

**Universidad de las Ciencias Informáticas.
Facultad 4**



**TÍTULO: Recuperaciones Dinámicas para
Aplicaciones de Gestión.**

Trabajo de Diploma para optar por el título de
Ingeniero en Ciencias Informáticas

Autores: Elisabeth Pérez Pérez
Dalkis Mok Vázquez

Tutor: MSc. Raquel Maria Vidal Soto
Consultante: Ing. Yuleisi Vives Sarría

Ciudad de la Habana, Junio 2007.
Año 49 de la Revolución

DECLARACIÓN DE AUTORÍA

Declaramos que somos los únicos autores de este trabajo y autorizamos al Ministerio de las Fuerzas Armadas Revolucionarias (MINFAR) y a la Universidad de las Ciencias Informáticas (UCI) para que hagan el uso que estimen pertinente con este trabajo.

Para que así conste firmo la presente a los ____ días del mes de _____ del año _____.

Dalkis Mok Vázquez

Elisabeth Pérez Pérez

MSc Raquel M. Vidal Soto

*“La sociedad que no se prepara para el uso de la computación está
liquidada”*

Fidel Castro Ruz.

AGRADECIMIENTOS

A Fidel y a la UCI, por su confianza en nosotros y darnos la posibilidad de ser parte de este proyecto futuro.

A nuestra tutora Raquel por sus valiosos consejos y su ayuda.

A nuestros amigos del proyecto: Karina, Palma, Borbón, Rolando, Wilber por su colaboración para alcanzar este triunfo.

A Yoisi por su ayuda, por su preocupación y cariño.

A nuestros compañeros y amigos que juntos hemos transitado por este camino y que de una forma u otra, nos han ayudado en la realización de este trabajo.

A mi madre y mi manita Valkis por todo el amor, apoyo y confianza que me brindaron.

A mis abuelos Américo y Belkis porque sin escatimar esfuerzo alguno, han sacrificado gran parte de su vida para formarme.

A Chichi, mi tío Alex y mi tía Aurora porque gracias a su cariño, apoyo y consejo he llegado a realizar la más grande de mis metas.

A ti Alieski, mi amor por tener tanta paciencia, y noches de desvelos a mi lado.

A Elisabeth, Yadira, Mayte, Jordanis y Alain porque gracias a ustedes y su gran amistad he podido lograr este triunfo.

Dalkis

*A mis padres, por todo el amor que me brindan
Por haber sido mis guías en todo momento.
Por haber confiado siempre en mí.
Por ser unos padres maravillosos.*

*A mi abuela y a mi tío por su cariño y su apoyo.
Por ser tan especiales para mí.*

A mi hermanita por sus locos y tiernos consejos.

*A Jorde por su inmensa ayuda, por su amor
y comprensión, por estar siempre en los
buenos y malos momentos. Por ser la
maravillosa persona que es.*

*A Raúl y Odelis por su afecto, su apoyo
y por todas sus palabras de aliento.*

*A Dalkis y Alieski por su ayuda y su amistad,
porque juntos alcanzamos este triunfo.*

Elisabeth

DEDICATORIA

A mi mamá y mi manita Valkis por ser las personas más especiales en mí vida, y que sepan siempre que este triunfo es de ustedes también.

A mis abuelos Belkis y Américo que soñaron tanto con este momento y que me enorgullece darles este pedacito de felicidad.

A ti Ale, por confiar tanto en mí y por todo el amor que siento por ti.

A ti lisi mi amiga porque sin ti nunca hubiera logrado este gran sueño

A mis padres, a mi abuela, a mi tío y a mi hermano para darles otro motivo por el que sentirse orgullosos.

A mi Jorde por su confianza en mí y por toda la admiración y amor que siento por él.

A mi hermanita tete, para que siga el ejemplo de su hermana que la adora.

A Dalkis por ser mi mejor amiga, por ser la compañera ideal en este último reto.

A mis abuelos porque, aunque ya no están, siempre quisieron ver realizado este sueño.

Dalkis

Elisabeth

RESUMEN

El uso eficiente de las Tecnologías de la Información es un reto en la sociedad cubana actual. La recuperación dinámica de la información es un proceso al que se le ha dado una significativa importancia dentro de los procesos de informatización que se realizan en el Ministerio de las Fuerzas Armadas Revolucionarias (MINFAR).

El presente trabajo está dirigido a la realización de Recuperaciones Dinámicas para Aplicaciones de Gestión, teniendo en cuenta que el proceso de gestión de la información, que incluye la generación de informes, juega un papel fundamental en el funcionamiento de las diferentes entidades pertenecientes al MINFAR, por lo que se hace imprescindible darle un adecuado tratamiento a este proceso. Actualmente existe una necesidad evidente de realizar recuperaciones dinámicas que permitan cubrir las necesidades informativas de los usuarios, de una manera eficiente, segura y con calidad.

Este trabajo tiene como objetivo concreto la modelación de una aplicación Web, de interfaz amigable, que garantice la realización de recuperaciones dinámicas donde se le permita especificar al usuario exactamente la información que él necesita recuperar y de esta forma lograr satisfacer sus necesidades informativas.

Con la modelación de esta herramienta esperamos ofrecer resultados como:

- Proveer de la modelación de una herramienta de apoyo en el análisis de información considerada relevante, al Ministerio de las Fuerzas Armadas Revolucionarias.
- Brindar información relacionada con la recuperación de información, así como las características de las tecnologías con las que se propone desarrollar el sistema.

Este documento recoge los resultados de todo el trabajo realizado. Se estudian características correspondientes a algunas de las aplicaciones de recuperación de información existentes; se incluyen los conceptos relacionados con la recuperación de información y se exponen características de las tecnologías seleccionadas para desarrollar la aplicación. Finalmente se muestran los resultados del análisis y diseño de la propuesta del sistema, y se proponen algunas recomendaciones para el desarrollo futuro del mismo.

Palabras Claves

Información, Recuperaciones dinámicas, Aplicación Web.

ÍNDICE

AGRADECIMIENTOS	II
DEDICATORIA	III
RESUMEN	IV
INTRODUCCIÓN	1
CAPÍTULO 1 FUNDAMENTACIÓN TEÓRICA	5
1.1 INTRODUCCIÓN	5
1.2 RECUPERACIÓN DE LA INFORMACIÓN	5
1.2.1 Teoría de grafos y algoritmos de grafos.....	8
1.3 OBJETO DE ESTUDIO.....	9
1.3.1 Objetivos estratégicos de la organización	9
1.3.2 Flujo actual de los procesos	10
1.3.3 Análisis crítico de la ejecución de los procesos.....	10
1.4 SISTEMAS AUTOMATIZADOS EXISTENTES VINCULADOS AL CAMPO DE ACCIÓN.....	11
1.5 TENDENCIAS Y TECNOLOGÍAS ACTUALES	13
1.5.1 Aplicaciones Web	13
1.5.2 Arquitectura n capas	14
1.5.3 Software Libre	15
1.5.4 Lenguajes de programación para la Web.....	15
1.5.5 Navegador.....	18
1.5.6 Metodología de Ingeniería del software.....	19
1.5.7 Herramientas CASE.....	20
1.5.8 Bases de Datos.....	21
1.5.9 Sistema Gestor de Bases de Datos (SGBD).....	21
1.5.10 Dreamweaver 8.....	24
1.6 CONCLUSIONES.....	24
CAPÍTULO 2 CARACTERÍSTICAS DEL SISTEMA	25
2.1 INTRODUCCIÓN.....	25
2.2 DESCRIPCIÓN DE LOS PROCESOS DEL NEGOCIO PROPUESTO.....	25
2.3 MODELO DEL DOMINIO	26
2.3.1 Conceptos principales.....	26
2.3.2 Principales Eventos	27
2.3.3 Diagrama de clases del dominio.....	27
2.4 REQUERIMIENTOS FUNCIONALES	28
2.5 REQUERIMIENTOS NO FUNCIONALES	30

2.6	MODELO DE CASOS DE USO DEL SISTEMA	32
2.6.1	Diagrama de Caso de Uso del Sistema.....	35
2.6.2	Expansión de los Casos de Uso.....	36
2.7	CONCLUSIONES	60
CAPÍTULO 3 ANÁLISIS Y DISEÑO		61
3.1.	INTRODUCCIÓN	61
3.2.	MODELO DEL ANÁLISIS	61
3.2.1	Diagrama de clases del análisis	61
3.3.	MODELO DEL DISEÑO	66
3.3.1	Mecanismos de diseño	66
3.3.2	Diagrama de clases del diseño.....	66
3.3.3	Diagramas de Interacción.....	77
3.4	DISEÑO DE LA BASE DE DATOS	87
3.4.1	Diagrama de clases persistentes.....	87
3.4.2	Modelo de datos	88
3.5	PRINCIPIOS DE DISEÑO	96
3.6	TRATAMIENTO DE ERRORES	96
3.7	CONCLUSIONES	97
CAPÍTULO 4 IMPLEMENTACIÓN Y PRUEBA		98
4.1	INTRODUCCIÓN	98
4.2	MODELO DEL DESPLIEGUE	98
4.3	MODELO DE IMPLEMENTACIÓN	99
4.4	DISEÑO DE CASOS DE PRUEBA	103
4.5	CONCLUSIONES	106
CONCLUSIONES GENERALES		107
RECOMENDACIONES		108
BIBLIOGRAFÍA		109
GLOSARIO DE TÉRMINOS		111

INTRODUCCIÓN

En la era de Internet se han dado varios pasos en la accesibilidad a la información, pero se han dado varios pasos atrás en la facilidad de entender el contexto y encontrar conclusiones. Debido a la gran importancia que posee el acceso a la información, se busca la forma más eficiente de recuperarla, analizarla y de esta forma, poner a disposición de los usuarios la información necesitada sin tener que recurrir a terceros.

La *información* es la comunicación o adquisición de conocimientos que permiten ampliar o precisar los que se poseen sobre una materia determinada.

La información tiene como función:

- ✓ Aumentar el conocimiento del usuario
- ✓ Proporcionar a quien toma una decisión probabilidades para la elección, reduciendo la gama de decisiones
- ✓ Proporcionar una serie de reglas de evaluación y reglas de decisión para fines de control.

El incremento de información ha traído consigo el desarrollo de las tecnologías de la información buscando la mejor forma de poner a disposición del usuario todo tipo de recursos.

La *recuperación de información* forma una parte importante de las aplicaciones de software, permitiendo el acceso rápido y objetivo a los datos, haciendo la información manejable y de gran utilidad, teniendo como principales objetivos resaltar la información mas relevante, resumir la información obtenida y recuperar exactamente la información deseada.

La *recuperación dinámica de información* permite que los usuarios puedan personalizar sus propios informes y trae consigo un incremento de la flexibilidad para representar y analizar datos.

La visualización de la información es esencial en casi cualquier sistema de hoy en día. Existen diferentes formas de representar la información recuperada siendo un factor determinante la necesidad informativa del usuario. De las diferentes formas de representar información se puede resaltar que cuando se desea recuperar gran cantidad de información, la mejor forma de representarla es conformando un listado de los diferentes elementos o datos recuperados. Es también cierto que muchos datos e información deben ser procesados y representados por tablas o gráficos para que sirvan de alguna ayuda,

especialmente cuando se trata de información estadística. La representación de información estadística utilizando tablas o gráficos ofrece mensajes más claros y por tanto fáciles de entender. El uso de tablas o gráficos se aprovecha también para resumir información, o para resaltar información relevante.

El Ministerio de las Fuerzas Armadas Revolucionarias (MINFAR) dada su gran importancia, no ha quedado exento al desarrollo de la informática y ha adoptado nuevas medidas para llevar la informatización a la mayoría de los procesos fundamentales de su funcionamiento. Dentro de todo esto se le ha dado gran importancia a la gestión de la información de forma dinámica para una mayor interacción con la misma, delegando gran parte de esta a la representación de la información, buscando una mejor visualización que permita la representación organizada de la información y de esta forma poder percibir fácilmente los datos esenciales y realizar un análisis óptimo y flexible de los mismos.

Actualmente en el Ministerio de las Fuerzas Armadas Revolucionarias pocos sistemas informativos cuentan con herramientas que permitan la recuperación, visualización o el análisis del comportamiento de información. Los sistemas que existen, o las herramientas automatizadas en las que se apoyan para este fin, no son sistemas dinámicos, y además la representación de la información no tiene la calidad suficiente porque no siempre se abarcan las necesidades informativas del usuario. En la mayoría de los casos se hace necesaria la intervención de terceros para la búsqueda y representación de la información necesitada. Estos pocos sistemas que existen están limitados a un sistema específico y por tanto a un análisis y diseño obsoleto que no permite cambios. Por lo que se presenta la necesidad de un sistema dinámico que permita la representación de la información recuperada dinámicamente de forma clara, fácil de entender que posibilite un análisis flexible de esta.

Por tanto el **problema** a resolver queda formulado a modo de interrogante de la siguiente forma: ¿Cómo modelar un sistema que permita recuperar y representar la información contenida en las bases de datos del MINFAR de una manera flexible a las necesidades de los usuarios?

El **objeto de estudio** lo constituyen todos los procesos de recuperación y visualización de la información.

El **campo de acción** del trabajo abarcaría todas las mejoras posibles en los procesos de informatización de la recuperación y visualización de la información que se tramita en el MINFAR.

Como **Hipótesis** se parte de la idea de que, si se contara con la modelación de una solución de software para la gestión de procesos de recuperación y visualización de información de forma dinámica y a través de listados, tablas o gráficos, se logrará la implementación de una aplicación que jugará un papel fundamental en el análisis de la información.

Con el propósito de dar solución al problema anteriormente planteado se ha trazado como **objetivo:** Modelar un sistema informático para la ayuda en los procesos de recuperación de la información en las bases de datos del MINFAR.

De acuerdo a esta propuesta se derivan los siguientes **objetivos específicos:**

- ✓ Realizar un estudio sobre formas y mecanismos de recuperación de información.
- ✓ Analizar y Diseñar una aplicación Web que permita visualizar a través de listados, tablas o gráficos la información recuperada dinámicamente.

Para cumplir con el objetivo y resolver la situación problemática planteada, se proponen las siguientes **tareas:**

- ✓ Investigación de las formas y mecanismos de la representación de la información.
- ✓ Investigación de las principales tecnologías y herramientas que se utilizan para llevar a cabo sistemas como el que queremos desarrollar.
- ✓ Selección de la metodología de Análisis y Diseño de sistemas informáticos, que facilite la creación y garantice la calidad del sistema.
- ✓ Aplicación de la metodología, con todos sus flujos de trabajo en el proceso de desarrollo del sistema.
- ✓ Diseño de una base de datos que soporte las funcionalidades del sistema.

Este documento esta estructurado en 4 capítulos, en los que se expone todo el proceso de desarrollo de este trabajo:

Capítulo 1: Se describen conceptos relacionados con la representación de información, se hace referencia a sistemas existentes en el mundo que realizan acciones similares. Se mencionan los principales problemas que llevaron a la necesidad del cambio en los procesos de recuperación de información del MINFAR y por tanto al desarrollo de este trabajo. Se hace, además, un análisis de las tecnologías a utilizar en el desarrollo del sistema.

Capítulo 2: Se describe el negocio a través de un modelo del dominio y se definen las principales funcionalidades describiéndolas y representándolas en un diagrama de casos de uso del sistema.

Capítulo 3: Aborda aspectos relacionados con la construcción de la solución, se modelan los diagramas de clases de análisis y de diseño, se plantea además el modelo de datos.

Capítulo 4: Se realiza el diagrama de despliegue para describir la distribución física del sistema. Se modela la implementación, se realizan los diagramas de componentes y se diseñan los casos de prueba.

