

UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA

INSTITUTO DE INGENIERÍA

MAESTRÍA EN CIENCIAS EN INGENIERÍA DE SISTEMAS



**“DOCUMENTACIÓN ELECTRÓNICA DE UN SISTEMA DE
GESTIÓN DE CALIDAD. ESTUDIO DE CASO.”**

TESIS

PARA OBTENER EL GRADO DE MAESTRO EN CIENCIAS EN
INGENIERÍA DE SISTEMAS

PRESENTA

NORMA MIER LUNA

DIRECTOR

M.C. MOISÉS RIVAS LÓPEZ

Dedicatoria

A mis padres:

Manuel(†) y Amalia.

A mi familia:

Oscar Manuel, Leslie,
Aniana y Oscarito.

Agradecimientos

A mi director de tesis M.C. Moisés Rivas López por su tiempo para la conclusión de esta investigación.

A el coordinador de la Maestría Dra. Sara Ojeda Benítez por su disposición.

A M.I. Moisés Galindo Duarte por su ayuda proporcionada durante su etapa de coordinador de la Maestría.

A mis maestros de posgrado por hacer posible la formación de investigadores, especialmente a Brenda y Fernando.

A los miembros del comité por su interés.

Al Centro de Estudios Superiores del Estado de Sonora por el apoyo en el estudio de la presente maestría.

Al Consejo Nacional de Ciencia y Tecnología por hacer posible mejorar el nivel académico de los profesionistas.

Al Programa de Mejoramiento del Profesorado de Educación Superior por fortalecer académicamente a los profesores.

ÍNDICE

	Pág.
CAPÍTULO I. INTRODUCCIÓN.	1
1.1 Sistema de Gestión de Calidad (SGC).	2
1.1.1 Familia ISO 9000.	3
1.1.2 ISO 9001:2000.	6
1.2 Diseño de Software Orientado a Objetos (OO).	8
1.2.1 Lenguaje Unificado de Modelado (UML).	10
1.3 Problemática.	11
1.3.1 Problemas de los Sistemas de Gestión de Calidad (SGC).	12
1.4 Justificación.	14
1.5 Objetivo General y Objetivos Específicos.	16
1.6 Estructura del documento.	16
 CAPÍTULO II. MARCO TEÓRICO.	 18
2.1 Certificación.	18
2.1.1 Normas de certificación.	18
2.1.2 Organismos de certificación.	20
2.2 Instituciones de Educación Superior y Sistemas de Gestión de Calidad.	21
2.3 Modelado.	26
2.3.1 Modelado de sistemas, procesos y datos.	29
2.3.2 Diagramas y Herramientas para el Diseño de Software con UML.	30
 CAPÍTULO III. METODOLOGÍA.	 38
3.1 Método.	38
3.2 Sujetos.	38
3.3 Materiales.	38
3.4 Procedimiento.	39

CAPÍTULO IV. ESTUDIO DE CASO: SISTEMA DE GESTIÓN DE CALIDAD DEL INSTITUTO DE INGENIERÍA, UABC.	40
4.1 Descripción.	40
4.1.1 Estructura Organizacional y descripción general de puestos.	41
4.1.2 Descripción de las entrevistas de los roles.	42
4.2 Análisis del Sistema de Gestión de Calidad.	46
 CAPÍTULO V. RESULTADOS OBTENIDOS.	 51
5.1 Modelos obtenidos.	51
5.1.1 Modelos del SGC del II.	51
5.1.2 Modelos de requisitos del SI.	64
5.1.2.1 Diagramas de casos de uso.	64
5.1.2.2 Diagramas de secuencia.	72
5.1.3 Modelo conceptual para el SI.	97
5.1.4 Modelo de emplazamiento.	100
5.2 Conclusiones y recomendaciones.	102
 BIBLIOGRAFIA Y REFERENCIAS.	 104
 ANEXOS.	
A. Historia de UML.	106
B. Diagramas de UML.	110
C. Diseño de Entrevista semiestructurada.	116

ÍNDICE DE TABLAS

Núm.	Nombre	Pág.
1	Familia actual del ISO 9000.	6
2	Matriz de Responsabilidades.	46
3	Contenido del Manual de Calidad.	48
4	Procedimientos Escritos del SGC del II.	49
5	Relación de diagramas de actividades de procesos viables para automatizar.	54

ÍNDICE DE FIGURAS

Núm.	Nombre	Pág.
1	Mejora continua del Sistema de Gestión de Calidad.	2
2	Países con mayor número de empresas certificadas.	4
3	Ubicación de la Documentación Electrónica dentro de un SGC.	13
4	Estructura Organizacional del SGC del II.	41
5	Enfoque sistémico del SGC del II.	47
6	Elementos de la Documentación.	48
7	Diagrama de Interacción de Procesos del laboratorio de Metrología.	50
8	Diagrama de caso de uso 1 del proceso de calibración en el Laboratorio de Metrología.	52
9	Diagrama de caso de uso 2 del proceso de calibración en el Laboratorio de Metrología.	53
10	Diagrama de Actividades para el proceso de Alta de Documentos.	55
11	Diagrama de actividades para el proceso Baja de Documentos.	56
12	Diagrama de actividades para el proceso de Cambio de Documentos.	57
13	Diagrama de actividades para el control de Documento de Referencia.	58
14	Diagrama de actividades para el Control de Registros.	60
15	Diagrama de actividades para el entrenamiento.	62
16	Diagrama de caso de uso de los procedimientos del SGC.	63
17	Módulos del SI.	65
18	Módulo Administrar dentro del SI.	65
19	Diagrama de Caso de Uso para la Configuración.	66
20	Diagrama de Caso de Uso para Administrar documentos.	66

21	Diagrama de Caso de Uso, documentos por usuario.	67
22	Módulo de mantenimiento a Archivos.	67
23	Diagrama de Caso de Uso a archivo de Usuarios.	67
24	Diagrama de Caso de Uso a archivo de áreas o departamento.	68
25	Diagrama de Caso de Uso a archivo de puesto o cargo.	68
26	Módulo documentar.	69
27	Diagrama de Caso de Uso de documentos vigentes.	69
28	Diagrama de Caso de Uso de documentos en edición.	70
29	Diagrama de Caso de Uso de documentos obsoletos.	70
30	Módulo controlar.	70
31	Diagrama de Caso de Uso del control de registros de calidad.	71
32	Diagrama de secuencia para aviso a usuarios.	72
33	Diagrama de secuencia para crear a Usuario administrador.	73
34	Diagrama de secuencia para proceso de seguridad de registros.	74
35	Diagrama de Secuencia para proceso de asignación de Código para documentos.	74
36	Diagrama de Secuencia para Nuevo documento.	75
37	Diagrama de Secuencia para la edición de documentos.	76
38	Diagrama de Secuencia para reasignar documento a otro usuario.	76
39	Diagrama de Secuencia de Código para documentos.	77
40	Diagrama de Secuencia para asignar permiso a usuarios.	77
41	Diagrama de Secuencia para eliminar permiso a usuarios.	78
42	Diagrama de Secuencia para captura de datos a usuarios.	78
43	Diagrama de Secuencia para captura de departamento a usuario.	79

44	Diagrama de Secuencia para Nivel de seguridad a usuario.	80
45	Diagrama de Secuencia para crear área o departamento.	81
46	Diagrama de Secuencia para Actualizar área.	82
47	Diagrama de Secuencia para eliminar área o departamento.	83
48	Diagrama de Secuencia para crear cargo.	84
49	Diagrama de Secuencia para Actualizar cargo.	85
50	Diagrama de Secuencia para Eliminar cargo.	86
51	Diagrama de Secuencia para propiedades del documento.	87
52	Diagrama de Secuencia para Consulta del documento.	88
53	Diagrama de Secuencia para Consulta general del documento.	89
54	Diagrama de Secuencia para Propiedades del documento.	90
55	Diagrama de Secuencia para Edición del documento.	91
56	Diagrama de Secuencia para generar Diagrama del documento.	92
57	Diagrama de Secuencia para Propiedades de documento.	93
58	Diagrama de Secuencia para Consulta de documento.	94
59	Diagrama de Secuencia para Consulta general de documento.	94
60	Diagrama de Secuencia para Crear registros.	95
61	Diagrama de Secuencia para Consulta de registros.	96
62	Clase Usuario.	97
63	Clase Documento.	98
64	Clase Cargo.	99
65	Clase Área.	99
66	Clase Registro.	100
67	Modelo de Emplazamiento del SI.	101

Capítulo I. Introducción.

La Organización Internacional de Normalización (ISO) es una federación mundial de organismos nacionales de normalización que son llamados miembros de ISO. La elaboración de las normas internacionales, generalmente se realiza a través de los comités técnicos del ISO. Las organizaciones internacionales públicas y privadas, en coordinación con ISO, también participan en la elaboración de las normas. ISO colabora estrechamente con la Comisión Electrónica Internacional (CEI) en todas las materias de normalización electrotécnica (Mariño, 2003).

En el desarrollo de software, se utiliza la programación Orientada a Objetos (OO), un paradigma de programación que aporta grandes ventajas al software generado, como son el acercamiento entre el concepto del usuario y el resultado programado o la disponibilidad a la reutilización del código generado (Martín, 1992). El Lenguaje Unificado de Modelado (UML) permite modelar, construir y documentar los elementos que forman un sistema de software OO, convirtiéndose en estándar para la elaboración de sistemas, debido a que fue impulsado por los autores Grady Booch, Ivar Jacobson y Jim Rumbaugh. De esta forma, UML permite a un ingeniero del software expresar un modelo de análisis utilizando una notación de modelado con reglas sintácticas y prácticas (Fowler, 1999).

En este trabajo de tesis, se suman éstas dos importantes áreas, para obtener un modelo de un Sistema de Gestión de Calidad (SGC), específicamente para la norma ISO 9001:2000. En él se presentarán los principales elementos del SGC y la forma en la que se lleva a cabo, se modelarán los procesos y se identificarán los elementos y sujetos involucrados en dicho sistema, obteniendo con ello un producto para el soporte en la automatización del SGC.

1.1 Sistema de Gestión de Calidad (SGC).

El desarrollo de la calidad ha sido continuo durante los últimos 100 años. Aunque la calidad existía anteriormente en ese tiempo, su sistemático interés y su denominación de calidad, produjo cambios que se observan actualmente en la sociedad. Este desarrollo no es atribuible solamente a una persona o a unas pocas, sino a la práctica de las estrategias de dirección a través de los años, que buscaban satisfacer las necesidades y deseos de los clientes. En la Figura 1, se presenta el proceso para la mejora continua del Sistema de Gestión de Calidad (SGC).

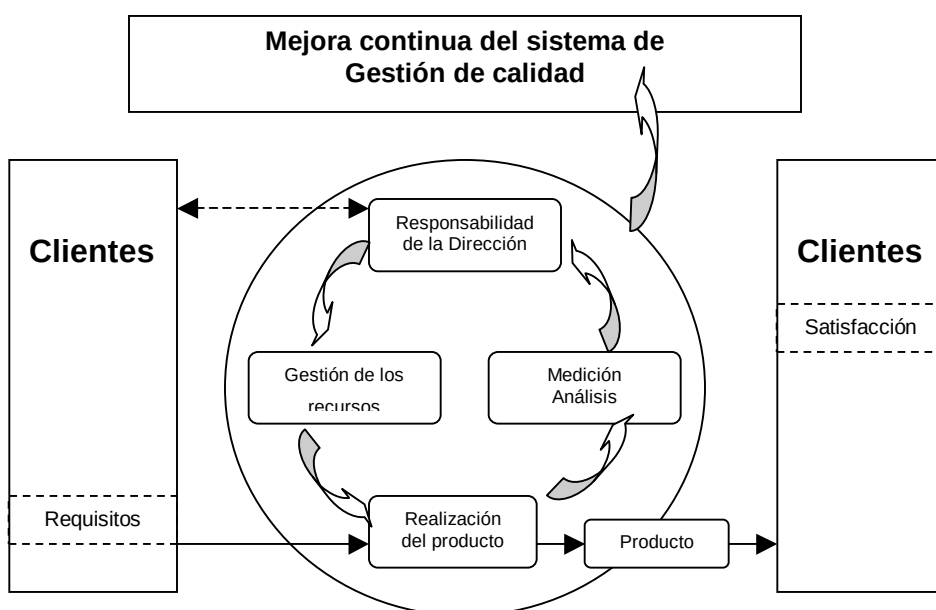


Figura 1. Mejora continua del Sistema de Gestión de Calidad.

La gestión de calidad, determina e implementa la política de calidad incluyendo una planeación estratégica, la asignación de recursos y otras acciones sistemáticas en el campo de la calidad, tales como la planeación de la calidad y el desarrollo de actividades operacionales y de evaluación relativa a la calidad (Gutiérrez, 2002). En la planeación, se incluye cuál va a ser la forma para realizar la gestión de recursos, como parte del proceso de mejora continua.

Un sistema de calidad es la forma en que una organización pone en marcha toda una serie de acciones con el objetivo de generar productos o servicios o una mezcla de ambos, que satisfagan las expectativas de sus clientes. Son aquellas acciones planificadas y sistematizadas, necesarias para ofrecer la adecuada confianza que el producto o servicio satisfará los requisitos de calidad (James, 1997). Para tener un sistema de calidad se debe de contar con una estructura organizacional, recursos, responsabilidad y procedimientos previamente establecidos para asegurar que los productos, procesos o servicios cumplan en una forma satisfactoria con el fin para los que están destinados y obviamente dirigidos hacia la gestión de calidad.

1.1.1. Familia ISO 9000.

Como ya se mencionó, un SGC es una mezcla de recursos humanos, recursos técnicos, procedimientos, procesos, organización, todo ello buscando proporcionar calidad en el servicio o en el producto suministrado. Cada empresa tiene por lo tanto, su propio sistema de calidad adaptado a su organización y a sus necesidades.

Para tener un sistema de calidad, la empresa está orientada a cumplir los objetivos relacionados con la calidad. Debe alcanzar y sostener la calidad real del producto o servicio producido, para satisfacer las necesidades que el cliente requiere. De igual manera proporcionar confianza a su misma dirección de que la calidad propuesta está siendo alcanzada y es mantenida.

Una definición de ISO se refiere a “la federación mundial de los organismos nacionales de normalización que fue fundada en 1947” (UNIT, 2003). Las normas de la serie ISO 9000 fueron creadas en el año 1987, a partir de la Norma Internacional de Aseguramiento de Calidad. La familia de normas ISO 9000 se

componía de aproximadamente 20 normas, de las cuales únicamente eran certificables las denominadas ISO 9001, 9002 y 9003. Luego de la elaboración de las primeras normas de la serie ISO 9000, se ha generado un profundo cambio en las necesidades y expectativas de las empresas y en la forma de entender las relaciones comerciales (Rothery, 1992).

En 1994 se revisaron las normas de la serie, los requisitos de la norma prácticamente no se modificaron. Se añadieron algunos elementos para facilitar la aplicación y algunos párrafos para evitar ambigüedades. A partir del año 2000, se presenta una nueva edición de la norma, en la cual se toman los elementos más importantes, unificando las tres normas (ISO 9001, 9002 y 9003) llamadas contractuales, orientadas a los procesos de la empresa.

El objetivo principal por el que se creó esta serie de normas fue armonizar los requerimientos en todos los países, conjugar todas las normas en un único juego. Debido a su origen, están fundamentalmente orientadas a la Unión Europea y también es en ella donde su grado de implantación es mayor. En Figura 2, se presentan los 10 países con mayor número de empresas certificadas hasta el 2003, de los cuales el 40% pertenecen a la Unión Europea.

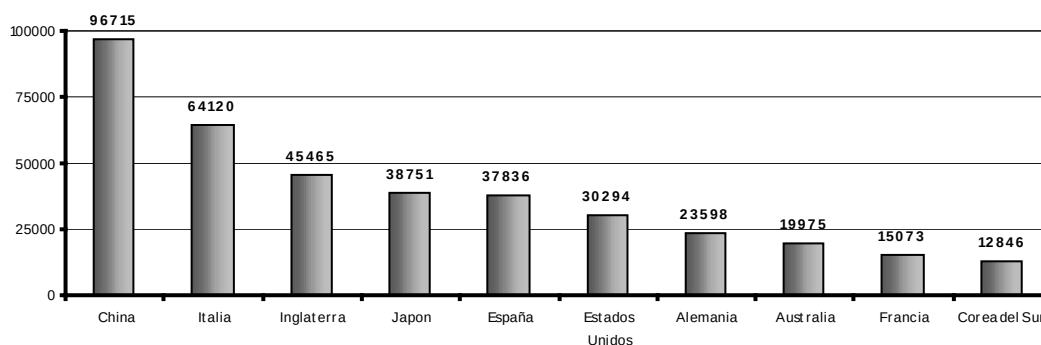


Figura 2. Países con mayor número de empresas certificadas.

Fuente:©2000 The Quality Times S.A. de C.V. (INDA) 03-2002-041511383000-01.

Una de las mayores ventajas que tiene la norma ISO, es que sirve para todo tipo de negocio, tanto para empresas manufactureras, empresas de servicios,

instituciones educativas, entre otros. Estas normas especifican los requisitos que debe cumplir un SGC cuando se debe demostrar la capacidad de una empresa para suministrar un producto o servicio de calidad. El propósito fundamental de sus requisitos es conseguir la satisfacción del cliente mediante la prevención de no conformidades en todas las etapas, desde el diseño hasta el servicio post-venta.

La aplicación de la norma requiere implantar un sistema de calidad para los procesos indirectos, es decir, no sólo para los procesos de fabricación, sino también para aquellos como formación de personal o control de documentación, que no parecen, aparentemente afectar al producto. El cumplimiento de esta norma demuestra la capacidad de una empresa para el aseguramiento de la calidad.

Para comprobar que esta norma se está cumpliendo, se realiza una verificación por una entidad externa que certifique que el sistema de calidad funciona y es efectivo para el aseguramiento de la calidad. Si la entidad externa es una entidad de certificación acreditada, entonces la empresa obtendrá un certificado de aseguramiento de calidad válido en todos los países de la Unión Europea y en algunos del resto del mundo.

La norma base ISO-9000 proporciona una guía para seleccionar la norma que debe cumplir una empresa específica para acreditarse de acuerdo con la ISO-9000 y establece los principales conceptos de calidad (Munguia, 2001). Esta familia se volvió vigente a partir del 15 de diciembre del 2000 y sustituye a la versión 1994, cuya vigencia fue hasta el 14 de diciembre del 2003. ISO 9001:1994, ISO 9002:1994 e ISO 9003:1994 desaparecen (Moreno, 2001). La familia ISO 9000 versión 2000 se simplifica, con el objetivo de evitar la proliferación de normas. Algunas normas de guía de la versión 1994 desaparecen y otras se convierten en reportes técnicos. La nueva familia ISO 9000 versión 2000 se presenta en la Tabla 1.

ISO 9000:2000	Fundamentos y Vocabulario.
ISO 9001:2000	Requisitos.
ISO 9004:2000	Recomiendas para la mejora del desempeño.
ISO 19011:2000	Orientación relativa a las auditorías a SGC.

Tabla 1. Familia actual del ISO 9000.

Un factor primordial en la operación de una empresa, es la calidad de sus productos y/o servicios. Además, en los últimos años ha existido una orientación mundial por parte de los clientes, hacia una mayor exigencia de los requisitos y expectativas con respecto a la calidad. Conjuntamente con esta orientación hay una creciente comprensión y toma de conciencia de que el mejoramiento continuo en la calidad es necesario para alcanzar y sostener un buen desarrollo económico (Stair, 2000).

Las organizaciones industriales, comerciales o gubernamentales, proveen productos o servicios que pretenden satisfacer las necesidades o requisitos del usuario. Tales requisitos son presentados como especificaciones; sin embargo, las especificaciones técnicas no pueden por sí mismas garantizar que los requisitos del usuario están satisfechas. Consecuentemente, esto ha conducido al desarrollo de normas de sistemas de calidad que complementen los requisitos del producto o servicio dados en las especificaciones técnicas.

1.1.2. ISO 9001:2000.

ISO 9001:2000 se aplica en empresas de diseño, desarrollo y servicios, su enfoque es desarrollar un equipo de calidad, documentar lo que se hace, representarlo mediante modelos, informar lo que se hace y demostrarlo con auditorías. Obteniendo con ello la certificación.

En la familia ISO 9000 la única certificable de la familia es ISO 9001:2000, la cual tiene una estructura basada en procesos, y consta de los siguientes puntos principales:

- a) Responsabilidad de la Dirección
- b) Gestión de recursos
- c) Realización del Producto
- d) Medición, análisis y mejora

Siendo un método de trabajo, que se considera como una de las formas para mejorar la calidad y por ende satisfacer al consumidor. La versión actual, es del año 2000 ISO 9001:2000, que ha sido adoptada como modelo a seguir para obtener la certificación de calidad. Es la tendencia que se tiene y a la cual debe de aspirar toda empresa competitiva, que quiera permanecer y sobrevivir en el exigente mercado actual.

Sin calidad técnica no es posible producir en el competitivo mercado. Y sin una buena organización se tienen productos con deficiente calidad. La calidad técnica se presupone e ISO 9001 propone principios para mejorar la calidad final del producto mediante sencillas mejoras en la organización de la empresa. Toda mejora, redundará en un beneficio de la calidad final del producto, y de la satisfacción del consumidor; lo que se pretende para quien adopta la norma como guía de desarrollo empresarial.

El modelo a desarrollar representa el SGC en el que se presenta la forma en la que la empresa pone en marcha toda una serie de acciones con el objetivo de ofrecer productos o servicios o una mezcla de ambos, que satisfagan las expectativas de sus clientes.

1.2. Diseño de Software Orientado a Objetos (OO).

Levine define al software como el nombre genérico dado a los programas de una computadora, pero que implica una responsabilidad adicional: asegurar que el programa o sistema cumple por completo con sus objetivos, opera con eficiencia, está adecuadamente documentado y es sencillo de operar (Levine, 2001). El realizar un software con dichas características, adecuado para su uso, de calidad duradera, y con una sólida base arquitectónica flexible al cambio, es importante utilizar el modelado como una parte fundamental.

Se construyen modelos para poder visualizar el comportamiento del sistema y poder controlar su arquitectura; estos se crean para entender mejor la entidad que se va a construir. El diseño, desarrollo y distribución de software introduce la posibilidad de establecer una línea de trabajo en el modelado del mismo. Pressman menciona que el modelado de software debe ser capaz de modelar la información que transforma el software, las funciones (y subfunciones) que permiten que ocurran las transformaciones y el comportamiento del sistema cuando ocurren estas transformaciones (Pressman, 2002). Al obtener el modelo, éste se compara con la realidad para verificar si se cumple con las metas.

En todos los ámbitos de la ingeniería se construyen modelos, que son, simplificaciones de la realidad, para comprender mejor el sistema que se va a desarrollar. Por ejemplo los arquitectos utilizan y construyen planos de los edificios; los diseñadores de empresas automovilísticas preparan modelos en sistemas de Diseño Asistido por Computadora (CAD) y Manufactura Asistida por Computadora (CAM) incluyendo todos sus detalles; y los ingenieros de software igualmente construyen modelos de sistemas de software. Un enfoque sistemático permite construir estos modelos de una forma consistente, demostrando su utilidad en sistemas de cierto tamaño.

Cuando se trata de un programa de cincuenta a cien líneas, la utilidad del modelado parece discutible pero cuando involucramos a cientos de desarrolladores trabajando y compartiendo información, el uso de modelos y el proporcionar información sobre las decisiones tomadas es vital. No sólo durante el desarrollo del proyecto, sino una vez finalizado éste, cuando se requiere algún cambio en el sistema. En realidad, incluso en el proyecto más simple los desarrolladores hacen algo de modelado. Para la construcción de modelos, hay que centrarse en los detalles relevantes mientras se ignoran los demás, por lo cual con un único modelo no es suficiente. Varios modelos aportan diferentes vistas de un sistema los cuales ayudan a comprenderlo desde varias perspectivas.

En la realización de software es necesario realizar un análisis y un modelado para obtener software de mejor calidad. Si el sistema a desarrollar es grande y complejo es más importante hacer un buen modelado ya que ayudará a entender mejor el comportamiento del sistema en su totalidad, que si este no esta modelado. El software es un elemento clave de la evolución de los sistemas y productos informáticos. En los últimos 50 años ha pasado de ser una resolución de problemas especializados y una herramienta de análisis de información, a ser una industria por sí misma. Los sistemas de información están compuestos de programas, datos y documentos, cada uno de estos elementos ponen una configuración que se crea como parte del proceso de ingeniería del software.

La ingeniería de software proporciona un marco de trabajo para que el software sea de calidad, para tener un desempeño eficiente en cuanto a velocidad y uso de recursos, de bajo costo, confiable y fácil de mantener. En relación a esto, Pressman especifica la calidad de un sistema, aplicación o producto es tan buena como los requisitos que describen el problema, el diseño que modela la solución, el código que conduce a un código ejecutable, y la pruebas que ejercitan el software para detectar errores (Pressman, 2002).

A lo largo de los años, se ha vuelto cada vez más complicado, debido al incremento en la potencia de las computadoras y a la aparición de modernos dispositivos y tecnologías, lo que ha generado la necesidad de nuevos sistemas operativos y aplicaciones de mayor complejidad. Simultáneamente, esta complejidad ha traído consigo la demanda de nuevos métodos y herramientas que ayuden en la construcción del mismo.

Uno de los problemas más comunes en el desarrollo de software es que la mayoría de los programadores y analistas son renuentes a documentar las decisiones tomadas, bien por falta de tiempo o bien por negligencia y cuando se hace, suele ser algo incompleto, no actualizado y poco consistente, ya que cada miembro del equipo utiliza una serie de símbolos familiares para él pero no para los demás. Otro problema muy habitual, relacionado con el anterior, es la absoluta escasez de procedimientos establecidos dentro de una empresa para desarrollar software. Estos problemas y otros muchos surgen principalmente por la interpretación que se hace del desarrollo de Software. En este contexto, el Lenguaje Unificado de Modelado (UML) surge como respuesta al primer problema reseñado para contar con un lenguaje estándar para escribir planos de software.

1.2.1. Lenguaje Unificado de Modelado (UML).

En el análisis y diseño Orientado a Objetos (OO), se modela el mundo y términos de objetos que tienen propiedades así como su comportamiento, y eventos que activan operaciones que modifican el estado de los objetos. Los objetos interactúan de manera formal con otros objetos. UML es una notación estándar para el modelado de sistemas de software, resultado de una propuesta de estandarización promovida por el consorcio OMG Object Management Group (OMG), del cual, a partir de 1996, forman parte las empresas más importantes que se dedican al desarrollo de software (Schmuller, 1999). Dicho lenguaje representa además, la unificación de las notaciones de los métodos Booch, Objectory (Ivar Jacobson) y OMT (James Rumbaugh) siendo su sucesor directo y compatible.

Igualmente, UML incorpora ideas de otros metodólogos entre los que podemos incluir a Peter Coad, Derek Coleman, Ward Cunningham, David Harel, Richard Helm, Ralph Johnson, Stephen Mellor, Bertrand Meyer, Jim Odell, Kenny Rubin, Sally Shlaer, John Vlissides, Paul Ward, Rebecca Wirfs-Brock y Ed Yourdon (Booch, 1999).

Con UML se documenta y especifican los elementos creados, mediante un lenguaje común, describiendo modelos con un vocabulario gráfico y con reglas para la presentación de sistemas desarrollados. El diseño de software con UML prescribe un conjunto de notaciones y diagramas estándares, para modelar sistemas y describe la semántica esencial de lo que estos diagramas significan. UML se utiliza para modelar distintos tipos de sistemas: sistemas de software, sistemas de hardware y organizaciones del mundo real (Fowler, 1999).

1.3. Problemática.

En el desarrollo de proyectos de software se presentan una serie de problemas, estos pueden ser restricciones de tiempo, funcionalidad o costo. Comúnmente en las organizaciones, los proyectos se desarrollan con sus propios lenguajes y esto es difícil de entender para una persona ajena o una nueva al grupo; además una vez que el sistema está en funcionamiento, se presenta el problema del mantenimiento. El software bien diseñado puede ayudar a reducir los costos de mantenimiento (Kendall, 1997). Una de las estrategias para abordar estos problemas consiste en utilizar técnicas de modelado de procesos para la captura, evaluación y rediseño en el proceso de desarrollo de software. Los modelos gráficos ayudan a representar el proceso en estudio, al disminuir la complejidad, propiciar un entendimiento común entre los participantes y permitir el estudio de alternativas.

La certificación del SGC es actualmente, un requisito obligado por el propio mercado. Tener un SGC certificado representa una ventaja competitiva en el esquema de globalización que se vive, así como una útil herramienta para lograr la eficiencia de toda organización.

La norma ISO 9001:2000 presenta los requisitos que se deben de tomar dentro de una empresa en relación a la organización, la formalización y las acciones preestablecidas para que el cliente este seguro de que va a recibir el objeto, previamente ofertado, conforme a la propuesta, al catálogo o a la descripción del contrato.

La documentación es un elemento básico de los SGC. Permite contar con un registro fiel, ya sea en papel o electrónico, de las operaciones del sistema. Por ello, la norma ISO 9001:2000 requiere los siguientes elementos básicos para la documentación: un manual de calidad, un manual de procedimientos e instrucciones de trabajo. Además de ser complementados con otras normas aplicables, planes, especificaciones de ensayo, contratos en vigor, cuestionarios y registros de auditorias, cuestionarios de evaluación a proveedores, registros de calidad, entre otros elementos.

1.3.1. Problemas de los Sistemas de Gestión de Calidad (SGC).

Frecuentemente, el sistema de calidad de la empresa se describe, al menos a grandes rasgos, en un documento que se le llama manual de la calidad. La lectura de este manual, informa sobre qué acciones ha puesto en marcha la empresa para suministrar servicios o productos de calidad.

En un sistema basado en papel, se deben imprimir tantas copias como sea necesario para que todo el personal las tenga a su alcance. Esto puede ocasionar costos extras y pérdida de tiempo. Conforme se van presentando cambios en los documentos, el proceso debe ser repetido en igual número de ocasiones. El

potencial de error es mayor, mientras se incrementa el número de copias y revisiones, la oportunidad de que los documentos terminen en lugares donde no deberían, se incrementa.

El proceso de liberación de documentos es extremadamente lento en un ambiente basado en papel, ya que éstos deben ser pasados de persona a persona hasta que sean aprobados o rechazados. En algunas ocasiones los papeles pueden ser extraviados u olvidados. Un sistema manual seguirá siendo moroso mientras no se haga uso de las tecnologías que se encuentran en prácticamente todas las empresas, como es el correo electrónico.

En contraste con el sistema basado en papel, en el cual hay varias copias de un mismo documento, en un sistema electrónico se mantiene sólo un documento original, y varias personas pueden consultarlo. Un sistema electrónico elimina automáticamente muchos de los pasos necesarios en un proceso manual: imprimir documentos, sacar copias y distribuirlas, ya sea para su aprobación o para su distribución final. En la Figura 3, se presenta la ubicación de la documentación como parte importante del SGC.

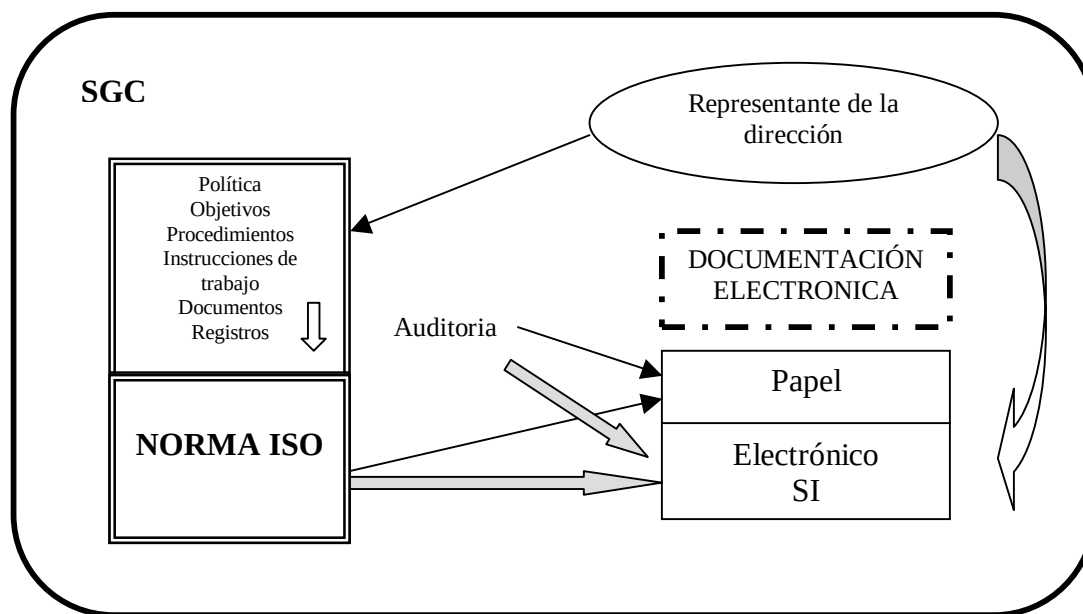


Figura 3. Ubicación de la Documentación Electrónica dentro de un SGC.

