

PONTIFICIA UNIVERSIDAD CATOLICA DE VALPARAISO

FACULTAD DE INGENIERIA

ESCUELA DE INGENIERIA INFORMATICA

**Sistema de Control de Archivos para el Departamento de
Tránsito y Transporte Público de la Ilustre Municipalidad de
Viña del Mar**

Profesor Guia :

Cristian Alexandru Rusu

INFORME FINAL DEL PROYECTO PARA OPTAR AL TITULO

PROFESIONAL DE

INGENIERO DE EJECUCION EN INFORMATICA

JORGE RENE MUÑOZ SILVA - FELIPE ANDRES RIVERA CID

Diciembre 2005

*A nuestros padres por habernos apoyado en todo este proceso.
Al cuerpo docente por prestar sus servicios de manera tan dedicada.*

*Jorge Rene Muñoz Silva
Felipe Andrés Rivera Cid*

AGRADECIMIENTOS

A nuestros padres y familia por el constante apoyo brindado durante estos años.

A la Pontificia Universidad Católica de Valparaíso por brindarnos una excelente educación y valores basados en el espíritu cristiano.

A nuestro profesor guía Alexandru Cristian Rusu y profesor correferente Pamela Hermosilla Monckton por toda la ayuda, tiempo y gran calidad humana.

Al departamento de tránsito de la Ilustre Municipalidad de Viña del Mar por la predisposición, tiempo y los recursos otorgados para la realización de este proyecto.

Sin estas personas este proyecto no podría haber llegado a término.

RESUMEN

El presente documento corresponde al informe final del proyecto para optar al título de Ingeniero ejecución en informática, llamado “Sistema de control de archivos para el departamento de

transito y transporte de la I. Municipalidad de Viña del Mar”, el cual tiene como objetivo solucionar un problema específico que se presenta dentro del departamento

En la sección de procesamiento de datos y archivos del departamento de transito, se mantiene toda la información sobre los vehículos que están inscritos en esta municipalidad, manteniendo un registro histórico el cual es almacenado a través de un sistema de carpetas.

El problema surge debido a la enorme cantidad de información y actualización, la cual requiere de un constante movimiento de las carpetas lo que conlleva a un desorden dentro de la organización de las mismas. Este sistema informático soluciona esta situación implementando un control lógico de las carpetas físicas junto con proporcionar parte de la información que estas contienen.

PALABRAS CLAVES: Sistema de información, Ingeniería de software, Control de archivos, Transito y transporte, Pruebas.

ABSTRACT

The following document corresponds to the final project with the core principle of obtaining the degree of “ingeniero ejecución en informática”, called “Sistema de control de archivos para el departamento de transito y transporte de la I. Municipalidad de Viña del Mar”. The main objective of the present project is to solve a specific problem occurring in the department.

In the section of the file and data processing of the auto traffic department, all the information about vehicles that are registered in the municipality is filed, keeping a historical register which is stored through a system of files.

The problem is related to the enormous amount of information and actualization, which requires a constant movement of files, which leads to a disorder within the organization of the files. This computer system solves the file issue by implementing a logic control of the physic files and providing part of the information that is stored in the files.

KEY WORD: Information system, software engineering, file control, auto traffic and transport, tests.

1. INTRODUCCION

El presente documento da a conocer todo el proceso realizado durante el desarrollo de este proyecto informático, el cual propone una mejora al sistema actual de trabajo que se realiza dentro de la sección de procesamiento de datos y archivo perteneciente al departamento de tránsito y transporte público, Ilustre Municipalidad de Viña del Mar.

Descripción de la organización

La I. Municipalidad de Viña del Mar, a través de su Departamento de Tránsito y Transporte Público, cumple con la aplicación de la Ley 18290. Ahora bien, el Departamento de Tránsito Municipal a lo largo de su gestión ha intentado irse poniendo en la senda modernizadora en la cual se encuentra inserto casi todo el denominado “Sector Público” en Chile.

El Departamento de Tránsito de la I. Municipalidad de Viña del Mar, tiene como misión propender a la entrega de servicios en estas materias, lo más eficiente y dinámica posible. En este sentido su Misión será la entrega en las condiciones arriba señaladas de Permisos de Circulación, Licencias de Conducir, Fiscalización de Tránsito, Asesoría, Regulación. Lo anterior como parte del subsistema mayor denominado I. Municipalidad de Viña del Mar.

A continuación se presenta la estructura organizacional del Departamento de Tránsito y Transporte Público:



Figura 1.1: Organigrama del departamento de tránsito y transporte

Situación actual de la empresa

En el presente proyecto se trabajara dentro de la sección de procesamiento de datos y archivos del departamento de tránsito, la cual tiene como objetivos:

- Mantenición de Bases de Datos sobre eventos, hechos, relativos al tránsito.
- Seleccionar, analizar, y procesar la información contenida en un documento para su recuperación.
- Coordinar y controlar el presupuesto asignado al Departamento. de Tránsito para proyectos de gestión de tránsito.

Actualmente esta sección cuenta con un sistema para el control de las carpetas las cuales contienen los archivos físicos necesarios para el trabajo dentro del departamento del tránsito (como por ejemplo permiso de circulación, seguro obligatorio, sello de normas de emisión, entre otros), el cual consiste en una serie de planillas Excel, las cuales siguen el siguiente formato:

Tabla 1.1: Formato planillas Excel

Patente	Ultimo Año Pagado	Archivo		Carpetas Sacadas Del Archivo		Carpetas Devueltas	Observaciones.	Motos y acoplados	Digitador	Fecha
		Activo	Pasivo	Por	En Fecha	En Fecha				
1	2	3	4	5	6	7	8	9	10	11

A continuación se procede a dar una pequeña descripción de los campos que contiene la tabla 1.1

Patente	:	Patente de Carpeta.		
Ultimo Año Pagado	:	Ultimo Boletín Pagado (Que está en carpeta)		
Archivo	}	Activo	:	Se coloca una X sí la carpeta esta en Tránsito.
		Pasivo	:	Se coloca una X sí la carpeta está en Container ó viene de él
Carpetas Sacadas Del Archivo	}	Por	:	Funcionario que la pide.
		En fecha	:	Fecha en que es retirada.
Carpetas Devueltas En Fecha	:	Fecha en que la carpeta fue devuelta.		
Observaciones	:	Si es Fondo Propio, porque sale, si es para enviar a container, si esta en Kardek, etc.		
Motos y Acoplados	:	Si es Moto ó Acoplado. (Columna Oculta)		
Digitador	:	Quien digita la información.		
Fecha	:	Fecha en que digita la información.		

Toda la información acerca de las salidas se recauda en un libro durante la jornada de trabajo y las entradas se dejan en una caja que se encuentra en la sección de archivo, siendo transferido a la planilla Excel correspondiente al finalizar la jornada (esta actividad se realiza todos los días).

Además, manejan otro sistema que maneja la información detallada de los documentos contenidos en las carpetas, pero este sistema es administrado por una empresa externa (PROEXIS) por lo cual no se tendrá acceso a este.

Debido a esto, alguna información se encuentra en ambos sistemas, por ejemplo: patente, fecha de pago, etc.

Esto ultimo no refleja un problema de duplicación de información, ya que el sistema externo muchas veces tiene problemas, por lo que la información no esta disponible a tiempo y se requiere otra forma para acceder a ella.

Proceso actual de control de archivos

El sistema que maneja actualmente la sección de procesamiento de datos y archivos posee varias deficiencias debido mayormente al volumen y a la importancia de la información que se maneja, siendo la mas critica de ellas la completa falta de seguridad dentro del sistema. Además se han encontrado otras deficiencias durante la recolección de información, las cuales son:

La información se maneja en planillas Excel, por el número limitante de filas de una planilla.

Debido a lo anterior se hace engorroso el insertar una nueva fila a la planilla, por el hecho de tener que estar buscando la ubicación en donde se debe colocar la fila (para mantener el orden alfabético).

No posee un bloqueo de las carpetas, es decir, no muestra cuales carpetas no deberían salir del departamento debido a ciertos problemas, lo cual hace que algunas carpetas no se encuentren cuando son requeridas dentro del departamento.

La búsqueda de datos específicos puede llevar mucho tiempo, debido a la cantidad de información.

La generación de informes se tiene que delegar a otra sección, lo cual puede llevar días la entrega de estos.

Dados todas estas falencias, el principal problema a resolver es el manejo y control de la información que se encuentra en la sección de archivos.

El sistema a desarrollarse será muy importante para el departamento, ya que facilitara enormemente la búsqueda de información junto con el manejo lógico de las carpetas físicas. En otras palabras, si este sistema no se aplicara seria problemático encontrar las carpetas ya que estas son movidas continuamente y la información sobre esta no se podría obtener de manera rápida.

Solución propuesta

Se propone desarrollar un prototipo de sistema mediante el uso de herramientas tecnológicas actuales, que nos permitan un manejo fluido y fácil de la información. Esta alternativa nos presenta las ventajas de utilizar una base de datos, la cual nos otorga la facilidad de manejar toda la información de manera consistente y segura, junto con interfaces correctamente diseñadas,

fáciles de utilizar y que permitan el correcto procesamiento de la información. Todo esto junto con cumplir los requerimientos del cliente.

Esta alternativa traerá mejoras en el aspecto tecnológico (mayor seguridad que otorga las nuevas herramientas, resistencia contra fallos, nuevas funcionalidades) como también en el humano (disminución de tiempo en actividades de búsqueda e ingreso (entre otras), satisfacción en el manejo del sistema).

2. OBJETIVOS

2.1 Objetivo General

Desarrollar un prototipo de sistema que gestione el almacenamiento y la ubicación de las carpetas que contienen documentos específicos para la dirección del tránsito de la Ilustre municipalidad de Viña del Mar.

2.2 Objetivos específicos

Gestionar las entradas y salidas diarias de las carpetas en la sección de archivos.

Facilitar el trabajo de la sección de archivos de la dirección del tránsito.

Mejorar los tiempos de búsqueda de información.

Ofrecer un nivel de seguridad adecuado para la información requerida

Generar informes estadísticos y gráficos según el requerimiento del usuario.

3. ESTUDIO DE FACTIBILIDAD

3.1 Factibilidad técnica

El sistema que se va a realizar debe poseer la capacidad técnica de manejar una gran cantidad de datos. Por lo tanto el departamento de tránsito debe poseer el equipamiento adecuado para soportar las necesidades básicas del sistema a construir.

Para esto, se necesita computadores que puedan soportar una aplicación cliente que maneje las interfaces que interactuaran con el usuario para el manejo de los datos dentro del sistema. Junto con esto se necesita una red interna que conecte estos computadores con el servidor de datos que se posee y para finalizar un servidor lo suficientemente capaz de soportar el sistema administrador de base de datos SQL Server 7 (referencia capítulo 10).

Las dependencias del departamento de tránsito poseen computadores con suficiente capacidad para manejar estas aplicaciones sin ningún contratiempo y estos están conectados mediante la red interna que los enlaza con el servidor. Las características de los computadores se describen a continuación:

Procesador :	Pentium II 550MHz
Memoria RAM :	128 MB
Disco duro :	10 GB
Sistema operativo:	Microsoft Windows 2000 service pack 4
Tarjeta de red:	D-Link 100 MB

La red que se posee dentro de las instalaciones pertenece a la categoría 5 de par trenzado, 100 base-Tx. La cual soporta todo el flujo de información que es necesario para el correcto funcionamiento del sistema.

Dado que todos los componentes técnicos se poseen y son aptos para el proyecto, se puede determinar que es factible técnicamente.

3.2 Factibilidad Operacional

Como se describió anteriormente, el departamento de tránsito posee equipos computacionales adecuados para soportar el sistema y a los empleados se les ha capacitado para utilizar estas herramientas computacionales (Excel, Word, etc.), por lo tanto se posee el personal para el manejo de este.

El nuevo sistema mantendrá la metodología básica de trabajo del usuario (por ejemplo: ingreso de datos en letras mayúsculas, visualización básica de los datos, formato de los datos) y cambiara otros aspectos con el fin de mejorar el rendimiento y evitar errores masivos de datos (restricciones en las interfaces, pantallas propias para cada funcionalidad, nombres representativos para cada campo).

A pesar de esto, se requerirá de una pequeña capacitación la cual les otorgara los conocimientos básicos de manejo del sistema a realizarse. Junto con esto, el sistema poseerá un manual de usuario con las ayudas necesarias para que el personal posea un completo control sobre el sistema. Dado la capacitación a implementar, junto con la herramienta de ayuda que se va a entregar, se puede determinar que es factible operacionalmente.

3.3 Factibilidad legal

Dentro del departamento de tránsito, se posee la licencia para desarrollo y uso de la herramienta de administración de base de datos SQL Server 7. En lo referente a la herramienta Visual Basic, será necesaria la compra de la licencia para desarrollo y uso.

Junto con lo anterior, el departamento de tránsito tiene que cumplir con la aplicación de la ley 18290, la cual es la ley de tránsito (publicada en el 07 / 02 / 1984), en este aspecto de la ley tiene

que cumplir lo que es referente al manejo de patentes, documentos de estas, por lo que el sistema a desarrollarse debe cumplir con esta ley en todos sus aspectos. Dado el cumplimiento de las leyes y la adquisición de las licencias requeridas, se puede determinar que es factible legalmente.

3.4 Factibilidad económica

En cuanto a la factibilidad económica, el departamento de tránsito tendrá una gran cantidad de beneficios ya que se mejorara el manejo de los datos y el tiempo de trabajo, pudiendo destinar mayores esfuerzos en otras áreas que requieran mayor atención. Esto se puede demostrar según los datos expuestos a continuación:

Sueldo promedio de la sección de archivos (5)	Promedio horas trabajo por hombre (mensual)	Costo hora/hombre	Vida útil proyecto	TRMA para proyectos sociales (MIDEPLAN)	Utilización promedio del sistema
\$ 700.000	180 horas	\$ 3.900	5 años	8 %	68 horas mensuales

Tabla 3.1: Descripción de los datos cuantificables

Capacitación: 1 hora diaria por una semana ((5 personas * 5 horas) * \$ 3.900) = ~ \$ 100.000

Inversión inicial: \$ 1.100.000 (licencia Visual Basic.NET) + \$ 100.000 (capacitación) = \$ 1.200.000

Beneficios: se estimo que se rebajara en un 35% menos el tiempo de utilización del sistema, comparado con el antiguo

Ahorro: \$265.200 (68*3900) - \$ 89.700 ((%35 de 68)*3.900) = \$ 175.500 mensual

Ahorro anual: \$ 2.106.000

VAUE = - \$ 1.200.000 *(A/P, 8%, 5) + \$ 2.106.000 (- Inversión inicial* flujo + Ahorro anual)

VAUE = 1.805.448

Si el VAUE es mayor a cero el proyecto es conveniente, por lo tanto se puede determinar que es factible económicamente (los valores expuestos anteriormente son aproximaciones).

4. ESTUDIO DE LOS PARADIGMAS DE DESARROLLO

4.1 Ciclo de vida clásico

Este paradigma sugiere un enfoque sistemático y secuencial para el desarrollo de software. Este paradigma comienza en un nivel de ingeniería de sistemas y progresa con el análisis, diseño, codificación, pruebas y mantenimiento, para llegar a un resultado [4], [5].

- **Ingeniería y modelado de sistemas/información:** Consiste en establecer los requisitos de todos los elementos del sistema, los cuales se obtienen de las áreas de nivel estratégico y nivel de negocio de una empresa, y asignando al software algún subgrupo de estos.
- **Análisis:** En esta parte del proceso se intensifica la recolección de requerimientos y se centran en el software para comprender la naturaleza de (los) programa(s) a construir, el dominio de la información, la función requerida, el comportamiento, el rendimiento y la interconexión de los mismos.
- **Diseño:** El proceso de diseño traduce los requisitos en una representación del software donde se puede evaluar su calidad antes de que comience la codificación.
- **Generación de código:** Este paso realiza la tarea de traducir el diseño a una forma legible por la maquina, si el diseño es llevado detalladamente, la generación de código se realiza mecánicamente.

- **Pruebas:** Se realizan las pruebas para la detección de errores y para asegurar que la entrada definida produce resultados reales de acuerdo con lo requerido.
- **Mantenimiento:** El software indudablemente sufrirá cambios después de ser entregado al cliente, ya sea por errores encontrados, por cambios en la empresa, etc. Por este motivo en esta fase se vuelven a aplicar las etapas precedentes a un programa ya existente.

Ventajas:

El proceso de desarrollo del software es bastante rápido.

Entrega una facilidad de planificación y control durante sus fases.

Otorga un marco de referencia para el análisis, diseño, codificación, pruebas y mantenimiento de un proyecto de software.

Desventajas:

Los requerimientos no siempre están claros al comienzo del proceso, lo que es necesario para este paradigma.

No hay interacción con el cliente, salvo en la recolección de requerimientos.

Los errores detectados en forma tardía producen un alto costo al proyecto.

4.2 Construcción de prototipos

Es un paradigma iterativo, que se basa en la construcción y refinamiento de prototipos funcionales que se van entregando al cliente para satisfacer los requerimientos de este. Consta de tres fases sobre las cuales se realizan las iteraciones para llegar a un resultado [4], [5]. Estas fases son:

- **Escuchar al cliente:** se realiza la recolección de requerimiento en donde el desarrollador y el cliente definen los objetivos globales para el software, identifican los requisitos conocidos y las áreas del esquema en donde es obligatoria más definición.

- **Construir/Revisar el prototipo:** En esta fase se realiza un diseño rápido, el cual se centra en los aspectos del software que serán visibles para el usuario. El diseño rápido lleva a la construcción de un prototipo funcional del sistema.
- **El cliente prueba el prototipo:** El prototipo es evaluado por el cliente, lo que utiliza para refinar los requerimientos dados y pasar luego a la segunda fase.

Ventajas:

Existe una buena interacción con el cliente.

No es necesario tener los requerimientos claros en un comienzo del proyecto, esto debido a que es un paradigma iterativo.

Este paradigma se puede utilizar en conjunto con otros paradigmas para obtener un mejor resultado.

Desventajas:

El cliente puede creer que se puede pasar del prototipo a la versión final del software en forma rápida.

El desarrollador, a menudo, hace compromisos de implementación para hacer que el prototipo funcione rápidamente, los cuales no siempre se pueden cumplir.

Se puede caer en un error al escoger las herramientas de desarrollo debido a que se debe preparar un prototipo para ser evaluado por el cliente de forma rápida.

4.3 Modelo espiral

Es un paradigma evolutivo de desarrollo de software que conjuga la naturaleza iterativa de construcción de prototipos con los aspectos controlados y sistemáticos de ciclo de vida clásico, el cual proporciona el potencial para el desarrollo rápido de versiones incrementales del software [4], [5]. Este paradigma se divide en seis actividades:

- **Comunicación con el cliente:** Tareas requeridas para establecer la comunicación entre el cliente y el desarrollador.
- **Planificación:** Tareas requeridas para definir recursos, tiempo y otras informaciones relacionadas con el proyecto.
- **Análisis de riesgos:** Se evalúan los riesgos técnicos y de gestión.
- **Ingeniería:** Tareas requeridas para construir una o mas representaciones de la aplicación
- **Construcción y acción:** Tareas requeridas para construir, probar, instalar y proporcionar soporte al usuario.
- **Evaluación del cliente:** El cliente evalúa las representaciones del software creadas en las etapas de ingeniería e implementadas durante la etapa de instalación.

Ventajas:

Existe una buena interacción con el cliente

No es necesario tener los requerimientos claros en un comienzo del proyecto, esto debido a que es un paradigma iterativo.

Al ir iterando se van descubriendo y entendiendo los riesgos mas importantes del proyecto.

Desventajas:

Para proyectos demasiado grandes el paradigma puede ser confuso y difícil de controlar.

Requiere una considerable habilidad de evaluación de riesgos, por lo mismo si no se descubren riesgos importantes en forma temprana puede producir problemas.

4.4 Proceso unificado de desarrollo (RUP)

Además de ser un paradigma de desarrollo de software basado en la metodología de orientación a objeto, RUP es un marco de trabajo genérico capaz de especializarse en una gran variedad de sistemas, diferentes áreas de aplicación, diferentes tipos de organizaciones, diferentes niveles de aptitud y diferentes tamaños de proyectos. RUP está compuesto por un conjunto de actividades necesarias para pasar de los requisitos dados por el cliente a un producto de software [3], [8].

Está basado en componentes, esto significa que el sistema en construcción está formado por componentes software interconectados a través de interfaces bien definidas. Utiliza normalmente el Lenguaje Unificado de Modelado (Unified Modeling Language, UML) para preparar todos los esquemas del modelo. Los verdaderos aspectos definitorios se resumen en tres frases clave; * dirigido por casos de uso, * centrado en la arquitectura, * iterativo e incremental. Esto es lo que hace único al Proceso Unificado (lo que no significa que este paradigma no pueda utilizar otras herramientas para el modelado).

Está dirigido por casos de uso. Un sistema debe brindar servicios a sus usuarios. Para que sea exitoso, se debe conocer lo que sus futuros usuarios necesitan y desean. Usuario: representa alguien o algo que interactúa con el sistema en desarrollo. Los casos de uso representan los requisitos funcionales. Todos juntos constituyen el modelo de casos de uso, que describe la funcionalidad completa del sistema. Los casos de uso deben contestar a la pregunta ¿Qué debe hacer el sistema, para cada usuario?.

Está centrado en la arquitectura. El papel de la arquitectura es parecido al que juega en la construcción de edificios. El concepto de arquitectura de software incluye los aspectos estáticos y dinámicos más significativos del sistema. Surge de las necesidades de la empresa, de la percepción de los usuarios e inversores. Se ve influenciada por factores como la plataforma sobre la que funcionará el software (hardware, sistema operativo, gestión de base de datos, protocolos de comunicaciones), bloques de construcción reutilizables (marco de trabajo para interfaces gráficas), consideraciones de implementación, sistemas heredados y requisitos no funcionales (rendimiento, fiabilidad). Es una vista de alto nivel del diseño completo. Su valor depende de las personas responsables de su creación. Cada producto tiene tanto una función como una forma, o

arquitectura, que debe diseñarse para permitir que el sistema evolucione desde el inicio y a lo largo de futuras iteraciones.

Es iterativo e incremental. Todos sabemos que el desarrollo supone un gran esfuerzo que puede durar meses o años, por lo que es práctico dividirlo en partes más pequeñas o mini proyectos. Cada mini proyecto es una iteración que resulta en un incremento. Las iteraciones hacen referencia a pasos en el flujo de trabajo, y los incrementos, al crecimiento del producto. Las iteraciones deben estar controladas para mayor efectividad, esto es que deben seleccionarse y ejecutarse en forma planificada. La selección de las iteraciones se basa en el tratamiento de un grupo de casos de uso, que amplían la utilidad del producto y los riesgos más importantes. En cada iteración, los desarrolladores identifican y especifican los casos de uso relevantes, crean un diseño utilizando la arquitectura guía, implementan el diseño mediante componentes, y verifican que los componentes satisfacen los casos de uso. Si una iteración cumple con sus objetivos, el desarrollo continúa con la siguiente. Cuando esto no ocurre se deben revisar las decisiones previas y probar con un nuevo enfoque.

Beneficios de una iteración controlada:

Reduce el costo del riesgo al costo de un solo incremento.

Se identifican los riesgos en fases tempranas.

Acelera el ritmo del esfuerzo de desarrollo en su totalidad.

Reconoce que las necesidades del usuario y sus requisitos no pueden definirse al principio.

El proceso unificado de desarrollo nos presenta una serie de fases en las cuales dividir el desarrollo del prototipo las cuales son: inicio, elaboración, construcción y transición. Dentro de las cuales se realizaran las iteraciones necesarias cumpliendo los hitos planteados para cada una de ellas. A continuación se describen las fases antes mencionadas junto con describir las acciones a realizar:

Fase de inicio: Comprende todas las actividades necesarias para definir el alcance del proyecto.

- **Recolección de información:** Se recolecta la información para luego analizarla y dar a conocer al cliente los mayores problemas junto con las posibles soluciones.

- **Definición de requerimientos:** Se le pide al cliente que presente lo que le gustaría que el nuevo sistema realizara, después del análisis de estos se definen los requerimientos.
- **Análisis y elección de paradigmas, metodologías y herramientas:** Después de tener listo los requerimientos, se procede a elegir las herramientas y métodos para llevar a cabo el proyecto, entre varias alternativas se realizan análisis y se elige cual es la mas optima.
- **Construcción casos de uso:** Los casos de uso nos permiten graficar las iteraciones entre los usuarios y el sistema. También nos permite ver que tipos de usuarios existen para luego clasificarlos.

Fase de elaboración: Se planifica en todos sus aspectos el proyecto junto con desarrollar la arquitectura de este.

- **Preparación de iteraciones:** Se analizan los casos de uso y se dividen estos en iteraciones para un mejor desarrollo de estos.
- **Iteraciones:** Se refinan lo requerimientos y se diseña el sistema para cada caso de uso dependiendo los usuarios.

Fase de construcción: Dada las especificaciones en el diseño, se construyen los módulos necesarios para el funcionamiento del sistema.

- **Construcción de módulos:** Después del diseño de cada modulo, se procede a la construcción de cada uno con sus respectivos casos de uso

- **Pruebas de los módulos:** Al termino de cada caso de uso se realizan pruebas unitarias para ver el correcto funcionamiento de cada uno y al finalizar el modulo se realizan pruebas de integración entre los casos de usos, para finalizar con pruebas de integración entre los módulos.

Fase de transición: Se entrega la primera versión del sistema para que el cliente lo pruebe.

Marcha blanca: Debido que el cliente posee otro sistema con el cual trabaja actualmente, el nuevo sistema trabajara a la par con el antiguo por un periodo de tiempo, el cual tiene como objetivo lograr descubrir falencias no encontradas en las pruebas.

Ventajas:

Permite modelar los requisitos de cada usuario en particular de una manera muy precisa.

Los riesgos más importantes se identifican en la etapa inicial del proyecto, por lo tanto en el desarrollo los riesgos que aparezcan serán más controlables.

Al estar dividido en iteraciones, cada iteración entrega un prototipo y en caso de acortarse el plazo de entrega del proyecto por lo menos se tendrá como producto un prototipo utilizable y no solamente algo en papel.

Desventajas:

Al realizarse prototipos en las iteraciones, se pueden malentender que el prototipo realizado es el producto final, llevando a confusiones al final del proyecto.

No es recomendable utilizar este paradigma en proyectos pequeños, ya que la realización de iteraciones junto con los modelos puede llevar a realizar tareas innecesarias, alargando la entrega del proyecto.

Complejidad de adaptación. No hay un libro único o guía sobre RUP y al haber muchos textos diferentes, puede llevar a confusión por parte del programador en saber cual es el mas ideal.

4.5 Elección del paradigma de desarrollo

Se escoge RUP para desarrollar el presente proyecto por las siguientes razones:

Por ser un paradigma que permite una buena interacción con el cliente, dando avances detallados de el trabajo realizado al cliente sin tener que recurrir necesariamente a prototipos funcionales como es el caso del paradigma de prototipos y el paradigma espiral mencionados anteriormente.

Por ser un paradigma adaptable a las necesidades de los proyectos, sin dejarse guiar por la envergadura de los mismos, a diferencia de los mencionados anteriormente.

Por permitir un manejo exhaustivo de las actividades que se desarrollan durante el proyecto, asignando tareas específicas a cada uno de los integrantes del grupo, tomando en cuenta los conocimientos, experiencia y especialidades de los desarrolladores.

Por ser un paradigma iterativo que además integra las ventajas del modelo lineal secuencial (o ciclo de vida clásico) dentro de cada una de sus iteraciones o fases.

Al ser una de sus características “iterativo e incremental” permite la detección de errores y riesgos importantes dentro del proyecto en forma temprana, de manera de no caer en estos en la fase de pruebas, lo que seria un costo adicional al proyecto en caso de suceder; Además de tener la posibilidad de ir agregando requerimientos adicionales durante el desarrollo del proyecto.

Además de las razones mencionadas anteriormente, se tomo en cuenta la experiencia de ambos integrantes del grupo de trabajo con RUP, lo cual se traduce en una ventaja dentro de lo planificado para el presente proyecto.

5. METODOLOGIA DE TRABAJO

5.1 Estructurada

Este es un enfoque que se utiliza para el análisis, diseño y codificación de sistemas. Una característica fundamental de este enfoque es el hecho de que el dominio del software se describe en términos de los procesos que lo componen, por lo que se utilizan algoritmos para describirlos y se implementan mediante funciones dentro del sistema.

Las ventajas más notorias de este enfoque son:

Una amplia difusión, por lo que es ampliamente conocido y utilizado.

Además, los procesos de análisis, diseño y codificación están muy bien definidos y acotados, por lo que se facilita el manejo de los recursos.

Pero este enfoque también posee grandes desventajas:

Los costos en que se incurre en caso de que el programa no cumpla con los requerimientos dados, ya que es bastante difícil realizar cambios cuando el proyecto se encuentra en fases muy avanzadas.

Además puede que para el grupo de trabajo sea imposible el entregar el producto terminado sin incurrir en retrasos, por lo que se tiene problemas cuando no se pueden realizar postergaciones en las fechas de entrega.

A la hora de codificar, al hacerlo de forma estructurada puede acarrear un problema, ya que los cambios realizados en funciones específicas del sistema pueden alterar el comportamiento del programa completo.

5.2 Orientación a objetos

El diseño orientado a objetos se basa en la idea de utilizar el ocultamiento de la información como principal criterio de descomposición y en la noción de los tipos de datos abstractos. En este

diseño, los componentes del software se ven más como objetos que como funciones o procesos. Cada objeto tiene un conjunto asociado de operaciones permitidas de acuerdo con la funcionalidad específica del programa de la cual están encargados, además estos objetos poseen la habilidad de recibir y enviar mensajes con lo cual interactúan entre sí para lograr su fin.

Durante la fase de análisis del proyecto se descubren los objetos necesarios para el correcto funcionamiento del sistema, los cuales se implementan por medio de clases en la etapa de diseño agregándose las clases que sean necesarias para facilitar la implementación del sistema a construir. Además de las clases que se necesitan, se realizan interfaces para mejorar la retroalimentación con el usuario del sistema lo cual mejora aún más las funcionalidades del sistema desarrollado.

Las ventajas más evidentes que posee este enfoque es el ocultamiento de información o encapsulamiento, el cual se utiliza para que los objetos solo conozcan la información que estos necesitan para desarrollar su labor y realizarla de la manera más autónoma posible, de esta manera los programas con errores en algunos objetos no necesitan alterar el comportamiento de todos los objetos del sistema para arreglarlos, sino solo aquellos en los que se han encontrado errores, lo que no debería afectar el funcionamiento de los demás objetos.

5.3 Elección de metodología de trabajo

Para el presente proyecto se decidió utilizar la metodología de orientación a objeto debido a las siguientes ventajas que posee esta metodología de trabajo en relación con el enfoque estructurado mencionado anteriormente:

El enfoque orientado a objeto proporciona una mejor forma de validar los requerimientos.

Al dividir el problema en objetos se simplifica su resolución, en comparación con la metodología estructurada que se resuelve el problema como un todo. (“Divide y Vencerás”).

Permite la iteración durante el desarrollo de un proyecto, lo que conlleva a un mejor entendimiento y resolución del sistema a desarrollar.

Además esta metodología va de la mano con la utilización de RUP como paradigma de desarrollo y UML como herramienta de modelado, lo que facilita aun más el desarrollo de un sistema informático.

6. HERRAMIENTAS DE DESARROLLO

6.1 Lenguaje Unificado de modelado UML

UML es una especificación para una notación orientada a objetos, la cual se basa en las anteriores especificaciones BOOCH, RUMBAUGH y COAD-YOURDON. Esta notación divide cada proyecto en un número de diagramas que representan las diferentes vistas del proyecto. Estos diagramas juntos son los que representa la arquitectura del proyecto [2], [13].

Con UML nos debemos olvidar del protagonismo excesivo que se le da al diagrama de clases, este representa una parte importante del sistema pero solo representa una vista estática, es decir, muestra al sistema de forma estática. Sabemos su estructura pero no sabemos que le sucede a sus diferentes partes cuando el sistema empieza a funcionar. UML introduce nuevos diagramas que representa una visión dinámica del sistema, gracias al diseño de la parte dinámica del sistema podemos darnos cuenta en la fase de diseño de problemas de la estructura al propagar errores o de las partes que necesitan ser sincronizadas, así como del estado de cada una de las instancias en cada momento.

El diagrama de clases continua siendo muy importante, pero se debe tener en cuenta que su representación es limitada, y que ayuda a diseñar un sistema robusto con partes reutilizables, pero no a solucionar problemas de propagación de mensajes ni de sincronización o recuperación ante estados de error. En resumen, un sistema debe estar bien diseñado, pero también debe funcionar bien.

UML también intenta solucionar el problema de propiedad de código que se da con los desarrolladores, al implementar un lenguaje de modelado común para todos los desarrollos se crea una documentación también común, que cualquier desarrollador con conocimientos de UML será capaz de entender, independientemente del lenguaje utilizado para el desarrollo.

UML es ahora un estándar, no existe otra especificación de diseño orientado a objetos, ya que es el resultado de las tres opciones existentes en el mercado. Su utilización es independiente del

lenguaje de programación y de las características de los proyectos, ya que UML ha sido diseñado para modelar cualquier tipo de proyectos, tanto informáticos como de arquitectura, o de cualquier otro ramo. Además permite la modificación de todos sus miembros mediante estereotipos y restricciones. Un estereotipo nos permite indicar especificaciones del lenguaje al que se refiere el diagrama de UML. Una restricción identifica un comportamiento forzado de una clase o relación, es decir mediante la restricción estamos forzando el comportamiento que debe tener el objeto al que se le aplica.

Se dispone de dos tipos diferentes de diagramas: los que dan una vista estática del sistema y los que dan una visión dinámica. Los diagramas estáticos son:

- **Diagrama de clases:** Muestra las clases, interfaces, colaboraciones y sus relaciones. Son los más comunes y dan una vista estática del proyecto.
- **Diagrama de objetos:** Es un diagrama de instancias de las clases mostradas en el diagrama de clases. Muestra las instancias y como se relacionan entre ellas. Se da una visión de casos reales.
- **Diagrama de componentes:** Muestran la organización de los componentes del sistema. Un componente se corresponde con una o varias clases, interfaces o colaboraciones.
- **Diagrama de despliegue:** Muestra los nodos y sus relaciones. Un nodo es un conjunto de componentes. Se utiliza para reducir la complejidad de los diagramas de clases y componentes de un gran sistema. Sirve como resumen e índice.
- **Diagrama de casos de uso:** Muestran los casos de uso, actores y sus relaciones. Muestra quien puede hacer que y las relaciones que existen entre acciones (casos de uso). Son muy importantes para modelar y organizar el comportamiento del sistema.

Lo diagramas dinámicos son:

- **Diagrama de secuencia, Diagrama de colaboración:** Muestran a los diferentes objetos y las relaciones que pueden tener entre ellos, los mensajes que se envían entre ellos. Son dos diagramas diferentes, que se puede pasar de uno a otro sin pérdida de información, pero que nos dan puntos de vista diferentes del sistema. En resumen, cualquiera de los dos es un Diagrama de Interacción.
- **Diagrama de estados:** Muestra los estados, eventos, transiciones y actividades de los diferentes objetos. Son útiles en sistemas que reaccionen a eventos.
- **Diagrama de actividades:** Es un caso especial del diagrama de estados. Muestra el flujo entre los objetos. Se utilizan para modelar el funcionamiento del sistema y el flujo de control entre objetos.

Como podemos ver el número de diagramas es muy alto, en la mayoría de los casos excesivos, y UML permite definir solo los necesarios, ya que no todos son necesarios en todos los proyectos. Pero para efecto del presente proyecto, se utilizaran solo los siguientes diagramas: Diagrama de casos de uso, Diagramas de secuencias de sistema, Diagramas de actividades, Diagrama de clases y Diagrama de componentes.

Entre las ventajas mas notorias que posee este lenguaje destaca una notación estándar y modelos simples de comprender, los cuales son bastante útiles para comprender no solo los requerimientos, sino también el ámbito del sistema a desarrollar y los aspectos funcionales del mismo, por lo cual se decidió utilizar esta herramienta en conjunto a RUP para la realización del sistema requerido.

6.2 Sistema administrador de bases de datos (SABD)

6.2.1 SQL Server 7

Es un sistema administrador de bases el cual proporciona un ambiente grafico de fácil uso y adaptable a las necesidades de los usuarios y administradores, además de otras aplicaciones para el mejor manejo de los recursos del sistema y las funcionalidades existentes. SQL Server 7 trabaja con un lenguaje procedimental (Transact SQL) el cual se maneja a través de un intérprete incluido en el sistema y posee la ventaja de utilizar varios estándares y utilizarse en un lenguaje fácil de entender.

Transact-SQL es fundamental para trabajar con SQL Server. Todas las aplicaciones que se comunican con SQL Server lo hacen enviando instrucciones Transact-SQL al servidor, independientemente de la interfaz de usuario de la aplicación.

Algunos comandos básicos de SQL son:

CREATE TABLE: Crea un objeto tabla relacional. El formato es darle un nombre y luego los tipos y tamaños de los datos que va a contener.

CREATE TRIGGER: Crea un disparador antes o después de una determinada acción

CREATE PROCEDURE: Crea una función que puede realizar operaciones sobre datos de una base de datos.

CREATE INDEX: Pone índice a algunos tipos de objetos de una base de datos para un acceso más expedito.

DROP: Permite borrar un objeto de la base de datos.

INSERT: Permite insertar datos a una tabla ya creada.

SELECT: Permite seleccionar filas de una tabla que cumplan con una condición dada

DELETE: Borra datos existentes en una fila.

UPDATE: Actualiza los valores de una tabla.

6.2.2 Oracle

Oracle es una empresa que crea un sistema de bases de datos por más de 20 años. Su principal producto es el servidor Oracle cuya última versión es la 9i. **Oracle9i** apareció hace un par de meses. El servidor Oracle es un administrador de base de datos objeto-relacional que provee una aproximación integral al manejo de información. Un servidor Oracle consiste en una base de dato y una instancia [11].

Cada vez que una base de datos se inicia, el servidor crea un área global de sistema donde los procesos de Oracle son almacenados. Las áreas globales de sistema son áreas de memoria usadas por la información compartida por la base de datos y por los usuarios. Esta combinación de procesos y buffers de memoria es una instancia. Una instancia consta de dos tipos de procesos, los del usuario que ejecutan el código de una aplicación o herramienta y los de Oracle que realizan los procesos del usuario y la mantención del servidor.

La estructura lógica de una base de datos Oracle consta de:

- **Espacios de tabla:** Es la unidad de almacenamiento lógica de una base de datos. Agrupa las estructuras relacionadas juntas.
- **Esquemas y objetos de esquema:** Son colecciones de objetos de las bases de datos. Un objeto de esquema es la estructura lógica que se refiere directamente al dato de la base de datos. Entre estos objetos encontramos:
 - Tablas: donde se almacenan los esquemas de los datos mediante lenguaje SQL.
 - Vistas: consultas estáticas o dinámicas de los datos en las tablas.
 - Secuencias: variables que almacenan valores en orden decreciente.
 - Procedimientos almacenados: funciones sobre los datos y otros como sinónimos, índices, clusters y enlaces a otras bases de datos.
- **Control:** Con el fin de controlar más detalladamente el funcionamiento de la base de datos, existen segmentos que constituyen otro nivel de estructura lógica.

