**Proyecto de Titulo**

**Sistema de Apoyo de administración de fichas y control de inventario de Lentes de contacto (SAFCIC)**

Sr. Emilio Arancibia Riquelme.

Sr. Juan Hiriarte Torres.

Profesor guía: Rodrigo Espinola Guerrero

Junio 2016

# Índice

[Índice 2](#_Toc454005064)

[1. Introducción 5](#_Toc454005065)

[2. Descripción de la empresa 6](#_Toc454005066)

[2.1 MISIÓN 6](#_Toc454005067)

[2.2 VISIÓN 6](#_Toc454005068)

[2.3 CALIDAD DE SERVICIO 7](#_Toc454005069)

[2.4 RESPETO 7](#_Toc454005070)

[2.5 NUESTRO PERSONAL 7](#_Toc454005071)

[2.6 ESTRUCTURA ORGANIZACIONAL 7](#_Toc454005072)

[3. PROBLEMATICA 8](#_Toc454005073)

[4. Objetivos. 9](#_Toc454005074)

[4.1 Objetivos Generales. 9](#_Toc454005075)

[4.2 Objetivos Específicos. 9](#_Toc454005076)

[5. Motivación 10](#_Toc454005077)

[6. Propuestas de la solución (lenguajes de programación) 11](#_Toc454005078)

[7. Propuesta de solución recomendada 12](#_Toc454005079)

[8. Planificación 13](#_Toc454005080)

[9. Modelo de negocios 15](#_Toc454005081)

[10. Definición de requerimiento 17](#_Toc454005082)

[10.1 Requerimientos funcionales 18](#_Toc454005083)

[10.2 Requerimientos No funcionales 19](#_Toc454005084)

[10.3 Requerimientos de Dominio 19](#_Toc454005085)

[11. Modelo de clases conceptuales 20](#_Toc454005086)

[12. Casos de uso 21](#_Toc454005087)

[13. Descripción de los casos de uso 23](#_Toc454005088)

[13.1.1 Uc-01 Generar venta 23](#_Toc454005089)

[13.1.2 Diagrama de secuencia Generar venta 26](#_Toc454005090)

[13.1.3 Contrato de Operaciones “Generar Venta”. 27](#_Toc454005091)

[13.1.3 Diagrama de colaboración Generar venta 28](#_Toc454005092)

[13.2.1 Uc-02 Crear ficha 29](#_Toc454005093)

[13.2.2 Diagrama de Secuencian Crear ficha 32](#_Toc454005094)

[13.2.3 Contrato de Operaciones “Crear Ficha”. 32](#_Toc454005096)

[13.2.4 Diagrama de Colaboración Crear ficha 33](#_Toc454005097)

[13.3.3 Uc-03 Buscar 34](#_Toc454005098)

[13.3.2 Diagrama de Secuencia Buscar ficha 36](#_Toc454005099)

[13.3.3 Contrato de Operaciones “Buscar”. 36](#_Toc454005100)

[13.3.4 Diagrama de colaboración Buscar ficha 37](#_Toc454005101)

[13.4.1 Uc-04 Modificar ficha 38](#_Toc454005102)

[13.4.2 Diagrama de secuencia Administrar ficha 40](#_Toc454005103)

[13.4.4 Contrato de Operaciones “Administrar Ficha”. 40](#_Toc454005104)

[13.4.3 Diagrama de colaboración administrar ficha 41](#_Toc454005105)

[13.5.1 Uc-05 Generar OT 42](#_Toc454005106)

[13.5.2 Diagrama de secuencia Generar OT 44](#_Toc454005107)

[13.5.3 Contrato de Operaciones “Generar Orden de Trabajo”. 45](#_Toc454005109)

[13.5.3 Diagrama de colaboración Generar OT 46](#_Toc454005110)

[13.6 .1 Asignar control 47](#_Toc454005111)

[13.6.2 Diagrama de secuencia Asignar control. 49](#_Toc454005112)

[13.6.3 Contrato de Operaciones “Asignar Control”. 49](#_Toc454005113)

[13.6.4 Diagrama de colaboración Asignar control 50](#_Toc454005114)

[13.7.1 Uc-07 Generar Reportes 51](#_Toc454005115)

[13.7.2 Diagrama de secuencia Generar reporte 53](#_Toc454005116)

[13.7.3 Contrato de Operaciones “Generar Reporte”. 53](#_Toc454005117)

[13.7.4 Diagrama de colaboración Generar reporte 54](#_Toc454005118)

[13.9.1 Uc-07 Login 55](#_Toc454005119)

[13.9.2 Diagrama de secuencia login 57](#_Toc454005120)

[13.9.3 Contrato de Operaciones “Login”. 58](#_Toc454005121)

[13.9.9 Contrato de Operaciones “Login”. 58](#_Toc454005122)

[14. Diagrama de clases 59](#_Toc454005124)

[15. Modelo Relacional 60](#_Toc454005125)

# Introducción

En la actualidad las ópticas luchan por tener un lugar en el mercado, para ello se valen de diversas herramientas que hoy existen para mejorar y optimizar los procesos internos y externos de estas, sin lugar a duda una de estas herramientas es la tecnología.

Hoy en día las ópticas y las empresas en general hacen uso de esta tecnología a través de la computación con el fin de automatizar y estandarizar los procesos administrativos y de ventas, lo que deriva en la entrega más rápida y oportuna de la información y atención al cliente. La información es un elemento fundamental en la toma de decisiones, puesto que ella nos orienta y nos ayuda a verificar las metas y objetivos planteados

Es por esto que la óptica “TECNO-LENS” ha decidido sumarse e integrar el plus que genere comenzar a utilizar las herramientas tecnológicas que hoy en día se encuentran disponibles para dar una mejor y más rápida atención a nuestros clientes.

La óptica “TECNO-LENS” ha solicitado un sistema que entregue apoyo en las labores que se realizan en el área de clientes para así mejorar y agilizar los procesos relacionados con la información de los clientes y sus ventas.

El presente informe tiene como objetivo dar a conocer el proyecto que aportara a la óptica “TECNO-LENS”, este corresponde a un sistema computacional de escritorio que permitirá a las personas encargadas de la atención al cliente a brindar un más rápido y mejor servicio .

Para el desarrollo del proyecto se utilizaran todos los conocimientos y herramientas que se han adquirido en el proceso de estudio de la carrera “Análisis de Sistemas”, otorgados por el instituto tecnológico de Chile “ITC”.

# Descripción de la empresa

Tecno Lens Ltda. fue fundada a principios del año 2000 y cuenta con más de 13 años de experiencia en el mercado de la fabricación y comercialización de lentes Ópticos y de Contacto, destacando durante este periodo por su compromiso y responsabilidad de cada uno de nuestros clientes.

## 2.1 MISIÓN

Nuestra misión es constituirnos en la empresa más eficiente, comprometida y preocupada de entregar una atención integral, especializada y siempre mejorando la calidad de vida de cada uno de nuestros clientes.

## 2.2 VISIÓN

Porque conocemos las necesidades de nuestros clientes y entendemos que el tiempo de cada uno de ellos es valioso, es que ofrecemos una atención rápida y personalizada en cada una de nuestras sucursales. Destacando dentro de los servicios entregados la realización de chequeos oftalmológicos, de acuerdo a los requerimientos de nuestros clientes. Consecuente con lo anterior, nuestros chequeos están orientados a detectar problemas en la salud visual de nuestros clientes y satisfacerlos en forma seria, responsable y profesional en las áreas en que comprometemos nuestra atención.

Esta política que junto con diferenciarnos de la competencia, nos ha proyectado como una Empresa de gran carácter y seriedad empresarial.

## 2.3 CALIDAD DE SERVICIO

La calidad de nuestro servicio no sólo se limitará a un buen trato a nuestros clientes y usuarios, sino que se extiende a todos nuestros productos en sí, los cuales son confeccionados teniendo en cuenta los más altos estándares de calidad, además todos cuentan con garantía en nuestras propias instalaciones.

