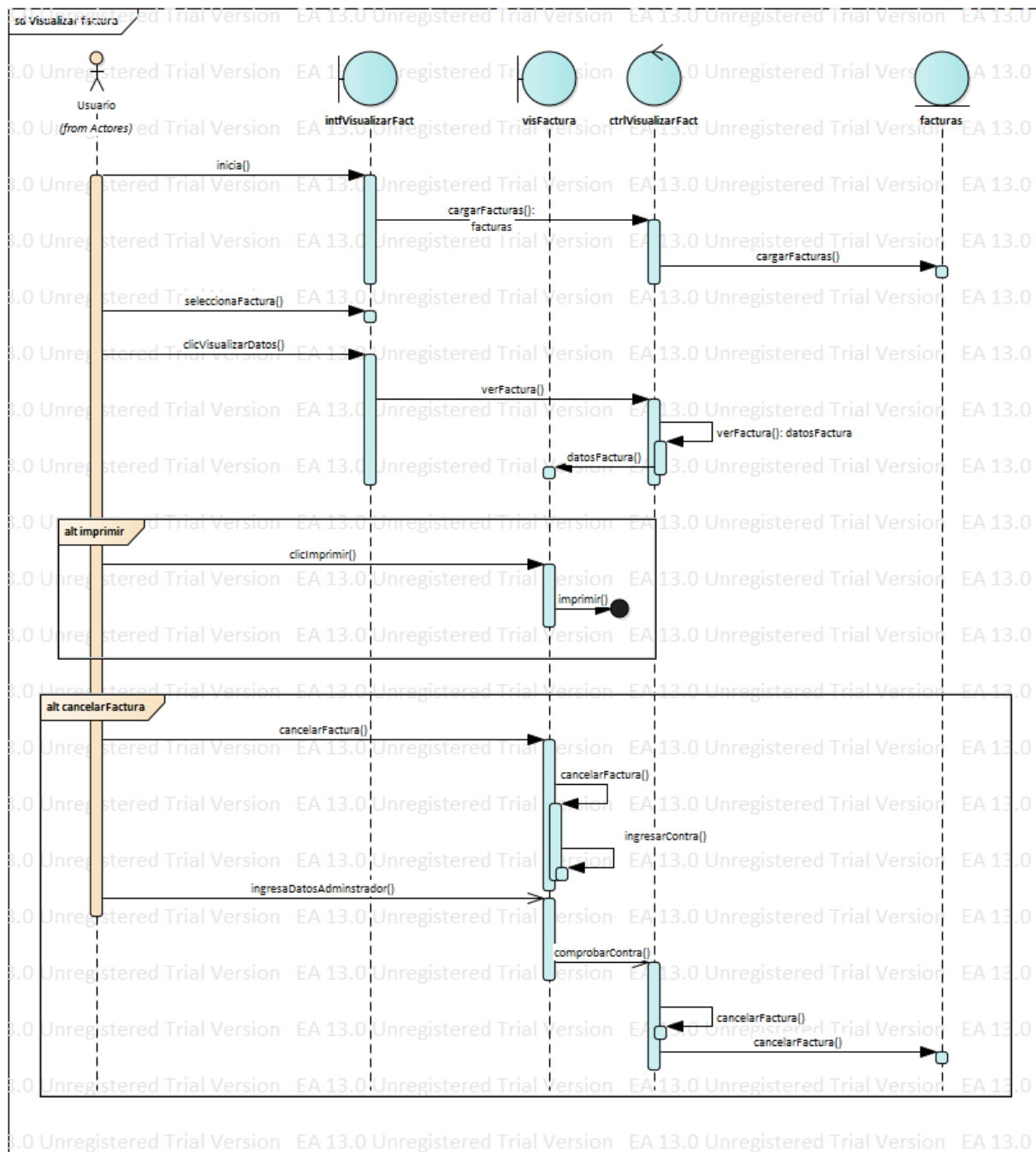
3.3. Diagramas de secuencia

3.3.1 Módulo de facturación



1.1.1. Crear factura

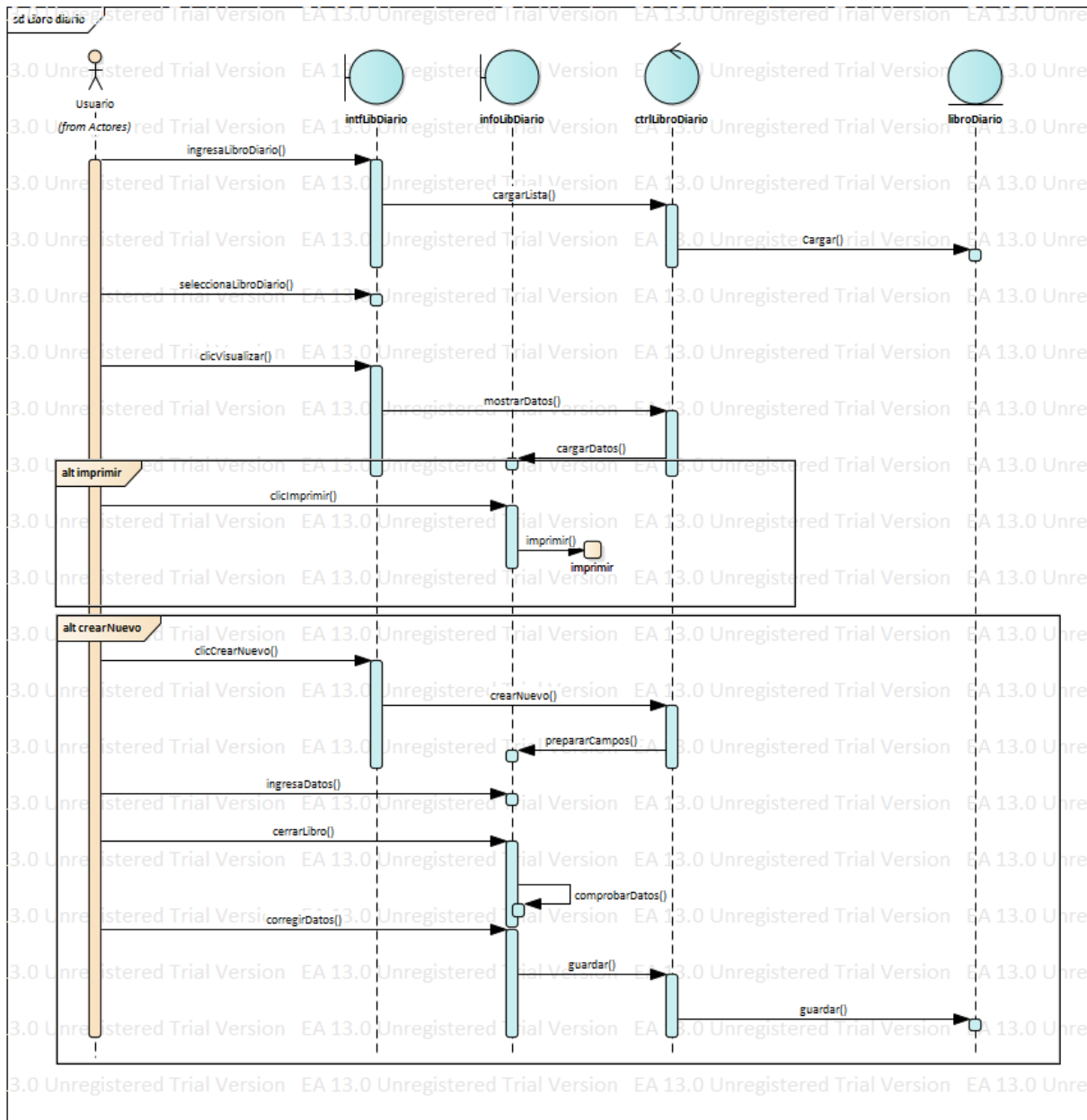
En este diagrama de secuencia se puede observar como el usuario interactúa con las funciones contenidas en la interfaz de crear factura, lo que se hizo aquí fue crear una factura lo más específica posible dando todos los campos básicos necesarios para una factura y guardándolas con su respectivo consecutivo.