- **Estructuras físicas de la base de datos:** Compuesta por archivos de datos, archivos de recuperación, archivos de control que permiten el correcto funcionamiento del sistema. Mediante SQL podemos definir el modelo de dato (DDL) y manipularlo (DML). También existen utilitarios para importar y exportar datos desde archivos o ASCII con SQL*Loader.

6.2.3 Elección de SABD

Para efectos del presente proyecto se utilizara el sistema administrador de bases de datos SQL Server 7 para manejar la base de datos necesaria para el correcto funcionamiento del sistema a desarrollar debido a las siguientes razones:

El costo de Oracle es mucho mayor al de SQL Server.

Mayor facilidad de uso e instalación en sistemas SQL Server que en Oracle.

Para sistemas de manejo de datos de bajo/mediano nivel, la relación potencia / calidad / precio es mucho mejor SQL Server que Oracle.

La Dirección de tránsito y transporte público posee la licencia necesaria para SQL Server 7.

El lenguaje Transct-SQL (usado por SQL Server) es más intuitivo e indica ciertos errores, lo cual el lenguaje PL/SQL (usado en Oracle) no hace.

Para entornos Windows definitivamente es mejor opción SQL Server que Oracle (este entorno se trabaja en la dirección del tránsito)

Y además la facilidad de uso que ofrece el sistema con el cual el grupo de trabajo ya ha trabajado con anterioridad.

6.3 Lenguaje de programación

6.3.1 Visual Basic.NET

Es un lenguaje de programación visual, también llamado lenguaje de 4° generación. Esto quiere decir que un gran número de tareas se realizan sin escribir código, simplemente con operaciones gráficas realizadas con el ratón sobre la pantalla [12].

Esta nueva versión de Visual Basic incorpora todas las características de un lenguaje orientado a objetos, como herencia, incorpora también la posibilidad de utilizar polimorfismo y la posibilidad de crear hilos y tramas de ejecución (lo que no se podía hacer con las versiones anteriores)

Visual Basic es un lenguaje que esta orientado a la realización de programas en la plataforma Windows, pudiendo incorporar todos los elementos de este entorno informático: ventanas, botones, cajas de diálogo y de texto, botones de opción y de selección, barras de desplazamiento, gráficos, menús, etc.

La aplicación **Visual Basic .NET** de Microsoft puede trabajar de dos modos distintos: en modo de diseño y en modo de ejecución.

- En **modo de diseño** el usuario construye interactivamente la aplicación, colocando controles en el formulario, definiendo sus propiedades, desarrollando funciones para gestionar los eventos.
- La aplicación se prueba en **modo de ejecución**. En ese caso el usuario actúa sobre el programa (introduce eventos) y prueba cómo responde el programa.

Hay algunas propiedades de los controles que deben establecerse en modo de diseño, pero muchas otras pueden cambiarse en tiempo de ejecución desde el programa escrito en Visual Basic .NET, también hay propiedades que sólo pueden establecerse en modo de ejecución y que no son visibles en modo de diseño.

Este es un lenguaje de programación de alto nivel que facilita la tarea de los programadores al otorgar un entorno grafico en el cual desarrollar las aplicaciones, además posee una alta compatibilidad y la capacidad de conectarse a gestores de bases de datos (en el caso del presente proyecto SQL Server 7)

Visual Basic .NET esta basado en .NET Framework y utiliza la jerarquía de clases que esta incluida en él. NET framework es un entorno para construir, instalar y ejecutar servicios Web y otras aplicaciones, esta compuesto de tres partes principales: El common lenguaje runtime, las clases framework y asp.NET.

Además, .NET framework es un entorno multi lenguaje, soporta no solo visual Basic .NET, sino que además permite trabajar con asp .NET, Visual C y C++. .NET framework es una nueva plataforma diseñada para simplificar el desarrollo de aplicaciones en el entorno distribuido de Internet

En Visual Basic .NET a diferencia de lo que ocurría en las versiones anteriores de Visual Basic, sólo existe un tipo de fichero de código, el cual tiene la extensión **.vb**, en este tipo de fichero pueden coexistir distintos tipos de *elementos*, por ejemplo: un módulo de clase, un formulario, un módulo de código, un control, etc.; mientras que en las versiones anteriores de Visual Basic, cada uno de estos elementos tenían su propio tipo de fichero con su respectiva extensión.

6.3.2 Java

Java es un lenguaje de programación que refleja mucha de la experiencia adquirida a lo largo de las últimas décadas sobre el desarrollo de programas y el diseño de lenguajes de programación. Esta experiencia se refleja en el diseño de java como un lenguaje *orientado a objetos* [1].

Las características más importantes que posee este lenguaje son:

Es un lenguaje diseñado para ser multiplataforma, es decir, que no importando la plataforma con que se trabaje (Windows, Linux, Unix, entre otros) el modo de programar sigue siendo el mismo; Esto gracias a que JAVA posee una “maquina virtual”, la cual se encarga de traducir y ejecutar el código escrito a un lenguaje reconocible por la plataforma que se este utilizando.

Java es un lenguaje orientado a objetos relativamente simple (en comparación con otros lenguajes orientados a objeto como C++), “*incluso un principiante puede hacer uso de las características que ofrece java para escribir programas elegantes*”.

Permite la utilización de herencia, lo cual permite extender el comportamiento de las clases.

Es un lenguaje recursivo: dentro de JAVA se pueden crear procedimientos recursivos (procedimientos que se llaman a si mismos).

Al igual que otros lenguajes java posee un conjunto de reglas que nos indican que frases o líneas de código son validas.

Seguridad: El código es "robusto", lo que aquí significa que, a diferencia de los programas escritos en C++ y quizá algunos otros lenguajes, los objetos Java no pueden tener referencias a datos externos a ellos mismos. Esto asegura que una instrucción no pueda tener la dirección de almacenamiento de datos en otra aplicación o en el sistema operativo mismo, casos que provocarían que el programa y quizá el sistema operativo se cerraran o colapsaran. La máquina virtual de Java realiza una serie de verificaciones de cada objeto para garantizar su integridad.

Java fue lanzado por Sun Microsystems en 1995 e instantáneamente creó una nueva percepción de las posibilidades interactivas de la Red. Los dos principales navegadores incluyen una máquina virtual de Java. Casi todos los principales desarrolladores de sistemas operativos (IBM, Microsoft y otros) han añadido compiladores de Java como parte de su oferta de productos.

La máquina virtual de Java incluye un compilador opcional "justo a tiempo" (just-in-time, JIT) que compila dinámicamente el código bytes en código ejecutable como una alternativa a la interpretación de una instrucción de código de bytes a la vez. En muchos casos la compilación dinámica JIT es más rápida que la interpretación de la máquina virtual.

6.3.3 Elección del lenguaje de programación

Se escogió Visual Basic .NET como lenguaje de programación a utilizar para la construcción del sistema debido a la simplicidad que ofrece Visual Basic para la creación de interfaces en entornos de trabajo Windows, contando con todas las características necesarias para la creación de formularios estilo Windows de forma rápida y transparente para el usuario en el modo gráfico y agregar las funcionalidades necesarias a través de código dentro del mismo editor. Ofrece también propiedades y funciones prediseñadas para todos los controles de los formularios (botones, menús, barras de herramientas, grupos, cuadros de texto, etc.) y permite un acceso rápido a estas tanto en el modo gráfico, como a través de código.

Visual Basic .NET incorpora además una gran cantidad de bibliotecas de trabajo con las que se puede agregar funcionalidades adicionales al lenguaje (como por ejemplo: Tipos de datos adicionales, controles especiales para formularios, etc.) y, gracias a estas bibliotecas, ofrece la posibilidad de conectarse a distintos servidores de bases de datos en forma rápida (Tanto en modo gráfico (a diferencia de java), como a través de código), además de poseer una gran cantidad de

documentación, manuales, ayuda, tanto en línea como incorporados en el ambiente de desarrollo es bastante completa y variada, ofreciendo una gran cantidad de autores y temas relacionados.

Otra de las características por las cuales se escogió este lenguaje es que ofrece la posibilidad de construir ensamblados, lo que facilita en gran medida la instalación de los programas, dando la posibilidad de no depender de un instalador, debido a que los ensamblados no necesitan agregar líneas al registro de Windows para su funcionamiento, ya que el mismo entorno de .NET ejecuta las aplicaciones.

Un estudio realizado por Microsoft y otras empresas comparando el rendimiento de ambos lenguajes en máquinas similares, tomando un programa que fue desarrollado en paralelo en ambos lenguajes (.NET y Java) demostró que .NET supera el rendimiento obtenido por Java en el sistema y además de tener una mejor escalabilidad y ofrece aplicaciones que son más fáciles de implantar.

Otra de las razones por las que se escogió este lenguaje es que en el Departamento de Tránsito y Transporte Público se utiliza la plataforma Windows y el lenguaje facilita su construcción.

7. REQUERIMIENTOS DEL SISTEMA

7.1 Requerimientos Funcionales

7.1.1 Requerimientos del usuario

Después de varias conversaciones y haber realizado distintos análisis con el usuario, se ha logrado capturar los siguientes requerimientos funcionales:

Permitir el ingreso, modificación y eliminación de las patentes junto con los datos asociados a estas.

Mantener el orden de las patentes en forma alfabética, utilizando las letras de la patente como índice.

Permitir la búsqueda de una patente en particular o de un grupo de estas.

Permitir la generación de informes relacionadas con una consulta específica.

Permitir el bloqueo de una patente, lo cual quiere decir que la carpeta correspondiente a esa patente no puede ser extraída de la sección procesamiento de datos y archivo.

Colocar notas y advertencias a las patentes en caso de que el usuario lo requiera.

Mostrar la opción para poder ver el historial de una patente elegida, es decir, todos los sucesos que han acontecido con anterioridad a esa carpeta.

El administrador debe poder crear, modificar y eliminar usuarios para restringir el acceso al sistema, manejando los permisos en cada caso.

7.1.2 Requerimientos del sistema

Desde el punto de vista del sistema, este debe cumplir con los siguientes requerimientos:

El sistema debe permitir agregar, modificar y eliminar filas cuando el usuario lo requiera (Referencias casos de uso: ingresar patente figuras 9.3, modificar patente figura 9.4, eliminar patente figura 9.5).

El sistema debe ordenar automáticamente las filas en forma alfabética al ser ingresadas por el usuario, utilizando las letras de la patente como índice.

El sistema debe ser capaz de encontrar una fila en particular o de un grupo de estas a través de una búsqueda solicitada por el usuario, la cual puede realizarse por campos específicos dentro de las filas (Referencias caso de uso: Búsqueda patente figura 9.10).

El sistema debe permitir al usuario generar informes según el usuario lo requiera, dándole al usuario la opción de escoger el informe a confeccionar y si se desea imprimir este o guardarlo en disco (Referencias caso de uso: Generación de informes figura 9.6).

El sistema debe permitir el bloqueo de una patente, lo cual quiere decir que la carpeta correspondiente a esa patente no puede ser extraída de la sección procesamiento de datos y archivo, junto con no poder modificar los datos de esa patente, avisando a los usuarios del sistema. También dar la opción de desbloquear una patente ya bloqueada con anterioridad (Referencias caso de uso: Bloquear patente figura 9.8, Desbloquear patente figura 9.9).

El sistema debe permitir colocar notas y/o advertencias a las patentes en caso de que el usuario lo requiera agregando los campos a la fila correspondiente (Referencias caso de uso: Notas y advertencias figura 9.7).

El sistema debe permitir, dada una patente específica, poder ver el historial de esa patente. Consiste en todos los sucesos anteriores que han ocurrido sobre la carpeta escogida (Referencias caso de uso: Historial figura 9.2)

El sistema debe permitir al administrador crear, modificar y eliminar usuarios para permitir a usuarios autorizados el acceso al sistema, y evitar que otros usuarios ingresen al mismo, junto con restringir los permisos de estos usuarios (Referencias caso de uso: Ingresar usuario figura 9.14, Modificar usuario figura 10.15, Eliminar usuario figura 9.16).

7.2 Requerimientos no funcionales

7.2.1 Requerimientos del usuario

Los requerimientos no funcionales del usuario son los siguientes:

Los tiempos de respuestas deben ser mínimos.

El sistema debe ser seguro.

El sistema debe ser de fácil uso y agradable para el usuario.

El sistema debe mantener la metodología básica de trabajo.

7.2.2 Requerimientos del sistema

De acuerdo a lo anterior, el sistema debe poseer los siguientes requerimientos no funcionales:

El tiempo de respuesta no debe ser mayor a 5 segundos.

El sistema debe ser resistente a fallos, lo que significa tener medidas de seguridad para evitar errores graves y si por alguna casualidad llega a ocurrir uno, tener una manera rápida y fácil de poder recuperarse.

Se debe desplegar una interfaz fácil de usar y acorde con el lugar de trabajo, con colores sobrios y tipos de letras que no molesten al usuario al momento de leer.

El sistema junto con mantener la metodología básica de trabajo (presentación de la información, formato de esta) adjuntara nuevas funcionalidades que facilitaran el tratamiento de la información y mejorar el trabajo en el departamento.

8. DIAGRAMA DE CASOS DE USO GENERALES

8.1 Caso de uso usuario registrado

La siguiente figura corresponde al caso de uso para los usuarios registrados del sistema, el cual se obtuvo realizando un análisis de los requerimientos del cliente (usuarios y personal administrativo).

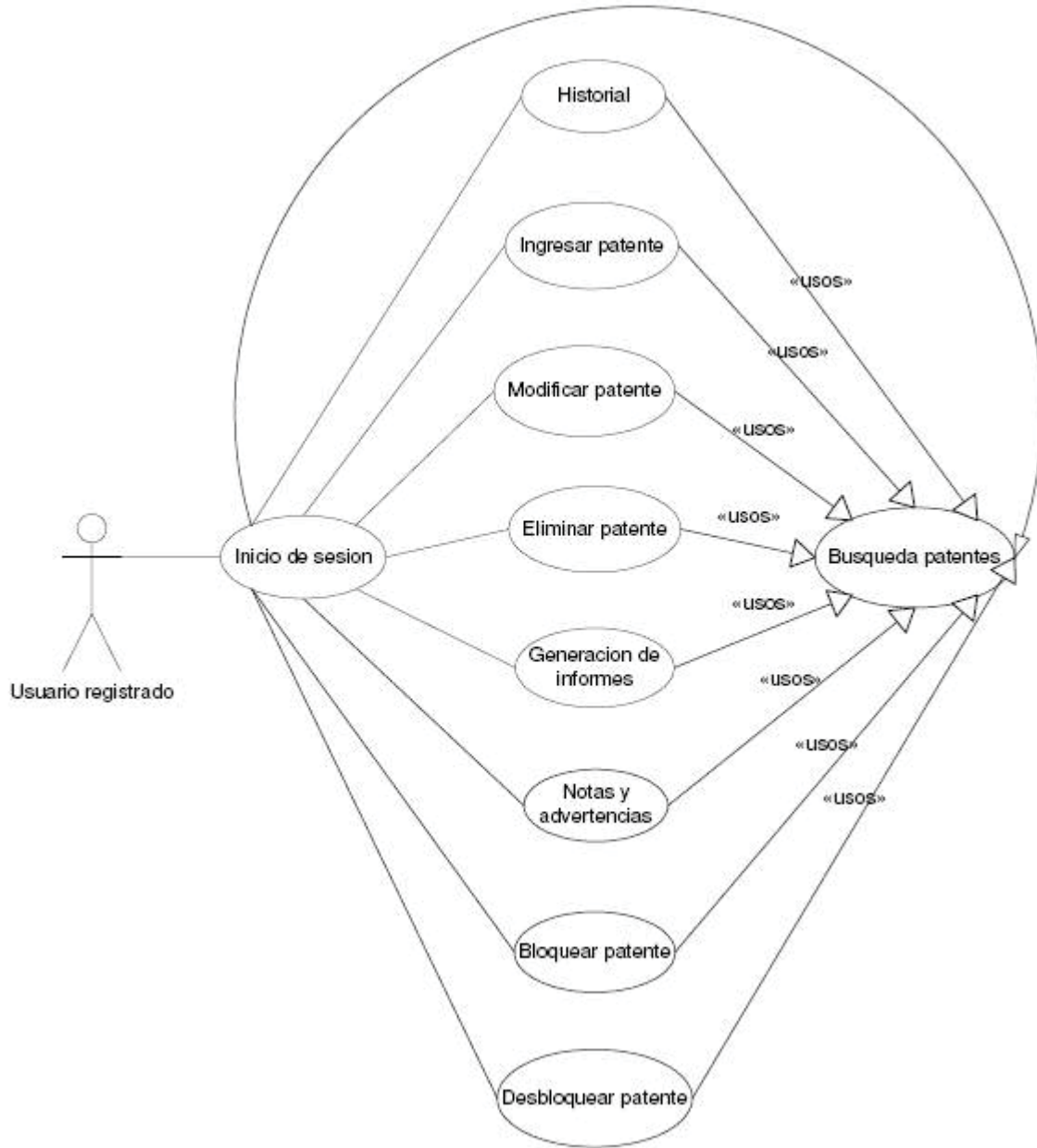


Figura 8.1: Casos de uso usuario registrado

8.2 Caso de uso usuario no registrado

La siguiente figura corresponde al caso de uso para los usuarios no registrados del sistema el cual se obtuvo realizando un análisis de los requerimientos del cliente (usuarios y personal administrativo) y se agregara al sistema a construir, aun cuando esto representa un riesgo a la seguridad de la información que se maneja.

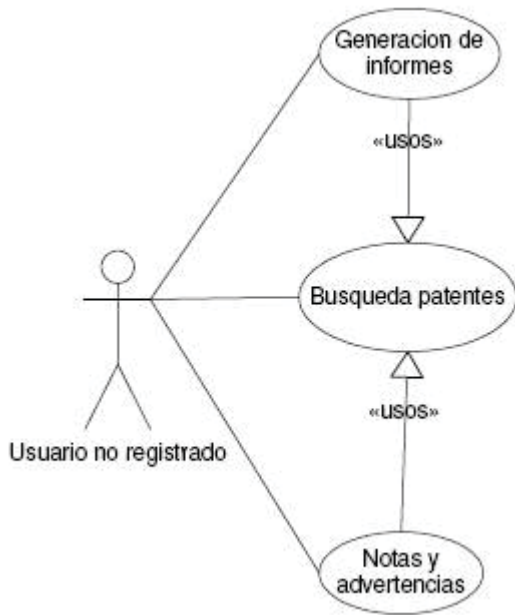


Figura 8.2: Casos de uso usuario no registrado

8.3 Caso de uso administrador

La siguiente figura corresponde al caso de uso referente al administrador del sistema, el cual se obtuvo realizando un análisis de los requerimientos del cliente (usuarios y personal administrativo), dándole las funcionalidades necesarias para el manejo de la seguridad y usuarios dentro del sistema.

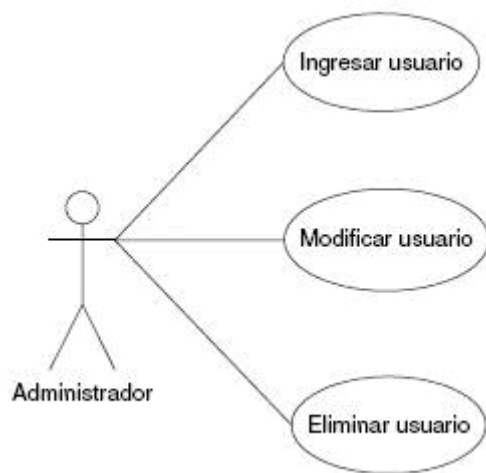


Figura 8.3: Casos de uso administrador

Como el administrador debe tener un total control sobre el sistema, tiene acceso a todas las funcionalidades de un usuario registrado.

9. CASOS DE USO EXPLICATIVOS

Las siguientes tablas y figuras corresponden a los casos de uso explicativos en su forma reducida, los cuales explican de forma básica cada uno de los casos correspondientes a las iteraciones.

9.1 Usuario registrado

Tabla 9.1: Caso de uso explicativo Inicio de sesión

Casos de uso	Inicio de sesión
Precondiciones	El usuario debe estar registrado con anterioridad.
Actores	Usuario registrado
Descripción	Validar el login y el password para permitir el acceso a los usuarios que tengan permiso.



Figura 9.1: Caso de uso Inicio de sesión

Tabla 9.2: Caso de uso explicativo Historial

--

Casos de uso	Historial
Precondiciones	El usuario debe estar registrado y haber ejecutado con anterioridad el caso de uso “Inicio de sesión”.
Actores	Usuario registrado
Descripción	Revisar el historial de una patente.

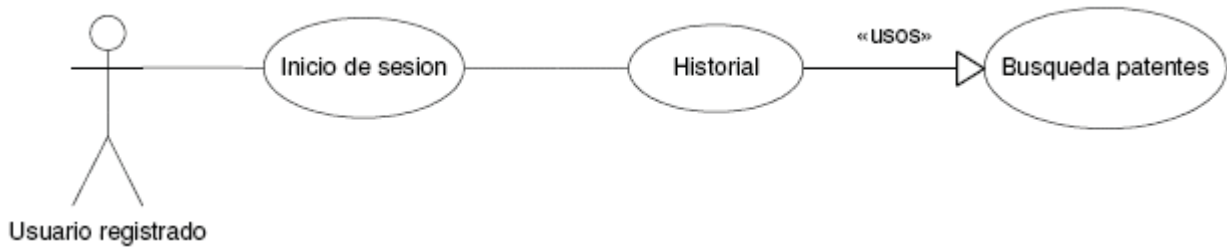


Figura 9.2: Caso de uso Historial

Tabla 9.3: Caso de uso explicativo Ingresar patente

Casos de uso	Ingresar patente
Precondiciones	El usuario debe estar registrado y haber ejecutado con anterioridad el caso de uso “Inicio de sesión”. Puede elegir entre las opciones “Caja” y “Libro”.
Actores	Usuario registrado
Descripción	El ingreso o actualización al sistema de una patenté con sus datos correspondientes.

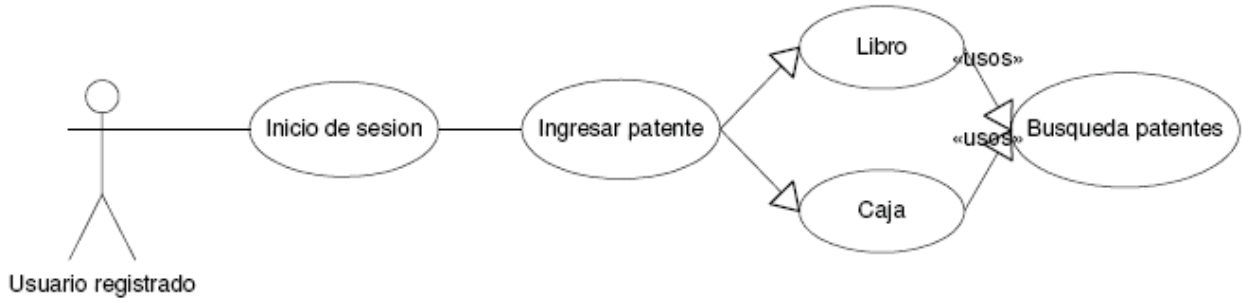


Figura 9.3: Caso de uso Ingresar patente

Tabla 9.4: Caso de uso explicativo Modificar patente

Casos de uso	Modificar patente
Precondiciones	El usuario debe estar registrado y haber ejecutado con anterioridad el caso de uso “Inicio de sesión”.
Actores	Usuario registrado
Descripción	La modificación de los datos asociados a una patente específica.

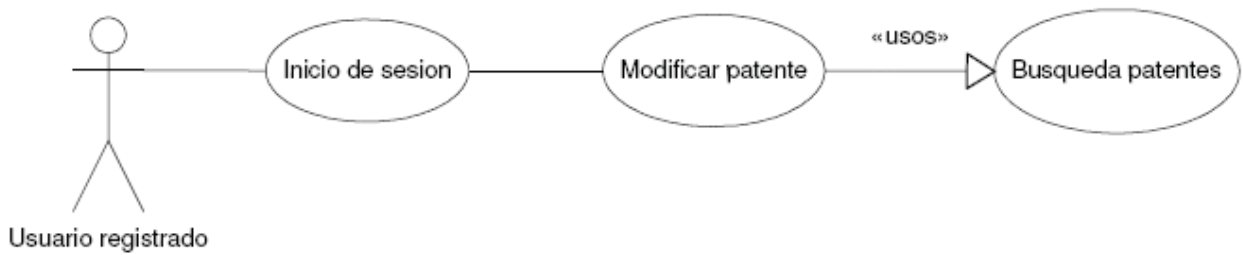


Figura 9.4: Caso de uso Modificar patente

Tabla 9.5: Caso de uso explicativo Eliminar patente

--	--

Casos de uso	Eliminar patente
Precondiciones	El usuario debe estar registrado y haber ejecutado con anterioridad el caso de uso “Inicio de sesión”.
Actores	Usuario registrado
Descripción	La eliminación de una patente.

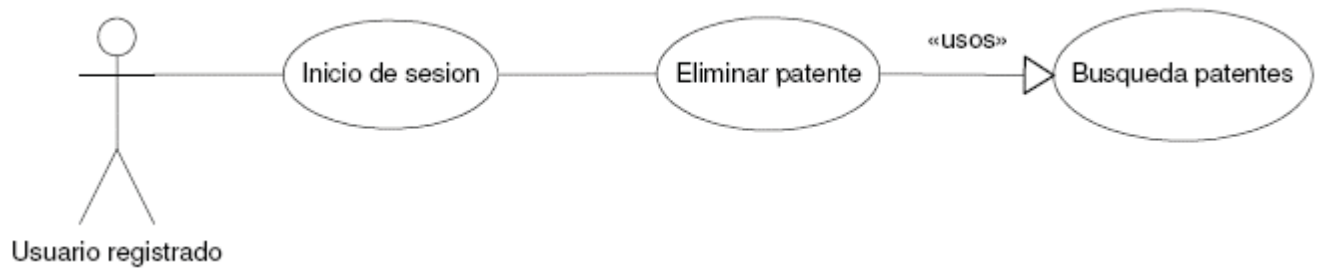


Figura 9.5: Caso de uso Eliminar patente

Tabla 9.6: Caso de uso explicativo Generación de informes

Casos de uso	Generación de informes
Precondiciones	El usuario debe estar registrado y haber ejecutado con anterioridad el caso de uso “Inicio de sesión”.
Actores	Usuario registrado
Descripción	La generación de ciertos informes según la necesidad del usuario

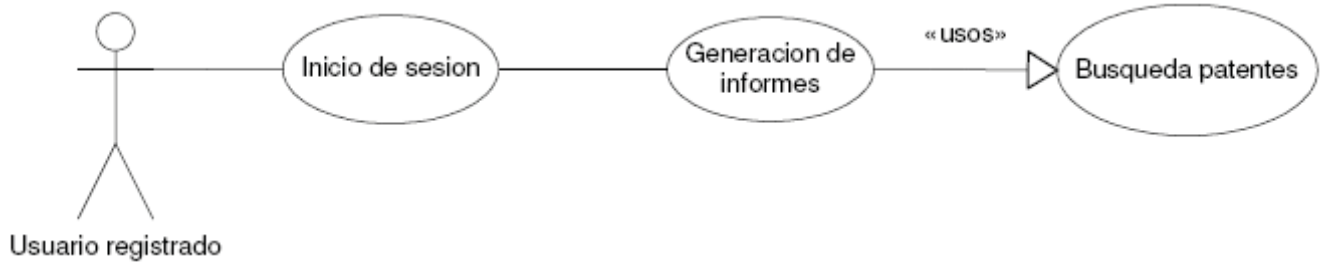


Figura 9.6: Caso de uso explicativo Generación de informes

Tabla 9.7: Caso de uso explicativo Notas y advertencias

Casos de uso	Notas y advertencias
Precondiciones	El usuario debe estar registrado y haber ejecutado con anterioridad el caso de uso “Inicio de sesión”.
Actores	Usuario registrado
Descripción	Poner una nota o advertencia sobre una patente seleccionada.

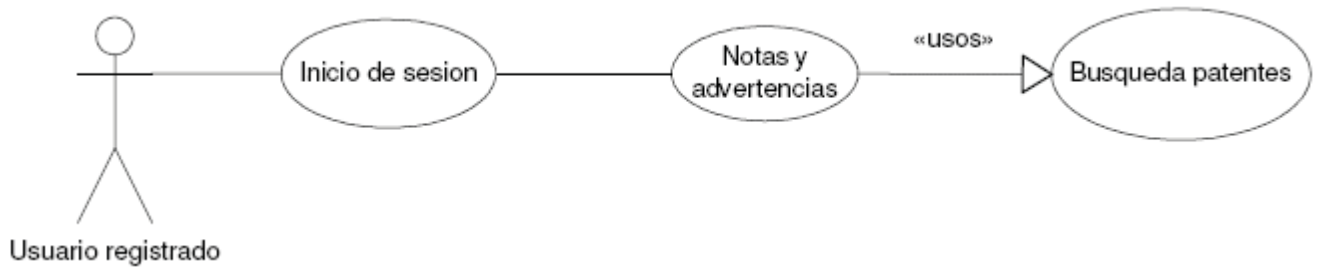


Figura 9.7: Caso de uso Notas y advertencias

Tabla 9.8: Caso de uso explicativo Bloquear patente

--

Casos de uso	Bloquear patente
Precondiciones	El usuario debe estar registrado y haber ejecutado con anterioridad el caso de uso “Inicio de sesión”.
Actores	Usuario registrado
Descripción	Permitir el bloqueo de una patente elegida, impidiendo que se puedan realizar cambios sobre esta.

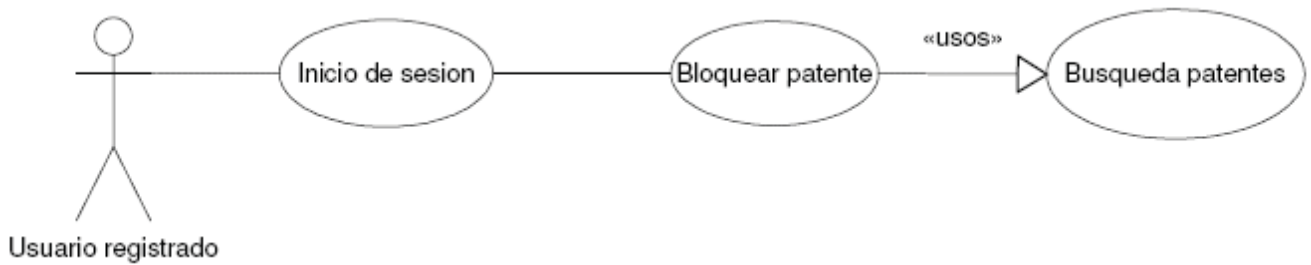


Figura 9.8: Caso de uso Bloquear patente

Tabla 9.9: Caso de uso explicativo Desbloquear patente

Casos de uso	Desbloquear patente
Precondiciones	El usuario debe estar registrado y haber ejecutado con anterioridad el caso de uso “Inicio de sesión”.
Actores	Usuario registrado
Descripción	Permitir el desbloqueo de una patente elegida, permitiendo realizar cambios sobre esta.

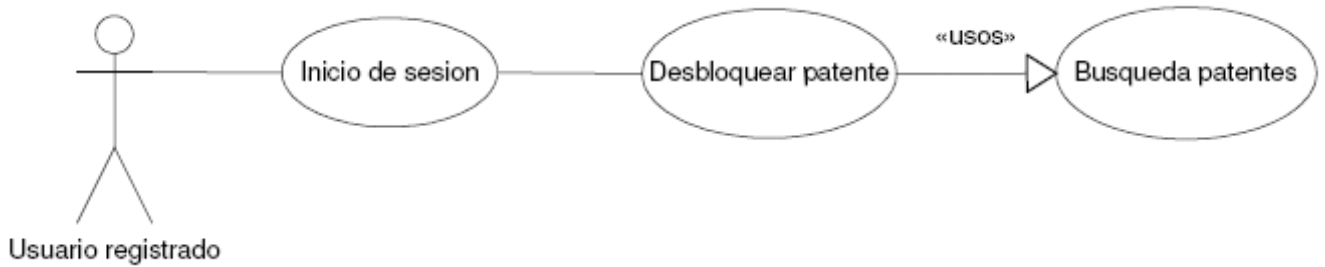


Figura 9.9: Caso de uso Desbloquear patente

Tabla 9.10: Caso de uso explicativo Búsqueda patentes

Casos de uso	Búsqueda patentes
Precondiciones	El usuario debe estar registrado y haber ejecutado con anterioridad el caso de uso “Inicio de sesión”.
Actores	Usuario registrado
Descripción	La búsqueda de una patente y sus respectivos datos o también de un grupo de estas.

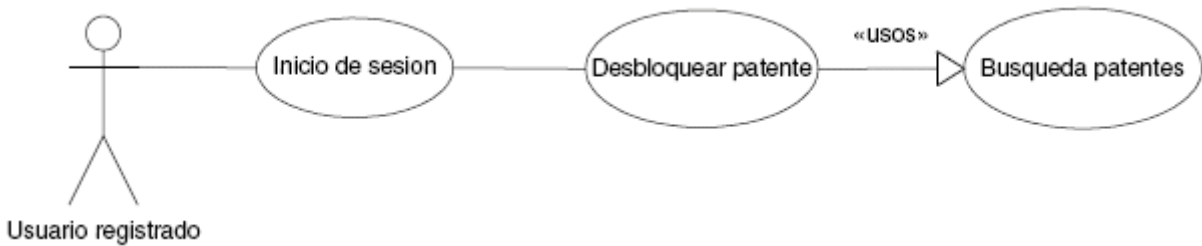


Figura 9.10: Caso de uso Búsqueda patentes

9.2 Usuario no registrado

Tabla 9.11: Caso de uso explicativo Generación de informes

Casos de uso	Generación de informes
Precondiciones	El usuario debe haber ejecutado con anterioridad el caso de uso “Inicio de sesión” y haber ingresado como invitado.
Actores	Usuario no registrado
Descripción	La generación de ciertos informes según la necesidad del usuario

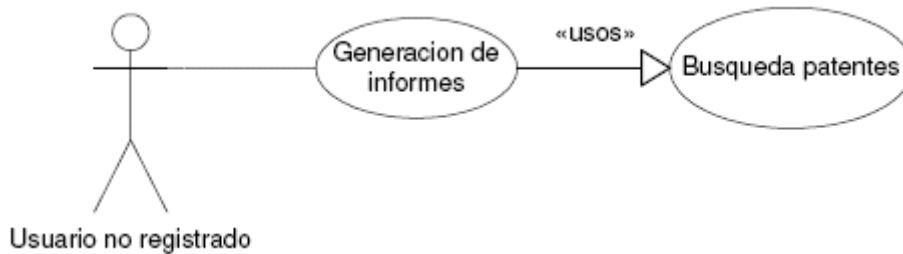


Figura 9.11: Caso de uso Generación de informes

Tabla 9.12: Casos de uso explicativo Búsqueda patentes

Casos de uso	Búsqueda patentes
Precondiciones	El usuario debe haber ejecutado con anterioridad el caso de uso “Inicio de sesión” y haber ingresado como invitado.
Actores	Usuario no registrado
Descripción	La búsqueda de una patente y sus respectivos datos o también de un grupo de estas.



Figura 9.12: Caso de uso Búsqueda patentes

Tabla 9.13: Caso de uso explicativo Notas y advertencias

Casos de uso	Notas y advertencias
Precondiciones	El usuario debe haber ejecutado con anterioridad el caso de uso “Inicio de sesión” y haber ingresado como invitado.
Actores	Usuario no registrado
Descripción	Leer una nota o advertencia de la patente seleccionada.



Figura 9.13: Caso de uso Notas y advertencias

9.3 Administrador

Tabla 9.14: Caso de uso explicativo Ingresar usuario

--

Casos de uso	Ingresar usuario
Precondiciones	El usuario debe haber ingresado al sistema como administrador
Actores	Administrador
Descripción	Crear usuarios y darles los permisos necesarios para que estos puedan acceder a las opciones del sistema

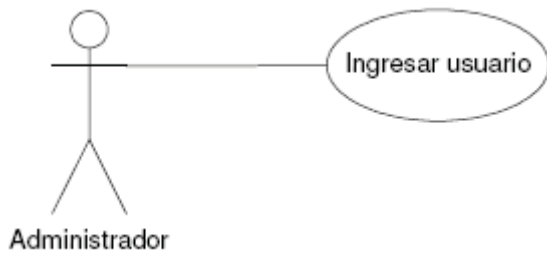


Figura 9.14: Caso de uso Ingresar usuario

Tabla 9.15: Caso de uso explicativo Modificar usuario

Casos de uso	Modificar usuario
Precondiciones	El usuario debe haber ingresado al sistema como administrador
Actores	Administrador
Descripción	Modificar los datos de un usuario cuando se requiera

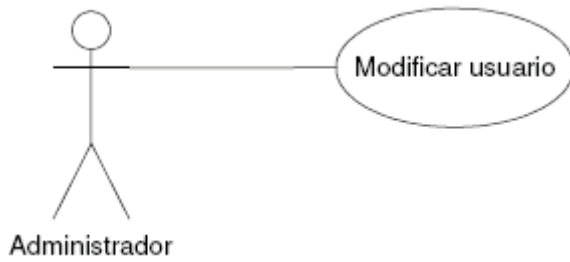


Figura 9.15: Caso de uso Modificar usuario

Tabla 9.16: Caso de uso explicativo Eliminar usuario

Casos de uso	Eliminar usuario
Precondiciones	El usuario debe haber ingresado al sistema como administrador
Actores	Administrador
Descripción	Eliminar un usuario del sistema para denegarle el ingreso a este.

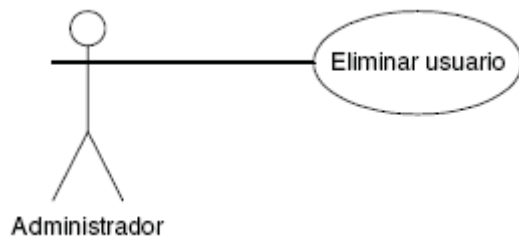


Figura 9.16: Caso de uso Eliminar usuario

10. ASIGNACION DE ITERACIONES

Las iteraciones dentro del Proceso Unificado de Modelado sirven para dividir el trabajo, lo que se podría decir en mini-proyectos, en los cuales se hace mucho más fácil la utilización de los flujos de trabajo.

En este proyecto se han asignado tres iteraciones, las cuales serán explicadas a continuación:

Primera iteración: Usuario registrado

En esta iteración se desarrollan los requerimientos relacionados con el usuario registrado. Este usuario es el principal en manejar el sistema, por lo tanto, será el que acapare la mayor cantidad de funcionalidades y ocupe el sistema la mayor parte del tiempo.

Segunda iteración: Usuario no registrado

En esta iteración se desarrollan los requerimientos relacionados con el usuario no registrado. Se podría decir que este usuario es un usuario registrado pero con restricciones, ya que podrá realizar algunas de las mismas funcionalidades del usuario registrado pero con ciertos cambios y otras simplemente no podrá realizarlas.

Tercera iteración: Administrador

En esta iteración se desarrollan los requerimientos relacionados con el administrador. El administrador es un usuario registrado con algunas ventajas, por lo tanto, tendrá las mismas funcionalidades del usuario registrado y tendrá otras exclusivas (las mismas funcionalidades entre el usuario registrado y el administrador, no serán trabajadas acá, ya que son exactamente las mismas del usuario registrado y serán trabajadas en la primera iteración).

11. PLAN DE PRUEBAS

Los casos de prueba que se realizarán en las iteraciones correspondientes a los módulos del sistema, seguirán la metodología de caja negra, con el fin de probar las funcionalidades de los módulos desarrollados por sobre el código fuente, tratando de probar todas las opciones y caminos posibles dentro de las interfaces correspondientes.