## 2.4 RESPETO

Nuestra empresa a lo largo de su historia ha demostrado un respeto por cada uno de nuestros clientes, teniendo siempre en cuenta su necesidad de mejorar la calidad de vida mediante el cuidado de su salud visual.

## 2.5 NUESTRO PERSONAL

Nuestro personal se ha especializado en prestar un servicio integral y personalizado, de acuerdo a las necesidades específicas de cada cliente y manteniendo en todo momento una flexibilidad que garantice su total satisfacción. Una de las principales ventajas competitivas de nuestra Empresa consiste precisamente en nuestro personal, altamente calificado e involucrado con cada cliente. Lo demuestra una rápida, oportuna y eficaz respuesta a sus necesidades.

## 2.6 ESTRUCTURA ORGANIZACIONAL



La óptica TECNO-LENS tiene una organización jerárquica, con distintas áreas especializadas, la cual permite una atención más personalizada de las distintas áreas.

# PROBLEMATICA

La problemática de la óptica “TECNO-LENS” surge por el almacenamiento de fichas de contactología.

Actualmente este proceso se realiza con fichas personales de pacientes, la cual es material de papel, esto produce que semanalmente se acumule gran cantidad de fichas ya que por promedio se atiende un flujo de pacientes bastante alto.

Por otro punto estas fichas al ser de material orgánico, producen que a través del tiempo se vallan deteriorando y borrando la información.

También produce el problema de tardanza en el llenado de estas ya que se debe realizar de forma manual. Junto con la demora que implica cada vez que el profesional “contactólogo” solicita una ficha del paciente que llega a la óptica.

También existe el problema de duplicidad de fichas de pacientes, ya que si la ficha no es encontrada se crea una nueva. Lo que genera que muchas veces los pacientes tengan dos fichas.

En cuanto a los lentes de contacto, al no llevar un inventario de lo que está en stock, muchas veces se producen ventas que no hay disponibilidad inmediata, lo que provoca perdida de ventas, así como también demora en las entregas de estos y por lo mismo no estamos gestionando una buena atención a nuestros clientes.

# Objetivos.

Analizada la toma de requerimientos y la problemática podemos deducir los siguientes objetivos para la solución:

## 4.1 Objetivos Generales.

Se pretende crear un sistema que permita el almacenaje de fichas de pacientes de contactología además de poder modificar y consultar estas, facilitando la tarea al área de ventas y mejorando la atención de pacientes, mejorando los tiempos de búsqueda y registro y dejando casi nula la pérdida de información

En cuanto el inventario, se busca consultar en el momento de la venta por el stock disponible, así evitar posibles quiebres de stock

## 4.2 Objetivos Específicos.

• Desarrollar una aplicación de escritorio, para la óptica “TECNO-LENS”, que colabore a la gestión del área de ventas, a través de una interfaz amigable e intuitiva que permita apoyar las labores que se realizan el área de pre y post venta.

• Permitir tomar decisiones en lo que refiere al precio de compras de lentes de contactos de los clientes.

• Administrar fichas de clientes que permita crear, modificar y consultar datos de clientes.

• Verificar al momento de realizar venta si hay stock de lentes de contacto para su posterior venta.

• Diseñar casos de uso relativos a la funcionalidad del sistema

• Diseñar diagramas de secuencia y clases en base a casos de uso

• Diseñar modelo entidad relación que permita a través de ella representar la interacción entre las distintas entidades existentes en el área de clientes.

• Determinar el modelo relacional basado en el modelo entidad relación, este permitirá establecer la estructura adecuada de la base de datos que permitirá registrar todo lo relacionado a la información de la ficha de cada cliente, el stock que existe para realizar las ventas.

• Generar la base de datos de la aplicación, en base a los modelos entidad relación y el modelo relacional que fueron diseñados anteriormente.

• Diseñar los formularios o ventanas que permitan el ingreso de los datos que correspondan a los requerimientos del área de clientes de la óptica “TECNO-LENS”.

• Recuperar datos desde la base de datos y mostrarlos en los formularios o ventanas que fueron diseñadas para la mantención de datos.

# Motivación

La motivación principalmente se debe a que después de 3 años trabajando en la empresa TECNOLENS hemos notado los inconvenientes que se suele tener con el almacenaje de las fichas de contactología, por esto mismo sabemos muy bien cuál es la problemática y la solución que se puede implementar, además con los conocimientos adquiridos durante nuestro estudio podemos lograr una herramienta que de verdad sea de mucha ayuda a la administración y a la atención de pacientes en la óptica.

# Propuestas de la solución (lenguajes de programación)

Nuestras primeras opciones de lenguaje para realizar la solución son:

**C#**: Como lenguaje de última generación y de propósito general, es orientado a objetos. C# no permite la inclusión de funciones ni variables globales que no estén incluidos en una definición de tipos, por lo que la orientación a objetos es más pura y clara que en otros lenguajes como C++. Además C# soporta todas las características del paradigma de la programación orientada a objetos, como son la encapsulación, la herencia y el polimorfismo.

**Java**: es un lenguaje orientado a objetos, eso implica que su concepción es muy próxima a la forma de pensar humana, También posee otras características muy importantes:

* Es un lenguaje que es compilado, generando ficheros de clases compilados, pero estas clases compiladas son en realidad interpretadas por la máquina virtual java. Siendo la máquina virtual de java la que mantiene el control sobre las clases que se estén ejecutando.
* Es un lenguaje multiplataforma: El mismo código java que funciona en un sistema operativo funcionará en cualquier otro sistema operativo que tenga instalada la máquina virtual java.
* Es un lenguaje seguro: La máquina virtual al ejecutar el código java realiza comprobaciones de seguridad, además el propio lenguaje carece de características inseguras, como por ejemplo los punteros.
* Gracias al API de java podemos ampliar el lenguaje para que sea capaz de, por ejemplo, comunicarse con equipos mediante red, acceder a bases de datos, crear páginas HTML dinámicas, crear aplicaciones visuales al estilo Windows,

**ASP.net**: Es una tecnología dinámica funcionando del lado del servidor, lo que significa que cuando el usuario solicita un documento ASP, las instrucciones de programación dentro del script son ejecutadas para enviar al navegador únicamente el código HTML resultante. La ventaja principal de las tecnologías dependientes del servidor radica en la seguridad que tiene el programador sobre su código, ya que éste se encuentra únicamente en los archivos del servidor que al ser solicitado a través del web, es ejecutado, por lo que los usuario no tienen acceso más que a la página resultante en su navegador.

# Propuesta de solución recomendada

Para esta solución nosotros hemos decidido a usar el lenguaje **C#** con la herramienta Visual Studio, ya que al ser una aplicación estática es más factible hacerla en este lenguaje ya que tanto Java como ASP.net están más destinadas a la utilización en sistemas WEB destinadas a la utilización en sistemas WEB

# Planificación

|  |  |  |
| --- | --- | --- |
| **Tarea** | **Días** | **Fechas** |
| Toma de requerimientos | 10 | 25 al 06 de Mayo |
| Análisis de requerimientos | 13 | 9 al 25 de mayo |
| Diseño | 7 | 30 de mayo al 7 de junio |
| Codificación | 20 | 8 de junio al 5 de julio |
| QA | 6 | 6 de julio al 13 de julio |
| Liberación | 3 | 14 de julio al 18 de julio |
|  |  |  |

* **Toma de requerimientos**

Etapa de suma importancia dentro del desarrollo de cualquier Software, Este se preocupa de descubrir y analizar las necesidades del usuario.

* **Análisis de Requerimientos**

Etapa que comprende todas las tareas o condiciones para la ejecución del desarrollo del Software, Tomando en cuenta las ideas de las partes interesadas.