1.1.2. Visualizar factura

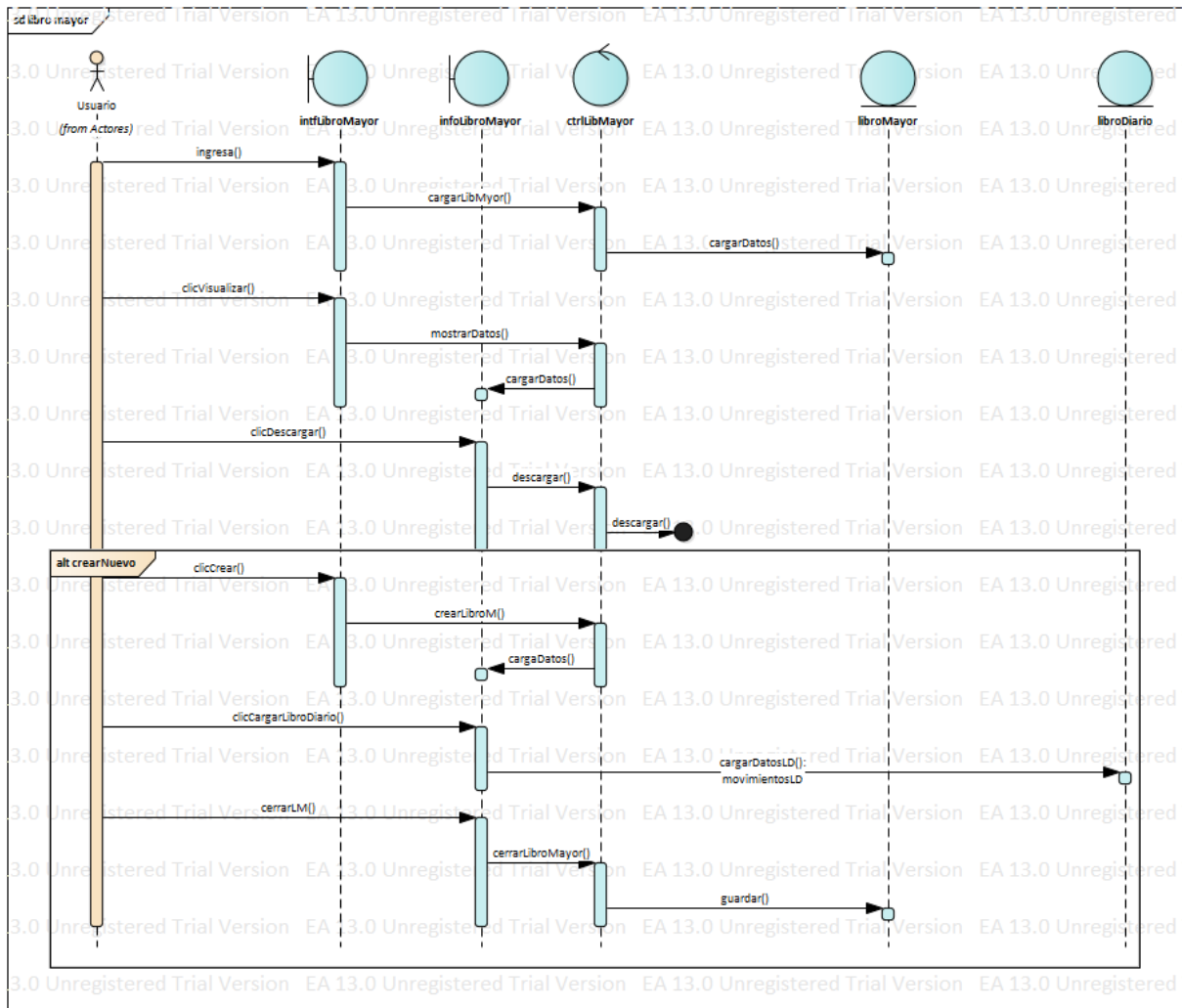
En este diagrama de secuencia se muestra lo que sería el proceso para visualizar una factura y lo que se debe de hacer para cancelar una de estas, dejando establecido que para la cancelación de estas se necesita una contraseña de nivel administrativo, aun así, estas se seguirán guardando con la etiqueta de cancelada.