Las pruebas de caja negra, también denominadas pruebas de comportamiento, se centran en los requisitos funcionales del software. Estas pruebas permiten al ingeniero de software obtener conjuntos de condiciones de entrada que ejerciten completamente todos los requisitos funcionales de un programa. Las pruebas de caja negra no son una alternativa a las pruebas de caja blanca. Más bien se trata de un enfoque complementario que intenta descubrir diferentes tipos de errores que los métodos de caja blanca. Las pruebas de caja negra intentan encontrar errores de las siguientes categorías:

Funciones incorrectas o ausentes.

Errores de interfaz.

Errores en estructuras de datos o en acceso a bases de datos externas.

Errores de rendimiento.

Errores de inicialización y de terminación.

Mediante las pruebas de caja negra se obtiene un conjunto de casos de prueba que satisfacen los siguientes criterios:

Casos de prueba que reducen, en un coeficiente que es mayor que uno, el número de casos de prueba adicionales que se deben diseñar para alcanzar una prueba razonable.

Casos de prueba que nos dicen algo sobre la presencia o ausencia de clases de errores en lugar de errores asociados solamente con la prueba que se está realizando.

Para escoger los casos de prueba se utilizaran los siguientes criterios mínimos que deberán cumplir:

Valores lo mas realistas posibles.

Valores límites dentro de las capacidades del sistema.

Valores inválidos, los que generaran mensajes de advertencia al usuario.

Estos casos de prueba tendrán como objetivo el encontrar posibles errores dentro de las funciones, interfaces, base de datos, rendimiento e intentar corregirlos para asegurar la efectividad del programa.

12. PRIMERA ITERACION: USUARIO REGISTRADO

12.1 Diagrama de clases

El diagrama de clases es el diagrama principal de diseño para un sistema. En él se especifican las clases del sistema, con relaciones entre clases y estructuras de herencia [7]. Como primera instancia del desarrollo de la primera iteración, se construyó y refinó este diagrama para tener una visión real del sistema.

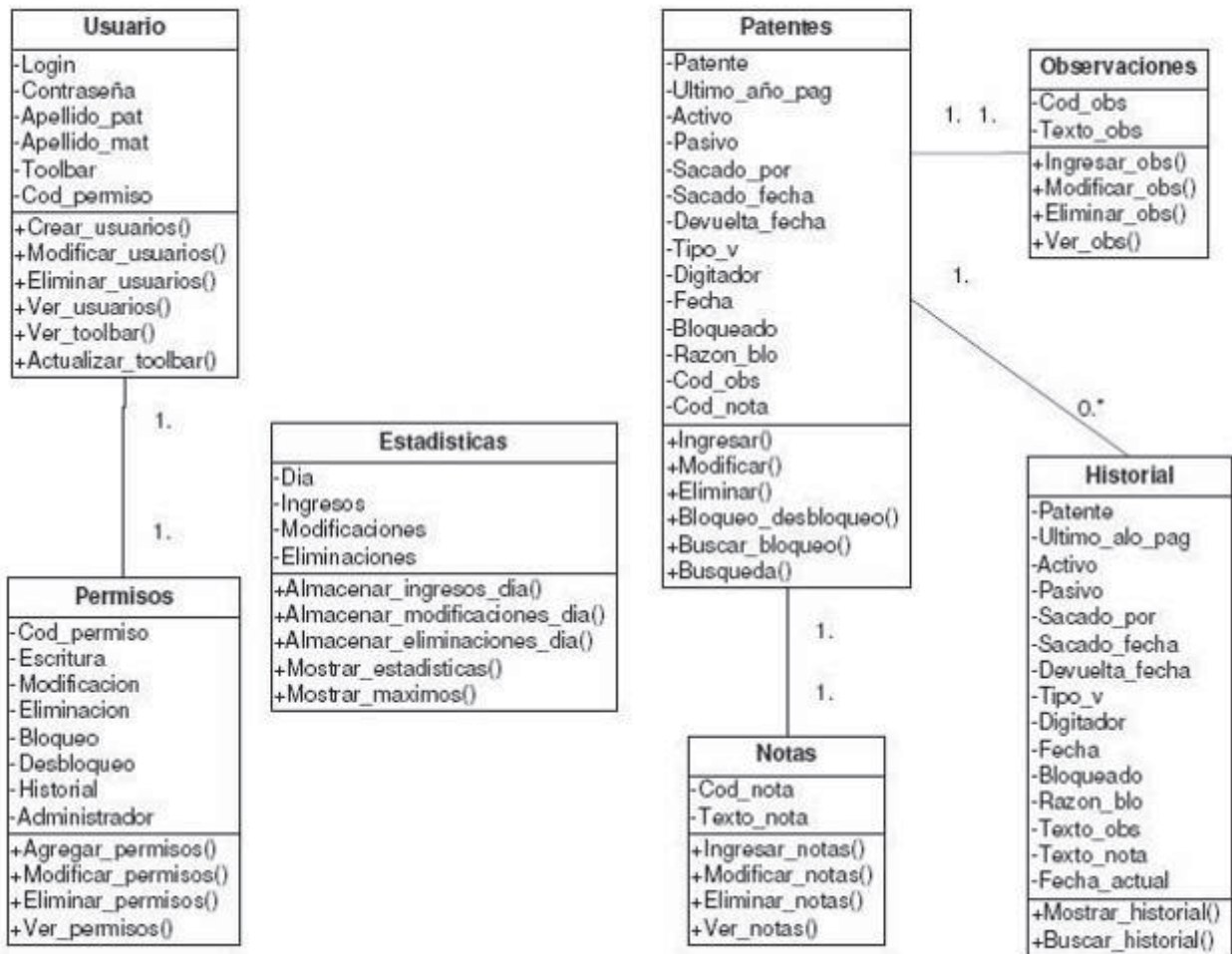


Figura 12.1: Diagrama de clases

12.2 Diagramas de caso de uso extendido

A continuación se presenta los casos de uso extendidos correspondientes a la iteración “Usuario registrado”, los cuales muestran todos los posibles caminos que podría tomar el caso de uso al ejecutarse.

Tabla 12.1: Caso de uso extendido Inicio de sesión

Caso de uso	Inicio de sesión.
Actor Primario	Usuario registrado.
Participantes e intereses	Usuario: Poder iniciar sesión de forma fácil y óptima.
Propósito	Validar el login y el password para permitir el acceso a los usuarios que tengan permiso.
Precondiciones	El usuario debe estar registrado con anterioridad.
Poscondiciones	No tiene.
Escenario Principal	<ol style="list-style-type: none"> 1. El usuario desea iniciar sesión. 2. El sistema despliega la ventana de inicio de sesión. 3. El usuario ingresa su login y password. 4. El usuario selecciona entre una de las siguientes opciones: “Aceptar” y “Cancelar”. 5. Se continúa con el ingreso al sistema y finaliza el caso de uso.

Extensiones	<p>4.a. Si la actividad seleccionada es “Aceptar”, el sistema verifica los datos ejecutando una función con los parámetros login y password.</p> <ol style="list-style-type: none"> 1. Si el login y el password son validos se inicia sesión y se continua al siguiente paso. 2. En caso de no ser valido el login y el password, se pide el ingreso nuevamente de estos. <p>4.b. Si la actividad seleccionada por el usuario es “Cancelar”, se saldrá del sistema inmediatamente.</p>
Requerimientos especiales	Debe ser la primera ventana que se le muestre al usuario. El resto de las ventanas no pueden ser accedidas sino hasta haber completado este caso de uso.
Frecuencia de ocurrencia	<p>Continua.</p> <p>La única forma de ingresar a las opciones del sistema es la identificación y validación de un usuario permitido.</p>

Tabla 12.2: Caso de uso extendido Historial

Caso de uso	Historial.
Actor Primario	Usuario registrado.
Participantes e intereses	Usuario: Revisar el historial de una patente seleccionada.
Propósito	Revisar el historial de una patente.
Precondiciones	El usuario debe estar registrado y haber ejecutado con

	anterioridad el caso de uso “Inicio de sesión”.
Poscondiciones	No tiene.
Escenario Principal	<ol style="list-style-type: none"> 1. El usuario ingresa a la sección “Historial”. 2. El sistema solicita al usuario la patente a buscar. 3. El usuario ingresa la patente, seleccionando una de las siguientes opciones: “Revisar”, “Cancelar”. 4. El usuario finaliza el caso de uso.
Extensiones	<p>3.a. Si la actividad seleccionada es “Revisar”, el sistema verificara si la patente seleccionada esta ingresada en el sistema.</p> <ol style="list-style-type: none"> 1. Si la patente es valida, se desplegara por la pantalla el historial correspondiente. 2. En caso de no ser valida, se desplegara un mensaje indicándole al usuario que esa patente no existe pidiéndole un nuevo ingreso. <p>3.b. Si la actividad seleccionada es “Cancelar”, el sistema cerrara la ventana Historial volviendo a la pantalla principal.</p>
Requerimientos especiales	El historial no puede ser modificado por ningún usuario.
Frecuencia de ocurrencia	Ocasional.

Tabla 12.3: Caso de uso extendido Ingresar patente

--

Caso de uso	Ingresar patente.
Actor Primario	Usuario registrado.
Participantes e intereses	Usuario: El usuario desea ingresar o actualizar los datos sobre una patente.
Propósito	El ingreso al sistema de una patenté o la actualización de esta con sus datos correspondientes.
Precondiciones	El usuario debe estar registrado y haber ejecutado con anterioridad el caso de uso “Inicio de sesión”.
Poscondiciones	Los datos ingresados deben quedar guardados en la base de datos.
Escenario Principal	<ol style="list-style-type: none"> 1. El usuario selecciona “Ingresar patente” en el sistema y elige entre las opciones: “Caja” o “Libro”. 2. El sistema solicita al usuario los datos a ingresar. 3. El usuario llena los campos con los datos. 4. El usuario elige entre las opciones: “Ingresar”, “Nuevo” o “Cancelar” 5. El usuario finaliza el caso de uso.
Extensiones	<p>1.a. Si el usuario elige la opción “Libro”, esta opción permite solamente actualizar una patente ingresada al sistema.</p> <p>1.b. Si el usuario elige la opción “Caja”, esta opción permite el ingreso de una nueva patente como también la actualización de una nueva.</p> <p>3.a. Algunos datos no serán necesarios y podrán dejarse en</p>

	<p>blanco. Esos campos serán correctamente indicados al usuario.</p> <p>4.a. Si la actividad seleccionada es “Ingresar”, los datos serán enviados a la base de datos para su almacenamiento, volviendo al paso numero 2.</p> <p>4.b. Si la actividad seleccionada es “Nuevo” el sistema borrara los datos ingresados hasta el momento, dejando en blanco las celdas pudiendo realizar un nuevo ingreso.</p> <p>4.c Si la actividad seleccionada es “Cancelar”, el sistema cerrara la ventana “Ingresar datos” volviendo a la pantalla principal.</p>
Requerimientos especiales	No tiene.
Frecuencia de ocurrencia	Continua.

Tabla 12.4: Caso de uso extendido Modificar patente

Caso de uso	Modificar patente.
Actor Primario	Usuario registrado.
Participantes e intereses	Usuario: El usuario puede modificar cualquier dato asociado a una patente.
Propósito	La modificación de los datos asociados a una patente especifica.
Precondiciones	El usuario debe estar registrado y haber ejecutado con anterioridad el caso de uso “Inicio de sesión”.
Poscondiciones	Los cambios deben guardarse en la base de datos.

Escenario Principal	<ol style="list-style-type: none"> 1. El usuario selecciona “Modificar patente” en el sistema. 2. El sistema solicita al usuario la patente a modificar. 3. El usuario ingresa la patente. 4. Se le presentan todos los datos de la patente al usuario para que este realice los cambios necesarios. 5. El usuario elige entre las opciones: “Modificar”, “Deshacer” o “Cancelar”. 6. El usuario finaliza el caso de uso.
Extensiones	<ol style="list-style-type: none"> 3.a. Si la patente ingresada no es encontrada, se le pedirá un nuevo ingreso al usuario. 4.a. Los cambios serán realizados borrando el texto sobre el campo sobre el cual se quiere y escribiendo un nuevo texto sobre este. 5.a. Si la actividad seleccionada es “Modificar”, la información de la patente será actualizada en la base de datos, volviendo a la pantalla principal. 5.b. Si la actividad seleccionada es “Deshacer” el sistema borrara los cambios realizados hasta el momento rellenando los campos que han sido cambiados con los originales. 5.c Si la actividad seleccionada es “Cancelar”, el sistema cerrara la ventana “Modificar datos” volviendo a la pantalla principal.
Requerimientos especiales	No tiene.
Frecuencia de ocurrencia	Alta.

Tabla 12.5: Caso de uso extendido Eliminar patente

Caso de uso	Eliminar patente.

Actor Primario	Usuario registrado.
Participantes e intereses	Usuario: El usuario puede eliminar una patente a elección.
Propósito	La eliminación de una patente.
Precondiciones	El usuario debe estar registrado y haber ejecutado con anterioridad el caso de uso “Inicio de sesión”.
Poscondiciones	La eliminación debe verse reflejada en la base de datos.
Escenario Principal	<ol style="list-style-type: none"> 1. El usuario selecciona “Eliminar patente” en el sistema. 2. El sistema solicita la patente a eliminar. 3. El usuario ingresa la patente. 4. El usuario elige entre las opciones “Eliminar” o “Cancelar”. 5. El usuario finaliza el caso de uso.
Extensiones	<p>3.a. Si la patente ingresada no es encontrada, se le pedirá un nuevo ingreso al usuario.</p> <p>4.a. Si la actividad seleccionada es “Eliminar”, el sistema muestra los datos de la patente y pide confirmación.</p> <ol style="list-style-type: none"> 1. De ser positiva la confirmación, la patente es eliminada y se va al paso numero 5. 2. De ser negativa la confirmación, se va directamente al paso numero 5. <p>4.b. Si la actividad seleccionada es “Cancelar”, el sistema cerrara la ventana “Eliminar datos” volviendo a la pantalla principal.</p>
Requerimientos especiales	No tiene.

Frecuencia de ocurrencia	Baja.
--------------------------	-------

Tabla 12.6: Caso de uso extendido Generación de informes

Caso de uso	Generación de informes.
Actor Primario	Usuario registrado.
Participantes e intereses	Usuario: Poder obtener un informe según una consulta específica.
Propósito	La generación de ciertos informes según la necesidad del usuario.
Precondiciones	El usuario debe estar registrado y haber ejecutado con anterioridad el caso de uso “Inicio de sesión”.
Poscondiciones	No tiene.
Escenario Principal	<ol style="list-style-type: none"> 1. El usuario selecciona “Generación de informes” en el sistema. 2. El sistema le da a elegir entre las opciones: “Listado de patentes” e “Informes estadísticos” 3. El usuario selecciona la opción y elige el informe e ingresa los datos requeridos. 4. Se despliega en pantalla el informe y el usuario selecciona entre las opciones “Imprimir”, “Exportar” o “Cancelar” 5. El usuario finaliza el caso de uso.

Extensiones	<p>2.a. La opción “Listado de patentes” le da la opción de elegir entre los informes: “Por fecha” (informe de las patentes desde la fecha elegida hasta la actual), “Listado de todas las patentes”, “Listado de las patentes que comienzan con una letra” y “Listado de patentes entre rangos de letras”.</p> <p>2.b. La opción “Informes estadísticos” le da la opción de elegir entre los informes: “Informe en el periodo” (se eligen 2 fechas y se da un informe estadístico del tiempo comprendido entre esas 2 fechas) y “comparación entre meses” (se elige 2 meses y se da un informe comparativo entre los meses elegidos)</p> <p>4.a. Si la actividad seleccionada es “Imprimir”, el sistema imprimirá el informe en la impresora del computador.</p> <p>4.b. Si la actividad seleccionada es “Exportar”, el sistema le dará la opción de guardar el informe en cualquiera de las carpetas del computador, en distintos formatos.</p> <p>4.c. Si la actividad seleccionada es “Cancelar”, el sistema cerrará la ventana “Generación de informes” volviendo a la pantalla principal.</p>
Requerimientos especiales	No tiene.
Frecuencia de ocurrencia	Baja.

Tabla 12.7: Caso de uso extendido Notas y advertencias

--

Caso de uso	Notas y advertencias.
Actor Primario	Usuario registrado.
Participantes e intereses	Usuario: El usuario puede poner una nota o advertencia sobre la patente elegida.
Propósito	Poner una nota o advertencia sobre una patente seleccionada.
Precondiciones	El usuario debe estar registrado y haber ejecutado con anterioridad el caso de uso “Inicio de sesión”.
Poscondiciones	Los cambios realizados deben guardarse en la base de datos.
Escenario Principal	<ol style="list-style-type: none"> 1. El usuario selecciona “Notas y advertencias” en el sistema. 2. El sistema solicita el ingreso de la patente. 3. El usuario ingresa la patente sobre la cual desea agregar, modificar o borrar una nota o advertencia. 4. El usuario llena el campo que se le presenta con la nota o advertencia (con un límite de 40 palabras). 5. El usuario selecciona entre las actividades “Aceptar”, “Borrar” o “Cancelar”. 6. El usuario finaliza el caso de uso.

Extensiones	<p>3.a. Si la patente ingresada no es encontrada, se le pedirá un nuevo ingreso al usuario.</p> <p>4.a. Si la patente posee una nota anterior el usuario puede agregar texto sobre esta nota, borrar la nota y escribir otra o eliminar la nota.</p> <p>5.a. Si la actividad seleccionada es “Aceptar”, los datos serán enviados a la base de datos, retornando a la pantalla principal del sistema.</p> <p>5.b. Si la actividad seleccionada es “Borrar” el sistema borrara la nota de la patente.</p> <p>5.c Si la actividad seleccionada es “Cancelar”, el sistema cerrara la ventana “Notas y advertencias” volviendo a la pantalla principal.</p>
Requerimientos especiales	No tiene.
Frecuencia de ocurrencia	Ocasional.

Tabla 12.8: Caso de uso extendido Bloquear patente

Caso de uso	Bloquear patente
Actor Primario	Usuario registrado.
Participantes e intereses	Usuario: El usuario puede bloquear una patente seleccionada.

Propósito	Permitir el bloqueo de una patente elegida, bloqueando la patente para impedir que se puedan realizar cambios sobre esta.
Precondiciones	El usuario debe estar registrado y haber ejecutado con anterioridad el caso de uso “Inicio de sesión”.
Poscondiciones	El bloqueo debe registrarse en el sistema.
Escenario Principal	<ol style="list-style-type: none"> 1. El usuario selecciona “Bloquear” en el sistema. 2. El usuario elige una patente para bloquear. 3. El usuario ingresa el texto de bloqueo y elige entre las opciones “Bloquear” o “Cancelar” 4. El usuario finaliza el caso de uso.
Extensiones	<ol style="list-style-type: none"> 2.a. Si la patente ingresada no es encontrada, se le pedirá un nuevo ingreso al usuario. 2.b. En caso de intentar bloquear una patente ya bloqueada con anterioridad, se le indicara esto al usuario y se le pedirá un nuevo ingreso 3.a. Si la actividad seleccionada es “Bloquear” y se encuentra llenado el campo de texto, se procede a bloquear la patente. De no encontrarse el campo de texto se le pedirá el ingreso de este. 3.c. Si la actividad seleccionada es “Cancelar”, el sistema cerrara la ventana “Bloquear” volviendo a la pantalla principal.
Requerimientos	No tiene.

especiales	
Frecuencia de ocurrencia	Ocasional.

Tabla 12.9: Caso de uso extendido Desbloquear patente

Caso de uso	Desbloquear patente
Actor Primario	Usuario registrado.
Participantes e intereses	Usuario: El usuario puede desbloquear una patente bloqueada con anterioridad.
Propósito	Permitir el desbloqueo de una patente permitiendo el realizar cambios sobre esta.
Precondiciones	El usuario debe estar registrado y haber ejecutado con anterioridad el caso de uso “Inicio de sesión”.
Poscondiciones	El desbloqueo debe registrarse en el sistema.
Escenario Principal	<ol style="list-style-type: none"> 1. El usuario selecciona “Desbloquear” en el sistema. 2. El usuario elige una patente para desbloquear. 3. El usuario elije entre las opciones “Desbloquear” o “Cancelar” 4. El usuario finaliza el caso de uso.
Extensiones	2.a. Si la patente ingresada no es encontrada, se le pedirá un nuevo ingreso al usuario.

	<p>2.b. En caso de intentar desbloquear una patente desbloqueada, se le indicara esto al usuario y se le pedirá un nuevo ingreso</p> <p>3.a. Si la actividad seleccionada es “Desbloquear” se desbloqueara la patente.</p> <p>3.c. Si la actividad seleccionada es “Cancelar”, el sistema cerrara la ventana “Desbloquear” volviendo a la pantalla principal.</p>
Requerimientos especiales	No tiene.
Frecuencia de ocurrencia	Ocasional.

Tabla 12.10: Caso de uso extendido Búsqueda patentes

Caso de uso	Búsqueda patentes.
Actor Primario	Usuario registrado.
Participantes e intereses	Usuario: El usuario puede buscar una patente o un grupo de estas.
Propósito	La búsqueda de una patente y sus respectivos datos o también de un grupo de estas.
Precondiciones	El usuario debe estar registrado y haber ejecutado con anterioridad el caso de uso “Inicio de sesión”.

Poscondiciones	No tiene
Escenario Principal	<ol style="list-style-type: none"> 1. El usuario selecciona “Búsqueda patentes” en el sistema. 2. El sistema despliega la pantalla con las opciones de búsqueda. 3. El usuario selecciona las opciones e ingresa los datos correspondientes. 4. El usuario selecciona entre las actividades “Buscar” o “Cancelar”. 5. Se le presentan por pantalla al usuario los resultados de la búsqueda.
Extensiones	<p>4.a. Si la actividad seleccionada es “Buscar”, se realizara la búsqueda y desplegara los datos en la pantalla principal. En caso de no encontrarse datos se le indicara la usuario</p> <p>4.c. Si la actividad seleccionada es “Cancelar”, el sistema cerrara la ventana “Búsqueda patentes” volviendo a la pantalla principal.</p>
Requerimientos especiales	No tiene.
Frecuencia de ocurrencia	Alta.

12.3 Diagramas de secuencia

Un diagrama de secuencia muestra la interacción de un conjunto de objetos en una aplicación a través del tiempo. Esto da un detalle de los casos de usos, mostrando el traspaso de mensaje entre los objetos y el manejo de estos en el contexto de una operación [9].