CAPÍTULO 1 FUNDAMENTACIÓN TEÓRICA

1.1 Introducción

En el presente capítulo se abordan de forma general aspectos relacionados con la recuperación y representación de la información dinámicamente. Se hace una valoración de softwares similares existentes en el mundo.

Como elemento necesario se aborda el tema del lenguaje de programación para la Web que se utilizará para la implementación del sistema. Además se analiza la metodología a utilizar para el análisis y diseño del sistema, se propone la tecnología y herramientas a utilizar para el desarrollo de la aplicación.

1.2 Recuperación de la información

El valor de cualquier sistema de información está condicionado por la calidad y cantidad de información contenida en él. El proceso conocido como Recuperación de Información trata los aspectos relacionados con el almacenamiento, organización y representación de la información, algoritmos de búsqueda, y acceso a la misma.

La recuperación de información es el conjunto de tareas mediante las cuales el usuario localiza y accede a los recursos de información que son pertinentes para la solución del problema planteado.

En principio, la recuperación de información engloba las acciones encaminadas a identificar, seleccionar y acceder a los recursos de información útiles al usuario. [1]

La recuperación de información esta muy ligada al tipo de información que se desea recuperar. La recuperación por listados posee gran valor porque nos permite realizar recuperaciones de información de un amplio volumen, esta consiste en mostrar los elementos o datos recuperados conformando un listado de estos, y de esta forma poder satisfacer las necesidades informativas de los usuarios. Pero para este tipo de recuperación no seria conveniente la sola representación de números porque, lejos de claridad en la visualización, se provocaría un agotamiento visual en el usuario y muy poca comprensión de lo que se representa.

La información estadística juega un papel fundamental en todos los ámbitos de la vida diaria: en la educación, la salud, el deporte, comercio, economía etc., la representación de este tipo de información de la forma adecuada nos ayuda a comprender mejor los datos a los que se accede. Habitualmente se

observa la información estadística representada en tablas numéricas o gráficos que permiten facilitar la comprensión de la misma o resaltar datos relevantes.

Para que las estadísticas sean de utilidad en cuanto a la caracterización de los fenómenos y al conocimiento de la realidad, deben cumplir determinados requisitos, siendo el principal el de veracidad. Asimismo, los datos deben ser conceptualmente significativos, es decir, obtenidos a partir de definiciones previamente establecidas. Pero otro requisito que no se puede olvidar es la calidad de la representación de este tipo de información, y en la mayoría de los casos esta información deberá ser representada en tablas o gráficos estadísticos para su mejor comprensión.

Estadísticas: Es la rama de las Matemáticas que se va a encargar de Recopilar, Organizar, y Procesar datos con el fin de inferir las características de la población objetivo. [2]

La imagen a continuación muestra información estadística representada a través de una tabla:

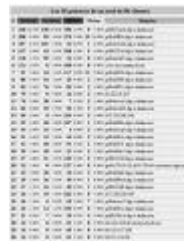
The image shows a screenshot of a data table with multiple columns and rows. The text is small and difficult to read, but it appears to be a list of numerical data points organized in a structured format, likely representing statistical information as mentioned in the caption.

Figura 1.1: Estadísticas a través de tablas.

La llegada de los ordenadores y de programas para la generación de gráficos ha puesto en manos del usuario común una poderosa herramienta. Los gráficos son imágenes que, combinando la utilización de sombreado, colores, puntos, líneas, símbolos, números, texto y un sistema de referencia (coordenadas), permiten presentar información. [3]

Estos resultan ser muy buenos para organizar y sintetizar la información, y de esta forma comunicar los aspectos e ideas esenciales. La utilización de un gráfico puede ayudarnos a representar de un modo más eficiente nuestros datos.

Una ventaja de tablas y gráficos es que, esta forma de representar la información permite realizar comparaciones de los datos representados, sin provocar una saturación visual del usuario, siendo en ocasiones los medios más efectivos para describir y resumir información.

Los gráficos además tienen una función estética que facilita atraer la atención del usuario y convencerlo del mensaje que se quiere transmitir.

Existen varios tipos de gráficos. Cada tipo de gráfico se diferencia de los demás por la clase de marcas de datos que utiliza. El motivo de que haya tantos tipos de gráficos diferentes no es solamente estético, cada uno de los tipos de gráficos está especialmente indicado para representar los datos de una manera distinta. Para obtener la máxima eficacia al crear los gráficos y presentar los datos de la mejor manera posible se debe tener en cuenta; que cada tipo de grafico esta destinado para una labor especifica.

Los gráficos pueden ser: de barras (bar), de líneas (lines), de puntos (points), de líneas y puntos (linepoints), de pastel (pie), y de área (area).

Estos gráficos se muestran a continuación:

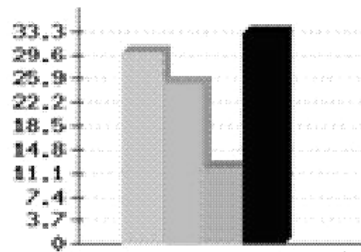


Figura 1.2: Gráfico de barras

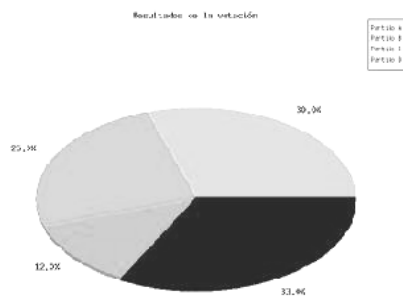


Figura 1.3: Gráfico de pastel

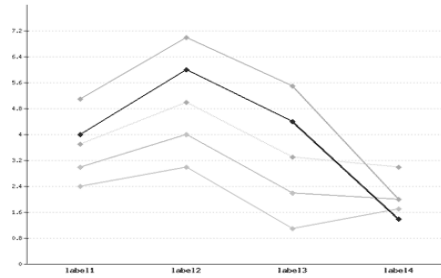


Figura 1.4: Gráficos de líneas, puntos y líneas-puntos.

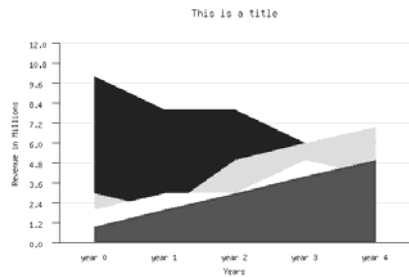


Figura 1.5: Gráfico de área.

1.2.1 Teoría de grafos y algoritmos de grafos.

Gran parte del dinamismo que tendrá como característica el sistema que se desea construir estará relacionado con la teoría de grafos y la implementación de algoritmos relacionados con estos.

En matemáticas y ciencias de la computación, la teoría de grafos estudia las propiedades de los grafos, estos son colecciones de objetos llamados vértices (o nodos) conectados por líneas llamadas aristas (o arcos) que pueden tener orientación (dirección asignada). Típicamente, un grafo está diseñado por una serie de puntos (los vértices) conectados por líneas (las aristas).

Existen diferentes formas de almacenar grafos en una computadora. La estructura de datos usada depende de las características del grafo y el algoritmo usado para manipularlo. Teóricamente se pueden distinguir las estructuras de listas y las de matrices, pero usualmente, lo mejor es una combinación de ambas. Las listas son preferidas en grafos dispersos porque tienen un eficiente uso de la memoria. Por otro lado, las matrices proveen acceso rápido, pero pueden consumir grandes cantidades de memoria.

Un grafo es una pareja $G = (V, A)$, donde V es un conjunto de puntos, llamados vértices, y A es un conjunto de pares de vértices, llamadas aristas.

Los grafos pueden ser: grafos simples, grafos conexos, grafos completos, árboles, grafos ponderados.

Existe gran cantidad de algoritmos para el recorrido de un grafo y la selección de uno o más, está en dependencia del tipo de grafo a que nos enfrentamos y al objetivo final que perseguimos con este recorrido

Los algoritmos con los que se propone que trabaje el sistema para la recuperación de información de manera dinámica son el Dijkstra y el Grasp.

El algoritmo de Dijkstra, también llamado algoritmo de caminos mínimos, resuelve el problema de encontrar desde un nodo origen fijado, el camino mínimo desde dicho nodo a cualquier otro dentro del conjunto V de vértices de un grafo G .

Este algoritmo resultara de gran utilidad para el trabajo que deseamos realizar puesto que permitirá generar un nuevo grafo con las siguientes características:

- Grafo acíclico, conexo y no direccional.
- Los vértices representan las relaciones (tablas).
- Las aristas representan las descripciones de la mínima cantidad de tablas que hay que recorrer entre cada par de nodos. [4]

A cada arco se le asigna un valor que estará dado por la suma de los valores de las aristas involucradas en el camino mínimo obtenido para cada par de nodos.

A partir del nuevo grafo obtenido y teniendo en cuenta los nodos obligatorios que se deben recorrer, aplicamos un algoritmo Grasp (gride rando adapter search procedure) para obtener el camino mínimo.

Los algoritmos GRASP son algoritmos de tipo iterativo en los que cada iteración incluye una fase de construcción de una solución y otra de postprocesamiento en la cual se optimiza la solución generada en la primera fase. [5]

1.3 Objeto de estudio

1.3.1 Objetivos estratégicos de la organización

El Ministerio de las Fuerzas Armadas Revolucionarias (MINFAR), constituye la institución militar básica del estado cubano, este juega el papel fundamental en la protección y desarrollo económico de nuestro

país. Debido a esto, debe mantener un liderazgo nacional en el campo de la tecnología, por lo que se ha marcado como objetivo la informatización de los principales procesos que en esta institución se realizan. Dada la gran cantidad de información que se maneja y la importancia de realizar un análisis óptimo de esta, son de gran importancia los procesos que permiten la recuperación de la información de forma dinámica.

1.3.2 Flujo actual de los procesos

Actualmente en todas las entidades pertenecientes al MINFAR existe gran cantidad de información almacenada en formato digital, así como copia dura (papeles). Cuando se realiza un pedido de un reporte en el cual se hace necesario representar información ya sea por listados, tablas o gráficos. Los especialistas encargados de realizar el mismo, analizan la información, la procesan y luego con la ayuda de alguna herramienta automatizada (generalmente se utilizan los componentes de office para la Web o sistemas elaborados por estos especialistas que dan una solución particular a su problema), elaboran el reporte a partir de los datos obtenidos.

1.3.3 Análisis crítico de la ejecución de los procesos

En estos momentos los procesos de recuperación de información se desarrollan de forma independiente, los sistemas no poseen una herramienta común para la obtención y análisis de la información, es decir, la búsqueda se hace teniendo en cuenta las opiniones primarias del usuario y dependiendo del criterio del especialista informático encargado de la materia a recuperar. Esto trae consigo que no se cumpla, en muchas de las ocasiones, las necesidades informativas de los usuarios.

El flujo de información que se maneja es muy grande y cambiante por lo que surgen nuevas necesidades informativas constantemente. Los procesos existentes para recuperar la información no se encuentran preparados para darle respuesta a las nuevas necesidades y poseen un nivel de dinamismo demasiado bajo puesto que se encuentran limitados a un sistema con un análisis y diseño obsoletos que no permite cambios; por tanto se hace muy engorroso todo el mecanismo que se lleva a cabo para satisfacer los nuevos requerimientos produciéndose un retraso en la recuperación de la información lo que obstaculiza el proceso de análisis de la información necesitada.

Existe también gran cantidad de información almacenada que posee un carácter confidencial, por lo que la recuperación de este tipo de datos es una tarea que va más allá de las actividades propias de los archiveros y documentalistas. (Este gran volumen de información influye en gran medida en la toma de decisiones, por parte de los jefes de la institución por lo que acceder a ella de forma eficiente es vital para el sistema.)

Todo esto trae como consecuencia la necesidad de un sistema dinámico que permita la presentación de información de forma más clara y eficaz, que permita facilitar su comprensión y además un análisis flexible de la misma.

1.4 Sistemas automatizados existentes vinculados al campo de acción.

Los gestores de información han dejado de ser simples herramientas operativas para convertirse en verdaderos aliados de las organizaciones en etapas de modernización, contribuyendo esencialmente en el análisis de la información.

A nivel mundial existen diversos gestores para la recuperación de la información. Algunos de estos se listan a continuación:

- Microsoft Query: Es un programa que permite incorporar datos de orígenes externos a otros programas de Microsoft Office, especialmente a Microsoft Excel. Si utiliza Query para recuperar datos de las bases de datos y de los archivos corporativos, no es necesario que vuelva a escribir en Excel los datos que desee analizar. También puede actualizar los informes y resúmenes de Excel automáticamente de la base de datos de origen inicial siempre que la base de datos se actualice con información nueva. Es posible recuperar datos de varios tipos de bases de datos, incluidos Microsoft Access, Microsoft SQL Server y los servicios OLAP de Microsoft SQL Server. También puede recuperar datos de las listas de Excel y de archivos de texto. [6]
- Cristal Report: Es el generador de reporte por excelencia de Visual Basic desde versiones anteriores a .NET. Esta no es la primera vez que se distribuye una versión de este generador de reporte junto a una versión de Visual Studio; en la versión 4 de Visual Basic se incluía una versión de Crystal Reports, aunque no tan integrada como la versión .NET. Cristal es un producto creado

en su esencia orientado al usuario final; es decir que un ejecutivo pueda crear sus propios informes sin necesidad de asistencia de un desarrollador. [7]

- Agata Report: es un generador de reportes multi-plataforma, una herramienta de consulta y generación gráficos como el Crystal Reports que se conecta a Bases de Datos como PostgreSQL, MySQL, Oracle, DB2, MS-SQL, Informix, InterBase, Sybase, o Frontbase y permite exportar los reportes en formatos como PostScript, plain text, HTML, XML, PDF o CSV (StarCalc, Excel). Permite definir niveles de datos, subtotales y totales. Permite crear documentos, como cartas y conjugar dinámicamente con los datos provenientes del reporte, así como crear etiquetas de direccionamiento y hasta generar un diagrama ER completo a partir de su banco de datos. [8]
- Sistema de reportes OLAP: Con cualquiera de los sistemas OLAP, se encuentran dos grupos principales de usuarios finales: los creadores de cubos y los usuarios de los cubos. Con los reportes OLAP, los creadores de cubos pueden ser personas que tengan acceso a los datos, los limpien y generen cubos y gráficos de reporte, o personas que construyan modelos, que hagan las mismas tareas anteriores, y adicionalmente modelen y hagan data mining. Los generadores de cubos usan SPSS Básico, dependiendo de las necesidades analíticas. Los dos productos generan un “reporte” que contiene cubos de reporte, gráficos, texto y otra información que esta disponible a los usuarios de los cubos. [9]

Estos gestores son sin dudas realmente efectivos, pero presentan inconvenientes que dificultan su utilización. La mayoría de estos sistemas tienen como desventajas que son propietarios para los cuales hay que pagar grandes sumas de dinero en licencias para su uso, están basados en tecnologías muy avanzadas y no compatibles con las que se utilizan en el proyecto MINFAR, y ninguno de ellos se adapta completamente a las necesidades de los usuarios. Algunos de estos gestores como es el caso de Agata Report, presentan una interfaz complicada, cargada de operaciones para los que se necesita mucha preparación para utilizarlos. Para trabajar con el Microsoft Query generalmente es necesaria la intervención de un especialista que conforme las consultas bases, que permitan recuperar la información necesaria, siendo esto uno de los principales aspectos que se debe tratar de eliminar.

Después de analizar varios gestores de información, concluimos que se hace necesaria la modelación de una herramienta de software dinámica que pueda recuperar toda la información existente en las bases

de datos del MINFAR y representarlas a través de listados, tablas o gráficos, además efectuar búsquedas eficientes y exactas.