3.3.2. Módulo de contabilidad



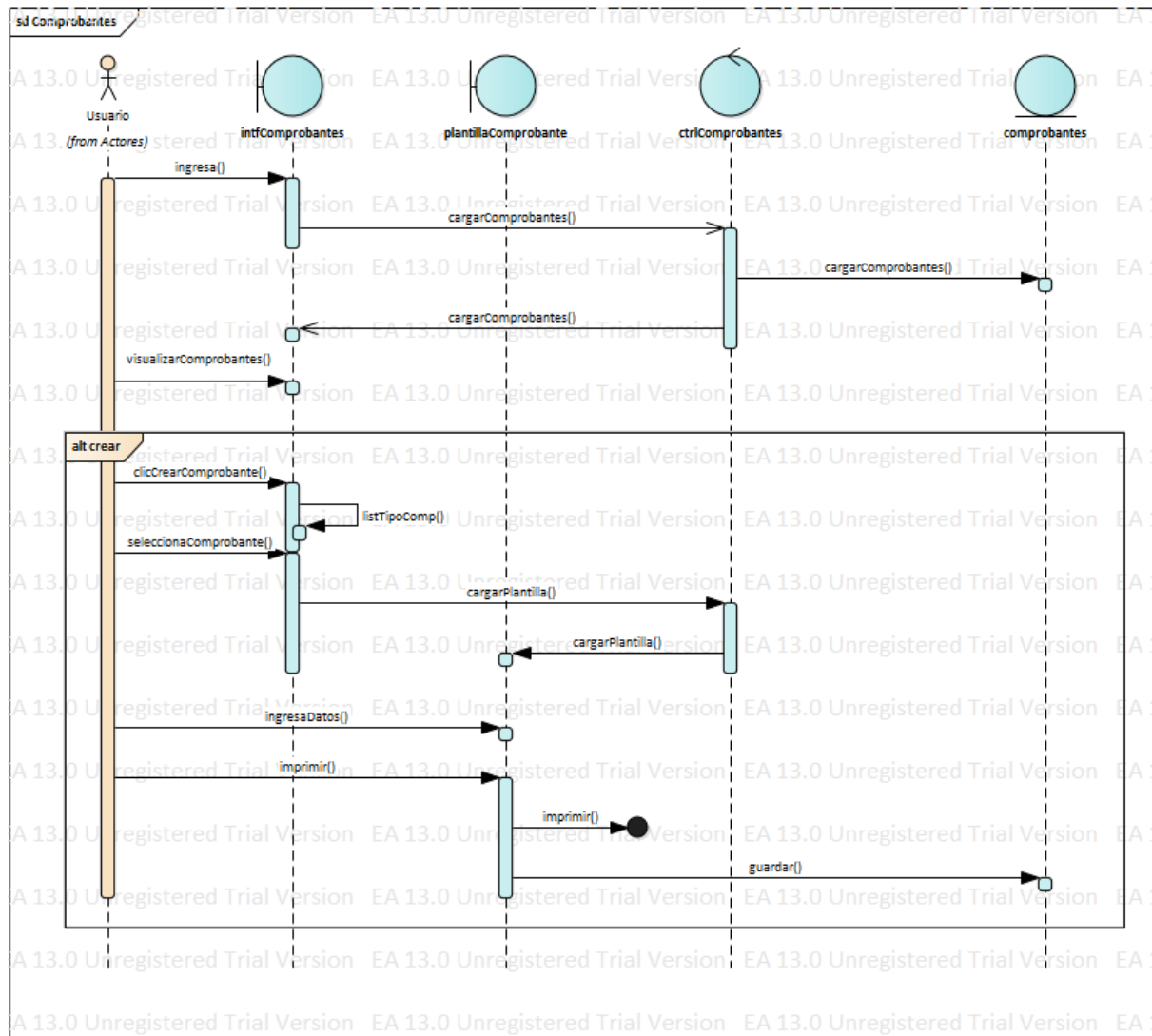
1.2.1. Libro diario

Aquí vemos lo que sería el funcionamiento de módulo de libro diario el cual nos servirá para recaudar los movimientos diarios de cualquier empresa en particular, cada vez que se cree una factura el valor y el consecutivo de este se agregara al libro diario, se puede editar la información de este para agregar información que este por fuera de lo automático, a partir de los libros diarios podremos crear el libro mayor.



1.2.2. Libro mayor

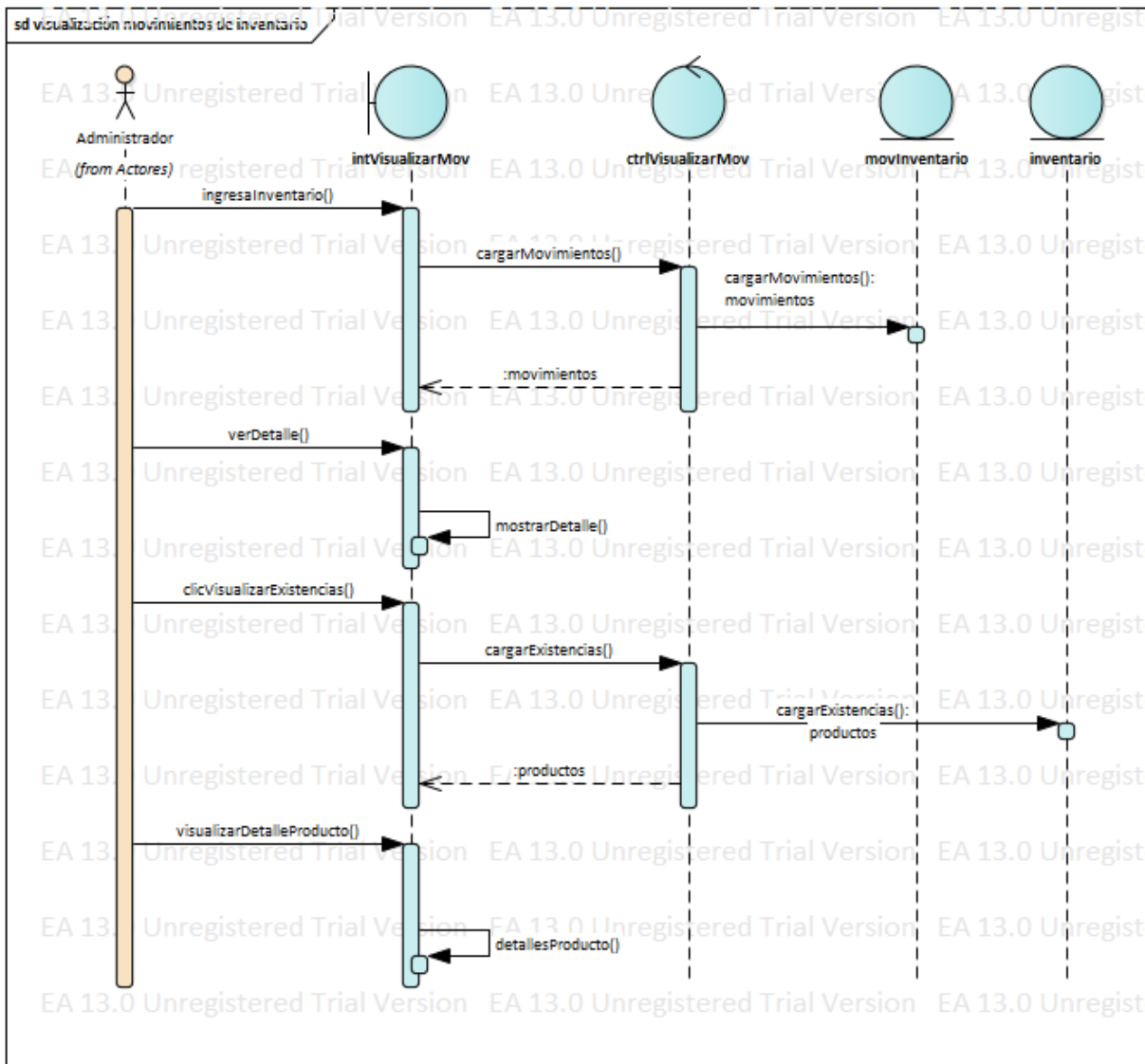
Aquí se muestra el funcionamiento del libro mayor, el cual sirve para llevar la contabilidad mensual y anual, tal como se ve en el diagrama a este se le podrán cargar los datos de los libros diarios que se seleccionen para crear el libro mayor, se puede además visualizar y descargar.