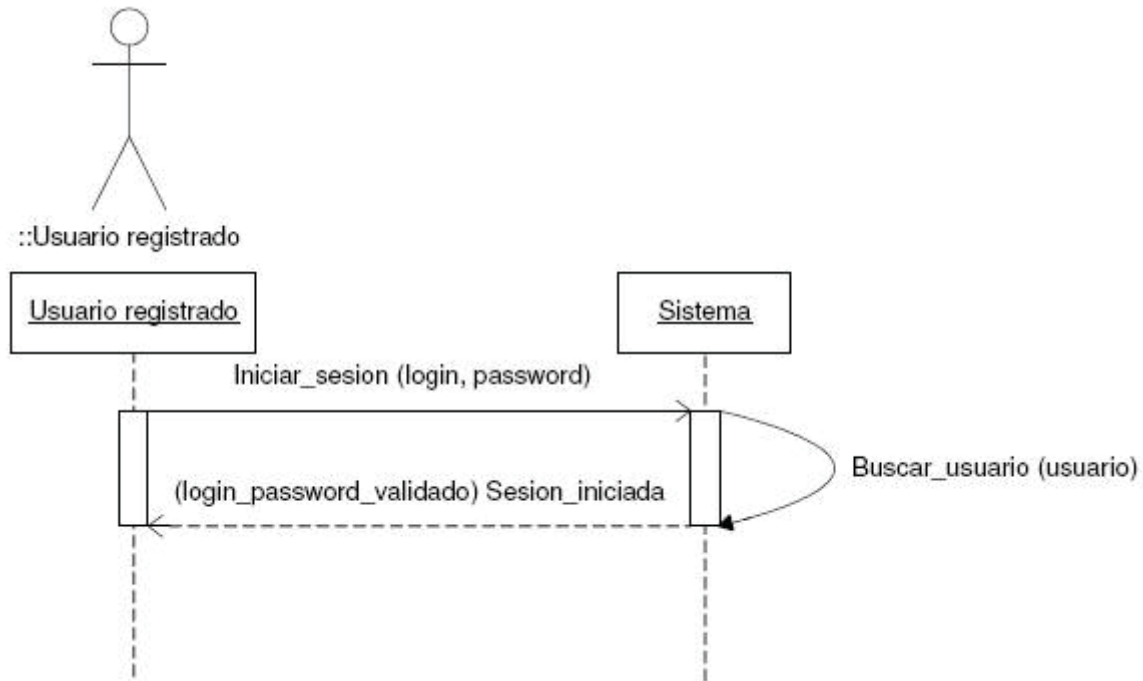


Figura 12.2: Diagrama de secuencia Inicio de sesión

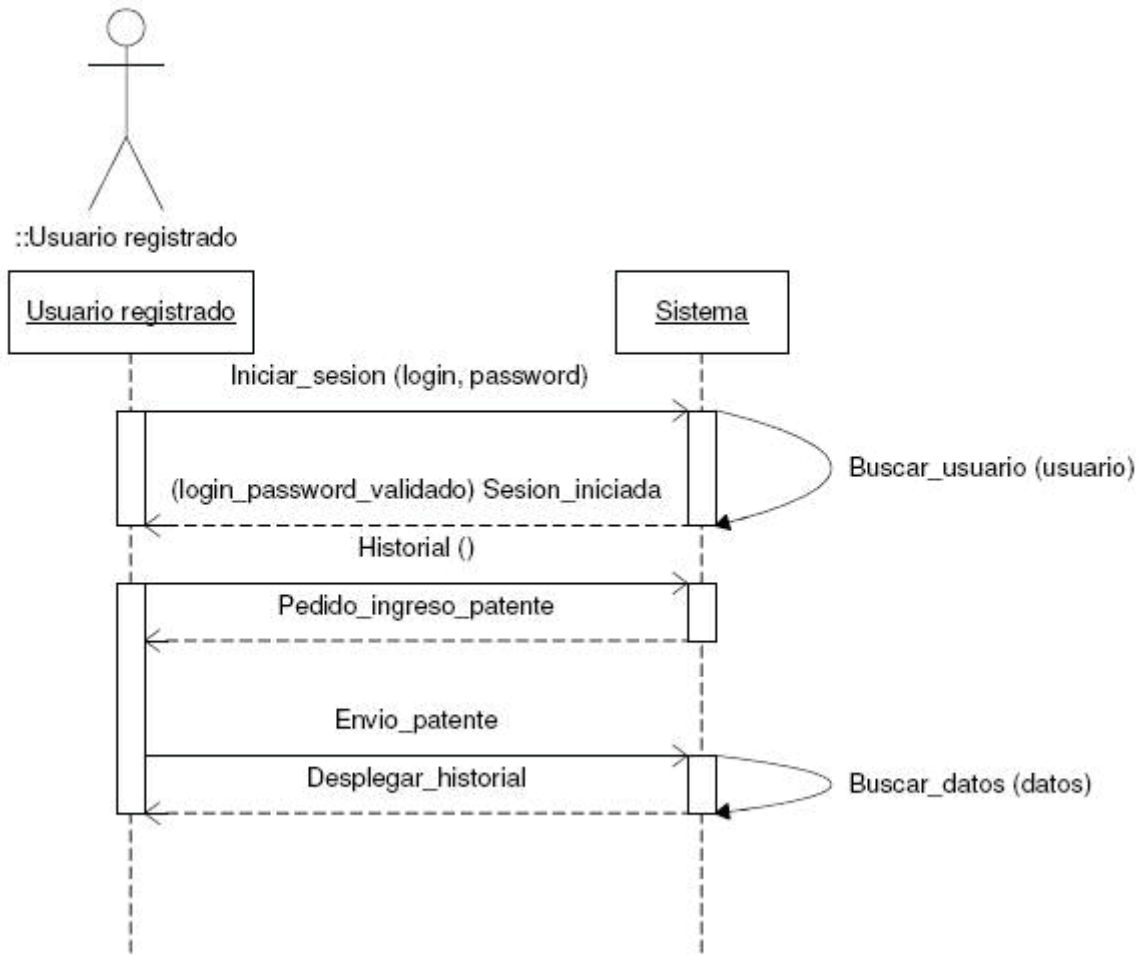


Figura 12.3: Diagrama de secuencia Historial

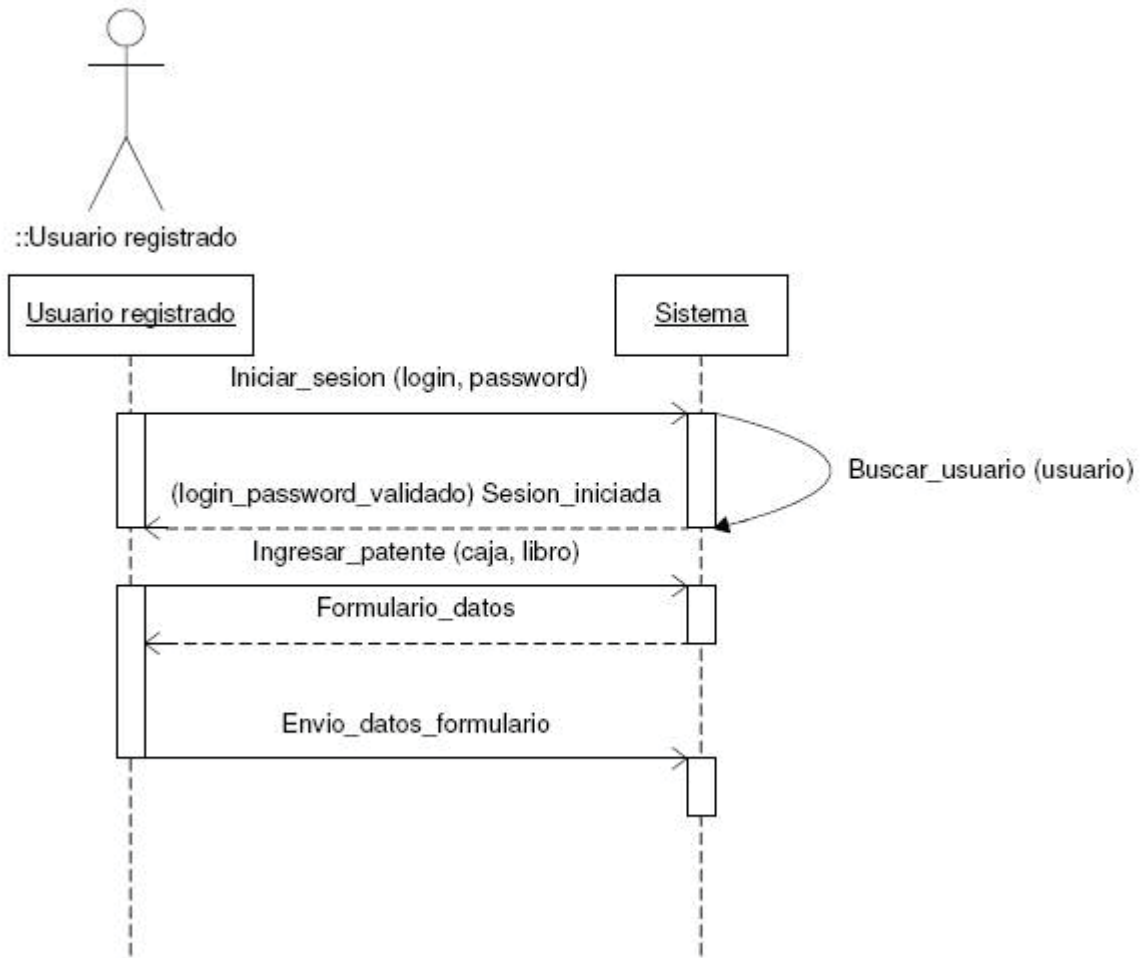


Figura 12.4: Diagrama de secuencia Ingresar patente

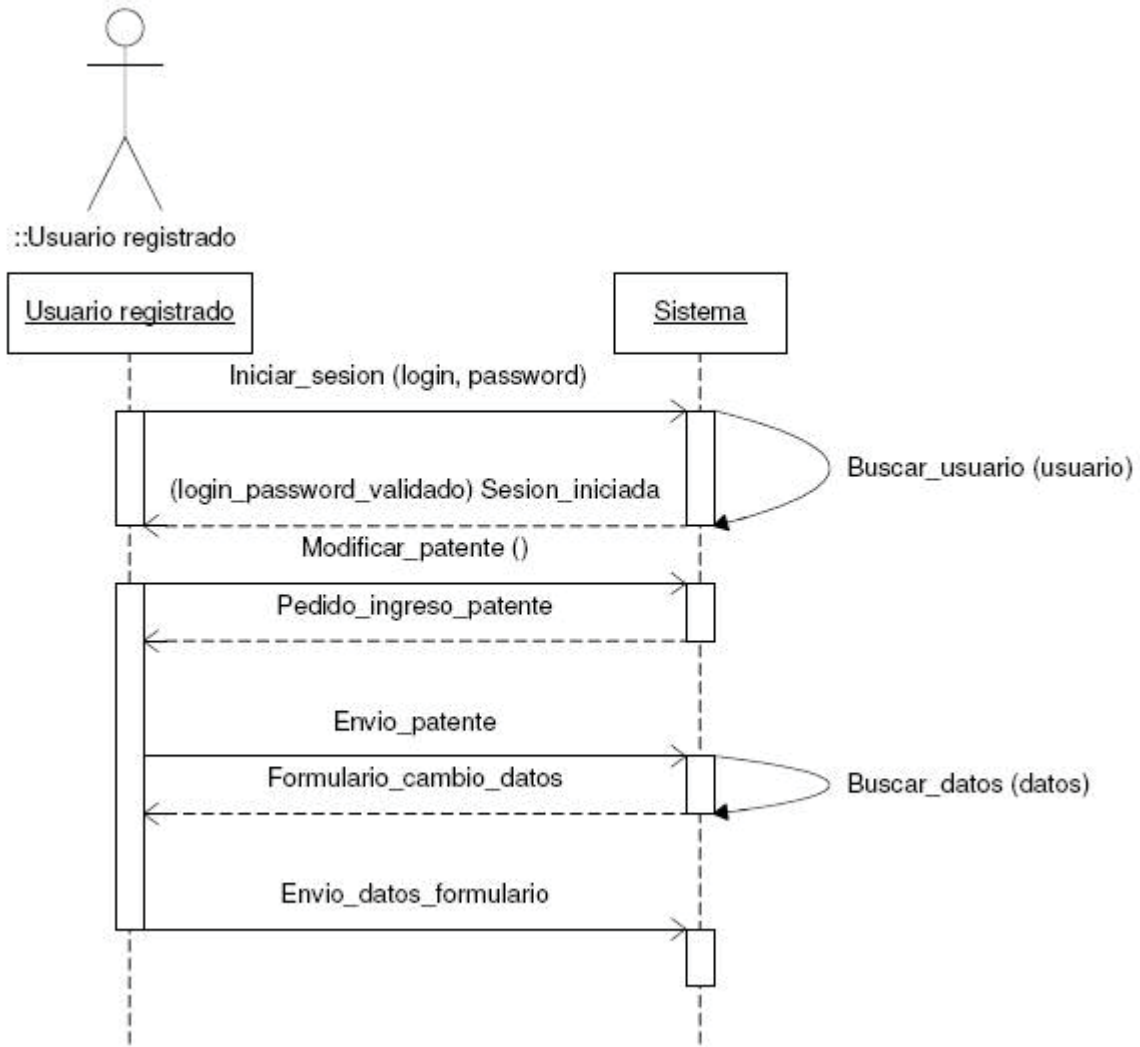


Figura 12.5: Diagrama de secuencia Modificar patente

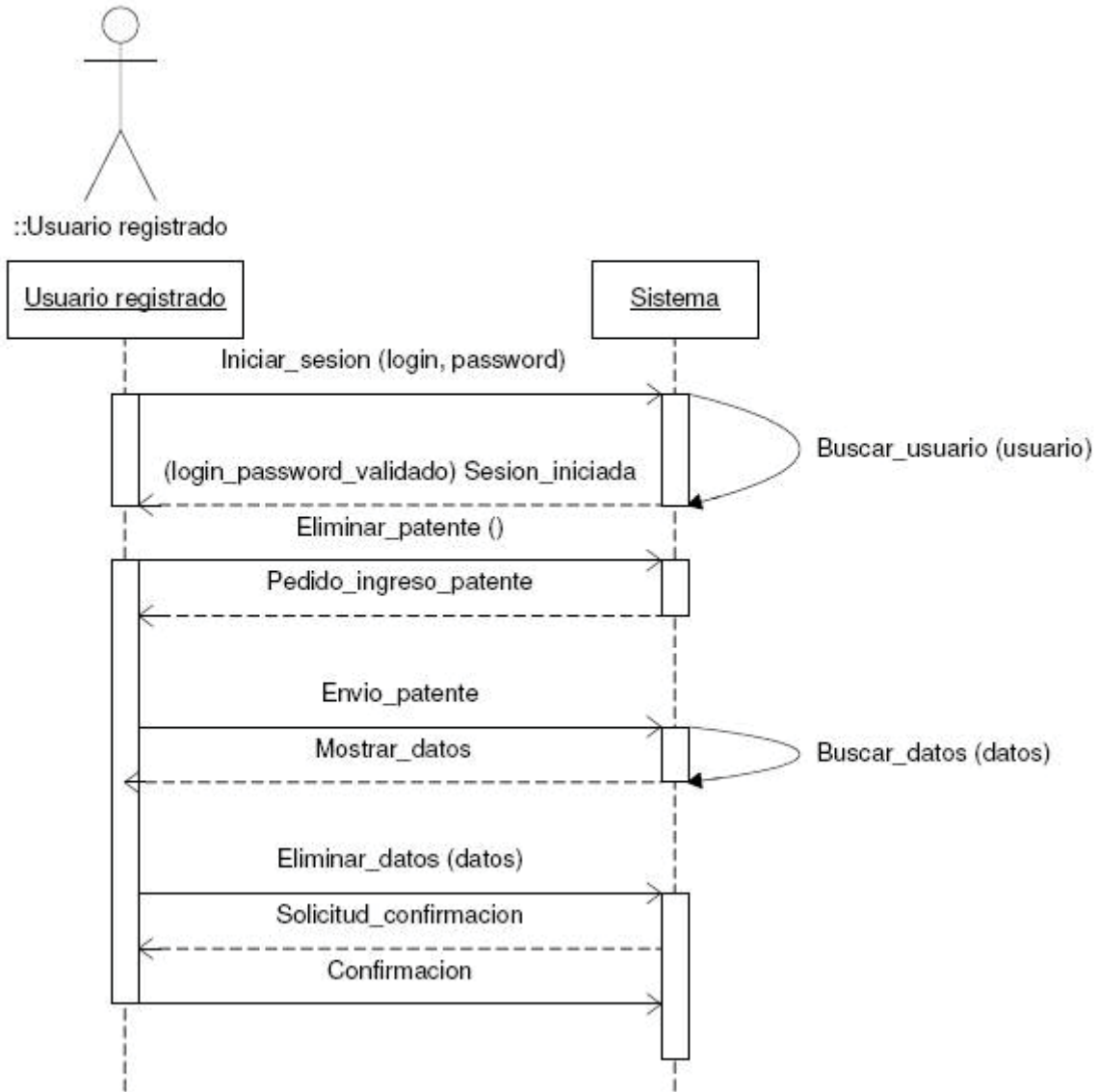


Figura 12.6: Diagrama de secuencia Eliminar patente

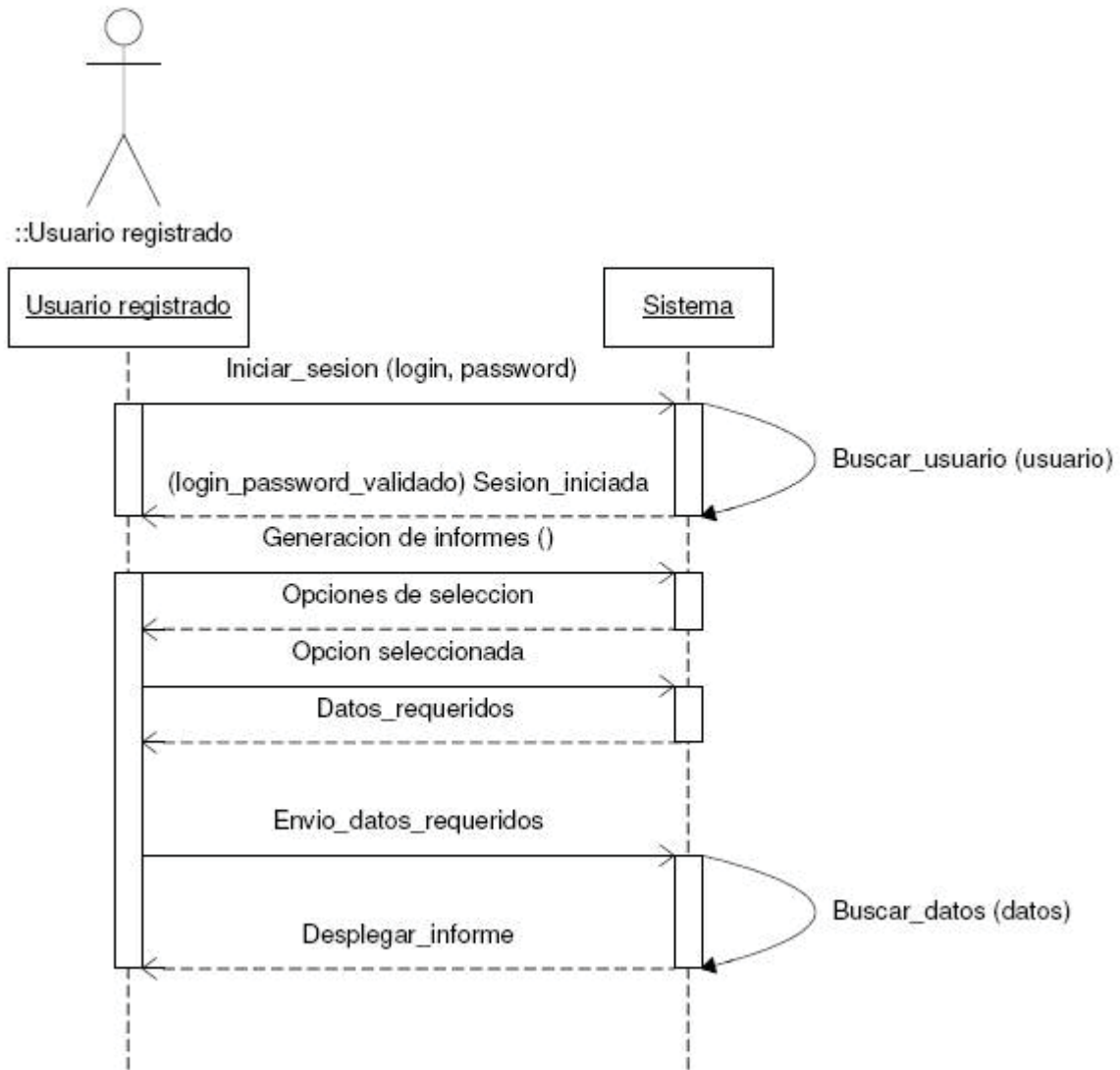


Figura 12.7: Diagrama de secuencia Generación de informes

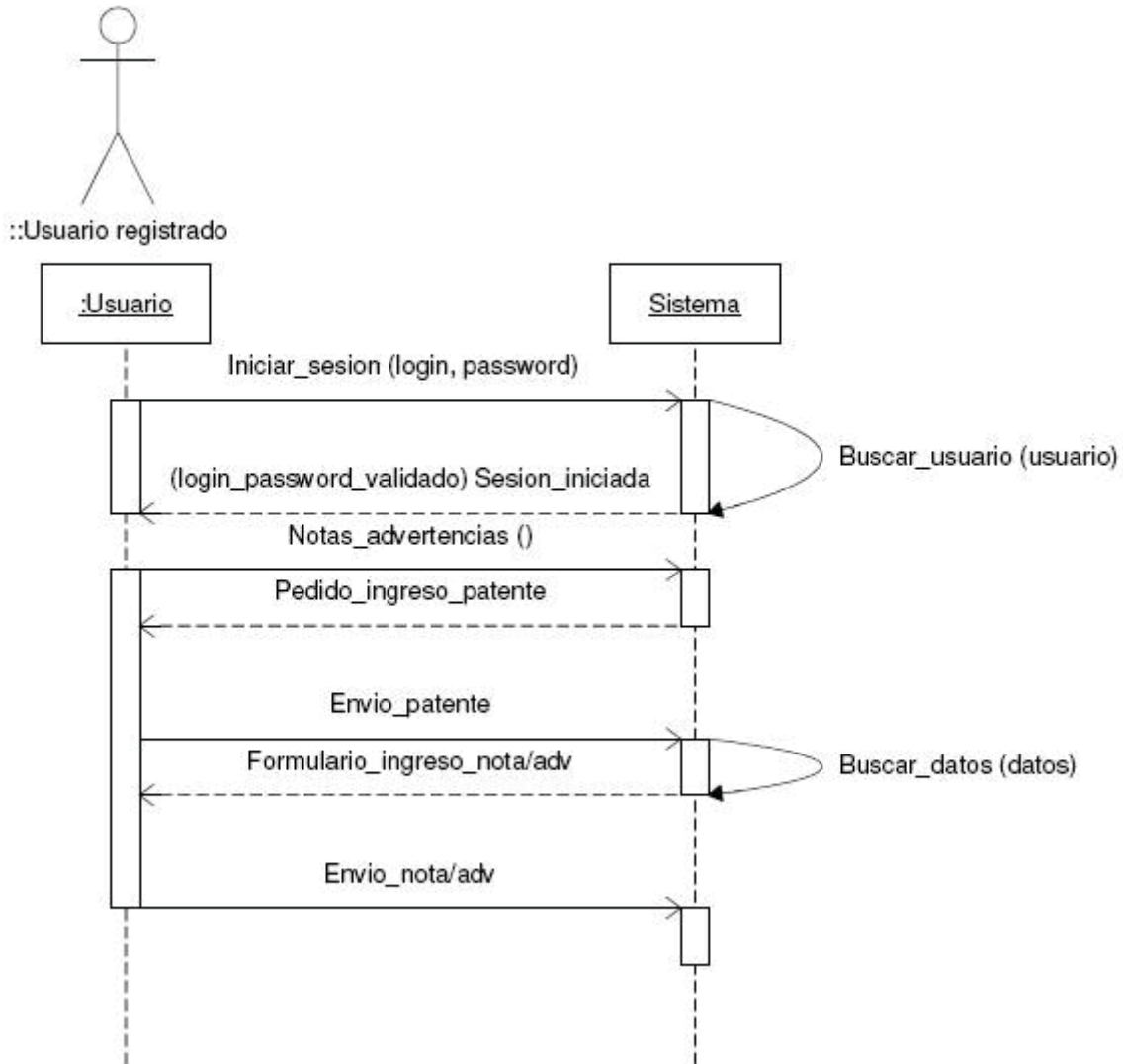


Figura 12.8: Diagrama de secuencia Notas y advertencias

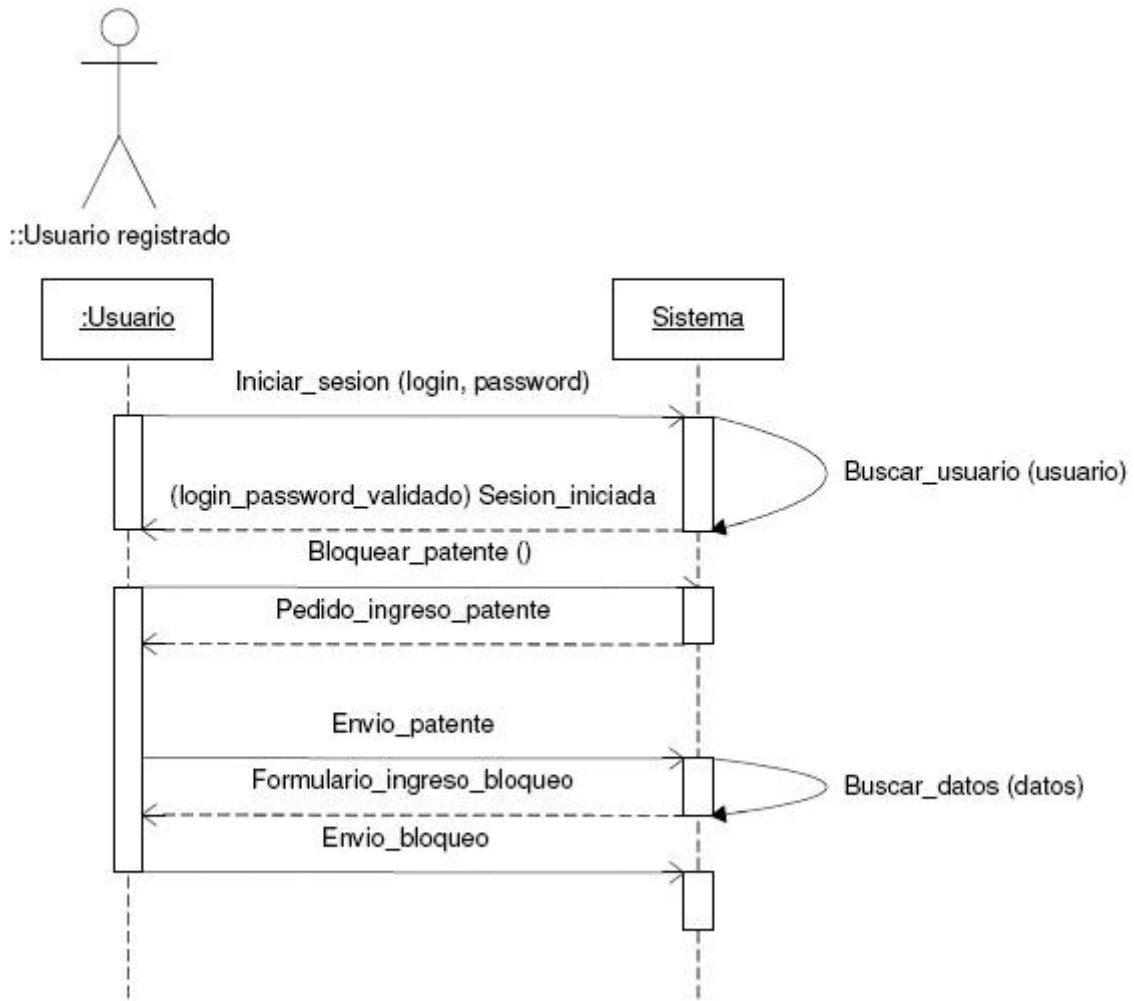


Figura 12.9: Diagrama de secuencia Bloquear patente

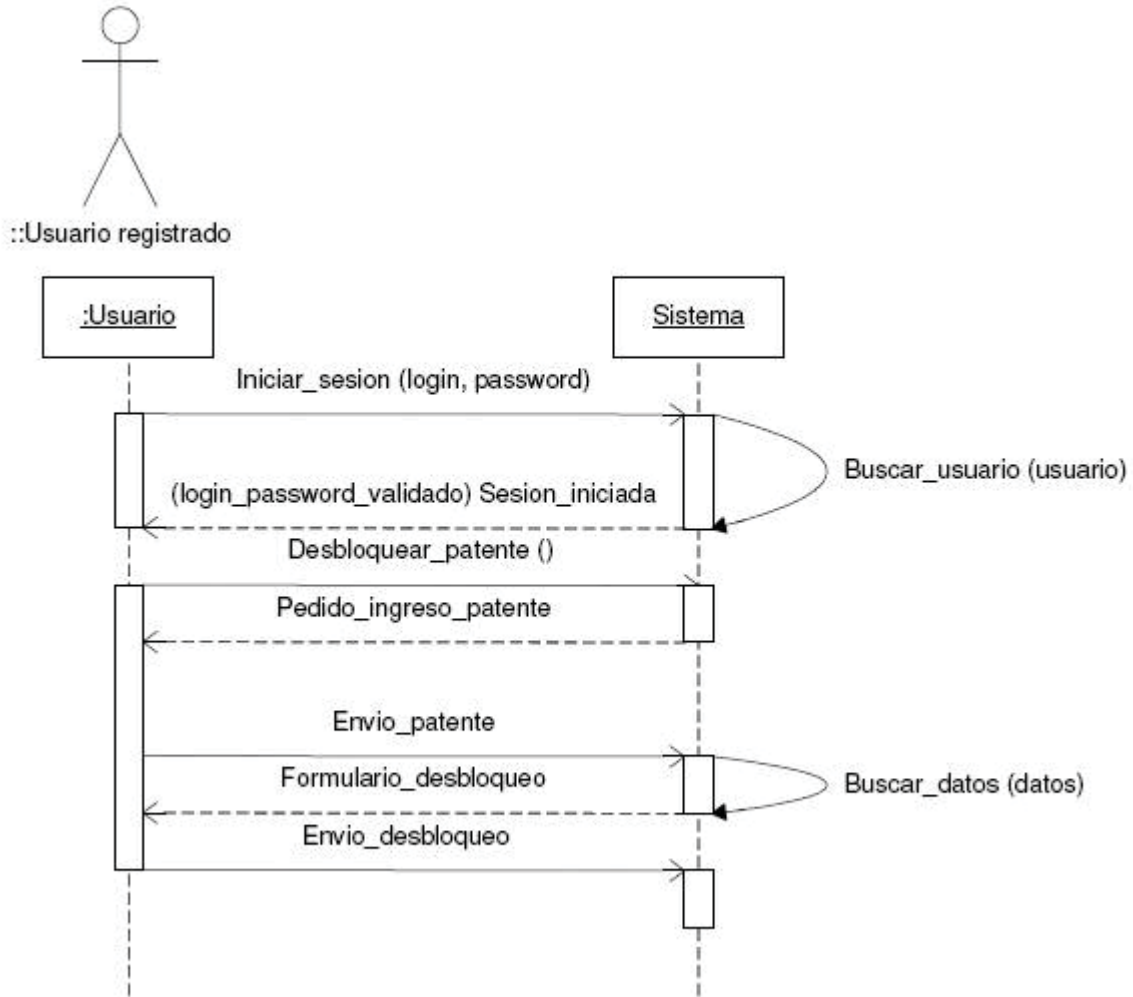


Figura 12.10: Diagrama de secuencia Desbloquear patente

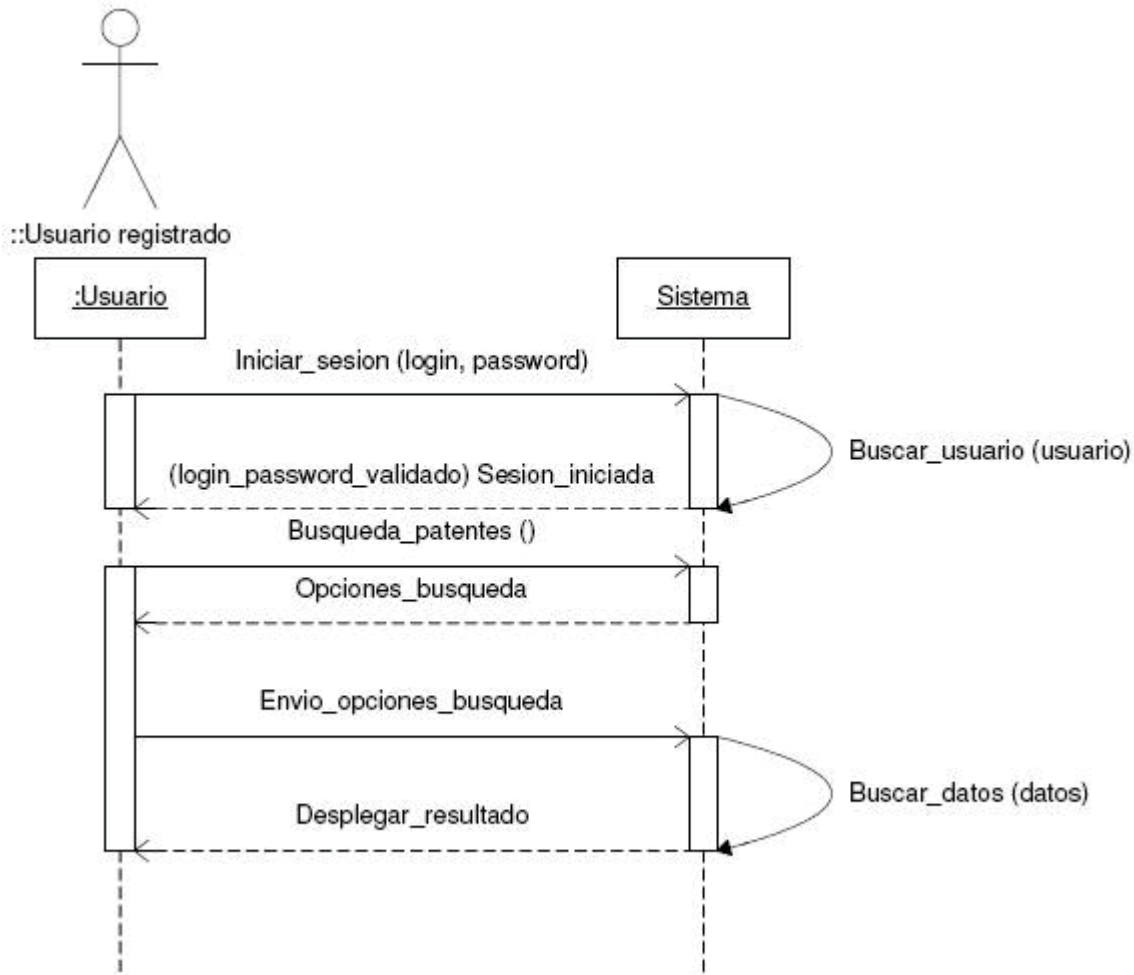


Figura 12.11: Diagrama de secuencia Búsqueda patentes

12.4 Diagramas de actividad

El diagrama de actividad es un diagrama de flujo del proceso multi-propósito que se usa para modelar el comportamiento del sistema. Los diagramas de actividad pueden mostrar procesos paralelos, lo que es importante cuando se quiere mostrar varios hilos en los programas concurrentes [6].