La ventaja de esto es que mientras los usuarios pueden ver un documento simultáneamente, en realidad sólo existe un original. Cuando el documento cambia, éste cambia inmediatamente para todas las personas. El proceso completo puede ser automatizado, desde la creación hasta la aprobación final y la distribución. Este nuevo paradigma en el control de los documentos elimina una de las no conformidades en las auditorías, al hacer imposible que existan documentos obsoletos en circulación.

1.4 Justificación.

En el proceso de integración de la tecnología informática, los sistemas de información juegan un papel central, debido a que mediante el desarrollo de éstos, satisfacen las necesidades de una institución y la coloca en ventaja competitiva. Esto es lo que se presentará en este documento sobre modelado de procesos; además del uso de herramientas para obtener el modelo desde una perspectiva OO.

El modelo a obtener será un sistema de software para administrar los requisitos documentales del SGC sobre la base de la normatividad ISO 9001:2000. En el modelo se observará cómo el software reemplaza la documentación impresa en papel mediante registros electrónicos el cual permite un alto grado de automatización en una interfase gráfica. El modelo presentará otro panorama de ISO, no solo como parece ser, papeleo, sino un flujo de comunicación con el propósito de compartir el conocimiento a través de la organización, brindándole al recurso humano los elementos para realizar eficientemente su trabajo.

Además de lo anterior, hoy en día, el diseñador o arquitecto de software que busca representar el diseño de un sistema puede escoger entre una amplia variedad de lenguajes de notación, cada uno de los cuales viene acompañado de

una metodología particular de análisis y diseño. En ocasiones, esta amplia variedad de opciones es un impedimento para los beneficios que promete la reutilización del software. UML representa uno de los desarrollos más significativos en el enfoque OO. Apoyados por una amplia base de compañías de líderes en el ramo, el UML fusiona lo mejor de las notaciones utilizadas por las tres metodologías más populares de análisis y diseño: Booch, OOSE y OMT.

Detrás de cada símbolo de la notación de UML hay una semántica bien definida. De esta forma, un desarrollador puede escribir un modelo en UML y otro desarrollador aún con otra herramienta puede interpretar ese modelo sin ambigüedades, debido a que sus características son interpretadas aún por personas ajenas al área de software. Esto brinda un soporte para el presente trabajo, porque el producto final podrá ser interpretado por los sujetos que participan en el SGC, quienes no necesariamente poseen conocimiento o nociones de modelado en UML. Así mismo serviría como base para el desarrollo de un SGC electrónico.

En base a lo anterior, el modelo en UML del SGC, del caso de estudio permitirá tener una planificación sobre el software que sustituye a la documentación y por la tanto a la automatización del mismo. El presente trabajo abordará la forma en la cual se trabaja en el laboratorio de metrología del Instituto de Ingeniería (II) de la Universidad Autónoma de Baja California (UABC) certificado en el año 2003 en la norma ISO 9001:2000. La documentación del SGC del Instituto se lleva a cabo en papel, es decir, se desarrollan las operaciones sobre la base de formatos escritos en papel y almacenados en archivos físicos. Esta forma de operar ha originado diversas situaciones ineficientes como la limitación de espacios de almacenamiento, la lenta velocidad de respuesta en las operaciones de recuperación de información, el uso exagerado de copias, impresiones y empastados, la pérdida de tiempo al realizar la búsqueda física de algún documento, la dificultad de desarrollar análisis de información cruzada de variables, entre otras.

El modelo obtenido del SGC del II con UML, es único, ya que actualmente no se conoce un caso de estudio de esta naturaleza, por lo cual, se contempla entre otras cosas como una buena justificación para la realización de este trabajo.

1.5 Objetivo General y Objetivos Específicos.

El objetivo general de este trabajo es el desarrollar un modelo para el sistema de gestión de calidad ISO 9001:2000 como un proyecto de software utilizando UML.

Para lograr lo anterior, se han definido los siguientes objetivos específicos:

1. Identificar los procesos de la norma ISO 9001:2000 que son susceptibles de automatizar.
2. Aplicar el lenguaje unificado de modelado UML.
3. Identificar en el modelo del proyecto los distintos tipos de diagramas que existen en UML y que puedan ser aplicados específicamente al sistema de gestión de calidad del caso de estudio, como parte del diseño de software OO.
4. Presentar el diseño de modelado mediante la herramienta SmartDraw ®.

1.6 Estructura del documento.

Este trabajo presenta un caso de estudio del sistema de gestión de calidad utilizando UML, como herramienta, aplicado a los procesos que se llevan a cabo en el Instituto de Ingeniería de la UABC.

La organización del presente trabajo consta de cinco capítulos, los cuales se describen a continuación:

Capítulo II. Presenta la información relacionada al marco teórico de los temas mas relevantes, como es SGC, Organismos certificadores modelos de objetos, modelado estático, los diagramas UML y diversas herramientas que hacen uso de la notación de UML.

Capítulo III. Describe la metodología empleada para el desarrollo del estudio apoyada del enfoque sistémico.

Capítulo IV. Se presenta el estudio de caso del SGC del Instituto de Ingeniería de la UABC, la estructura organizacional y los roles de cada uno de los sujetos involucrados en el SGC.

Capítulo V. Corresponde a los resultados obtenidos, representados en los modelos que muestran una visión mas general del caso del estudio, además como la justificación para los trabajos futuros.

Anexo A. Historia de UML. Se presentan los antecedentes de UML. Conociendo su evolución y cuáles son los cambios que tienen las diferentes versiones.

Anexo B. Diagramas de UML. Se ve en forma general cada uno de los diagramas que se utilizan en UML.

Anexo C. Diseño de Entrevista semiestructurada. Formato de entrevista para el personal involucrado en el SGC, del Instituto de Ingeniería.

Capítulo II. Marco Teórico.

En este capítulo se presenta la información, relacionada al marco teórico de los temas más relevantes. Estos son: Sistema de Gestión de Calidad (SGC), Organismos certificadores, modelos de objetos, modelado estático, los diagramas UML y diversas herramientas que hacen uso de la notación de UML. Estos elementos son una referencia para la aplicación de la metodología, y justificación para la elaboración del presente trabajo.

2.1. Certificación.

Actualmente, la certificación constituye, sobre todo, una ventaja competitiva. Una organización certificada goza de una mayor credibilidad y prestigio dentro y fuera de su entorno. El certificado es una garantía que da confianza al consumidor y que mejora la imagen pública de la empresa (Sandholm, 1995). Además de estas ventajas relativas al mercado, la certificación de un Sistema de Calidad beneficia a la organización por las ventajas que reporta el adoptar una metodología de trabajo sistemática.

2.1.1. Normas de certificación.

Para la elaboración y revisión de las normas de certificación, participan la Organización Internacional para la Normalización, el Organismo Europeo de Certificación (CEN) y otros organismos de Normalización de todo el mundo. Lo que constituye un lenguaje común en más de 70 países.

En la Unión Europea su adopción ha supuesto la sustitución de la serie EN-29000. De forma que ahora en Europa la Norma de Aseguramiento de Calidad se llama EN-ISO 9000. Diversas empresas están entrando en la certificación porque se lo

impone el mercado, sus clientes o la casa matriz, y al final descubren que conlleva muchos beneficios.

La certificación exige que la empresa disponga de una documentación que recoge la forma en que debe actuarse con respecto a aquellos capítulos que contempla la norma. Una vez que la documentación ha sido creada, el siguiente paso es cumplirla escrupulosamente. La empresa tiene un cierto grado de libertad en definir de qué forma quiere trabajar, pero una vez que se ha escogido una forma de hacer las cosas, debe cumplirlo.

Para demostrar el cumplimiento de lo establecido en los procedimientos, la empresa recurre a los registros de calidad. Los registros se generan como consecuencia de la aplicación de los procedimientos y la empresa debe archivarlos y conservarlos para demostrar el cumplimiento de los mismos. El registro es un documento que proporciona evidencia objetiva de las actividades realizadas o de los resultados obtenidos.

Una vez que la empresa está en condiciones de demostrar que cumple con lo estipulado en su documentación solicita al organismo certificador, que inicie el proceso de certificación. Dicho proceso consta de dos fases (Moreno, 2001):

1. Revisión de la documentación. Tiene por objeto comprobar que la documentación de la empresa (manual y procedimientos) recogen los requisitos requeridos por la norma.
2. Auditoria de campo. Tiene por objeto comprobar que se cumple lo estipulado en la documentación del sistema.

La primera fase la realiza normalmente el auditor en sus oficinas y consiste básicamente en comparar la documentación con los requisitos definidos por la norma. No se requieren entrevistas ni reuniones, es una labor de lectura y verificación.

Una vez que se comprueba que la documentación reúne los requisitos establecidos por la norma, es necesario comprobar el cumplimiento de los mismos. Para ello en la segunda fase, el auditor realiza una auditoria manteniendo entrevistas con distintas personas, solicitando registros, cruzando la información que recibe en distintas áreas de la empresa y en definitiva recogiendo todos los datos necesarios para hacerse una idea del grado de cumplimiento de los procedimientos.

2.1.2. Organismos de certificación.

En un mundo donde las reglas del mercado se rigen por la Calidad, las empresas tienen oportunidad de demostrar su capacidad, competitividad y calidad a través de los certificados ISO de Calidad. Estas son acreditaciones otorgadas a las organizaciones que cumplen con un marco de cumplimiento, que deberán desarrollar las organizaciones que determinen certificarse bajo esa Norma Internacional.

El certificado brinda oportunidades de realizar mejorías a través de una auditoria independiente y efectiva, así como herramientas para medir la eficacia de la gestión y el establecimiento de una base para la calidad total. El ciclo o la validez de la Certificación ISO es de tres años, durante el cual se realizan auditorias de mantenimiento cada seis meses a las empresas galardonadas.

Las instituciones certificadoras de la Calidad ISO son organizaciones reconocidas internacionalmente. Para América Latina existe Lloyd's Register Quality Assurance, una organización del Grupo Lloyd's Register, que ha emitido certificaciones desde 1990. El Grupo fue fundado en Londres en 1760 y nació como una sociedad clasificadora de barcos para orientación de aseguradores, armadores y de constructores navales.

En 1985, y debido a su constante preocupación por el aseguramiento de la Calidad, fue seleccionada por el gobierno Británico para certificar Sistemas de Gestión de Calidad. De esta manera, se creó Lloyd's Register Quality Assurance (LRQA) que ha sido la primera certificadora del mundo acreditada por UKAS (el antiguo NACCB), otorgándole la Acreditación Número 001.

La norma esta avalada por la Sociedad Americana de Calidad (ASQ), los Institutos Europeos de Estándares y el comité Japonés de Estándares Industriales. En México, existen normas oficiales para sistemas de calidad, de acuerdo al giro de la organización, es la norma que se aplica (IMNC, 2003).

2.2. Instituciones de Educación Superior y Sistemas de Gestión de Calidad.

La educación es un proceso sistémico de cambio cultural y de mejora continua. La cultura de calidad implica participar, cooperar y renovar las ideas y las estructuras, adecuarse a las necesidades y expectativas de la vida actual, guardando una actitud de dirección al futuro, y a largo plazo. La educación es una herramienta para forjar ciudadanos más libres, aptos, competentes, responsables y creativos, al servicio de los intereses de su comunidad.

La educación de excelencia es una disposición permanente de aprender a ser, y de construir visiones, valores y modelos de comportamiento nuevos, de aprender a aprender. El cuestionamiento y la exigencia de la enseñanza se plantean con fuerza. Esta demanda está vinculada a las exigencias que la sociedad actual le plantea a los ciudadanos, empresas, organismos, instituciones y actores sociales en general (Figuerola, 1995). Su campo de acción abarca toda la vida y todas las facetas, dimensiones, facultades y posibilidades del ser humano y de su ambiente natural; pero para poder llevarlos a realización feliz se requiere del compromiso y el esfuerzo sostenido de las organizaciones y personas involucradas: los

educandos, los maestros, los padres de familia, los empresarios y profesionales, los directivos de instituciones educativas, los organismos docentes no-lucrativos y los líderes y/o grupos interesados de la población.

Un movimiento hacia la calidad educativa exige de trabajo interdisciplinario en equipo y el nuevo papel del director como gestor eficiente del cambio (Lafourcade, 1993). De igual forma es recomendable fomentar la participación de todos los elementos democráticos y la concertación, abrir los espacios y las oportunidades de mejora, capacitación y superación académica para todos.

Para garantizar la calidad educativa no todos deben de pensar igual, pues la pluralidad de opiniones y puntos de vista enriquecen la discusión y, frecuentemente, se generan opciones de mejora y compromisos, como resultado del consenso, cuidándose dos aspectos principales:

1. Un ambiente de trabajo profesional con libertad y autonomía en donde todos sientan que pueden expresarse.
2. Una actitud de prudente respeto a las visiones, creencias, opiniones y contribuciones de los otros, saber escuchar con atención y empatía.

El aprendizaje significativo se facilita cuando se cuenta con el apoyo de otros aliados importantes: los profesionales y trabajadores que dominan el proceso, los padres de familia, los directivos experimentados, los investigadores del proceso educativo y los líderes de la población. Corresponde a cada universidad e institución, con base en sus normas e interpretaciones que su personal académico y directivo hace de las circunstancias del entorno social y físico, definir esos compromisos, ponderar su potencial en la enseñanza, en la investigación y en la extensión académica juzgar su resultado (UABC, 1997).

El sector educativo de México es enorme y heterogéneo, está hoy sujeto a presiones externas fuertes y adolece de grandes carencias y rezagos, para vencer

estas resistencias. El modelo que se ha aprendido en décadas pasadas permitió trabajar con las cosas, diseñar programas y cursos y a reducir los costos, pero no se aprendió a cooperar en equipo, ni a compartir el poder y a sustentar comportamiento en principios y valores éticos.

El nuevo modelo de calidad es la administración cuidadosa de las personas, los procesos y los recursos necesarios, en base a la visión y el liderazgo gerencial, para desarrollar nuestra organización hacia la excelencia. Los clientes más importantes de la institución educativa son sus propios profesionales, su coraje y decisión de ser mejores, reconociéndoles su dignidad, entusiasmo, habilidades y saber que se comparten. Se debe crear un propósito indeclinable de mejorar los servicios y los productos y de satisfacer plenamente las expectativas de los usuarios, es muy importante instituir el liderazgo y demoler las barreras entre el personal, principalmente aquéllas que impiden que los maestros se enorgullezcan de su trabajo

Las Instituciones de Educación Superior (IES), reciben apoyos federales, para establecer sistemas de calidad en las IES desde el sexenio pasado de acuerdo al plan nacional de educación del período 1996-2000. La actual administración establece que la calidad se debe presentar en el sector educativo, por lo que ha comenzado a desarrollar mecanismos para elevar la calidad de los programas de nivel superior.

En el ámbito educativo la instituciones son susceptibles a la certificación, programas de estudios, procesos administrativos y servicios que puedan impartir las instituciones educativas, ejemplo de ello es la Universidad Autónoma de Baja California (UABC).

En el acuerdo que fija las bases para la transparencia a la información (UABC, 2003) relacionada con la administración de los recursos económicos y los

indicadores de calidad institucional, unidades académicas y programas educativos de la UABC, se expide el siguiente acuerdo:

Primero: La Universidad hará pública, de manera permanente y sin necesidad de que medie solicitud de algún miembro de la comunidad universitaria o ciudadanía en general, a través de medios remotos o locales de comunicación electrónica de la UABC, la siguiente información:

- Los indicadores de calidad de los programas Institucionales, unidades académicas y programas educativos siguientes:
 1. Certificación ISO 9000:2000 de los procesos de administración de recursos humanos, control escolar, control y ejercicio presupuestal, biblioteca, adquisiciones y construcciones.

Segundo: La Secretaria General tendrá la responsabilidad de coordinar y supervisar la tarea de recabar y colocar a través de medios remotos o locales de comunicación electrónica de la UABC, la información a que se refiere el numeral anterior, además de propiciar que las dependencias administrativas y unidades académicas la actualicen periódicamente.

En el 2003, la UABC ya contaba con el 36% de la matrícula acreditada, ocupando el segundo lugar, a nivel nacional, estando por encima de instituciones privadas, como lo es el Tecnológico de Monterrey. La UABC sobresale en el ámbito nacional de la Educación Superior por ser una de las Instituciones con mas programas acreditados y por la calidad de sus programas de postgrado (Gaceta, 2003). Actualmente se tiene mas del 90% de sus programas de estudios acreditados.

El Instituto de Ingeniería tuvo sus inicios en el año de 1963, con el nombre de Instituto de Investigaciones Industriales y de Ingeniería, con sede en la ciudad de Tijuana, Baja California. Posteriormente suspendió sus labores en enero de 1974,

para reiniciarlas, el día 5 de enero de 1981, en la ciudad de Mexicali. Durante este tiempo el instituto ha logrado obtener un reconocido prestigio en la solución de problemas relacionados con el Ahorro de Energía Eléctrica, Transporte, Climatología Sinóptica, Manejo de Residuos Sólidos, Corrosión y Materiales, Aguas Residuales, Evaluación y Exploración de Yacimientos Geotérmicos y Metrología Eléctrica.

Sin embargo, con el propósito de emprender acciones estratégicas que coadyuven a la consolidación del instituto y a una búsqueda permanente de nuevas áreas de oportunidad, que le permitan ser más competitivo ante los nuevos retos que enfrenta nuestro país, con la evidente tendencia hacia la globalización y los mercados de libre competencia, se propuso elaborar un plan estratégico, partiendo de la base de lo que es el Instituto y la firme convicción de lo que desea ser.

El problema actual en el que se encuentran las IES es tener que operar y crecer dentro del ambiente cada vez más competitivo. Para hacer frente a esta competitividad, las instituciones no sólo deben elevar la calidad de los productos y servicios, sino reducir tiempos y minimizar costos. Para lograr objetivos competitivos, las instituciones deben optimizar su funcionamiento, un primer momento de esta optimización es realizar la tarea de modelado de institución, el cual debe explicitar su composición y reflejar los problemas de tiempos perdidos, costo y calidad. Con esta tarea, los componentes de la institución pueden ser ajustados y entonces el modelo se analiza para examinar los efectos del cambio. Además con el modelo se permite experimentar el cambio y visualizar el éxito antes de la implementación real del cambio.

Históricamente, los modelos del negocio o de las instituciones han sido sólo la representación jerárquica de la estructura de la organización y no contempla tiempos perdidos, costos y problemas de calidad. La verdadera optimización institucional se da perfeccionando los procesos que la institución realiza. Un proceso de la institución representa cómo se hace el trabajo en la organización en

lugar de mirar la organización verticalmente (usando una jerarquía), la vista correcta del negocio se toma horizontalmente, analizando los procesos de la institución que fluyen por los límites organizacionales. De manera natural, un proceso institucional inicia por un evento, consume recursos, se realiza actividades y elaboran productos o proporciona servicios.

2.3. Modelado.

El modelado es una técnica utilizada en todas las ingenierías. Como tal la ingeniería de software debe basarse en el modelado como una parte central de todas las actividades que conducen a la producción de software de calidad.

El uso de los modelos se presentan para:

1. Facilitar la comprensión. Un modelo generalmente es más sencillo que su entidad. Es más fácil entender la entidad si sus elementos y sus interrelaciones se presentan de manera simplificada.
2. Facilitar la comunicación. Una vez que la persona que va a resolver un problema entiende la entidad. Es común que necesite comunicar ese entendimiento a otros. Quizá el analista de sistemas deba comunicarse con el gerente o con el programador. O tal vez el gerente deba comunicarse con otros miembros del equipo que va a resolver el problema.
3. Predecir el futuro. La precisión con que el modelo matemático puede representar su entidad le confiere una capacidad especial que no pueden ofrecer los otros tipos de modelos.

Las pruebas técnicas del modelado se han desarrollado desde los 90s, inicialmente debido a una extensión de la Reingeniería de los Procesos del Negocio (BPR) y las iniciativas del enfoque OO. Hoy los modelos de flujo de trabajo, el uso del UML, la Tecnología Relacional, la OO y el Modelamiento de Casos de Uso representan estándares que son intercambiables de manera importante. Herramientas de esta naturaleza en los años ochenta originalmente se

denominaron herramientas CASE (Computer Assisted Software Engenering); cedieron el paso a las herramientas RAD (Rapid Applications Developmen) de la arquitectura cliente/servidor, la emergencia de herramientas para el Análisis y Diseño Orientado a Objetos (OO A&D). Generalmente, el problema global que estas herramientas intentan resolver es proporcionar una solución integrada del concepto-al-código para desarrollo de los sistemas comerciales.

Estas herramientas no han mantenido la expectativa debido a las razones siguientes: estar dirigidos al diseño del sistema antes que a los requisitos del negocio, dificultad para transitar de los requisitos del negocio a los componentes del sistema, no proporcionan ninguna posibilidad de pasar de las necesidades del negocio a los componentes del sistema, carencia de una comprensión amplia y profunda de la empresa que se intenta automatizar. La estrategia del concepto-al-código no se lleva a cabo con un solo conjunto de herramientas. Pero la tecnología, el análisis y métodos de diseño, y el campo principalmente independiente de la BPR están evolucionando para hacer del concepto-al-código una posibilidad. El ciclo de vida del desarrollo de una aplicación inicia con la organización y termina en el código. En este enfoque se considera que el ciclo de vida del desarrollo de software tiene varios **s** ciclos de interacción, pero en general, se pueden resumir en tres estadios mayores o clásicos: el conceptual, el lógico y el físico.

El estadio conceptual del ciclo de vida involucra comprender el negocio. Algunas veces esto se realiza para mejorar los procesos del negocio, otras veces se utiliza como una guía de desarrollo o venta de una aplicación. El trabajo aquí es llamado Modelamiento Empresarial o Análisis del Negocio. Un componente clave de este estadio se llama requerimiento del negocio.

En el estadio lógico, los requerimientos del negocio ayudan a guiar la ingeniería de los componentes del sistema. Este estado se identifica como la arquitectura y diseño del sistema. El producto final son las especificaciones del sistema. En el

estadio físico esto involucra la ingeniería de nivel bajo, la codificación, la prueba, el mantenimiento y desarrollo. El producto final es el sistema.

En esencia, el ciclo de vida de desarrollo de una aplicación involucra dos grandes elementos: el negocio y el sistema. La BPR y enfoques complementarios representan la moderna metodología para mejorar los procesos del negocio y los sistemas. La BPR cambia las prácticas actuales para conducir el negocio, los cuales satisfacen las demandas del mercado y la incorporación de las capacidades de la tecnología moderna. Para afectar los cambios en los negocios se emplea la tecnología. La moderna tecnología para el análisis, diseño y desarrollo de los sistemas es la OO, el modelamiento del Caso de Uso, un enfoque de interacciones entre las personas y sistemas que funcionan.

El modelado de la institución del caso de estudio involucra el estado conceptual del ciclo de vida. Esto es donde la empresa se fragmenta en piezas para un mejor entendimiento de los procesos del negocio y de su reingeniería, los requerimientos de la empresa, o sea el análisis del negocio. Regularmente el modelado de una empresa reside principalmente en la metas de sus empleados y en sus aplicaciones del software.

Los beneficios del modelo obtenido son los siguientes: alinear objetivos del sistema con objetivos de la institución; ventajas competitivas exitosas al desplegar aplicaciones que satisfacen los requisitos del negocio; modelos con base en el consenso que guían el desarrollo, selección o adaptación del software; valoración, perfeccionamiento y mantenimiento de aplicaciones evaluando modelos, no código; se tiene un patrón arquitectónico que permite la reutilización, el refinado y despliegue de objetos de negocios comunes en múltiples sistemas de cómputo y en diversos ambientes (Pressman, 2002).

Una incongruencia en empresas grandes y modernas es la diferencia entre los modelos mentales de sus empleados (que reflejan el mundo real que ellos

comporten) y la implantación de procesos comerciales en sus aplicaciones del software (McLeod, 2000). El modelado de los procesos del negocio y de los sistemas antes de codificarlos proporcionan un “prototipo” conceptual que asegura que los usuarios y desarrolladores de software tengan una comprensión común de los sistemas a ser desarrollados. Esto es semejante a la construcción aerospacial o automotriz donde los modelos, prototipos, y los diagramas de ingeniería se desarrollan antes de caer en la elaboración compleja y costosa de los productos.

Los proyectos de desarrollo de software que no buscan la correspondencia con modelos de negocios conceptuales y sistemas lógicos, productos de estos modelos, tienen menores posibilidades de éxito, tanto a corto como a largo plazo. Para construir software congruente con las necesidades de una empresa es clave importante para la estrategia del concepto-al-código.

2.3.1 Modelado de sistemas, procesos y datos.

Para construir un modelo del sistema, se deben considerar algunas restricciones:

1. *Supuestos* que reducen el número de permutaciones y variaciones posibles, permitiendo así al modelo reflejar el problema de manera razonable.
2. *Simplificaciones* que permitan crear el modelo a tiempo.
3. *Limitaciones* que ayudan a delimitar el sistema.
4. *Restricciones* que guían la manera de crear el modelo y el enfoque que se toma al implementar el modelo.
5. *Preferencias* que indican la arquitectura preferida para todos los datos, funciones y tecnología.

El modelo de sistema resultante puede reclamar una solución completamente automática, semiautomática o un enfoque manual. De hecho es posible a menudo caracterizar modelos de cada tipo que sirven de soluciones alternativas para el problema que se tenga. El ingeniero del sistema modifica la influencia relativa de

los diferentes elementos del sistema para crear modelos de cada tipo (Lardent, 2001).

Al crear un sistema, es preciso documentar, o modelar, tanto sus datos como sus procesos. Algunas de las herramientas para trabajar con sistemas se pueden agrupar, tomando como criterio esas dos tareas. Las herramientas de modelado de datos describen los datos que un sistema o conjunto de sistemas usa y las herramientas de modelado de procesos describen los procesos que un sistema ejecuta. Las herramientas de modelado de objetos combinan el modelado de datos y de procesos (James, 1997).

Las herramientas para modelar datos y procesos también pueden clasificarse como base en su nivel de detalle. Algunas herramientas documentan sus datos o procesos de manera resumida, mostrando sólo sus características principales, otras herramientas documentan sus datos o procesos de forma muy específica, revelando los detalles más intrincados.

Antes de que surgiera el análisis y el diseño estructurado, las herramientas sólo podían documentar en los niveles de mayor síntesis y de mayor detalle, no había término medio. La situación era cómo usar un mayor detalle; no había término medio. La llegada de las herramientas estructuradas en forma descendente hizo posible documentar los datos y procesos con diferentes niveles de detalle entre los dos extremos (Jacobson, 2000).

2.3.2. Diagramas y Herramientas para el Diseño de Software con UML.

Una de las partes más importantes en el desarrollo del software es la escritura del código, aunado a esto las siguientes razones justifican el uso de UML.

- Aprendizaje OO. Las técnicas en UML fueron diseñadas en cierta medida para ayudar a los usuarios a hacer un buen desarrollo de OO.

- Comunicación con los expertos del dominio. Logra una buena comunicación, junto con una comprensión adecuada del mundo del usuario, es la clave para el desarrollo de buen software.
- Comprensión del panorama general. Presenta una visión general y amplia de proyecto en cuestión.

Los diagramas de UML son valiosos para entender de manera general el sistema. Para la construcción de modelos, hay que centrarse en los detalles relevantes mientras se ignoran los demás, por lo cual con un único no es suficiente. Varios modelos aportan diferentes vistas de un sistema los cuales ayudan a comprenderlo desde varias perspectivas. Así, UML recomienda la utilización de nueve diagramas para representar las distintas vistas de un sistema. Estos diagramas de UML son los siguientes (Fowler, 1999), los cuales se muestran más ampliamente en el anexo B.

- Diagrama de Casos de Uso: modela la funcionalidad del sistema agrupándola en descripciones de acciones ejecutadas por un sistema para obtener un resultado.
- Diagrama de Clases: muestra las clases (descripciones de objetos que comparten características comunes) que componen el sistema y cómo se relacionan entre sí.
- Diagrama de Objetos: muestra una serie de objetos (instancias de las clases) y sus relaciones.
- Diagrama de Secuencia: enfatiza la interacción entre los objetos y los mensajes que intercambian entre sí junto con el orden temporal de los mismos.
- Diagrama de Colaboración: igualmente, muestra la interacción entre los objetos resaltando la organización estructural de los objetos en lugar del orden de los mensajes intercambiados.
- Diagrama de Estados: modela el comportamiento de acuerdo con eventos.
- Diagrama de Actividades: simplifica el Diagrama de Estados modelando el comportamiento mediante flujos de actividades.

- Diagrama de Componentes: muestra la organización y las dependencias entre un conjunto de componentes.
- Diagrama de Despliegue: muestra los dispositivos que se encuentran en un sistema y su distribución en el mismo.

Algunas aplicaciones de UML es primeramente para sistemas de software. Este ha sido usado efectivamente en campos como:

- Sistemas de información de empresas.
- Servicios de banco y financieros.
- Telecomunicaciones.
- Transportación
- Defensa/aeroespacial.
- Ventas
- Electrónica médico
- Científicos
- Servicios basados en Web.

UML no está limitado al modelado de software. De hecho éste es suficientemente expresivo para modelar los diferentes tipos de sistemas como es: sistemas de software, sistemas de hardware, organizaciones del mundo real, etc.

En el diseño de software por medio de UML es útil por los siguientes motivos:

- Obliga a separar claramente la fase de diseño de la de codificación. Por más experiencia que se tenga en la programación, ésta se ve mejorada si al inicio se toman buenas decisiones y así al estar codificando, implementamos eficientemente los algoritmos. Es decir se atienden los verdaderos problemas de programación propiamente dicho.
- Obliga a desarrollar programas mejor estructurados. Esto es debido a que se basa en un modelo OO. Usar una técnica basada en UML, permite ver

rápidamente cuando alguna decisión de diseño *no tiene lógica* y debe ser cambiada.

- Es un buen mecanismo de documentación. Dado que UML sigue una notación estándar otras personas podrán entenderlo y por tanto incorporarse al equipo de desarrollo sin tener que aprender notaciones nuevas.

Gran cantidad de aplicaciones de software modernas se han desarrollado basadas en principios OO (clases, atributos, métodos, herencia). Una de las ventajas del enfoque OO es que se presta a la representación notacional de sus principios. Las herramientas de Computer Aided Software Engineering (CASE) proporcionan el soporte para las metodologías y notaciones OO e incluso permiten la generación de parte del código y de documentación de las aplicaciones. Las nuevas versiones de las herramientas OOCASE están ya incorporando los lenguajes de programación OO como Java, C++, Ada y Gestión Estándar del Grupo de Gestión de Objetos (CORBA). Algunas de estas herramientas también dan soporte a bases de datos relacionales, permitiendo el modelado y diseño lógico y en algunos casos el físico, de bases de datos, incluyendo la generación de esquemas e ingeniería inversa sobre tablas y otros elementos.

Las herramientas CASE ofrecen grandes ventajas a los desarrolladores de sistemas software de gran tamaño. Mientras una espiral de requisitos del sistema sigue llevando la complejidad del sistema a nuevos niveles, las herramientas CASE permiten abstraernos del código fuente, a un nivel donde la arquitectura y el diseño son mas aparentes y fáciles de entender y modificar. Se puede decir que cuanto mayor es el proyecto software, más importante resulta el utilizar tecnologías CASE.

Debido a que los desarrolladores en este tipo de sistemas interactúan sólo con una o varias partes del sistema diseñado por los diseñadores, deben encontrar de manera sencilla el subconjunto de clases y métodos, así como asimilar y entender

cuales son las interfaces con ellos. En el mismo sentido el gestor del proyecto debe ser capaz de ver una representación del diseño a alto nivel y entender cómo funciona en un tiempo razonable. Por estas razones, las herramientas CASE junto con las metodologías nos proporcionan una forma de representar sistemas demasiado complejos de entender ya sea a partir del código subyacente o de alguna forma basada en esquemas.

Para alcanzar el grado de abstracción necesario, las herramientas OOCASE permiten crear diagramas que representan el modelo de objetos utilizando los elementos notacionales de metodologías específicas como: OMT (Object Modeling Technique), Booch, Jacobson Use Cases , UML o Fusión. El soporte notacional es sólo una forma de clasificar las herramientas CASE, otras características a tomar en cuenta son: el soporte de varios lenguajes de programación, el uso de archivos o bibliotecas para almacenar la información del modelo y la integración con software de control de versiones.

Las herramientas OOCASE crean diagramas que representan el modelo de objetos utilizando los elementos notacionales de las diferentes metodologías. Estas notaciones representan gráficamente clases (incluyendo atributos, métodos y eventos), relaciones entre objetos (herencia, agregación, colaboración, realización, generalización) y cardinalidad. Además la mayoría, aunque no todas, las herramientas OOCASE ofrecen la posibilidad de generar un esqueleto del código de la aplicación. Este esqueleto generalmente incluye la definición de las clases y los prototipos de las funciones.

Analizando la mayor parte de las aplicaciones software nos podemos encontrar con elementos tan dispares como páginas Web, acceso a bases de datos, código C++, código Java, interfaz CORBA de acceso a un servidor C++, etc. Esta disparidad es el reto básico que tienen que afrontar los vendedores de herramientas CASE que quieran proporcionar un entorno de diseño integrado. Con los años va siendo mayor la capacidad de trabajar más con menos herramientas,

lo que supone una mayor integración y una menor complejidad en las herramientas CASE, que lleva a una mayor productividad.