* **Diseño.**

Es la creación de los componentes del software, según los requerimientos dados por el cliente.

* **Codificación**

Es el ingreso de códigos en el software, para que este envié y reciba datos a través de nuestro diseño y así poder tener resultados a partir de ellos.

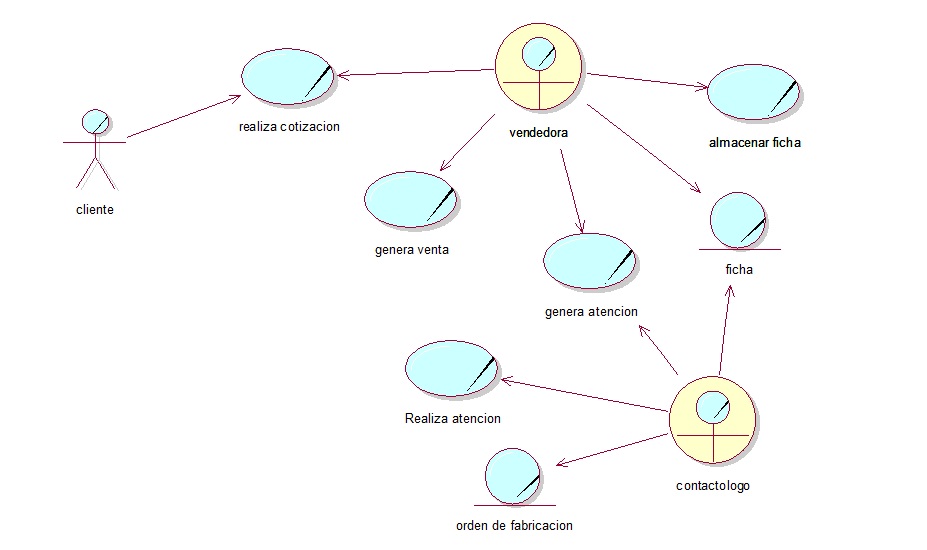
* **QA (**Quality Assurance).

Etapa donde se realizan todas las pruebas de funcionamiento de nuestro Software, es acá, en donde se visualizan todos los posibles errores y también si se están cumpliendo las solicitudes de los clientes a través de los requerimientos mencionados por él.

* **Liberación**

Es la entrega del software a o los clientes terminado en las instalaciones de este. Y funcionando perfectamente. Es acá en donde el cliente nos libera del proyecto mediante una carta que indique que el software solicitado por él está perfecto y hecho según sus requerimientos.

# Modelo de negocios



|  |  |
| --- | --- |
| **Actores** | **Características** |
| cliente (externo) | es la persona que se fabricara el lente y con los  datos de esta se poblara la ficha |
| Vendedora (interno) | es la encargada de generar la venta e ingresar los  primeros datos a la ficha y deriva la atención a contactología |
| contactólogo (interno) | es la persona que ingresa los datos restantes de la  ficha y genera la atención de contactología |
|  |  |
|  |  |
| **Casos de uso del negocio** | **Características** |
| realizar cotización | es donde participa el cliente y la vendedora y de  indican precios y tiempos |
| generar venta | luego de realizada la cotización y ser aceptada la  vendedora genera la venta, también se llenan los primeros datos de la ficha, además debe consultar si el lente fuese desechable si este tiene stock |
| generar atención | se deriva la atención a contactología |
| realizar atención | contactólogo realiza atención y termina de llenar  la ficha |
| almacenar ficha | una vez terminada la atención la vendedora  almacena la ficha en una carpeta |
|  |  |
| **Entidades** | **Características** |
| ficha | es llenada por el contactólogo y la vendedora |
| orden de fabricación | es emitida por el contactólogo después de la  atención |

# Definición de requerimiento

Los requerimientos nos especifican lo que el sistema debe hacer (sus funciones) de manera objetiva y precisa

además sus propiedades deben ser esenciales y deseables.

La captura de los requerimientos tiene como objetivo principal cumplir con las expectativas del cliente y usuario

y principalmente que posee todas las funcionalidades que se especificaron al inicio del proyecto-

En esta etapa se detallaran los principales requerimientos del sistema en los cuales estan:

Requerimientos funcionales

Requerimientos no funcionales

Requerimientos de dominio

## 10.1 Requerimientos funcionales

1° Generar Venta

2° Crear Ficha

3° Administrar Ficha

4° Modificar Valores

5° Generar Orden de Trabajo

7° Generar Reportes

8° Consultar Stock

9° Generar listado con todos los clientes registrados

10°Generar listado de ventas de un cliente especifico

11°Permitir búsqueda de clientes

12°Permitir eliminación de productos

13°Permitir que cada venta, se le deberá asignar un identificador único

## 10.2 Requerimientos No funcionales

Eficiencia.

1° El programa debe ser capaz de procesar cierta cantidad de transacciones

2° El programa debe ser capaz de operar adecuadamente con varios usuarios a la vez operando

3° Los datos modificados en la BD deben ser actualizados a todos los usuarios que acceden

4° Reuniones con personal del área para revisión de datos generados

5° Entrevistas con el usuario directo del sistema, para la aprobación de mapas de navegación.

Seguridad de datos

1° Los permisos de acceso al sistema podrán ser cambiados solamente por el adm de acceso de satos

2° Todos los sistemas deben respaldarse cada 24 hrs.

3° Si se identifican ataques de seguridad o brecha del sistema, el mismo no continuará operando hasta

ser desbloqueado por un administrador de seguridad.

Usabilidad.

1°El tiempo de aprendizaje del sistema por un usuario deberá ser menor a 1 día.

2°El sistema debe contar con manuales de usuario estructurados adecuadamente.

3°El sistema debe proporcionar mensajes de error que sean informativos y orientados a usuario final.

4°El sistema debe contar con un módulo de ayuda en línea o telefónica.

5°El sistema debe poseer interfaces gráficas bien formadas.

## 10.3 Requerimientos de Dominio

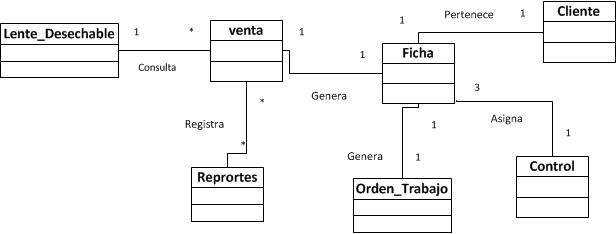
1° programa debe tener todos los manuales y documentación necesaria para ser aprobada por iso 9001.

2° Deberá existir interfaz de usuario estándar para todas las BD

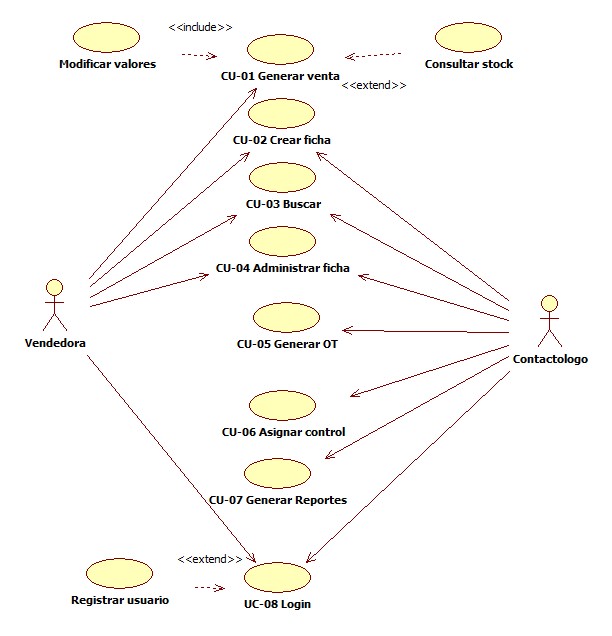
3° restricciones a derecho de autor.