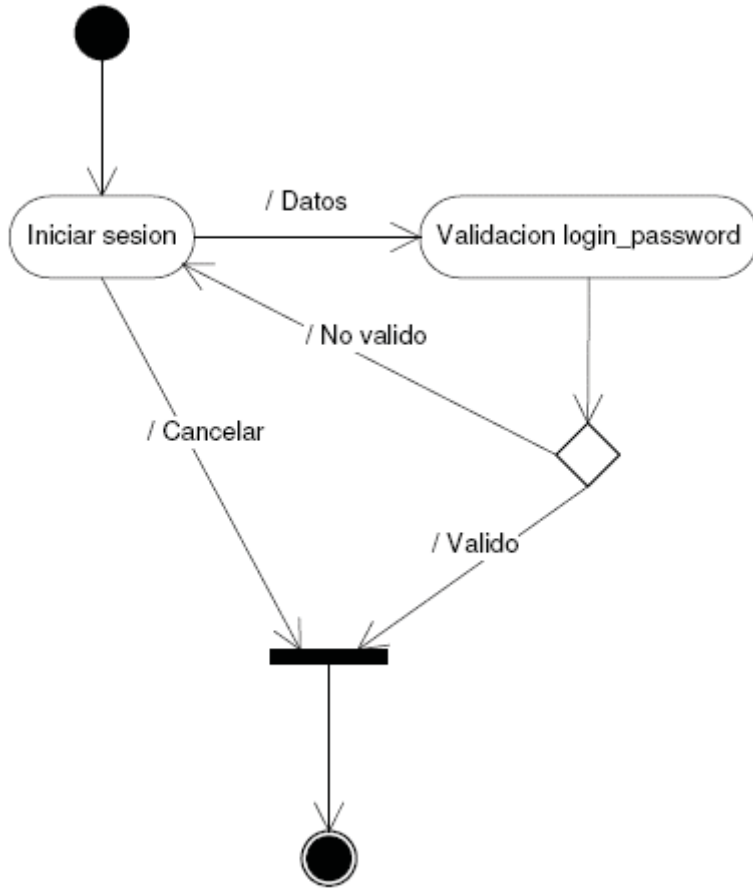


Figura 12.12: Diagrama de actividad Iniciar sesión

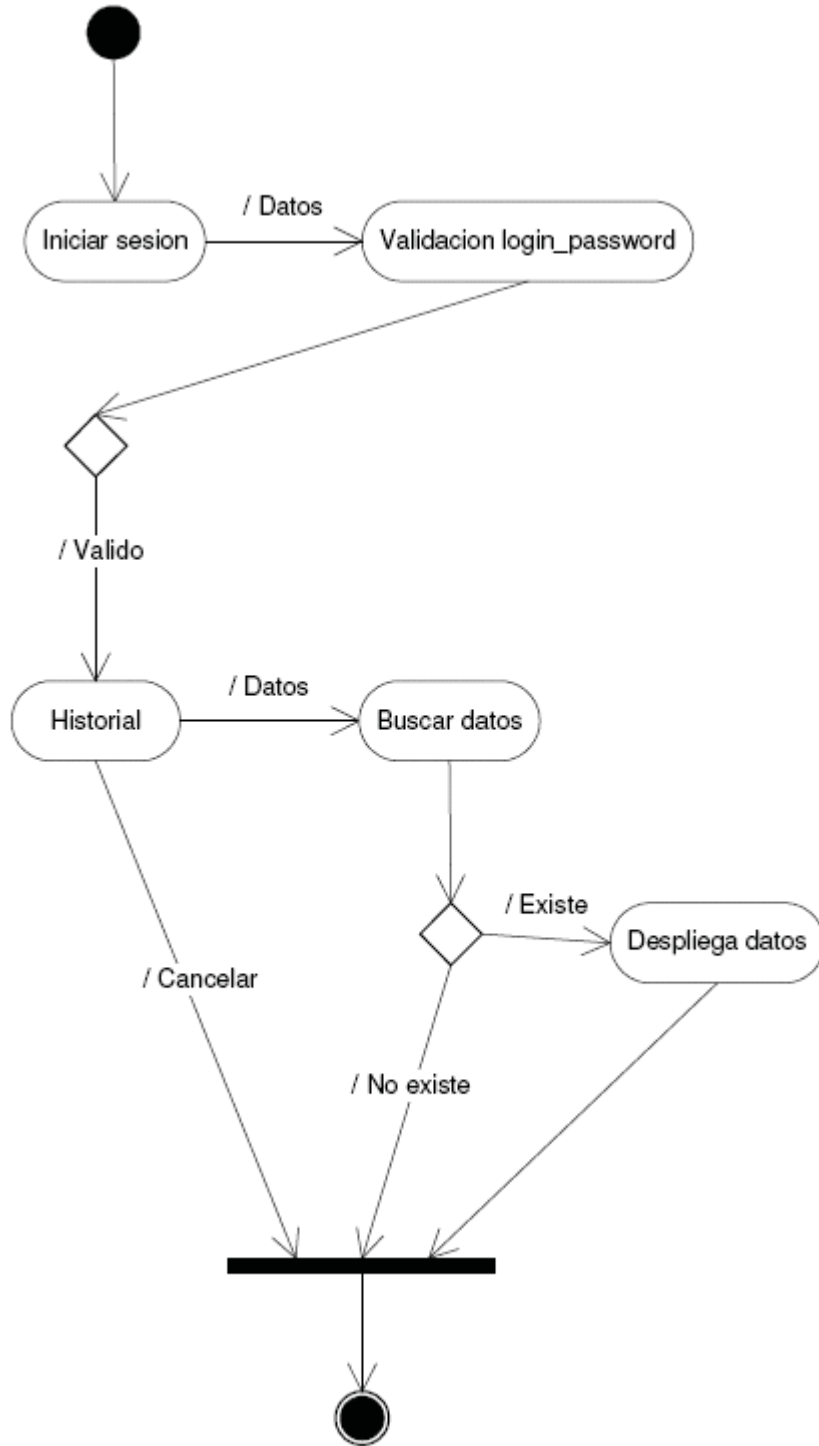


Figura 12.13: Diagrama de actividad Historial

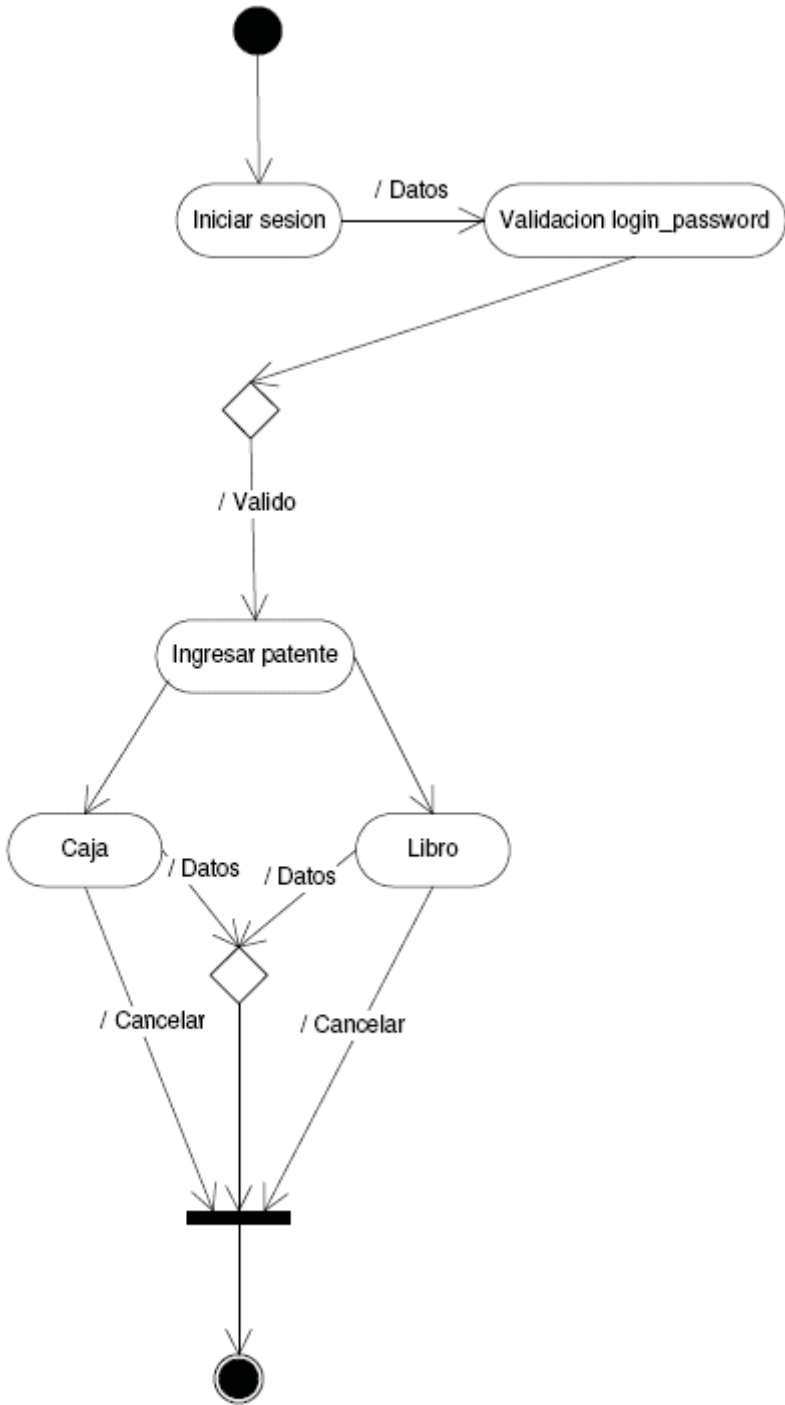


Figura 12.14: Diagrama de actividad Ingresar patente

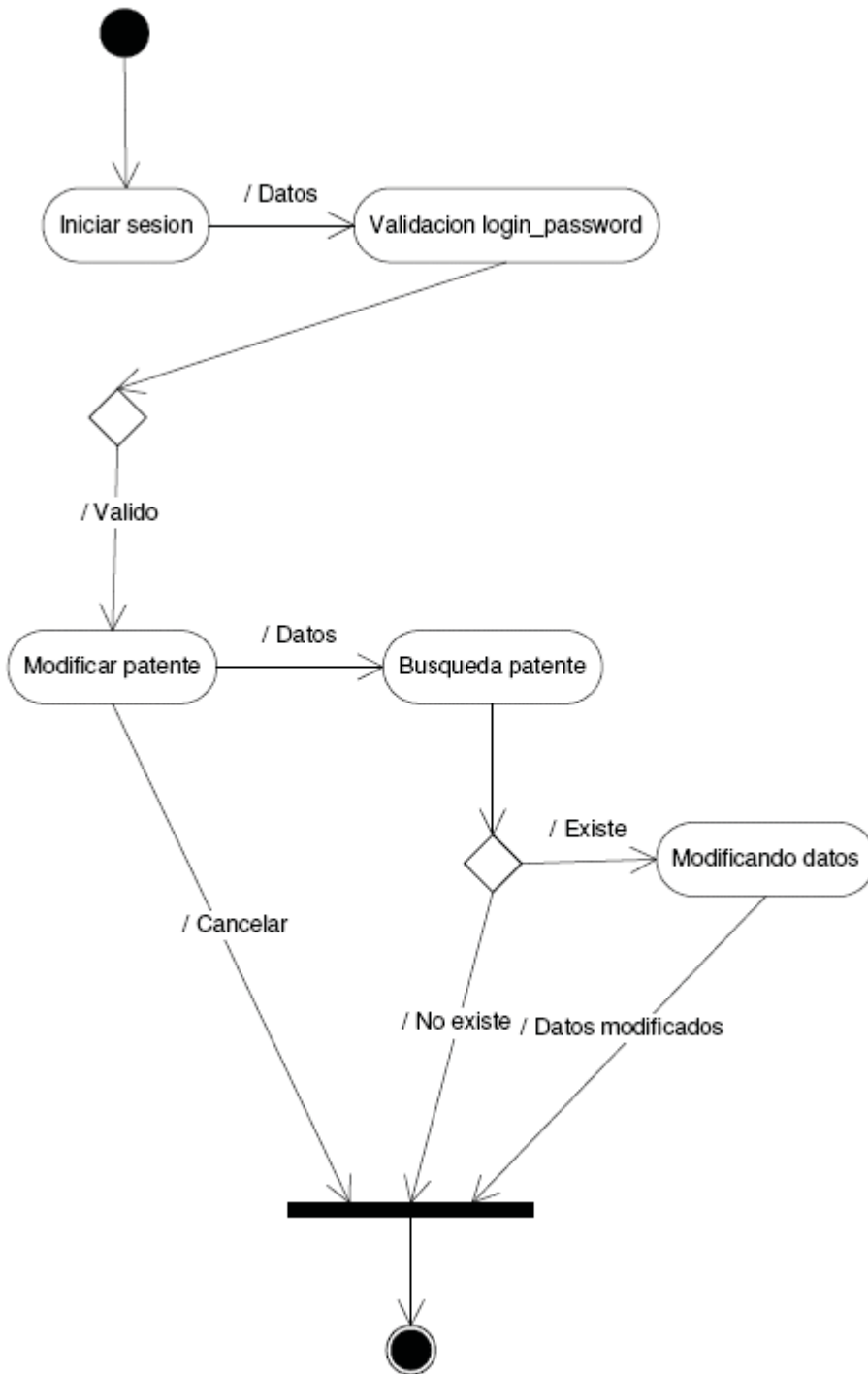


Figura 12.15: Diagrama de actividad Modificar patente

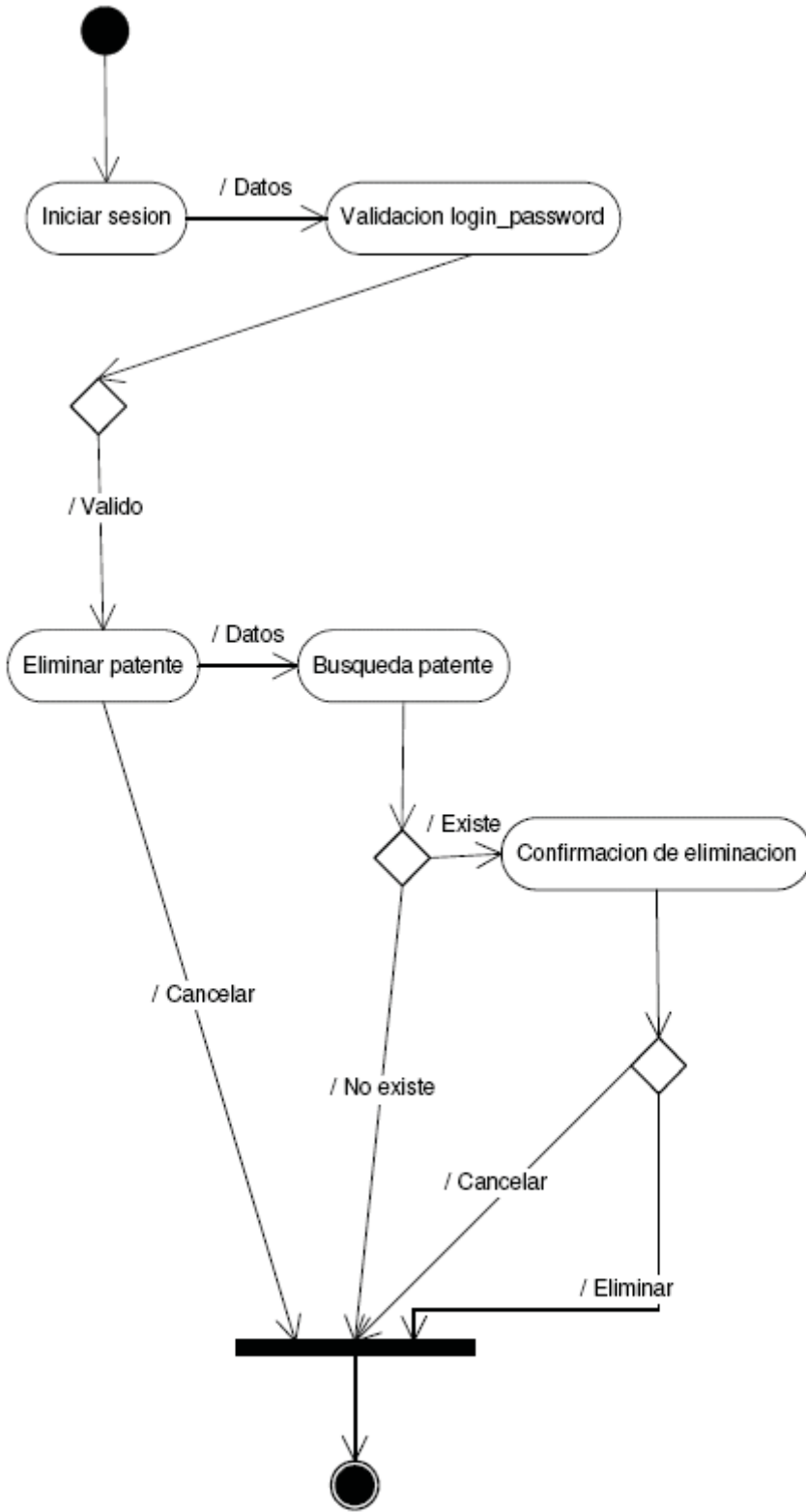


Figura 12.16: Diagrama de actividad Eliminar patente

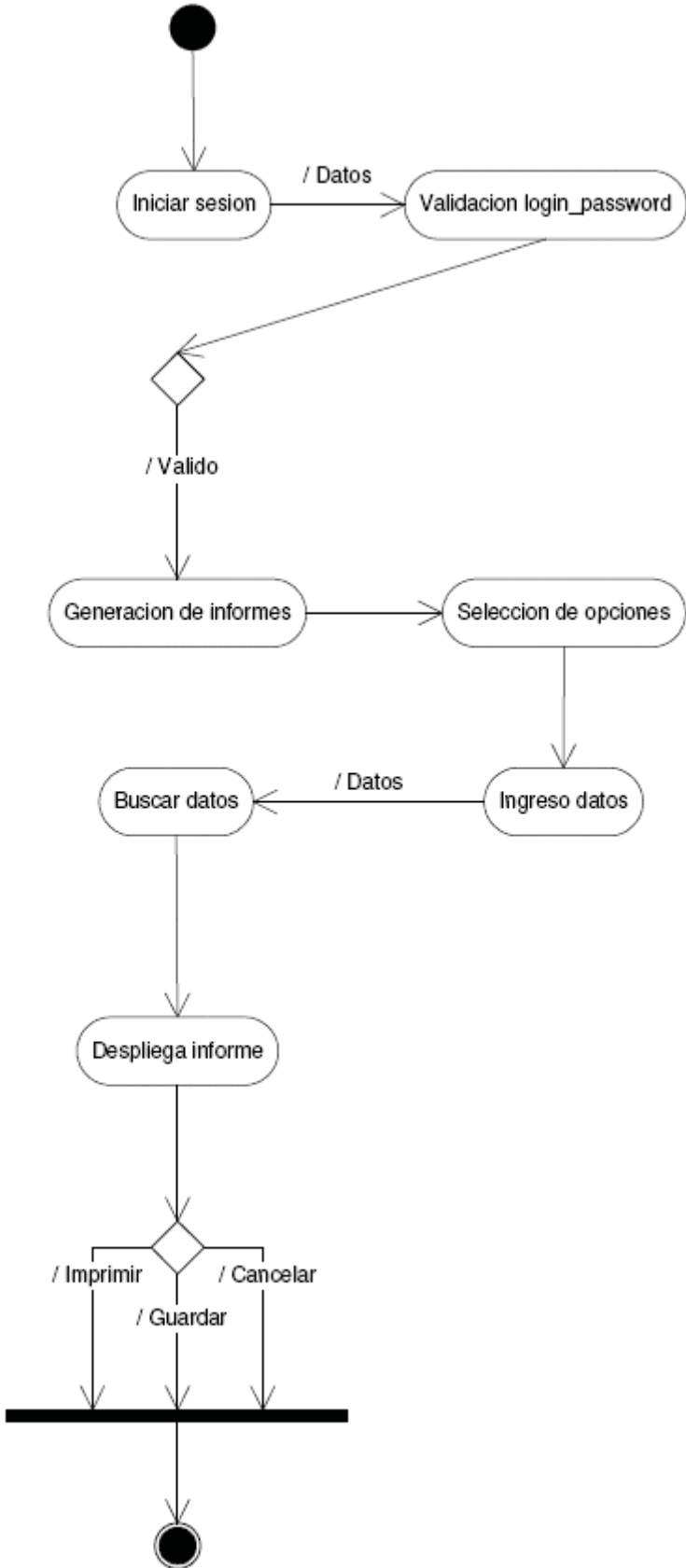


Figura 12.17: Diagrama de actividad Generación de informes

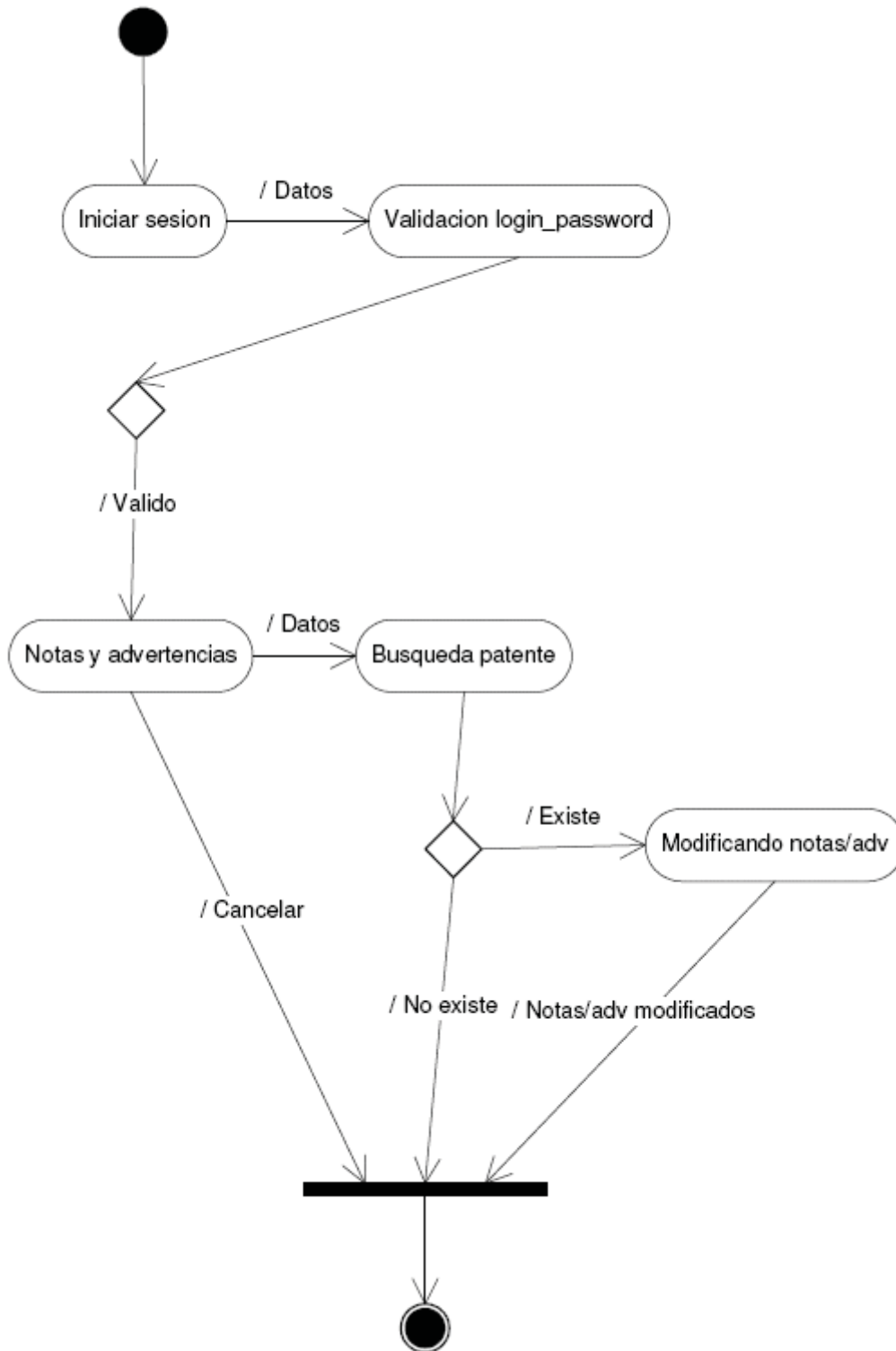


Figura 12.18: Diagrama de actividad Notas y advertencias

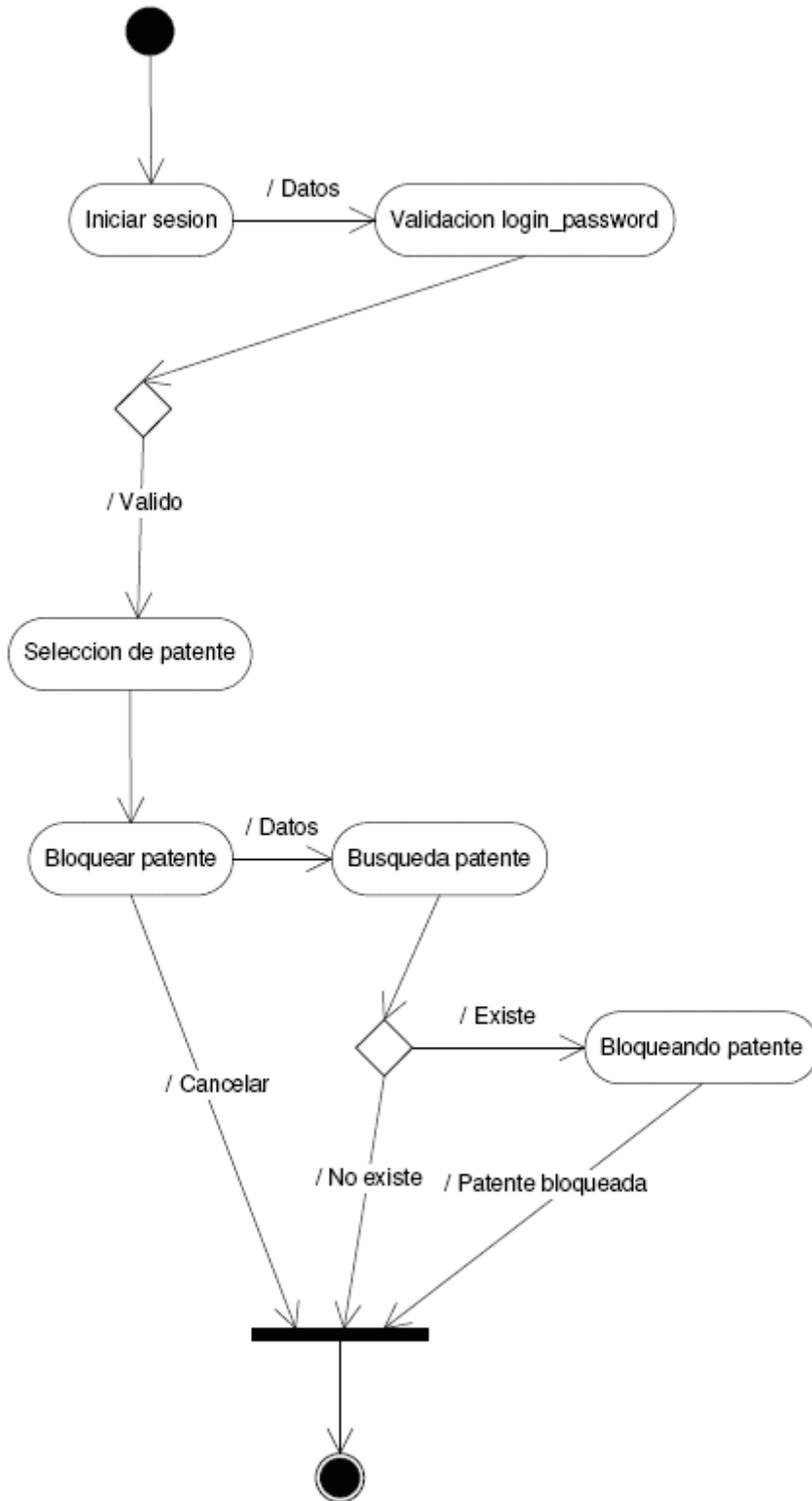


Figura 12.19: Diagrama de actividad Bloquear patente

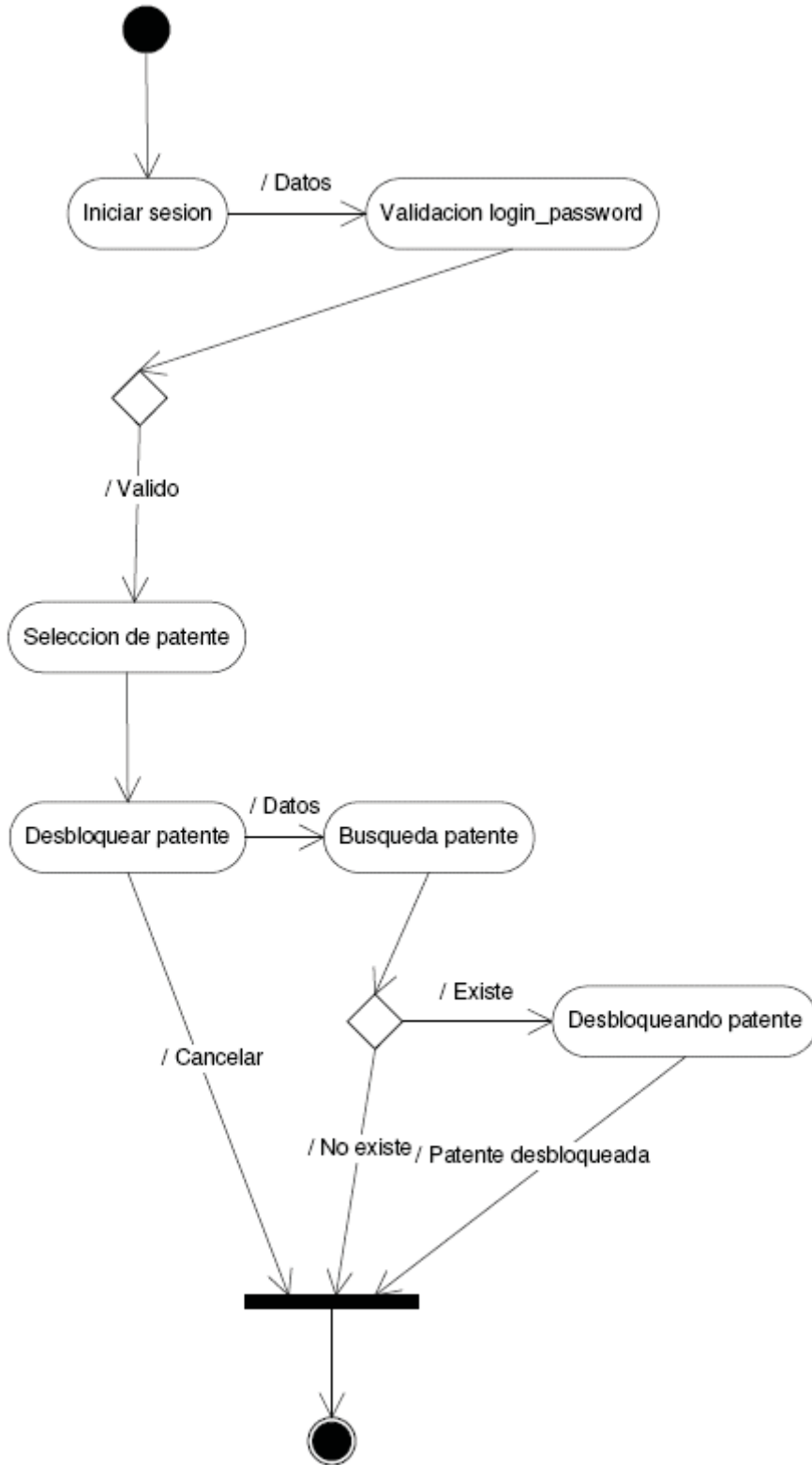


Figura 12.20: Diagrama de actividad Desbloquear patente

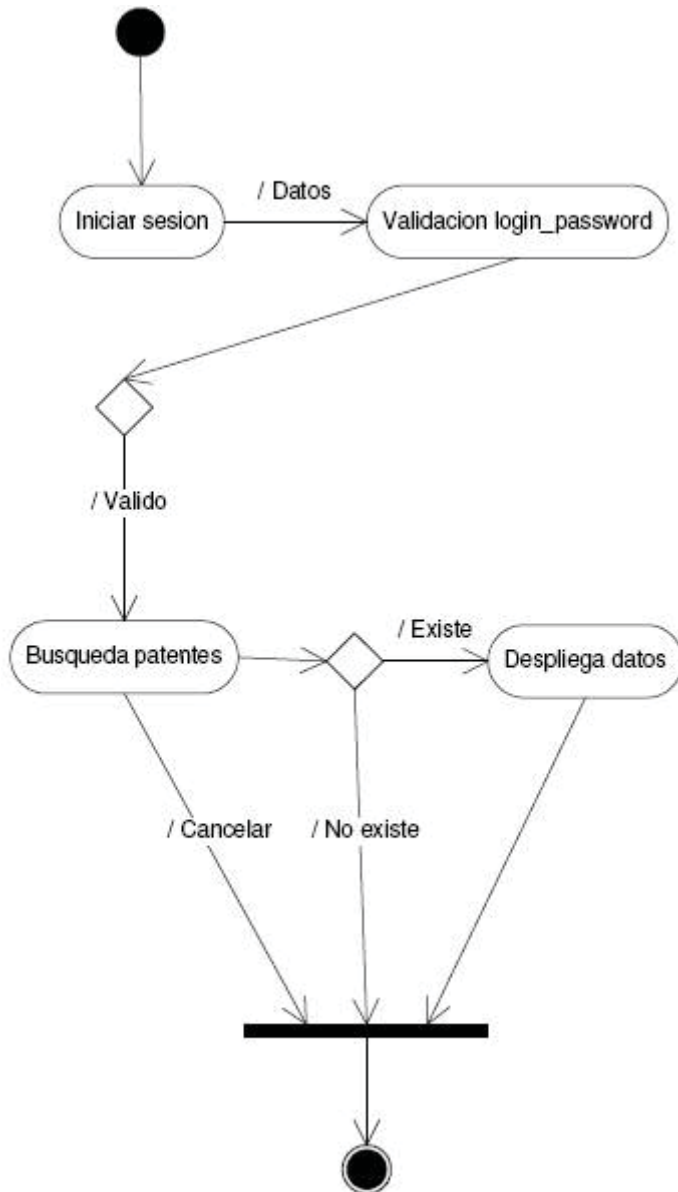


Figura 12.21: Diagrama de actividad Búsqueda patentes

12.5 Diagramas de componentes

Los diagramas de componentes es un diagrama de implementación que muestra las dependencias lógicas entre componentes software, sean estos componentes fuentes, binarios o ejecutables. Estos componentes software tienen tipo, indicando si son útiles en tiempo de compilación, enlace o ejecución [10].

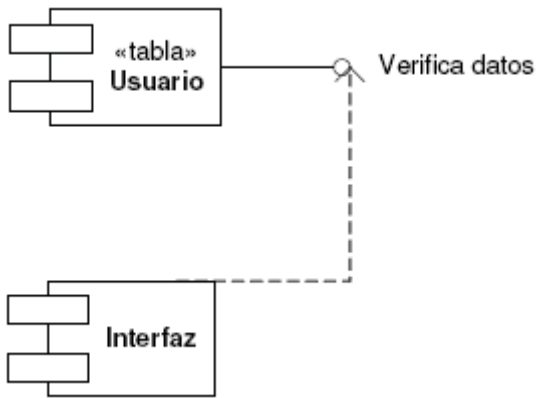


Figura 12.22: Diagrama de componente Inicio de sesión

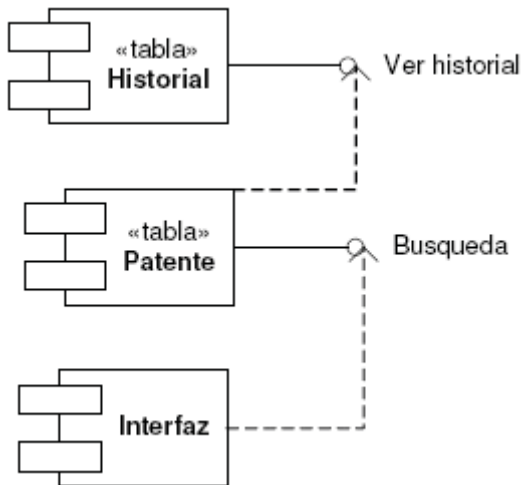


Figura 12.23: Diagrama de componente Historial

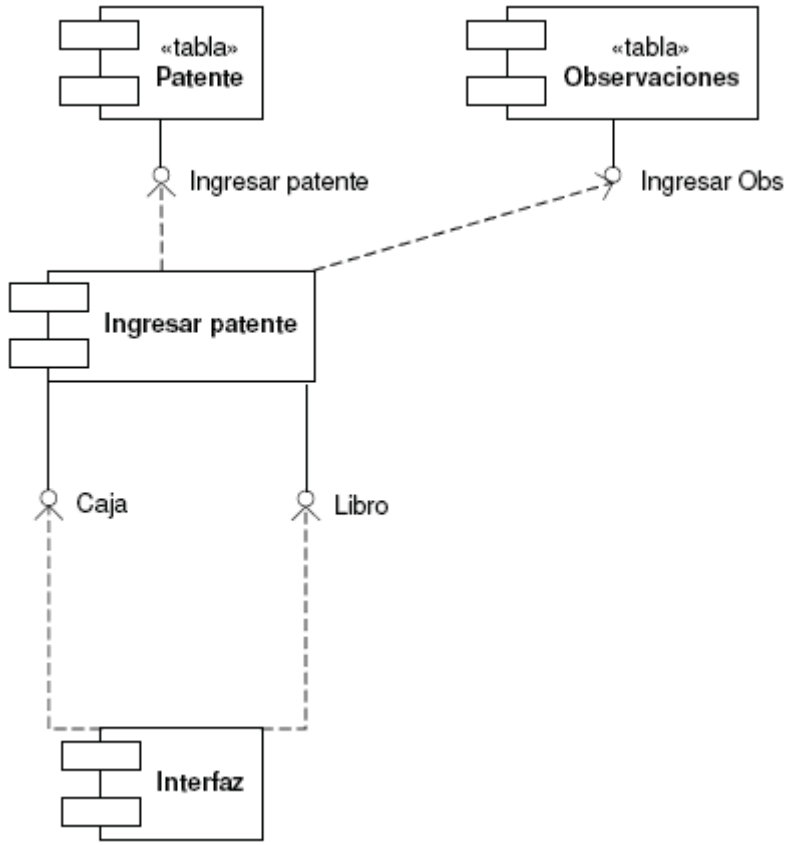


Figura 12.24: Diagrama de componente Ingresar patente

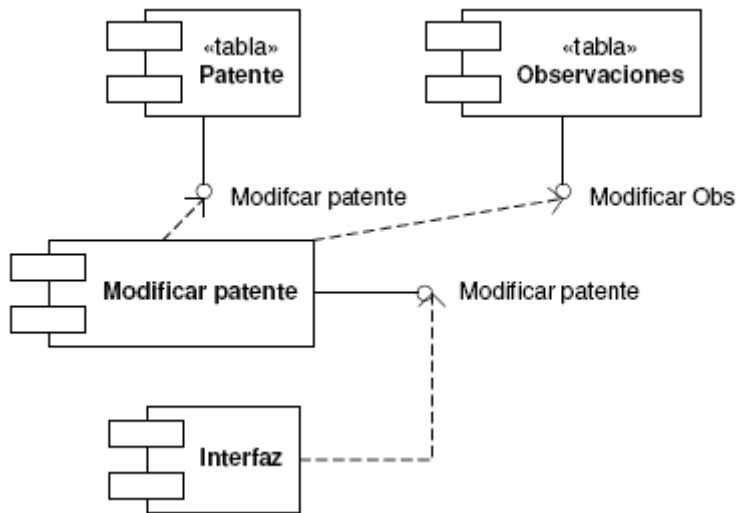


Figura 12.25: Diagrama de componente Modificar patente

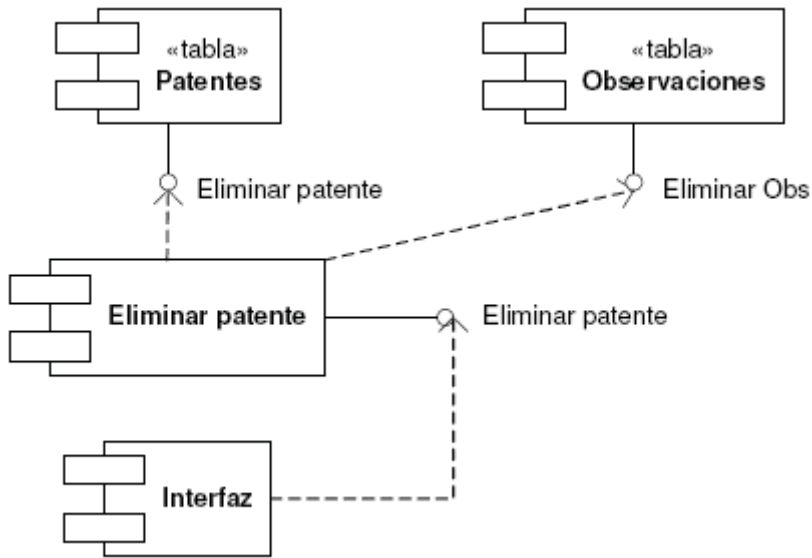


Figura 12.26: Diagrama de componente Eliminar patente

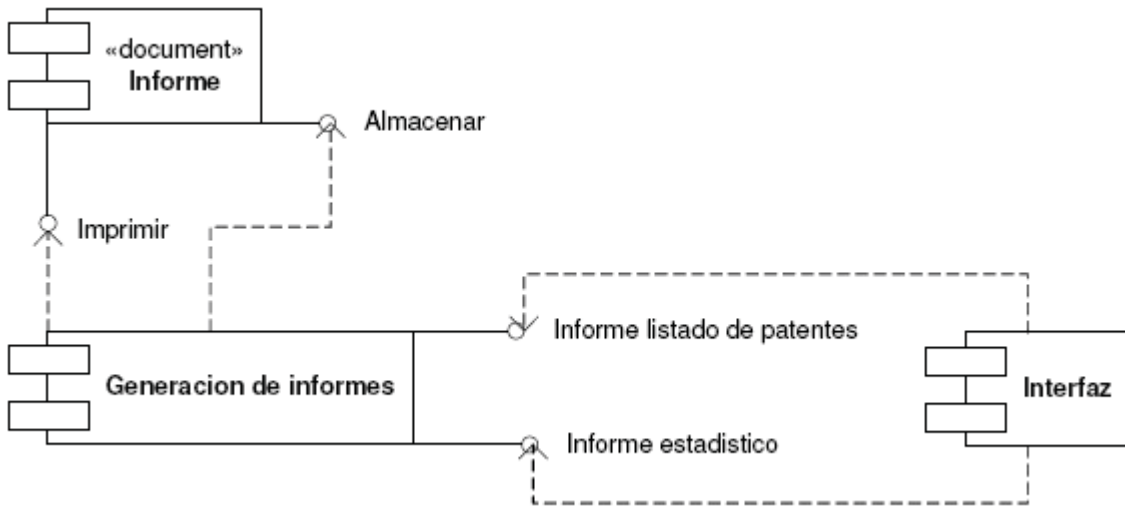


Figura 12.27: Diagrama de componente Generación de informes

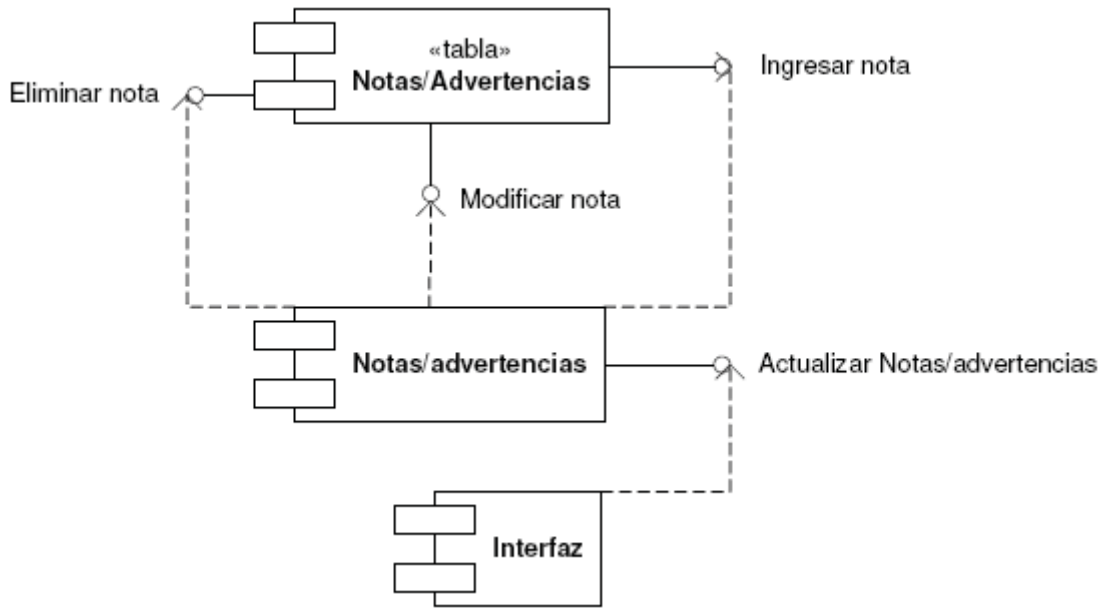


Figura 12.28: Diagrama de componente Notas y advertencias

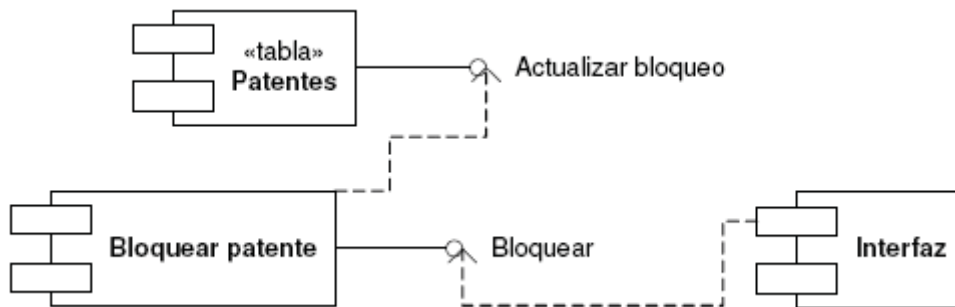


Figura 12.29: Diagrama de componente Bloquear patente

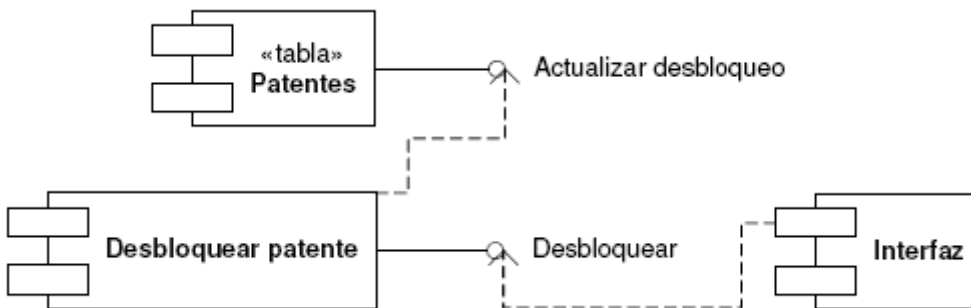


Figura 12.30: Diagrama de componente Desbloquear patente

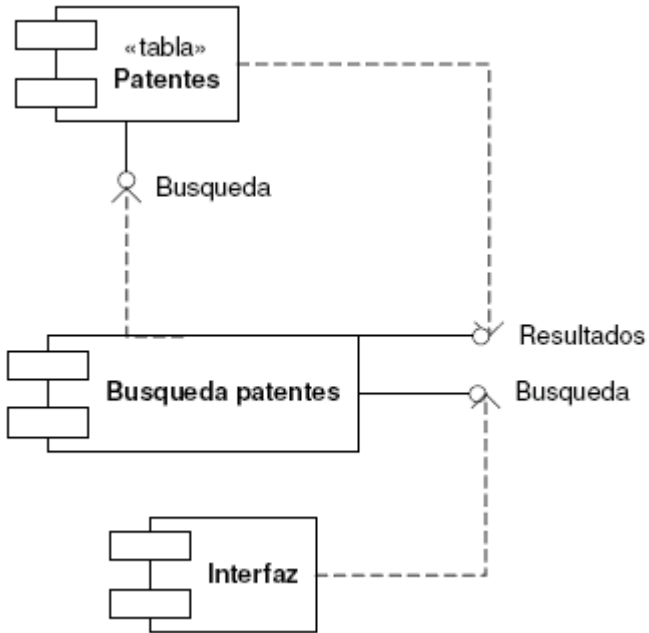


Figura 12.31: Diagrama de componente Búsqueda patentes

12.6 Interfaces

A continuación se presentan las interfaces a la que un usuario registrado tendrá acceso dentro del sistema.

The screenshot shows a login window titled "Inicio sesion". It has a blue header bar. Below the header, there are two radio buttons: "Iniciar sesion como usuario del sistema" (selected) and "Iniciar sesion como invitado". To the right of the "Iniciar sesion como usuario del sistema" option is a blue crest logo with the motto "VIRE ROS MARI". Below the radio buttons are two text input fields: "Nombre de Usuario" and "Contraseña". At the bottom right, there are two buttons: "Aceptar" and "Cancelar".

Figura 12.32 Interfaz Inicio de sesión

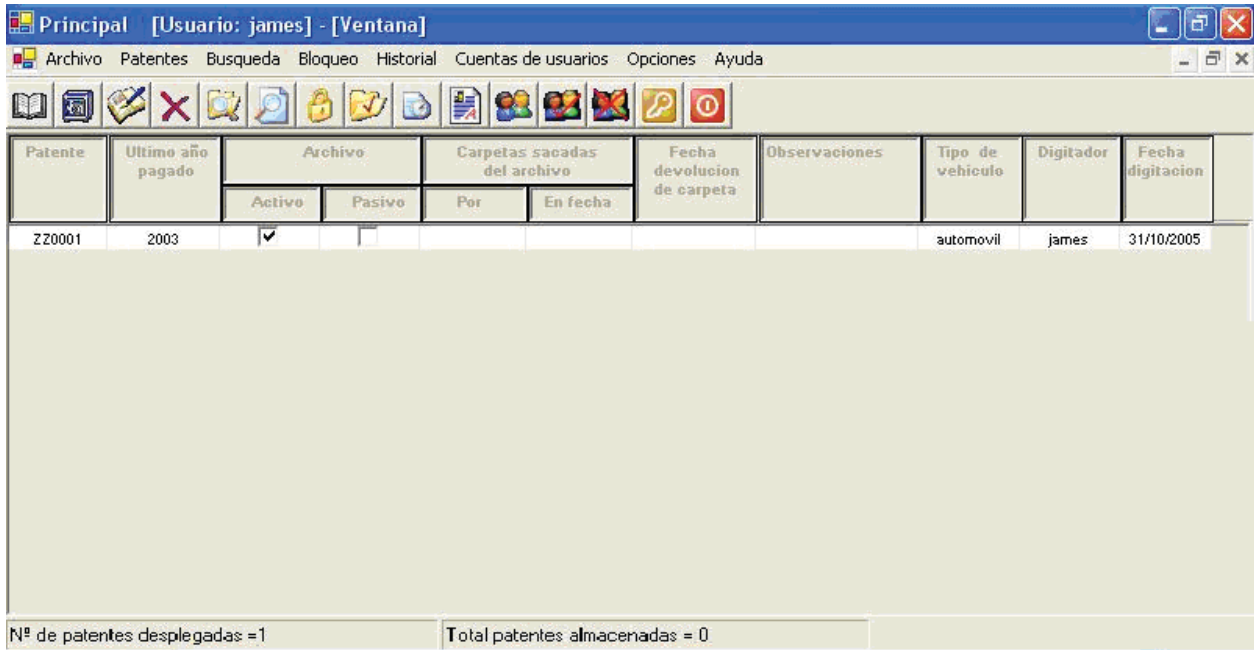


Figura 12.33: Interfaz principal

Esta interfaz corresponde a la pantalla principal del sistema, en la cual serán desplegados los datos de las búsquedas y ofrece las funcionalidades dependiendo del tipo de usuario.

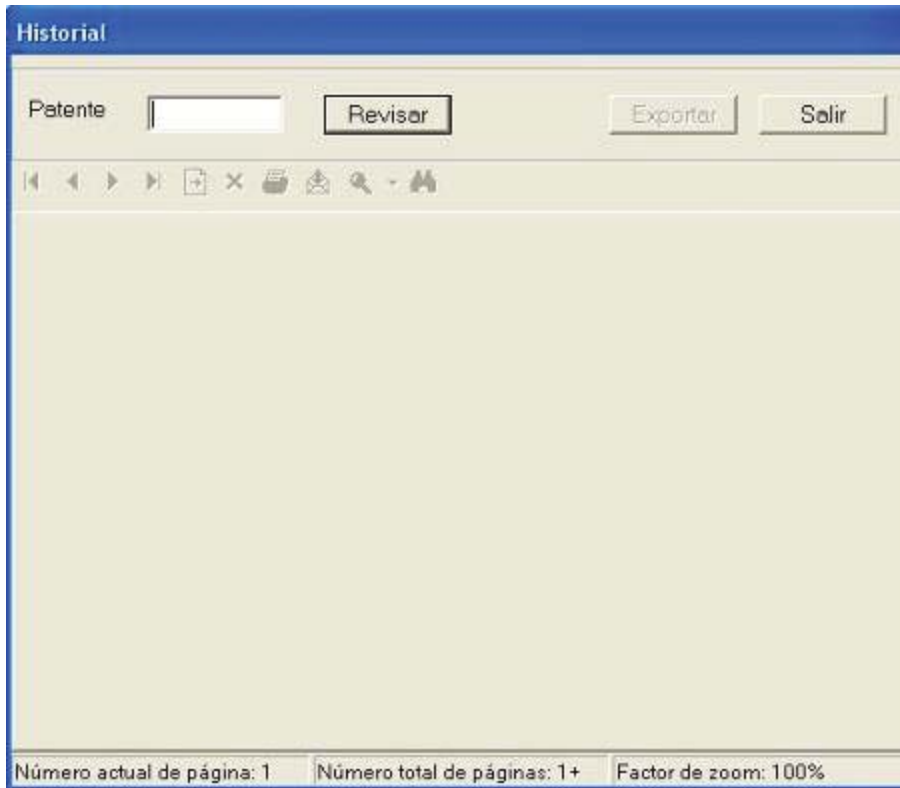


Figura 12.34: Interfaz Historial

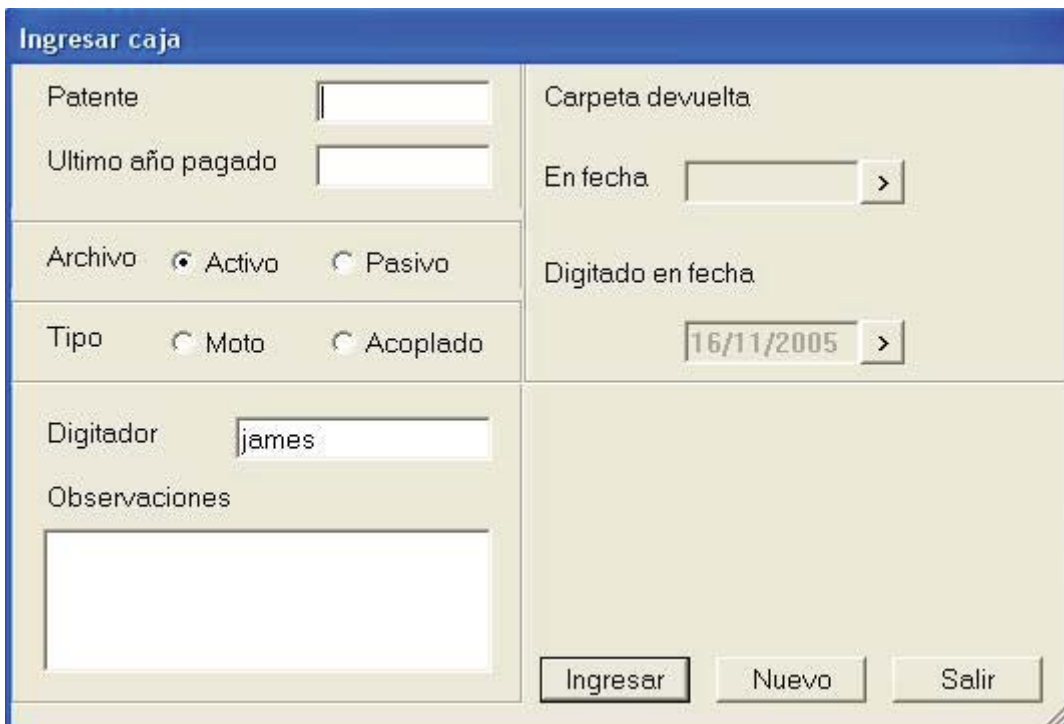


Figura 12.35 Interfaz Ingresar patente: Caja

Ingresar libro

Patente

Archivo Activo Pasivo

Digitador

Observaciones

Carpeta sacada del archivo

Por

En fecha

Digitado en fecha

>

Figura 12.36 Interfaz Ingresar patente: Libro

Modificar patente

Busqueda

Patente

Datos

Ultimo año pagado

Archivo Activo pasivo

Tipo Moto Acopladas

Digitador

Observaciones

Carpetas sacadas del archivo

Por

En fecha

Carpeta devuelta

En fecha

Digitado en fecha

Figura 12.37: Interfaz Modificar patente

Eliminar patente

Patente

Detos

Ultimo año pagado Observaciones

Archivo

Carpetas sacadas del archivo

Por Motos o acoplados

En fecha Digitador

Carpeta devuelta

En fecha Fecha

Figura 12.38: Interfaz Eliminar patente

Generacion de informes

Listado de patentes Informe estadístico

Por fecha de

Por letra

Listado de todas las patentes

Listar todas las patentes que comienzen con

Listar las patentes entre rangos de letras

Desde

Hasta

Informe en el periodo

Desde

Hasta

Comparar meses

Mes 1

Mes 2

Figura 12.39: Interfaz Generación de informes

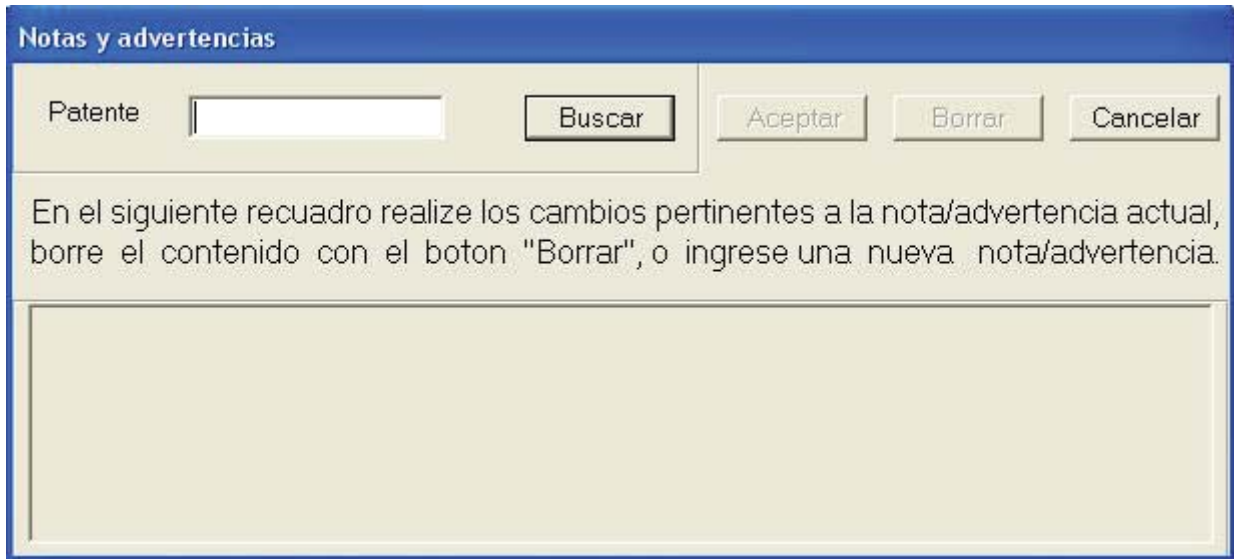


Figura 12.40: Interfaz Notas y advertencias

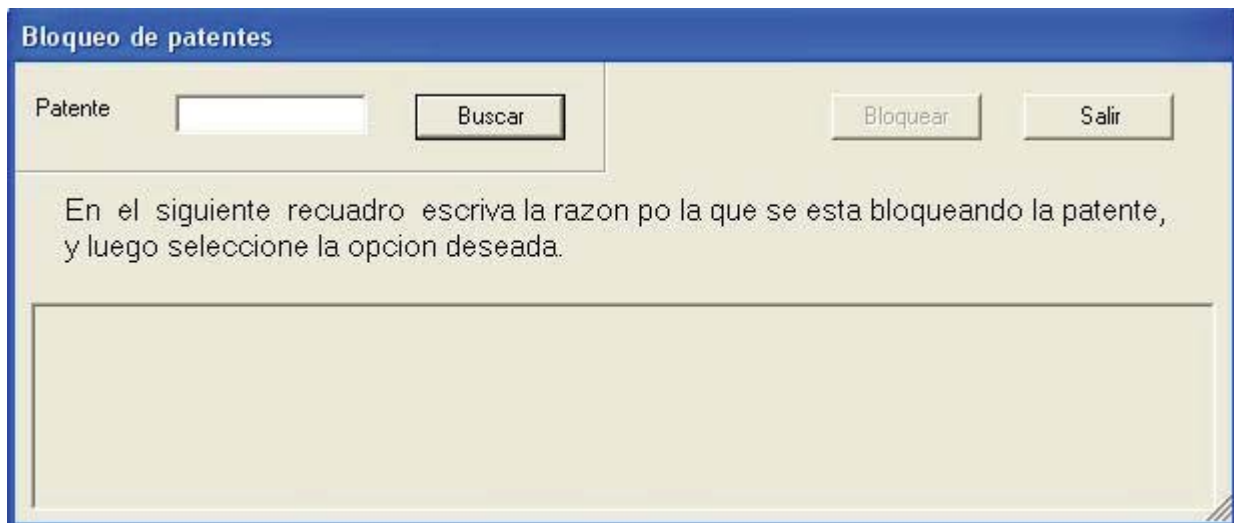


Figura 12.41: Interfaz Bloquear patente

Desbloqueo de patentes

Patente

En el siguiente recuadro escriba la razon po la que se esta bloqueando la patente, y luego seleccione la opcion deseada.