1.5 Tendencias y Tecnologías Actuales

1.5.1 Aplicaciones Web

La plataforma Web posee una arquitectura Cliente/Servidor. Un usuario, utilizando un navegador, realiza una petición de una página al servidor Web a través de la red. El servidor recibe dicha petición, la procesa y envía la respuesta al navegador o cliente (en formato HTML), el cual la recibe y se desconecta.

Las aplicaciones Web constituyen un caso más complejo de esta arquitectura, permitiendo manejar lógica de negocio. Éstas se ejecutan en el servidor, encargándose de controlar el estado de dicho negocio y de gestionar los datos almacenados con ayuda de algún Sistema Gestor de Bases de Datos. [10]

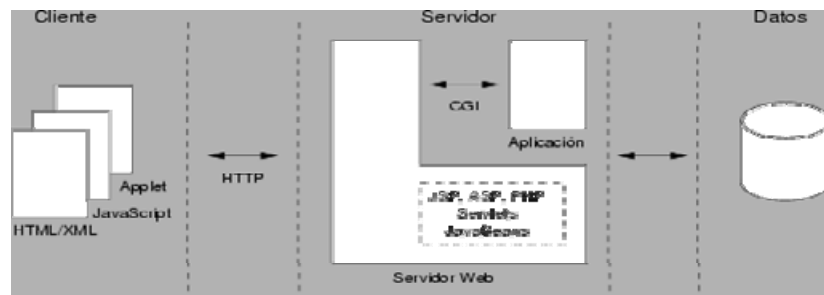


Figura 1.6: Arquitectura Web

Con la introducción de Internet y del Web en concreto, se han abierto infinitas posibilidades en cuanto al acceso a la información desde casi cualquier sitio. Esto representa un desafío a los desarrolladores de aplicaciones, ya que los avances en tecnología demandan cada vez aplicaciones más rápidas, ligeras y robustas que permitan utilizar el Web.

Actualmente el uso de aplicaciones Web se ha extendido dado que brindan grandes posibilidades, los clientes sólo necesitan un navegador, que interpreta el código con formato HTML, todo esto evita tener que instalar componentes de software adicionales para hacer uso de ellas. Además, permiten el aprovechamiento de todas las características de Internet. Son fáciles de usar, pues no requieren conocimientos avanzados de computación y su disponibilidad es alta ya que pueden realizar consultas en cualquier parte del mundo donde se tenga acceso a Internet.

Todo esto constituyen ventajas que influyen en que el sistema a desarrollar sea un aplicación Web.

1.5.2 Arquitectura n capas

Las aplicaciones en n capas se han convertido en el modelo para el desarrollo del software empresarial actual. Para la mayoría de usuarios, una aplicación en n capas es algo dividido en distintas partes lógicas. Las aplicaciones en n capas surgieron por primera vez como una forma de resolver algunos de los problemas asociados a las aplicaciones cliente/servidor tradicionales, pero con la llegada de la Web, esta arquitectura ha llegado a dominar el nuevo desarrollo.

Dividir una aplicación en partes lógicas resulta útil debido a que cuando dividimos software de gran tamaño en partes más pequeñas este será más fácil de generar, reutilizar y modificar. La opción más habitual está formada por una división en tres partes (presentación, lógica de negocio y datos), aunque existen otras posibilidades.

- Capa de presentación: La capa de presentación contiene los componentes necesarios para habilitar la interacción del usuario con la aplicación. Esta capa presenta al usuario la interfaz generada en el servidor, de forma tal que éste pueda trabajar con ella.
- Capa lógica del negocio: El comportamiento de la aplicación es definido por los componentes que modelan la lógica de negocio. Estos componentes reciben las acciones a realizar a través de la capa de presentación, y llevan a cabo las tareas necesarias utilizando la capa de datos para manipular la información del sistema. Los servicios de esta capa son encapsulados en dos paquetes, las Entidades del Negocio, que representan objetos que van a ser manejados o consumidos por toda la aplicación, y Lógica del Negocio que contienen las clases principales relacionadas con dicho negocio.
- Capa de acceso a datos: Tiene como misión el almacenamiento, la actualización y la consulta de todos los datos contenidos en el sistema, es por esto que contiene a la base de datos y las clases de acceso a la misma.

El estilo n capas se define como una organización jerárquica tal que cada capa proporciona servicios a la capa inmediatamente superior y se sirve de las prestaciones que le brinda la inmediatamente inferior.

Las ventajas del estilo en capas son obvias. Primero que nada, el estilo soporta un diseño basado en niveles de abstracción crecientes, lo cual a su vez permite a los implementadores la partición de un problema complejo en una secuencia de pasos incrementales. En segundo lugar, el estilo admite muy naturalmente optimizaciones y refinamientos. En tercer lugar, proporciona amplia reutilización.

1.5.3 Software Libre

Si bien hasta hace muy pocos años la informática era una gran desconocida para la mayor parte de las personas, en pocos años se ha convertido en un requisito indispensable para la casi totalidad de los puestos de trabajo de oficina, donde la utilización del software libre ha tomado gran importancia porque con el mismo disminuyen los problemas económicos de la actualización y mejora del software, ya que todas las versiones posteriores y mejoradas del software que se eligió son automáticamente puestas a disposición de todos los posibles usuarios.

El Software Libre es un tipo particular de software que le permite al usuario el ejercicio de cuatro libertades básicas:

1. Ejecutarlo con cualquier propósito
2. Estudiar como funciona y adaptarlo a sus necesidades
3. Distribuir copias
4. Mejorarlo, y liberar esas mejoras al público

Con la única restricción del copyleft (o sea, cualquiera que redistribuya el software, con o sin cambios, debe dar las mismas libertades que antes), y con el requisito de permitir el acceso al código fuente (imprescindible para ejercer las libertades 1 y 3). [11]

El software libre, incluso en los casos en que se comercializa, siempre es más barato que sus equivalentes propietarios, y casi siempre acaba estando disponible gratuitamente.

1.5.4 Lenguajes de programación para la Web

Los lenguajes de programación para la Web se clasifican en dos tipos en dependencia de donde se implementan respecto a la arquitectura cliente servidor, estos son los llamados lenguajes del lado del

cliente y lenguajes del lado del servidor. Del primer tipo se pueden señalar: Visual Basic Script y Javascript y, siendo el primero un lenguaje de programación de scripts del lado del cliente, pero sólo compatible con Internet Explorer. Es por ello que su utilización está discordante a favor de Javascript.

- Lenguaje del lado del cliente

JavaScript es un lenguaje interpretado, es decir, no requiere compilación, utilizado principalmente en páginas web. Todos los navegadores interpretan su código integrado dentro de las páginas web. Gracias a su compatibilidad con la mayoría de los navegadores modernos, es el lenguaje de programación del lado del cliente más utilizado.

Entre las acciones típicas que se pueden realizar en Javascript se tiene, por un lado, los efectos especiales sobre páginas web, para crear contenidos dinámicos y elementos de la página que tengan movimiento, cambien de color o cualquier otro dinamismo. Por el otro, javascript nos permite ejecutar instrucciones como respuesta a las acciones del usuario, con lo que podemos crear páginas interactivas con programas como calculadoras, agendas, o tablas de cálculo.

Javascript es un lenguaje con muchas posibilidades, permite la programación de pequeños scripts, pero también de programas más grandes, orientados a objetos, con funciones, estructuras de datos complejas, etc. Además, Javascript pone a disposición del programador todos los elementos que forman la página web, para que éste pueda acceder a ellos y modificarlos dinámicamente. [12]

- XML (EXtensible Markup Language):

XML es una tecnología en realidad muy sencilla que tiene a su alrededor otras tecnologías que la complementan y la hacen mucho más grande y con posibilidades mucho mayores.

XML, con todas las tecnologías relacionadas, representa una manera distinta de hacer las cosas, más avanzada, cuya principal novedad consiste en permitir compartir los datos con los que se trabaja a todos los niveles, por todas las aplicaciones y soportes. El XML juega un papel fundamental en este mundo actual, que tiende a la globalización y la compatibilidad entre los sistemas, ya que es la tecnología que permitirá compartir la información de una manera segura, fiable y fácil. Además permite al programador y los soportes dedicar sus esfuerzos a las tareas importantes cuando trabaja con los datos, ya que algunas tareas tediosas como la validación de estos o el recorrido de las estructuras corre a cargo del lenguaje y está especificado por el estándar, de modo que el programador no tiene que preocuparse por ello.

- Lenguajes del lado del servidor

PHP (PHP Hypertext Pre-processor) es un lenguaje de programación gratuito e independiente de plataforma, con una gran librería de funciones y mucha documentación. Resulta muy útil para diseñar de forma rápida y eficaz aplicaciones Web dirigidas a bases de datos. Su interpretación y ejecución se da en el servidor, en el cual se encuentra almacenado el script, y el cliente sólo recibe el resultado de la ejecución. Una de sus características más potentes es su soporte para gran cantidad de bases de datos. Por ejemplo: InterBase, MySQL, Oracle, PostgreSQL, entre otras. PHP también ofrece la integración con las varias bibliotecas externas, que permiten que el desarrollador haga varias cosas, desde generar documentos en pdf hasta analizar código XML.

Ventajas de PHP

1. Es un lenguaje multiplataforma.
2. Capacidad de conexión con la mayoría de los manejadores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL
3. Leer y manipular datos desde diversas fuentes, incluyendo datos que pueden ingresar los usuarios desde formularios HTML.
4. Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.
5. Permite las técnicas de Programación Orientada a Objetos.
6. Biblioteca nativa de funciones sumamente amplia e incluida
7. No requiere definición de tipos de variables ni manejo detallado del bajo nivel.

Desventajas de PHP

- 4.1 Todo el trabajo lo realiza el servidor. Por tanto puede ser más ineficiente a medida que las solicitudes aumenten de número.
- 4.2 La legibilidad del código puede verse afectada al mezclar sentencias HTML y PHP.
- 4.3 La orientación a objetos es aún muy deficiente para aplicaciones grandes. [13]

- **PHPlot**

Proponemos que en el sistema que se realice a partir de nuestro trabajo se utilice la librería phplot para mostrar los reportes gráficos en las recuperaciones gráficas.

No hace mucho tiempo en PHP, faltaban soluciones que permitieran la creación de gráficos de forma sencilla, gracias a la realización de PHPlot se cuenta con una librería que permite crear gráficos de apariencia profesional y con la ayuda de pocas líneas de código.

La librería PHPlot no es una librería revolucionaria su código completo se basa en la conocida librería GD, pero la utilización de GD implicaría largas horas de pruebas y errores y, en definitiva, tener que escribir centenares de líneas de código. PHPlot oculta toda la complejidad de la librería GD, ofreciendo a los usuarios únicamente algunas decenas de funciones entendibles y sencillas que permiten crear gráficos que no se alejan de la apariencia y posibilidades de éstas.

Los dibujos que son creados con PHPlot pueden estar compuestos por varios gráficos y cada uno de estos por varios elementos, es importante saber diferenciar entre un dibujo y un gráfico. Un dibujo es un área en la que puede haber uno o más gráficos

Las funciones de esta librería trabajan con el sistema de coordenadas cartesianas con el origen en el punto medio de la intersección de los ejes Y y X, haciendo uso de las unidades características para el conjunto de datos presentados en el gráfico. Estas funciones también pueden utilizar la unidad de resolución de la pantalla (píxeles); en este caso, el punto inicial del sistema de coordenadas tiene su origen en la esquina superior izquierda de la pantalla.

Actualmente la librería PHPlot se encuentra en desarrollo aunque se esta desarrollando de manera bastante rápida, con el fin de agregar el trabajo con otros tipos de gráficos, su reciente aparición trae consigo la desventaja de que solo se encuentren pocos tutoriales para su uso en la red, pero su gran facilidad de uso es una significativa ventaja así como también lo es, la calidad de los gráficos que permite representar.

1.5.5 Navegador

Un navegador web o browser es una aplicación software que permite al usuario recuperar y visualizar documentos de hipertexto, comúnmente descritos en HTML, desde servidores web de todo el mundo a

través de Internet. La funcionalidad básica de un navegador web es permitir la visualización de documentos de texto, posiblemente con recursos multimedia incrustados.

- Mozilla Firefox

Es un navegador de código abierto, multiplataforma, con interfaz gráfica de usuario desarrollado por la Mozilla Corporation y cientos de voluntarios. Con más de 25 millones de descargas en los 99 días siguientes a la liberación de la v1.0, Firefox se convirtió en una de las aplicaciones libres más descargadas, especialmente entre los usuarios domésticos. Firefox incorpora bloqueo de ventanas emergentes, navegación por pestañas, marcadores dinámicos, soporte para estándares abiertos, y un mecanismo para añadir funcionalidades mediante extensiones. Aunque otros navegadores también incluyen estas características, Firefox fue el primero en incluir algunas de ellas y conseguir una amplia difusión.

1.5.6 Metodología de Ingeniería del software

- Proceso Unificado de desarrollo (RUP):

El Proceso Racional Unificado o RUP (Rational Unified Process), es un proceso de desarrollo de software y junto con el Lenguaje Unificado de Modelado UML, constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos. El RUP es un producto de Rational (IBM). Incluye artefactos (que son los productos tangibles del proceso como por ejemplo, el modelo de casos de uso, el código fuente, etc.) y roles (papel que desempeña una persona en un determinado momento, una persona puede desempeñar distintos roles a lo largo del proceso).

El RUP divide el proceso de desarrollo en ciclos, teniendo un producto final al terminar cada ciclo, estos se dividen en fases (inicio, elaboración, construcción y transición) que finalizan con un hito donde se debe tomar una decisión importante:

Se caracteriza por ser:

Dirigido por casos de uso: Los casos de uso reflejan lo que los usuarios futuros necesitan y desean, lo cual se capta cuando se modela el negocio y se representa a través de los requerimientos. A partir de

aquí los casos de uso guían el proceso de desarrollo ya que los modelos que se obtienen, como resultado de los diferentes flujos de trabajo, representan la realización de los casos de uso (cómo se llevan a cabo).

Centrado en la arquitectura: La arquitectura muestra la visión común del sistema completo en la que el equipo de proyecto y los usuarios deben estar de acuerdo, por lo que describe los elementos del modelo que son más importantes para su construcción, los cimientos del sistema que son necesarios como base para comprenderlo, desarrollarlo y producirlo económicamente.

Iterativo e Incremental: RUP propone que cada fase se desarrolle en iteraciones. Una iteración involucra actividades de todos los flujos de trabajo, aunque desarrolla fundamentalmente algunos más que otros. Por ejemplo, una iteración de elaboración centra su atención en el análisis y diseño, aunque refina los requerimientos y obtiene un producto con un determinado nivel, pero que irá creciendo incrementalmente en cada iteración.

- Lenguaje Unificado de Modelado. (UML)

Por sus siglas en inglés, *Unified Modeling Language*) es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad; Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema de software. UML ofrece un estándar para describir un "plano" del sistema (modelo), incluyendo aspectos conceptuales tales como procesos de negocios y funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes de software reutilizables. UML es un "lenguaje" para especificar y no un método o un proceso.

1.5.7 Herramientas CASE

Las herramientas CASE (**C**omputer **A**ided **S**oftware **E**ngineering, Ingeniería de Software Asistida por Ordenador) son diversas aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de software reduciendo el coste de las mismas en términos de tiempo y de dinero. Estas herramientas pueden ayudar a ingenieros, desarrolladores en todos los aspectos del ciclo de vida de desarrollo del software en tareas como el proceso de realizar un diseño del proyecto, calculo de costes, implementación de parte del código automáticamente con el diseño dado, compilación automática, documentación o detección de errores, entre otras.

- Visual Paradigm

Es una herramienta Case para UML, de fácil de uso y completa, con soporte multiplataforma, posibilita una alta interoperabilidad con otras aplicaciones. La herramienta ayuda al equipo del desarrollo del software a maximizar y acelerar el desarrollo del software y contribuciones individuales. Apoya un gran número de idiomas en la generación de código y la ingeniería inversa en Java, C++, PHP, entre otros.