1.2.3. Comprobantes

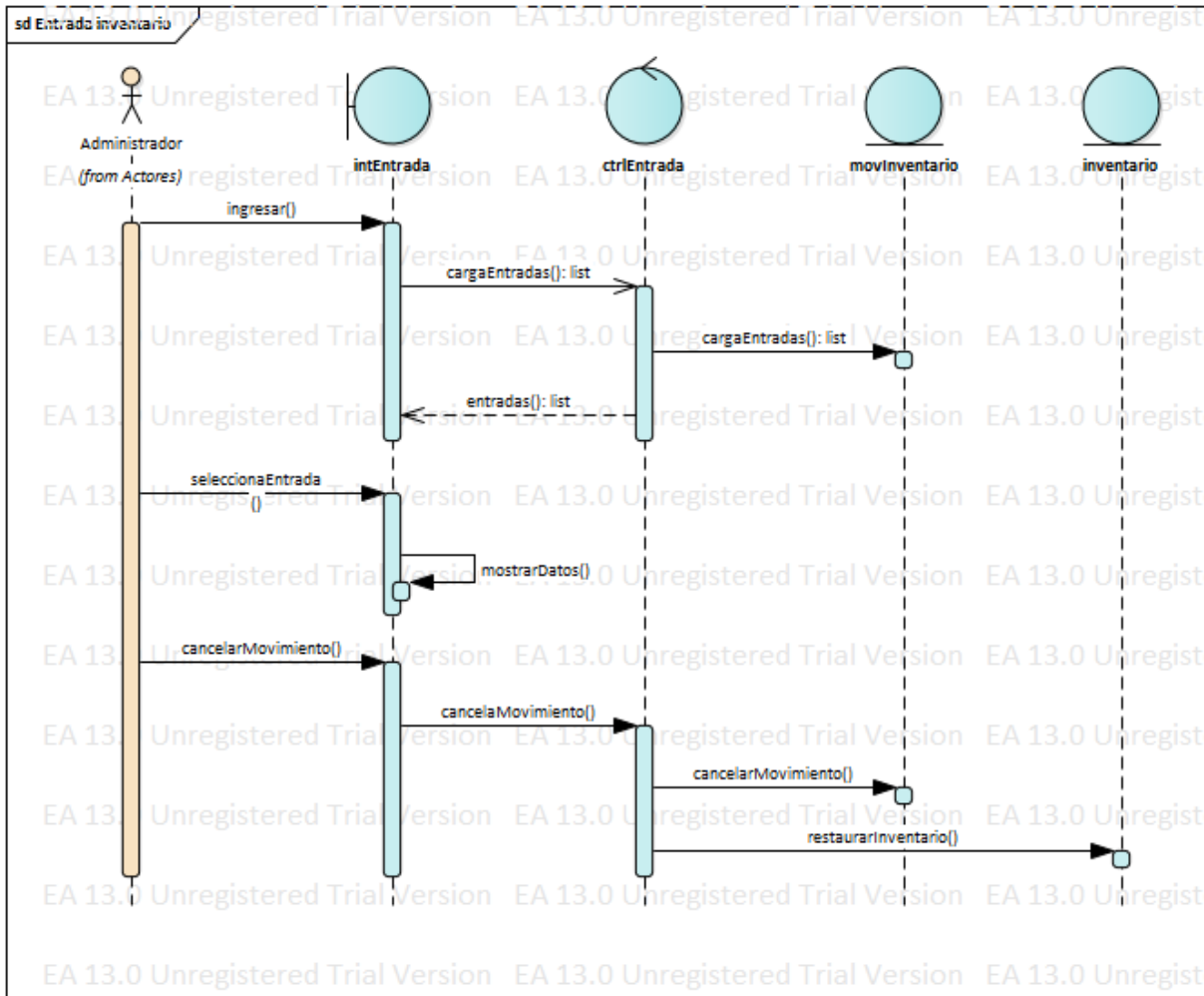
En este módulo se contienen distintas clases de plantillas para comprobantes de acuerdo a las que llegue a necesitar el usuario final.

3.3.3 Módulo de inventario



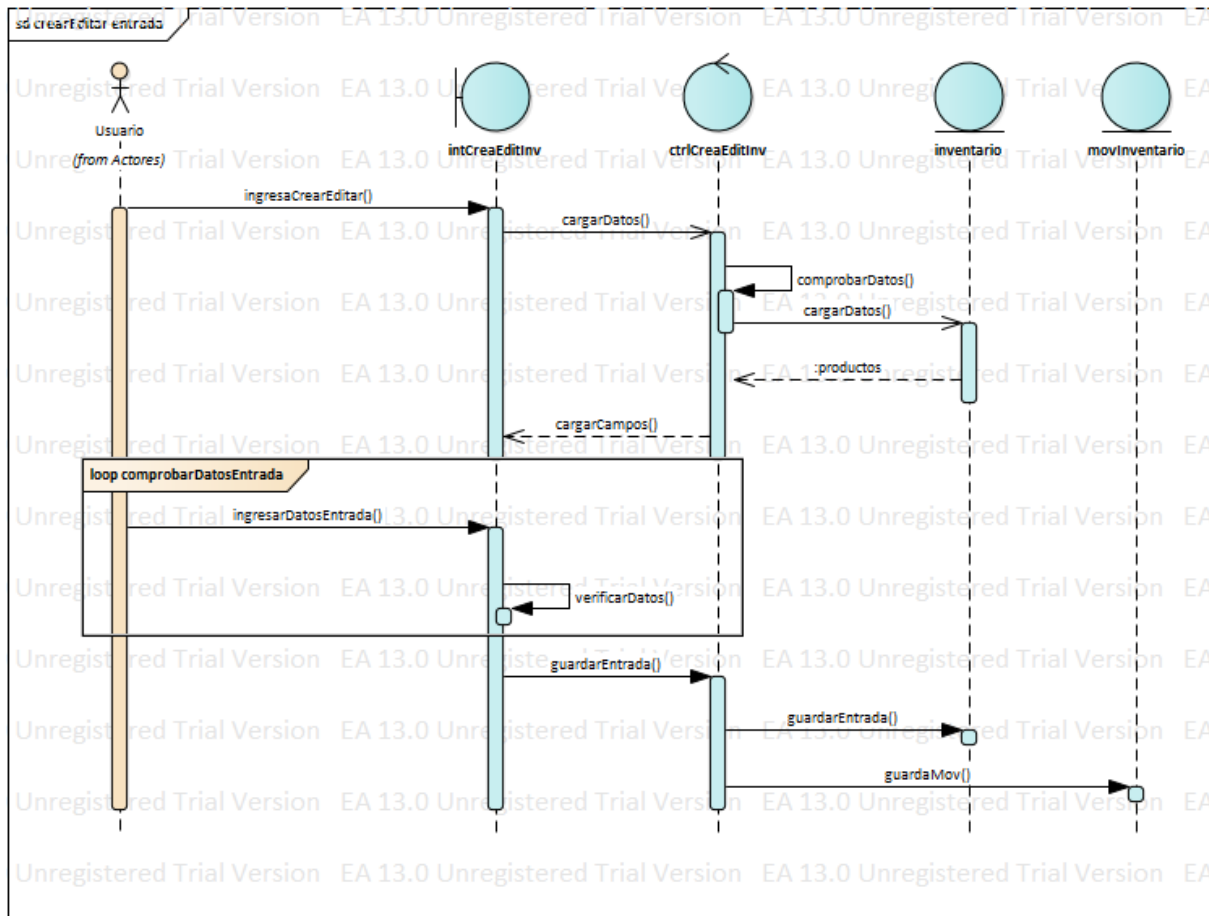
1.3.1. visualización de los movimientos de inventario