Figura 12.42: Desbloquear patente

Busqueda patentes

Por patente

Por letras

Otros criterios

Por tipo

Por estado Activo Pasiva

Ultimo año pagado

Por fecha de

Desde

Hasta

Figura 12.43: Interfaz Búsqueda patentes

12.7 Pruebas correspondientes a la primera iteración

En la presente sección del informe se pretende dar a conocer algunas de las pruebas realizadas en el modulo correspondiente a la primera iteración de la fase de construcción del proyecto (Modulo usuario registrado), relacionados con los siguientes casos de uso:

Inicio de sesión
Ingresar caja
Ingresar libro
Modificar patente
Eliminar patente
Búsqueda de patentes
Notas y advertencias
Bloquear patente
Desbloquear patente
Historial
Generación de informes

Casos de prueba “Inicio de sesión”

Tabla 12.11: Caso de prueba inicio de sesión.

Caso n°	1.
Descripción	Ingreso de un usuario valido a la interfaz inicio de sesión.
Entrada	Se ingresa como nombre de usuario “james” y como contraseña “12345” luego se presiona el botón “Aceptar”.

Salida	Inicio de sesión valido, se inicia la interfaz principal del sistema para el usuario.
Datos	Usuario “james”, contraseña “12345”.

Tabla 12.12: Caso de prueba inicio de sesión.

Caso n°	2.
Descripción	Ingreso de usuarios inválidos a la interfaz de inicio de sesión.
Entrada	Se ingresan usuarios incorrectos luego se presiona el botón “Aceptar”.
Salida	Inicio de sesión invalido, se recibe mensaje de error: “Error, nombre de usuario y/o contraseña incorrectos”.
Datos	Usuario “xxxxx”, contraseña “1235”. Usuario “Andrés”, contraseña “12345”.

Tabla 12.13: Caso de prueba inicio de sesión.

Caso n°	3.
Descripción	Ingreso de contraseñas invalidas a la interfaz de inicio de sesión.
Entrada	Se ingresan usuarios validos, pero las contraseñas erradas y se presiona el botón “Aceptar”.
Salida	Inicio de sesión invalido, se recibe mensaje de error: “Error, nombre de usuario y/o contraseña incorrectos”.

Datos	Usuario “james”, contraseña “xxxxxxx”. Usuario “james”, contraseña “nose”.
-------	---

Casos de prueba “Historial”

Tabla 12.14: Caso de prueba Historial.

Caso n°	4.
Descripción	En la interfaz principal del sistema se presiona el menú “Historial”, luego se ingresa la patente a buscar en el campo correspondiente y se realiza la búsqueda.
Entrada	Se ingresa una patente inválida en el campo correspondiente y se presiona el botón “Revisar”.
Salida	Se muestra un mensaje de error, “No existe la patente especificada en el registro”.
Datos	Patentes “XX5203”, “ADF236”, “RF5326”, “WE5698”, “AS5230”.

Tabla 12.15: Caso de prueba Historial.

Caso n°	5.
Descripción	En la interfaz principal del sistema se presiona el menú “Historial”, luego se ingresa la patente a buscar en el campo correspondiente y se realiza la búsqueda.
Entrada	Se ingresa una patente válida (la cual tiene instancias en el historial) en el campo correspondiente y se presiona el botón “Revisar”.
Salida	La interfaz despliega las instancias anteriores de la patente junto con sus datos y

	espera una acción del usuario.
Datos	Patentes “ZZ0010”, “ZZ0026”, “ZZ0050”, “ZZ0040”, “ZZ0015”.

Tabla 12.16: Caso de prueba Historial.

Caso n°	6.
Descripción	En la interfaz principal del sistema se presiona el menú “Historial”, luego se ingresa la patente a buscar en el campo correspondiente y se realiza la búsqueda.
Entrada	Se ingresa una patente válida (la cual no posee instancias en el historial) en el campo correspondiente y se presiona el botón “Revisar”.
Salida	Se muestra un mensaje de error, “No existe registro en el historial de la patente especificada”.
Datos	Patentes “AJM325”, “AA3625”, “CC5421”, “ZZ5020”, “AA0036”.

Casos de prueba “Ingresar caja”

Tabla 12.17: Caso de prueba ingreso caja.

Caso n°	7.
Descripción	Ingreso de una patente valida con sus datos a la interfaz de ingreso caja.
Entrada	En la interfaz principal del sistema se despliega el menú “Patentes”, luego se va al menú “ingresar” y por ultimo se presiona el menú “caja”, ingresando en la interfaz los valores correspondientes.

Salida	Luego de ingresar los datos la interfaz se prepara para recibir un nuevo ingreso borrando el campo patente.
Datos	Patente “ZZ0050”, Ultimo año pagado “2005”, activo “1”, pasivo “0”, tipo “automóvil”, digitador “james”, “02/02/2005”, observación “hola”, fecha actual “08/09/2005”.

Tabla 12.18: Caso de prueba ingreso caja.

Caso n°	8
Descripción	Ingreso de una patente invalida (menos de 6 caracteres) con sus datos a la interfaz de ingreso caja.
Entrada	En la interfaz principal del sistema se despliega el menú “Patentes”, luego se va al menú “ingresar” y por ultimo se presiona el menú “caja”, ingresando en la interfaz los valores correspondientes.
Salida	Se recibe mensaje de error, “La patente debe tener 6 dígitos”
Datos	Patente “ZZ”, Ultimo año pagado “2005”, activo “1”, pasivo “0”, tipo “automóvil”, digitador “james”, “02/02/2005”, observación “hola”, fecha actual “08/09/2005”. Patente “AA203”, Ultimo año pagado “2005”, activo “1”, pasivo “0”, tipo “automóvil”, digitador “james”, “02/02/2005”, observación “hola”, fecha actual “08/09/2005”.

Tabla 12.19: Caso de prueba ingreso caja.

--	--

Caso n°	9.
Descripción	Ingreso de una patente invalida (más de 6 caracteres) con sus datos a la interfaz de ingreso caja.
Entrada	En la interfaz principal del sistema se despliega el menú “Patentes”, luego se va al menú “ingresar” y por ultimo se presiona el menú “caja”, ingresando en la interfaz los valores correspondientes.
Salida	La interfaz no permite ingresar el séptimo digito al usuario.
Datos	Patente “AB02154”, Ultimo año pagado “2005”, activo “0”, pasivo “1”, tipo “moto”, digitador “james”, “02/02/2005”, observación “hola”, fecha actual “08/09/2005”. Patente “ZZZ0001”, Ultimo año pagado “2005”, activo “1”, pasivo “0”, tipo “acoplado”, digitador “james”, “02/02/2005”, observación “hola”, fecha actual “08/09/2005”.

Casos de prueba “Ingresar libro”

Tabla 12.20: Caso de prueba ingreso libro.

Caso n°	10.
Descripción	Ingreso de una patente nueva en la interfaz de ingreso libro.
Entrada	En la interfaz principal del sistema se despliega el menú “Patentes”, luego se va al menú “Ingresar” y por ultimo se presiona el menú “libro”, ingresando en la interfaz los valores correspondientes.
Salida	Se muestra un mensaje de error, “No existe la patente especificada en el registro”

Datos	<p>Patente “ZZ0050”, Sacado por “yo”, sacado fecha “02/02/2005”, activo “0”, pasivo “1”, tipo “moto”, digitador “james”, observación “hola”, fecha actual “08/09/2005”.</p> <p>Patente “AAA550”, Sacado por “yo”, sacado fecha “02/02/2005”, activo “1”, pasivo “0”, tipo “acoplado”, digitador “james”, observación “hola”, fecha actual “08/09/2005”.</p>
-------	---

Tabla 12.21: Caso de prueba ingreso libro.

Caso n°	11.
Descripción	Ingreso de una patente vacía en la interfaz de ingreso libro.
Entrada	En la interfaz principal del sistema se despliega el menú “Patentes”, luego se va al menú “ingresar” y por ultimo se presiona el menú “libro”, ingresando en la interfaz los valores correspondientes.
Salida	Se muestra un mensaje de error, “Faltan campos requeridos”
Datos	<p>Patente “”, Sacado por “yo”, sacado fecha “02/02/2005”, activo “0”, pasivo “1”, tipo “moto”, digitador “james”, observación “hola”, fecha actual “08/09/2005”.</p> <p>Patente “AAA550”, Sacado por “yo”, sacado fecha “02/02/2005”, activo “1”, pasivo “0”, tipo “acoplado”, digitador “james”, observación “hola”, fecha actual “08/09/2005”.</p>

Tabla 12.22: Caso de prueba ingreso libro.

--	--

Caso n°	12.
Descripción	Ingreso de una patente valida a la interfaz de ingreso libro
Entrada	En la interfaz principal del sistema se despliega el menú “Patentes”, luego se va al menú “ingresar” y por ultimo se presiona el menú “libro”, ingresando en la interfaz los valores correspondientes.
Salida	Luego de ingresar los datos la interfaz se prepara para recibir un nuevo ingreso borrando los campos correspondientes.
Datos	Patente “ZZ0050”, Sacado por “yo”, sacado fecha “02/02/2005”, activo “0”, pasivo “1”, tipo “moto”, digitador “james”, observación “hola”, fecha actual “08/09/2005”.

Casos de prueba “Modificar patente”

Tabla 12.23: Caso de prueba Modificar patente.

Caso n°	13.
Descripción	En la interfaz principal del sistema se despliega el menú “Patentes”, luego se va al menú “Modificar”, luego se ingresa la patente a buscar en el campo correspondiente y se realiza la búsqueda.
Entrada	Se ingresan patentes que no existen en la interfaz modificar datos, luego se presiona el botón “Buscar”.
Salida	Se muestra un mensaje de error, “No existe la patente especificada en el registro”.
Datos	Patentes “ABC001”, “ZAE002”, “AAA002”, “BS5023”, “CC5236”.

Tabla 12.24: Caso de prueba Modificar patente.

Caso n°	14.
Descripción	En la interfaz principal del sistema se despliega el menú “Patentes”, luego se va al menú “Modificar”, luego se ingresa la patente a buscar en el campo correspondiente, se realiza la búsqueda y se cambian los datos que se presentan (se realizan cambios aleatorios).
Entrada	Se ingresan patentes validas en la interfaz modificar datos, luego se presiona el botón buscar, para luego cambiar los datos que se despliegan y presionar el botón “Modificar”
Salida	Se presenta un mensaje, “Cambios realizados con éxito”
Datos	Patentes “ZZ0001”, “ZZ0002”, “ZZ0006”, “ZZ0050”, “ZZ0040”.

Tabla 12.25: Caso de prueba Modificar patente.

Caso n°	15.
Descripción	En la interfaz principal del sistema se despliega el menú “Patentes”, luego se va al menú “Modificar”, luego se ingresa la patente a buscar en el campo correspondiente, se realiza la búsqueda y se cambian los datos que se presentan (se realizan cambios aleatorios), intentando deshacer los cambios realizados.
Entrada	Se ingresan patentes validas en la interfaz modificar datos, luego se presiona el botón buscar, para luego cambiar los datos que se despliegan y presionar el botón “Deshacer”
Salida	La interfaz vuelve “TODOS” los datos de la patente buscada a su estado original y

	espera una acción por parte del usuario
Datos	Patentes “ZZ0009”, “ZZ0030”, “ZZ0070”, “ZZ0053”, “ZZ0038”.

Casos de prueba “Eliminar patente”

Tabla 12.26: Caso de prueba Eliminar datos.

Caso n°	16.
Descripción	En la interfaz principal del sistema se despliega el menú “Patentes”, luego se va al menú “Eliminar”, luego se ingresa la patente a buscar en el campo correspondiente y se realiza la búsqueda.
Entrada	Se ingresan patentes que no existen en la interfaz Eliminar datos, luego se presiona el botón “Buscar”.
Salida	Se muestra un mensaje de error, “No existe la patente especificada en el registro”
Datos	Patentes “AZ0009”, “XX0030”, “XXX070”, “TBV053”, “TY0038”.

Tabla 12.27: Caso de prueba Eliminar patente.

Caso n°	17.
Descripción	En la interfaz principal del sistema se despliega el menú “Patentes”, luego se va al menú “Eliminar”, luego se ingresa la patente a buscar en el campo correspondiente, se realiza la búsqueda y se presiona el botón “Eliminar”.
Entrada	Se ingresan patentes que no existen en la interfaz Eliminar datos, luego se presiona

	el botón “Buscar”, y luego de revisar los datos desplegados se presiona “Eliminar”.
Salida	Se presenta un mensaje solicitando confirmar la decisión (Si se presiona “Aceptar”), la interfaz elimina la patente y se cierra.
Datos	Patentes “ZZ0050”, “ZZ0051”, “ZZ0006”, “ZZ0071”, “ZZ0056”.

Tabla 12.28: Caso de prueba Eliminar patente.

Caso n°	18.
Descripción	En la interfaz principal del sistema se despliega el menú “Patentes”, luego se va al menú “Eliminar”, luego se ingresa la patente a buscar en el campo correspondiente, se realiza la búsqueda y se presiona el botón “Eliminar”.
Entrada	Se ingresan patentes que no existen en la interfaz Eliminar datos, luego se presiona el botón “Buscar”, y luego de revisar los datos desplegados se presiona “Eliminar” y se presenta una confirmación negativa
Salida	En este caso la interfaz limpia el campo patente y espera una nueva búsqueda.
Datos	Patentes “ZZ0001”, “ZZ0002”, “ZZ0003”, “ZZ004”, “ZZ0005”.

Casos de prueba “Generación de informes”

Tabla 12.29: Caso de prueba Generación de informes.

Caso n°	19.
Descripción	En la interfaz principal del sistema se despliega el menú “Archivo”, luego se va al menú “Generación de informes” y por ultimo se ingresa el tipo de informe y los

	datos correspondientes.
Entrada	Se selecciona el informe estadístico por intervalo de fechas, se ingresan ambas fechas (fechas en donde no se encuentren valores para realizar el informe) y se presiona el botón generar.
Salida	Se envía el mensaje de error correspondiente al caso y se sale de la interfaz
Datos	Intervalo 1 = "01/01/1900", "01/01/1930". Intervalo 2 = "01/01/2006", "01/01/2030"

Tabla 12.30: Caso de prueba Generación de informes.

Caso n°	20.
Descripción	En la interfaz principal del sistema se despliega el menú "Archivo", luego se va al menú "Generación de informes" y por ultimo se ingresa el tipo de informe y los datos correspondientes.
Entrada	Se selecciona el informe estadístico por comparación de meses, se ingresan ambos meses (uno de los cuales no posee información) y se presiona el botón generar.
Salida	Se envía el mensaje de error correspondiente al caso y se sale de la interfaz
Datos	Prueba 1 = "Marzo/1900", "Noviembre/2005". Prueba 2 = "Noviembre/2005", "noviembre/2006"

Tabla 12.31: Caso de prueba Generación de informes.

Caso n°	21.

Descripción	En la interfaz principal del sistema se despliega el menú “Archivo”, luego se va al menú “Generación de informes” y por ultimo se ingresa el tipo de informe y los datos correspondientes.
Entrada	Se selecciona el informe estadístico por comparación de meses, se ingresan ambos meses (meses en los que se posean datos suficientes para realizar el informe) y se presiona el botón generar.
Salida	Se presenta al usuario el informe generado y las opciones de navegación, impresión, de guardado, zoom, búsqueda y salir del informe.
Datos	Prueba 1 = “Octubre/2005”, “Noviembre/2005”.

Casos de prueba “Notas y advertencias”

Tabla 12.32: Caso de prueba Notas y advertencias.

Caso n°	22.
Descripción	En la interfaz principal del sistema se despliega el menú “Búsqueda”, luego se va al menú “Notas”, luego se ingresa la patente a buscar en el campo correspondiente, se realiza la búsqueda.
Entrada	Se ingresa una patente inexistente en el campo correspondiente y se presiona el botón “Buscar”.
Salida	Se muestra un mensaje de error, “No existe la patente especificada en el registro”.
Datos	Patentes “XX0052”, “XX0485”, “AC1569”, “ASD025”, “QWESDF”.

Tabla 12.33: Caso de prueba Notas y advertencias.

Caso n°	23.
Descripción	En la interfaz principal del sistema se despliega el menú “Búsqueda”, luego se va al menú “Notas”, luego se ingresa la patente a buscar en el campo correspondiente y se realiza la búsqueda, se cambian los datos en la nota y se presiona “Aceptar”.
Entrada	Se ingresa una patente (la cual no tiene asociada una nota) en el campo correspondiente y se presiona el botón “Buscar”, luego se ingresa un nueva nota y se presiona “Aceptar”
Salida	Se presenta el mensaje “Nota / Advertencia ingresada en el sistema” y luego se cierra la interfaz
Datos	Patentes “ZZ0020”, “ZZ0015”, “ZZ0040”, “ZZ0030”, “ZZ0010”.

Tabla 12.34: Caso de prueba Notas y advertencias.

Caso n°	24.
Descripción	En la interfaz principal del sistema se despliega el menú “Búsqueda”, Luego se va al menú “Notas”, luego se ingresa la patente a buscar en el campo correspondiente y se realiza la búsqueda, se cambian los datos en la nota y se presiona “Aceptar”.
Entrada	Se ingresa una patente (la cual ya posee una nota) en el campo correspondiente y se presiona el botón “Buscar”, luego se cambia la nota y se presiona “Aceptar”
Salida	Se presenta el mensaje “Nota / Advertencia ingresada en el sistema” y luego se cierra la interfaz
Datos	Patentes “ZZ0020”, “ZZ0015”, “ZZ0040”, “ZZ0030”, “ZZ0010”.

Casos de prueba “Bloquear patente”

Tabla 12.35: Caso de prueba Bloquear patente.

Caso n°	25.
Descripción	En la interfaz principal del sistema se despliega el menú “Bloqueo”, luego se va al menú “Bloquear”, luego se ingresa la patente a buscar en el campo correspondiente, se realiza la búsqueda.
Entrada	Se ingresa una patente inexistente en el campo correspondiente y se presiona el botón “Buscar”.
Salida	Se muestra un mensaje de error, “No existe la patente especificada en el registro”.
Datos	Patentes “YY0623”, “AY5213”, “RE1563”, “RF0258”, “TRG582”.

Tabla 12.36: Caso de prueba Bloquear patente.

Caso n°	26.
Descripción	En la interfaz principal del sistema se despliega el menú “Bloqueo”, luego se va al menú “Bloquear”, luego se ingresa la patente a buscar en el campo correspondiente, se realiza la búsqueda, se ingresa la razón del bloqueo y se presiona “Bloquear”.
Entrada	Se ingresa una patente (la cual no esta bloqueada) en el campo correspondiente y se presiona el botón “Buscar”, luego se ingresa la razón por la que se bloqueo y se presiona “Bloquear”.
Salida	Se presenta el mensaje “Patente “xx” bloqueada” y luego se cierra la interfaz

Datos	Patentes “ZZ0010” razón “hola”, “ZZ0026” razón “no se”, “ZZ0050” razón “bueno”, “ZZ0040” razón “se me ocurrió”, “ZZ0015” razón “zzzzzz”.
-------	--

Tabla 12.37: Caso de prueba Bloquear patente.

Caso n°	27.
Descripción	En la interfaz principal del sistema se despliega el menú “Bloqueo”, luego se va al menú “Bloquear”, luego se ingresa la patente a buscar en el campo correspondiente y se realiza la búsqueda.
Entrada	Se ingresa una patente (la cual ya esta bloqueada) en el campo correspondiente y se presiona el botón “Buscar”.
Salida	Se presenta el mensaje “La patente ya se encuentra bloqueada”, la interfaz muestra la razón por la que se bloqueo la patente y espera una acción del usuario (ya sea salir o realizar nueva búsqueda).
Datos	Patentes “ZZ0010”, “ZZ0026”, “ZZ0050”, “ZZ0040”, “ZZ0015”.

Casos de prueba “Desbloquear patente”

Tabla 12.38: Caso de prueba Desbloquear patente.

Caso n°	28.
Descripción	En la interfaz principal del sistema se despliega el menú “Bloqueo”, luego se va al menú “Desbloquear”, luego se ingresa la patente a buscar en el campo correspondiente, se realiza la búsqueda.

Entrada	Se ingresa una patente inexistente en el campo correspondiente y se presiona el botón “Buscar”.
Salida	Se muestra un mensaje de error, “No existe la patente especificada en el registro”.
Datos	Patentes “AA0050”, “BB0030”, “CC0050”, “CB0063”, “FD4523”.

Tabla 12.39: Caso de prueba Desbloquear patente.

Caso n°	29.
Descripción	En la interfaz principal del sistema se despliega el menú “Bloqueo”, luego se va al menú “Desbloquear”, luego se ingresa la patente a buscar en el campo correspondiente, se realiza la búsqueda, y se desbloquea la patente.
Entrada	Se ingresa una patente (la cual se encuentra bloqueada) en el campo correspondiente y se presiona el botón “Buscar” y se presiona “Desbloquear”.
Salida	Se presenta el mensaje “Patente “xx” desbloqueada” y luego se cierra la interfaz
Datos	Patentes “ZZ0010”, “ZZ0026”, “ZZ0050”, “ZZ0040”, “ZZ0015”.

Tabla 12.40: Caso de prueba Desbloquear patente.

Caso n°	30.
Descripción	En la interfaz principal del sistema se despliega el menú “Bloqueo”, luego se va al menú “Desbloquear”, luego se ingresa la patente a buscar en el campo correspondiente y se realiza la búsqueda.

Entrada	Se ingresa una patente (la cual no se encuentra bloqueada) en el campo correspondiente y se presiona el botón “Buscar”.
Salida	Se presenta el mensaje “La patente no se encuentra bloqueada actualmente”, la interfaz limpia el campo patente y espera un nuevo ingreso.
Datos	Patentes “ZZ0010”, “ZZ0026”, “ZZ0050”, “ZZ0040”, “ZZ0015”.

Casos de prueba “Búsqueda de patentes”

Tabla 12.41: Caso de prueba Búsqueda de patentes.

Caso n°	31.
Descripción	En la interfaz principal del sistema se despliega el menú “Búsqueda”, luego se va al menú “Buscar patentes”, luego se seleccionan los criterios y se realiza la búsqueda.
Entrada	Se selecciona el primer criterio de búsqueda (“Por patente”), luego se ingresa una patente valida e inexistente.
Salida	Se muestra un mensaje al usuario señalándole que no se encontraron patentes que coincidan con los criterios de búsqueda seleccionados y se muestra la pantalla principal del sistema sin objetos desplegados.
Datos	Patentes “AZ0009”, “XX0030”, “XXX070”, “TBV053”, “TY0038”.

Tabla 12.42: Caso de prueba Búsqueda de patentes.

Caso n°	32.

Descripción	En la interfaz principal del sistema se despliega el menú “Búsqueda”, luego se va al menú “Buscar patentes”, luego se seleccionan los criterios y se realiza la búsqueda.
Entrada	Se selecciona el segundo criterio de búsqueda (“Por letra”), luego se ingresan las iniciales de una patente.
Salida	Se despliegan los datos obtenidos en la interfaz principal del sistema. En el caso de no obtener datos se muestra un mensaje al usuario señalándole que no se encontraron patentes que coincidan con los criterios de búsqueda seleccionados y se muestra la pantalla principal del sistema sin objetos desplegados.
Datos	Patentes “CA”, “A”, “AA”, “AB”, “Z”, “D”, “JA”, “XD”, “PD”.

Tabla 12.43: Caso de prueba Búsqueda de patentes.

Caso n°	33.
Descripción	En la interfaz principal del sistema se despliega el menú “Búsqueda”, luego se va al menú “Buscar patentes”, luego se seleccionan los criterios y se realiza la búsqueda.
Entrada	Se selecciona el tercer criterio de búsqueda (“Otros criterios”), luego se selecciona uno o más sub. criterios y se ingresan los datos correspondientes
Salida	Se despliegan los datos obtenidos en la interfaz principal del sistema. En el caso de no obtener datos se muestra un mensaje al usuario señalándole que no se encontraron patentes que coincidan con los criterios de búsqueda seleccionados y se muestra la pantalla principal del sistema sin objetos desplegados.
Datos	Ultimo año pagado = 2005, pasivo, Tipo = automóvil, fechas “31/12/2004” y “31/11/2005”

13. SEGUNDA ITERACION: USUARIO NO REGISTRADO

13.1 Diagramas de caso de uso extendido

A continuación se presentan los casos de uso correspondientes al “Usuario no registrado”.

Tabla 13.1: Caso de uso extendido Generación de informes

Caso de uso	Generación de informes
Actor Primario	Usuario no registrado
Participantes e intereses	Usuario: Poder obtener un informe según la elección que se elija.
Propósito	La generación de ciertos informes según la necesidad del usuario.
Precondiciones	El usuario debe haber ejecutado con anterioridad el caso de uso “Inicio de sesión” y haber ingresado como invitado.
Poscondiciones	No tiene
Escenario Principal	<ol style="list-style-type: none"> 1. El usuario selecciona “Generación de informes” en el sistema. 2. El sistema le da a elegir entre los informes “Por fecha” o “Por letra” 3. El usuario selecciona la opción y elige el informe e ingresa

	<p>los datos requeridos.</p> <p>4. Se despliega en pantalla el informe y el usuario selecciona entre las opciones “Imprimir”, “Exportar” o “Cancelar”</p> <p>5. El usuario finaliza el caso de uso.</p>
Extensiones	<p>2.a. La opción “Por fecha”, es un informe de las patentes desde la fecha elegida hasta la actual.</p> <p>2.b. La opción “Por letra” le da acceso a los siguientes informes: “Listado de todas las patentes”, “Listado de las patentes que comienzan con una letra” y “Listado de patentes entre rangos de letras”.</p> <p>4.a. Si la actividad seleccionada es “Imprimir”, el sistema imprimirá el informe en la impresora del computador.</p> <p>4.b. Si la actividad seleccionada es “Exportar”, el sistema le dará la opción de guardar el informe en cualquiera de las carpetas del computador, en distintos formatos.</p> <p>4.c. Si la actividad seleccionada es “Cancelar”, el sistema cerrará la ventana “Generación de informes” volviendo a la pantalla principal.</p>
Requerimientos especiales	No tiene
Frecuencia de ocurrencia	Baja

Tabla 13.2: Caso de uso extendido Búsqueda patentes

Caso de uso	Búsqueda patentes.
Actor Primario	Usuario no registrado.
Participantes e intereses	Usuario: El usuario puede buscar una patente o un grupo de estas.
Propósito	La búsqueda de una patente y sus respectivos datos o también de un grupo de estas.
Precondiciones	El usuario debe haber ejecutado con anterioridad el caso de uso “Inicio de sesión” y haber ingresado como invitado.
Poscondiciones	No tiene
Escenario Principal	<ol style="list-style-type: none"> 1. El usuario selecciona “Búsqueda patentes” en el sistema. 2. El sistema despliega la pantalla con las opciones de búsqueda. 3. El usuario selecciona las opciones e ingresa los datos correspondientes. 4. El usuario selecciona entre las actividades “Buscar” o “Cancelar”. 5. Se le presentan por pantalla al usuario los resultados de la búsqueda.
Extensiones	<p>4.a. Si la actividad seleccionada es “Buscar”, se realizara la búsqueda y desplegara los datos en la pantalla principal. En caso de no encontrarse datos se le indicara la usuario</p> <p>4.c. Si la actividad seleccionada es “Cancelar”, el sistema</p>

	cerrara la ventana “Búsqueda patentes” volviendo a la pantalla principal.
Requerimientos especiales	No tiene.
Frecuencia de ocurrencia	Alta.

Tabla 13.3: Caso de uso extendido Notas y advertencias

Caso de uso	Notas y advertencias.
Actor Primario	Usuario no registrado.
Participantes e intereses	Usuario: El usuario puede leer una nota o advertencia sobre la patente elegida.
Propósito	Leer una nota o advertencia sobre una patente seleccionada.
Precondiciones	El usuario debe haber ejecutado con anterioridad el caso de uso “Inicio de sesión” y haber ingresado como invitado.
Poscondiciones	Los cambios realizados deben guardarse en la base de datos.
Escenario Principal	<ol style="list-style-type: none"> 1. El usuario selecciona “Notas y advertencias” en el sistema. 2. El sistema solicita el ingreso de la patente. 3. El usuario ingresa la patente. 5. El usuario selecciona la actividad “Cancelar”. 6. El usuario finaliza el caso de uso.

Extensiones	<p>3.a. Si la patente ingresada no es encontrada, se le pedirá un nuevo ingreso al usuario.</p> <p>3.b. La nota se le presentara al usuario en el cuadro inferior, en caso de presentar nota la patente, el cuadro no sufrirá cambios.</p> <p>5.c El sistema cerrara la ventana “Notas y advertencias” volviendo a la pantalla principal.</p>
Requerimientos especiales	No tiene.
Frecuencia de ocurrencia	Ocasional.

13.2 Diagramas de secuencia

A continuación se presentan los diagramas de secuencia correspondientes al caso de uso “Usuario no registrado”, el cual muestra la secuencia de pasos que seguirá al momento de ejecutarse.

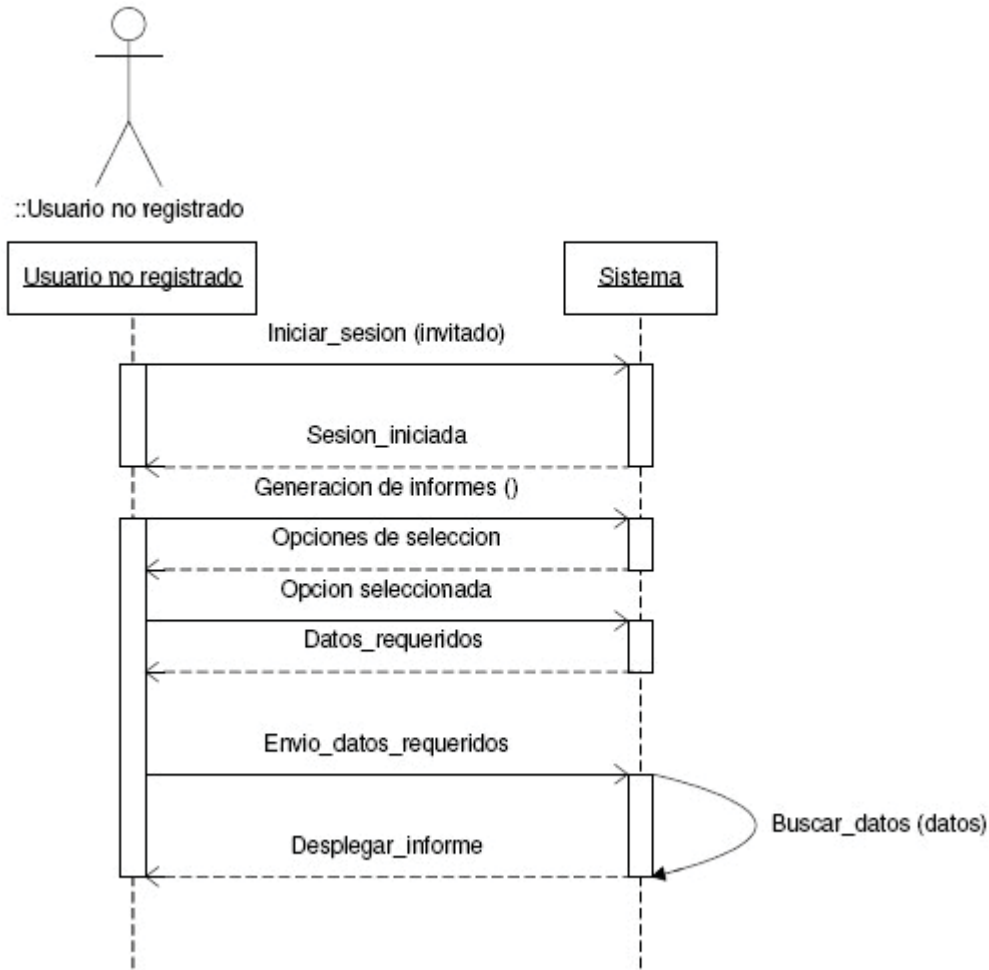


Figura 13.1: Diagrama de secuencia Generación de informes

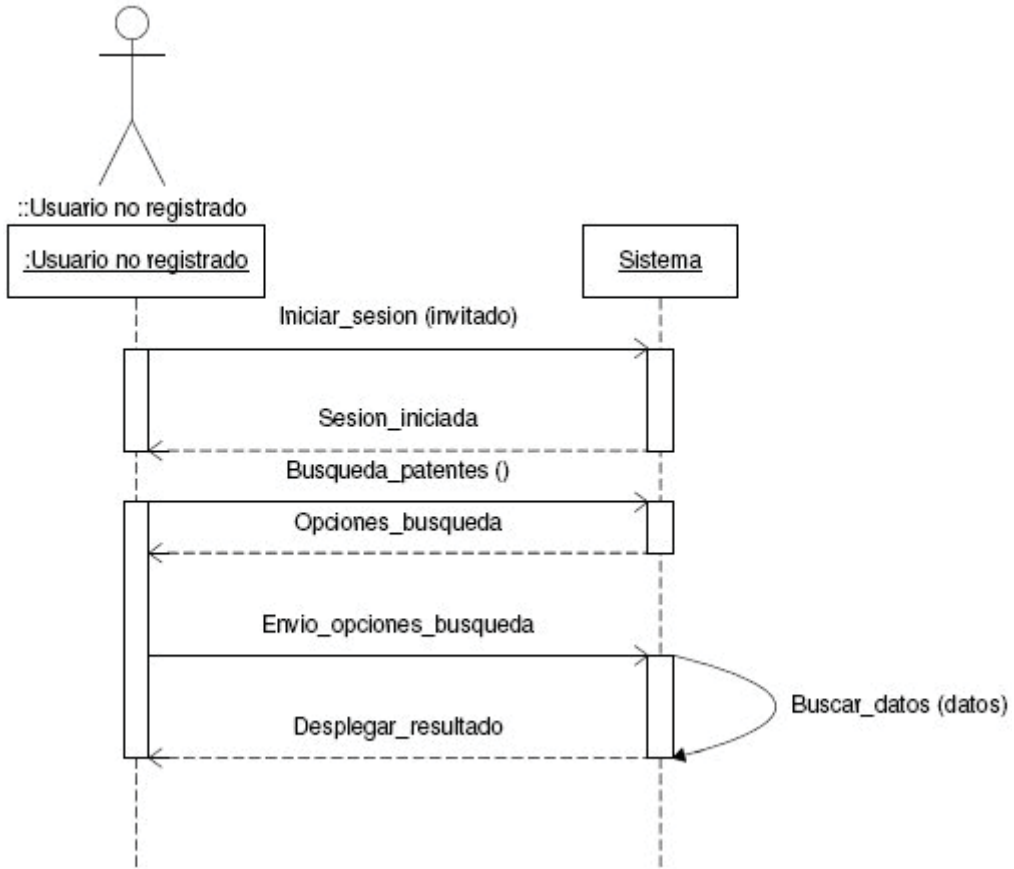


Figura 13.2: Diagrama de secuencia Búsqueda patentes

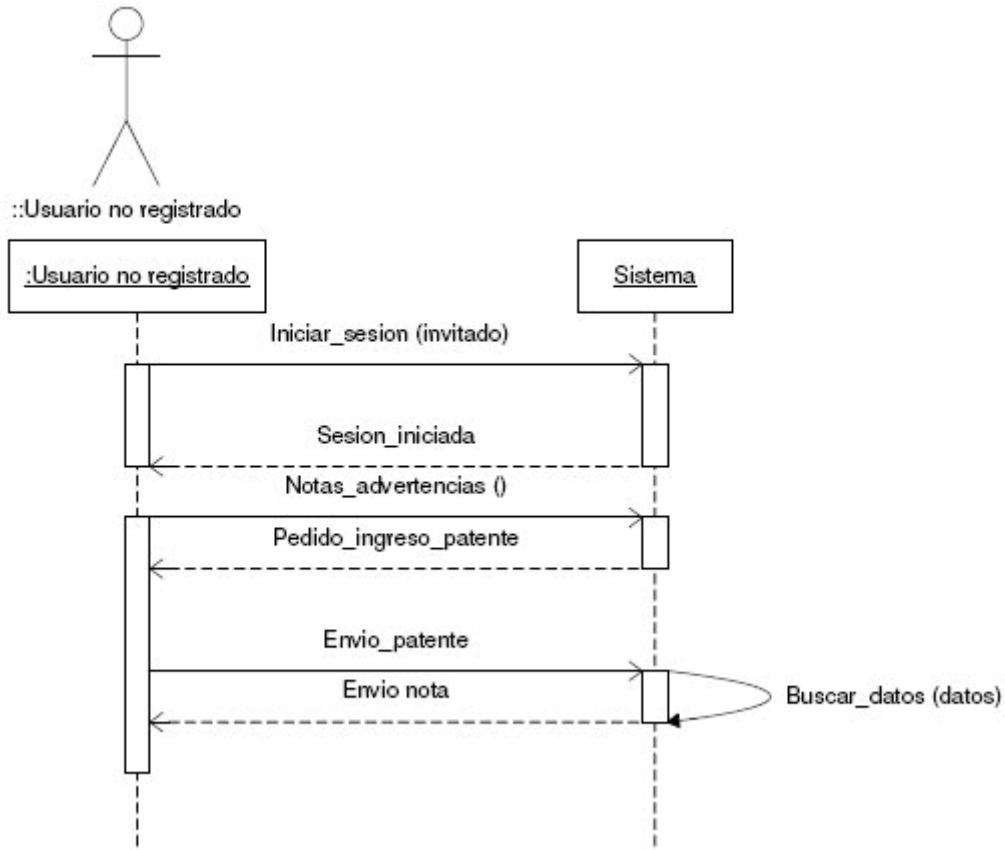


Figura 13.3: Diagrama de secuencia Notas y advertencias

13.3 Diagramas de actividad

A continuación se presentan los diagramas de actividad correspondientes al caso de uso “Usuario registrado”