Tiene la capacidad de crear el esquema de clases a partir de una base de datos y crear la definición de base de datos a partir del esquema de clases. Permite invertir código fuente de programas, archivos ejecutables y binarios en modelos UML al momento, creando de forma simple toda la documentación. Está diseñada para usuarios interesados en sistemas de software de gran escala con el uso de la aproximación orientado a objeto. Incorpora el soporte para trabajo en equipo, que permite que varios desarrolladores trabajen a la vez en el mismo diagrama y vean en tiempo real los cambios hechos por sus compañeros de equipo.

1.5.8 Bases de Datos

Colección o depósito de datos integrados con redundancia controlada y con una estructura que refleje las interrelaciones y restricciones existentes en el mundo real; los datos, que han de ser compartidos por diferentes usuarios y aplicaciones, deben mantenerse independientes de éstas, y su definición y descripción, únicas para cada tipo de datos, han de estar almacenadas junto con los mismos. Los procedimientos de actualización y recuperación comunes y bien determinadas, habrán de ser capaces de conservar la integridad, seguridad y confidencialidad del conjunto de los datos. [14]

1.5.9 Sistema Gestor de Bases de Datos (SGBD)

Es un conjunto de herramientas que proporciona a administradores, analistas, programadores, usuarios los medios necesarios para describir, recuperar y manipular los datos almacenados en la BD, manteniendo la seguridad, integridad y confidencialidad de los mismos.

Sus principales funciones son:

1. Definir los datos.
2. Manipular los datos.

3. Garantizar la seguridad e integridad de los datos.
4. La gestión de las transacciones y el acceso concurrente.
5. Evitar la redundancia de los datos, eliminando así la inconsistencia de los mismos.
6. Mejorar la eficacia de acceso a los datos, en especial en el caso de consultas inesperadas.

Algunos ejemplos de los SGBD más utilizados son: SQL Server, My SQL, Postgres SQL, Oracle, entre otras.

- PostgreSQL

Postgres es un motor de base de datos, es servidor de base de datos relacional libre, liberado bajo la licencia BSD.

Este gestor de bases de datos brinda un almacenamiento confiable, consistente, robusto. Es Multiplataforma. Existen varias herramientas gráficas de alta calidad para administrar las bases de datos (pgAdmin, pgAccess). Manipulación potente, flexible, eficiente de datos. Diseñado para ambientes de alto volumen. Tiene un buen soporte para triggers, vistas, procedimientos almacenados en el servidor, además tiene ciertas características orientadas a objetos.

¿Qué características tiene PostgreSQL que lo hace nuestra elección?

PostgreSQL ofrece muchas ventajas para una compañía o negocio, respecto a otros sistemas de bases de datos, en cuanto a:

Instalación ilimitada

Es frecuente que las bases de datos comerciales sean instaladas en más servidores de lo que permite la licencia. Algunos proveedores comerciales consideran a esto la principal fuente de incumplimiento de licencia. Con PostgreSQL, nadie puede demandarlo por violar acuerdos de licencia, puesto que no hay costo asociado a la licencia del software.

Esto trae consigo ventajas adicionales como:

- Modelos de negocios más rentables con instalaciones a gran escala.

- No existe la posibilidad de ser auditado para verificar cumplimiento de licencia en ningún momento.
- Flexibilidad para hacer investigación y desarrollo sin necesidad de incurrir en costos adicionales de licenciamiento.

Estabilidad y confiabilidad legendarias

En contraste a muchos sistemas de bases de datos comerciales, es extremadamente común que compañías reporten que PostgreSQL nunca ha presentado caídas en varios años de operación de alta actividad.

Extensible

El código fuente está disponible para todos sin costo. Si un equipo necesita extender o personalizar PostgreSQL de alguna manera, pueden hacerlo con un mínimo esfuerzo, sin costos adicionales. Esto es complementado por la comunidad de profesionales y entusiastas de PostgreSQL alrededor del mundo que también extienden PostgreSQL todos los días.

Multiplataforma

PostgreSQL está disponible en casi cualquier Unix (34 plataformas en la última versión estable), y una versión nativa de Windows está actualmente en estado beta de pruebas.

Diseñado para ambientes de alto volumen

PostgreSQL usa una estrategia de almacenamiento de filas llamada MVCC para conseguir una mejor respuesta en ambientes de grandes volúmenes. Los principales proveedores de sistemas de bases de datos comerciales usan también esta tecnología, por las mismas razones.

Herramientas gráficas de diseño y administración de bases de datos.

Existen varias herramientas gráficas de alta calidad para administrar las bases de datos (pgAdmin, pgAccess) y para hacer diseños de bases de datos (Tora, Data Architect).

Además PostgreSQL ofrece una serie de características técnicas, al igual que otros gestores, que permiten un mejor trabajo con las bases de datos.

Replicación (soluciones comerciales y no comerciales) que permiten la duplicación de bases de datos maestras en múltiples sitios de réplica.

1.5.10 Dreamweaver 8

Es la herramienta líder para el desarrollo de Web, permitiendo al usuario un diseño eficiente, desarrollo y mantenimiento de las páginas Web y aplicaciones Web.

Dreamweaver es sin duda el producto más aplaudido por miles de webmasters que día a día lo utilizan en su ambiente de trabajo. Por ello, esta nueva versión viene preparada con muchas novedades buscando apoyar las mejores prácticas del Web.

Dreamweaver 8 incluye herramientas para trabajar aplicaciones que manejan XML, así como mejoras a su manejo de hojas de estilo (CSS).

Del lado del XML, incluye interesantes herramientas visuales para incluir contenidos de este formato como son los feeds RSS, integrándolos fácilmente en sitios Web y aplicaciones.

Para el trabajo con CSS han simplificado la creación y manejo de diferentes estilos, promoviendo los estándares para nuevos usuarios y facilitando su aplicación para usuarios avanzados. [15]

1.6 Conclusiones

En este capítulo se trataron conceptos relacionados con la recuperación de información por listados, estadísticas y gráficos. Se realizó un estudio de las tecnologías a utilizar durante el desarrollo del software, sus ventajas, desventajas y características. Las herramientas que ayudarán a que el desarrollo del ciclo de vida del software se realice eficientemente, tiene como principales características que la mayoría son de un soporte multiplataforma y licencias de uso libre.

CAPÍTULO 2 CARACTERÍSTICAS DEL SISTEMA

2.1 Introducción

En este capítulo se hace la descripción de la propuesta que trae este trabajo, para ello se describen los procesos del negocio que tienen que ver con el objeto de estudio, de acuerdo a esto se llega a la conclusión que debido a la poca estructuración de esos procesos, para poder entender el contexto en que se emplaza el sistema, se necesita definir conceptos que podemos agrupar en un Modelo de Dominio para capturar correctamente los requisitos y poder construir un buen sistema.

Se enumeran los requisitos funcionales y no funcionales que debe tener el sistema que proponemos, lo que nos permite tener una idea general de las funcionalidades que tendrá este, e identificar mediante un Diagrama de Casos de Uso, las relaciones de los actores que interactúan con el sistema y las secuencias de acciones con las que interactúan.

2.2 Descripción de los procesos del negocio propuesto.

Actualmente en todas las entidades pertenecientes al MINFAR existe gran cantidad de información almacenada en formato digital y además en papeles. Cuando se realiza un pedido de un reporte en el cual se hace necesario representar datos en listados, estadísticos y gráficos. Los especialistas encargados de realizar el mismo, analizan la información, la procesan realizando cálculos, etc., y luego con la ayuda de alguna herramienta informatizada (generalmente se utiliza los componentes de office para la Web o sistemas elaborados por estos especialistas que dan una solución particular a su problema), elaboran los reportes para listados, estadísticos y gráficos a partir de los datos obtenidos.

No siempre se logran cubrir todas las necesidades informativas de los usuarios, y es obligatoria la intervención de un especialista para realizar la actividad que se desea desarrollar, esto trae como consecuencia la necesidad de un sistema dinámico que permita la representación los reportes para listados, estadísticos y gráficos personalizados por sus usuarios, que muestren la información de forma más clara y eficaz, que permita facilitar la comprensión representada, por terceros y además un análisis flexible de la misma.

2.3 Modelo del dominio

Gracias a la descripción de los procesos del negocio realizada anteriormente podemos llegar a la conclusión de que el negocio que se está estudiando tiene un bajo nivel de estructuración, y cuenta con soluciones muy diversas y dispersas a pesar de que todas tienen como fin la recuperación de información. Por lo que sería más adecuado realizar un modelo de dominio, el cual nos permite capturar los tipos más importantes de objetos que existen o los eventos que suceden en el entorno donde estará el sistema. El modelo de dominio brindará a clientes, usuarios, desarrolladores e interesados, un vocabulario común para comprender el contexto del negocio en que se enmarcará el sistema. Además servirá de ayuda en la captura correcta de los requisitos que debe tener el sistema.

2.3.1 Conceptos principales

- Entidad: Organización a la que pertenece el especialista.
- Especialista: Cualquier persona autorizada que trabaje o pertenezca a la institución que necesite y solicite, o no, la realización de un informe.
- Usuario Informativo: Persona que manipula la información almacenada en formato digital o en papel, que es un especialista y que satisface su necesidad o el pedido de un informe.
- Solicitante: Especialista que se auxilia de un usuario para obtener una recuperación de información.
- Componente Informático: Componente informático utilizado por el usuario informativo para generar un informe.
- Datos: Conjunto organizado e integrado de información almacenada y clasificada en computadora o papel para su posterior consulta, actualización o cualquier tarea de mantenimiento mediante operaciones específicas.
- Datos Digitales: Representan datos almacenados en formato digital.
- Datos Papel: Representan datos almacenados en papel.
- Informe: Documento digital o en papel que contiene toda la información necesitada.
- Informe Listado: Documento digital o en papel que contiene toda la información necesitada representada en un listado de los elementos recuperados.

- Informe Estadístico: Documento digital o en papel que contiene toda la información estadística necesitada representada en tablas numéricas.
- Informe Gráfico: Documento digital o en papel que contiene toda la información necesitada representada en un reporte gráfico.

2.3.2 Principales Eventos

- Pertenece: proceso que identifica al especialista especificando la entidad a la que pertenece.
- Solicita: proceso mediante el cual el Solicitante pide la recuperación de una información determinada al usuario informativo.
- Busca: proceso mediante el cual el Usuario Informativo manipula la información necesaria para dar respuesta al pedido de un Solicitante o para satisfacer su propia necesidad.
- Utiliza: proceso que permite el uso de un Componente Informático para generar un informe.
- Genera: proceso mediante el cual se genera el informe con la información especificada.
- Recibe: proceso de entrega del informe al especialista.

2.3.3 Diagrama de clases del dominio

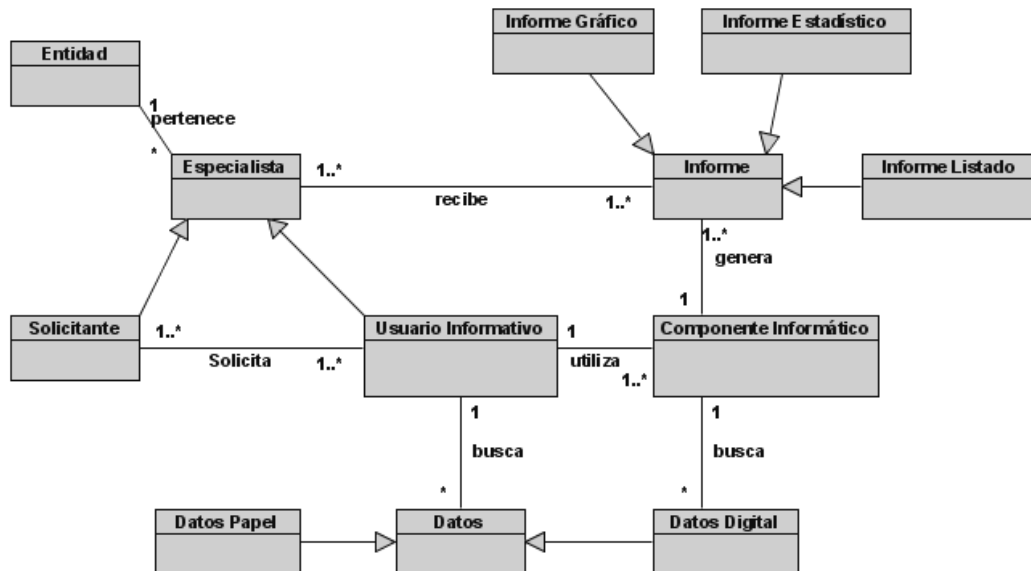


Figura 2.1: Modelo del dominio

2.4 Requerimientos funcionales

R1. Definir condiciones del informe

- 1.1. Agregar condiciones especificando los datos que formaran una condición: campo, condición, valor, opción
- 1.2. Modificar condiciones
- 1.3. Eliminar condiciones

R2. Definir formato del informe

- 2.1. Definir el título que se mostrara en el informe y el formato de este título (Tipo de letra, tamaño de la letra, color de la letra, alineación)
- 2.2. Definir formato del cuerpo del informe que se mostrará como resultado de la recuperación especificando: tipo de letra, tamaño de la letra, color de la letra, alineación

R3. Definir campos a mostrar en la recuperación por listado.

- 3.1. Agregar campos especificando la etiqueta con la que será mostrado el campo en el informe, el color de letra, el color de fondo y la alineación
- 3.2. Eliminar campo
- 3.3. Modificar campo
- 3.4. Permitir especificar el orden que se mostrarán los valores de un campo seleccionado (ascendente, descendente)
- 3.5. Permitir especificar como agrupar valores por campos en el informe.

R4. Definir Campos en una recuperación estadística.

- 4.1 Definir campo por encabezado
- 4.2 Definir campo por columna
- 4.3 Definir campos por filas
 - 4.3.1 Permitir seleccionar campos como fila
 - 4.3.2 Permitir eliminar campos filas
- 4.4 Calcular totales para campo de encabezado, campo de columna y campo de fila

R5. Definir campos en una recuperación por gráfico

5.1. Seleccionar el tipo de gráfico a visualizar

5.2. Definir campo de datos

5.1.1. Permitir seleccionar un campo como campo de datos

- El área de datos recoge los campos de datos que proporcionan los valores que deben resumirse en el gráfico

5.1.2. Permitir especificar el incremento del campo de datos para gráficos de tipo barra, líneas y áreas

- Se especificará el incremento de los valores en el eje Y (Ej.: 0; 5; 10; 15, el incremento será de 5)

5.1.3. Permitir especificar el título del eje Y para gráficos de tipo barra, línea y área.

5.3. Definir campos del Eje X

5.3.1. Permitir seleccionar hasta cuatro campos como campos del Eje X

5.3.2. Permitir eliminar campo del eje X seleccionado

5.3.3. Permitir especificar el título del Eje X para gráficos de tipo barra, línea y área

5.4. Definir campos de Leyenda

5.4.1. Permitir seleccionar varios campos como campos de Leyenda

5.4.2. Permitir eliminar el campo de leyenda seleccionado

R6. Personalizar el campo seleccionado en recuperaciones estadísticas y recuperaciones por gráfico.

6.1. Definir etiquetas.

6.2. Gestionar condiciones por etiquetas del campo.

R7. Visualizar Informe

7.1. Visualizar reporte por listado

7.2. Visualizar reporte por estadísticas

7.3. Visualizar reporte por gráficos

7.4. Permitir guardar la información visualizada en los formatos: .doc, .pdf, .xls, además un informe gráfico podrá ser guardado en formato jpg y png.