Esta es una de las vistas reservadas únicamente para los administradores, en la cual podrán visualizar todos los movimientos realizados con la respectiva información y los datos del usuario que realizo el movimiento.



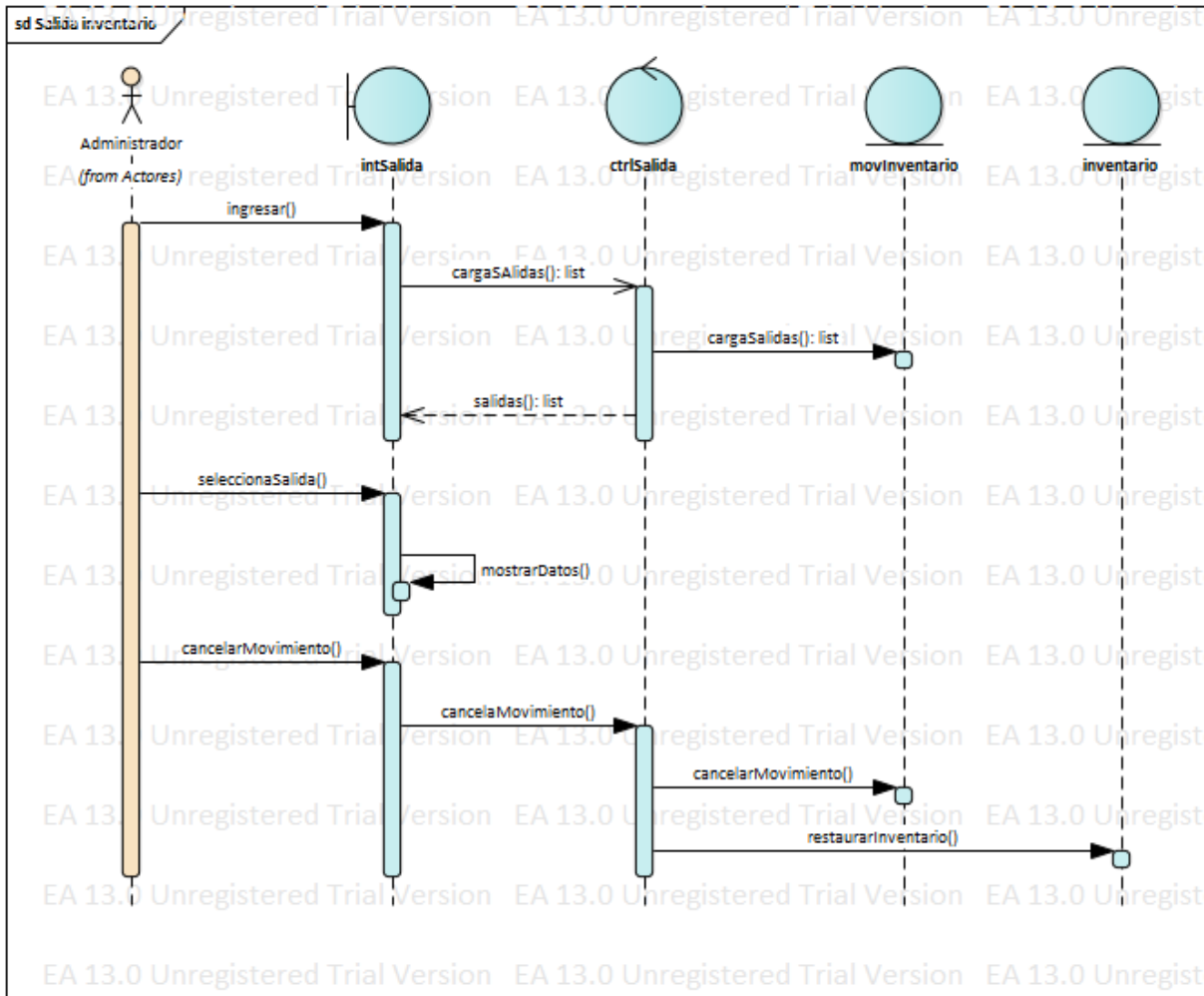
1.3.2. Entradas de inventario

Aquí el usuario visualiza las entradas que se han realizado, al seleccionar una se activan las opciones de editar o de cancelar movimiento.