4° No duplicidad del proyecto.

# Modelo de clases conceptuales



# Casos de uso



# Descripción de los casos de uso

## 13.1.1 Uc-01 Generar venta



|  |  |
| --- | --- |
| Datos Generales | |
| Nombre | Vendedora |
| Descripción | Rol Generar la venta a un cliente |
| Versión | 1.0 |

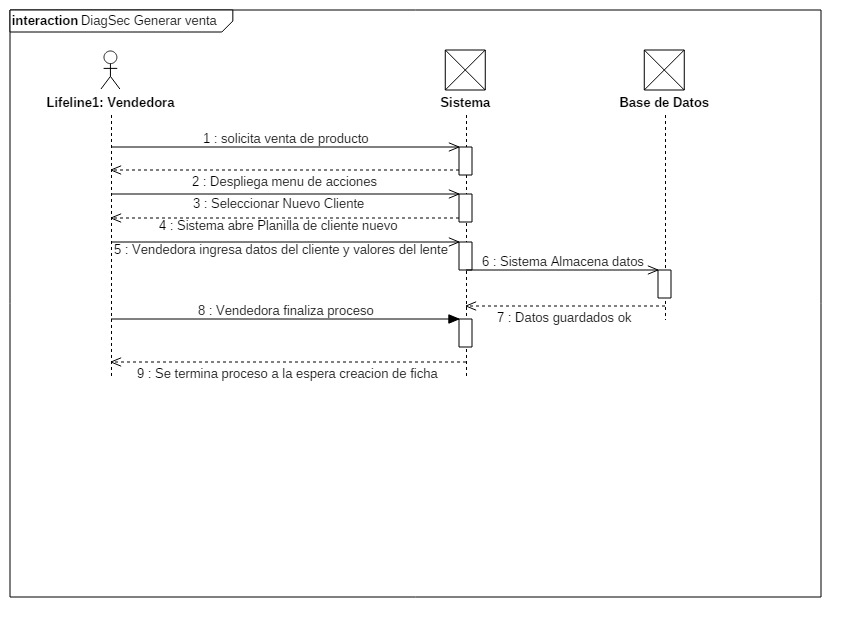
|  |  |
| --- | --- |
| Datos Generales del Caso de Uso | |
| ID Caso de Uso: | Uc-01 |
| Caso de Uso: | Generar venta |
| Actores: | Vendedora |
| Propósito: | Genera una venta, luego de aceptada esta, en caso de que fuese lente desechable se consulta si se encuentra en stock |
| Resumen: | luego de realizada la cotización y ser aceptada la  vendedora genera la venta, también se llenan los primeros datos de la ficha, además debe consultar si el lente fuese desechable si este tiene stock, además permite modificar tanto los datos como los pagos del lente |
| Tipo: | Primario |
| Referencias Cruzadas: | CU - 08 |

|  |  |
| --- | --- |
| **Escenario Exitoso** | |
| Responsabilidad del Actor | Responsabilidad del Sistema |
| 1.- actor abre el sistema | 2.- sistema despliega menú de acciones |
| 3.- se selecciona nuevo cliente | 4.- sistema abre pantalla de nuevo cliente con los valores de lente a modificar y datos de identificación |
| 5.- vendedora ingresa valor de lente, además de abono y saldo si este no fuese cancelado completo, además se ingresa nombre y rut del paciente, | 6. sistema almacena los datos |
| 7.- vendedora finaliza proceso | 8.- se termina el proceso a la espera de creación de ficha |

|  |  |
| --- | --- |
| **Caso Alternativo** | |
| Responsabilidad Actor | Responsabilidad Sistema |
| Paso 6 almacenamiento de datos | |
|  | * 1. Valores ingresados no corresponden |
| Se regresa al paso 5 | |

|  |  |
| --- | --- |
| **Caso Alternativo** | |
| Responsabilidad Actor | Responsabilidad Sistema |
| Paso 6 almacenamiento de datos | |
|  | a.- el paciente ya se encuentra ingresado, se despliega un mensaje de alerta indicando dicha situación |
| Se regresa al paso 2 y sistema queda en espera | |

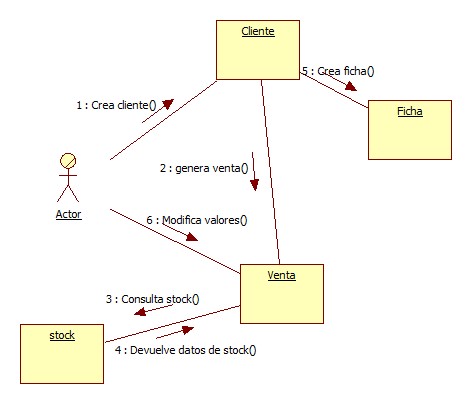
### 13.1.2 Diagrama de secuencia Generar venta



### 13.1.3 Contrato de Operaciones “Generar Venta”.

|  |  |
| --- | --- |
| **Operación** | **Genera Venta.** |
| Referencia Cruzada | Caso de Uso: Genera Venta. |
| Descripción | Usuario abre sistema ingresando Login y password, para ingresar |
| Pre-Condiciones | 1. Sistema debe estar Operativo. 2. Usuario abre sistema con login y password otorgados por el administrador del sistema (Contactólogo). |
| Post-Condiciones | 1. Se crea instancia de la clase cliente, ficha, venta, stock, inventario de ventas, usuario. 2. Se asocia cliente con ficha. 3. Se asocia cliente con venta. 4. Se asocia clase venta con stock. 5. Se modifican atributos de la clase ventas. 6. Se modifican atributos de la clase fichas. 7. Se consultan atributos de la clase stock. 8. Se destruyen asociaciones. |

### 13.1.3 Diagrama de colaboración Generar venta



## 13.2.1 Uc-02 Crear ficha



Actor

|  |  |
| --- | --- |
| Nombre | Contactólogo |
| Descripción | Rol Creación de ficha |
| Versión | 1.0 |

|  |  |
| --- | --- |
| Datos Generales del Caso de Uso | |
| ID Caso de Uso: | Uc-02 |
| Caso de Uso: | Crear ficha |
| Actores: | Contactólogo |
| Propósito: | Luego de realizada la venta se crea ficha, con los datos de queratometrias |
| Resumen: | Una vez realizada la venta, el contactólogo procederá a crear ficha con los datos de queratometrias y dioptrías de lente de contacto |
| Tipo: | Primario |
| Referencias Cruzadas: | CU-01 |

|  |  |
| --- | --- |
| **Escenario Exitoso** | |
| Responsabilidad del Actor | Responsabilidad del Sistema |
| 1.- actor abre el sistema | 2.- sistema despliega menú de acciones |
| 3.- se selecciona cliente | 4.- sistema abre pantalla de cliente |
| 5.- selecciona opción ficha | 6. sistema abre formulario de ficha con los datos a ser llenados |
| 7.- contactólogo presiona opción guardar | 8.- almacena los datos |
| 9.- finaliza el proceso | 8.- sistema cierra el proceso y vuelve a pantalla de inicio |

|  |  |
| --- | --- |
| **Caso Alternativo** | |
| Responsabilidad Actor | Responsabilidad Sistema |
| Paso 4 abre ventana de cliente | |
|  | * 1. Si cliente ya posee ficha, opción de crear ficha no estará disponible |
| Se salta al paso 6 | |

|  |  |
| --- | --- |
| **Caso Alternativo** | |
| Responsabilidad Actor | Responsabilidad Sistema |
| Paso 8 almacenamiento de datos | |
|  | * 1. Valores ingresados no corresponden |
| Se regresa al paso 6 | |