R8. Gestionar plantilla

- 8.1. Permitir que el recuperador salve como plantilla cualquier configuración de recuperación que haya realizado.
- 8.2. Permitir al recuperador abrir una plantilla almacenada con anterioridad.
- 8.3. Permitir eliminar alguna(s) de las plantillas almacenadas.

2.5 Requerimientos no funcionales

Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener. Debe pensarse en estas propiedades como las características que hacen al producto atractivo, usable, rápido o confiable.

Apariencia o interfaz externa:

- Debe ser amigable y de fácil comprensión en su funcionamiento permitiendo la utilización del sistema sin mucho entrenamiento.
- Empleo de colores gris, blanco, azul y rojo principalmente que son los definidos en los estándares en el proyecto.
- La interfaz de la aplicación debe ser lo más atractiva y clara posible para el usuario final.

Interfaces de Comunicaciones:

- El sistema debe ser capaz de comunicarse con bases de datos remotas alojadas en servidores de datos. Para lo cual debe emplear protocolos estándares de comunicación.

Mantenibilidad:

- El sistema debe ser susceptible a ampliaciones. Por tanto su diseño e implementación deberá realizarse atendiendo a que sea fácilmente mantenible, aplicando para su desarrollo las metodologías que para ello sean precisas.

Portabilidad:

- El sistema debe ser multiplataforma.

Legales:

- El sistema se basa en el manual de normas y principios establecidos por el MINFAR.

Usabilidad:

- El sistema podrá ser usado por cualquier persona que posea conocimientos mínimos de base de datos y conocimientos medios en el manejo de la computadora.
- El software tendrá siempre la posibilidad de ayuda disponible para cualquier tipo de usuario, lo que le permitirá un avance considerable en la explotación de la aplicación en todas sus funcionalidades.

Rendimiento:

- Respuestas al usuario en término menor a los 5 segundos.

Soporte:

- Se requiere un servidor de bases de datos con las siguientes características:
- Soporte para grandes volúmenes de datos y velocidad de procesamiento.
- Tiempo de respuesta rápido en accesos concurrentes.
- Por parte del cliente se requiere un navegador capaz de interpretar JavaScript.

Funcionalidad:

- Mínima cantidad de páginas para ejecutar todas las funciones posibles.

Implantación:

- Entregar toda la documentación asociada al proyecto.
- Organizar el adiestramiento de los usuarios.

Confiabilidad

- Garantía de un tratamiento adecuado de las excepciones y validación de las entradas del usuario

Seguridad:

- Garantizar que la información sea vista únicamente por quien tiene derecho a verla.
- Protección contra acciones no autorizadas o que puedan afectar la integridad de los datos.

Software

Un servidor:

- Plataforma del sistema operativo Windows Advancer Server 2000 o superior, o Linux kernel 2.4 o superior.
- Apache 2.0 o superior como servidor Web, con módulo PHP 5 disponible y debe estar configurado con la extensión pgsql incluida.
- PostgreSQL como Sistema Gestor de Base de Datos.

Clientes:

- Navegador Mozilla FireFox o Internet Explorer 5.5 o superior.
- Sistema operativo Linux o Windows 2000 profesional o superior.

Hardware:

Servidor:

- Tarjeta de red 100 Mbps.
- 512 Mb. de RAM o superior.
- 40 Gb. de disco duro o superior.
- Pentium IV a 2.4 GHz.

Clientes:

- Tarjeta de red 100 Mbps.
- Procesador Pentium o similar a 1 GHz.
- 128 Mb de memoria RAM o superior.

2.6 Modelo de casos de uso del sistema

El modelo de casos de uso del sistema sirve como acuerdo entre clientes y desarrolladores, y proporciona la entrada fundamental para el análisis, el diseño y las pruebas.

Un modelo de casos de uso del sistema incluye los casos de uso del sistema y los actores del sistema. Los requisitos funcionales se estructuran de forma natural mediante casos de uso.

En el modelo de casos de uso del sistema los actores representan terceros fuera del sistema que colaboran con este. Cada forma en la que los actores usan el sistema se representa como casos de uso. Los casos de uso son fragmento de funcionalidad que el sistema ofrece para aportar un resultado de valor para sus actores. Un caso de uso especifica una secuencia de acciones que el sistema debe llevar a cabo interactuando con sus actores, incluyendo alternativas dentro de la secuencia.

En el modelo de sistema que se ha definido se considera la existencia de un solo actor, que se le llamará con el nombre de Recuperador:

Actores	Justificación
Recuperador	Representa a una persona que tiene permisos para recuperar la información a la que tiene acceso y conformar un informe. Puede utilizar el sistema para recuperar información por listado, por estadística o por gráficos.

A continuación se presentan los casos de uso determinados para estructurar los requerimientos funcionales del sistema:

CU-1	Definir Condiciones del Informe
Actor	Recuperador.
Descripción	El recuperador desea definir las condiciones que se tomarán en cuenta para la recuperación.
Referencia	R1

CU-2	Definir Formato del Informe.
Actor	Recuperador.
Descripción	El recuperador define el formato que tendrá el informe (informe por listado, estadístico o gráfico) en el que se mostrará la información recuperada.

Referencia	R2
------------	----

CU-3	Visualizar Informe
Actor	Recuperador.
Descripción	El recuperador desea visualizar los reportes de su recuperación.
Referencia	R7

CU-4	Visualizar Informe por Listado
Actor	Recuperador.
Descripción	El recuperador especifica los campos que se van a mostrar en el informe de una recuperación por listado.
Referencia	R3, R7

CU-5	Visualizar Informe Estadístico
Actor	Recuperador.
Descripción	El recuperador especifica los campos que van a formar parte del informe de una recuperación por estadísticas.
Referencia	R4, R7

CU-6	Visualizar Informe Gráficos.
Actor	Recuperador.
Descripción	El recuperador especifica los campos que van a formar parte del informe de una recuperación por gráfico.
Referencia	R5,R7

CU-7	Generar Vista Previa
Actor	Recuperador.
Descripción	El recuperador accederá al tab vista previa con el fin de mostrar el informe
Referencia	R7

CU-8	Personalizar.
Actor	Recuperador.
Descripción	El recuperador podrá personalizar un campo definiendo diferentes condiciones para este
Referencia	R6

CU-9	Gestionar Plantilla
Actor	Recuperador.
Descripción	El Recuperador guardará como plantilla los procesos de recuperación de información que se hayan realizado y abrirá o eliminará procesos de recuperación de información previamente guardados.
Referencia	R9

2.6.1 Diagrama de Caso de Uso del Sistema

UML nos permite presentar el modelo de casos de usos en diagramas que contendrán la relación de los actores y los casos de uso del sistema, esto ayudará a realizar un análisis de lo representado desde diferentes puntos de vista y con diferentes propósitos.

A continuación se representa, en un diagrama de casos de uso del sistema, el actor definido y su interacción con los casos de uso de este sistema:

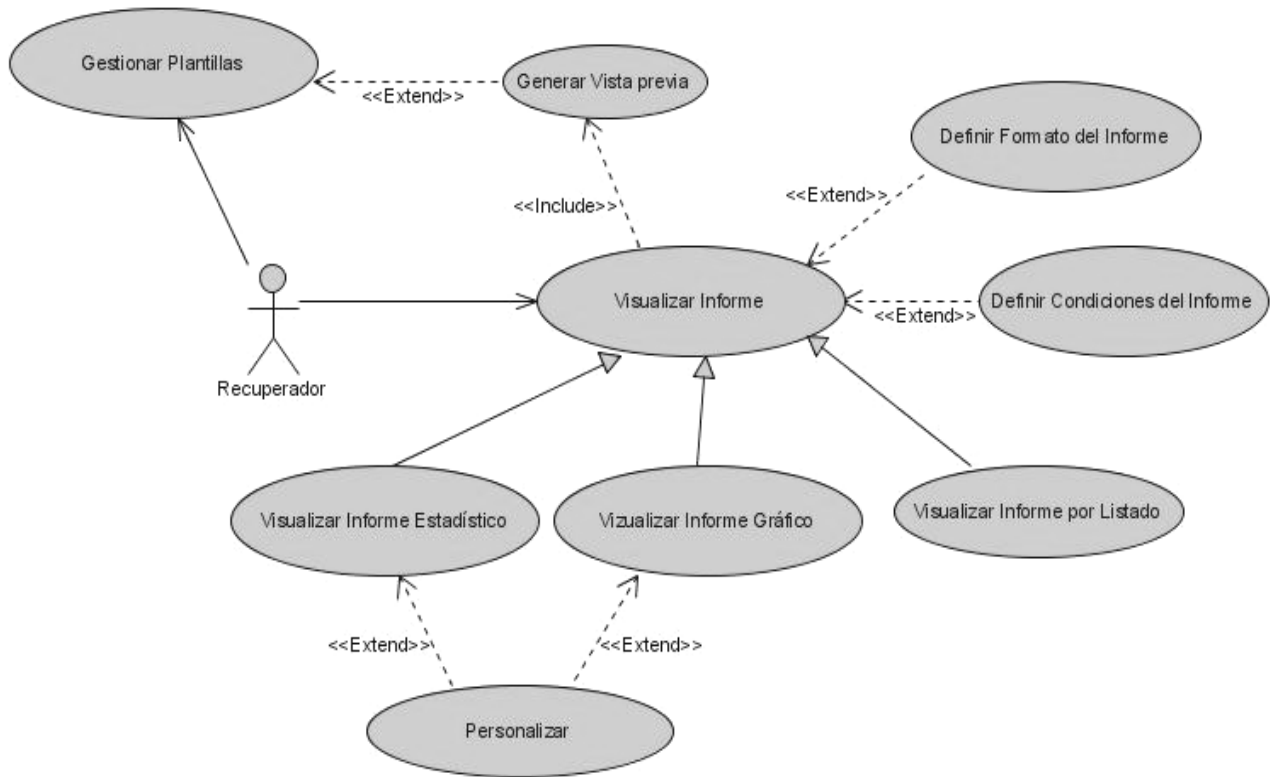
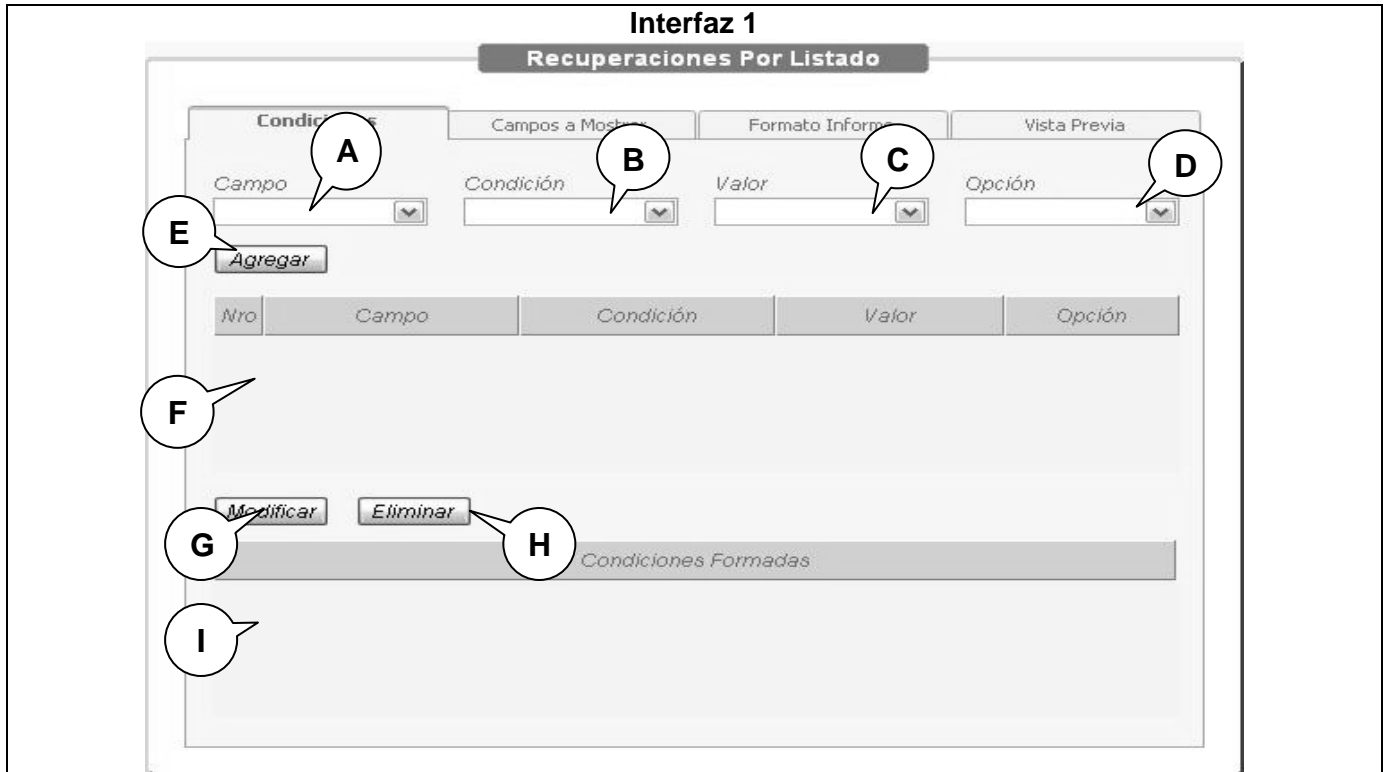


Figura 2.2: Diagrama de casos de uso del sistema

2.6.2 Expansión de los Casos de Uso

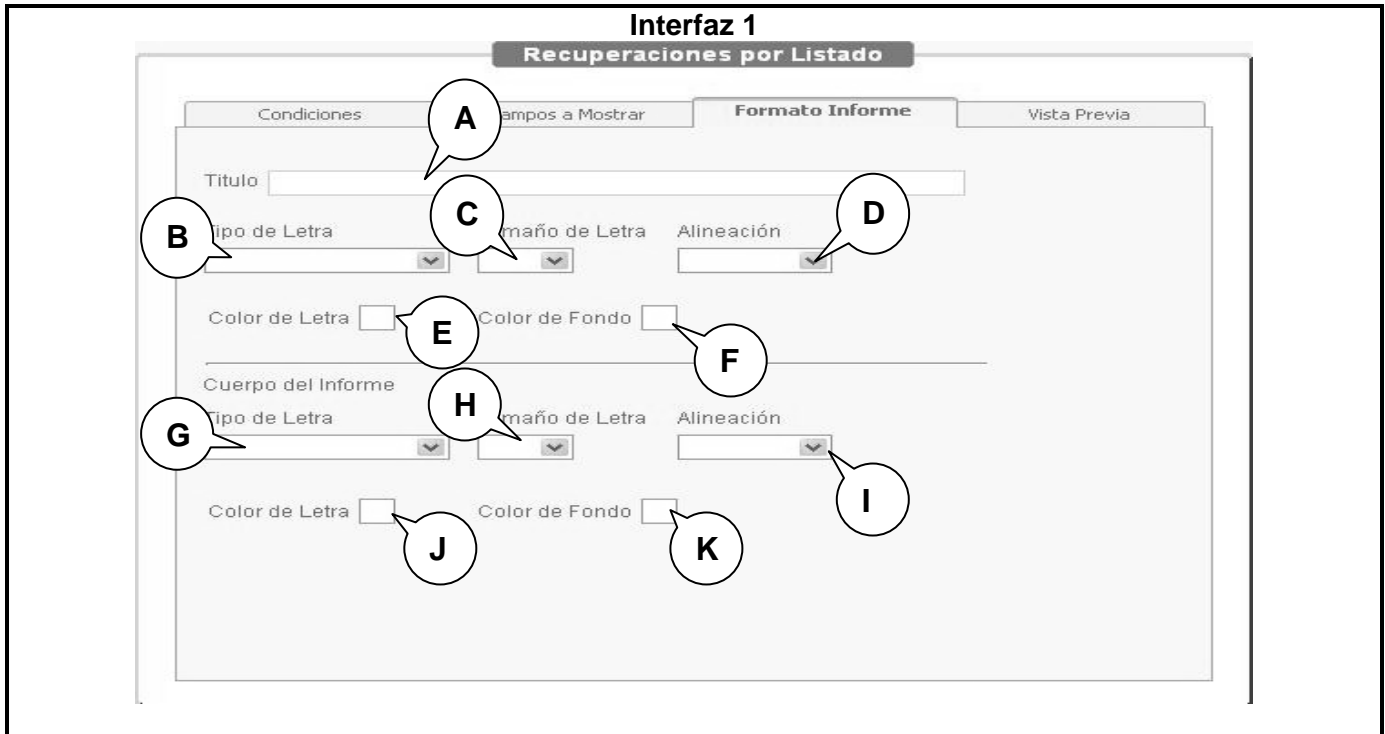
Caso de Uso:	Definir Condiciones del Informe
Actor(es):	Recuperador (inicia)
Propósito:	Definir las condiciones que cumplirá la recuperación de información que se va a realizar.
Resumen:	El caso de uso inicia cuando el recuperador desea definir las condiciones que tendrá su recuperación. El caso de uso finaliza cuando el recuperador termina la definición de condiciones y accede a otras opciones de la aplicación.
Referencias:	R1
Precondiciones:	Debe haber sido autenticado como recuperador. Deben existir definidos los campos a los que puede acceder el recuperador según su rol.
Requisitos especiales.	