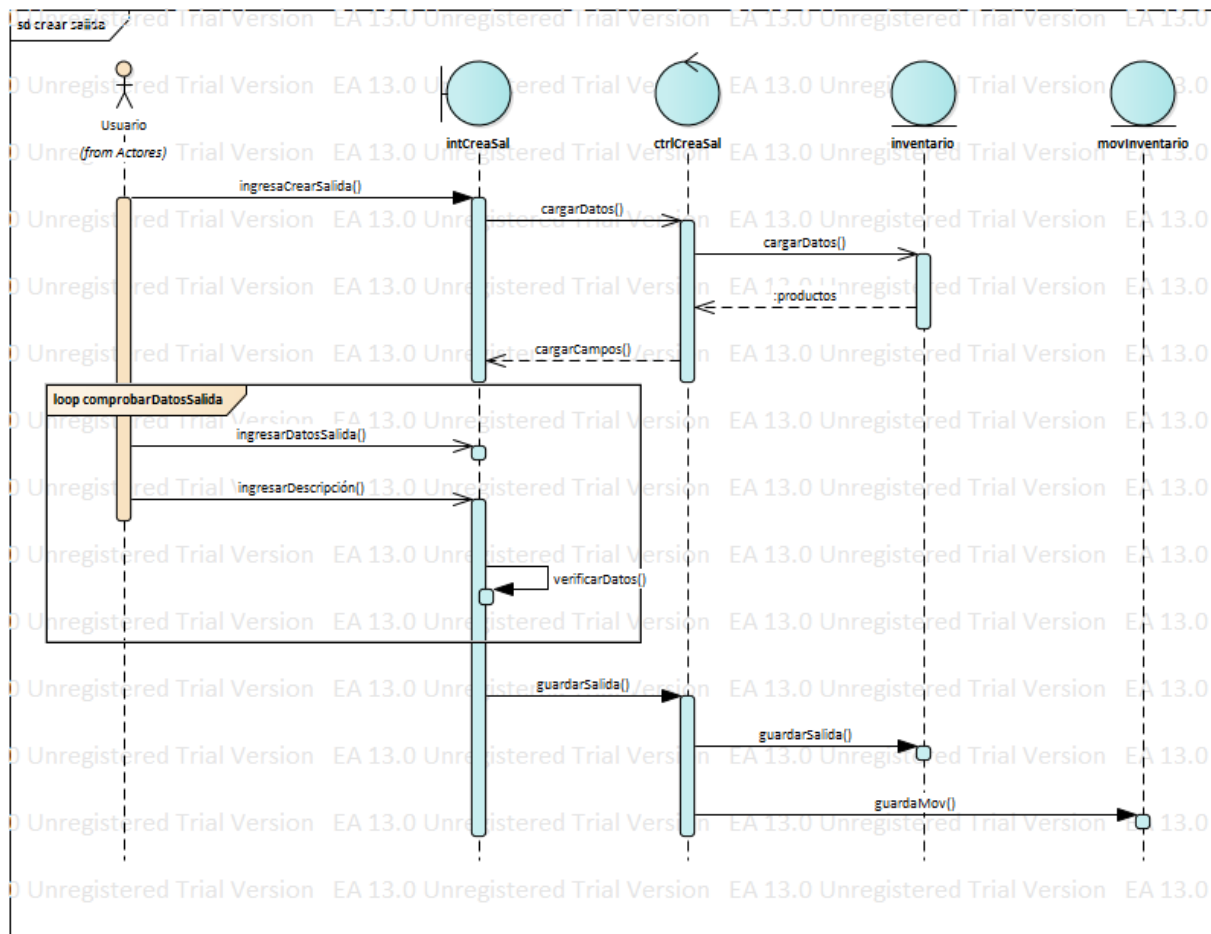
1.3.3. Crear o editar entradas de inventario

Si el usuario entra a la opción de crear nueva entrada o de editar alguna de las existentes entra a esta interfaz en la cual puede registrar los productos o modificar algún dato que haya quedado erróneo.



1.3.4. Salidas de inventario

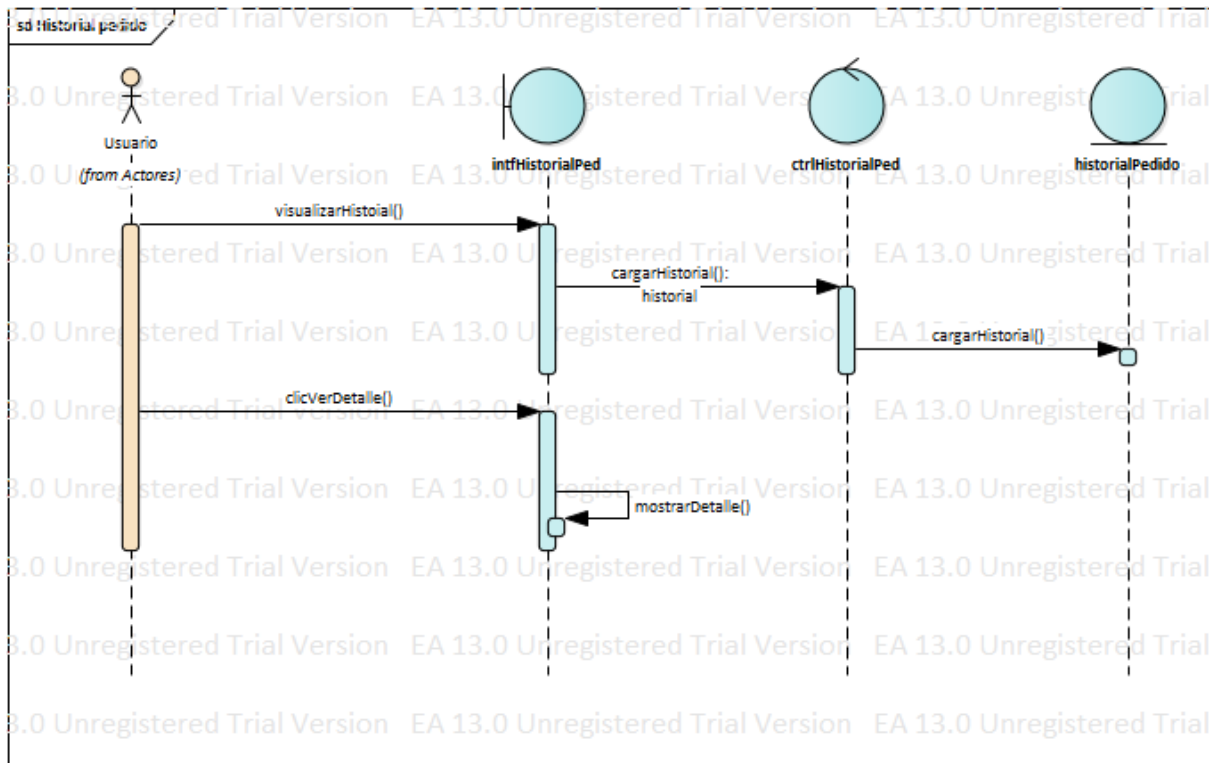
Al igual que en entradas de inventario, aquí se muestra una lista filtrable de las salidas que se presentaron, al seleccionar una se pueden visualizar sus detalles y cancelar el movimiento.