A la hora de alcanzar una mayor integración, UML se presenta como una buena alternativa, ya que se basa en un meta modelo extensible. El meta modelo UML es un modelo que describe modelos. Una visión de este modelo nos muestra elementos orientados a objetos como clases, atributos y métodos, todos ellos y sus interrelaciones representados como meta clases. Al modelo UML se le pueden añadir más meta clases, por ejemplo para describir modelos entidad-relación, así como realizar extensiones para tratar con interfaces como las definidas usando CORBA. El estándar UML define notaciones para representar la información y su interrelación desde una perspectiva gráfica, además soporta el uso de un formato de intercambio. El CASE Data Interchange Format (CDIF <http://www.cdif.org>) es un anexo importante al UML, ya que, aunque UML especifica el tipo y la relación que existe entre la información que ha de ser almacenada en una biblioteca (repository), no especifica cómo se debe guardar esta información, dejando la elección a los distintos vendedores.

El CDIF permite una representación ASCII para una biblioteca de diseños, dicha representación puede ser utilizada para transferir los datos del diseño de una herramienta CASE a otra, pudiendo ser utilizada por generadores de informes y otras herramientas de este estilo. Cuando se dice que una herramienta OOCASE "genera código", nos estamos refiriendo a cabeceras y prototipos en el caso de C++ o Java y a información de esquema en el caso de bases de datos relacionales. O sea, se proporciona la base para clases, métodos, y atributos en aplicaciones orientadas a objetos, así como tablas, columnas y relaciones en DBMS relacional. El tener una herramienta CASE que proporcione código de aplicación y de base de datos ayuda a garantizar que el modelo y su implementación estén sincronizados. Esta sincronización hace que la herramienta de modelado sea más útil para los desarrolladores. Sin un mecanismo integrado para guardar la implementación y el modelo sincronizados, se produce con

frecuencia una desconexión entre ambos. Dicho de otra forma, es frecuente que las mejores soluciones se hagan evidentes en el proceso de implementación, si estas soluciones, que llevan con frecuencia a modificaciones en el diseño, no son realimentadas hacia atrás en el modelo de una forma conveniente, el modelo puede pasar a estar "out of date". El valor intrínseco del modelo se pierde desde el momento en que no podemos confiar en él y por lo tanto deja de usarse.

Para soportar la generación de archivos de cabecera y prototipos de métodos, debe haber una forma de asegurar que la herramienta no sobrescribe el cuerpo de los métodos y otro código y comentarios introducidos por el desarrollador. La forma más fácil y frecuente de hacer esto es introducir información del modelo dentro del código generado. Esto suele llevar a las protestas de los desarrolladores cuando utilizan por primera vez una herramienta de este tipo. Estas protestas son entendibles, ya que la legibilidad del código fuente se reduce de una forma significativa debido a la información introducida por la herramienta CASE en bloques de comentarios. Dos razones reducen la importancia del problema de "polución" del código:

- Los desarrolladores están acostumbrados a ver los comentarios extra y filtrarlos mentalmente.
- La utilización cada vez más frecuente de entornos de desarrollo integrado cada vez mas sofisticados (ej.: Visual Basic de Microsoft). Estos entornos ofrecen browsers para clases y métodos que hacen la navegación hacia declaraciones y definiciones específicas más sencillas, con lo cual los desarrolladores pueden evitar el "ruido" introducido en los ficheros de código nativos.

Las herramientas OOCASE necesitan colaborar con otras aplicaciones a lo largo del ciclo de vida del proyecto. Esto es particularmente cierto en el caso del software de control de versiones. Debido a que los modelos están expuestos a frecuentes cambios durante el análisis y el diseño , muchos vendedores integran ya este tipo de software. Además, las herramientas CASE deben trabajar en un

entorno multiusuario ya que los proyectos grandes los modelan y construyen equipos de desarrolladores. Las herramientas CASE que almacenan los modelos en ficheros del sistema operativo en vez de utilizar una biblioteca almacén, deben definir unidades de modelo que puedan ser utilizadas de forma concurrente por diferentes diseñadores y analistas. Estas unidades además deben estar bajo el control de algún software de versiones. La granularidad de las unidades de este tipo es un factor importante ya que con frecuencia diferentes partes del modelo están entrelazadas. Cuando un desarrollador coge una parte del modelo, esa parte debería ser lo suficientemente pequeña como para que otros desarrolladores puedan coger otros elementos del modelo y trabajar de una forma independiente. Hay que tener en cuenta que cualquier parte del modelo que es común a más de una unidad y a la que se acceda con frecuencia, puede convertirse en un cuello de botella para los procesos de modelado.

Las herramientas OOCASE seguirán en el futuro soportando diferentes metodologías. Incluso cuando UML alcance su esperada penetración en el mercado, aparecerán otras metodologías con gran número de seguidores, existiendo siempre varias notaciones y metodologías activas. Es por eso, que será importante que las herramientas CASE que ahora almacenan la información del modelo en ficheros evolucionen hacia almacenes a los que puedan acceder diferentes herramientas. Esto hará más fácil el intercambio de información entre herramientas que realizan clases específicas de modelado. De todas formas es también deseable que se reduzca el número de herramientas CASE necesarias para modelar una aplicación.

Una empresa de software con éxito es aquella que produce de una manera consistente software de calidad que satisface las necesidades de los usuarios. Un empresa que puede desarrollar este software de forma predecible y puntual con un uso eficiente y efectivo de recursos, tanto humanos como materiales, tiene un negocio sostenible.

Capítulo III. Metodología.

En este capítulo, se describe la metodología utilizada en la investigación, comenzando por los sujetos que fueron objeto de estudio, los materiales utilizados en la investigación y los procedimientos con los que se desarrolló el presente estudio.

3.1. Método.

En el presente trabajo se utilizó la metodología Orientada a Objetos (OO), debido a que el objetivo es elaborar el diseño y modelación del Sistema de Gestión de Calidad (SGC) ISO 9001:2000 como un proyecto de software utilizando UML. Con la metodología OO se llegó al objetivo que es construir un modelo de dominio de aplicación añadiendo detalles de implementación durante la fase del diseño de software.

3.2. Sujetos.

En este estudio se trabajó con el personal que está involucrado con el SGC del Instituto de Ingeniería de la UABC, ellos son: el director, representante de la dirección, administrador, subdirector, jefe de laboratorio de metrología y técnico de laboratorio de metrología.

3.3. Materiales.

Los materiales utilizados en el presente trabajo fueron: la documentación del SGC del Instituto de Ingeniería, la norma ISO 9001:2000 cuestionarios, equipos de cómputo y software que soporte UML.

3.4. Procedimiento.

Al iniciarse este proyecto se contemplaron varias metodologías de sistemas de información. Considerando el objetivo principal de este trabajo, se optó por la metodología OO ya que en las diversas etapas de esta metodología se utiliza el modelado como herramienta principal para el análisis y diseño de sistemas de información.

En la primera etapa que es el análisis se realizaron encuestas con las personas que participan en el SGC. Obteniendo información en la cual se presentan las necesidades de los usuarios, así como el alcance que tiene actualmente el sistema. Se construyó un modelo de la situación del mundo real mostrando sus propiedades importantes. El modelo obtenido en el análisis es una abstracción resumida y precisa de lo que debe hacer el sistema deseado y no la forma en que se hará. Los objetos del modelo deberán ser conceptos del dominio de la aplicación y no conceptos de implementación. En esta etapa también se consultan los lineamientos de la norma ISO 9001:2000.

En la segunda etapa que es el diseño del sistema, se tomaron decisiones en base a cómo se está llevando actualmente el SGC en forma global; se organiza en subsistemas basados tanto en la estructura del análisis como en el diseño propuesto. Se decide qué características de rendimiento hay que optimizar.

En la tercera etapa que es el diseño de objetos se construye un modelo de diseño basándose en el modelo de análisis que lleven incorporados detalles de implementación.

Capítulo IV. Estudio de caso: Sistema de Gestión de Calidad del Instituto de Ingeniería, UABC.

4.1 Descripción.

Como ya se mencionó el II inició en la década de los 60's y a través del tiempo ha obtenido prestigio. Buscando siempre contribuir a mejorar la calidad de vida de los mexicanos y en particular de la comunidad bajacaliforniana, a través de la investigación en áreas prioritarias para el desarrollo del país y del estado, la formación de recursos humanos de alto nivel y la oferta de servicios de ingeniería de alta calidad, a los diferentes sectores de la sociedad.

El II de la Universidad Autónoma de Baja California (UABC), tiene en su haber un laboratorio certificado bajo la norma ISO 9001:2000, esta norma gestiona y asegura la calidad de los sistemas. Se cuenta con un manual de calidad, especificando la visión general del sistema de calidad. Dicho documento fue creado por el comité de calidad y contiene la Política y los Objetivos de Calidad del II, reflejando la forma adecuada para llegar a la calidad deseada.

Las instrucciones de trabajo y procedimientos fueron creados por las partes involucradas, el jefe de laboratorio de Metrología y el técnico calibrador, donde se refleja y se describen los procesos de trabajo de dicho laboratorio. En este manual se explica: qué se hace, dónde se aplica, cuándo se aplica, quién es el responsable y porqué se hace. Las instrucciones presentan paso a paso lo que los empleados siguen para llenar los requerimientos de cada trabajo, esto se deben seguir para realizar la operación correctamente.

Se cuenta con instrucciones detalladas en las cuales se presenta el “cómo” se hacen las operaciones.

También se tienen formas y registros que son evidencia documentada de que una operación está acorde con el sistema establecido por los primeros tres niveles de documentación. Por ejemplo los empleados llenan una lista de chequeo para

indicar que han realizado una operación de acuerdo a las instrucciones documentadas.

El personal y departamentos involucrados en el SGC del II se presenta en la siguiente Figura 4.

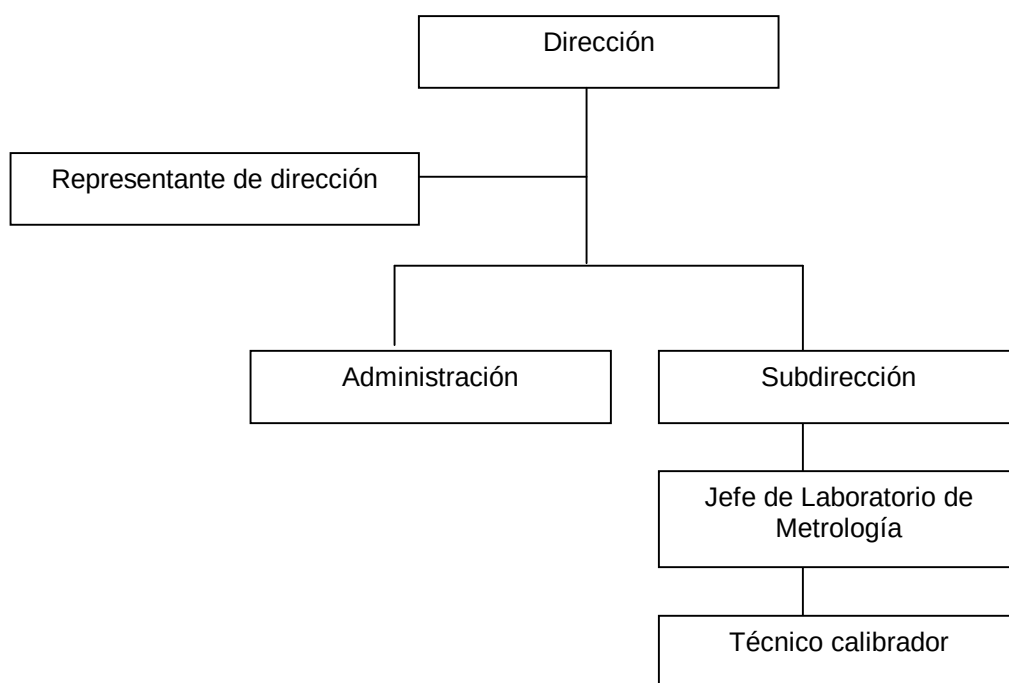


Figura 4. Estructura Organizacional del SGC del II.

4.1.1. Estructura Organizacional y descripción general de puestos.

La Dirección del Instituto de Ingeniería proporciona evidencia de su compromiso con el desarrollo e implementación del SGC, así como con la mejora continua de su eficacia: comunicando la importancia de satisfacer, tanto los requisitos del cliente como los legales y reglamentarios a través de juntas de revisión directiva; estableciendo la política de calidad y transmitiéndola en cascada; asegurando que se establecen los objetivos de la calidad y midiendo su desempeño en base a los indicadores establecidos durante las juntas de revisión directiva; llevando a cabo las revisiones trimestralmente bajo un programa y una agenda definida en donde

se registran los participantes de dichas reuniones. La evidencia son las minutas, las cuales se conservan en los archivos de SGC por 2 años. También asegurando la disponibilidad de recursos, definiéndolos y detectándolos en la ejecución del SGC.

La Dirección designa al Representante de la Dirección (RD) quien, con independencia de otras responsabilidades, tiene la responsabilidad y autoridad que incluye: asegurarse de que se establecen, implementan y mantienen los procesos necesarios para el SGC, informar a la alta dirección sobre el desempeño del SGC y de cualquier necesidad de mejora, asegurarse de que se promueva la toma de conciencia de los requisitos del cliente en todos los niveles, relaciones con partes externas sobre asuntos relacionados con el SGC.

En la investigación se aplicaron cuestionarios, al personal que participa con el SGC del II para identificar la forma en la cual se esta llevando dicho sistema.

4.1.2. Descripción de las entrevistas de los roles.

El personal y departamentos involucrados en el SGC del II participaba en el 2005 de la siguiente forma:

Rol: director.

El director es la cabeza del sistema, apoyado por el RD en la parte operativa. Se asegura que el sistema este funcionando, que los procedimientos se actualicen, que se lleven a cabo de acuerdo a cómo se estableció en el sistema documentado.

La dirección realiza lo siguiente: establecer una política de calidad, objetivos de calidad los cuales son medibles a través de indicadores, nombra al RD, revisa el sistema con cierta periodicidad, en el instituto se revisa 3 veces por año, se le llama revisión o juntas directivas, en las reuniones se revisa que la política siga siendo adecuados, se revisan necesidades de presupuesto, programas de mejora

continua, avances de mejora continua, resultados de auditorias, acciones preventivas y se le da seguimiento.

En las auditorias se hace un plan de auditoria, debe de haber un auditor líder y un grupo de auditores. En una auditoria interna se verifica punto a punto de la norma y se revisa evidencia objetiva de todos los registros, de todos los procedimientos, de todo lo que se dice que forma el sistema; y ahí se trata de encontrar oportunidades de mejora. En una auditoria interna se detectan las fallas del sistema y se detectan oportunidades de mejora y se corrigen mediante acciones correctivas. Se ponen fechas, responsable se identifica cual fue la causa raíz, y luego se realiza esa acción para corregir esa no conformidad. Cabe mencionar que los auditores internos es personal del instituto, pero la norma establece que un auditor no puede auditar su propia área, para ser auditor se debe tener experiencia en la norma, o preferentemente debe haberse formado como auditor, ser una persona competente.

Rol: representante de dirección.

Es responsable de las siguientes actividades. Coordinar los esfuerzos necesarios que se establecen en el manual de calidad para poder cumplir con la meta y política de calidad. Tener conocimiento de las modificaciones realizadas a los registros e instrucciones de trabajo. Guardar respaldos de esa información y enviar información de ello a la dirección, a los auditores internos y externos cuando sea requerido por ellos. También programa las auditorias internas y externas, así como la parte de la coordinación de la auditoria o seguimiento de ésta. Otra actividad que realiza es la programación de capacitación, entrenamiento y auditoria. Información con los proveedores de productos sobre las especificaciones de los servicios o de los equipos.

La información que se necesita para realizar estas actividades son: se sigue un procedimiento de control de documentos especificando la manera de llevar el control, tomándose como una especificación del manual de calidad, se generan archivos que se respaldan con copias (documentos escritos).

El RD tiene copia de todos los procedimientos, instrumentos de trabajo, registros y formatos que se requieran para el sistema, también las diferentes áreas involucradas cuentan con ella, tienen los documentos requeridos en cada caso. En general el RD respalda la información del SGC, por escrito, carpeta y electrónicamente.

Las solicitudes de servicio y de compra, se encuentran también en archivos electrónicos. Cada área que genere registros debe de guardarla.

Se tienen archivos digitalizados en hojas de cálculo específicamente en Excel , sobre información de algunas verificaciones que se realizan diarias de instrumentos de medición, lo cual permite asegurar si están dentro o fuera de las especificaciones. Esta información se imprime y se tiene como documento escrito, no se considera como documento electrónico; pero se necesita de algunos programas para poder generar información como gráficas de resultados, indicadores, quejas, equipos que se calibren, solicitudes.

El control de información requerida se lleva a cabo de la siguiente forma: Mantener la actualizaciones de las revisiones si se presentan cambios, actualizarlos y los anteriores, marcarlos como obsoletos, además de imprimir. Como son varios pasos en ocasiones se omiten algunos.

El programa que se tiene para trabajar en prueba es poderoso pero no se está utilizando completamente, se necesita de un sistema que permita un reporte o aviso automático de los próximos trabajos a realizar por ejemplo de mandar equipo, hacer el mantenimiento, reporte de clientes para continuar con la atención de los clientes, proveedores.

Rol: Administrador.

La participación en el SGC es para realizar principalmente la planificación, comunicación, ejecución y control de lo procesos, compra de equipo y contratación de servicios que requiera el laboratorio de metrología. Esto se realiza mediante un

procedimiento que incluye entre otras cosas que el producto a comprar cumpla con los requisitos especificados, estos puntos se encuentran establecidos en el procedimiento de compras y evaluación de proveedores (GC-PC-007). La información que se necesita para realizar estas actividades son: estipular lo que se requiere, ya sea la compra de un equipo, contratación de algún servicio, con sus especificaciones, cantidades, marcas, requerimientos que ellos ocupan, tiempo de entrega, etc.

Para la adquisición de estos servicios la UABC cuenta con un reglamento de adquisición, el cual indica cuando se va a licitar de acuerdo a montos, cuando invitar a 3 proveedores, y cuando se va a hacer la compra directa. El control de información requerida y generada para estas actividades se hace a través de un formato "solicitud de compra" y ahí se indica si es un servicio o la compra de un equipo, si se gestiona a través del ejercicio del gasto o mediante oficio, según sea el caso, ante el departamento de servicio administrativo.

Rol: Técnico calibrador.

Es responsable de las siguientes actividades. Procedimientos de calibración que proporcionara el fabricante del equipo que va a calibrar, dentro del manual del servicio o equipo a través de las páginas por Internet, actualización de los programas vía Internet. Estar al pendiente del mantenimiento del mismo y de la programación de mantenimiento y calibración de acuerdo a la fecha programada. Tener la responsabilidad con el cliente y con los proveedores y reportar con su jefe inmediato.

El rol que desempeña cada uno de los integrantes del SCG, se puede visualizar de una forma mas clara en la Tabla 2. En esta matriz de responsabilidades se presenta la descripción de las actividades mas importantes del SGC y el o los responsables de las actividades, así como soporte de otro integrante.

Descripción	Dirección	Subdirección	Administración	Jefe de laboratorio	Técnico	Gestión de calidad
Requisitos de la documentación	O	O	O	O	O	X
Control de documentos	O	O	O	O	O	X
Control de los registros	O	O	O	O	O	X
Compromiso de la dirección	X	O				O
Planificación del SGC	O	O	X	X	X	X
Responsabilidad, autoridad y comunicación	X	X	O	O	O	O
Comunicación interna	X	X	X	X	X	X
Revisión por la dirección	X	X	O	O	O	X
Gestión de los recursos	X	X	O	O		
Recursos humanos	O	O	O	X	O	X
Infraestructura	X	X	X	X		
Ambiente de trabajo	O	O		X	O	
Planificación de la realización del servicio/producto				X	X	
Procesos relacionados con el cliente	X	X	O	X		
Diseño y desarrollo	NO APLICA					
Compras	O	O	X	O		O
Verificación de los productos comprados			X	O		
Control de la prestación del servicio			X	O		
Identificación y trazabilidad			X	X		
Propiedad del cliente			X	X		
Preservación del producto			X	X		
Control de calibración			O	X	O	X
Satisfacción del cliente	O	O	O	O	O	X
Auditoría interna	O	O	O	O	O	X
Seguimiento y medición de los procesos				X	O	X
Control del producto no conforme				X	O	X
Análisis de datos	X	X	X	X		X
Mejora continua	X	X	X	X	O	X
Acción correctiva	X	X	X	X	O	X
Acción preventiva	X	X	X	X	O	X

X = Responsabilidad principal

O = Soporte

Tabla 2. Matriz de Responsabilidades.

4.2. Análisis del Sistema de Gestión de Calidad.

Para que un elemento de la organización funcione de manera eficaz, tiene que identificar y gestionar numerosas actividades relacionadas entre sí. Una actividad que utiliza recursos, y que se gestiona con el fin de permitir que las entradas se

transformen en resultados, se puede considerar como un proceso. En la Figura 5 se observa, el enfoque sistémico, del SGC.

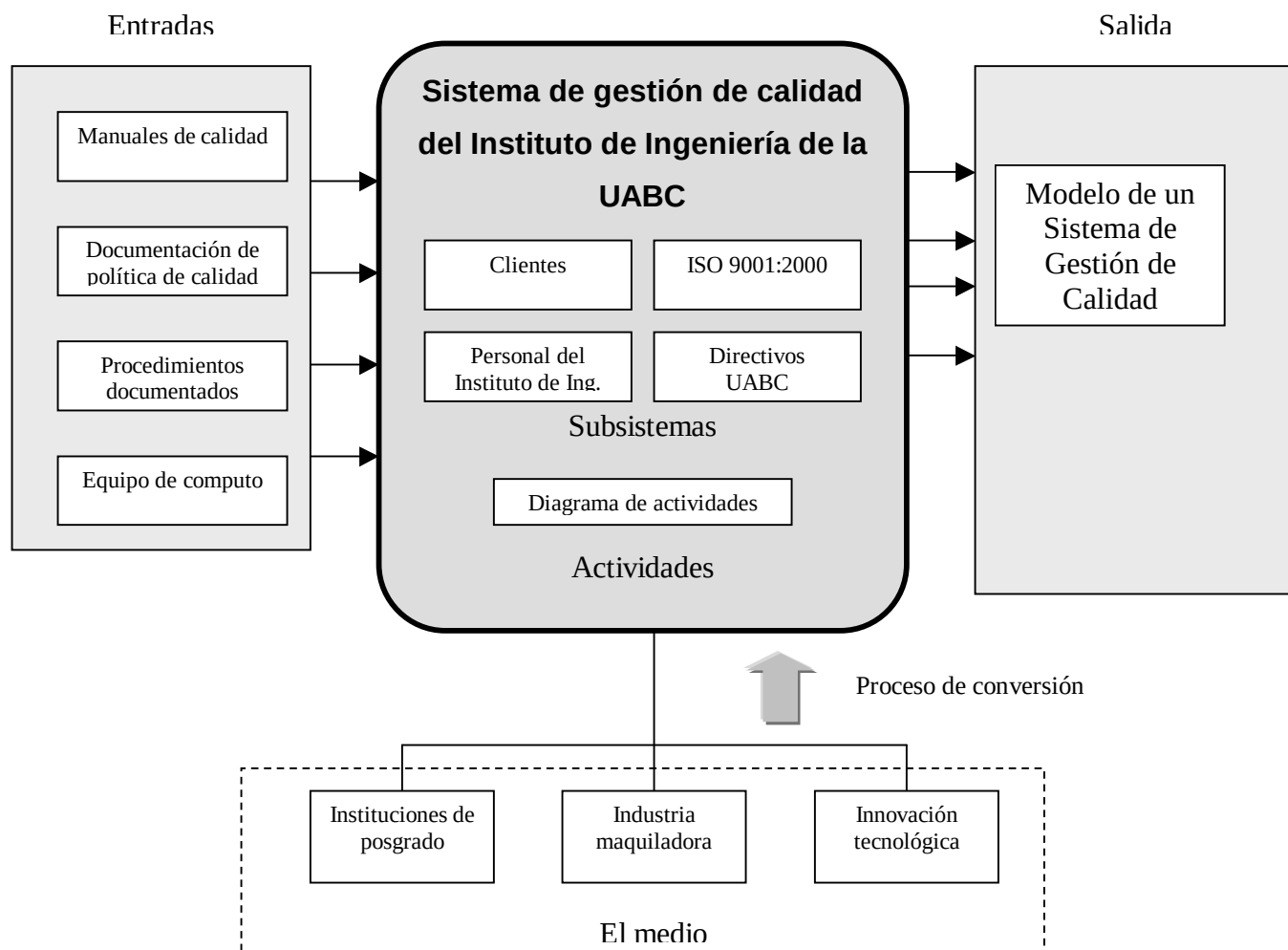


Figura 5. Enfoque sistémico del SGC del II.

El sistema de gestión desarrollado, implantado y mantenido el II está basado en el cumplimiento de los requisitos de la norma ISO 9001:2000, y en la orientación a la mejora continua del desempeño. El sistema se basa en la planificación y mejora continua de la eficacia y eficiencia del desempeño.

En cuanto a la documentación, el sistema de gestión, tiene la estructura que se muestra en la Figura 6.



Figura 6. Elementos de la Documentación.

Manual de Calidad. Establece la política de calidad y describe el Sistema de Gestión de Calidad y sus elementos. En el caso de estudio, la política de calidad tiene el siguiente lema: "Nuestro Compromiso es ofrecer servicios de ingeniería que satisfagan las exigencias de calidad que nuestros clientes requieren, en un marco de mejora continua". Se incluye también el alcance de los procesos identificados en el diagrama de interacción para el Laboratorio de Metrología. Los servicios particulares que ofrece el laboratorio se encuentran descritos en los procedimientos e instrucciones de trabajo de los mismos. En el manual de calidad del SGC del II incluye los siguientes puntos que se muestran en la Tabla 3.

No.	Nombre del proceso
1	Tabla de Cambios.
2	Alcance.
3	Términos y definiciones.
4	Sistema de Gestión de la Calidad.
5	Responsabilidad de la Dirección.
6	Gestión de los Recursos.
7	Realización del Servicio.
8	Medición, análisis y mejora.

Tabla 3. Contenido del Manual de Calidad.

Procedimientos escritos. Los servicios particulares que ofrece el laboratorio se encuentran descritos en los procedimientos e instrucciones de trabajo de los mismos. Describen la forma en que deberán realizarse algunas actividades, es decir se explica: que se hace, donde se aplica, cuando se aplica, quien es el responsable y porque se hace. El SGC del II tiene en los procedimientos escritos de las siguientes actividades que se muestran en la Tabla 4.

NOMBRE	OBJETIVO DEL PROCEDIMIENTO
Control de documentos.	Establecer un método para administrar el control de documentos del SGC.
Control de Registros.	Establecer un método que cumpla con los requisitos mínimos necesarios para un buen control de evidencia requerida por el SGC.
Entrenamiento.	Lograr la competencia del personal basándose en la educación, formación, habilidades y experiencia apropiadas, mediante el establecimiento de un procedimiento de capacitación acorde a la normatividad del SGC.
Auditoria Interna.	Realizar auditorias internas para verificar la conformidad del SGC con la norma internacional ISO 9001:2000.
Acciones Correctivas.	Eliminar la causa de no conformidades de la manera mas apropiada para prevenir que vuelva a ocurrir.
Acción Preventiva.	Establecer un método para determinar las posibles no conformidades y evitar su ocurrencia reforzando de esta forma la política de mejora continua de la organización.
Compra y evaluación de proveedores.	Definir el procedimiento para solicitar la compra de equipos, accesorios, subcontratación de servicios.

Tabla 4. Procedimientos Escritos del SGC del II.

Instrucciones de trabajo. Estas son las instrucciones paso a paso que los empleados siguen para llenar los requerimientos de cada trabajo. Son los pasos que deben ser seguidos para realizar la operación correctamente. Las instrucciones detallan el “cómo” se hacen las operaciones.

Evidencia documentada. Las formas y registros son evidencia documentada de que una operación está acorde con el sistema establecido por los primeros tres niveles de documentación. Por ejemplo los empleados llenan una lista de chequeo para indicar que han realizado una operación de acuerdo a las instrucciones documentadas.

En el SGC del II, la mejora continua en el laboratorio de metrología, es parte esencial. Los principales procesos que se realizan en el laboratorio se muestran en la Figura 7. En dicha figura se presenta la interacción de los procesos.

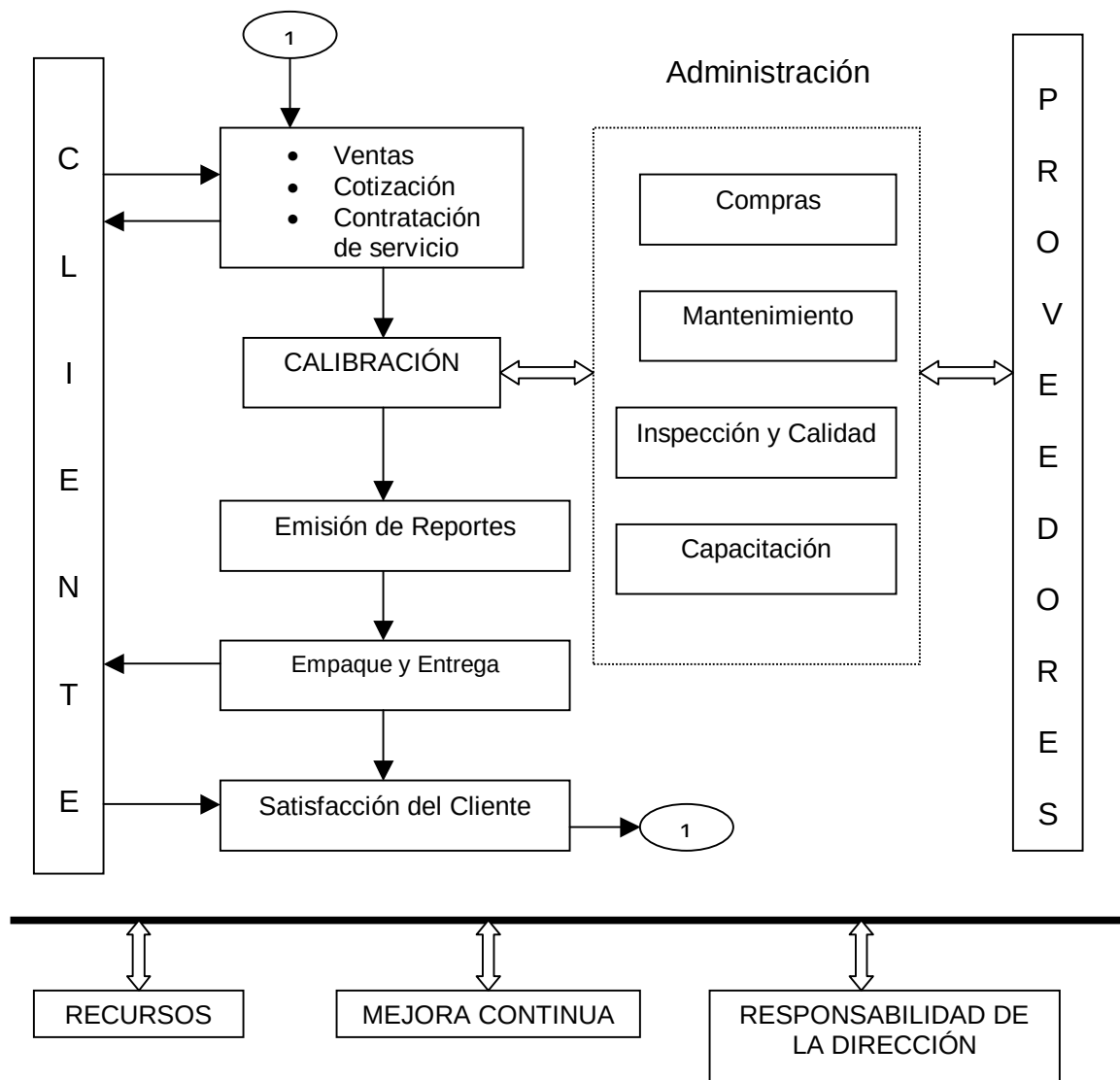


Figura 7. Diagrama de Interacción de Procesos del laboratorio de Metrología.

Capítulo V. Resultados obtenidos.

5.1. Modelos obtenidos.

El objetivo general de este trabajo fue desarrollar un modelo para el SGC ISO 9001:2000 como un proyecto de software utilizando UML. Los resultados se presentan en cuatro tipos de modelos:

- Modelo del SGC del II.
- Modelo de requisitos del Sistema de Información (SI).
- Modelo conceptual para el SI.
- Modelo de emplazamiento.

5.1.1 Modelos del SGC del II.

A continuación se presentarán los modelos obtenidos del SGC, en ellos se presentan los elementos más importantes, así como las actividades, que se realizan de forma general.

Un elemento importante en la estructura del SGC es el Laboratorio de Metrología. El diagrama de caso de uso de la Figura 8, muestra al actor Responsable de laboratorio, así como la interacción con servicios que ofrece el laboratorio. El laboratorio de metrología realiza los procesos de calibración de equipo, a los clientes que así lo soliciten. Al final de cada calibración se generan reportes con los datos correspondientes a esa actividad. Finalmente se empaca y se hace entrega del equipo al que se le realizó la calibración.