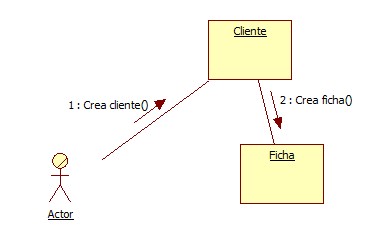
### 13.2.2 Diagrama de Secuencian Crear ficha

### 

### 13.2.3 Contrato de Operaciones “Crear Ficha”.

|  |  |
| --- | --- |
| **Operación** | **Crea Ficha.** |
| Referencias Cruzadas | Caso de Uso: Crear Ficha. |
| Descripción | Una vez realizada la venta, el contactólogo procederá a crear ficha con los datos de queratometrias y dioptrías de lente de contacto |
| Pre-Condiciones | 1. Sistema debe estar operativo. 2. Contactólogo abre sistema con Login y pasword de administrador. 3. Selecciona menú de acciones. 4. Elige opción clientes. |
| Post-Condiciones | 1. Se crea instancia de la clase usuario cliente, ficha. 2. Se asocia usuario con cliente. 3. Se asocia cliente con ficha. 4. Se modifican atributos de la clase cliente. 5. Se modifican atributos de la clase ficha. 6. Se destruyen asociaciones. |

### 13.2.4 Diagrama de Colaboración Crear ficha



## 13.3.3 Uc-03 Buscar



|  |  |
| --- | --- |
| Nombre | Contactólogo |
| Descripción | Rol Buscar ficha |
| Versión | 1.0 |

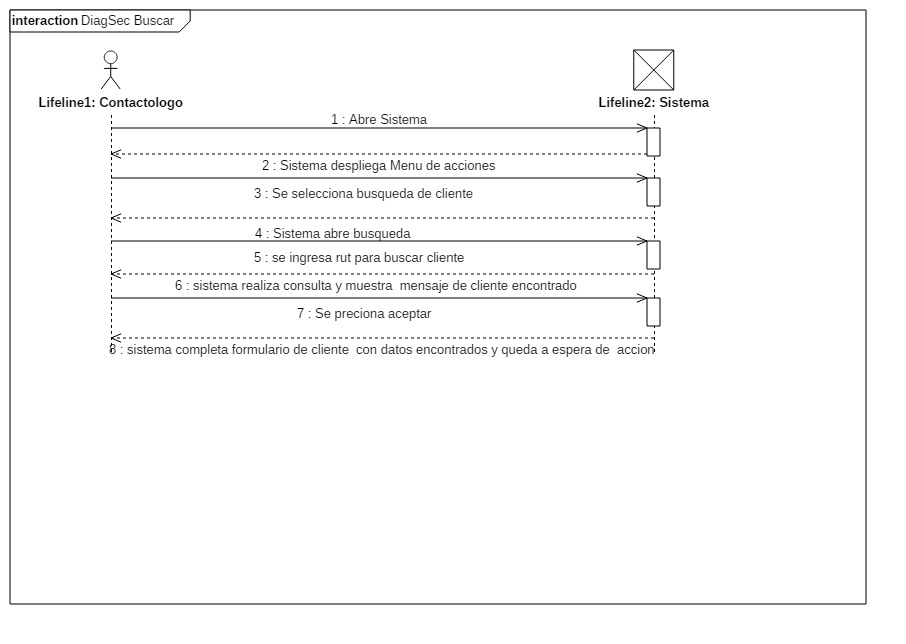
|  |  |
| --- | --- |
| Nombre | Vendedora |
| Descripción | Rol Creación de ficha |
| Versión | 1.0 |

|  |  |
| --- | --- |
| Datos Generales del Caso de Uso | |
| ID Caso de Uso: | Uc-03 |
| Caso de Uso: | Buscar |
| Actores: | Contactólogo, Vendedora |
| Propósito: | Buscar fichas de pacientes ingresados al sistema |
| Resumen: | Busca a los pacientes ya ingresados al sistema y los muestra en pantalla cliente |
| Tipo: | Secundario |
| Referencias Cruzadas: | CU-01,CU-02 |

|  |  |
| --- | --- |
| **Escenario Exitoso** | |
| Responsabilidad del Actor | Responsabilidad del Sistema |
| 1.- actor abre el sistema | 2.- sistema despliega menú de acciones |
| 3.- se selecciona búsqueda cliente | 4.- sistema abre búsqueda |
| 5.- actor en campo Rut ingresa el del cliente | 6. sistema realiza consulta y muestra mensaje de cliente encontrado |
| 7.- se presiona aceptar | 8.- sistema completa formulario de cliente con datos encontrados y queda a espera de accion |

|  |  |
| --- | --- |
| **Caso Alternativo** | |
| Responsabilidad Actor | Responsabilidad Sistema |
| Paso 6 sistema realiza consulta y muestra mensaje de cliente encontrado | |
|  | * 1. Si cliente no es encontrado, sistema muestra mensaje “cliente no encontrado” |
| Se regresa al paso 4 | |

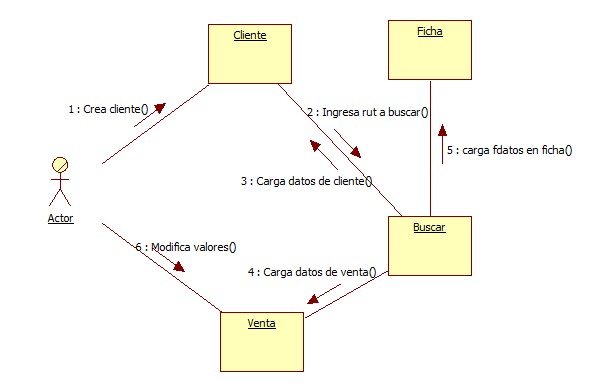
### 13.3.2 Diagrama de Secuencia Buscar ficha



### 13.3.3 Contrato de Operaciones “Buscar”.

|  |  |
| --- | --- |
| **Operación** | **Busca.** |
| Referencias Cruzadas | Caso de Uso: Buscar. |
| Descripción | Busca a los pacientes ya ingresados al sistema y los muestra en pantalla cliente. |
| Pre-Condiciones | 1. Sistema debe estar operativo. 2. Contactólogo o Vendedora deben abrir sistema con login y password correspondientes a usuario de sistema. 3. Selecciona menú de acciones. 4. Elige opción buscar. |
| Post-Condiciones | 1. Se crea instancia de la clase usuario, cliente, buscar, ficha, venta. 2. Se asocia usuario con cliente. 3. Se asocia cliente con buscar. 4. Se asocia buscar con ficha. 5. Se asocia clase buscar con venta. 6. Se consultan atributos de la clase ventas. 7. Se consultan atributos de la clase ficha. 8. Se destruyen asociaciones. |