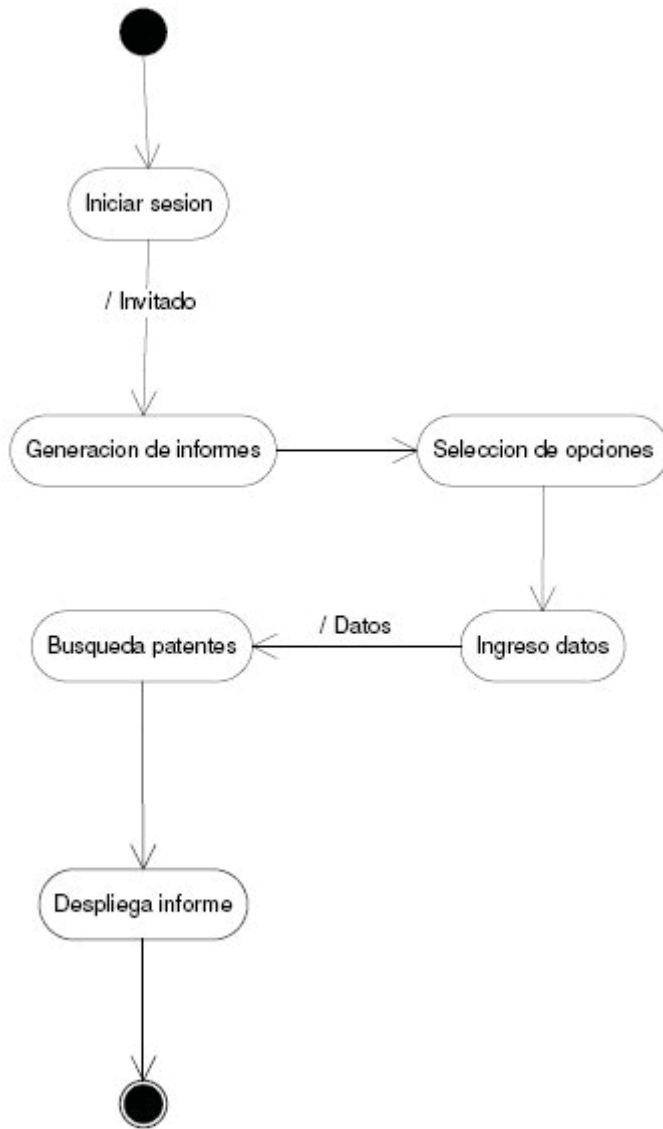


Tabla 13.4: Diagrama de actividad Generación de informes

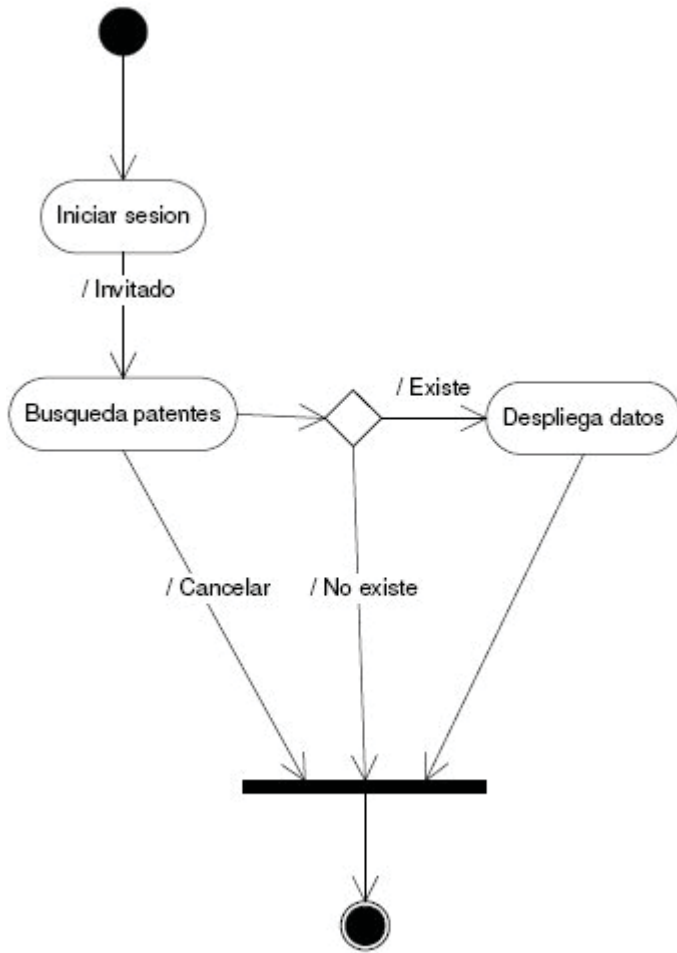


Figura 13.5: Diagrama de actividad Búsqueda patentes

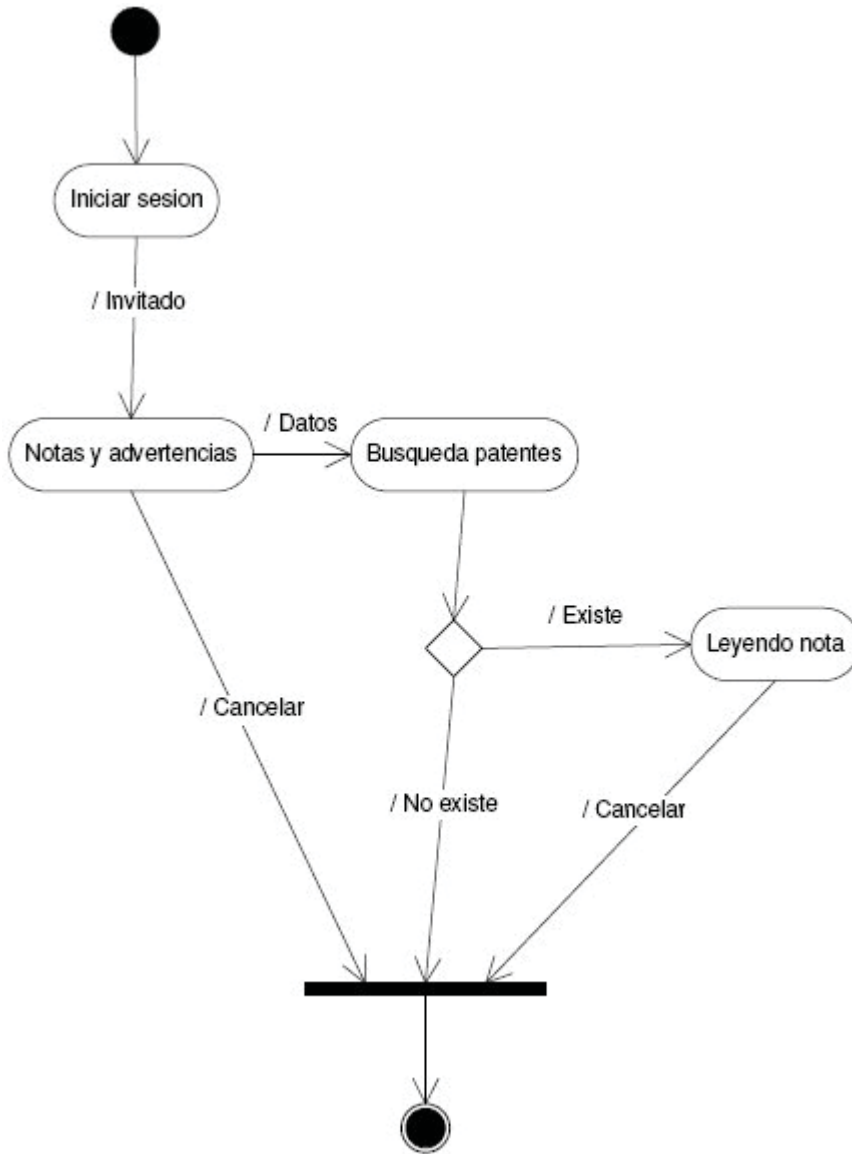


Figura 13.6: Diagrama de actividad Notas y advertencias

13.4 Diagramas de componentes

A continuación se presentan los diagramas de componentes correspondientes al caso de uso “Usuario registrado”

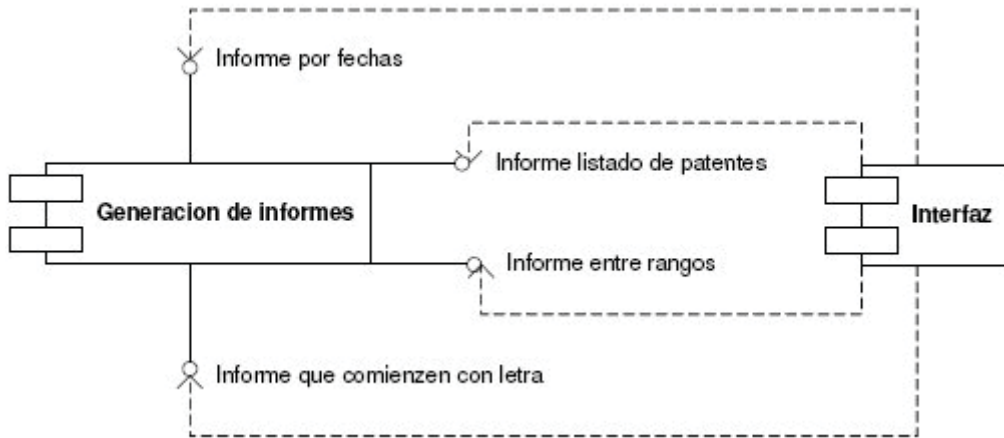


Figura 13.7: Diagrama de componente Generación de informes

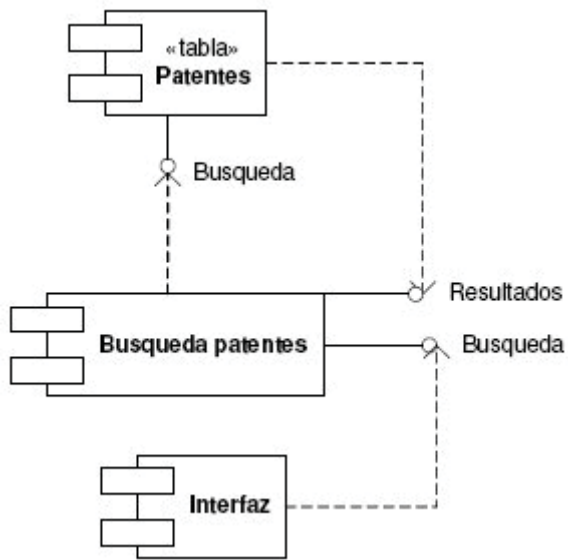


Figura 13.8: Diagrama de componente Búsqueda patentes

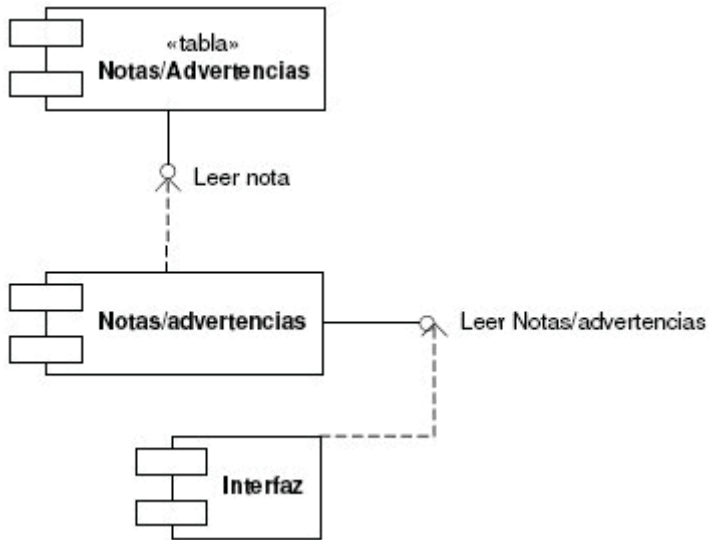


Figura 13.9: Diagrama de componente Notas y advertencias

13.5 Interfaces

A continuación se presentan las interfaces correspondientes al caso de uso “Usuario registrado”

Generación de informes

Por fecha de

Por letra

Listado de todas las patentes

Listar todas las patentes que comiencen con

Listar las patentes entre rangos de letras

Desde

Hasta

Figura 13.10: Interfaz Generación de informes

Busqueda patentes

Por patente

Por letras

Otros criterios

Por tipo

Por estado Activo Pasivo

Ultimo año pagado

Por fecha de

Desde

Hasta

Figura 13.11: Interfaz Búsqueda patentes

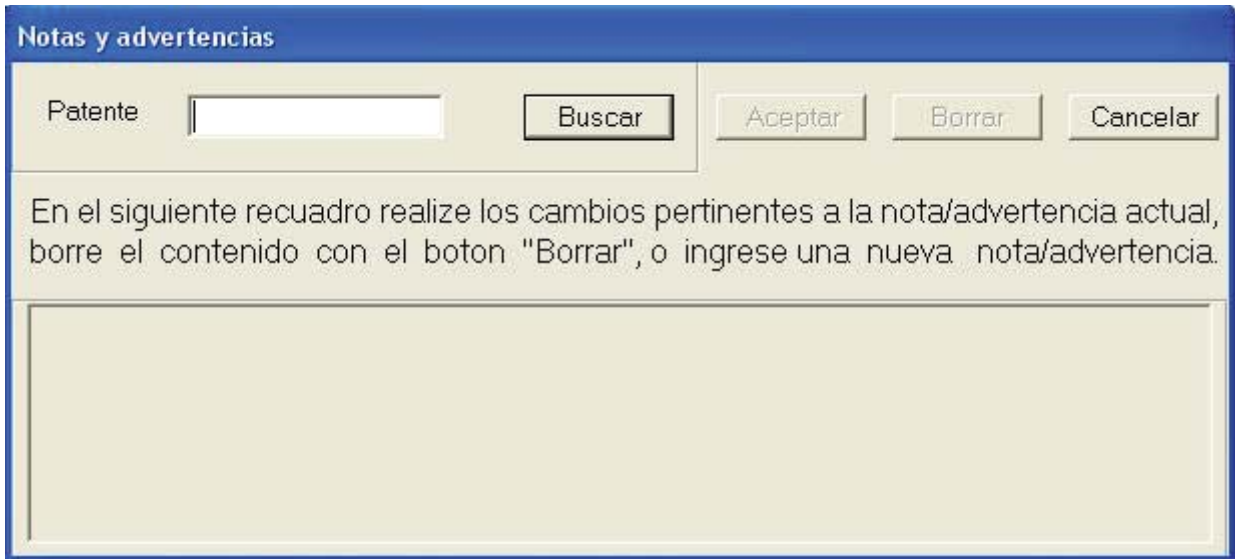


Figura 13.12: Interfaz Notas y advertencias

13.6 Pruebas correspondientes a la segunda iteración

En la presente sección del informe se pretende dar a conocer algunas de las pruebas realizadas en el modulo correspondiente a la segunda iteración de la fase de construcción del proyecto (Modulo usuario no registrado o invitado), relacionados con los siguientes casos de uso:

Inicio de sesión
Búsqueda de patentes
Notas y advertencias
Generación de informes

Casos de prueba “Inicio de sesión”

Tabla 13.4: Caso de prueba inicio de sesión.

Caso n°	1.
Descripción	Ingreso como usuario invitado.
Entrada	Se selecciona la opción “Ingresar como invitado” y se presiona el botón aceptar
Salida	Inicio de sesión valido, se inicia la interfaz principal del sistema para el usuario
Datos	Usuario = “invitado”

Casos de prueba “Generación de informes”

Tabla 13.5: caso de prueba Generación de informes.

Caso n°	2
Descripción	En la interfaz principal del sistema se despliega el menú “Archivo”, luego se va al menú “Generación de informes” y por ultimo se ingresa el tipo de informe y los datos correspondientes.
Entrada	Se selecciona el informe por fecha, se ingresan los datos correspondientes (fechas en donde no se encuentren valores para realizar el informe) y se presiona el botón generar.
Salida	Se envía el mensaje de error correspondiente al caso y se sale de la interfaz
Datos	Fecha 1 =”01/01/1900”, tipo = Digitación. Fecha 2 = “01/01/2006”, tipo = Devolución

Tabla 13.6: caso de prueba Generación de informes.

Caso n°	3.
Descripción	En la interfaz principal del sistema se despliega el menú “Archivo”, luego se va al menú “Generación de informes” y por ultimo se ingresa el tipo de informe y los datos correspondientes.
Entrada	Se selecciona el informe por letra, se selecciona “Listado de todas las patentes” y se presiona el botón generar.
Salida	Se presenta al usuario el informe generado y las opciones de navegación, impresión, de guardado, zoom, búsqueda y salir del informe.
Datos	Prueba 1 = “Marzo/1900”, “Noviembre/2005”. Prueba 2 = “Noviembre/2005”, “noviembre/2006”

Tabla 13.7: caso de prueba Generación de informes.

Caso n°	4.
Descripción	En la interfaz principal del sistema se despliega el menú “Archivo”, luego se va al menú “Generación de informes” y por ultimo se ingresa el tipo de informe y los datos correspondientes.
Entrada	Se selecciona el informe por letra, se selecciona “Listado de las patentes que comiencen”, se llena el campo pedido y se presiona el botón generar.
Salida	Se presenta al usuario el informe generado y las opciones de navegación, impresión, de guardado, zoom, búsqueda y salir del informe. En el caso de no existir datos relacionados se le presenta al usuario un mensaje y se sale de la interfaz.

Datos	Comienzan con A, Comienzan con C, Comienzan con Z, Comienzan con Q, Comienzan con S.
-------	--

Casos de prueba “Búsqueda de patentes”

Tabla 13.8: caso de prueba Búsqueda de patentes.

Caso n°	5.
Descripción	En la interfaz principal del sistema se despliega el menú “Búsqueda”, luego se va al menú “Buscar patentes”, luego se seleccionan los criterios y se realiza la búsqueda.
Entrada	Se selecciona el primer criterio de búsqueda (“Por patente”), luego se ingresa una patente valida y existente.
Salida	Se muestran los datos de la patente buscada en la interfaz principal.
Datos	Patentes “AA0001”, “ZZ0001”, “ZA0258”, “ZR2365”, “HI2365”.

Tabla 13.9: caso de prueba Búsqueda de patentes.

Caso n°	6.
Descripción	En la interfaz principal del sistema se despliega el menú “Búsqueda”, luego se va al menú “Buscar patentes”, luego se seleccionan los criterios y se realiza la búsqueda.
Entrada	Se selecciona el segundo criterio de búsqueda (“Por letra”), luego se ingresan las iniciales de una patente (iniciales que no contengan datos relacionados en la base

	de datos).
Salida	Se muestra un mensaje al usuario señalándole que no se encontraron patentes que coincidan con los criterios de búsqueda seleccionados y se muestra la pantalla principal del sistema sin objetos desplegados.
Datos	Patentes “SS”, “SX”, “QA”, “ER”, “SD”, “FG”, “QQ”, “WW”, “ZZZ”.

Tabla 13.10: caso de prueba Búsqueda de patentes.

Caso n°	7.
Descripción	En la interfaz principal del sistema se despliega el menú “Búsqueda”, luego se va al menú “Buscar patentes”, luego se seleccionan los criterios y se realiza la búsqueda.
Entrada	Se selecciona el tercer criterio de búsqueda (“Otros criterios”), luego se selecciona uno o más sub. criterios y se ingresan los datos correspondientes
Salida	Se despliegan los datos obtenidos en la interfaz principal del sistema. En el caso de no obtener datos se muestra un mensaje al usuario señalándole que no se encontraron patentes que coincidan con los criterios de búsqueda seleccionados y se muestra la pantalla principal del sistema sin objetos desplegados.
Datos	Archivo = activo, Tipo = automóvil, fechas “31/12/2004” y “31/11/2005”

Casos de prueba “Notas y advertencias”

Tabla 13.11: caso de prueba Notas y advertencias.

--	--

Caso n°	8.
Descripción	En la interfaz principal del sistema se despliega el menú “Búsqueda”, luego se va al menú “Notas”, luego se ingresa la patente a buscar en el campo correspondiente, se realiza la búsqueda.
Entrada	Se ingresa una patente inexistente en el campo correspondiente y se presiona el botón “Buscar”.
Salida	Se muestra un mensaje de error, “No existe la patente especificada en el registro”.
Datos	Patentes “XX0052”, “XX0485”, “AC1569”, “ASD025”, “QWESDF”.

Tabla 13.12: caso de prueba Notas y advertencias.

Caso n°	9.
Descripción	En la interfaz principal del sistema se despliega el menú “Búsqueda”, luego se va al menú “Notas”, luego se ingresa la patente a buscar en el campo correspondiente, se realiza la búsqueda, se realizan cambios en la nota y se presiona “Aceptar”.
Entrada	Se ingresa una patente (la cual no tiene asociada una nota) en el campo correspondiente y se presiona el botón “Buscar”, luego se intenta ingresar una nueva nota.
Salida	No se le permite ingresar una nueva nota (el campo es solo lectura para el invitado)
Datos	Patentes “ZZ0030”, “ZZ0035”, “ZZ0240”, “ZZ0430”, “ZZ0870”.

Tabla 13.13: caso de prueba Notas y advertencias.

--	--

Caso n°	10.
Descripción	En la interfaz principal del sistema se despliega el menú “Búsqueda”, luego se va al menú “Notas”, luego se ingresa la patente a buscar en el campo correspondiente, se realiza la búsqueda, se realizan cambios en la nota y se presiona “Aceptar”.
Entrada	Se ingresa una patente (la cual ya posee una nota) en el campo correspondiente y se presiona el botón “Buscar”, luego se intenta cambia la nota.
Salida	No se le permite cambiar la nota (el campo es solo lectura para el invitado)
Datos	Patentes “AA0020”, “XZ0015”, “AZ0040”, “ZX0030”, “ZA0010”.

14. TERCERA ITERACION: ADMINISTRADOR

14.1 Diagramas de caso de uso extendido

A continuación se presentan los casos de uso extendidos correspondientes al caso de uso “Administrador”

Tabla 14.1: Caso de uso extendido Ingresar usuario.

Caso de uso	Ingresar usuario.
Actor Primario	Administrador.
Participantes e intereses	Administrador: Crear usuario para el ingreso al sistema
Propósito	Crear usuarios y darles los permisos necesarios para que estos puedan acceder a las opciones del sistema
Precondiciones	El usuario debe haber ingresado al sistema como administrador
Poscondiciones	Los usuarios deben quedar almacenados en la base de datos
Escenario Principal	<ol style="list-style-type: none"> 1. El administrador selecciona “Ingresar usuario” en el sistema. 2. El sistema despliega una pantalla y solicita el ingreso de los datos. 3. El administrador ingresa los datos correspondientes al nuevo usuario.

	<p>4. El administrador selecciona entre las opciones “Crear” o “Cancelar”.</p> <p>5. El usuario finaliza el caso de uso.</p>
Extensiones	<p>4.a. Si la actividad seleccionada es “Crear”, el sistema guardara los datos, retornando a la pantalla principal.</p> <p>4.b. Si la actividad seleccionada es “Cancelar”, el sistema cerrara la ventana “Crear usuario” volviendo a la pantalla principal.</p>
Requerimientos especiales	No puede haber nombres de usuarios repetidos.
Frecuencia de ocurrencia	En un comienzo la frecuencia de este caso será alta, debido a la no existencia de usuarios en un primer momento. Luego de la creación de varios usuarios, la frecuencia de este caso pasara a ser baja.

Tabla 14.2: Caso de uso extendido Modificar usuario

Caso de uso	Modificar usuario.
Actor Primario	Administrador.
Participantes e intereses	Administrador: Crear usuario para el ingreso al sistema
Propósito	Modificar los datos de un usuario cuando se requiera
Precondiciones	El usuario debe haber ingresado al sistema como

	administrador
Poscondiciones	Los cambios deben quedar almacenados en la base de datos
Escenario Principal	<ol style="list-style-type: none"> 1. El administrador selecciona “Modificar usuario” en el sistema. 2. El sistema presenta los usuarios a modificar. 3. El administrador elige el usuario a modificar. 4. Se le presentan todos los datos del usuario para que este realice los cambios necesarios. 5. El administrador elige entre las opciones: “Modificar”, “Deshacer” o “Cancelar”. 6. El administrador finaliza el caso de uso.
Extensiones	<ol style="list-style-type: none"> 3.a. En caso de que el administrador se elija a si mismo, este podrá modificar todos los datos exceptuando el nombre de usuario y los permisos. 4.a. Los cambios serán realizados borrando el texto sobre el campo del cual se requiera y escribiendo un nuevo texto sobre este. 5.a. Si la actividad seleccionada es “Modificar”, la información del usuario será actualizada en la base de datos, volviendo a la pantalla principal. 5.b. Si la actividad seleccionada es “Deshacer” el sistema borrara los cambios realizados hasta el momento rellenando los campos que han sido cambiados con los originales. 5.c Si la actividad seleccionada es “Cancelar”, el sistema

	cerrara la ventana “Modificar usuario” volviendo a la pantalla principal.
Requerimientos especiales	No tiene.
Frecuencia de ocurrencia	Baja

Tabla 14.3: Caso de uso extendido Eliminar usuario

Caso de uso	Eliminar usuario.
Actor Primario	Administrador.
Participantes e intereses	Administrador: Eliminar un usuario del sistema.
Propósito	Eliminar un usuario del sistema para no permitirle el acceso a las opciones del sistema.
Precondiciones	El usuario debe haber ingresado al sistema como administrador.
Poscondiciones	El usuario debe ser eliminado de la base datos.
Escenario Principal	<ol style="list-style-type: none"> 1. El administrador selecciona “Eliminar usuarios” en el sistema. 2. El sistema presenta los posibles usuarios a eliminar. 3. El administrador elige el usuario a eliminar. 4. El administrador selecciona entre las opciones “Eliminar” o

	<p>“Cancelar”.</p> <p>5. El administrador finaliza el caso de uso.</p>
Extensiones	<p>3.a. En el caso de que el administrador se escoja a si mismo para eliminación, se envía una notificación y se espera un nuevo ingreso, ya que un usuario no puede eliminarse a si mismo.</p> <p>4.a. Si la actividad seleccionada es “Eliminar”, el sistema muestra los datos del usuario y pide confirmación.</p> <p>1. De ser positiva la confirmación, el usuario es eliminado y se va al paso numero 5.</p> <p>2. De ser negativa la confirmación, se va directamente al paso numero 5.</p> <p>4.b. Si la actividad seleccionada es “Cancelar”, el sistema cerrara la ventana “Eliminar usuario” volviendo a la pantalla principal.</p>
Requerimientos especiales	No tiene.
Frecuencia de ocurrencia	Baja.

14.2 Diagramas de secuencia

A continuación se presentan los diagramas de secuencia correspondientes al caso de uso “Administrador”

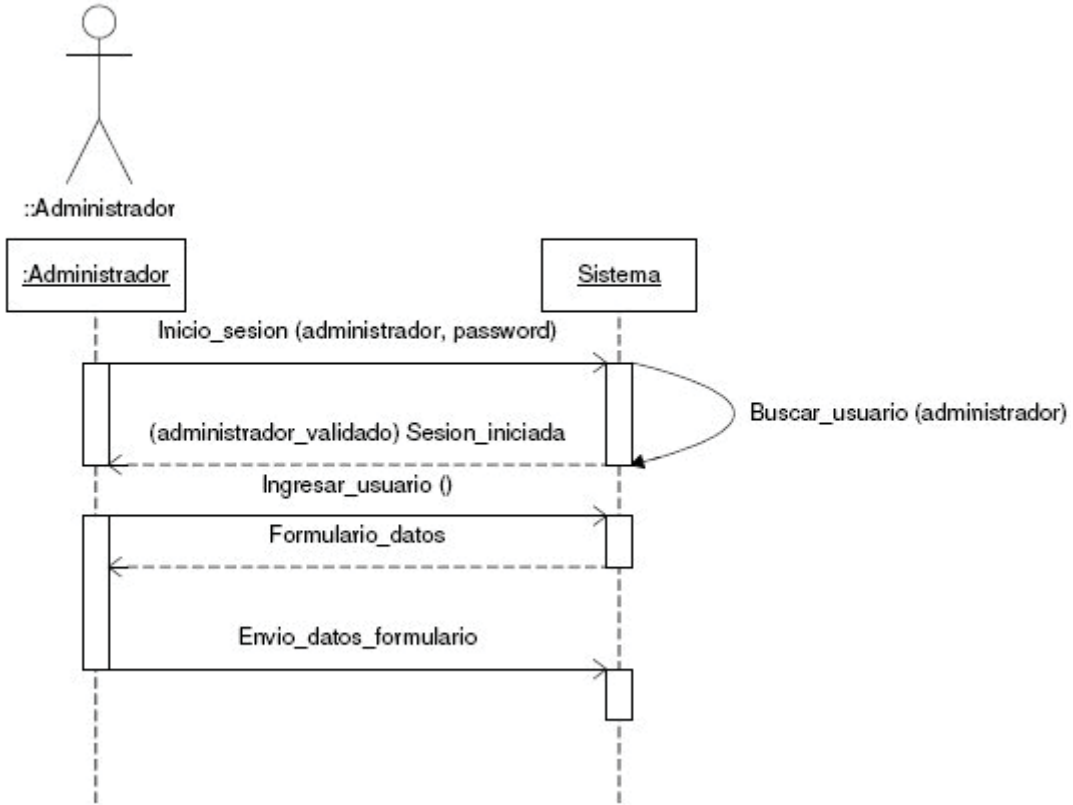


Figura 14.1: Diagrama de secuencia Ingresar usuario

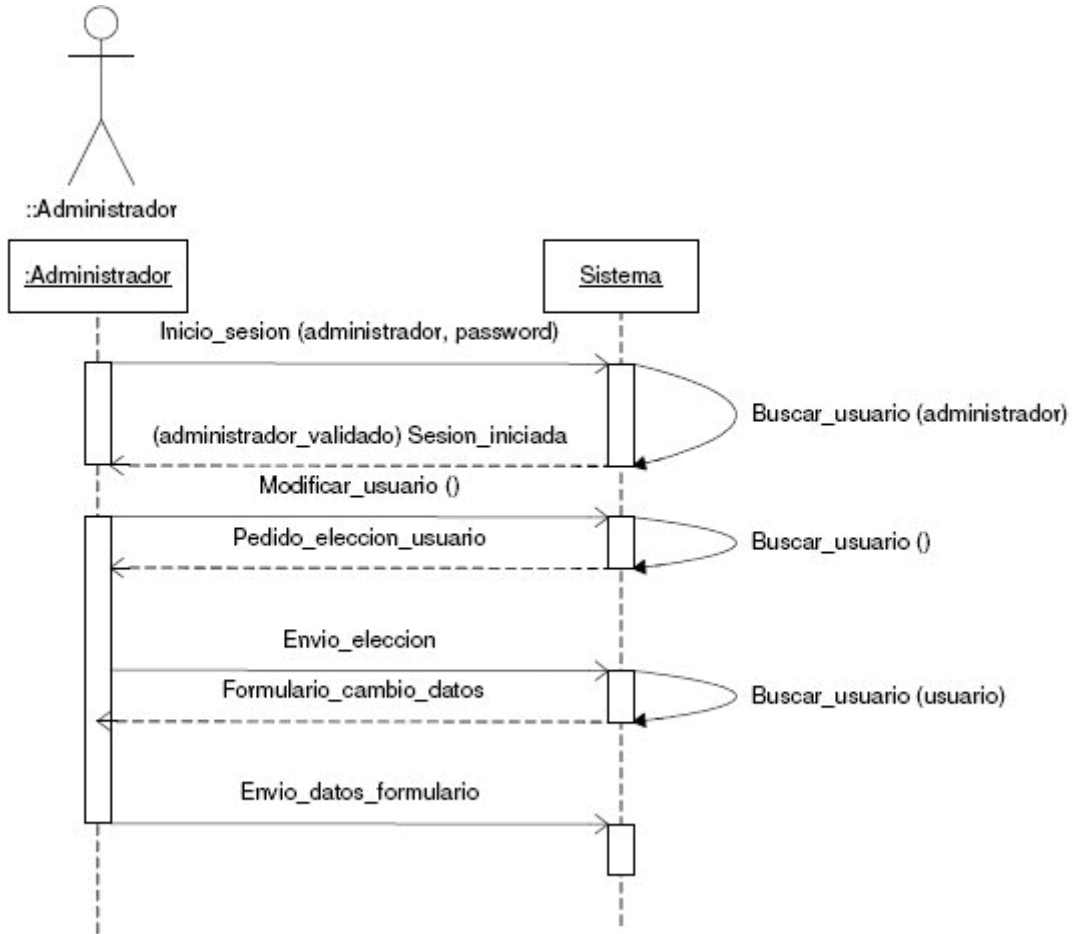


Figura 14.2: Diagrama de secuencia Modificar usuario

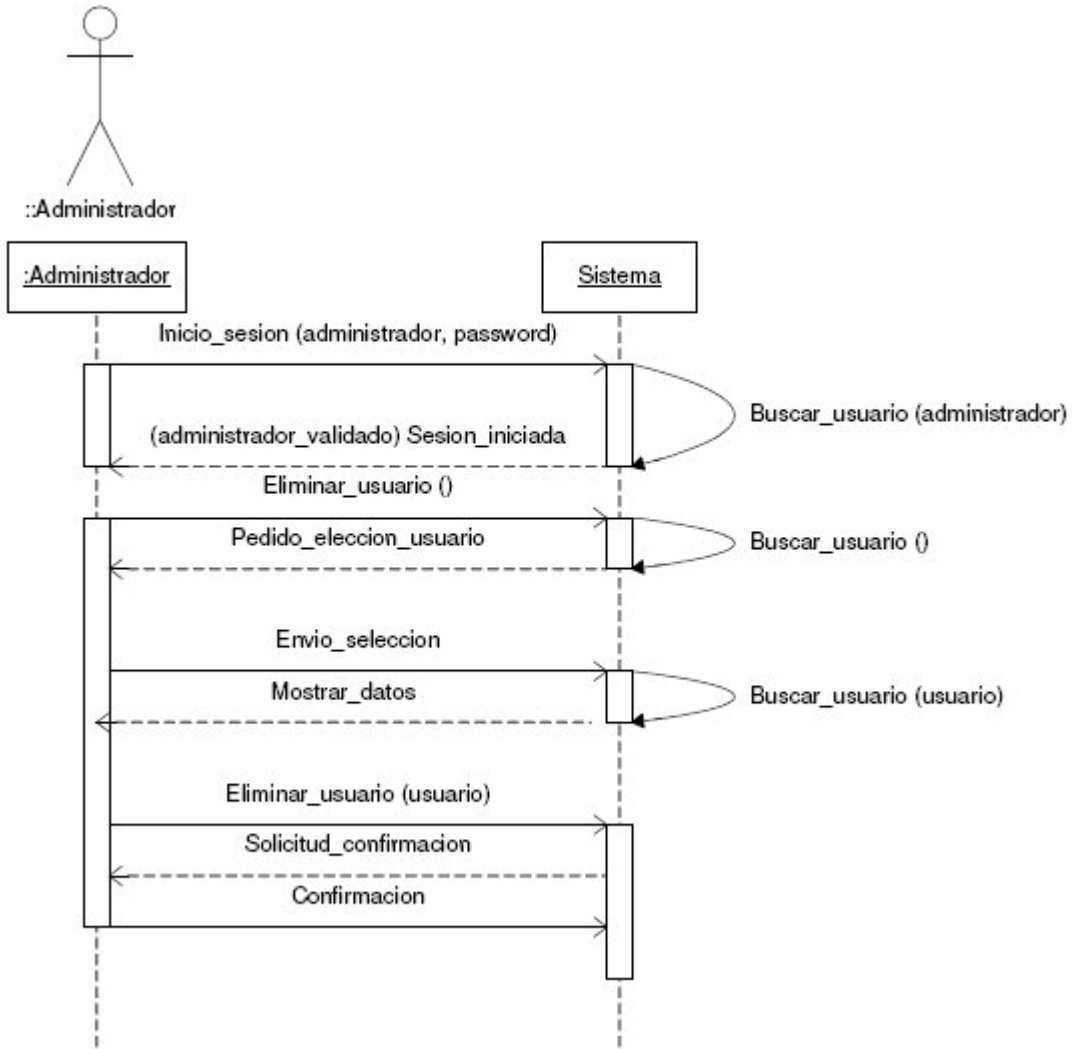


Figura 14.3: Diagrama de secuencia Eliminar usuario

14.3 Diagramas de actividad

A continuación se presentan los diagramas de actividad correspondientes al caso de uso “Administrador”