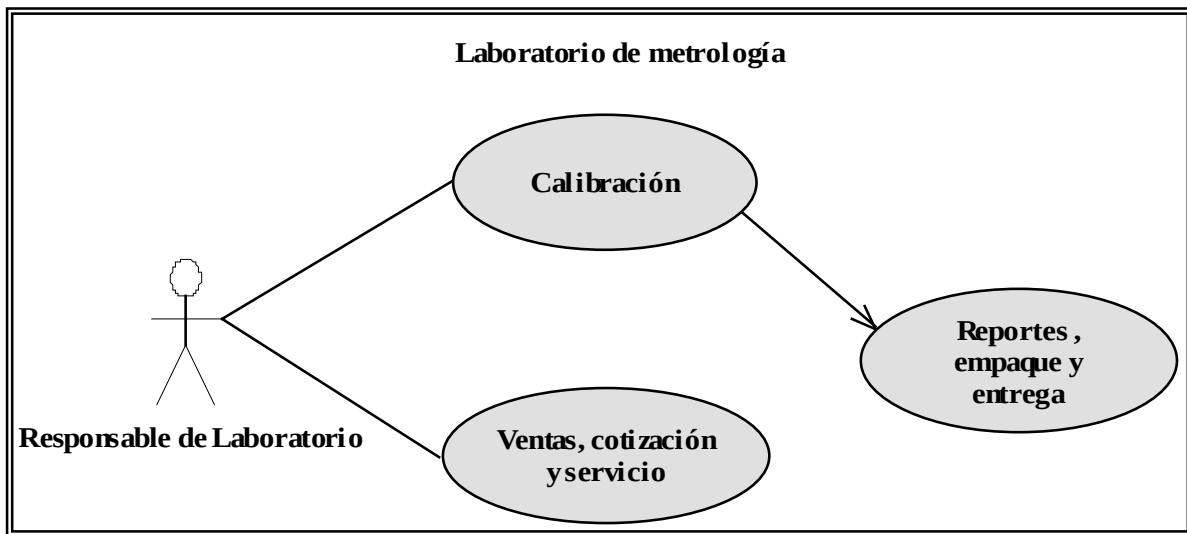


Figura 8. Diagrama de caso de uso 1 del proceso de calibración en el Laboratorio de Metrología.

En la Figura 9 se incluyen dos elementos al diagrama de caso uso del Laboratorio de Metrología, estos son: clientes y proveedores. Además se añade la interrelación con sus actividades.

Se incluye el actor Administrador como parte generadora para realizar principalmente la planificación, comunicación, ejecución y control de los procesos, compra de equipo y contratación de servicios que requiera el laboratorio de metrología.

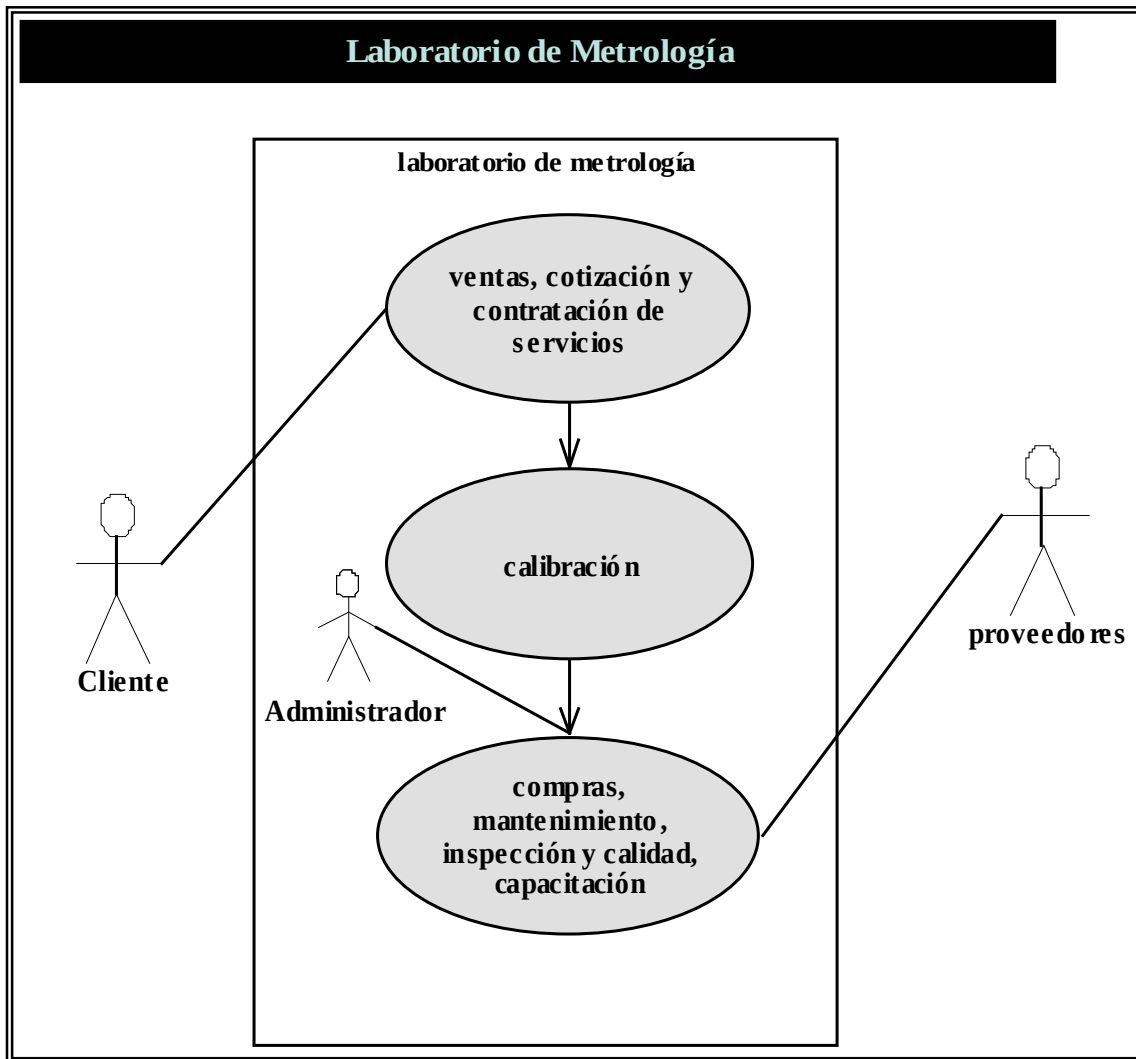


Figura 9. Diagrama de caso de uso 2 del proceso de calibración en el Laboratorio de Metrología.

Uno de los objetivos planteados en el Capítulo 1 es la de identificar los procesos de la norma ISO 9001:2000 que son viables para automatizar o documentar electrónicamente alguno de sus elementos. Estos procesos se presentan en la Tabla 5, con el nombre de la actividad correspondiente. La descripción de estas actividades se presentan en los diagramas de actividades utilizando UML, siendo este otro de los objetivos descritos en el Capítulo 1, esto es la utilización del UML.

Num. figura	Actividad
10	Diagrama de Actividades para el proceso de Alta de Documentos.
11	Diagrama de actividades para el proceso Baja de Documentos.
12	Diagrama de actividades para el proceso de Cambio de Documentos.
13	Diagrama de actividades para el control de Documento de Referencia.
14	Diagrama de actividades para el Control de Registros.
15	Diagrama de actividades para el entrenamiento.

Tabla 5. Relación de diagramas de actividades de procesos viables para automatizar.

Procedimiento para control de documentos: En este procedimiento se establece la manera de llevar el control de documentos del SGC. esto es: alta de documentos, baja de documentos, cambios de documentos y documentos de referencia. En los diagramas de actividades de las Figuras 10, 11, 12 y 13 se presentan los pasos para: la aprobación de los documentos en cuanto a su adecuación antes de su emisión. Revisar y actualizar los documentos cuando sea necesario y aprobarlos nuevamente; asegurar que se identifiquen los cambios y el estado de revisión actual de los documentos. Garantizar que las versiones pertinentes de los documentos aplicables se encuentren disponibles en los puntos de uso, a través de distribución de procedimientos aplicables a cada área. Asegurar que los documentos permanecen legibles y fácilmente identificables. Confirmar que se identifiquen los documentos de origen externo y se controle su distribución, a través del sello de documentos de referencia y listado de distribución. Prevenir el uso no intencionado de documentos obsoletos, y aplicarles una identificación adecuada en el caso de que se mantengan por cualquier razón mediante sello de documentos obsoletos.

Figura 10. Diagrama de Actividades para el proceso de Alta de Documentos.

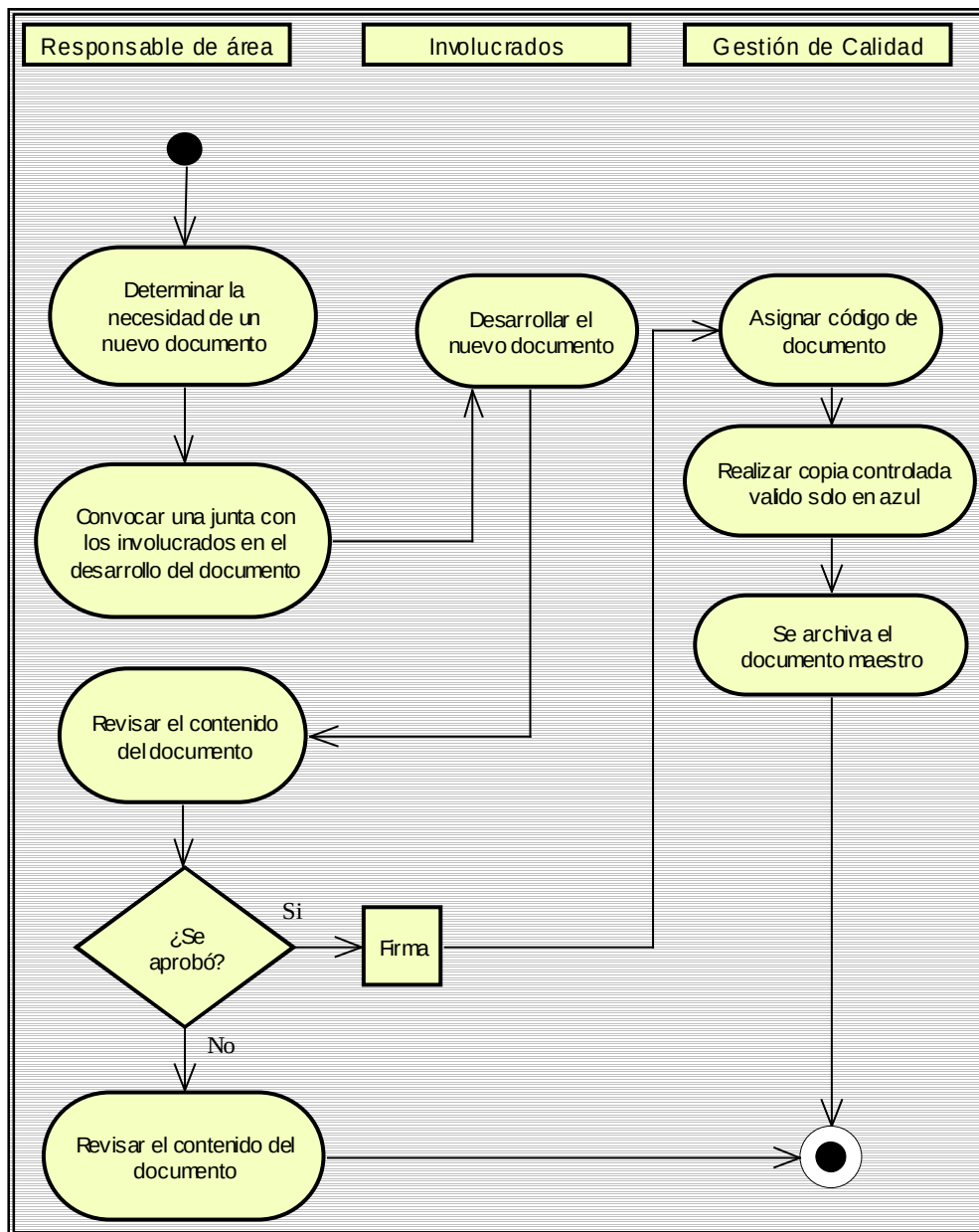
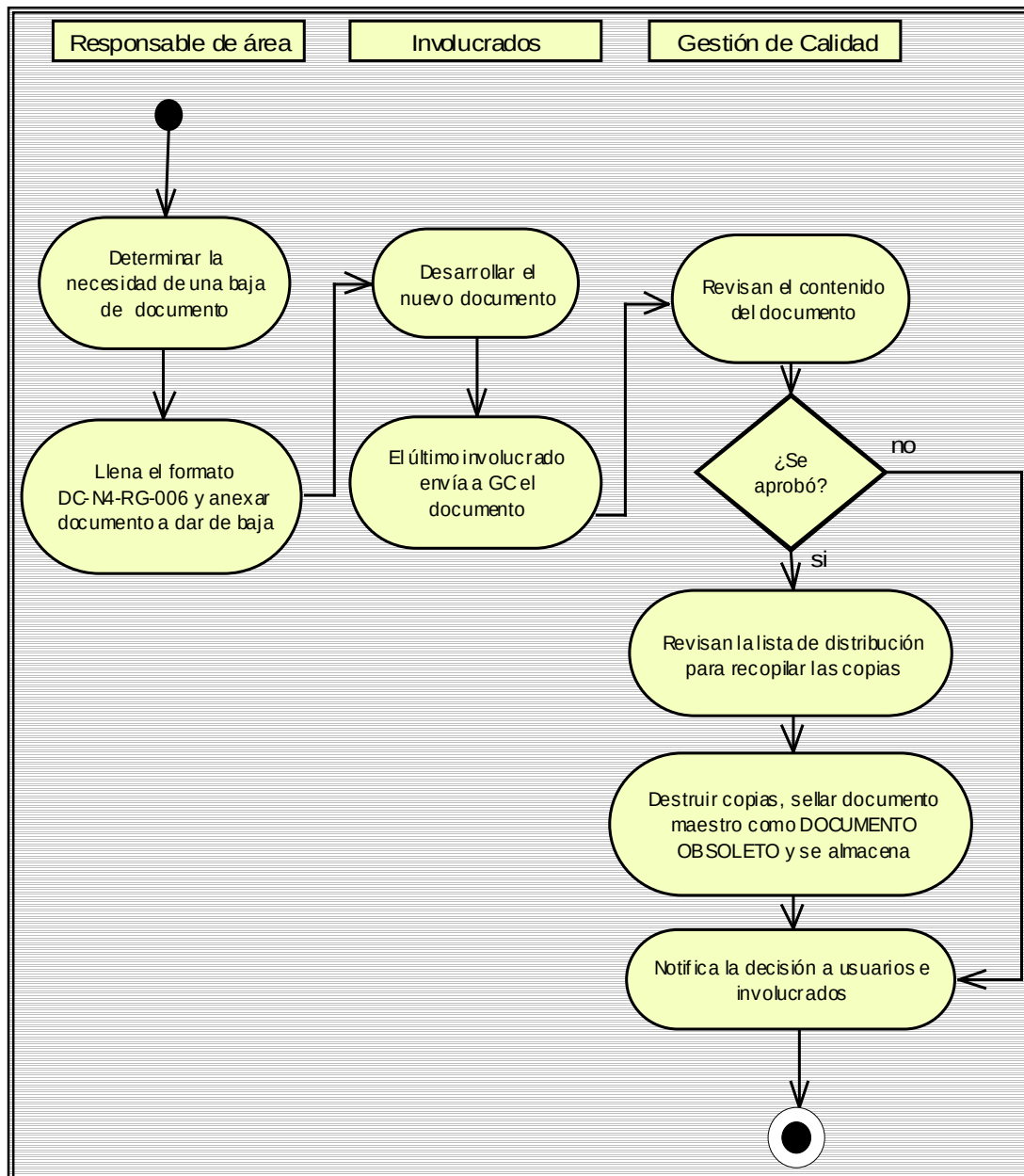


Figura 11. Diagrama de actividades para el proceso Baja de Documentos.



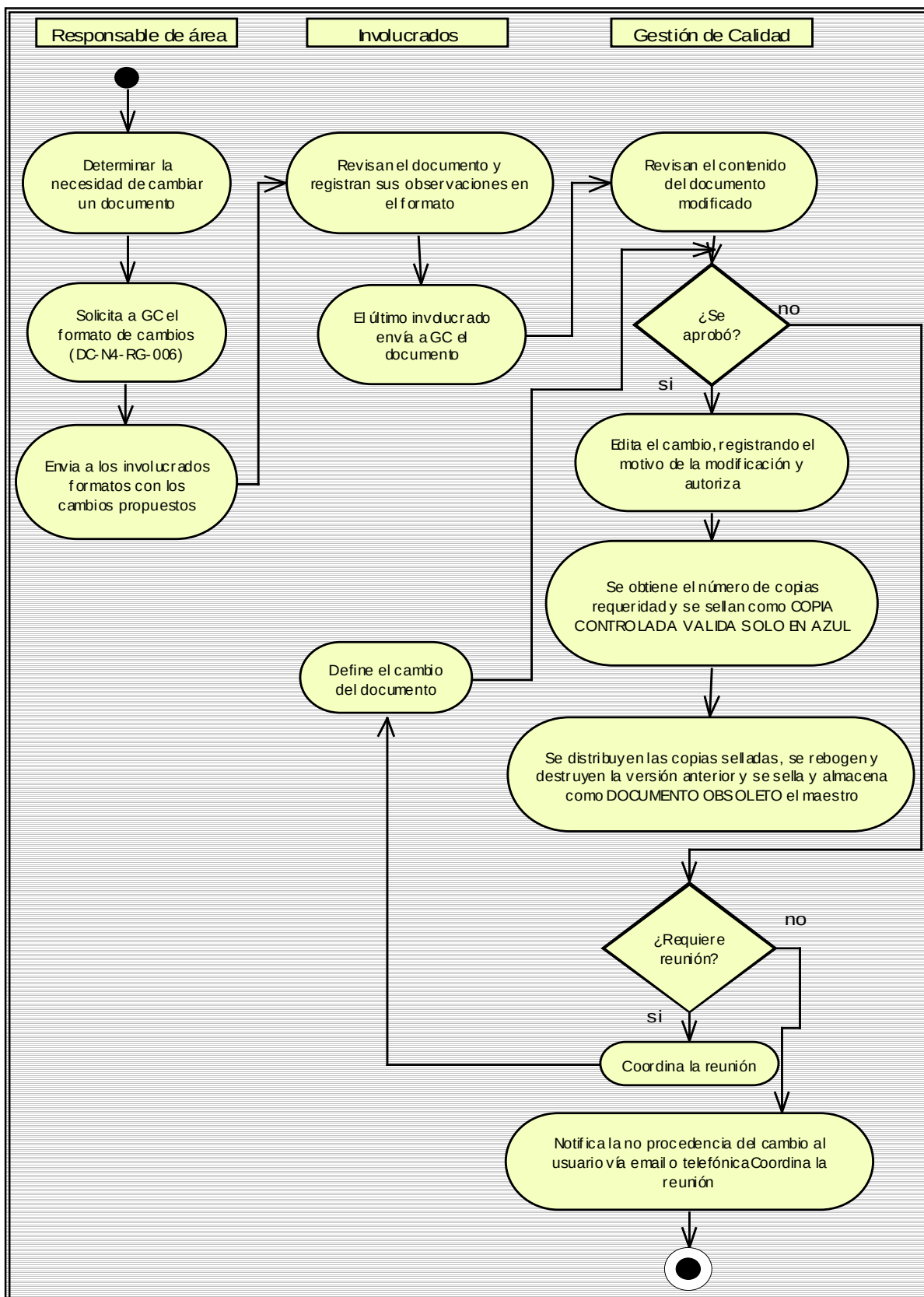
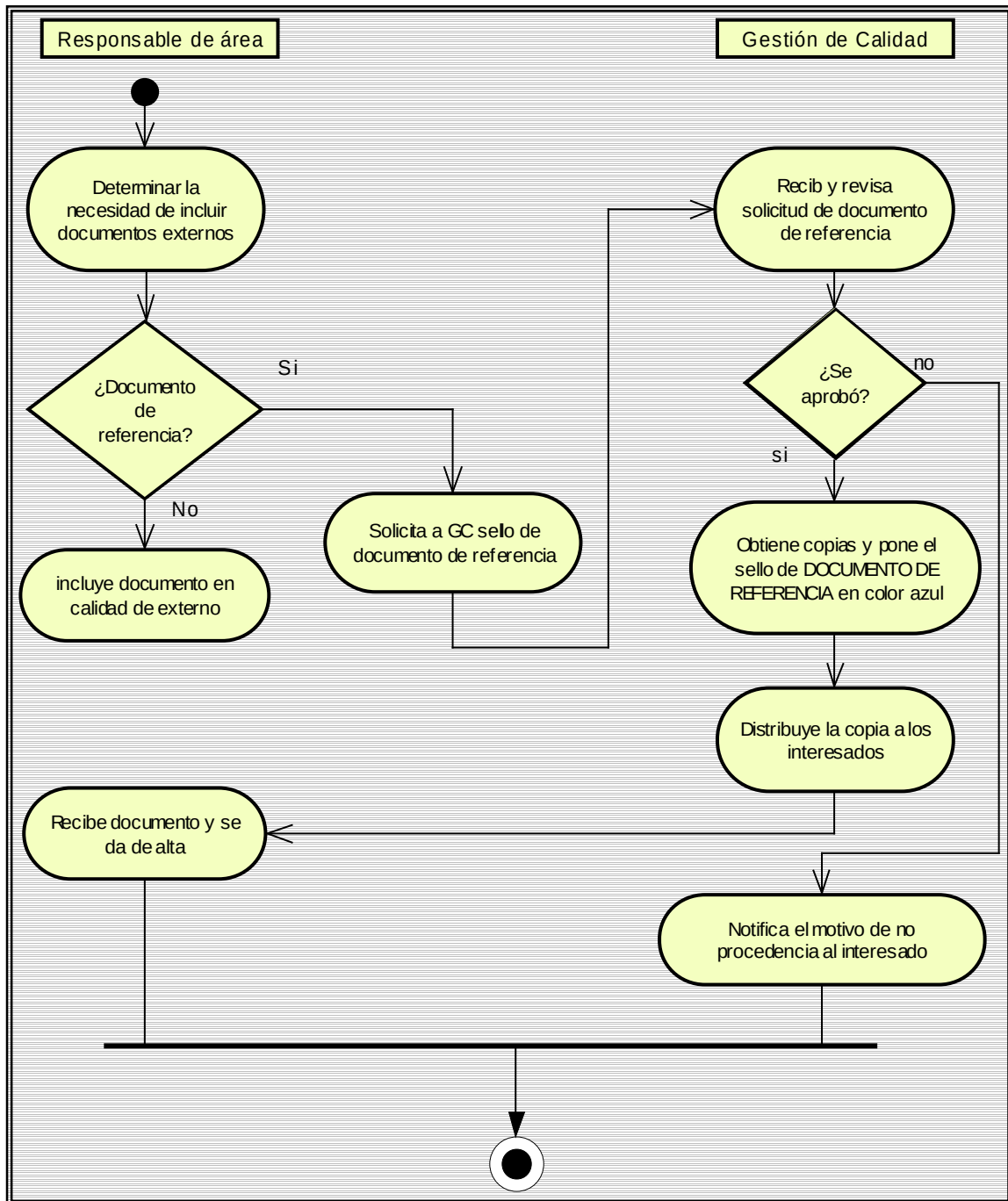


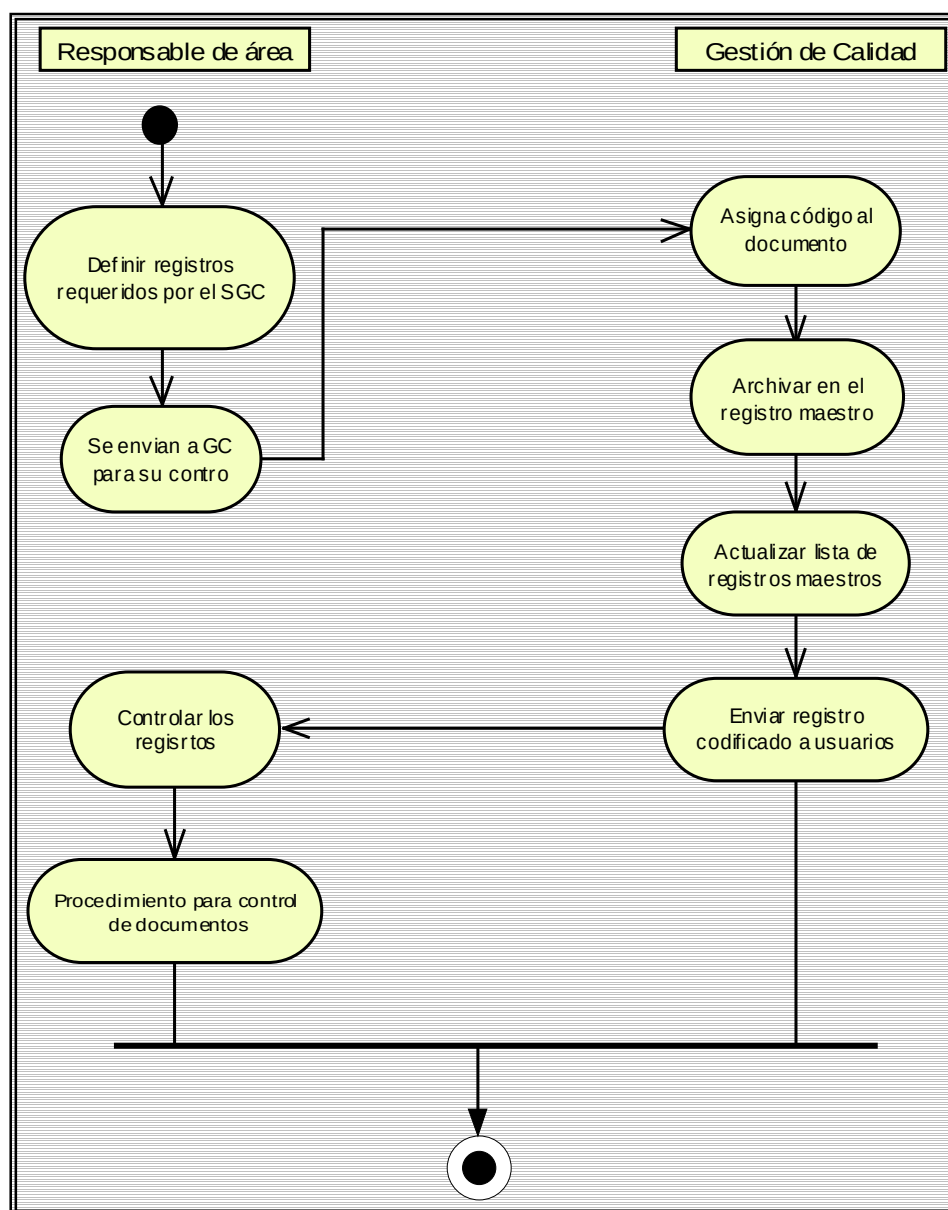
Figura 12. Diagrama de actividades para el proceso de Cambio de Documentos.

Figura 13. Diagrama de actividades para el control de Documento de Referencia.



Procedimiento para control de registros. En este procedimiento se establece un método para tener los requisitos mínimos necesarios tener un control de evidencia establecida por el SGC. En el diagrama de la Figura 14 se presentan las actividades para el control de registros. Los registros se establecen y mantienen para proporcionar evidencia de la conformidad con los requisitos, así como de la operación eficaz del SGC. Los requisitos permanecen legibles, fácilmente identificables y son recuperables. Además se tiene una tabla de registros en cada procedimiento que especifica: nombre, orden para el almacenamiento, responsable y el tiempo de retención. Transcurrido este tiempo, la documentación se revisa por GC para determinar su disposición final, la cual puede ser: eliminación de los registros, traspaso al archivo muerto del Instituto de Ingeniería por un tiempo determinado, el cual se establece en base a la importancia del documento.

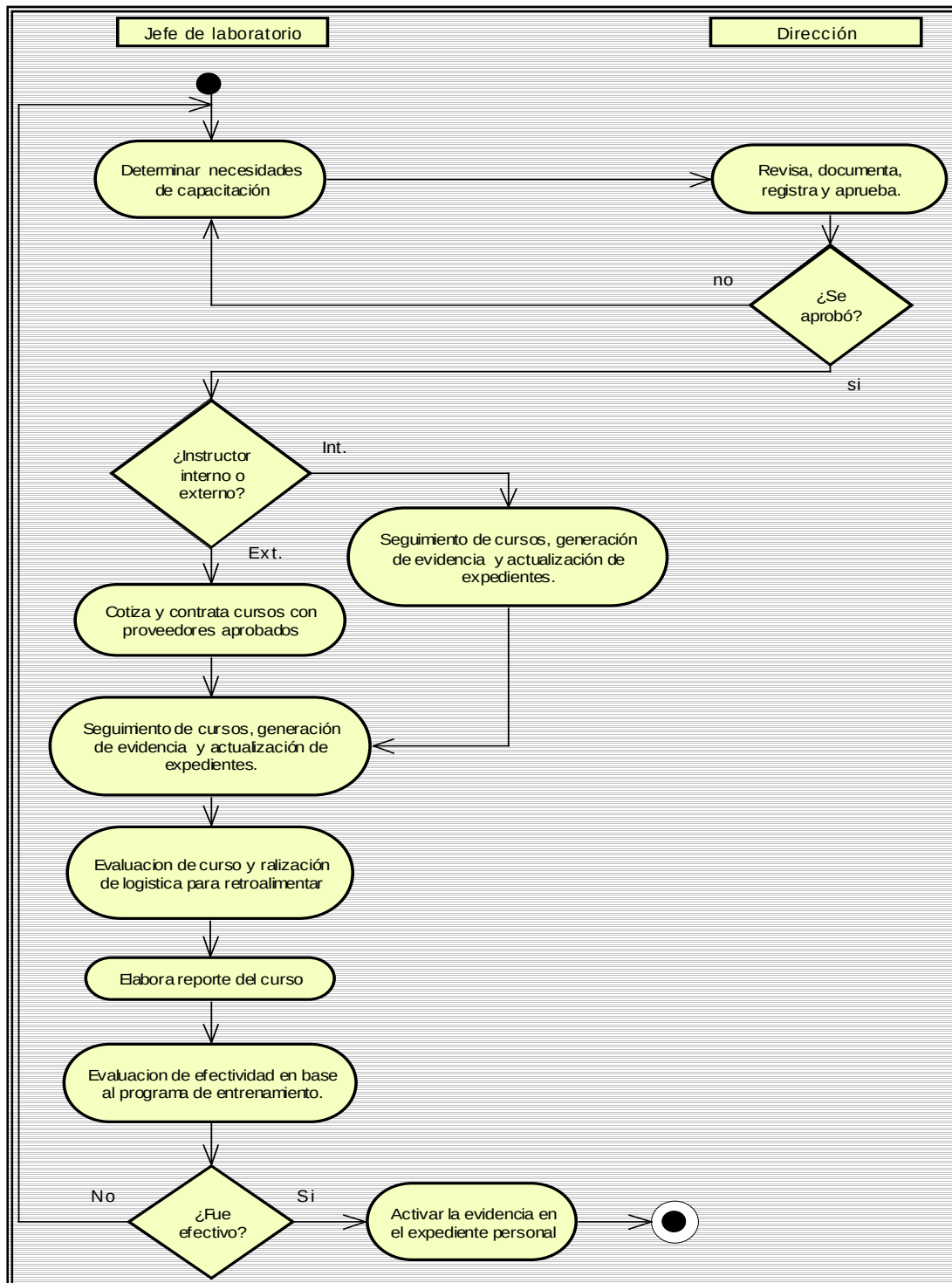
Figura 14. Diagrama de actividades para el Control de Registros.



Procedimiento de entrenamiento. Lograr la competencia del personal basándose en la educación, formación, habilidades y experiencia apropiadas, mediante el establecimiento de un procedimiento de capacitación acorde a la normatividad del SGC. En el diagrama de actividades de la Figura 15 se identifican los pasos a seguir para el proceso de capacitación. En el se genera lo siguiente: determinar

las habilidades necesarias para el personal de los laboratorios que realiza trabajos que afectan a la calidad del servicio a través de la certificación del personal en su puesto y/o habilidades. Proporcionar formación o toma otras acciones para satisfacer dichas necesidades, a través de cursos de capacitación acorde a sus funciones. Evalúa la eficacia de las acciones tomadas, por medio de la efectividad del entrenamiento, resultados del área o examen de conocimiento y el desarrollo de habilidades. Se asegura que el personal es conciente de la pertinencia e importancia de sus actividades y de cómo contribuyen al logro de los objetivos de la calidad a través de capacitación de procedimientos e instructivos de trabajo pertinentes a su área, así como darles a conocer los resultados de los indicadores en los pizarrones informativos y/o juntas. Mantiene los registros apropiados de la educación, formación, habilidades y experiencia, registros de asistencia, diplomas, reconocimientos, exámenes, currículo, etc.

Figura 15. Diagrama de actividades para el entrenamiento.



En la Figura 16, el diagrama de casos de uso muestra las distintas operaciones del SGC, así como su relación con su entorno, identificándose las actividades mas importantes y los elementos que participan.