### 13.3.4 Diagrama de colaboración Buscar ficha



## 13.4.1 Uc-04 Modificar ficha



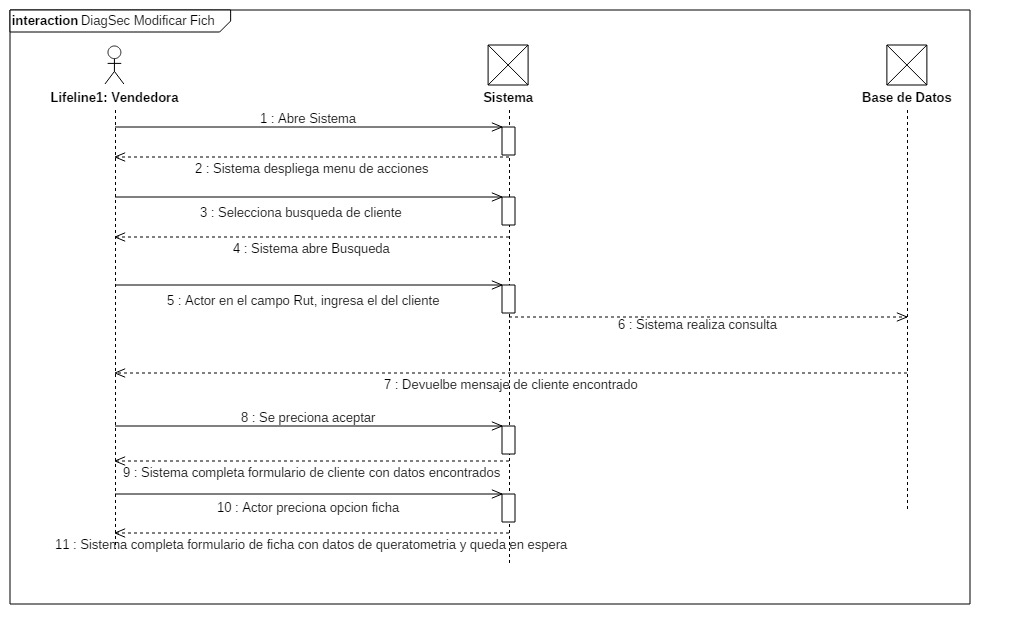
|  |  |
| --- | --- |
| Nombre | Contactólogo |
| Descripción | Rol Administrar ficha |
| Versión | 1.0 |

|  |  |
| --- | --- |
| Datos Generales del Caso de Uso | |
| ID Caso de Uso: | Uc-04 |
| Caso de Uso: | Administrar ficha |
| Actores: | Contactólogo |
| Propósito: | Modificar datos de ficha guardada con anterioridad |
| Resumen: | Modifica una ficha guardada. |
| Tipo: | Secundario |
| Referencias Cruzadas: | CU-02,CU-03 |

|  |  |
| --- | --- |
| **Escenario Exitoso** | |
| Responsabilidad del Actor | Responsabilidad del Sistema |
| 1.- actor abre el sistema | 2.- sistema despliega menú de acciones |
| 3.- se selecciona búsqueda cliente | 4.- sistema abre búsqueda |
| 5.- actor en campo Rut ingresa el del cliente | 6. sistema realiza consulta y muestra mensaje de cliente encontrado |
| 7.- se presiona aceptar | 8.- sistema completa formulario de cliente con datos encontrados |
| 9.- actor presiona opción ficha | 8.- sistema completa formulario de ficha con datos de queratometrias y queda en espera |

|  |  |
| --- | --- |
| **Caso Alternativo** | |
| Responsabilidad Actor | Responsabilidad Sistema |
| Paso 6 sistema realiza consulta y muestra mensaje de cliente encontrado | |
|  | * 1. Si cliente no es encontrado, sistema muestra mensaje “cliente no encontrado” |
| Se regresa al paso 4 | |

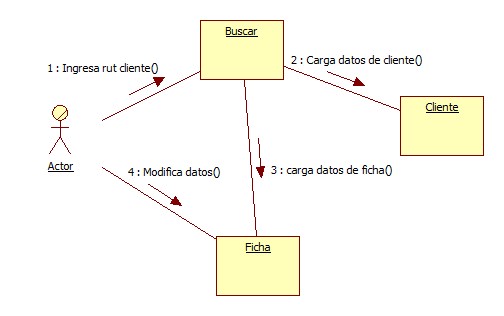
### 13.4.2 Diagrama de secuencia Administrar ficha



### 13.4.4 Contrato de Operaciones “Administrar Ficha”.

|  |  |
| --- | --- |
| **Operación** | **Modifica Ficha.** |
| Referencias Cruzadas | Caso de Uso: Modificar Ficha. |
| Descripción | Modifica una ficha guardada, tanto los datos como los pagos del lente. |
| Pre-Condiciones | 1. Sistema debe estar operativo. 2. Contactólogo o Vendedora deben abrir sistema con login y password correspondientes a usuario de sistema. 3. Selecciona menú de acciones. 4. Elige opción. |
| Post-Condiciones | 1. Se crea instancia de la clase usuario, buscar, cliente, ficha. 2. Se asocia usuario con buscar. 3. Se asocia buscar con cliente. 4. Se asocia buscar con ficha. 5. Se asocia usuario con ficha. 6. Se consultan atributos de la clase buscar. 7. Se consultan atributos de la clase cliente. 8. Se consultan atributos de la clase ficha. 9. Se destruyen asociaciones cliente cada vez que asista a la óptica. 10. Se deben guardar datos modificados en base de datos. 11. Se destruyen asociaciones. |

### 13.4.3 Diagrama de colaboración administrar ficha



## 13.5.1 Uc-05 Generar OT



|  |  |
| --- | --- |
| Nombre | Contactólogo |
| Descripción | Rol Generar OT |
| Versión | 1.0 |

|  |  |
| --- | --- |
| Datos Generales del Caso de Uso | |
| ID Caso de Uso: | Uc-05 |
| Caso de Uso: | Generar OT |
| Actores: | Contactólogo |
| Propósito: | Generar orden de trabajo |
| Resumen: | Genera una orden de trabajo con datos de fabricación de lente y fecha de entrega |
| Tipo: | Secundario |
| Referencias Cruzadas: | CU-01,CU-02,CU-04 |

|  |  |
| --- | --- |
| Responsabilidad del Actor | Responsabilidad del Sistema |
|  | 1.- sistema se encuentra en espera luego de crear ficha |
| 2.- actor presiona opción Orden de trabajo | 3.- sistema abre formulario de orden de trabajo |
| 4.- actor llena campos con datos requeridos | 5.- sistema almacena datos |
| 6.- se presiona aceptar | 7.- sistema imprime OT con datos necesarios para fabricación, asignando misma OT que generada en venta |
| 8.- actor presiona aceptar | 9.- sistema vuelve al formulario ficha y queda a espera |

|  |  |
| --- | --- |
| **Caso Alternativo** | |
| Responsabilidad Actor | Responsabilidad Sistema |
| Paso 5 sistema almacena datos | |
|  | * 1. Valores ingresados no corresponden |
| Se regresa al paso 3 | |

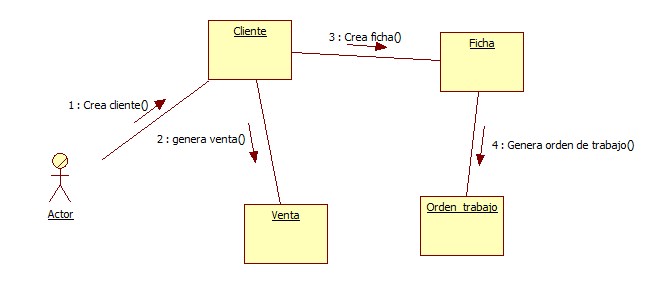
### **13.5.2 Diagrama de secuencia Generar OT**

### 

### 13.5.3 Contrato de Operaciones “Generar Orden de Trabajo”.

|  |  |
| --- | --- |
| Operación | **Orden de Trabajo.** |
| Referencias Cruzadas | Caso de Uso: Generar Orden de trabajo. |
| Descripción | Genera una orden de trabajo con datos de fabricación de lente y fecha de entrega. |
| Pre-Condiciones | 1. Sistema debe estar operativo. 2. Contactólogo abre sistema con login y password 3. Selecciona menú de acciones. 4. Elige opción. Orden de trabajo. 5. Se ingresa número de orden de trabajo asociado a una ID\_Cliente. |
| Post-Condiciones | 1. Se crea instancia de la clase usuario, cliente, ficha, orden de trabajo, venta. 2. Se asocia usuario con cliente. 3. Se asocia cliente con venta. 4. Se asocia cliente con ficha. 5. Se asocia ficha con orden de trabajo. 6. Se consultan atributos de la clase cliente. 7. Se consultan atributos de la clase venta. 8. Se consultan atributos de la clase ficha. 9. Se consultan atributos de la clase orden de trabajo. 10. Se modifican atributos de la orden de trabajo. 11. Se modifican atributos de ficha. 12. Se modifican atributos de venta. 13. Se destruyen asociaciones. |