1.3.5. Crear salida inventario.

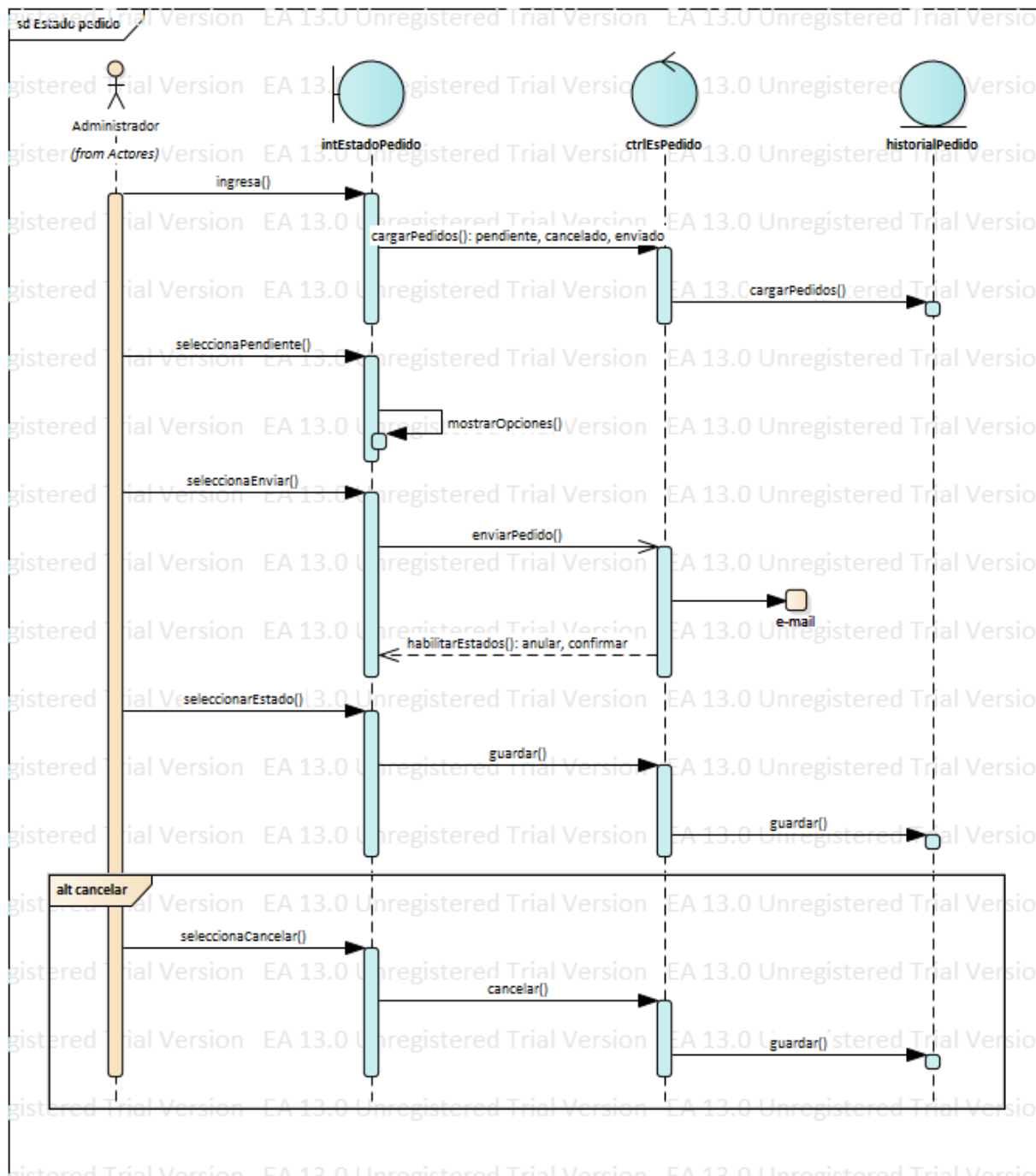
En este diagrama de secuencia se muestran las especificaciones básicas para poder cancelar un movimiento, en la cual se indica que producto y cantidad va a salir, además de la descripción del por qué.

3.3.4 Módulo de compras



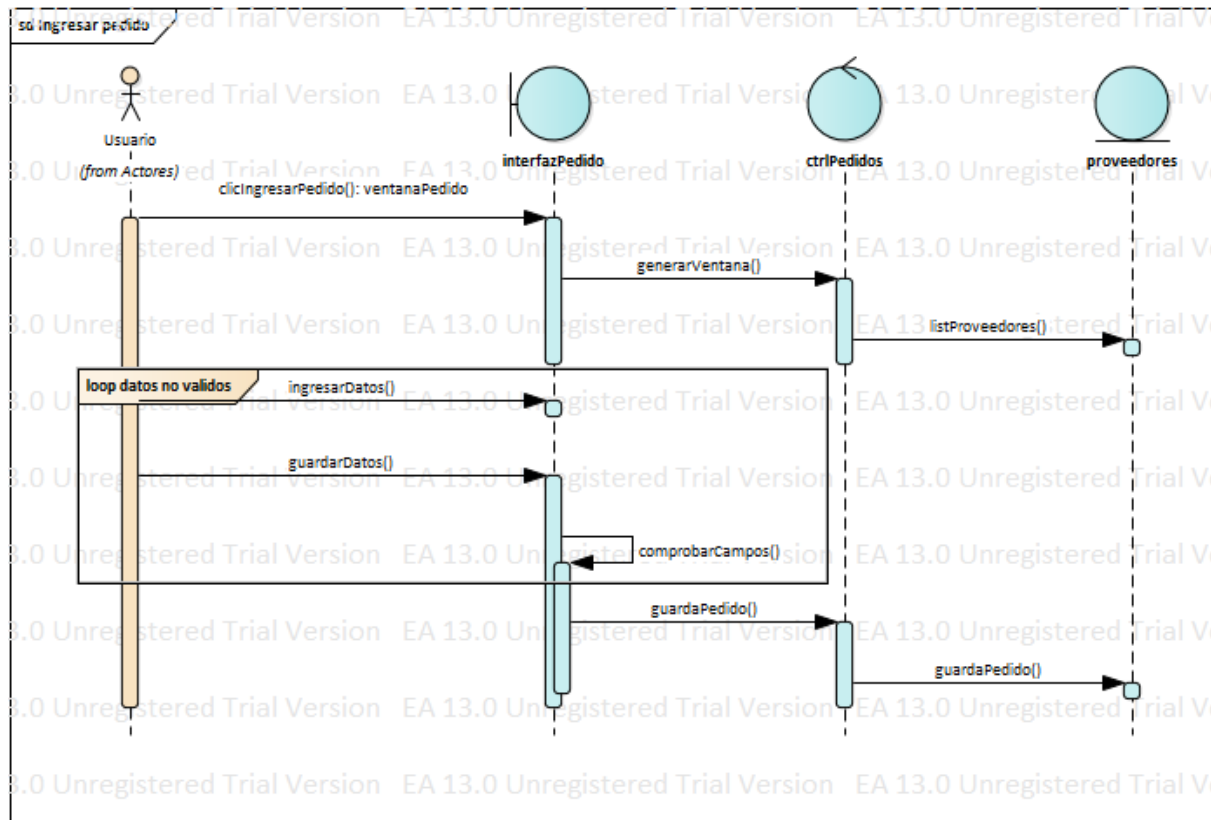
1.4.1. Historial de pedido

Aquí se muestra como el usuario puede visualizar el historial y seleccionar alguno de los elementos para visualizar los detalles de este.