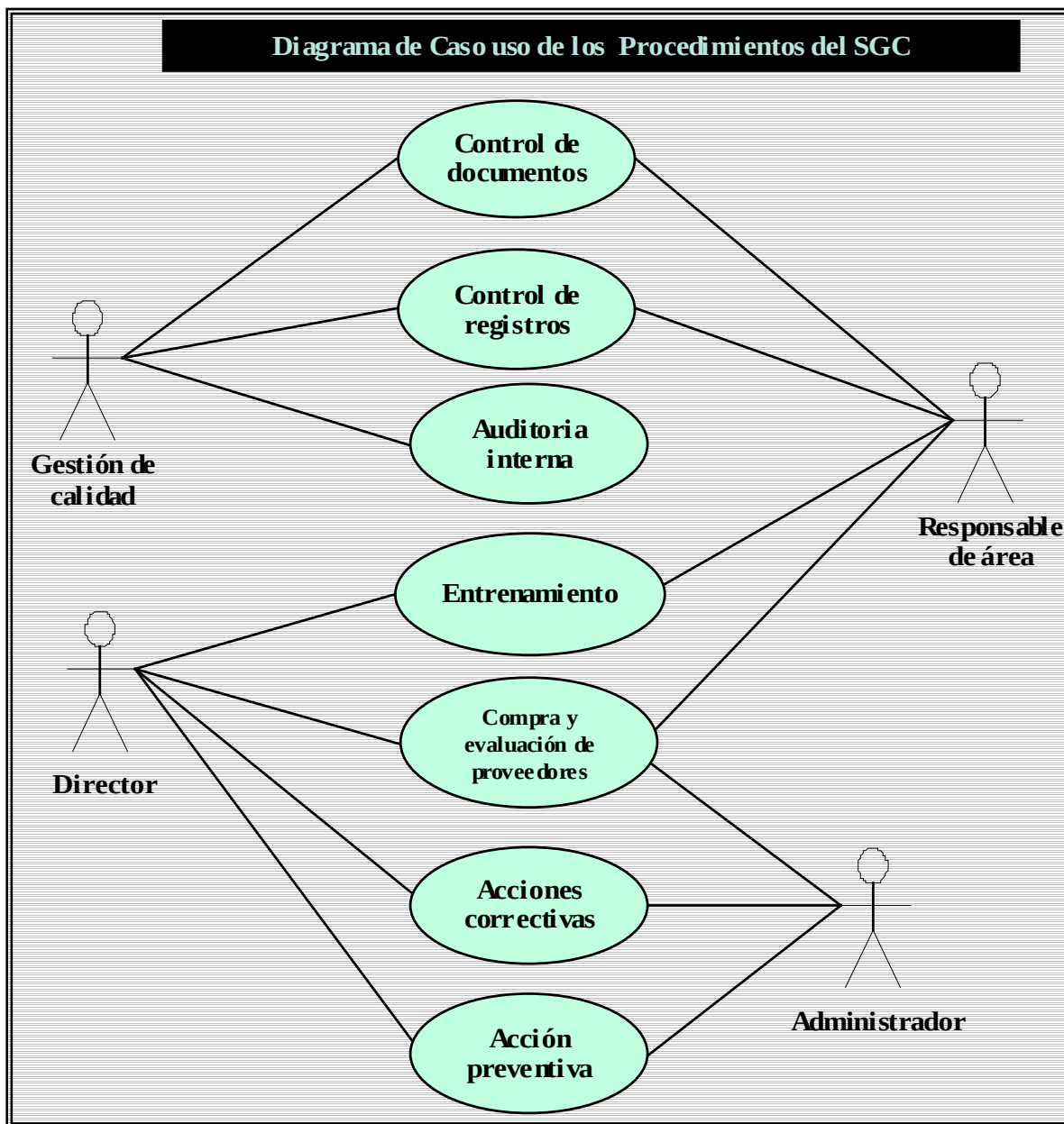


Figura 16. Diagrama de caso de uso de los procedimientos del SGC.

Para el SI, en el cual se controle la documentación susceptible de documentar de SGC, se presenta en los modelos de la siguiente sección.

5.1.2 Modelo de requisitos para el SI.

En el modelo de requisitos, se presenta una descripción de los objetos y el comportamiento externo del SI, es decir, qué debe hacer el sistema sin describir cómo debe hacerlo. Incluyéndose los aspecto funcionales, de comunicación y comportamiento a un nivel alto.

De acuerdo a la funcionalidad se presenta un estructura jerárquica en forma de árbol, esto para presentar la funcionalidad del sistema, presentándose en los niveles superiores los puntos más importantes del modelo.

1. Administrar
 - 1.1. Configuraciones
 - 1.2. Administrar documentos
 - 1.3. Documentos por usuarios
2. Archivos
 - 2.1. Usuarios
 - 2.2. Áreas
 - 2.3. Cargos
3. Documentar
 - 3.1. Documentos vigentes
 - 3.2. Documentos en edición
 - 3.3. Documentos obsoletos
4. Controlar
 - 4.1. Control de registros de calidad

5.1.2.1 Diagramas de casos de uso.

En esta sección, las funciones del sistema contenidas en la estructura jerárquica en forma de árbol, agrupándose según la organización de los paquetes y construyendo diagramas de casos de uso, en los cuales surgen relaciones entre

los casos de uso, actores, relaciones entre actores, relaciones entre casos de uso y actores, etc.

Módulos del SI: se representan los cuatro elementos en las que se han distribuido las funciones, las cuales son: Administrar, Archivos, Documentar y Controlar.

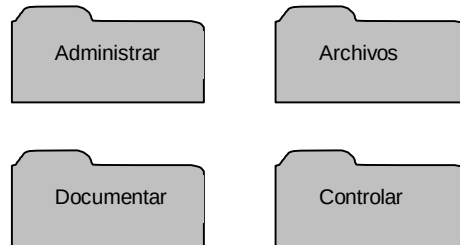


Figura 17. Módulos del SI.

1. **Administrar:** en módulo se incluyen las opciones que pertenecen al proceso administrativo del sistema. Configuraciones, Administrar documentos y Documentos por usuario.

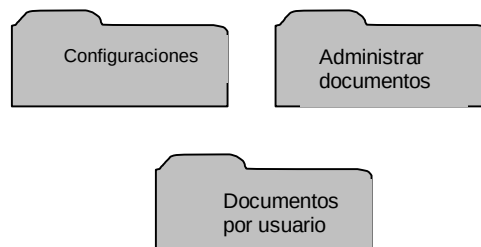


Figura 18. Módulo Administrar dentro del SI.

1.1. Configuraciones: Figura 19, en este módulo se realizan las configuraciones del sistema. Como son: avisos, usuario, seguridad, código para documentos.

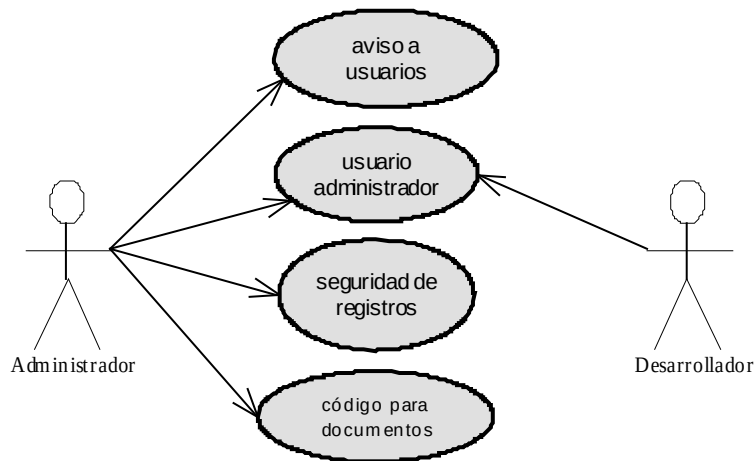


Figura 19. Diagrama de Caso de Uso para la Configuración.

1.2. Administrar documentos: crear un nuevo documento, pasar a edición un documento existente, reasignar responsables a un documento en edición, eliminar documento de edición y editar documento según solicitud de cambio.

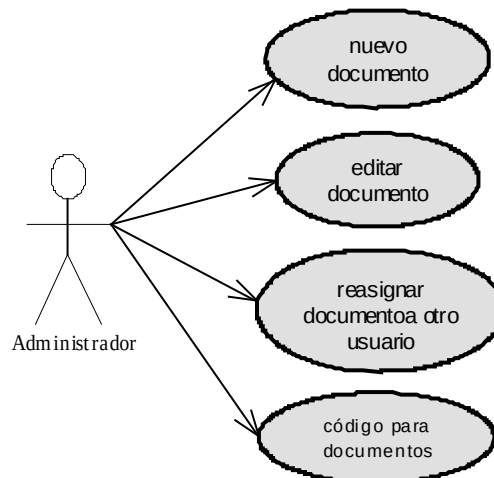


Figura 20. Diagrama de Caso de Uso para Administrar documentos.

1.3. Documentos por usuario: Figura 21, asignar permisos a los usuarios para acceder a los documentos.

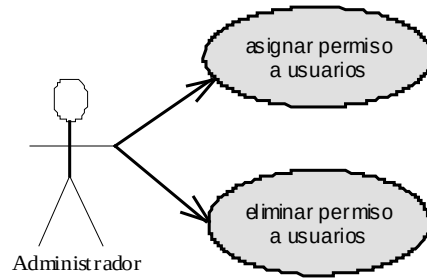


Figura 21. Diagrama de Caso de Uso, documentos por usuario.

2. **Archivos:** este modulo permite el mantenimiento de la información de las tablas básicas que utiliza el sistema para su funcionamiento, estos archivos son: usuarios, documentos y cargos.

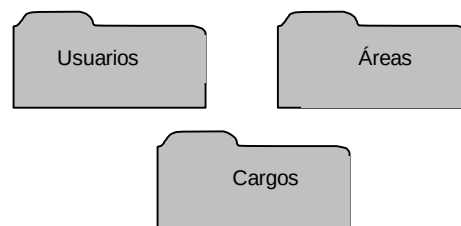


Figura 22. Módulo de mantenimiento a Archivos.

2.1. Usuarios: registrar a los usuarios que puedan acceder al sistema. básicamente es el mantenimiento de la información del archivo de usuarios.

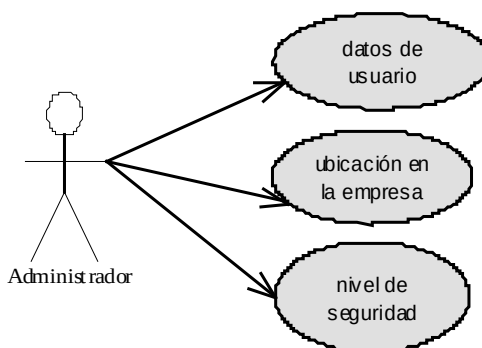


Figura 23. Diagrama de Caso de Uso a archivo de Usuarios.

2.2. Áreas: Figura 24, registra los códigos y nombres de áreas o departamentos que estén involucrados en el SGC.

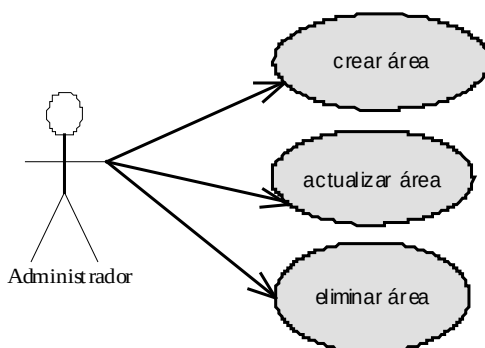


Figura 24. Diagrama de Caso de Uso a archivo de áreas o departamento.

2.3. Cargos: registra los códigos y nombres de los puestos o cargos que ocupan los usuarios.

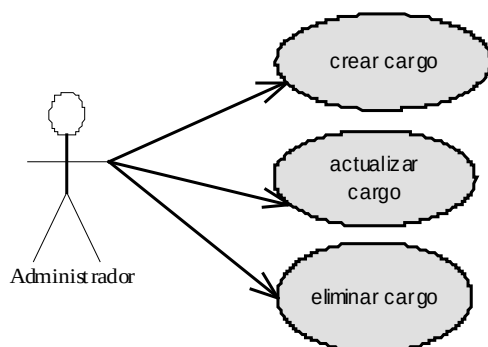


Figura 25. Diagrama de Caso de Uso a archivo de puesto o cargo.

3. **Documentar:** Figura 26, es el modulo mas importante, ya que se incluye la parte de la documentación, encontrándose los estados de los documentos. Comprende como modificar, consultar los documentos del sistema de calidad así como los diferentes estado en los que se pueden encontrar estos son: vigentes, en edición, obsoletos y solicitud de cambio.

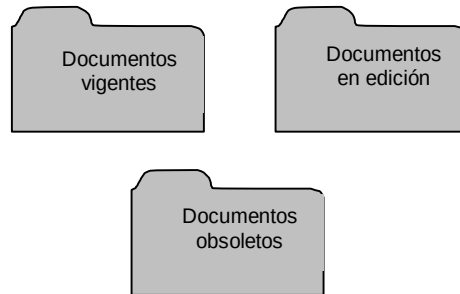


Figura 26 Módulo documentar.

3.1. Documentos vigentes: en esta tabla están los documentos que han seguido todo el proceso de documentación. (elaboración, revisión y aprobación). Estos documentos no pueden ser modificados en absoluto. Solo por el administrador o usuario habilitado.

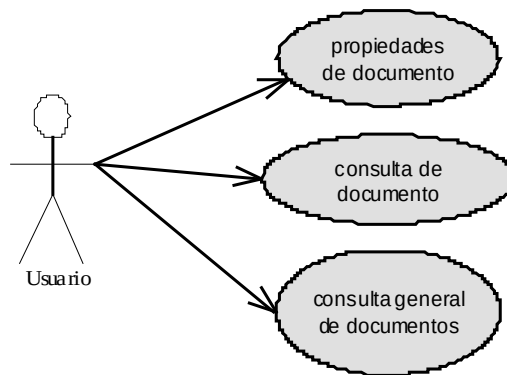


Figura 27. Diagrama de Caso de Uso de documentos vigentes.

3.2. Documentos en edición: Figura 28, estos documentos son los que se encuentran en proceso de elaboración, revisión o aprobación. Si el estado del documento es en edición, solo el usuarios responsable es quien puede editarlo.

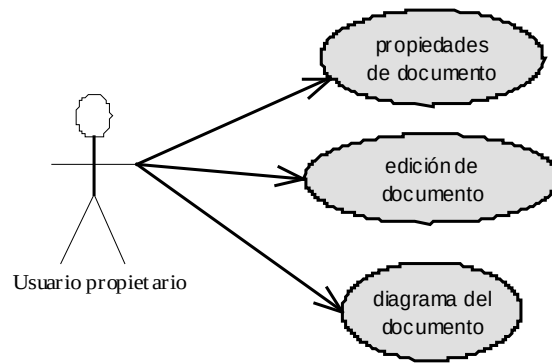


Figura 28. Diagrama de Caso de Uso de documentos en edición.

3.3. Documentos obsoletos: son los documentos que tienen una nueva versión y por lo tanto se le denomina obsoleto.

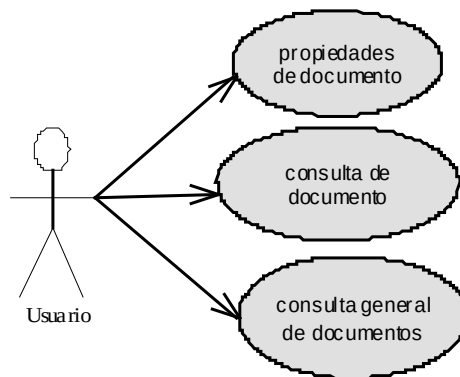


Figura 29. Diagrama de Caso de Uso de documentos obsoletos.

4. **Controlar:** en este modulo se incluyen: Control de registros de calidad y Recolección de registros. Básicamente se presenta como se configuran los registros dentro de el sistema.



Figura 30. Módulo Controlar.

4.1. Control de registros de calidad: Figura 31, crear registros incluyéndose lo siguiente: documento al cual pertenece, responsable de ese documento.

Eliminación de registros, actividad realizada por el administrador. También se incluye una vista global de los registros del sistema de calidad, además la consulta global de los registros puede ser elegida por criterios.

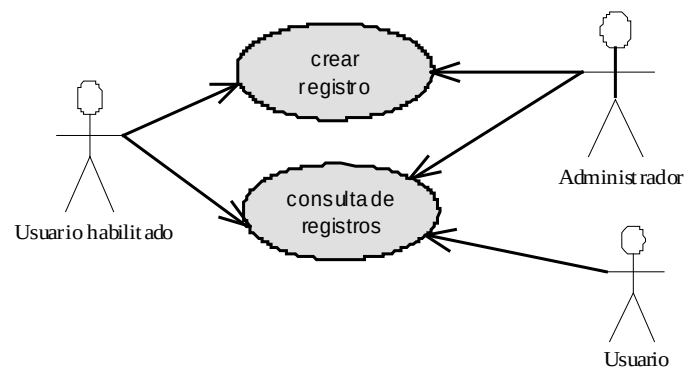


Figura 31. Diagrama de Caso de Uso del control de registros de calidad.

5.1.2.2. Diagramas de secuencia.

Continuando con el orden presentado en los diagramas caso de uso, se utilizan ahora los diagramas de secuencia para identificar y asigna responsabilidades a cada uno de los usuarios del sistema.

- 1.1.1. **Aviso a usuarios:** En esta parte se configura cómo notificará el sistema a los usuarios cuando haya un cambio en la documentación. Por defecto notifica a los usuarios de los cambios en el sistema cuando cada usuario ingrese a este.

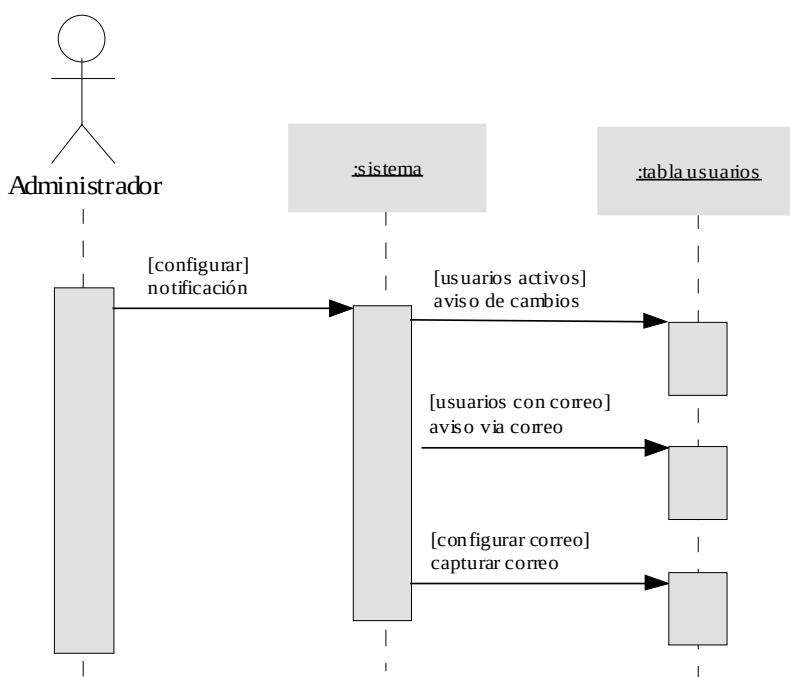


Figura 32. Diagrama de Secuencia para aviso a usuarios.

1.1.2. **Usuario administrador:** En esta parte se configura quien de todos los usuarios es el administrador, adicionalmente se configura la clave de emergencia.

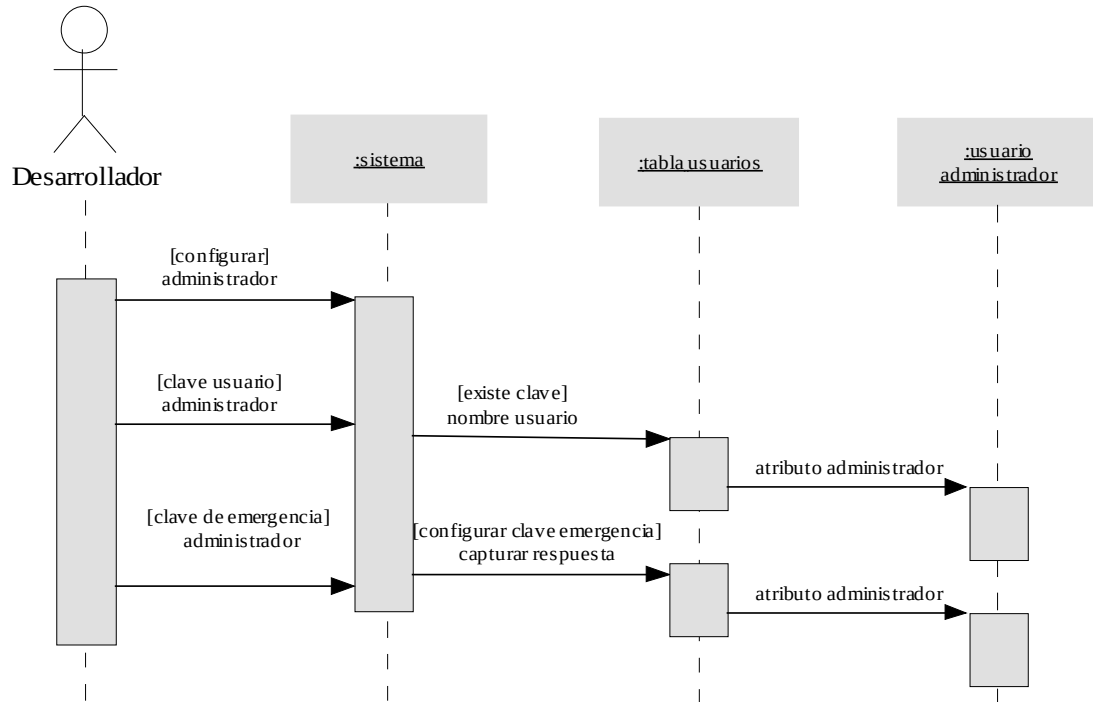


Figura 33. Diagrama de Secuencia para crear a Usuario administrador.

1.1.3. **Seguridad de registros:** Figura 34. En esta parte se configura si los usuarios pueden o no crear registros, estando a la par con el nivel de acceso que tiene cada usuario a los documentos.

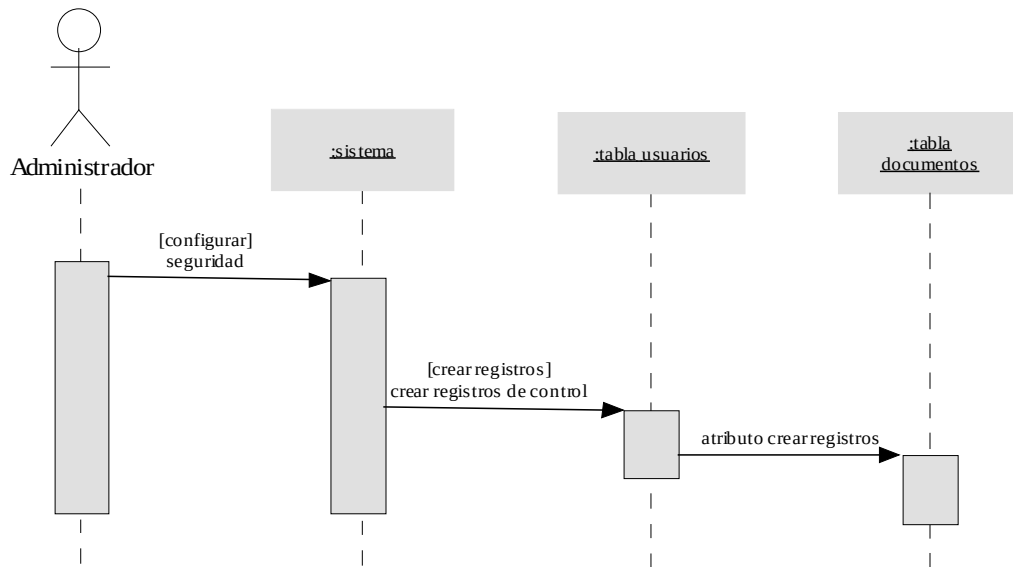


Figura 34. Diagrama de Secuencia para proceso de seguridad de registros.

1.1.4. Código para documentos: En esta parte se configura la longitud de los números de dígitos de los que se compone el código de un documento. Estos son: longitud código área, longitud código producto, longitud consecutivo documento.

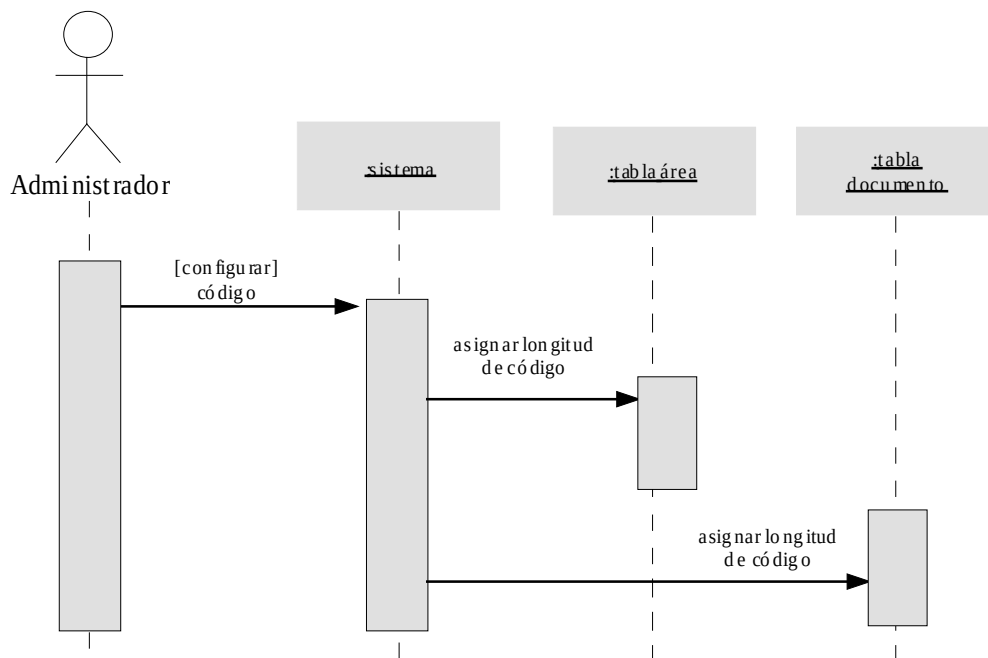


Figura 35. Diagrama de Secuencia para proceso de asignación de Código para documentos.

1.2.1. Nuevo documento: En la creación del nuevo documento se captura el código del documento, el responsable del documento, fecha inicial y fecha tentativa para la terminación de esta actividad.

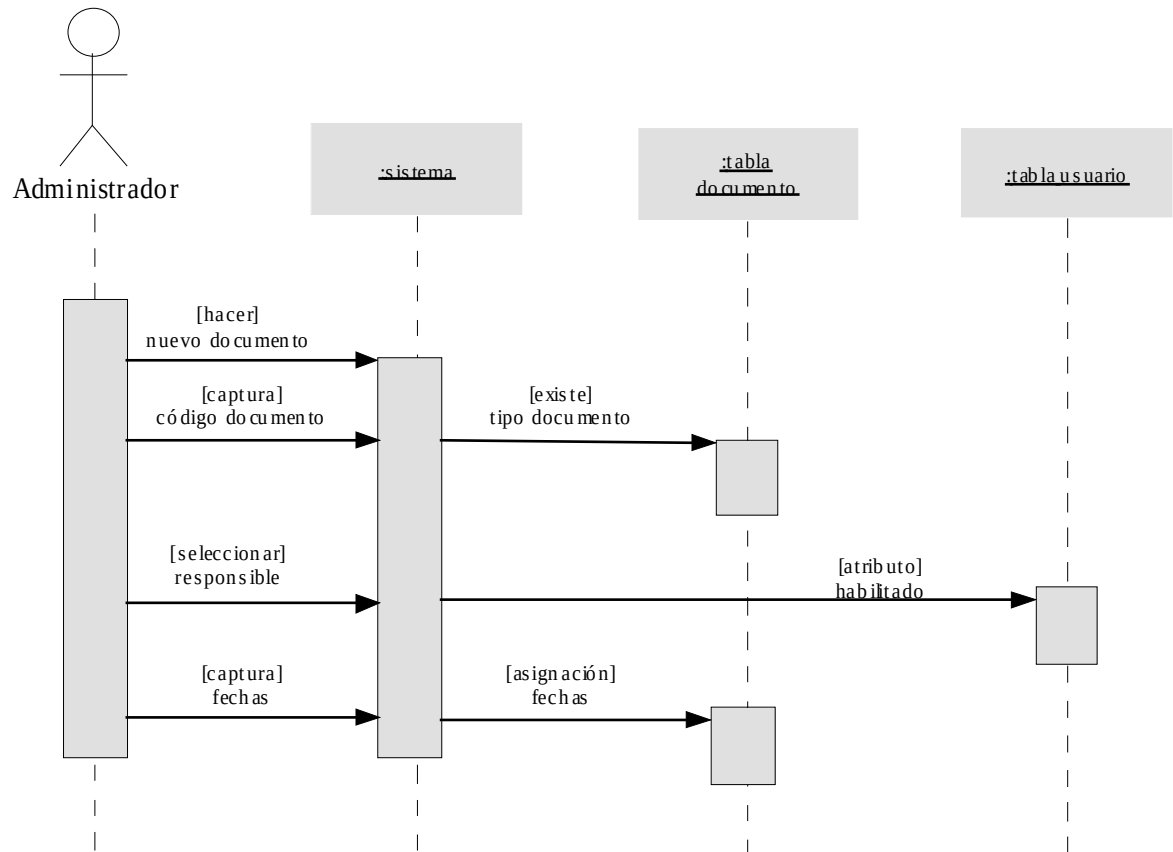


Figura 36. Diagrama de Secuencia para Nuevo documento.

1.2.2. Editar documento: Figura 37. En esta actividad, es cuando el administrador desea pasar a edición un documento que ya está en vigentes para hacerle algunas correcciones. Los documentos vigentes no se pueden modificar, solo el administrador lo puede hacer.

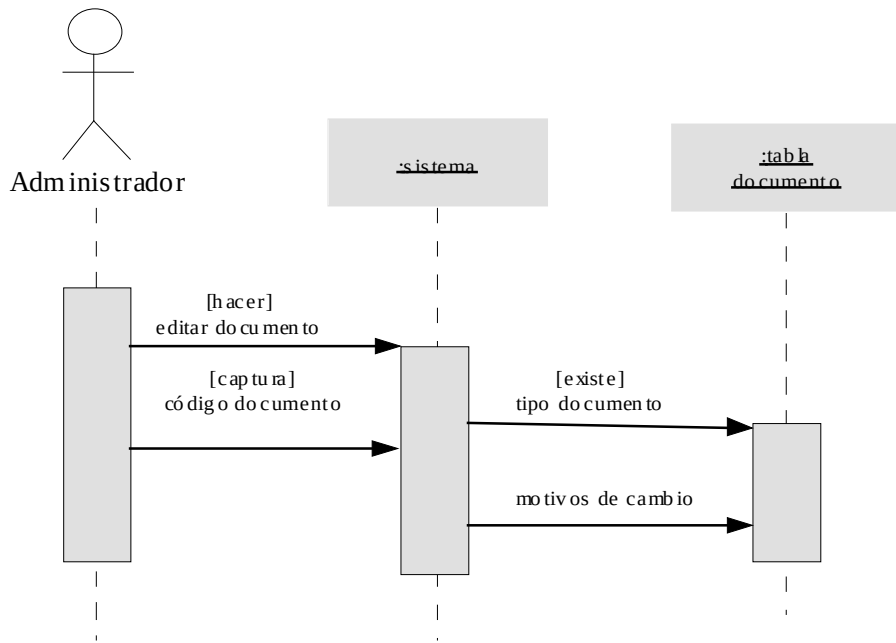


Figura 37. Diagrama de Secuencia para la edición de documentos.

1.2.3. Reasignar documento a otro usuario. En esta parte se reasigna responsable a un documento en edición.

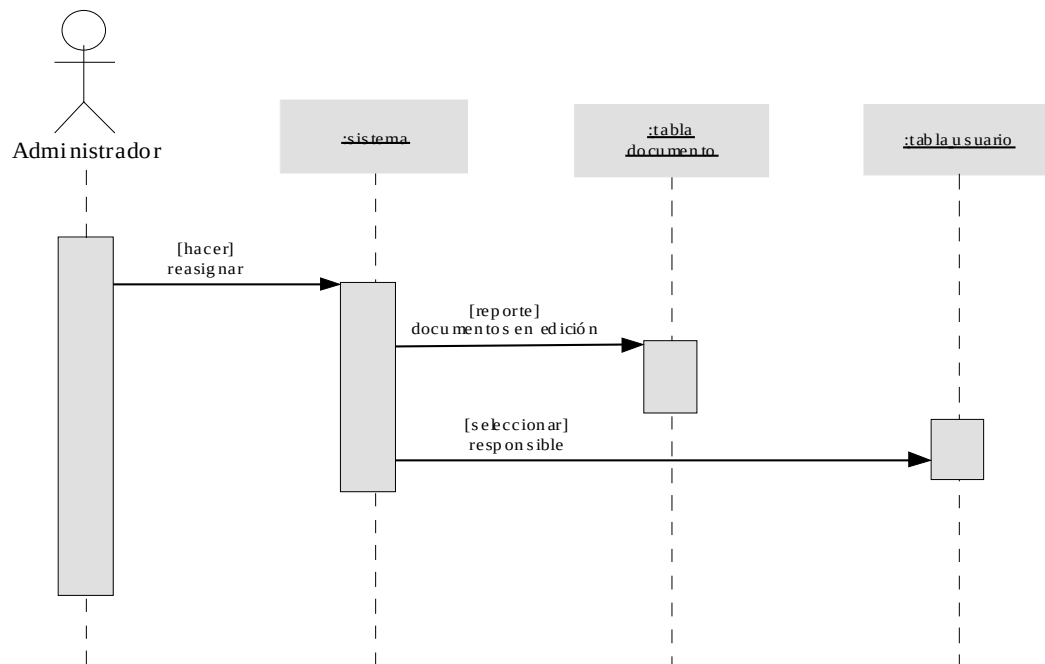


Figura 38. Diagrama de Secuencia para reasignar documento a otro usuario.