Acción del Actor	Respuesta del Sistema
<p>1. Accede a la interfaz de Condiciones , ya sea por listado, por estadísticas, o por gráficos, para definir las condiciones que restringirán la recuperación de información</p> <p>3. Escoge el campo al cual le aplicará una condición (Interfaz 1-A)</p> <p>4. Selecciona la condición que se aplicará (Interfaz 1-B)</p> <p>5. Inserta el valor que tendrá esta condición (Interfaz 1-C)</p> <p>6. Escoge la opción que determine (Y, O, FIN) (Interfaz 1-D)</p> <p>7. Acciona el botón “Agregar” (Interfaz 1-E)</p>	<p>2. Se visualiza la interfaz de Condiciones con todas sus opciones, ya sea por listado, por estadística o por gráfico</p> <p>8. El sistema va mostrando las condiciones que se van creando en la parte inferior de la interfaz (1-F)</p>
Evento 1 “Modificar Condición”	
<p>1. Desea modificar una condición previamente</p>	

creada	
2. Selecciona la condición que desea modificar dando clic sobre esta.	
3. Acciona el botón “Modificar” (Interfaz 1-G)	4. Se muestran los requisitos de la condición a modificar (Interfaz 1-A, B, C, D)
5. Modifica la condición	
6. Acciona el botón “Agregar” (Interfaz 1-E)	7. Muestra en la interfaz la condición modificada
Evento 2 “Eliminar Condición”	
1. Desea eliminar una o varias condiciones que ha creado	
2. Escoge la(s) condición(es) que desea eliminar dando clic sobre estas (Interfaz 1-F)	
3. Acciona el botón “Eliminar” (Interfaz 1-H)	4. Muestra mensaje de alerta “Está seguro que desea eliminar la selección”
	5. Elimina la(s) condición(es) que seleccionó y actualiza la interfaz sin la(s) condición(es) eliminada(s)
Cursos Alternos	
Curso normal de eventos	
Línea 7: El sistema muestra mensaje de error porque ya existe otra condición igual, “Ya existe una condición igual a esta”.	
Evento “Modificar Condición”	
Línea 6: El sistema muestra mensaje de error porque la condición ya ha sido creada “Ya existe una condición igual a esta”.	
Evento “Eliminar Condición”	
Línea 4: El usuario responde negativamente y el sistema no elimina.	
Poscondiciones: Las condiciones del informe que cumplirán los campos de la recuperación que se va a realizar han sido creadas.	

Caso de Uso:	Definir formato del informe
Actor(es):	Recuperador (inicia)
Propósito:	Definir el formato del informe que se desea crear.
Resumen:	El caso de uso inicia cuando el recuperador requiere definir el formato que tendrá el informe que desea crear. El caso de uso termina cuando el recuperador define el formato y accede a otras opciones de la aplicación.
Referencias:	R2
Precondiciones:	Debe haber sido autenticado como recuperador.



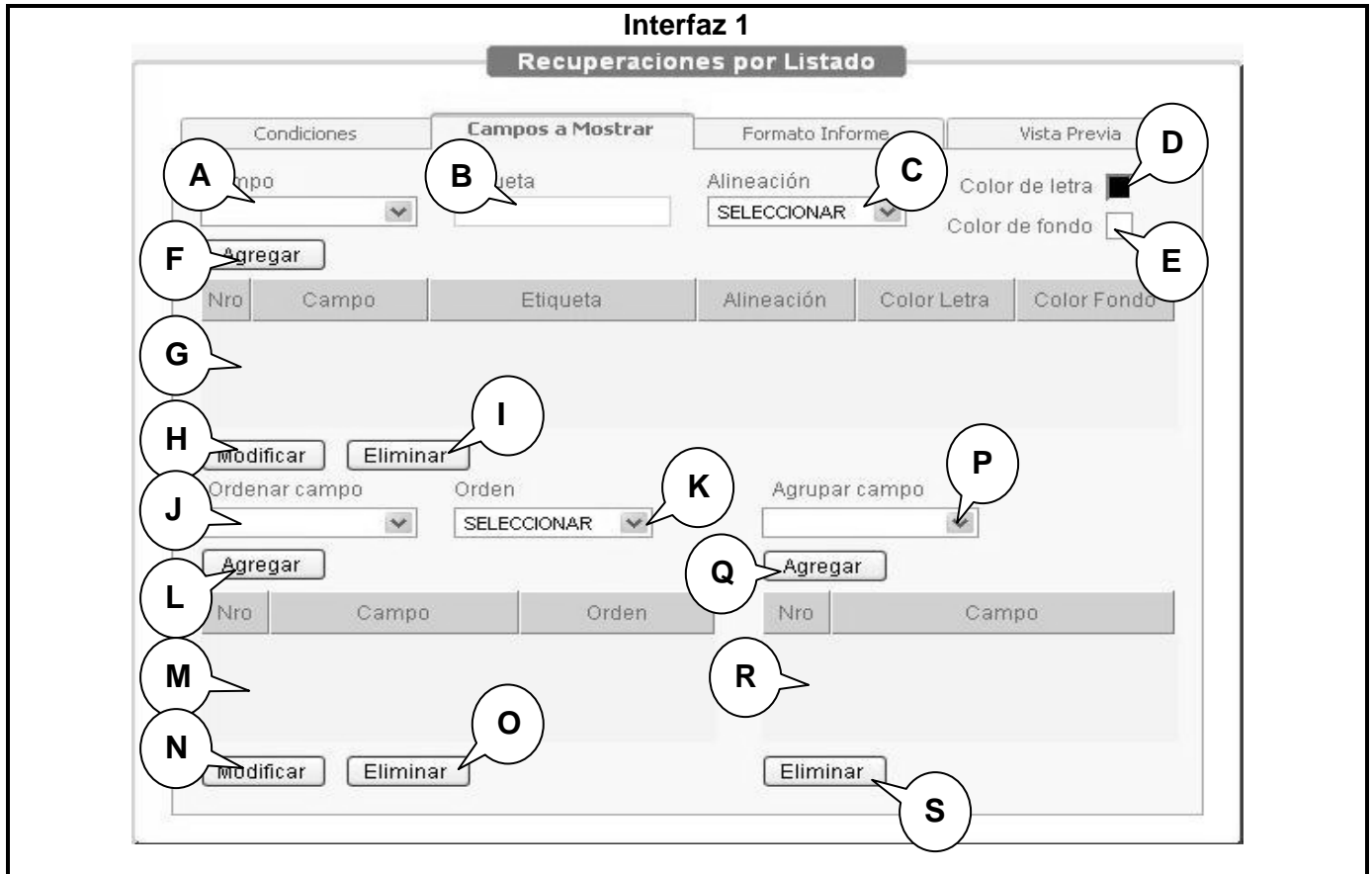
Acción del Actor	Respuesta del Sistema
<p>1. Accede a la interfaz Formato para definir el formato del informe (informe por listado, informe estadístico, informe gráfico) en el que se mostrará la recuperación</p> <p>3. Inserta el título que quiere mostrar en el informe de la recuperación (Interfaz 1-A)</p> <p>4. Selecciona la fuente, el color de fondo, el color de letra y la alineación que desea utilizar para el título del informe (Interfaz 1- B, C, D, E, F)</p> <p>5. Selecciona la fuente, el color de fondo, el color de letra y la alineación que desea utilizar para el cuerpo del informe (Interfaz 1-G, H, I, J, K)</p>	<p>2. Se visualiza la interfaz de Formato de la recuperación, ya sea por listado, por estadística o por gráfico (Interfaz 1)</p>

Cursos Alternos
<p>En el curso normal de eventos</p> <p>Línea 4: En caso de que no seleccione la fuente, el color de letra, el color de fondo y la alineación que desea utilizar para el título, el sistema utilizará el que se encuentra predeterminado.</p> <p>Línea 5: En caso de que no seleccione la fuente, el color de letra, el color de fondo y la alineación que desea utilizar para el cuerpo del informe, el sistema utilizará el que se encuentra predeterminado.</p>

Poscondiciones: El formato del informe quedará especificado.

Caso de Uso:	Visualizar Informe	
Actor(es):	Recuperador (inicia)	
Propósito:	Visualizar un informe por listado, por estadísticas o por gráfico	
Resumen:	El caso de uso inicia cuando el recuperador desea realizar una recuperación de información con el fin de visualizarla en un informe de listado, estadístico o gráfico. El caso de uso termina con la visualización del informe.	
Referencias:	R1, R2, R3, R4, R5, R7	
Precondiciones:	Debe haber sido autenticado como recuperador.	
Acción del Actor	Respuesta del Sistema	
Evento 1 “Configuración de la recuperación”		
1. Especifica las condiciones que restringirán la recuperación de información	2. Invoca al caso de uso “Definir condiciones del informe”	
3. Define el formato del informe	4. Invoca al caso de uso “Definir Formato del Informe”	
Evento 2 “Definición de campos de la recuperación”		
Evento 3 “Visualizar Información”		
1. Accede a la opción de “Vista Previa”	2. Invoca al caso de uso “Generar Vista previa”	
Cursos Alternos		
-		
Poscondiciones: La recuperación de la información por listado, por estadística, o por gráfico queda visualizada en un en un informe de listado, estadístico o gráfico respectivamente.		

Caso de Uso:	Visualizar informe por listado	
Actor(es):	Recuperador (inicia)	
Propósito:	Visualizar un informe por listado definiendo los campos que se mostrarán en él.	
Resumen:	El caso de uso inicia cuando el recuperador accede a definir los campos con el fin de mostrarlos en un informe por listado El caso de uso termina con la visualización de un informe por listado	
Referencias:	R3, R7	
Precondiciones:	Debe haber sido autenticado como recuperador. Deben existir definidos los campos a los que puede acceder el recuperador según su rol.	



Acción del Actor	Respuesta del Sistema
Evento 1 “Configuración de la recuperación”	
Evento 2 “Definir campos de la recuperación ”	
1. El Recuperador accede a la interfaz de Campos a Mostrar para definir los campos que formaran parte de esta recuperación	2. Se visualiza la interfaz de Campos a Mostrar de la recuperación por listado
3. Para agregar un campo, ir a Evento 2.1 “Agregar Campo”.	
Evento 2.1 “Agregar Campo”	
1. Escoge el campo que quiere mostrar en la recuperación (Interfaz 1-A)	
2. Inserta la etiqueta con que nombrará al campo (Interfaz 1-B)	
3. Selecciona la alineación que tendrá el campo en el informe de la recuperación	

<p>(Interfaz 1-C)</p> <p>4. Selecciona el color de letra (Interfaz 1-D)</p> <p>5. Selecciona el color de la fondo (Interfaz 1-E)</p> <p>6. Acciona el botón "Agregar" (Interfaz 1-F)</p> <p>8. Para ordenar campo y/o agrupar Campo, ir al Evento 2.4 "Ordenar Campos" y Evento 2.5 "Agrupar Campo" respectivamente.</p>	<p>7. Se muestran los campos seleccionados en la parte inferior de la interfaz (Interfaz 1-G).</p>
<p>Evento 2.2 "Modificar Campo"</p>	
<p>1. Desea modificar un campo de su recuperación</p> <p>2. Selecciona el campo que desea modificar dando clic sobre este(Interfaz 1-G)</p> <p>3. Acciona el botón "Modificar" (Interfaz 1-H)</p> <p>5. Modifica los datos (Interfaz 1-A,B,C)</p> <p>6. Acciona el botón "Agregar" (Interfaz 1-F)</p>	<p>4. Se muestran los datos del campo a modificar (Interfaz 1-A,B,C)</p> <p>7. Muestra en la interfaz el campo modificado (Interfaz 1-G)</p>
<p>Evento 2.3 "Eliminar Campo"</p>	
<p>1. Desea eliminar un(os) campo(s) seleccionados para mostrar en el informe de la recuperación.</p> <p>2. Selecciona el(los) campo(s) que desea eliminar dando clic sobre estos. (Interfaz 1-G)</p> <p>3. Acciona el botón "Eliminar" (Interfaz 1-I)</p>	<p>4. Muestra un mensaje de alerta "Esta seguro que desea eliminar el campo seleccionado"</p> <p>5. Elimina el(los) campos(s) que seleccionó y actualiza la interfaz sin el(los) campo(s) eliminado(s) (Interfaz 1-G)</p>
<p>Evento 2.4 "Ordenar Campos"</p>	
<p>1. Desea ordenar campo</p> <p>2. Selecciona el campo que desea ordenar(Interfaz 1-J)</p> <p>3. Selecciona ordenar los valores del campo</p>	