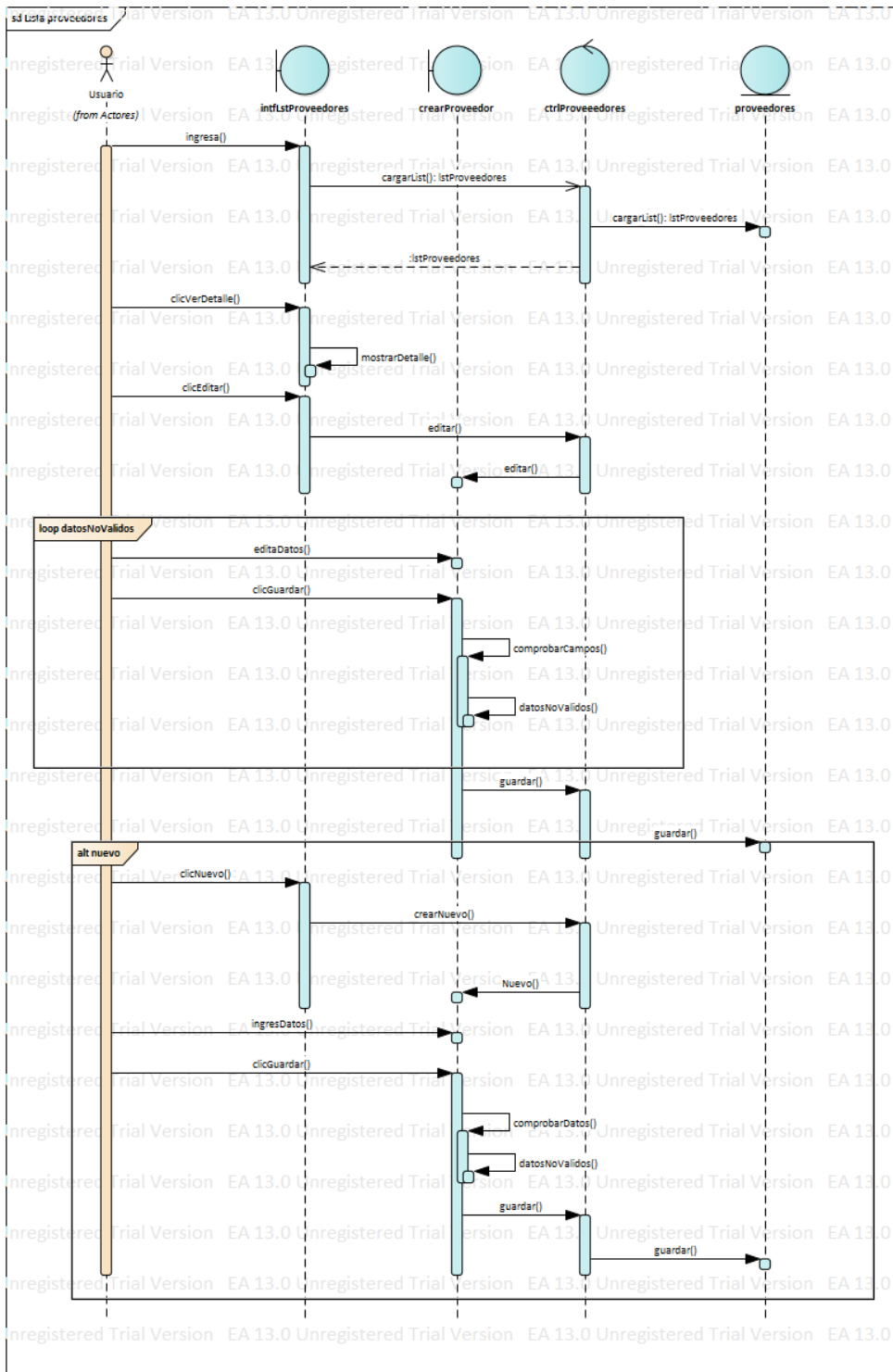
1.4.2. Estado pedido

En este módulo se visualizan todos los pedidos realizados y el estado en el que se encuentren, una vez estén completados se pueden enviar como lista al email del proveedor, una vez enviados no se pueden cancelar.



1.4.3. Ingresar pedido

Aquí se puede ver la manera en la que se generan los pedidos a los proveedores, el pedido se guarda, pero queda en estado pendiente y es editable hasta que se envía.



1.4.4. Lista proveedores

Aquí se ingresan todos los proveedores que posea la empresa, se organizan en una lista donde el usuario puede ver la descripción dada y los detalles del proveedor.

4. Análisis de resultados

Como resultado de la práctica realizada se ha probado la importancia de desarrollar un proyecto a partir de la documentación inicial, en la cual se elaboran diagramas UML a partir de los requerimientos, permitiendo así que el área encargada del desarrollo tenga una idea certera de la idea final de producto que se tiene como proyecto, otra de las ventajas de tener los modelos UML es que gracias al diseño y mapeo que se le da a lo que será el software es que el desarrollo puede ser distribuido entre varios desarrolladores.

Como característica general, para proyectos de gran complejidad, se notará de gran manera la disminución en los costos de tiempo y mantenimiento del software a comparación de cuando no se usan estas buenas prácticas de ingeniería de software.

Para agregar, como resultado personal de la práctica me han quedado conocimientos y experiencias relevantes a la hora de ejercer dentro de una empresa y el adaptarse a las metodologías de estas para que todo funcione de una manera más armoniosa.

5. Conclusiones

El trabajo de practica que se realizo fue logrado gracias a conocimientos previos y los adquiridos en la investigación y por el apoyo de mis compañeros que con mucho agrado me compartieron su experiencia.

Dentro del tiempo de la practica tuve la oportunidad de ver la importancia de la ingeniería de software al desarrollar un proyecto, ya que da por sentadas las bases, tiempos e idea del producto final antes de comenzar la fase de programación y durante la misma.

Al dejar la documentación del proyecto, este podrá ser realizado en el tiempo que la empresa lo requiera, lo que nos deja con la certeza de un trabajo realizado completamente.

6. Bibliografía

- OBJETOS Y SISTEMAS RELACIONALES OSR S.A.S. Disponible en línea:
<http://www.softwareosr.com/>
- Larman, C. (1999). *UML y Patrones*. Pearson.
- López, R. O., & Ramiro, L. V. (2011). *Ingeniería del software*. Centro de Estudios Financieros.