1.2.4. Código para documentos: En esta parte se asigna el código de los documentos, según características dadas en la sección de configuración de código para documentos.

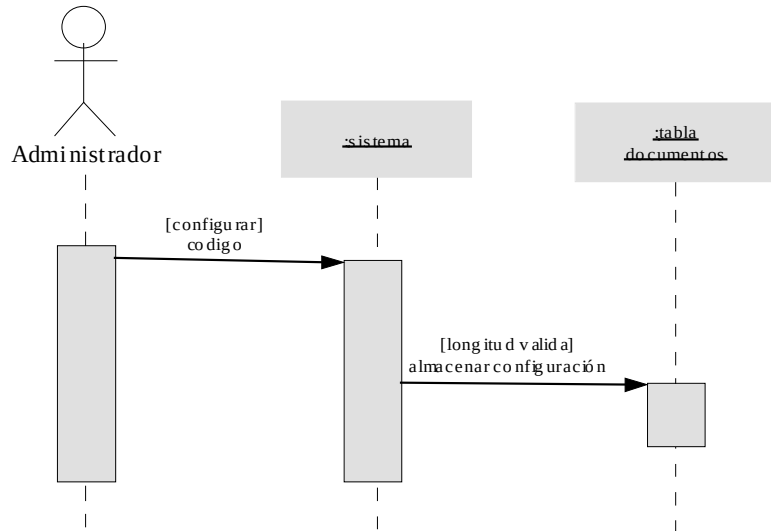


Figura 39. Diagrama de Secuencia de Código para documentos.

1.3.1. Asignar permiso a usuarios: En esta parte se define si un usuario puede ver los documentos asignados o cualquier documento sin importar que no lo tenga asignado.

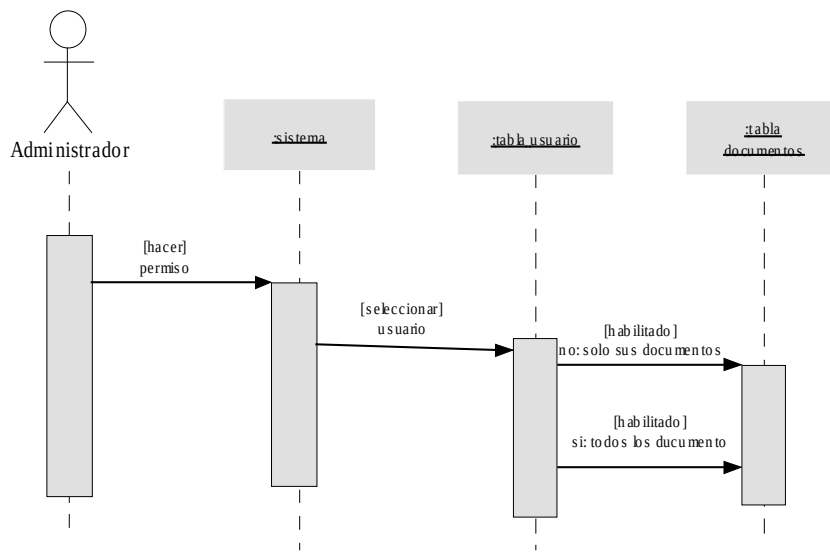


Figura 40. Diagrama de Secuencia para asignar permiso a usuarios.

1.3.2. Eliminar permiso a usuarios: En esta parte se define si un usuario puede ver solo los documentos asignados a él.

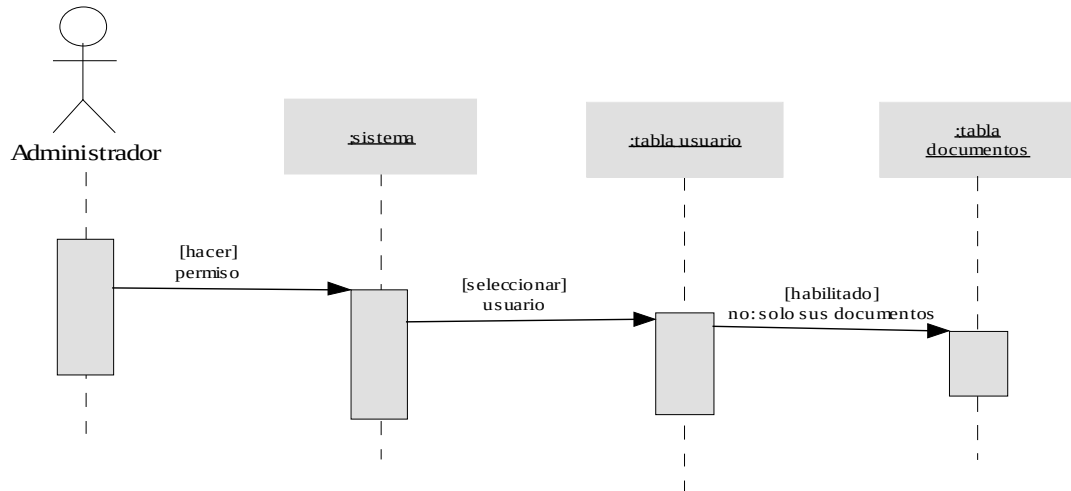


Figura 41. Diagrama de Secuencia para eliminar permiso a usuarios.

2.1.1. Datos de usuarios: Crea nuevos objetos de tipo usuario de acuerdo a los datos proporcionados por el actor principal (administrador). Es la alta a la tabla de usuarios.

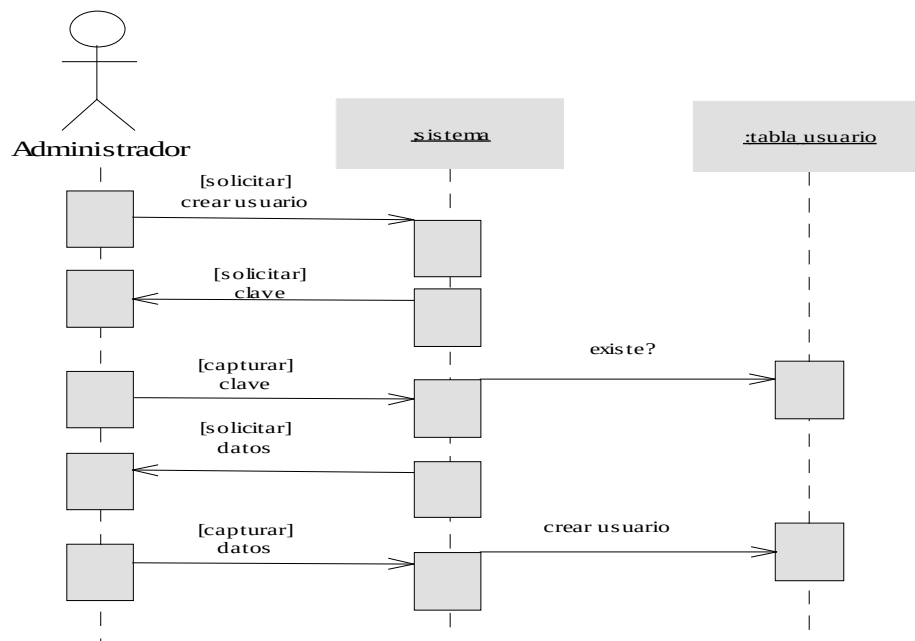


Figura 42. Diagrama de Secuencia para captura de datos a usuarios.

2.1.2. **Ubicación en la empresa.** En la tabla de usuarios se almacena también el áreas o departamento al cual pertenece el usuario.

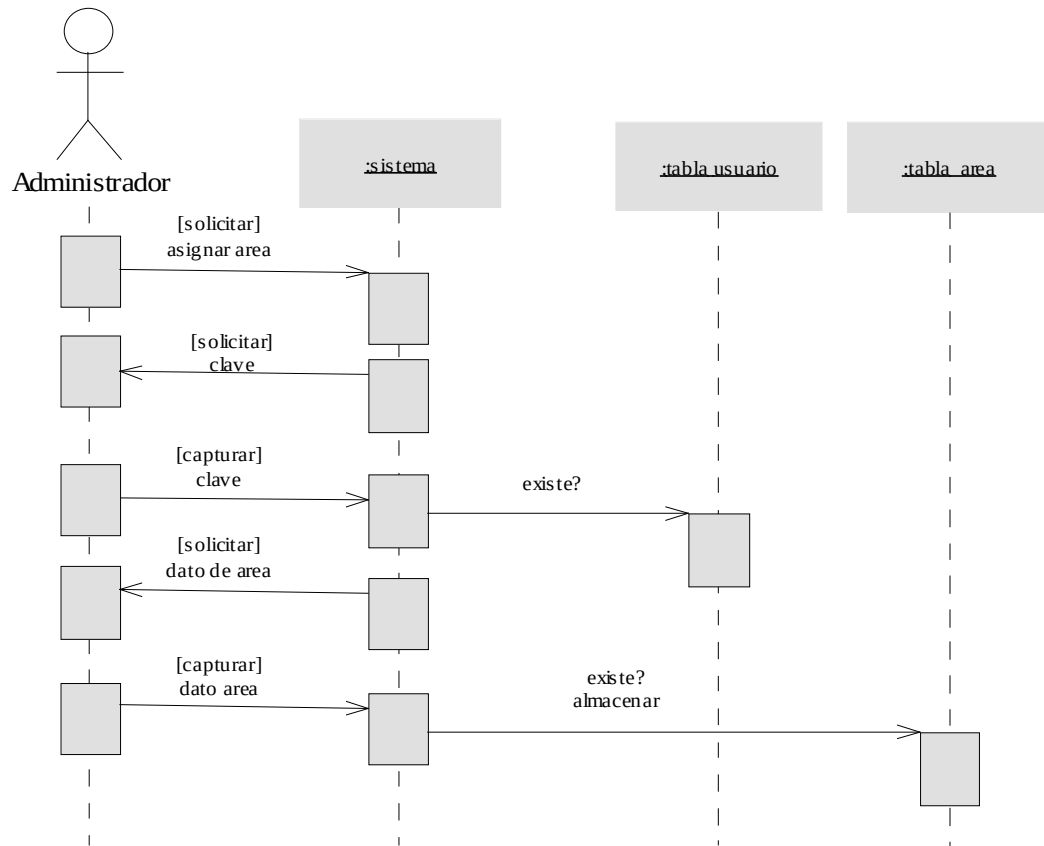


Figura 43. Diagrama de Secuencia para captura de departamento a usuario.

2.1.3. Nivel de seguridad: Es darle el atributo al usuario en cuestión de tal forma que pueda consultar y acceder a los documentos del sistema.

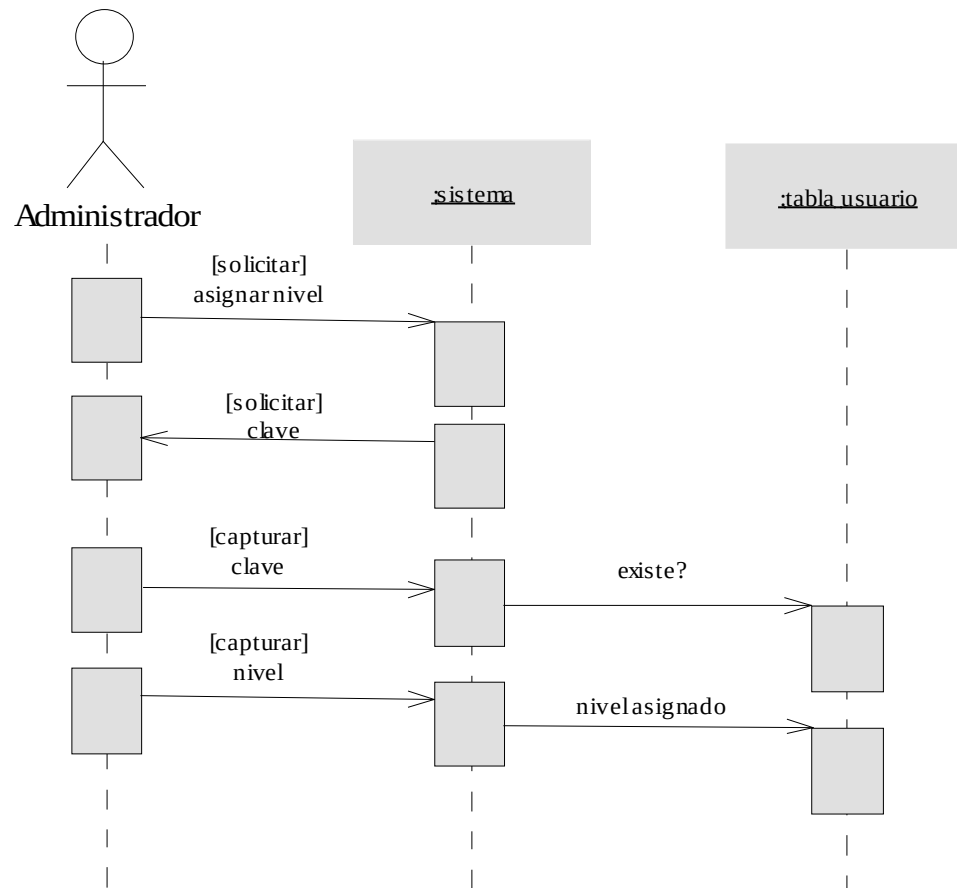


Figura 44. Diagrama de Secuencia para Nivel de seguridad a usuario.

2.2.1. **Crear área:** La tabla áreas guarda los códigos y nombres de áreas o departamentos de que se compone la organización o institución. En esta parte se añaden elementos área o departamento.

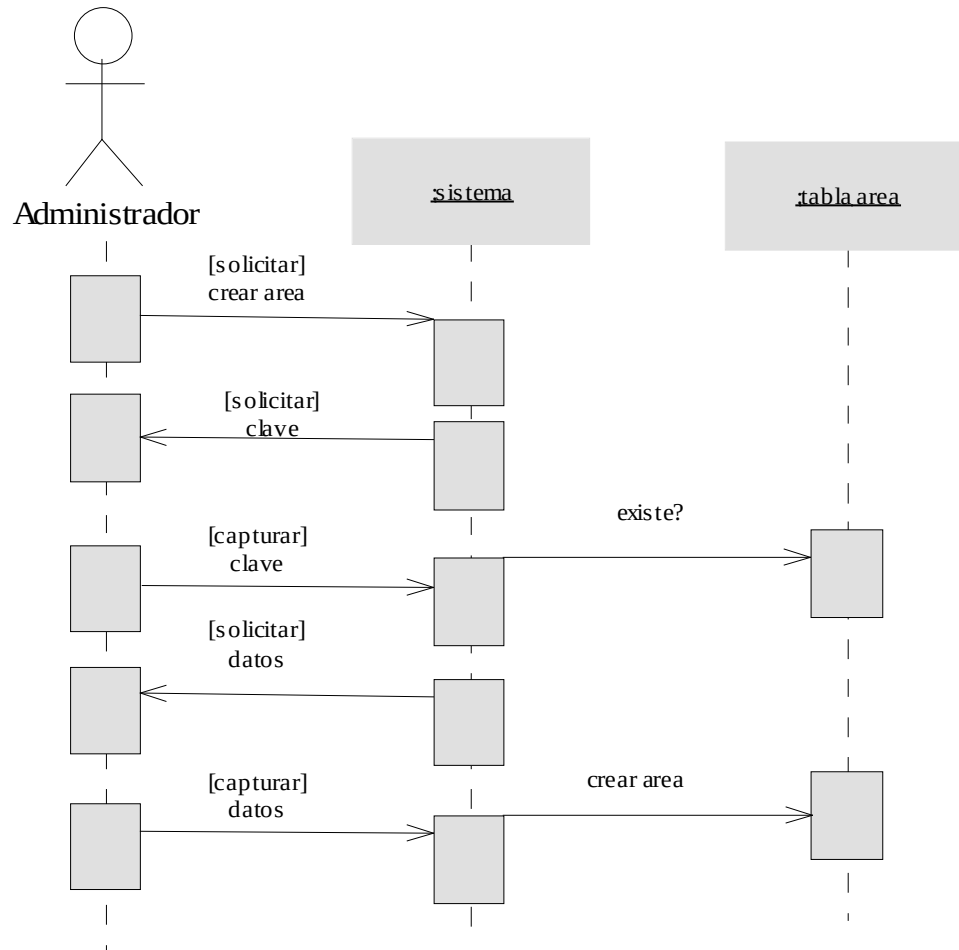


Figura 45. Diagrama de Secuencia para crear área o departamentos.

2.2.2. **Actualizar área:** Es el cambio a un elemento de la tabla área, el elemento que se modifica es el de nombre de área o departamento.

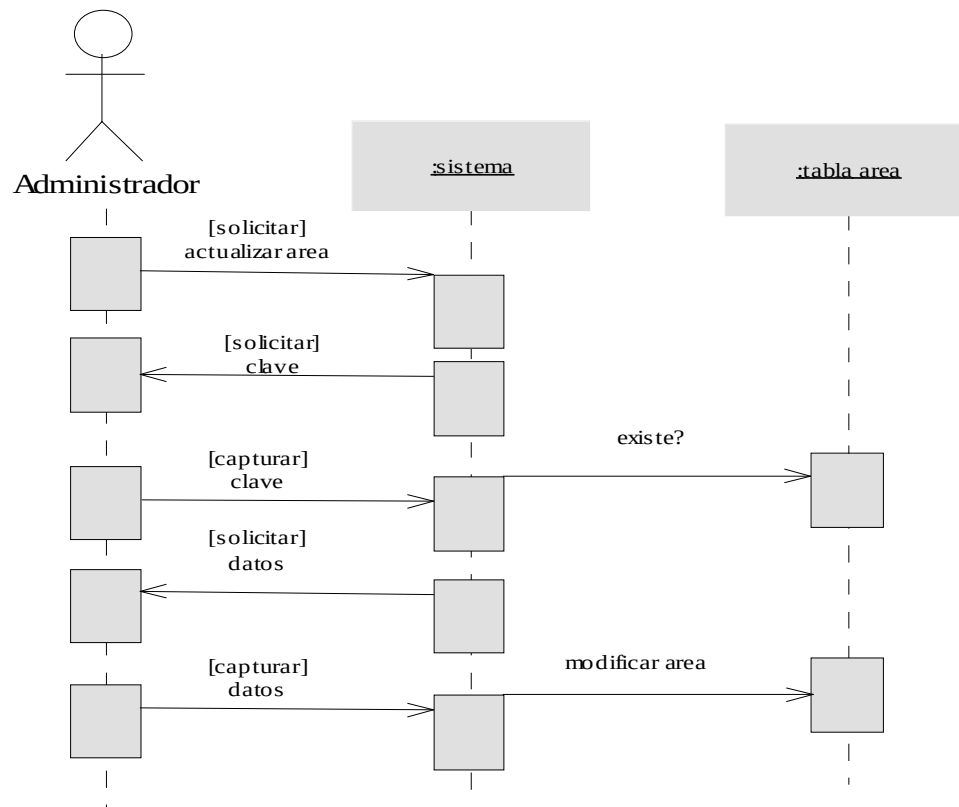


Figura 46. Diagrama de Secuencia para Actualizar área.

2.2.3. **Eliminar área:** Es eliminar un elemento de la tabla área o departamento.

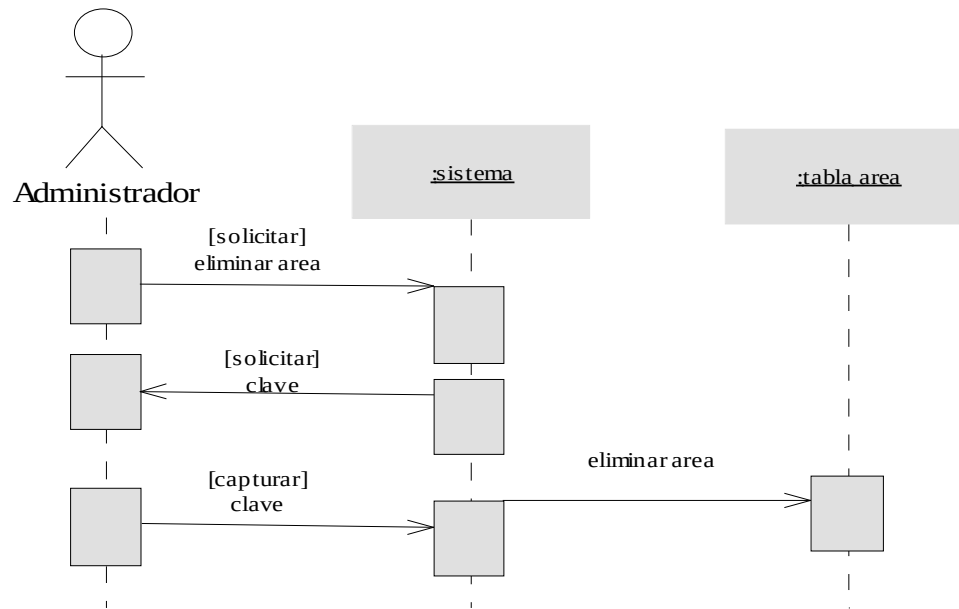


Figura 47. Diagrama de Secuencia para eliminar área o departamento.

2.3.1. **Crear cargo:** Figura 48. La tabla cargos guarda los códigos y nombres de los cargos que ocupan los usuarios en la organización. Básicamente consiste en incluir un elemento mas a la tabla cargo.

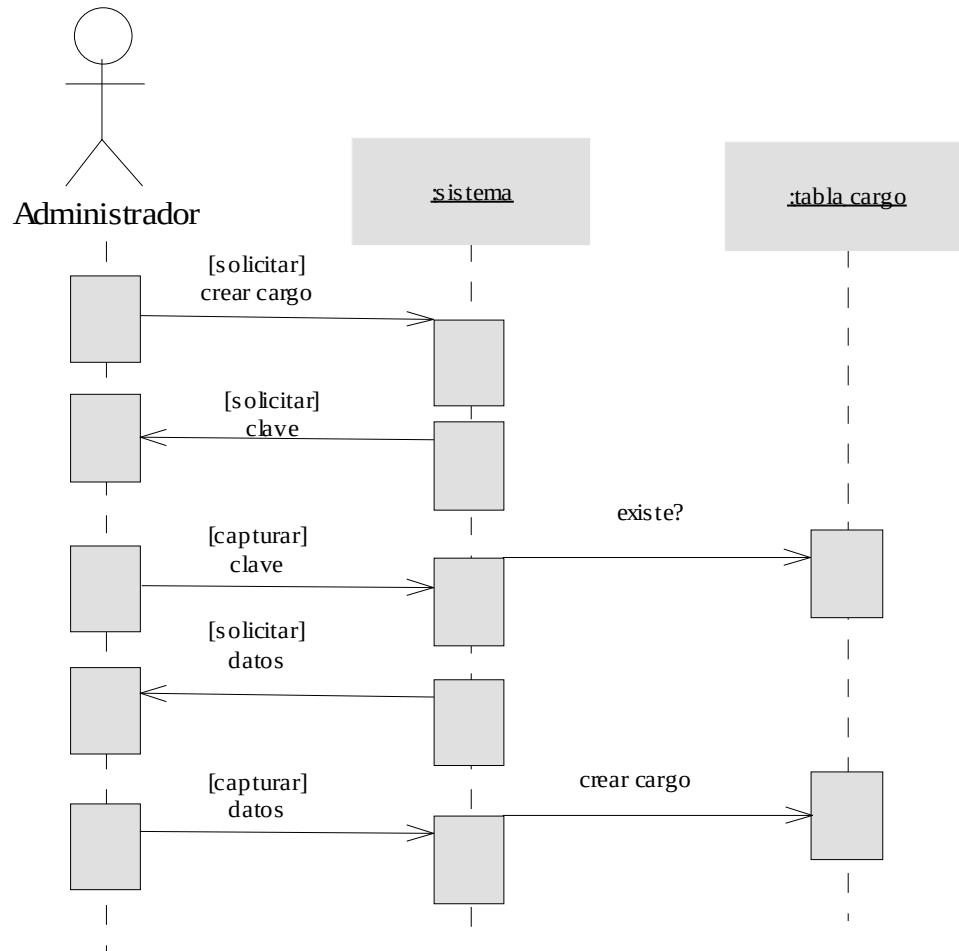


Figura 48. Diagrama de Secuencia para crear cargo.

2.3.2. **Actualizar cargo:** Es el cambio a un elemento de la tabla cargo, el elemento que se modifica es el de nombre del cargo o puesto.

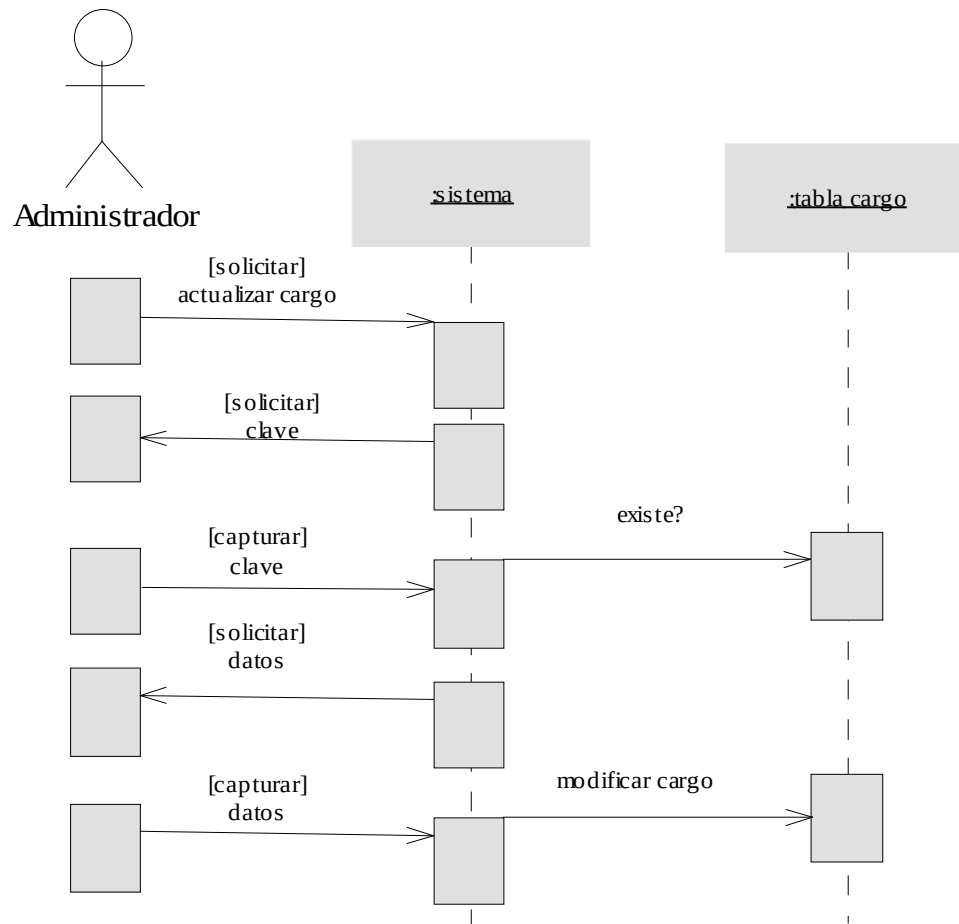


Figura 49. Diagrama de Secuencia para Actualizar cargo.

2.3.3. **Eliminar cargo:** Es eliminar un elemento de la tabla cargo o puesto.

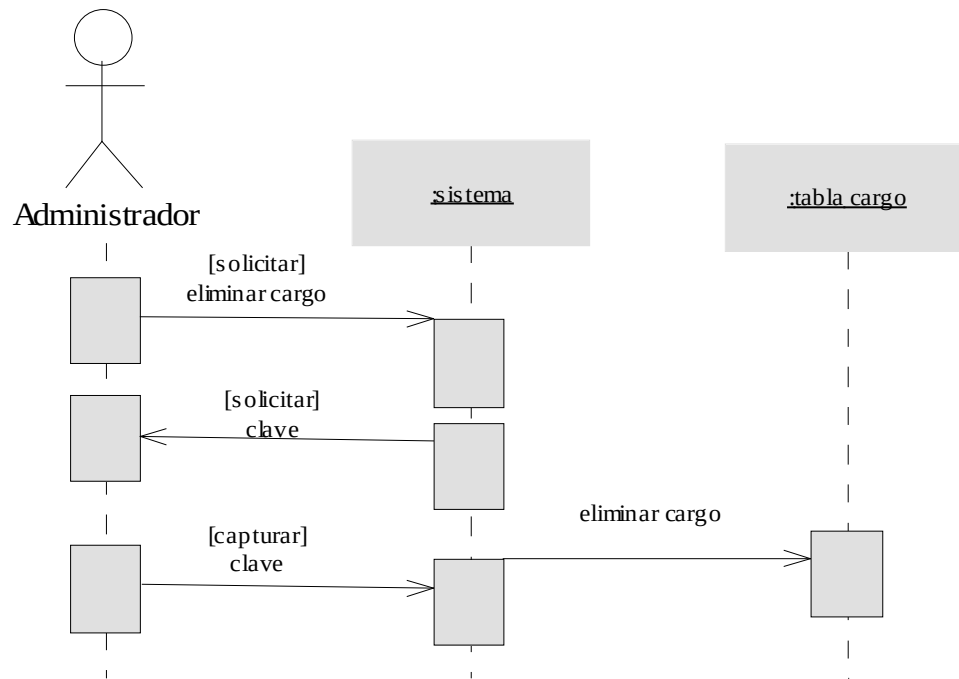


Figura 50. Diagrama de Secuencia para Eliminar cargo.

3.1.1 Propiedades del documento: En esta parte se presentan las propiedades del documento vigente; dado que ya fueron elaborados, revisados y aprobados, no es común que se modifiquen, cuando se requiera hacerlo, el cambio puede realizarse solo por el administrador o usuario habilitado.

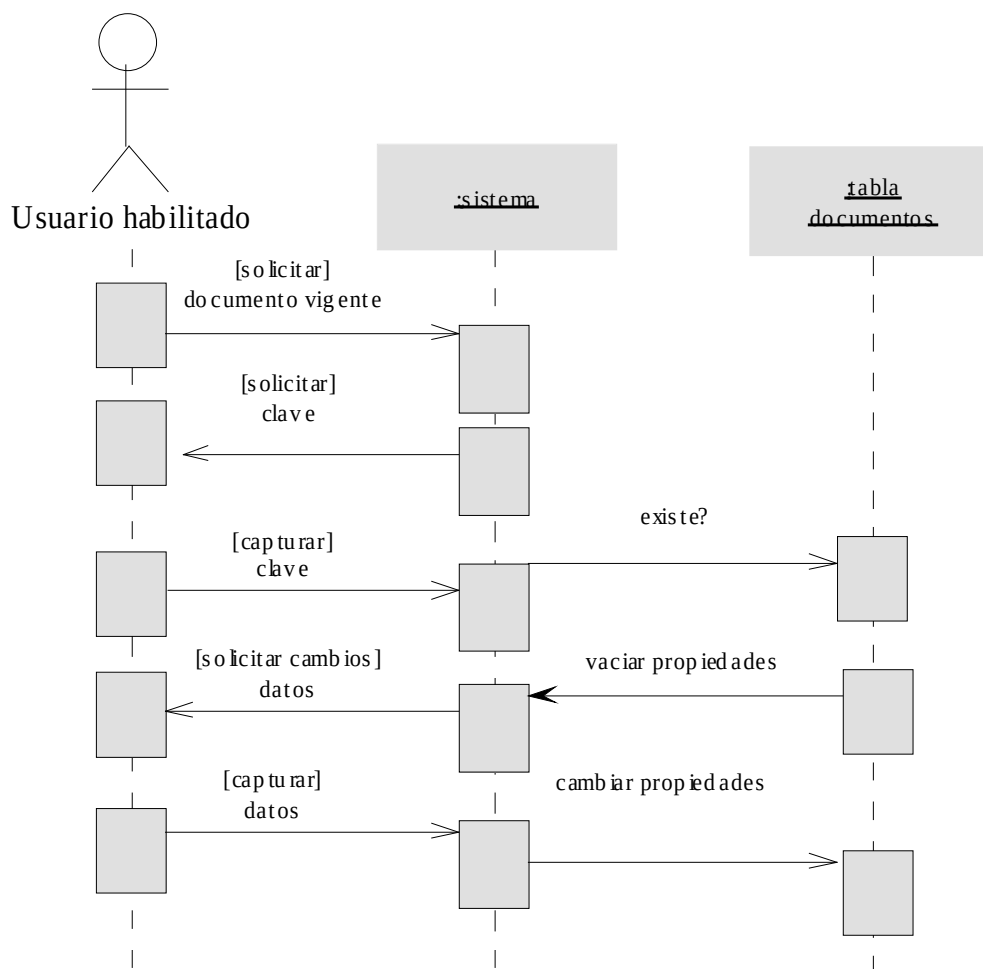


Figura 51. Diagrama de Secuencia para propiedades del documento.