### 13.5.3 Diagrama de colaboración Generar OT



## 13.6 .1 Asignar control



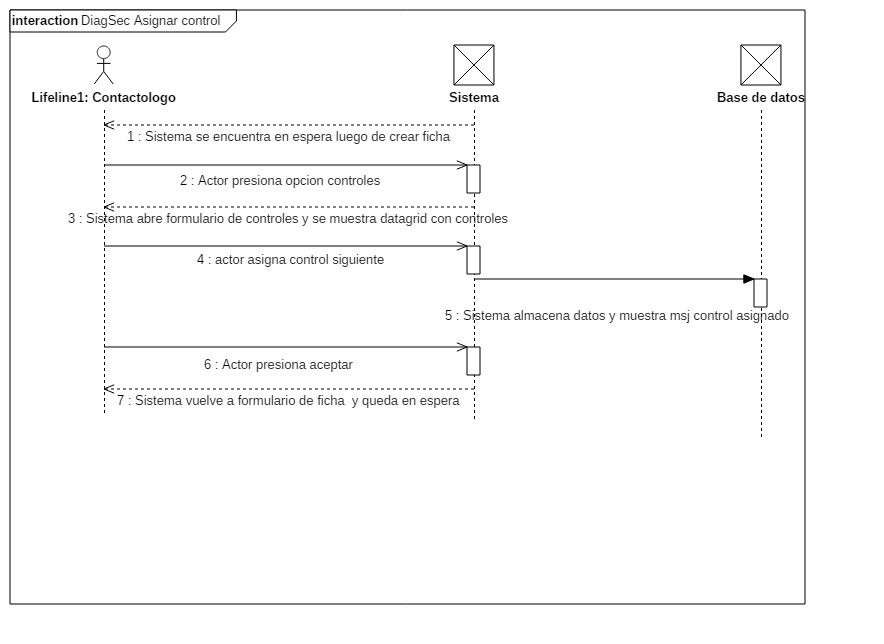
|  |  |
| --- | --- |
| Nombre | Contactólogo |
| Descripción | Rol Asignar control |
| Versión | 1.0 |

|  |  |
| --- | --- |
| Datos Generales del Caso de Uso | |
| ID Caso de Uso: | Uc-06 |
| Caso de Uso: | Asignar control |
| Actores: | Contactólogo |
| Propósito: | Asignar futuros controles a pacientes |
| Resumen: | Asigna futuros controles para ver avances en adaptación de lentes de contacto |
| Tipo: | Secundario |
| Referencias Cruzadas: | CU-02, CU-03, CU-04 |

|  |  |
| --- | --- |
| Responsabilidad del Actor | Responsabilidad del Sistema |
|  | 1.- sistema se encuentra en espera luego de crear ficha |
| 2.- actor presiona opción controles | 3.- sistema abre formulario de controles y se muestra DataGrid con controles |
| 4.- actor asigna control siguiente | 5. sistema almacena datos y muestra mensaje “control asignado” |
| 6.- se presiona aceptar | 7.- Sistema vuelve a formulario de ficha y queda en espera |

|  |  |
| --- | --- |
| **Caso Alternativo** | |
| Responsabilidad Actor | Responsabilidad Sistema |
| Paso 4.- actor asigna control siguiente | |
|  | * 1. Si se asigna tercer control, se mostrara mensaje “ultimo control” y al marcar control como efectuado se deshabilitara opción asignar |
| Se salta al paso 7 | |

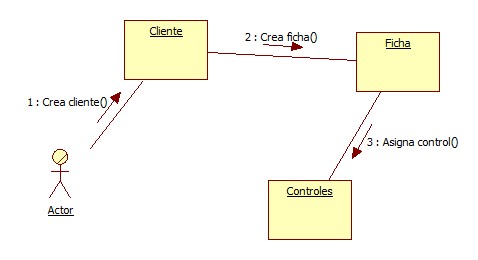
### 13.6.2 Diagrama de secuencia Asignar control.



### Contrato de Operaciones “Asignar Control”.

|  |  |
| --- | --- |
| **Operación** | **Control** |
| Referencias Cruzadas | Caso de Uso: Asignar Control |
| Descripción | Asigna futuros controles para ver avances en adaptación de lentes de contacto |
| Pre-Condiciones | 1. Sistema debe estar operativo. 2. Contactólogo abre sistema con login y password 3. Selecciona menú de acciones. 4. Elige opción. Controles. |
| Post-Condiciones | 1. Se crea instancia de la clase usuario, cliente, ficha, control. 2. Se asocia usuario con cliente. 3. Se asocia cliente con ficha. 4. Se asocia ficha con controles. 5. Se consultan atributos de la clase cliente. 6. Se consultan atributos de la clase ficha. 7. Se consultan atributos de la clase control. 8. Se modifican atributos de la clase ficha. 9. Se modifican atributos de la clase control. 10. Se destruyen asociaciones. |

### 13.6.4 Diagrama de colaboración Asignar control



## 13.7.1 Uc-07 Generar Reportes

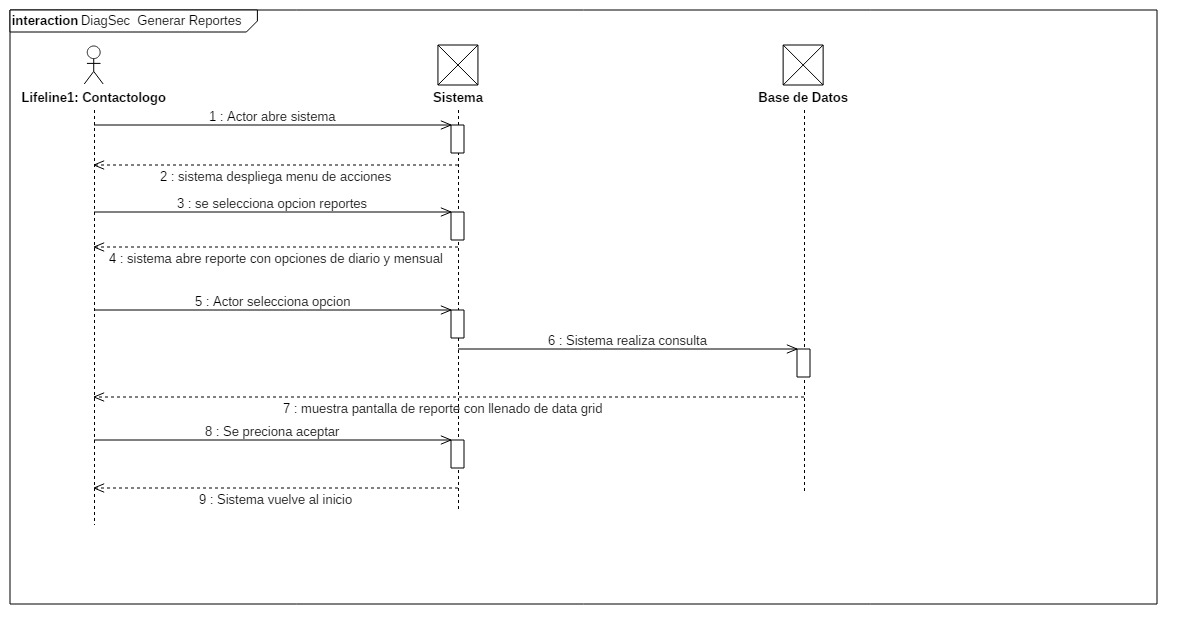


|  |  |
| --- | --- |
| Nombre | Contactólogo |
| Descripción | Rol Generar Reportes |
| Versión | 1.0 |

|  |  |
| --- | --- |
| Datos Generales del Caso de Uso | |
| ID Caso de Uso: | Uc-07 |
| Caso de Uso: | Generar reportes |
| Actores: | Contactólogo |
| Propósito: | Genera reportes sobre las ventas |
| Resumen: | Genera reportes sobre cantidad de lentes vendido y sus valores |
| Tipo: | Secundario |
| Referencias Cruzadas: | CU-1 |

|  |  |
| --- | --- |
| **Escenario Exitoso** | |
| Responsabilidad del Actor | Responsabilidad del Sistema |
| 1.- actor abre el sistema | 2.- sistema despliega menú de acciones |
| 3.- se selecciona reportes | 4.- sistema abre Reportes, con opciones:  -Reporte diario  -Reporte mensual  - |
| 5.- actor selecciona una opción | 6. sistema realiza consulta y muestra pantalla de reporte, llenando DataGrid con datos seleccionados |
| 7.- se presiona aceptar | 8.- sistema vuelve inicio |