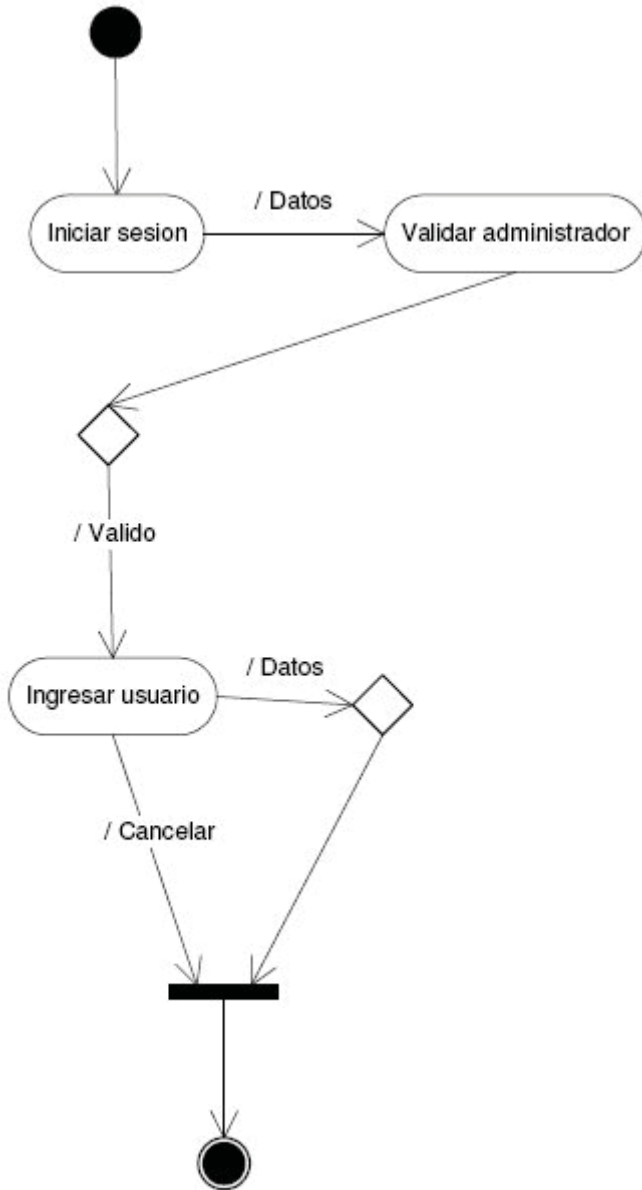


Figura 14.4: Diagrama de actividad Ingresar usuario

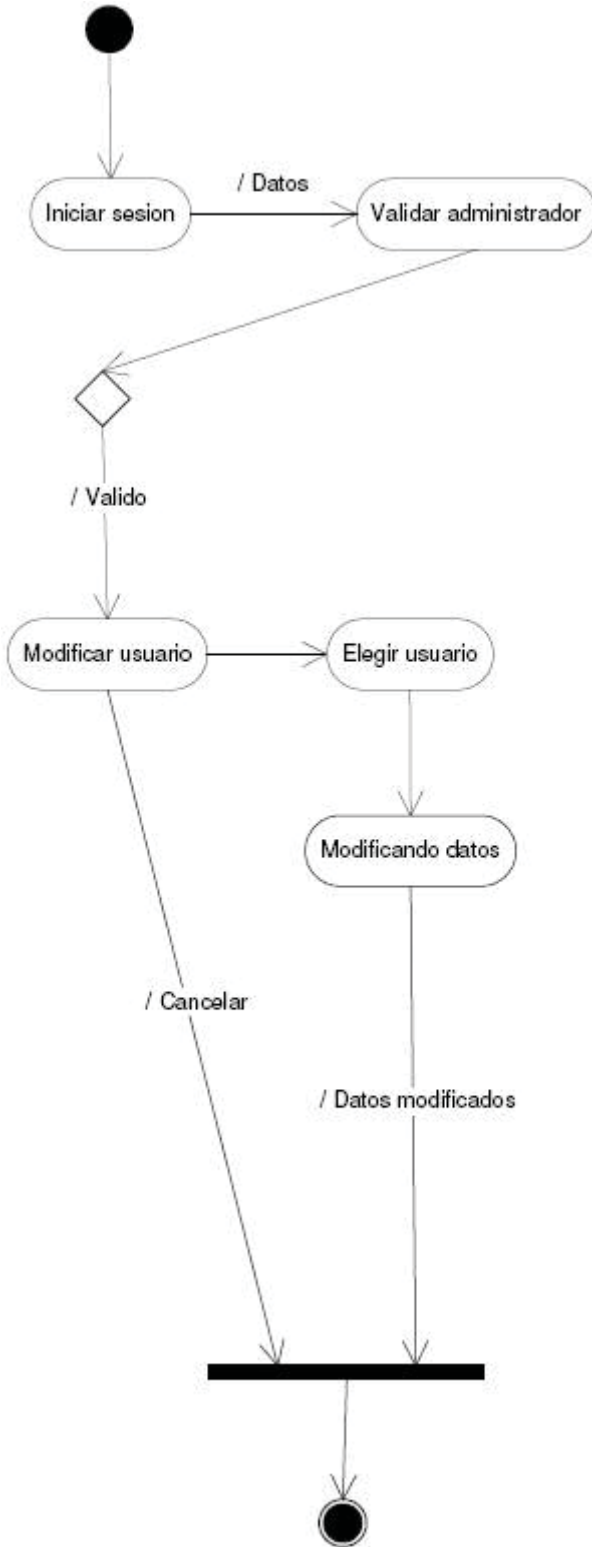


Figura 14.5: Diagrama de actividad Modificar usuario

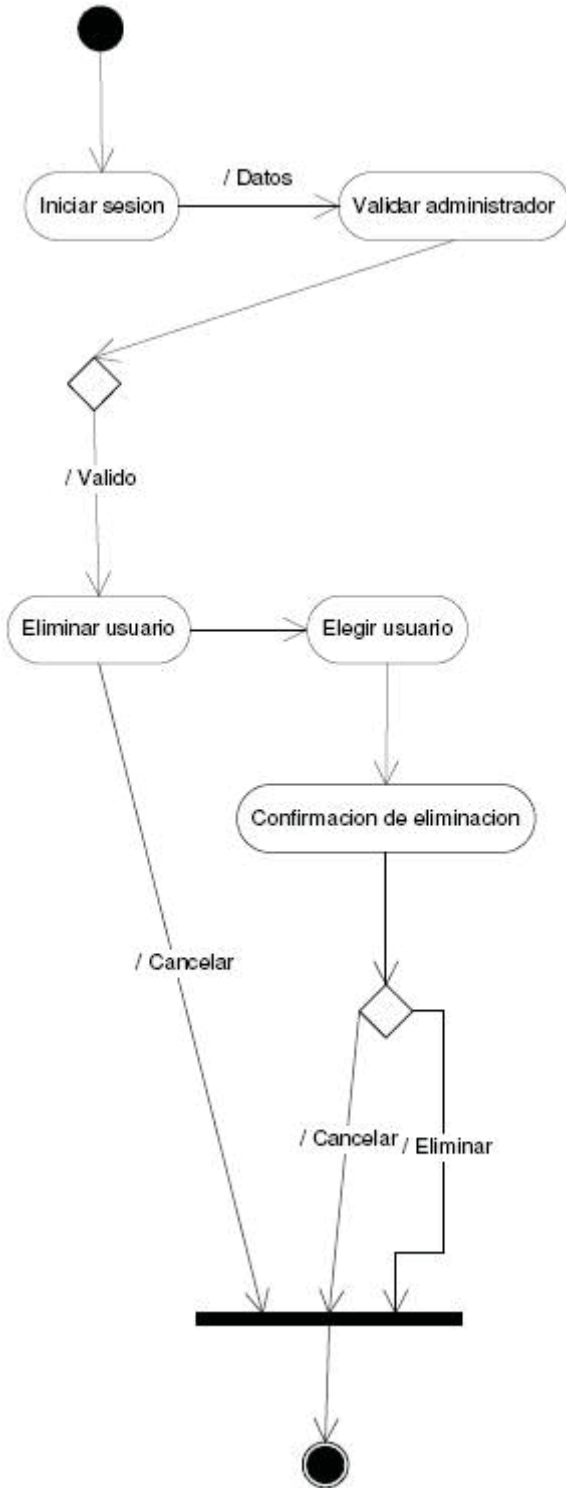


Figura 14.6: Diagrama de actividad Eliminar usuario

14.4 Diagramas de componentes

A continuación se presentan los diagramas de componentes correspondientes al caso de uso “Administrador”

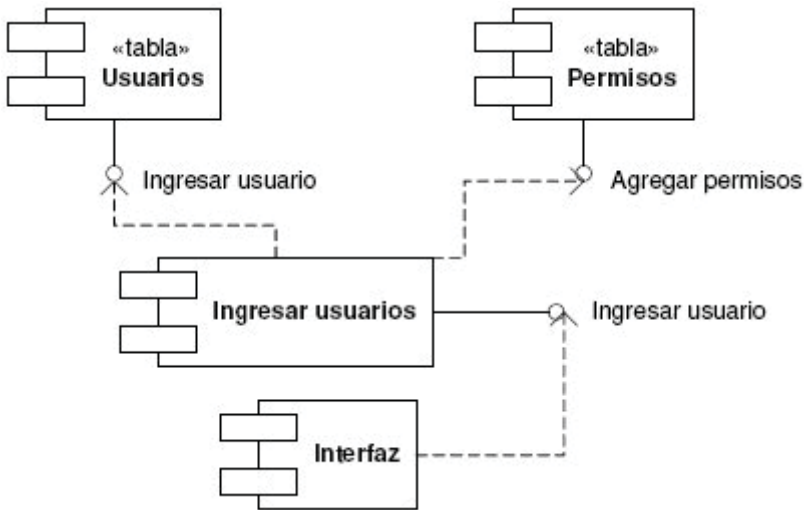


Figura 14.7: Diagrama de componente Ingresar usuario

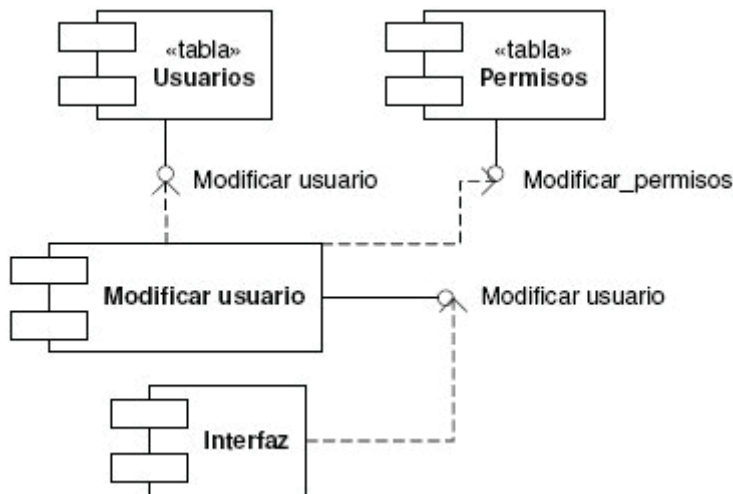


Figura 14.8: Diagrama de componente Modificar usuario

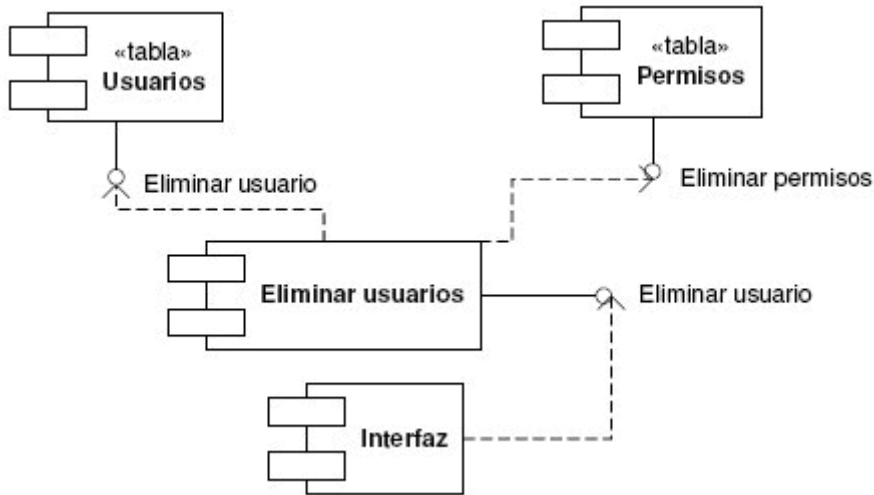


Figura 14.9: Diagrama de componente Eliminar usuario

14.5 Interfaces

A continuación se presentan las interfaces correspondientes al caso de uso “Administrador”

The screenshot shows a web application window titled 'Ingresar usuario'. The window is divided into two main sections: 'Cuenta de usuario' and 'Permisos'. The 'Cuenta de usuario' section contains three input fields: 'Nombre de usuario', 'Contraseña', and 'Confirmar contraseña'. The 'Permisos' section contains a list of permissions with checkboxes: 'Ingreso', 'Modificación', 'Eliminación', 'Bloqueo', 'Desbloqueo', 'Acceso al Historial', and 'Administrador'. At the bottom right of the window, there are two buttons: 'Crear' and 'Cancelar'.

Figura 14.10: Interfaz Ingresar usuario

Modificar usuario

Busqueda

Login de usuario

Datos del usuario

Cuenta de usuario

Nombre de usuario

Contraseña

Confirmar contraseña

Datos Personales

Nombre

Apellido paterno

Apellido materno

Permisos

Las casillas marcadas en la siguiente sección son los permisos que el usuario tiene asignados actualmente.

Ingreso Modificación

Eliminación Bloqueo

Desbloqueo Acceso al Historial

Administrador

Figura 14.11: Interfaz Modificar usuario

Eliminar usuario

Nombre de usuario

Datos del Usuario

Cuenta de usuario

Nombre de usuario

Contraseña

Datos Personales

Nombre

Apellido paterno

Apellido materno

Permisos

Las casillas marcadas en la siguiente sección representan las acciones a las que el usuario tiene acceso actualmente.

Ingreso Modificación

Eliminación Bloqueo

Desbloqueo Acceso al Historial

Administrador

Figura 14.12: Interfaz Eliminar usuario

Como el administrador del sistema es un usuario registrado, tiene acceso a todas las interfaces y funcionalidades completas del sistema (referencia 13.5 Interfaces usuario registrado).

14.6 Pruebas correspondientes a la tercera iteración

En la presente sección del informe se pretende dar a conocer algunas de las pruebas realizadas en el modulo correspondiente a la tercera iteración de la fase de construcción del proyecto (Modulo administrador), relacionados con los siguientes casos de uso:

Ingresar usuario
Modificar usuario
Eliminar usuario

En el caso de ser un administrador del sistema, se tiene acceso a todas las funcionalidades del proyecto, además de funcionalidades para el manejo de usuarios dentro del sistema (ningún otro tipo de usuario puede acceder a ellas).

Casos de prueba “Ingresar usuario”

Tabla 14.4: caso de prueba Ingresar usuario.

Caso n°	1.
Descripción	En la interfaz principal del sistema se despliega el menú “Cuentas de usuarios”, luego se va al menú “Ingresar usuario” y por ultimo se ingresan los datos correspondientes.
Entrada	Se ingresan los datos correspondientes omitiendo algún campo (uno o mas) y se presiona el botón crear
Salida	Se envía el mensaje de error correspondiente al caso y se espera el ingreso del campo omitido

Datos	<p>Se dejan vacíos los campos = contraseña y nombre.</p> <p>Se dejan vacíos los campos = nombre de usuario y apellido paterno.</p>
-------	--

Tabla 14.5: caso de prueba Ingresar usuario.

Caso n°	2.
Descripción	En la interfaz principal del sistema se despliega el menú “Cuentas de usuarios”, luego se va al menú “Ingresar usuario” y por ultimo se ingresan los datos correspondientes.
Entrada	Se ingresan los datos correspondientes y se presiona el botón crear
Salida	Se envía el mensaje confirmando el ingreso al usuario y preguntando si se va a realizar un nuevo ingreso.
Datos	Nombre de usuario = james, contraseña 12345, nombre = Jorge, apellido paterno = Muñoz, apellido materno = Silva, permisos = administrador.

Tabla 14.6: caso de prueba Ingresar usuario.

Caso n°	3.
Descripción	En la interfaz principal del sistema se despliega el menú “Cuentas de usuarios”, luego se va al menú “Ingresar usuario” y por ultimo se ingresan los datos correspondientes.
Entrada	Se ingresan los datos correspondientes a un usuario existente y se presiona el botón crear

Salida	Se notifica al usuario que el usuario ya existe y se espera una decisión del usuario (ya sea cambiar el nombre del usuario o cancelar el ingreso)
Datos	Nombre de usuario = james, contraseña 12345, nombre = Jorge, apellido paterno = Muñoz, apellido materno = Silva, permisos = administrador.

Casos de prueba “Modificar usuario”

Tabla 14.7: caso de prueba Modificar usuario.

Caso n°	4.
Descripción	En la interfaz principal del sistema se despliega el menú “Cuentas de usuarios”, luego se va al menú “Modificar usuario” y por ultimo se ingresan los datos correspondientes.
Entrada	Se selecciona un nombre de usuario existente del combo y se presiona buscar, luego se realizan cambios en los datos y se presiona modificar.
Salida	Se presenta al administrador los datos del usuario solicitado, al presionar modificar se presenta un mensaje confirmando que los cambios fueron realizados satisfactoriamente y se sale de la interfaz.
Datos	Nombre de usuario = james. Nombre de usuario = administrador.

Tabla 14.8: caso de prueba Modificar usuario.

Caso n°	5.

Descripción	En la interfaz principal del sistema se despliega el menú “Cuentas de usuarios”, luego se va al menú “Modificar usuario” y por ultimo se ingresan los datos correspondientes.
Entrada	Se selecciona un nombre de usuario existente del combo (se escoge el nombre del usuario con el que se ingreso al sistema) y se presiona buscar, luego se realizan cambios en los datos y se presiona modificar.
Salida	Se presenta al administrador los datos del usuario solicitado (la interfaz no permite que el administrador se cambie los permisos a si mismo), al presionar modificar se presenta un mensaje confirmando que los cambios fueron realizados satisfactoriamente y se sale de la interfaz.
Datos	Nombre de usuario = james. Nombre de usuario = administrador.

Tabla 14.9: caso de prueba Modificar usuario.

Caso n°	6.
Descripción	En la interfaz principal del sistema se despliega el menú “Cuentas de usuarios”, luego se va al menú “Modificar usuario” y por ultimo se ingresan los datos correspondientes.
Entrada	Se escribe un nombre de usuario inexistente y se presiona el botón buscar.
Salida	Se notifica al usuario que el usuario No existe.
Datos	Nombre de usuario = james. Nombre de usuario = administrador.

Casos de prueba “Eliminar usuario”

Tabla 14.10: caso de prueba Eliminar usuario.

Caso n°	7.
Descripción	En la interfaz principal del sistema se despliega el menú “Cuentas de usuarios”, luego se va al menú “Eliminar usuario” y por ultimo se ingresan los datos correspondientes.
Entrada	Se selecciona un nombre de usuario existente del combo y se presiona buscar, luego se presiona el botón buscar y luego el botón eliminar
Salida	Se presenta al administrador los datos del usuario solicitado, al presionar eliminar se presenta un mensaje solicitando confirmar la decisión, de ser respuesta positiva se presenta un mensaje confirmando la eliminación y se sale de la interfaz, si la respuesta es negativa se espera una nueva búsqueda.
Datos	Nombre de usuario = james. Nombre de usuario = administrador.

Tabla 14.11: caso de prueba Eliminar usuario.

Caso n°	8.
Descripción	En la interfaz principal del sistema se despliega el menú “Cuentas de usuarios”, luego se va al menú “Eliminar usuario” y por ultimo se ingresan los datos correspondientes.
Entrada	Se selecciona un nombre de usuario existente del combo (se escoge el nombre del

	usuario con el que se ingreso al sistema) y se presiona buscar.
Salida	Se presenta un mensaje al usuario indicándole que no se puede eliminar a si mismo.
Datos	Nombre de usuario = james. Nombre de usuario = administrador.

Tabla 14.12: caso de prueba Eliminar usuario.

Caso n°	9.
Descripción	En la interfaz principal del sistema se despliega el menú “Cuentas de usuarios”, luego se va al menú “Eliminar usuario” y por ultimo se ingresan los datos correspondientes.
Entrada	Se escribe un nombre de usuario inexistente y se presiona el botón buscar.
Salida	Se notifica al usuario que el usuario No existe.
Datos	Nombre de usuario = james. Nombre de usuario = administrador.

15. CONCLUSIONES

Durante el desarrollo de este proyecto, se pudo apreciar la gran dificultad de planificar y realizar un proyecto informático intentando mantener el orden y el control en el grupo de trabajo. Esto depende en gran parte a la colaboración entre los integrantes del mismo. A la vez mantener un contacto continuo con el cliente para poder obtener de manera correcta los requerimientos del sistema y no terminar realizando algo que no se pide o que el cliente no termine satisfecho. En muchas oportunidades el cliente no posee una idea clara de lo que desea, por lo que el entrevistador debe saber guiar al cliente y entre ambos llegar a un acuerdo de los requerimientos finales del sistema.

En forma paralela al análisis del sistema, se realizó una investigación de la empresa referente al reglamento interno, con el fin de interiorizarse con el proceso de trabajo y las normas que el sistema debía cumplir (por ejemplo: licencias, presentación, simbología) realizando además visitas a la zona de trabajo y a los usuarios finales del sistema. Lo cual resulto de gran ayuda al momento de construir las interfaces del programa y los accesos a las funcionalidades que este posee. Al interiorizarse dentro del modo de trabajo de la empresa se notaron deficiencias en algunos aspectos del flujo de trabajo (por ejemplo: no detallar salida o entrada de carpetas, ingreso de distintos formatos de la información, errores masivos de datos), por lo que el sistema aparte de cumplir su función principal permite superar o al menos disminuir las deficiencias encontradas durante la investigación.

Durante las primeras iteraciones, se tuvieron algunos problemas referentes al diseño, por lo que se opto por realizar algunos cambios en las clases definidas y en algunas funcionalidades del proyecto en si, lo que demuestra que el diseño no fue tan preciso como se esperaba, pero aun así se logro llevar a buen termino el proyecto, terminando con un prototipo que posee la funcionalidad completa del sistema. Dicho prototipo fue presentado ante una comisión perteneciente al departamento de transito, mostrando todas y cada una de las funcionalidades del prototipo y dando a conocer las mejoras no tangibles que presenta el sistema, con lo cual se pudo

apreciar la opinión del comité y las aprensiones que tenían los usuarios sobre el cambio al nuevo sistema.

Se presento un prototipo que presentaba parte de la funcionalidad del sistema a los usuarios finales, lo cual tenia como objetivo conocer la opinión, sugerencias y consultas que poseían los usuarios. Dando como resultado una evaluación favorable por parte de los usuarios y opiniones referentes a las interfaces del mismo (cambio en el color de algunos campos y en el tamaño de las interfaces). Gracias a esto se logro obtener una visión de lo que realmente quería el usuario del sistema, evitando de esta manera la necesidad de realizar cambios más drásticos en las etapas finales de la construcción del sistema.

Al termino de este proyecto, se logro cumplir con los objetivos propuestos en un comienzo, esto gracias a una planificación, la cual se intento cumplir a cabalidad, aun cuando se incurrió en retrasos parciales debido a la indecisión del cliente (diseño, construcción y pruebas del modulo generación de informes) y reuniones las cuales fueron pospuestas por distintos motivos.

Una gran ayuda que facilito el trabajo realizado fue la elección de un lenguaje estándar para el modelado de los requerimientos y el diseño de los módulos como es UML. Debido a que este posee una notación estándar y concisa permitió un entendimiento entre los integrantes del grupo de trabajo a la hora de dividir los modelos que debían ser realizados, con lo cual se logro trabajar de forma paralela e independiente.

Para finalizar, se puede concluir que no solo una buena planificación, conocimiento del tema y las herramientas son suficientes para poder llevar a término un proyecto, influyen en gran parte las relaciones entre los integrantes del grupo de trabajo, junto con la experiencia y el ámbito del proyecto.

16. REFERENCIAS

- [1] Arnow D., Weiss G.: Introducción a la programación con JAVA un enfoque orientado a objetos, Pearson educación SA, 2001.
- [2] Jacobson, Ivar, Booch, Grady, Rumbaugh, James: El lenguaje unificado de modelado, Pearson educación SA, 2000.

- [3] Jacobson, Ivar, Booch, Grady, Rumbaugh, James: El Proceso Unificado de Desarrollo de Software, Pearson educación SA, 1999.
- [4] Pressman R.: Ingeniería de software Un enfoque practico, Mc Graw-Hill, 2002.
- [5] Sommerville I.: Ingeniería de software, Pearson educación SA, 2002.
- [6] Popkin Software and Systems, Modelado de sistemas con UML, <http://es.tldp.org/Tutoriales/doc-modelado-sistemas-UML/multiple-html/x291.html> -
- [7] Popkin Software and Systems, Modelado de sistemas con UML, <http://es.tldp.org/Tutoriales/doc-modelado-sistemas-UML/multiple-html/x219.html>
- [8] ***, Documento sobre RUP, <http://lml.ls.fi.upm.es/mdp/si/pres-rup.pdf>
- [9] Figueroa P., Elementos notacionales de UML 1.0, <http://www.cs.ualberta.ca/~pfiguero/soo/uml/secuencia01.html>
- [10] Figueroa P., Elementos notacionales de UML 1.0, <http://www.cs.ualberta.ca/~pfiguero/soo/uml/implementacion01.html>
- [11] ***, Artículo sobre Oracle, http://www.desarrolloweb.com/directorio/bases_de_datos/oracle/
- [12] Som G., Curso de iniciación a la programación con Visual Basic .NET, <http://www.mundoprogramacion.com/NET/cursoVB.NET/indice.htm>-
- [13] ***, Tutorial de UML, http://www.mundotutoriales.com/tutoriales_uml-mdpal13987.htm -

ANEXO A: PLAN DE TRABAJO

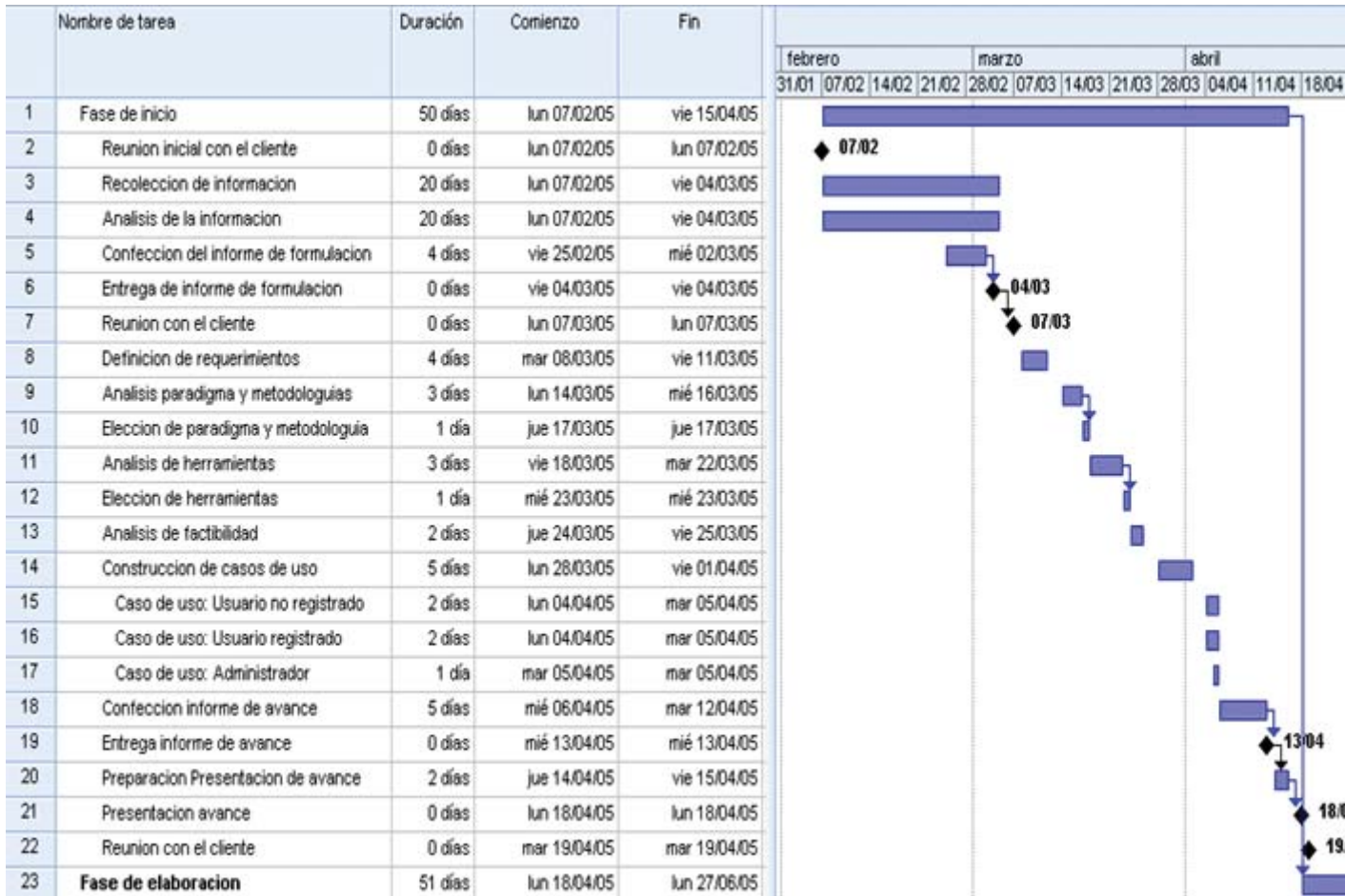


Figura A.1: Carta Gantt parte 1

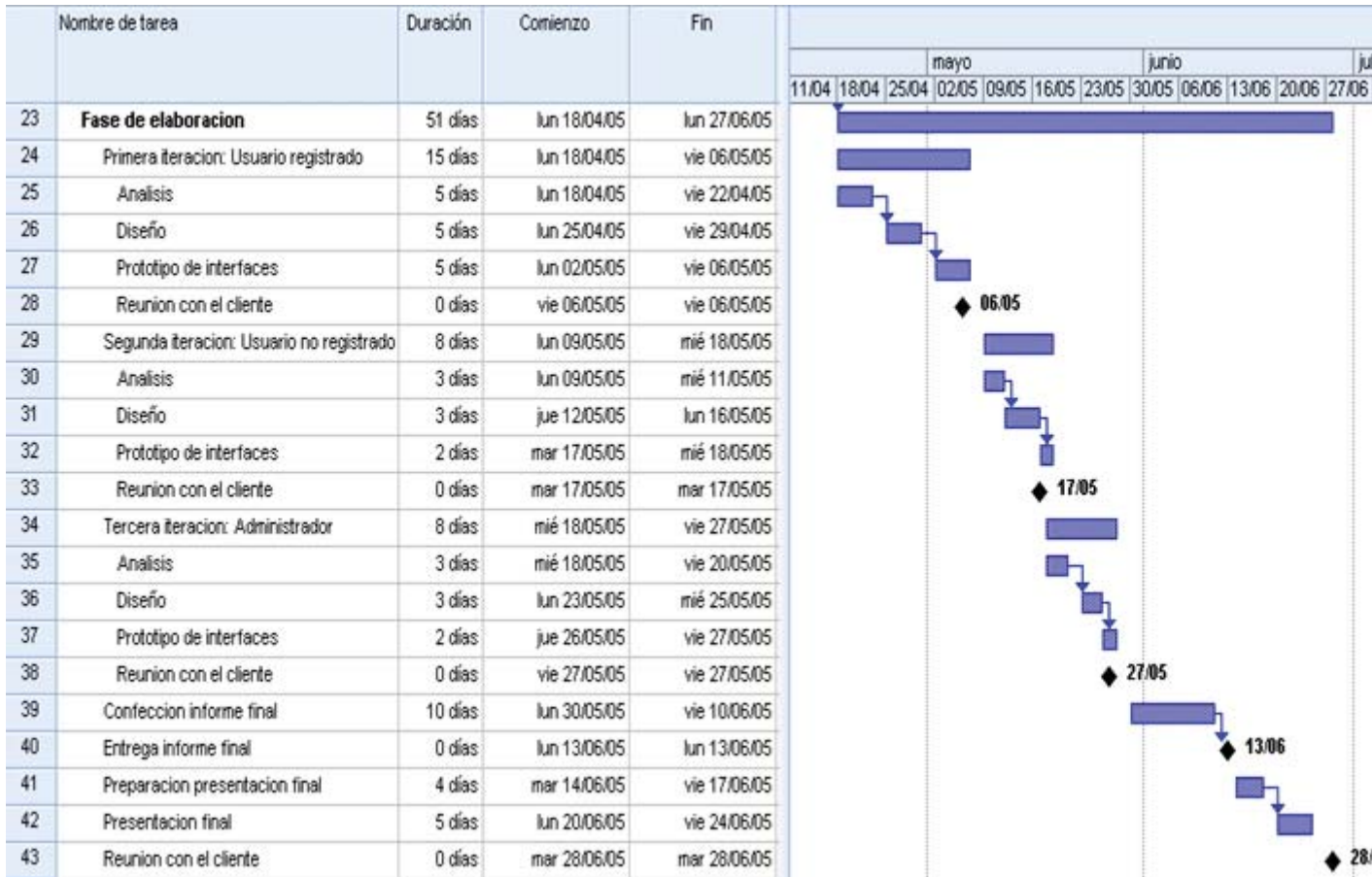


Figura A.2: Carta Gantt parte 2

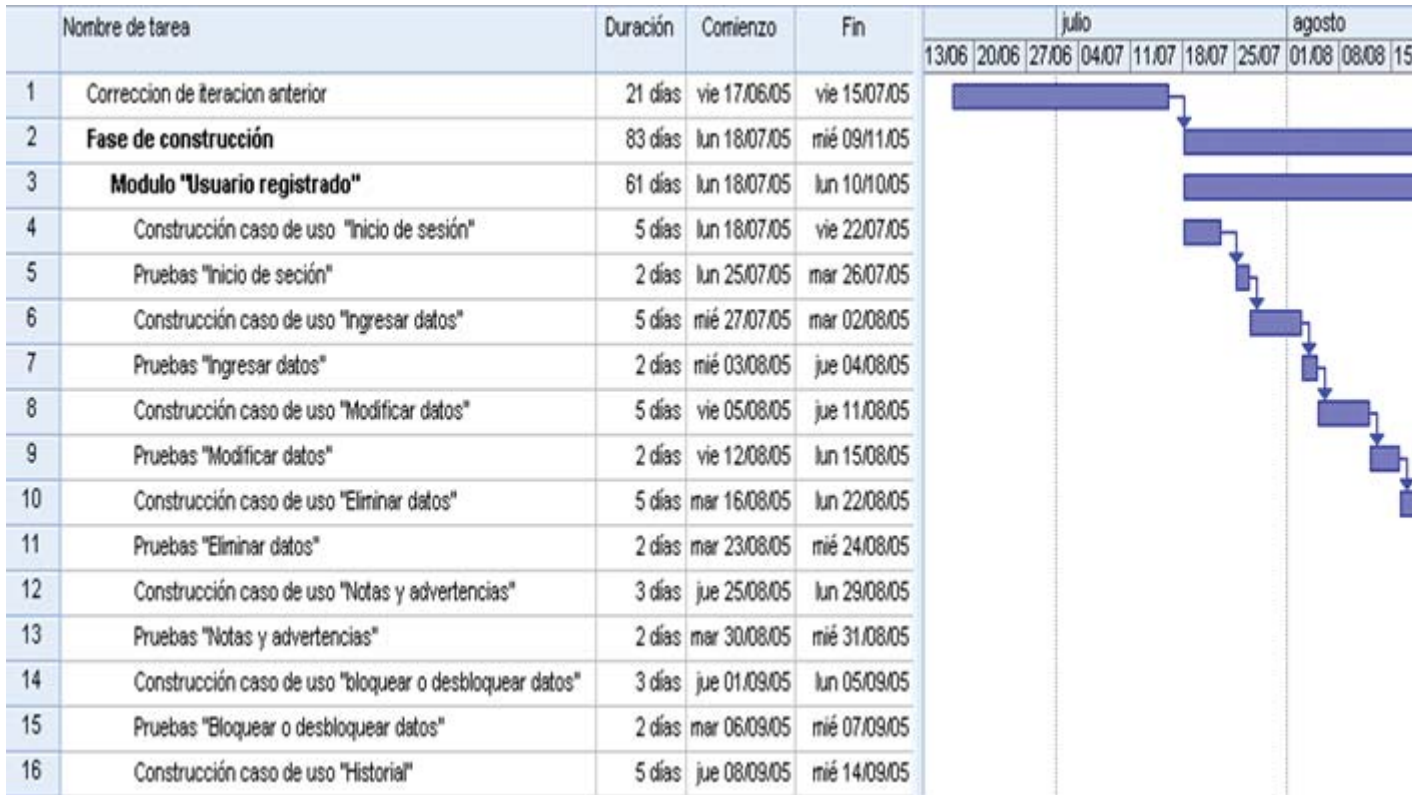


Figura A.3: Carta Gantt parte 3

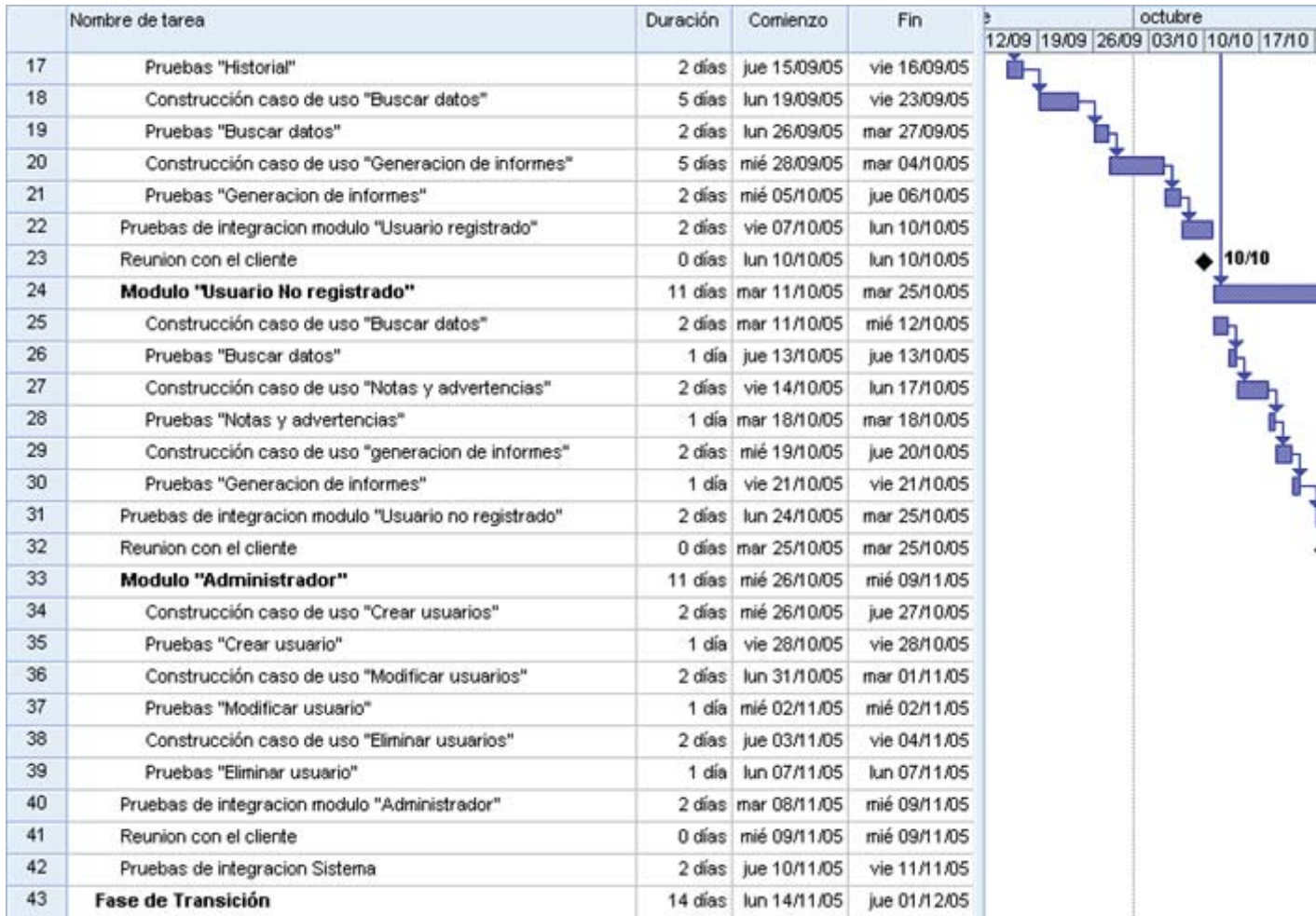


Figura A.4: Carta Gantt parte 4