3.1.2. Consulta del documento: Figura 52. Muestra los datos del documento solicitado por el usuario habilitado, los elementos que se muestran son: identificación del documento, versión, título del documento,

quien elaboró, quien revisó, quien aprobó, fecha de creación, fecha de elaboración, fecha de revisión y fecha de aprobación.

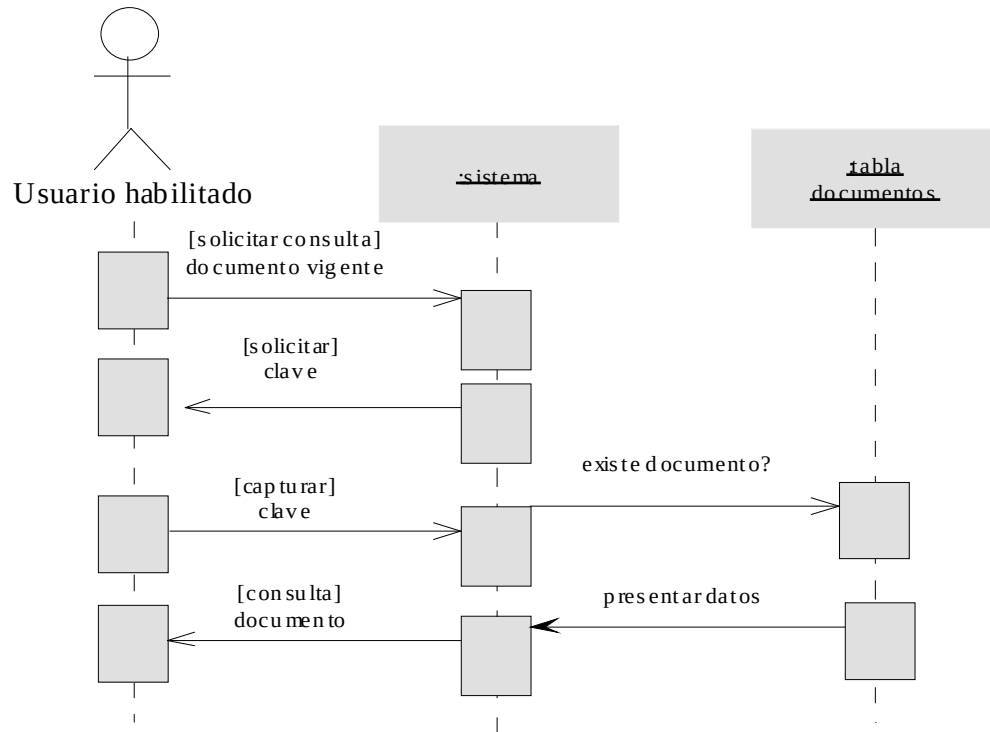


Figura 52. Diagrama de Secuencia para Consulta del documento.

3.1.3. Consulta general de documentos: Muestra todos los documentos disponibles para el usuario habilitado, los elementos que se muestran son: identificación del documento, versión y título del documento.

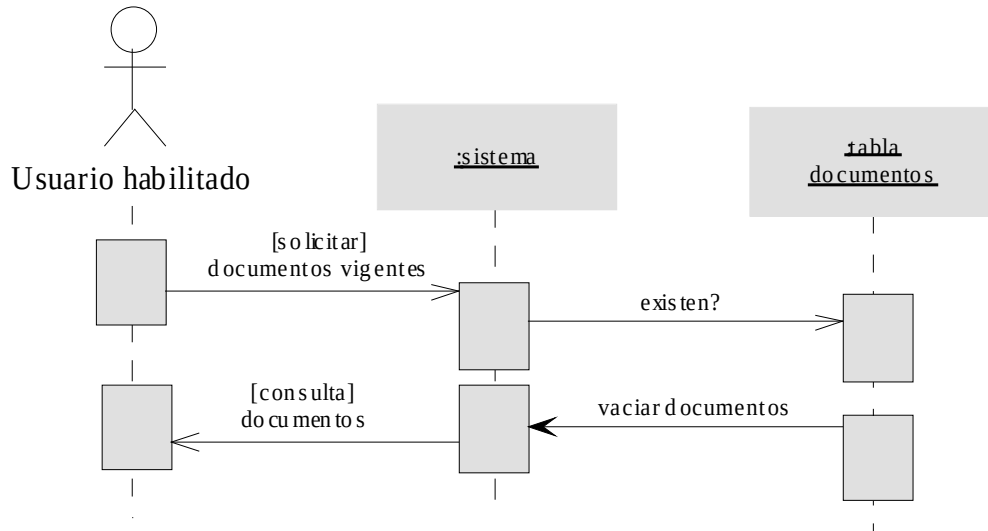


Figura 53. Diagrama de Secuencia para Consulta general de documentos.

3.2.1. Propiedades del documento: Figura 54. En esta parte se presentan las propiedades del documento en edición; esta opción se presenta cuando el usuario esta habilitado, por lo tanto puede modificar los datos del documento.

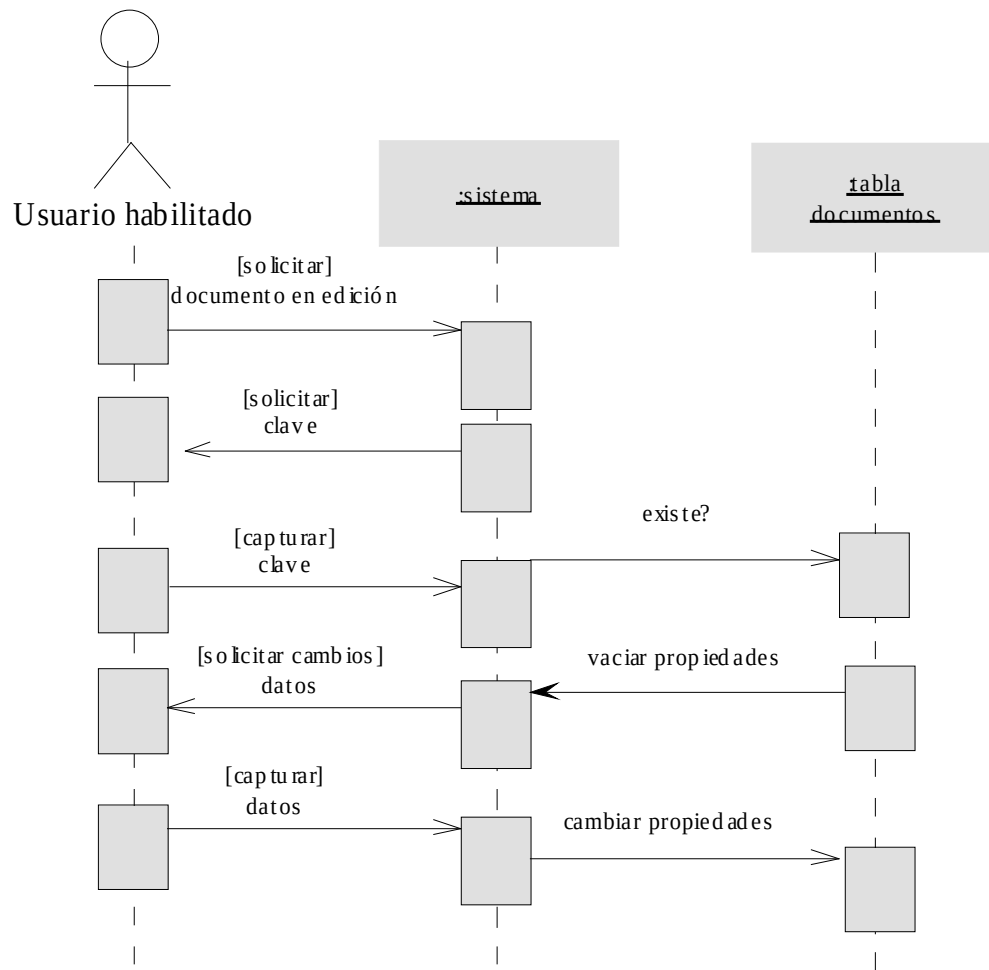


Figura 54. Diagrama de Secuencia para Propiedades del documento.

3.2.2. Edición de documento: Figura 55. El único usuario que puede editar un documento es el usuario responsable de elaboración, pero este sólo puede modificar los documentos cuando estos se encuentran en edición y su estado es editado.

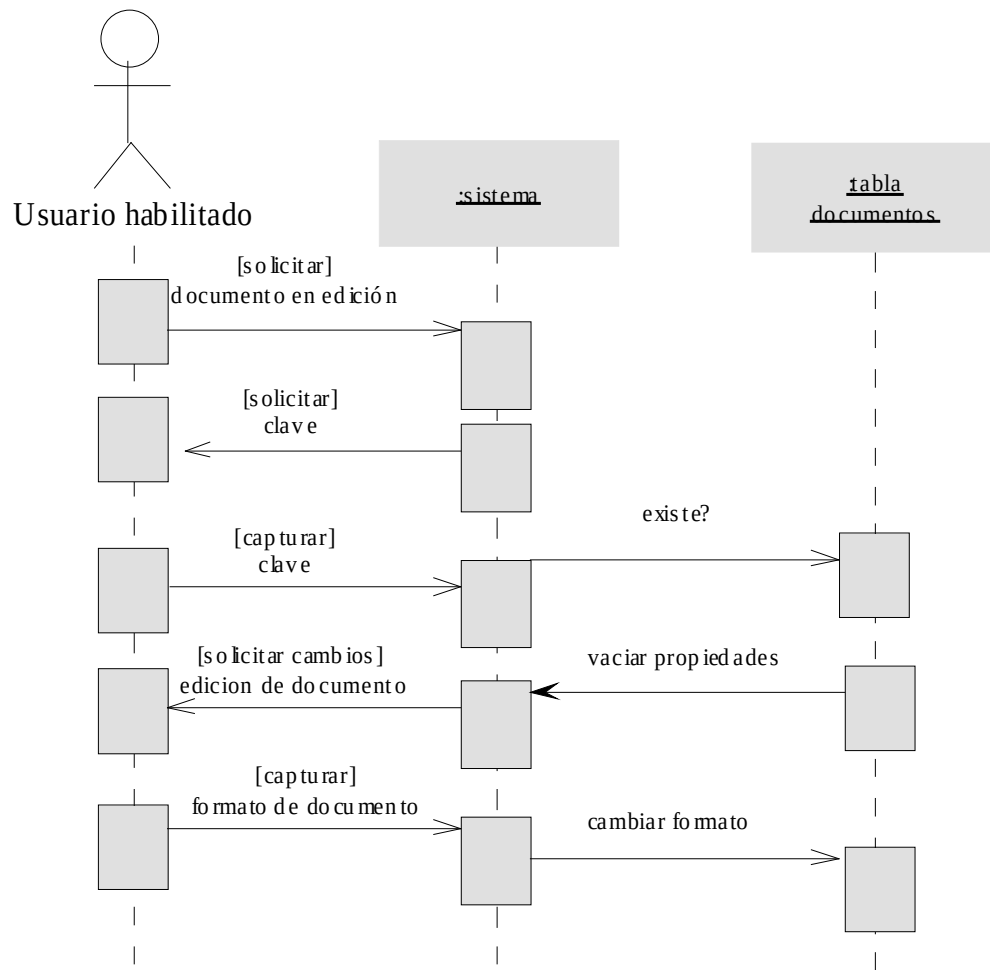


Figura 55. Diagrama de Secuencia para Edición del documento.

3.2.3. Diagrama del documento (flujograma): Para la creación del diagrama el usuario ya debe de saber que es lo que desea hacer, para ello el documento ya debe de existir y lo que desea es graficarlo .

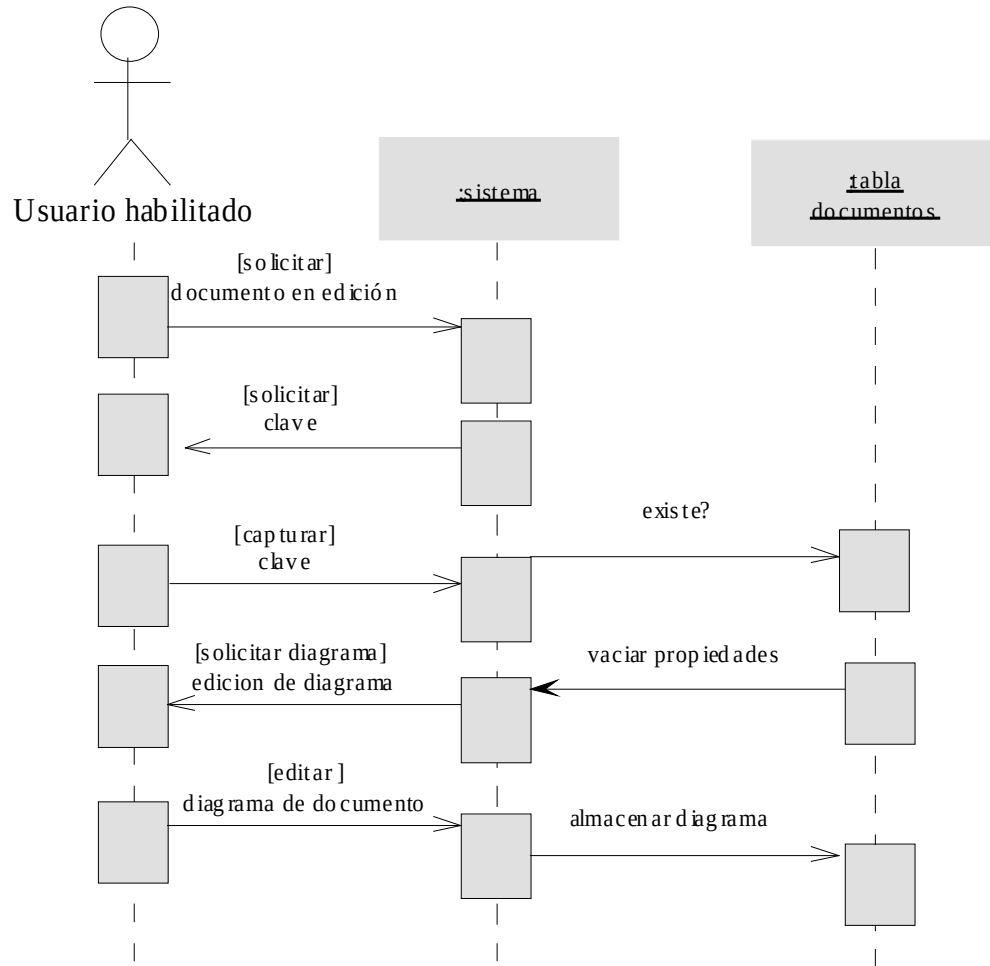


Figura 56. Diagrama de Secuencia para generar Diagrama del documento.

3.3.1. Propiedades de documento: Figura 57. En esta parte se presentan las propiedades del documento obsoleto; dado que ya fueron elaborados, revisados, aprobados y dados de baja, no es común que se modifiquen. Un documento es obsoleto cuando existe otro a partir de el, pero con una nueva

versión. Por lo tanto en esta parte un documento vigente puede cambiar a estado obsoleto.

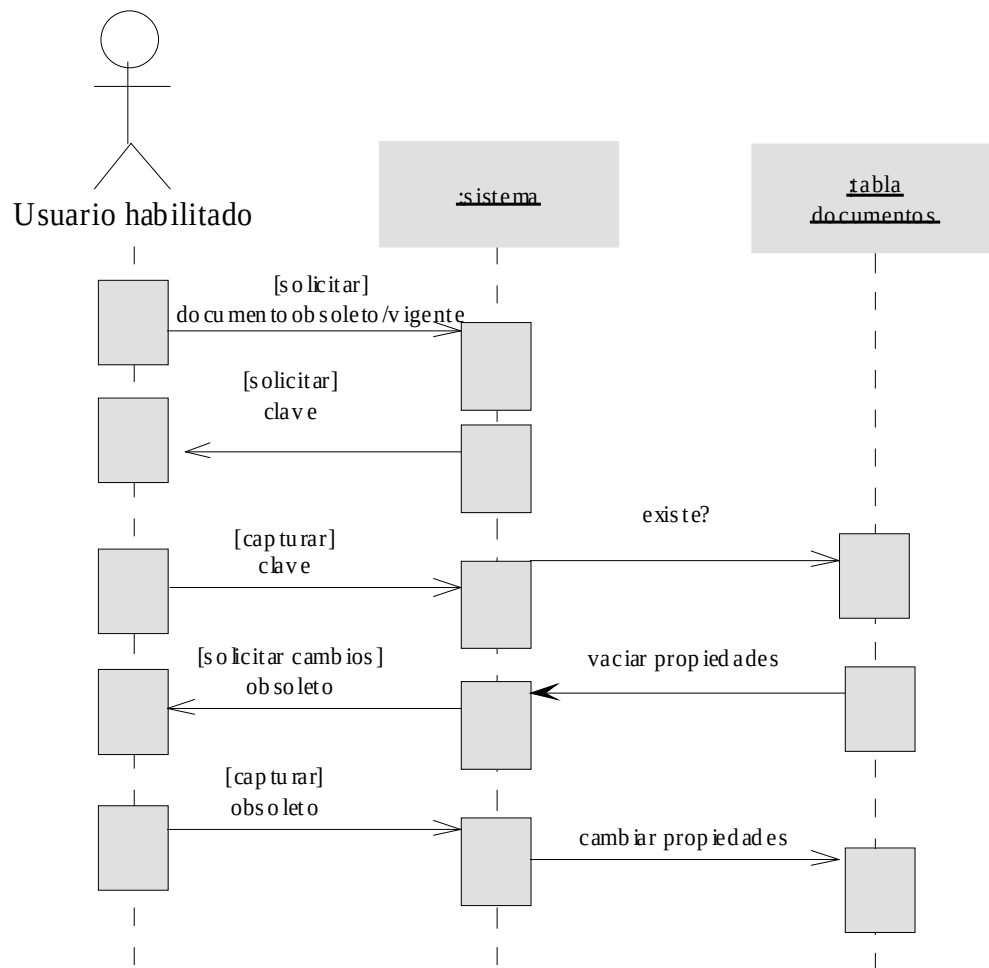


Figura 57. Diagrama de Secuencia para Propiedades de documento.

3.3.2. Consulta de documento: Figura 58. Muestra los datos del documento solicitado por el usuario habilitado, los elementos que se muestran son: identificación del documento, versión, título del documento, quien elaboró, quien revisó, quien aprobó, fecha de creación, fecha de elaboración, fecha de revisión, fecha de aprobación y fecha en la que fue asignado como obsoleto.

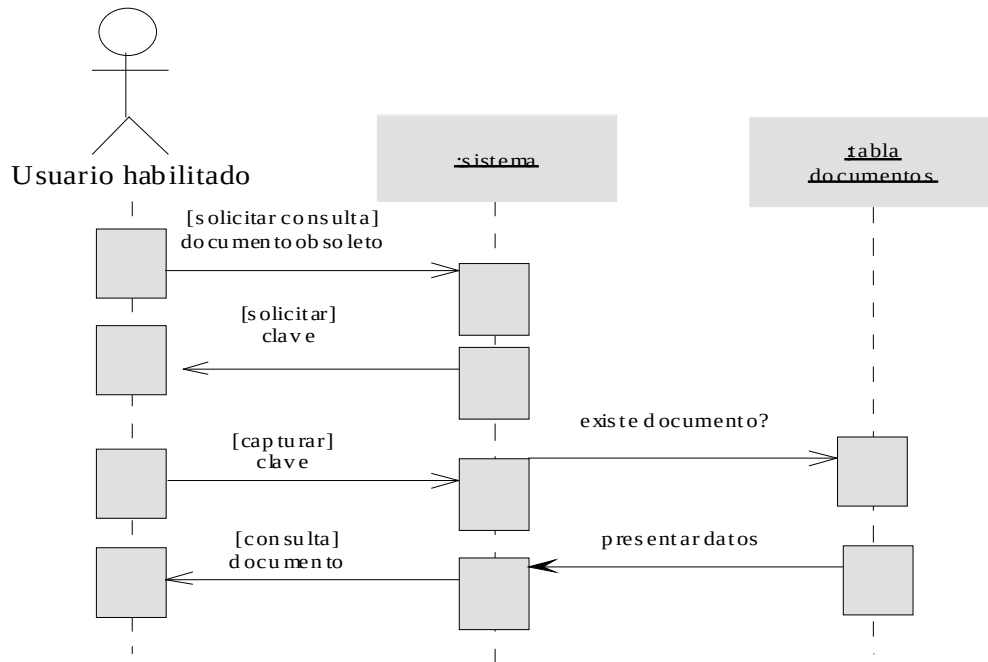


Figura 58. Diagrama de Secuencia para Consulta de documento.

3.3.3. Consulta general de documentos: Muestra todos los documentos obsoleto para el usuario habilitado, los elementos que se muestran son: identificación del documento, versión, título del documento, fecha en la cual fue dado de baja.

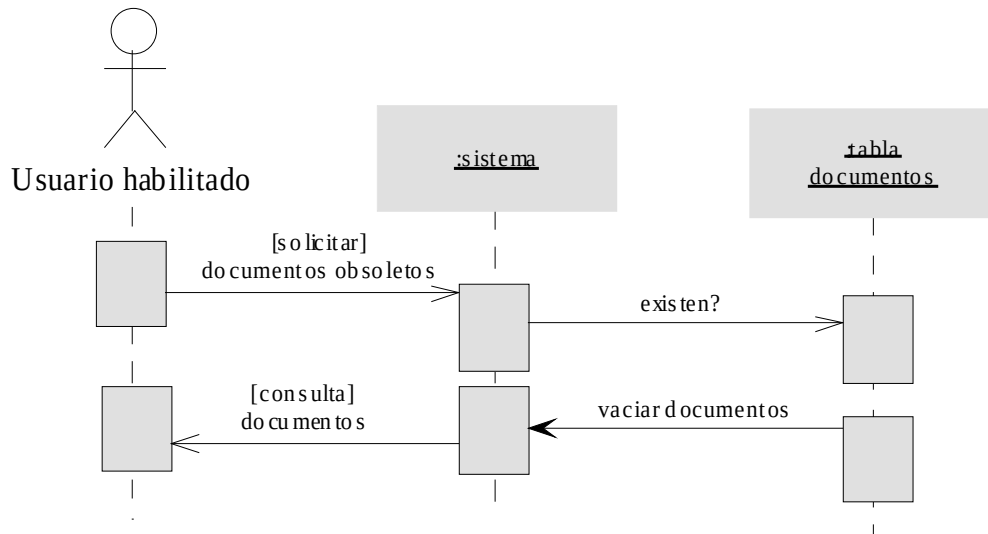


figura 59. Diagrama de Secuencia para Consulta general de documentos.

4.1.1. **Crear registro:** Los usuarios que son habilitados por el administrador puede crear nuevos registros de los documentos en edición que él este elaborando. Los registros en el sistema pueden ser registros impresos como preservados en otros medios: magnéticos, filmación, grabación, fotografía, etc.

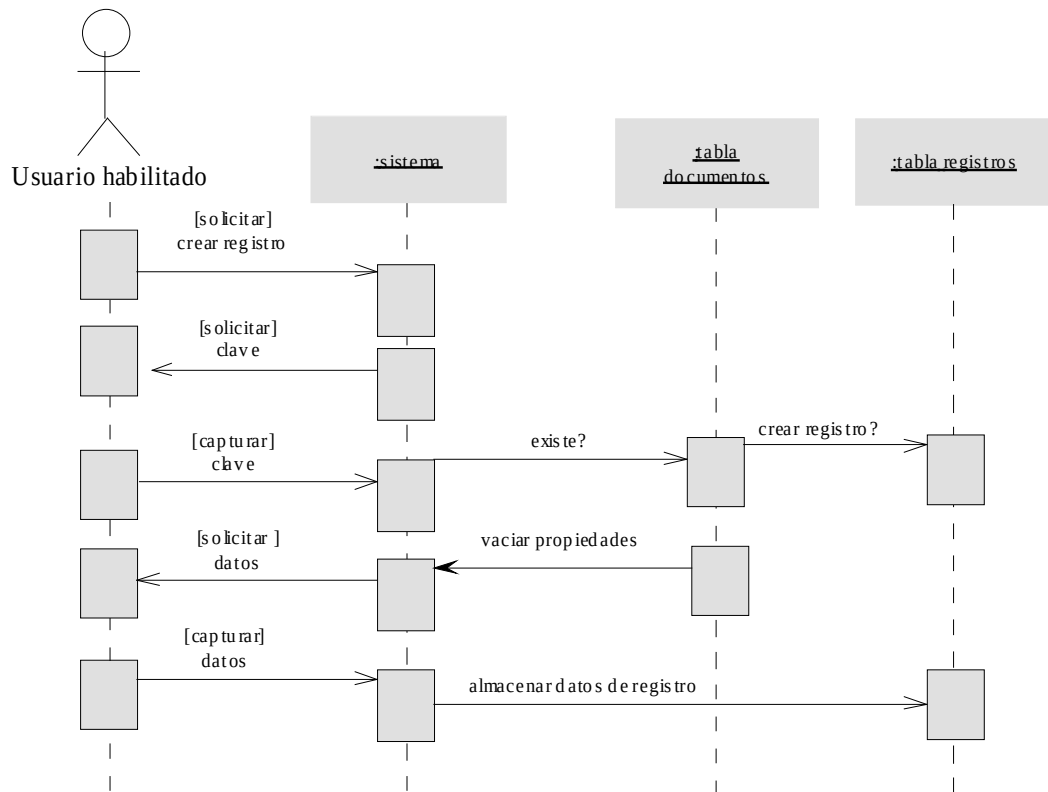


Figura 60. Diagrama de Secuencia para Crear registros.

4.1.2. **Consulta de registro:** Figura 61. Esta parte permite la consulta general de los registros del sistema de calidad, a la vez se puede realizar filtrar los registros de calidad para consultas mas concretas.

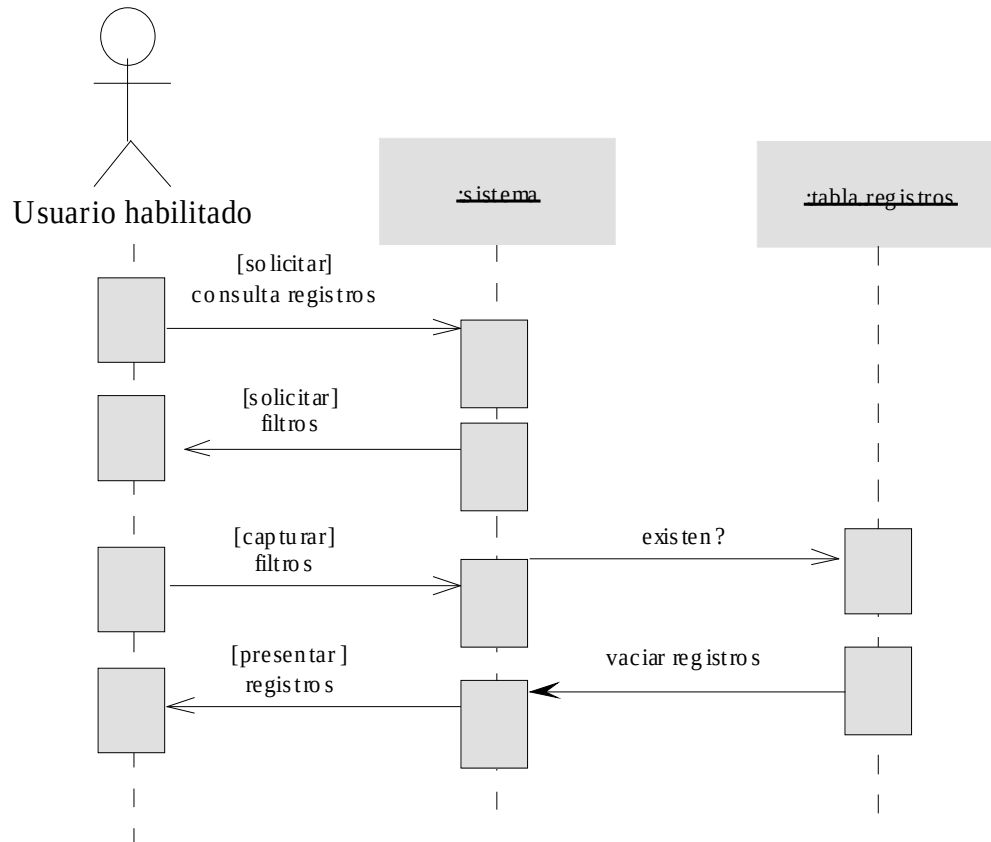


Figura 61. Diagrama de Secuencia para Consulta de registros.

5.1.3. Modelo conceptual para el SI.

En el modelo conceptual se describen los componentes objetos, sin entrar a detalle en los aspectos de implementación. La metodología de la programación orientada a objetos, utiliza el modelo de objetos para representar el esquema conceptual.

Modelo de objetos.

1.1 **Objeto usuario:** datos de los usuarios que utilizan el SGC. Los tipos de usuarios que pueden participar en el sistema son: Desarrollador, Administrador y usuario habilitado. El usuario habilitado puede realizar una o mas actividades como son: elaborar, revisar, aprobar y consultar.

Clase Usuario
Atributos: código: clave: nombre: profesión: área: cargo: nivel de acceso:(administrado desarrollador usuario habilitado) dirección electrónica:
Métodos: Administrar() Configuraciones() Administrar documentos() Documentos por usuario() Archivar() Usuarios() Áreas() Cargos() Documentar() Documentos vigentes() Documentos en edición() Documentos obsoletos() Controlar() Control de registros de calidad()

Figura 62. Clase Usuario.

1.2 **Objeto documentos:** Datos de los documentos que se siguen en los procesos de un SGC. Los tipos de documentos que se presentan en el sistema son: vigentes, edición y obsoletos.

Clase Documento
Atributos: tipo: código: área: consecutivo: titulo: profesión: quien elaboró: quien revisa: quien aprueba: fecha creación: fecha inicial: fecha final: versión:
Métodos: Administrar() Administrar documentos() Documentos por usuario() Documentar() Documentos vigentes() Documentos en edición() Documentos obsoletos() Controlar() Control de registros de calidad()

Figura 63. Clase Documento.

1.3 **Objeto cargo:** Datos de los cargos que pueden tener los usuarios del SGC. Un caso puede ser: gerente, directos, jefe de laboratorio, etc.

Clase Cargo
Atributos: código: nombre:
Métodos: Administrar() Configuraciones() Archivar() Usuarios() Cargos()

Figura 64. Clase Cargo.

1.4 **Objeto área:** Los datos de las áreas o departamentos a los cuales pertenecen los usuarios de SGC. Un caso puede ser: administración, dirección, laboratorio x, etc.

Clase Área
Atributos: código: nombre:
Métodos: Administrar() Configuraciones() Archivar() Usuarios() Área()

Figura 65. Clase Área.

1.5 Objeto registro: Un elemento importante en la documentación del SGC son los registros. En la clase registro, se incluyen a que documentos pertenece el registro, así como el proceso que se sigue para recabar, registrar y resguardar la evidencia de la documentación.

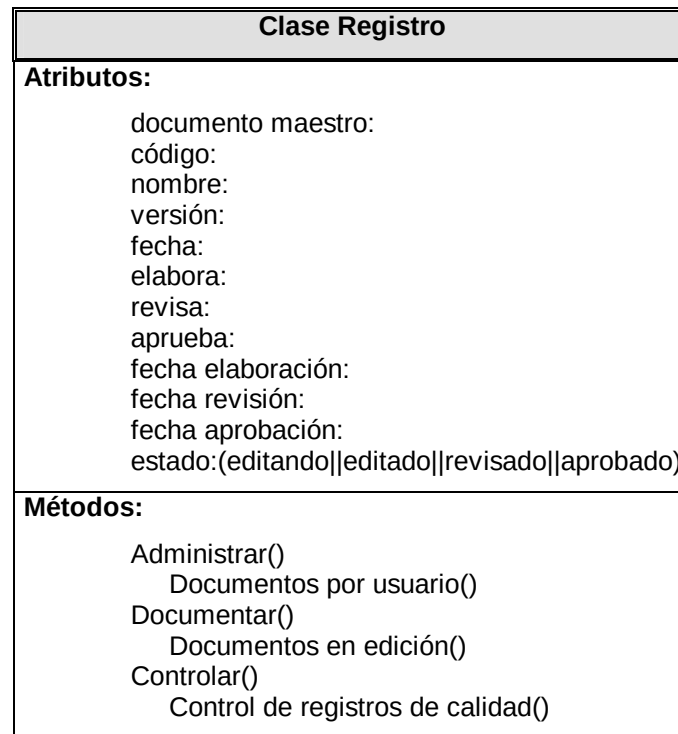


Figura 66. Clase Registro.