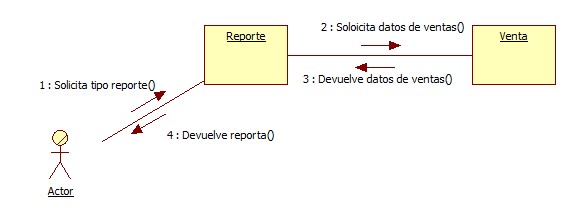
### 13.7.2 Diagrama de secuencia Generar reporte



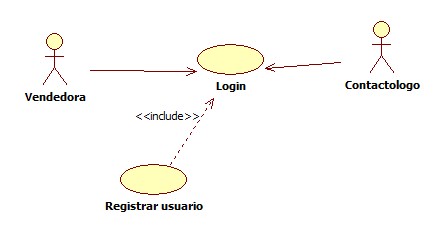
### 13.7.3 Contrato de Operaciones “Generar Reporte”.

|  |  |
| --- | --- |
| **Operación** | **Reportes.** |
| Referencias Cruzadas | Caso de Uso: Generar Reporte. |
| Descripción | Genera reportes sobre cantidad de lentes vendidos y sus valores.  Estos reportes se pueden solicitar diarios como también mensuales. |
| Pre-Condiciones | 1. Sistema debe estar operativo. 2. Contactólogo abre sistema con login y password 3. Selecciona menú de acciones. 4. Elige opción. Generar Reporte. 5. Elige opción, Diario o Mensual. |
| Post-Condiciones | 1. Se crea instancia de la clase usuario, reporte, venta 2. Se asocia usuario con reporte. 3. Se asocia reporte con venta.. 4. Se consultan atributos de la clase venta. 5. Se destruyen asociaciones |

### 13.7.4 Diagrama de colaboración Generar reporte



## 13.9.1 Uc-07 Login



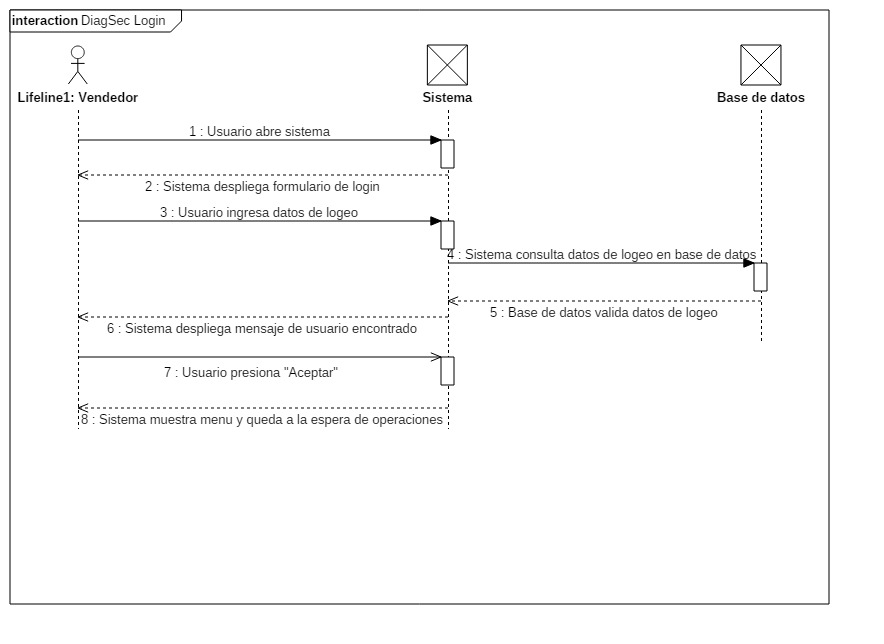
|  |  |
| --- | --- |
| Nombre | Contactólogo, Vendedora |
| Descripción | Rol Leguearse |
| Versión | 1.0 |

|  |  |
| --- | --- |
| Datos Generales del Caso de Uso | |
| ID Caso de Uso: | Uc-08 |
| Caso de Uso: | Login |
| Actores: | Contactólogo, Vendedora |
| Propósito: | Permite ingresar al sistema |
| Resumen: | Permite a los usuarios ingresar al sistema, y tener acceso según sus permisos, además de registrase en caso de no estar en este |
| Tipo: | Primario |
| Referencias Cruzadas: | N/A |

|  |  |
| --- | --- |
| Responsabilidad del Actor | Responsabilidad del Sistema |
| 1.- Usuario abre el sistema | 2.- Sistema despliega formulario de login |
| 3.- usuario ingresa su nombre y pass y presiona aceptar | 3.- sistema realiza consulta y valida datos registrados, despliega mensaje de “usuario encontrado” |
| 4.- usuario presiona aceptar | 5. sistema muestra pantalla de menú y queda a la espera |

|  |  |
| --- | --- |
| **Caso Alternativo** | |
| Responsabilidad Actor | Responsabilidad Sistema |
| 3.- sistema realiza consulta y valida datos registrados, despliega mensaje de “usuario encontrado” | |
|  | . - Si usuario no se encuentra registrado, sistema mostrara mensaje de “usuario no encontrado” |
| Se retorna al paso 2 | |

### 13.9.2 Diagrama de secuencia login



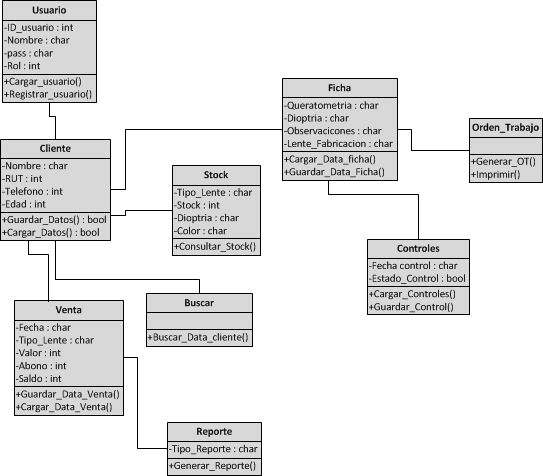
### 13.9.3 Contrato de Operaciones “Login”.

|  |  |
| --- | --- |
| **Operacion** | **Ingreso al sistema.** |
| Referencias Cruzadas | Caso de Uso “Login “. |
| Descripción | Permite a los usuarios ingresar al sistema, y tener acceso según sus permisos, además de registrase en caso de no estar en este. |
| Pre-Condicion | 1. Sistema debe estar operativo. 2. Usuario abre sistema. 3. Usuario debe contar con login y password. 4. Usuario ingresa login y password. 5. Sistema abre menú de acciones. |
| Post-Condiciones | 1. Se crea instancia de la clase login, ususario 2. Se asocia login con usuario. 3. Se modifican atributos de la clase login. 4. Se modifican atributos de a clase usuario. 5. Se destruyen acciones. |

### 13.9.9 Contrato de Operaciones “Login”.

****

# Diagrama de clases



# Modelo Relacional