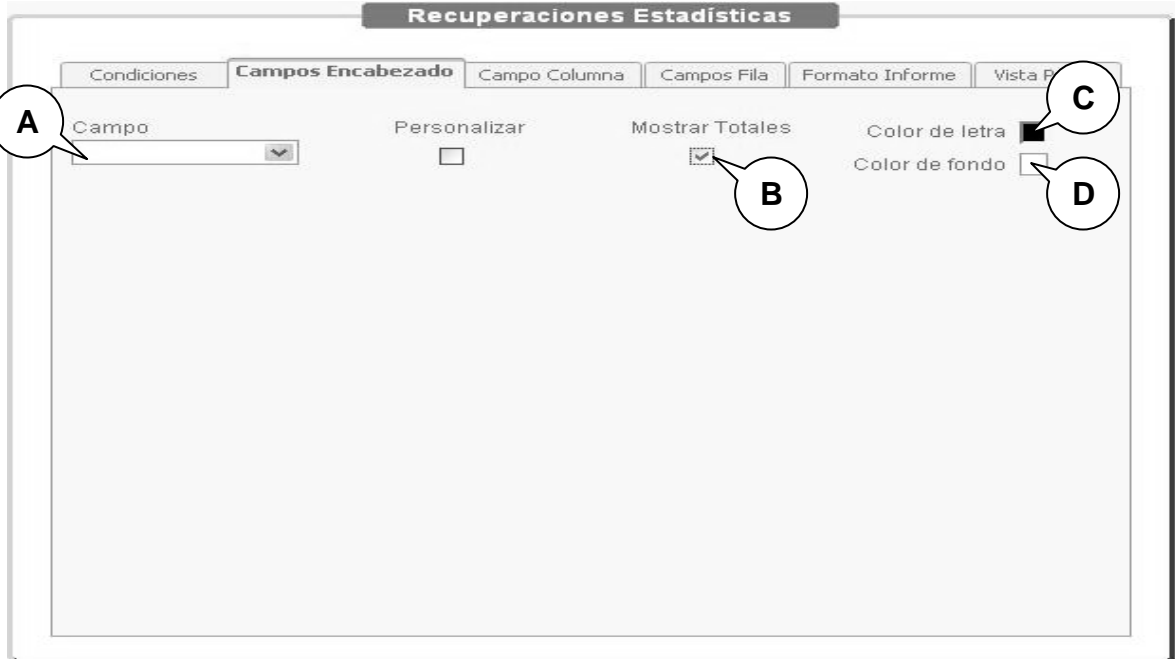
<p>seleccionado ascendente 0 descendentemente (Interfaz 1-K)</p> <p>4. Acciona el botón “Agregar” (Interfaz 1-L)</p>	<p>5. Se muestran los campos seleccionado con su orden respectivamente en la parte inferior de la interfaz (Interfaz 1-M).</p>
<p>Evento 2.4.1 “Modificar campo ordenado”</p>	
<p>1. Desea modificar el orden de un campo previamente seleccionado</p> <p>2. Selecciona el campo y el orden dando clic sobre este (Interfaz 1-M).</p> <p>3. Acciona el botón “Modificar” (Interfaz 1-N).</p> <p>5. Modifica el orden (Interfaz 1-K).</p> <p>6. Acciona el botón “Agregar” (Interfaz 1-L).</p>	<p>4. Se muestran el campo y el orden a modificar.</p> <p>7. Actualiza la interfaz con el campo y el nuevo orden (Interfaz 1-M).</p>
<p>Evento 2.4.2 “Eliminar campo ordenado”</p>	
<p>1. Desea eliminar el campo y el orden previamente establecidos</p> <p>2. Selecciona el campo y su orden dando clic sobre este (Interfaz 1-M).</p> <p>3. Acciona el botón “Eliminar” (Interfaz 1-O).</p>	<p>4. Muestra un mensaje de alerta “Esta seguro que desea eliminar el campo seleccionado”</p> <p>5. Muestra la interfaz actualizada sin los campos eliminados (Interfaz 1-M).</p>
<p>Evento 2.5 “Agrupar Campo”</p>	
<p>1. Desea agrupar campo</p> <p>2. Selecciona el campo que desea agrupar (Interfaz 1-P)</p> <p>3. Acciona el botón “Agregar” (Interfaz 1-Q)</p>	<p>4. Se muestran el campo seleccionado en la parte inferior de la interfaz (Interfaz 1-R).</p>
<p>Evento 2.5.1 “Eliminar Campo Agrupado”</p>	
<p>1. Desea eliminar el campo agrupado</p> <p>2. Selecciona el campo dando clic sobre este (Interfaz 1-R).</p>	

3. Acciona el botón “Eliminar” (Interfaz 1-S).	4. Muestra un mensaje de alerta “Esta seguro que desea eliminar el campo seleccionado” 5. Muestra la interfaz actualizada sin los campos eliminados (Interfaz 1-R).
Evento 3 “Visualizar información”	
Cursos Alternos	
Evento 2.1 “Agregar Campo” Línea 6: El sistema muestra mensaje de error porque ya el campo fue previamente seleccionado y/o porque ya existe otro campo con la misma etiqueta.	
Evento 2.2 “Modificar Campo” Línea 6: El sistema muestra mensaje de error porque ya el campo fue previamente seleccionado y/o porque ya existe otro campo con la misma etiqueta.	
Evento 2.3 “Eliminar Campo” Línea 4: Si el actor responde negativamente, la aplicación no elimina.	
Evento 2.4 “Ordenar campos” Línea 4: El sistema muestra un mensaje de error “Ya este campo ha sido agregado”	
Evento 2.4.1 “Modificar Campo Ordenado” Línea 6: El sistema muestra mensaje de error porque ya el campo fue previamente seleccionado con el mismo	
Evento 2.4.2 “Eliminar campo ordenado” Línea 4: Si el Recuperador responde negativamente el sistema no elimina.	
Evento 2.5 “Agrupar campos” Línea 3: El sistema muestra un mensaje de error “Ya este campo ha sido agregado”	
Evento 2.5.1 “Eliminar campo Agrupado” Línea 4: Si el Recuperador responde negativamente el sistema no elimina.	
Poscondiciones: Quedarán definidos los campos que se mostrarán en el informe por listado	

Caso de Uso:	Visualizar informe Estadístico
Actor(es):	Recuperador (inicia)
Propósito:	Visualizar un informe estadístico definiendo los campos que se mostrarán en él
Resumen:	El caso de uso inicia cuando el recuperador accede a definir los campos con el fin de mostrarlos en un informe estadístico El caso de uso termina con la visualización de un informe estadístico
Referencias:	R4, R7
Precondiciones:	Debe haber sido autenticado como recuperador. Deben existir definidos los campos a los que puede acceder el recuperador

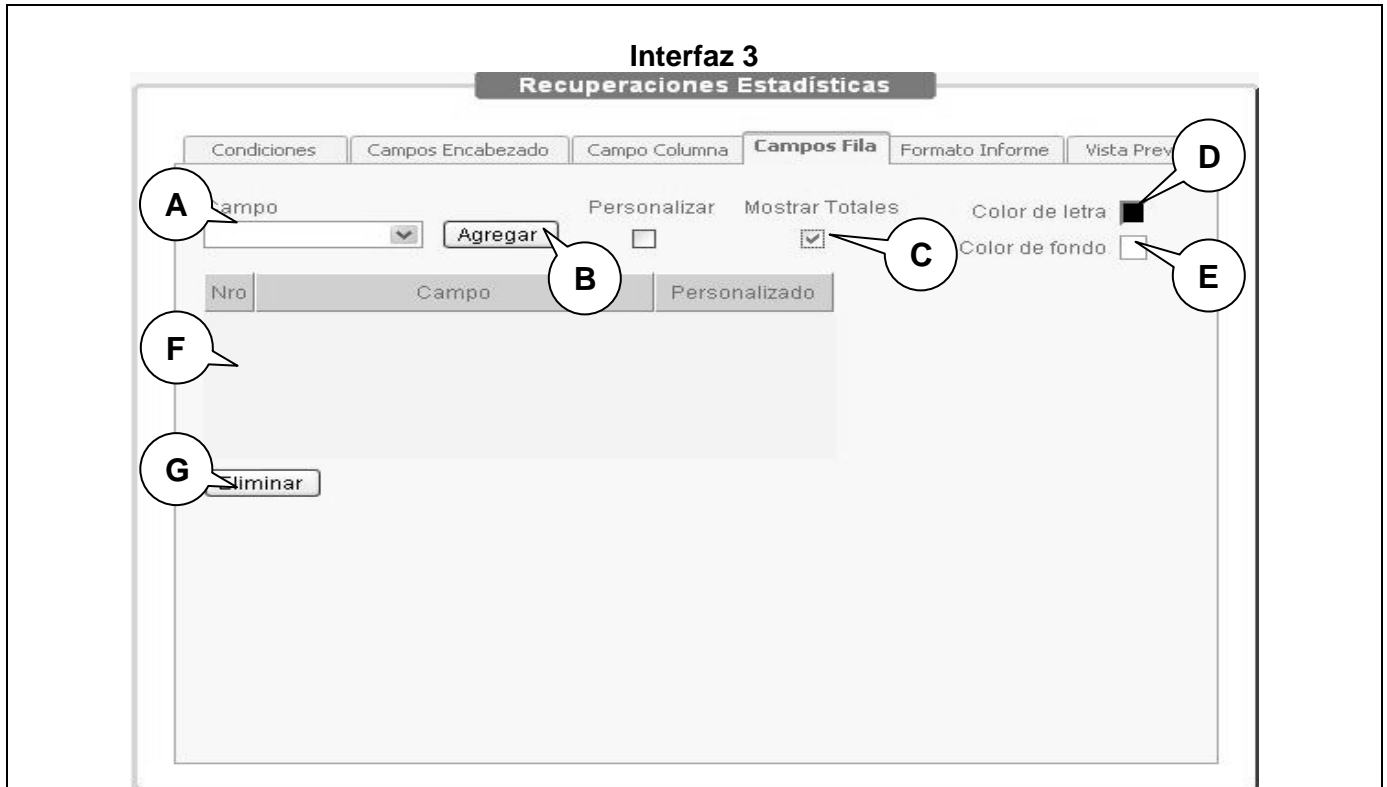
según su rol.

Interfaz 1



Interfaz 2





Acción del Actor	Respuesta del Sistema
Evento 1 “Configuración de la recuperación”	
Evento 2 “Definir campos de la recuperación”	
<p>1. El Recuperador accede a la interfaz de Recuperaciones por Estadística para definir los campos que formaran parte de esta recuperación</p> <p>3. Selecciona el tipo de campo que desea definir</p> <p>a) Si desea definir campo por encabezado ir al Evento 2.1 “Definir campo por encabezado”</p> <p>b) Si desea definir campo por columna ir al Evento 2.2 “Definir campo por columna”</p> <p>c) Si desea definir campos por fila ir al Evento 2.3 “Definir campos por filas”</p>	<p>2. Muestra la interfaz de recuperaciones por estadística con todas sus opciones.</p>

Evento 2.1 “Definir campo por encabezado”	
<p>1. Accede a la interfaz Campo Encabezado para definir el campo que se mostrará por encabezado</p> <p>3. Selecciona el campo que desea mostrar como encabezado en la recuperación (Interfaz 1-A)</p> <p>4. Si desea calcular totales para este campo ir a Evento 4 “Calcular totales”</p>	<p>2. Muestra la interfaz Campo Encabezado</p>
Evento 2.2 “Definir campo por columna”	
<p>1. Desea definir el campo de columna que se mostrará en la recuperación estadística</p> <p>3. Selecciona el campo que desea mostrar como columna en la recuperación (Interfaz 2-A)</p> <p>4. Si desea calcular totales para este campo ir a Evento 2.4 “Calcular totales”</p>	<p>2. Muestra la interfaz campo columna</p>
Evento 2.3 “Definir campos por fila”	
<p>1. Desea definir campo(s) a mostrar como filas</p> <p>3. Selecciona el(los) campo(s) que quiere mostrar (Interfaz 3-A)</p> <p>4. Acciona el botón “Agregar” (Interfaz 3-B)</p> <p>6. Si desea calcular totales para este campo ir a Evento 4 “Calcular totales”</p>	<p>2. Se visualiza la interfaz de campo fila</p> <p>5. Se muestran los campos seleccionados en la parte inferior de la interfaz. (Interfaz 3-F)</p>
Evento 2.3.1 “Eliminar campo de fila”	
<p>1. Desea eliminar un(os) campo(s) previamente seleccionado(s)</p> <p>2. Selecciona el(los) campo(s) que desea eliminar dando clic sobre estos (Interfaz 3-F)</p> <p>3. Acciona el botón “Eliminar” (Interfaz 3-G)</p>	<p>4. Muestra un mensaje de alerta “Esta seguro que desea eliminar el campo seleccionado”</p>

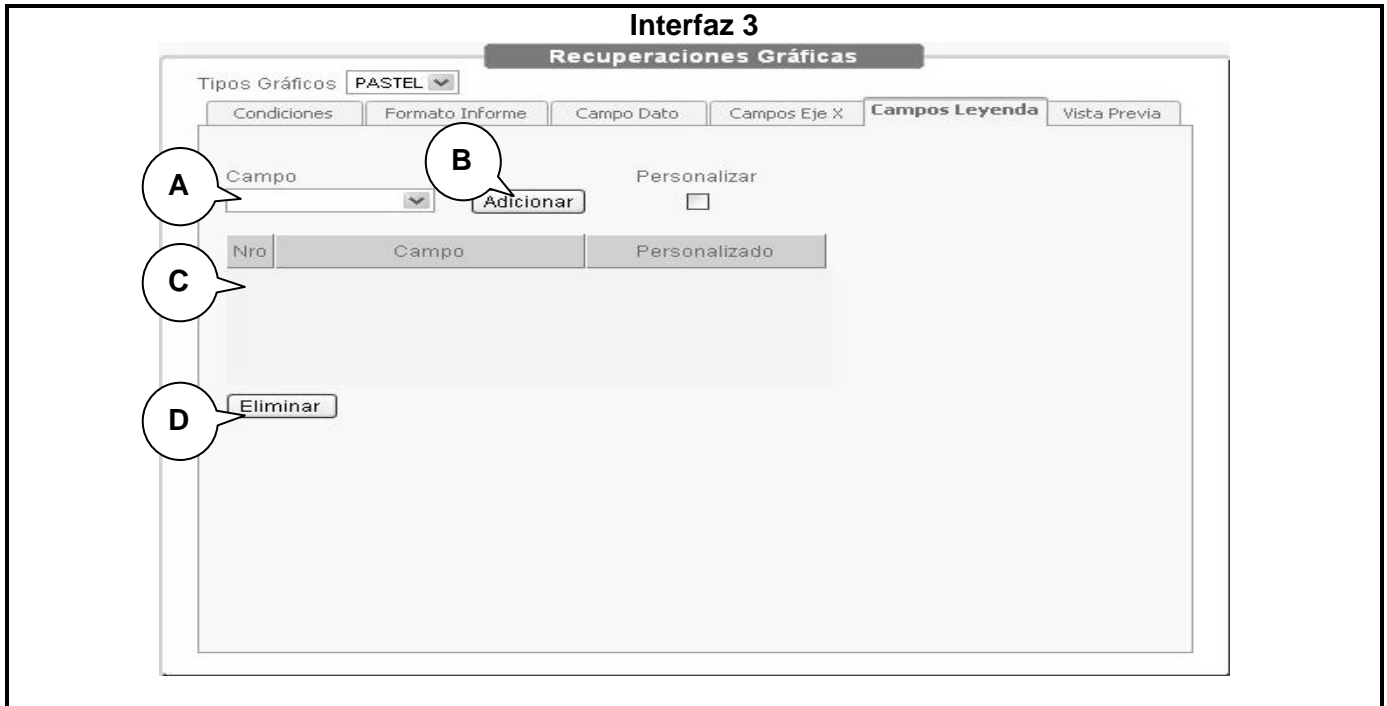
	<p>4. Elimina el(los) campos(s) que seleccionó y actualiza la interfaz sin el(los) campo(s) eliminado(s) (Interfaz 3-F)</p> <p>5.</p>
Evento 2.4 “Calcular totales”	
<p>1. Desea obtener totales</p> <p>2. Selecciona la opción “Totales” (Interfaz 3-B)</p> <p>6. Selecciona el color de letra y de fondo del total. (Interfaz 3-C, D).</p>	<p>3. Se activa en la interfaz la posibilidad de definir color de letra y color de fondo del total</p>
Evento 3 “Visualizar Información ”	
Cursos Alternos	
<p>Evento 2.1 “Definir campo por encabezado” Línea 3: El sistema muestra un mensaje de error porque este campo ha sido seleccionado anteriormente como campo de fila o campo de columna “Ya este campo ha sido seleccionado”</p> <p>Evento 2.2 “Definir campo por columna” Línea 3: El sistema muestra un mensaje de error porque este campo ha sido seleccionado anteriormente como campo de encabezado o como campo de fila “Ya este campo ha sido seleccionado”</p> <p>Evento 2.3 “Definir campos por fila” Línea 4: El sistema muestra un mensaje de error porque este campo ha sido seleccionado anteriormente como campo de encabezado, como campo de columna o como campo de fila “Ya este campo ha sido seleccionado”</p> <p>Evento 2.3.1 “Eliminar campo de fila” Línea 4: Si el recuperador responde negativamente el campo seleccionado no elimina</p> <p>Evento 2.4 “Calcular totales” Línea 2: Si el recuperador no desea mostrar los totales desactiva esta opción</p>	
Poscondiciones: Quedarán definidos los campos que se mostrarán en el informe estadístico.	