5.1.4. Modelo de emplazamiento.

En el modelo, en el cual interactúan todos los elementos del SGC del II, este modelo sirve como referencia para la elaboración de un trabajo futuro, el modelo está representado en un diagrama de emplazamiento. El modelo presenta que se requiere de un servidor en el que se encuentren todas las bases de datos con la documentación requerida, para la obtención de reportes, indicadores, memorandos, comisiones y demás información que proporciona el SGC. A la vez, cada elemento del SGC, contará con una estación de trabajo en la que se tendrá

restricciones de acuerdo al documento, que se quiera accede, que puede ser para algún mantenimiento o tan solo consultar. En el diagrama de emplazamiento de la siguiente figura, se presenta cuales son solo procedimientos mas importantes, que cada nodo o elemento debe de tener. En la Figura 67, representada por un esquema de emplazamiento, se presentan los elementos indispensables para el SI.

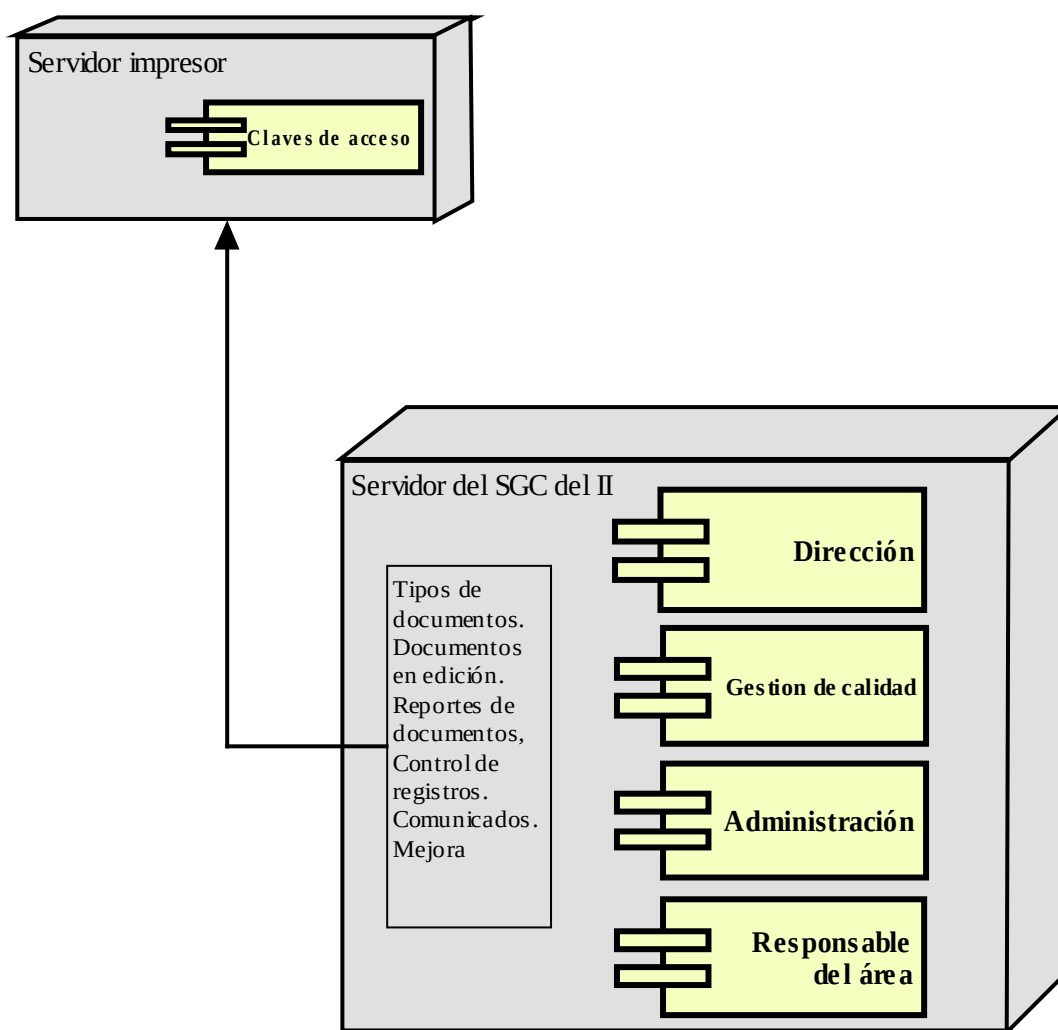


Figura 67. Modelo de Emplazamiento del SI.

5.2 Conclusiones y recomendaciones.

Una vez realizado el estudio de los elementos del SGC del II y desarrollado el modelo utilizando UML para representarlo en los diferentes esquemas, se presentan a continuación las conclusiones y recomendaciones.

Conclusiones:

- Se obtuvo el modelo para el SGC del II, en el cual se presentan los elementos mas relevantes.
- Los diagramas obtenidos en UML presentan los esquemas a utilizar en el proceso de diseño-codificación de el control de registros de un Sistema de Calidad aplicable al II en el que se desea sistematizar sus procesos de documentación y registros.
- Se presenta el modelo del SI para controlar electrónica y digitalmente los registros del SGC.
- Se utilizó UML como lenguaje para la realización del modelo, obteniendo las bases para darle continuidad en la fase de diseño-codificación.
- Se presenta la POO como una protección en el SI a los datos.
- Se identificaron los procesos a automatizar, representándolos en UML.
- Se utilizó el software SmartDraw, para la realización del diseño.

Recomendaciones:

- Para el proceso de certificación de cualquier organización es relevante que el personal tenga una cultura de calidad, para que una vez certificada la organización, esta continúe.
- Acompañar a la certificación con una cultura de calidad.
- Es importante que los desarrolladores de sistemas, utilicen los modelos para la creación de SI.

- El software bien desarrollado puede ayudar a reducir los costos de mantenimiento.
- Integrar a la tecnología informática en los procesos de certificación.
- El modelo propuesto en el presente trabajo, puede ser mejorado, utilizando nuevas herramientas desarrolladas en el campo de los SI, así como en el uso de nuevos desarrollos de métodos.
- Para el desarrollo de sistemas de información, es importante tomar como base los modelos como parte del diseño.

Anexo A.

Historia de UML

Los lenguajes de modelado orientado a objetos aparecieron, a mediados de los setenta y finales de los ochenta cuando los metodólogos, enfrentados a los nuevos lenguajes de programación orientada a objetos y a unas aplicaciones cada vez más complejas, empezaron a experimentar con enfoques alternativos al análisis y diseño. El número de métodos orientados a objetos se incrementó de 10 a 50 durante el período de 1989 y 1994. Gran parte de estos usuarios de estos métodos tenían problema al intentar encontrar un lenguaje de modelado que cubriera necesidades completamente alimentado de esta forma, la llamada guerra de métodos.

Aprendiendo de estas experiencias, comenzaron a aparecer nuevas generaciones de métodos, entre los que destacan de manera muy clara unos pocos métodos, en especial el método de Booch, el método OOSE (Object-Oriented Software Engineering, Ingeniería de Software Orientado a Objetos) de Jacobson y el método OMT (Object Modeling Technique, Técnica de Modelado de Objetos) de Rumbaugh. Otros métodos importantes fueron Fusion, Shlaer-Mellon y Coad-yourdon. Cada uno de estos era un método completo, aunque tenían sus puntos fuertes y sus debilidades. El método de Booch era particularmente expresivo durante las fases de diseño y construcción de los proyectos, OOSE proporcionaba un soporte excelente para los casos de uso como forma de dirigir la captura de requisitos, el análisis y el diseño de alto nivel, y OMT-2 era principalmente útil para el análisis y para los sistemas de información con gran cantidad de datos.

Una masa crítica de ideas comenzó a formarse en la primera mitad de los noventa, cuando Grady Booch de la Rational Software Corporation, Ivar Jacobson de Objectory y James Rumbaugh de General Electric, empezaron a adoptar ideas de cada uno de los otros métodos, los cuales habían sido reconocidos en conjunto como los tres principales métodos orientados a objetos a

nivel mundial. Como creadores principales de los métodos de Booch, OSSE y OMT, hubo una motivación para crear un lenguaje unificado de modelos por tres razones:

- Cada uno de los métodos estaba evolucionando independientemente hacia los otros dos, por lo que tenía sentido hacer continua evolución de forma conjunta en vez de hacerlo por separado, eliminando la posibilidad confundir aún más a los usuarios.
- Al unificar los métodos, se proporcionaría cierta estabilidad al mercado orientado a objetos permitiendo que los proyectos se pusieran de acuerdo a un lenguaje de modelado maduro y permitiendo a los constructores de herramientas que se centraran en proporcionar más características útiles.
- Se esperaba que la colaboración mutua de los tres métodos introducirían mejoras en los métodos anteriores, ayudando a los autores a capturar las lecciones aprendidas y a cubrir problemas que ninguno de los métodos independientes había manejado bien anteriormente.

Cuando se comenzó con la unificación, se establecieron tres metas para el trabajo:

- Modelar el sistema desde el concepto hasta los artefactos ejecutables, utilizando técnicas orientadas a objetos.
- Cubrir las cuestiones relacionadas con el tamaño, inherentes a los sistemas complejos y críticos.
- Crear un lenguaje de modelado utilizable tanto por las personas como por las máquinas.

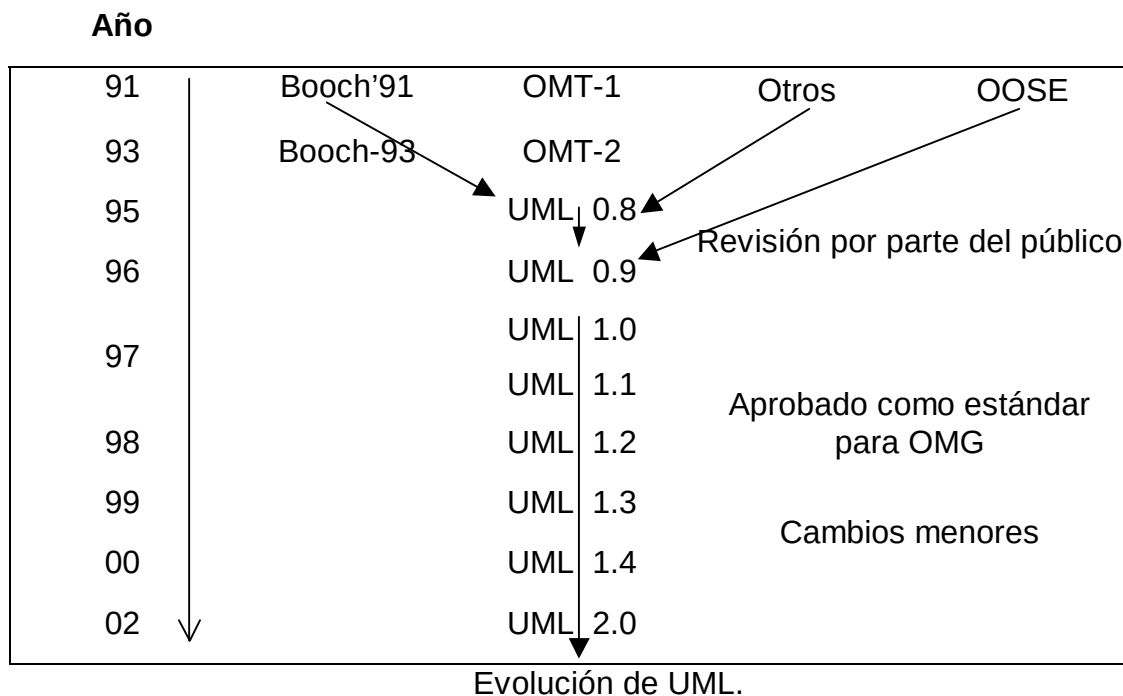
Desarrollar un lenguaje que se utilice en el análisis y diseño Orientado a Objetos (OO) no es muy diferente de diseñar un lenguaje de programación. En primer lugar, se debía acotar el problema. ¿Debería el lenguaje incluir la especificación de requisitos? ¿Debería el lenguaje incluir la especificación de requisitos? En segundo lugar, se tenía que conseguir un equilibrio entre expresividad y simplicidad. Un lenguaje demasiado simple limitaría demasiado el abanico de

problemas que se pondrían resolver; un lenguaje demasiado complejo sobrecargaría a las personas que lo utilizaran. En el caso de la unificación de métodos existentes, también se debía tener en cuenta que las bases se habían obtenido. Si se hacían demasiados cambios se confundiría a los usuarios existentes; si se resistía el avance del lenguaje, se perdería la oportunidad de incorporar a un conjunto mucho mas amplio de usuarios y de hacer un lenguaje más sencillo. La definición del Lenguaje Unificado de Modelado (UML) se ha esforzado por conseguir el mejor equilibrio en cada uno de estas áreas.

El esfuerzo de UML comenzó en Octubre de 1994, cuando Rumbaugh se unió a Booch en Rational. El objetivo inicial del proyecto fue la unificación de los métodos de Booch y OMT. El borrador de la versión 0.8 del Método Unificado, como fue llamado entonces, se publicó en Octubre de 1995. Por esa época, Jacobson se unió a Rational, y el enlace del proyecto UML se amplió para incorporar OOSE. Los esfuerzos se plasmaron en la versión 0.9 en junio de 1996. Durante 1996, pidieron y recibieron realimentación de la comunidad internacional relacionada con la ingeniería del software. Durante ese tiempo también se vio claro que muchas organizaciones de software veían a UML como un punto estratégico de su negocio. Se estableció un consorcio de UML, con varias organizaciones que querían dedicar recursos para trabajar hacia una definición fuerte y completa de UML.

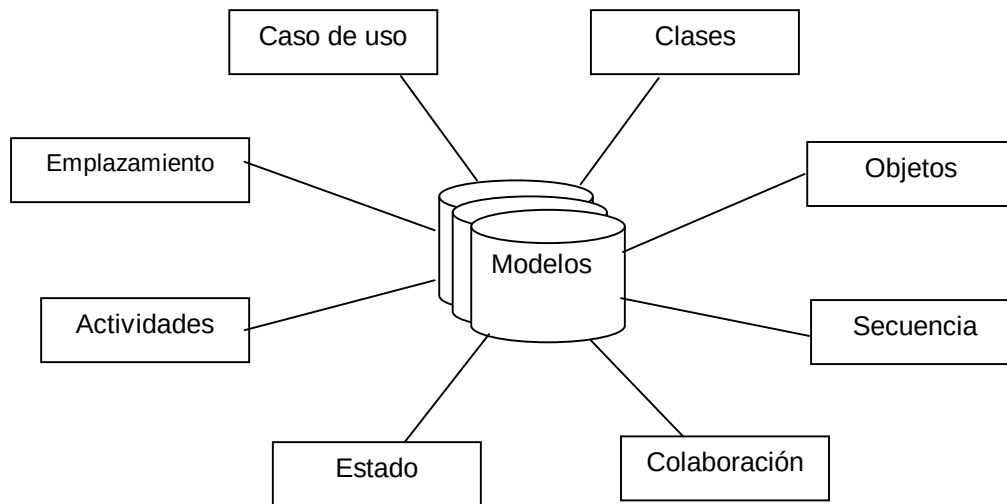
Algunas de las organizaciones que contribuyen a la definición de 1.0 de UML fueron Digital Equipment Corporation, Hewlett-Packard, IBM, MCI Systemhouse, Microsoft, Oracle, Rational, Texas Instrument, entre otros. Esta colaboración produjo UML 1.0, un lenguaje de modelado bien definido, expresivo, potente y aplicable a un amplio espectro de dominios de problema. UML 1.0 ofreció para su estandarización al Object Management Group (OMG) en Enero de 1997, en respuesta a su solicitud de propuesta para un lenguaje estándar de modelado. Entre Enero y Julio de 1997, el grupo inicial de colaboradores se amplió para incluir prácticamente a todas las demás organizaciones que habían enviado

alguna propuesta o habían contribuido en alguna medida en las respuestas iniciales al OMG, incluyendo Andersen Consulting, Ericsson, Softeam, Sterling Software y Taskon entre otras. Se formó un grupo de trabajo para la semántica, para formalizar la especificación de UML con otros esfuerzos de estandarización. Una versión de UML, la versión 1.1, se ofreció al OMG para su estandarización en Julio de 1997. En Septiembre de 1997, esta versión fue aceptada por la OMG Análisis and Design Task Force (ADTF) y el OMG Architecture Board y se sometió al voto de todos los miembros del OMG. UML 1.1 fue adoptada por el OMG el 14 de Noviembre de 1997. El control de mantenimiento fue asumido por la OMG Revisión Task Force (RFT). La RTF publicó una revisión editorial, UML 1.2 en Julio de 1998. Posteriormente, publicó UML 1.3



Anexo B. Diagramas de UML.

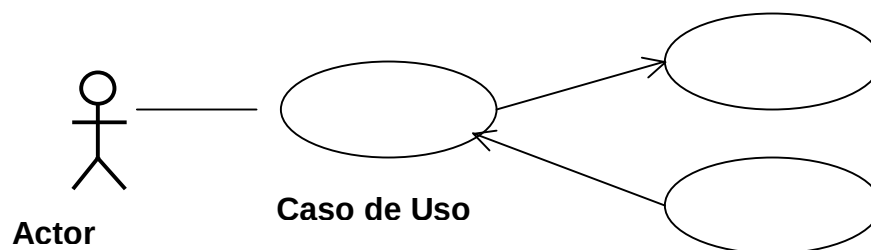
La siguiente figura muestra los diagramas de UML.



Diagramas de UML.

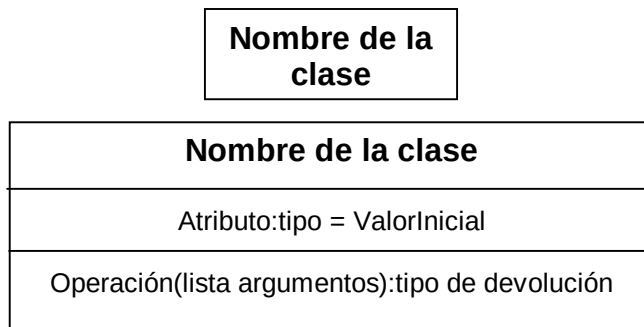
- *Diagramas de Caso de Uso:* muestran un tipo de caso de uso y actores (un tipo especial de clases) y sus relaciones. Cubren la vista de casos de uso estática de un sistema. Es la base para las pruebas del sistema.

Diagrama de caso de uso

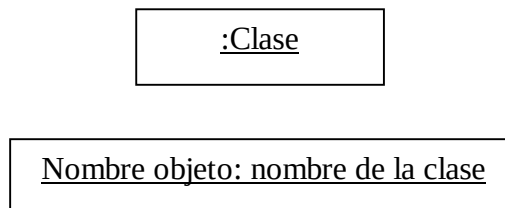


- *Diagramas de Clases:* muestran un conjunto de clases, interfaces y colaboraciones, así como sus relaciones. Estos diagramas son los diagramas más comunes en el modelo de sistemas orientados a objetos. Muestran la implementación de los componentes del software.

Clase

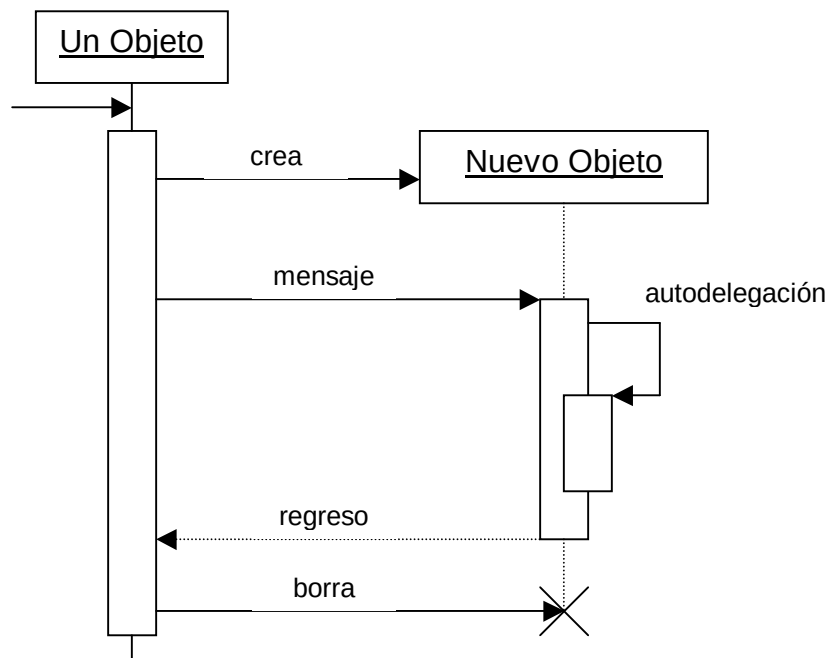


- *Diagramas de Objetos:* muestran un conjunto de objetos y sus relaciones. Los diagramas de objetos representan instantáneas de instancias de los elementos encontrados en los diagramas de clases.



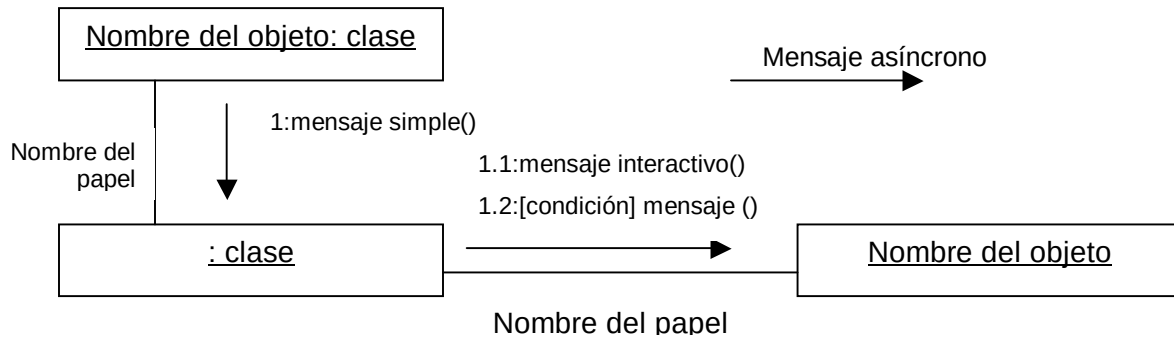
- *Diagramas de Secuencia:* es un diagrama de interacción que resalta la organización estructural de los objetos que envían y reciben mensajes.

Diagrama de secuencia



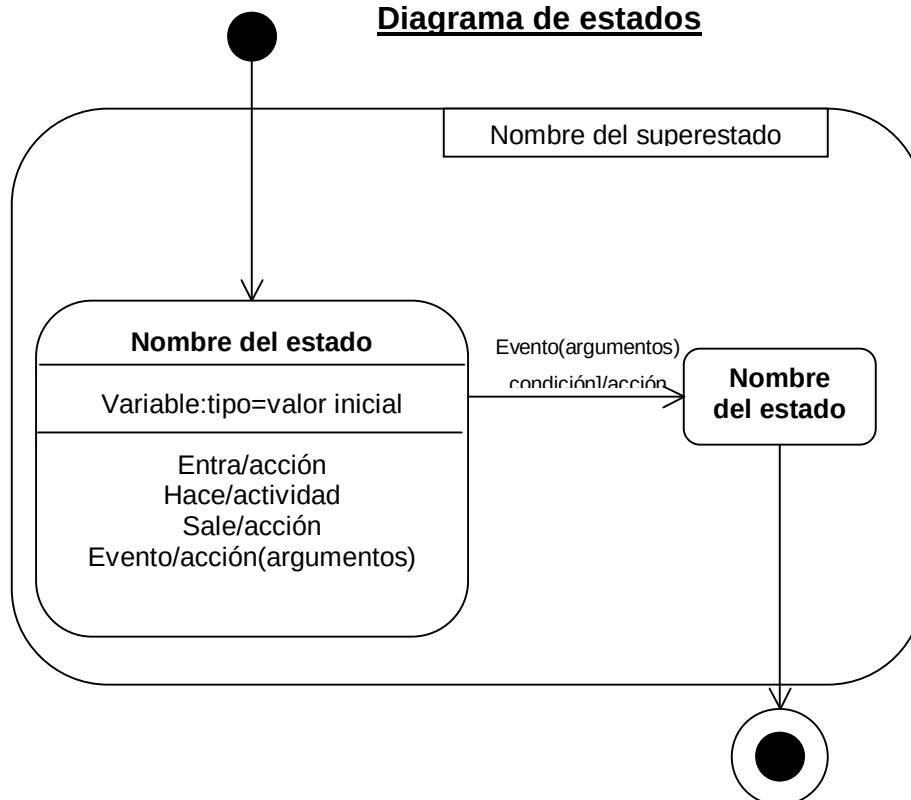
- *Diagramas de Colaboración*: es un diagrama que destaca la organización de los objetos que participan en una interacción.

Diagrama de Colaboración

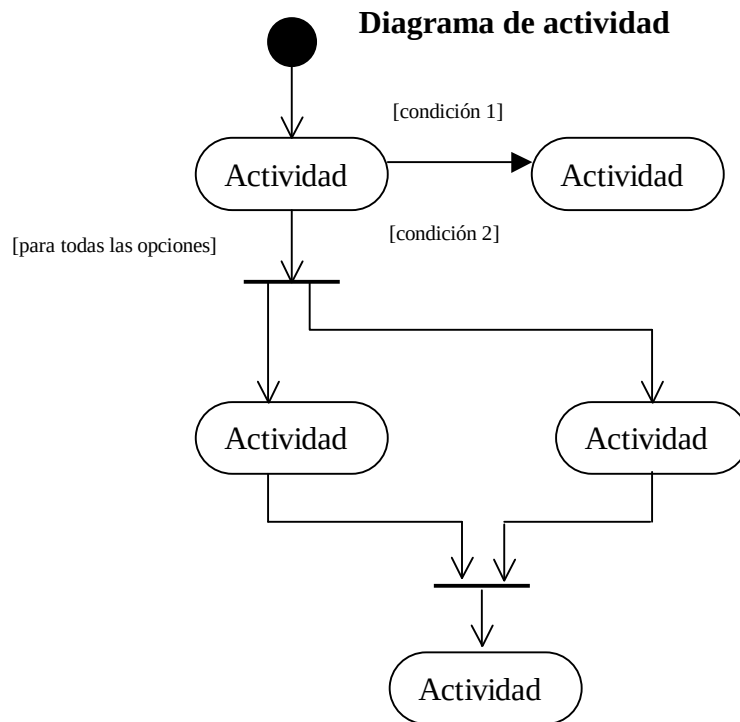


- *Diagrama de Estados*: muestran una máquina de estados, que consta de estados, transiciones, eventos y actividades. Muestran como un objeto particular se comporta en los casos de uso.

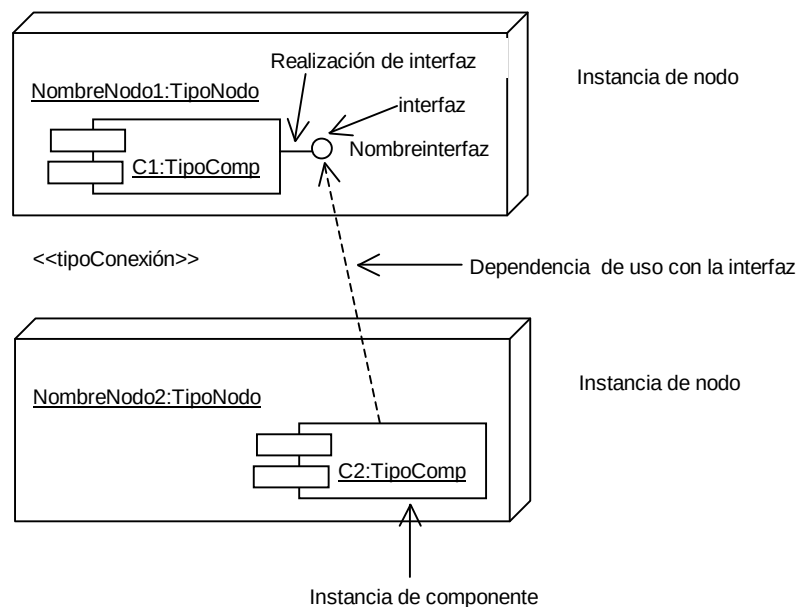
Diagrama de estados



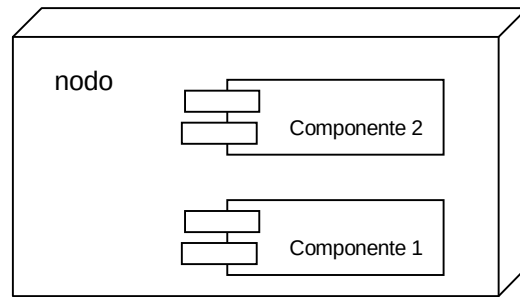
- *Diagrama de Actividades:* es un tipo especial de diagrama de estados que muestra un flujo de actividades dentro de un sistema. Muestra el comportamiento dentro de una estructura de control.



- *Diagrama de Componentes:* Describen los elementos físicos del sistema y sus relaciones, muestran las opciones de realización, incluyendo código fuente, binario y ejecutable.



- *Diagrama de Emplazamiento ó Implementación:* muestran los dispositivos que se encuentran en un sistema y su distribución en el mismo. Presentan la disposición física de los distintos nodos que componen un sistema y el reparto de los componentes sobre dichos nodos.



Anexo C. Diseño de Entrevista semiestructurada.

Entrevista a los Roles

Director.

Representante de la dirección.

Administrador.

Técnico Calibrador.

1. ¿En qué consiste el sistema de calidad?
2. ¿Cuál es su participación en el sistema de calidad?
3. ¿De qué actividad es usted responsable del sistema de calidad?
4. ¿Qué información necesita para realizar estas actividades?
5. ¿Cómo lleva a cabo el control de información requerida y generada de estas actividades?
6. ¿Ve usted algún inconveniente en la forma como trabaja, en cuanto al manejo de información?
7. ¿Cree usted que si contara con un sistema de información automatizado, se podrían resolver dichos inconvenientes?
8. ¿Qué características o requerimientos necesita usted que cumpla el sistema de información automatizado para apoyarla en sus actividades?
9. ¿Qué procedimientos o tareas realizará una vez que se instale el nuevo sistema?

Forma para la entrevista.

Persona entrevistada. _____ Puesto. _____

Tema. _____

Entrevistado por. _____ Puesto. _____

Fecha. _____ Hora de inicio. _____

Tiempo asignado. _____ Lugar. _____

BIBLIOGRAFÍA Y REFERENCIAS.

Booch, G. (1999). The Unified Modeling Language User Guide. Estados Unidos. Addison Wesley.

Figuerola, C (1995). Evaluación de la calidad y gestión del cambio. México. ANUIES. Pág. 54.

Fowler, M. (1999). UML gota a gota. México. Addison Wesley. Pág. 5, 16, 22.

Gaceta Universitaria (2003). Destaca Calidad de la UABC a nivel nacional. México. UABC. Pág. 10.

Gutiérrez, P. (2002). Calidad Total y Productividad. México. McGraw Hill. Pág. 274.

IMNC, Instituto Mexicano de Normalización y Certificación. (2003). México. Disponible en <http://uninet.mty.itesm.mx/legis-demo/indices/indcc.htm>. Fecha de consulta: 15 Mayo 2003.

Jacobson, I (2000). El proceso Unificado de Desarrollo de Software. Addison Wesley. Pág. 29, 25.

James, P. (1997). Gestión de la Calidad Total. España. Prentice Hall Iberia. Pág. 32.

Kendall, K. (1997). Análisis y diseño de sistemas. México. Pearson Educación.

Lafourcade, P. (1993). Planeamiento, Conducción y evaluación de la enseñanza superior. Argentina. Kapelusz. Pág. 95.

Lardent, A. (2001). Sistemas de Información para la Gestión empresarial. Argentina. Prentice Hall. Pág. 30.

Levine, G. (2001). Computación y programación moderna, perspectiva integral de la informática. México. Pearson Educación.

Mariño, P. (2003). Las Comunicaciones en la Empresa. México. Alfaomega. Pág. 41.

Martin, J. (1992). Análisis y Diseño Orientado a Objetos. México. Prentice Hall Hispanoamericana . Pág. 4, 185.

McLeod, R. (2000). Sistemas de Información gerencial. México. Prentice Hall. Pág. 197.

Moreno, M. (2001). Gestión de la Calidad y Diseño de Organizaciones. España. Prentice Hall. Pág. 391, 81.

- Munguia B. (2001). La calidad al alcance de todos. México. Pág. 5.
- Pressman, R. (2002). Ingeniería del Software, un enfoque práctico. México. McGraw Hill. Pág., 190, 63, 200.
- Rothery, B. (1992). ISO 9000, la norma y su implementación. México. Panorama. Pág. 30.
- Sandholm, L. (1995). Control total de Calidad. México. Trillas. Pág. 60.
- Schmuller, J. (1999). Teach Yourself UML. Estados Unidos. Pág. 7.
- Stair, R. (2000). Principios de sistemas de información. México. International Thomson. Pág. 412.
- UABC Universidad Autónoma de Baja California. (1997). Foro Académico, el ejercicio profesional ante el tratado de libre comercio. México. UABC. Pág. 101.
- UABC Universidad Autónoma de Baja California. (2003). Informe de rectoría 2003. México. Disponible en <http://insting.mxl.uabc.mx/www/documentos/informe2003.pdf>. Fecha de consulta: 20 Marzo 2005.
- UNIT Instituto Uruguayo de Normas Técnicas (2003). ¿Qué es ISO?. Uruguay. Disponible en <http://www.unit.org.uy/Acerca/iso.asp>. Fecha de consulta: 15 Marzo 2005.