Caso de Uso:	Visualizar Informe Gráfico
Actor(es):	Recuperador (inicia)
Propósito:	Visualizar un informe gráfico definiendo los campos que se mostrarán en él
Resumen:	El caso de uso inicia cuando el recuperador accede a definir los campos con el fin de mostrarlos en un informe gráfico. El caso de uso termina con la visualización de un informe gráfico.

Referencias:	R5, R7
Precondiciones:	Debe haber sido autenticado como recuperador. Deben existir definidos los campos a los que puede acceder el recuperador según su rol.

Interfaz 1

Interfaz 2



Acción del Actor	Respuesta del Sistema
Evento 1 “Configuración de la recuperación”	
Evento 2 “Definir campos de la recuperación”	
<p>1. El Recuperador accede a la interfaz de Recuperaciones por gráficos para definir los campos que formaran parte de esta recuperación</p> <p>3. Selecciona el tipo de gráfico que desea representar (Interfaz 1-A)</p> <p>5. Selecciona el tipo de campo que desea definir</p> <p>a) Si desea definir campo de Datos ir al Evento 2.1 “Definir campo de datos”</p> <p>b) Si desea definir campos del Eje X ir al Evento 2.2 “Definir campos del Eje X”</p> <p>c) Si desea definir campos de Leyenda ir al Evento 2.3 “Definir campos de Leyenda”</p>	<p>2. Muestra la interfaz de recuperaciones por gráficos con todas sus opciones.</p> <p>4. Muestra los tab activado para cada tipo de gráfico (si es de barra, línea y area utilizan los tres tab y pastel solo los tab de campo dato y campos leyenda)</p>
Evento 2.1 “Definir campo de dato”	
1. Accede a la interfaz Campo Dato para	2. Muestra la interfaz Campo Dato

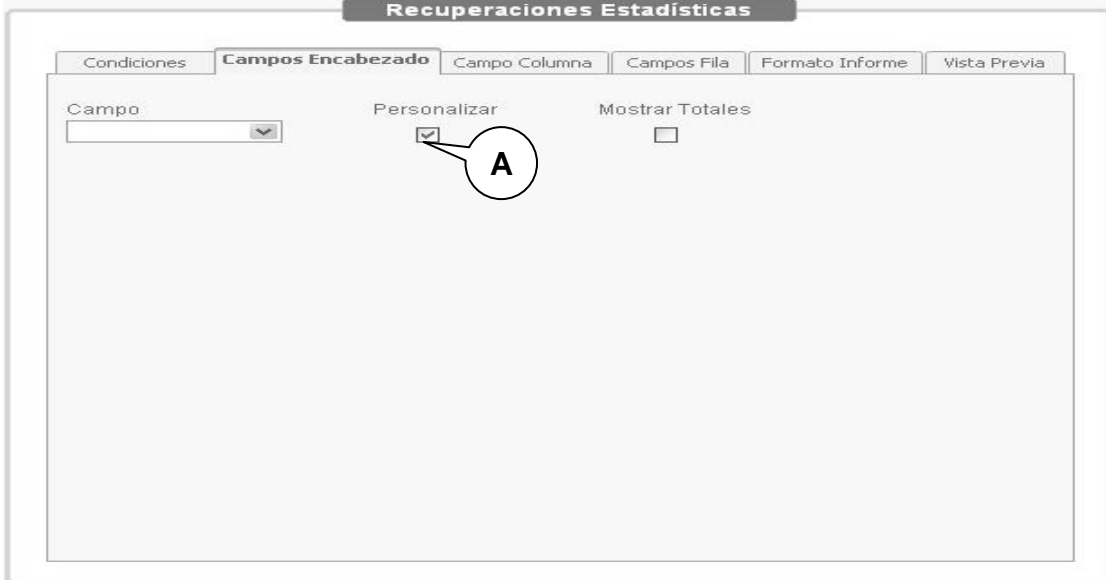
<p>definir el campo que se mostrará como campo de datos(en el caso de ser un grafico de línea, área o barra esta será el campo del Eje Y)</p> <p>3. Selecciona el campo que desea mostrar como campo de datos en la recuperación (Interfaz 1-B)</p> <p>4. Especifica el incremento del Eje y (gráficos líneas, área o barras) (Interfaz 1-C)</p> <p>5. Define el título que describe la información representada en el eje de coordenada Y. (Interfaz 1-D)</p>	
<p>Evento 2.2 “Definir campos del Eje X”</p>	
<p>1. Accede a la interfaz Campos del Eje X.</p> <p>3. Selecciona los campos que desea representar en el eje X. (Interfaz 2-A)</p> <p>4. Acciona el botón “Agregar” (Interfaz 2-B)</p> <p>6. Define el título que describe la información representada en el Eje X (Interfaz 2-C).</p>	<p>2. Muestra la interfaz Campos del Eje X con todas sus opciones. (Interfaz 2)</p> <p>5. Muestra los campos seleccionados en la pantalla. (Interfaz 2-D)</p>
<p>Evento 2.2.1 “Eliminar campos del Eje X”</p>	
<p>1. Desea eliminar un(os) campo(s) previamente seleccionado(s)</p> <p>2. Selecciona el(los) campo(s) que desea eliminar dando clic sobre estos (Interfaz 2-D).</p> <p>3. Acciona el botón “Eliminar” (Interfaz 2-E)</p>	<p>4. Muestra un mensaje de alerta “Esta seguro que desea eliminar el campo seleccionado”</p> <p>5. Elimina el(los) campos(s) que seleccionó y actualiza la interfaz sin el(los) campo(s) eliminado(s) (Interfaz 2-D)</p>
<p>Evento 2.3 “Definir campos de Leyenda”</p>	
<p>1. Accede a la interfaz campos de Leyenda</p> <p>2. Selecciona los campos que desea representar como Leyenda (Interfaz 3-A).</p> <p>3. Acciona el botón “Agregar” (Interfaz 3-B)</p>	<p>4. Muestra los campos seleccionados en la pantalla. (Interfaz 3-C).</p>

Evento 2.3.1 “Eliminar campo de Leyenda”	
<p>1. Desea eliminar un(os) campo(s) previamente seleccionado(s)</p> <p>2. Selecciona el(los) campo(s) que desea eliminar dando clic sobre estos (Interfaz 3-C).</p> <p>3. Acciona el botón “Eliminar” (Interfaz 3-D).</p>	<p>4. Muestra un mensaje de alerta “Esta seguro que desea eliminar el campo seleccionado”</p> <p>5. Elimina el(los) campos(s) que seleccionó y actualiza la interfaz sin el(los) campo(s) eliminado(s) (Interfaz 3-C).</p>
Evento 3 “Visualizar la información”	
Cursos Alternos	
Evento 2.1 “Definir campo de dato”	
Línea 3: El sistema muestra un mensaje de error porque este campo ha sido seleccionado anteriormente como campo de de Eje X o campo de Leyenda “Ya este campo ha sido seleccionado”	
Línea 4: Si se trata de un gráfico de pastel no se tiene que especificar un incremento	
Línea 5: Si se trata de un gráfico de pastel no se tiene que definir un título	
Evento 2.2 “Definir campos del Eje X”	
Línea 4: El sistema muestra un mensaje de error porque este campo ha sido seleccionado anteriormente como campo de Dato, como campo de Eje X o como campo de Leyenda “Ya este campo ha sido seleccionado”	
Evento 2.2.1 “Eliminar campo del Eje X”	
Línea 4: Si el recuperador responde negativamente el campo seleccionado no se elimina	
Evento 2.3 “Definir campos de Leyenda”	
Línea 4: El sistema muestra un mensaje de error porque este campo ha sido seleccionado anteriormente como campo de Dato, como campo del Eje X o como campo de Leyenda “Ya este campo ha sido seleccionado”	
Evento 2.3.1 “Eliminar campo de Leyenda”	
Línea 4: Si el recuperador responde negativamente el campo seleccionado no se elimina	
Poscondiciones: Quedarán definidos los campos que se mostrarán en el informe gráfico.	

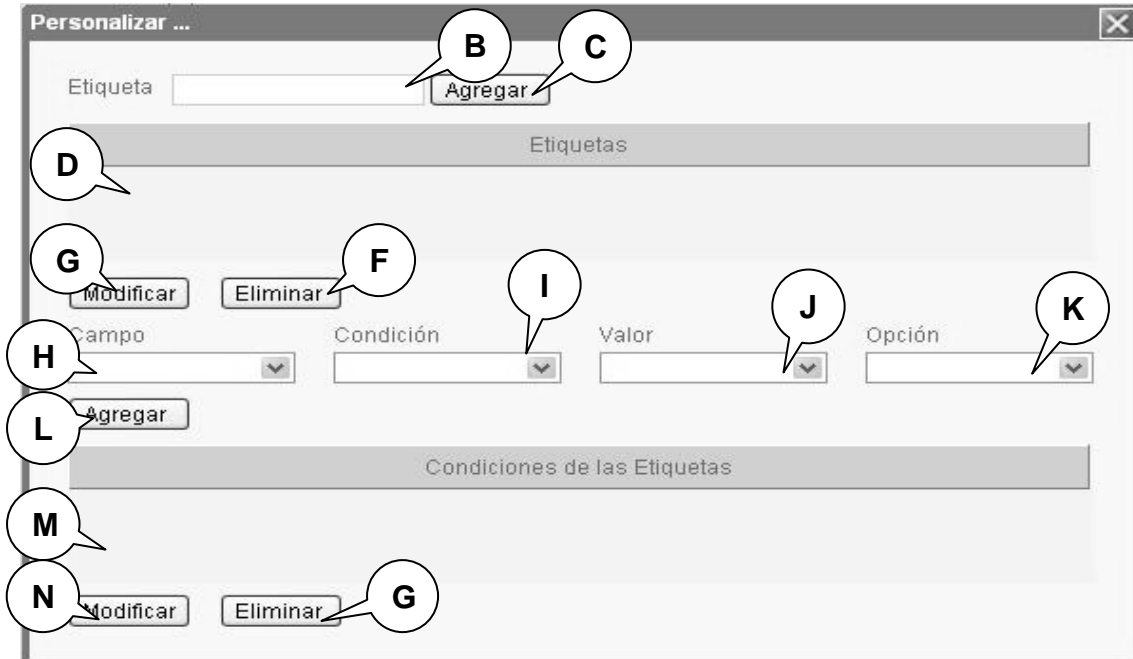
Caso de Uso:	Personalizar
Actor(es):	Recuperador (inicia)
Propósito:	Definir las condiciones por campos que aparecerán en la recuperación con el fin de personalizarlos.
Resumen:	El caso de uso inicia cuando el recuperador accede a definir la personalización de campos que han sido seleccionados previamente por el recuperador. El caso de uso termina cuando el recuperador termina y accede a otras opciones de la aplicación.

Referencias:	R6
Precondiciones:	Debe haber sido autenticado como recuperador. Debe haber seleccionado un campo.

Interfaz 1



Interfaz 2



Acción del Actor	Respuesta del Sistema
-------------------------	------------------------------

<p>1. Marca la opción Personalizar (Interfaz 1-A)</p> <p>3. Para agregar una etiqueta para este campo ir a Evento 1 “Agregar etiqueta”</p>	<p>2. Muestra la interfaz 2 personalizar con todas sus opciones</p>
<p>Evento 1 “Agregar etiqueta”</p>	
<p>1. Especifica el nombre de la etiqueta que tendrá este campo y que se mostrará en la recuperación (Interfaz 2-B).</p> <p>2. Acciona el botón “Agregar” (Interfaz 2-C).</p> <p>4. Para agregar una o varias condiciones para esta etiqueta ir a Evento 2 “Agregar condiciones”.</p>	<p>3. Muestra en la interfaz la etiqueta creada (Interfaz 2-D).</p>
<p>Evento1.1 “Modificar Nombre de la Etiqueta”</p>	
<p>1. Desea modificar el nombre de una etiqueta previamente creada</p> <p>2. Selecciona el nombre de la etiqueta que desea modificar dando clic sobre esta (Interfaz 2 -D).</p> <p>3. Acciona el botón “Modificar” (Interfaz 2 -G)</p> <p>5. Modifica el nombre de la etiqueta (Interfaz 2 -B)</p> <p>6. Acciona el botón “Agregar” (Interfaz 2 - C)</p>	<p>4. Muestra en Interfaz el nombre de la etiqueta a modificar (Interfaz 2 -B)</p> <p>7. Muestra el nuevo nombre en la interfaz. (Interfaz 2 -B).</p>
<p>Evento1.2 “Eliminar Nombre de la Etiqueta ”</p>	
<p>1. Desea eliminar una etiqueta previamente creada</p> <p>2. Selecciona la etiqueta que desea eliminar dando clic sobre esta</p> <p>3. Acciona el botón “Eliminar” (Interfaz 2-F)</p>	<p>4. Muestra un mensaje de alerta “Está seguro de eliminar la etiqueta seleccionada”</p> <p>5. Actualiza la interfaz sin la etiqueta eliminada y si existen condiciones asociadas a esta etiqueta las elimina también.</p>

Evento 2 “Agregar Condiciones”	
<p>1. Desea crear una o varias condiciones para la etiqueta anteriormente creada</p> <p>2. Selecciona el campo que formara parte de la condición (Interfaz 2-H)</p> <p>4. Selecciona la condición que utilizará (Interfaz 2-I).</p> <p>5. Selecciona el valor y la opción (Interfaz 2-J, K)</p> <p>6. Acciona el botón “Agregar” (Interfaz 2-L).</p>	<p>7. Las condiciones que se van formando se muestran en la parte inferior de la interfaz (Interfaz 2-M)</p>
Evento “Modificar Condición”	
<p>1. Desea modificar una condición previamente creada</p> <p>2. Selecciona la condición que desea modificar dando clic sobre esta.</p> <p>3. Acciona el botón “Modificar” (Interfaz 2-N)</p> <p>5. Modifica los datos</p> <p>6. Acciona el botón “Agregar” (Interfaz 2-L)</p>	<p>4. Muestra los datos de la condición para que puedan ser modificados (Interfaz 2-H, I, J, K)</p> <p>7. Muestra la condición que se esta modificando en la interfaz (Interfaz 2-M)</p>
Evento “Eliminar Condición”	
<p>1. Desea eliminar una o varias condiciones que ha creado.</p> <p>2. Escoge la condición que desea eliminar dando clic sobre esta</p> <p>3. Acciona el botón “Eliminar” (Interfaz 2-G)</p>	<p>4. Muestra un mensaje de alerta “Está seguro que desea eliminar la condición seleccionada”</p> <p>5. Elimina la condición que seleccionó y actualiza la interfaz sin la condición eliminada</p>
Cursos Alternos	
Evento 1 “Agregar etiqueta”	
Línea 2: El sistema muestra un mensaje de error “Ya existe una etiqueta con ese nombre ”	
Evento 1.1 “Modificar Etiqueta”	
Línea 6: El sistema muestra un mensaje de error “Ya existe una etiqueta con ese nombre ”	
Evento 1.2 “Eliminar Etiqueta”	

<p>Línea 4: Si el actor responde negativamente el sistema no elimina la etiqueta especificada.</p> <p>Evento 2 “Agregar Condición”</p> <p>Línea 6: El sistema muestra mensaje de error porque ya existe otra condición igual, “Imposible agregar esta condición”.</p> <p>Evento 2.1 “Modificar Condición”</p> <p>Línea 6: El sistema muestra mensaje de error porque ya existe otra condición igual, “Imposible agregar la nueva condición”.</p> <p>Evento “Eliminar Condición”</p> <p>Línea 4: Si el actor responde negativamente, la aplicación no elimina la condición seleccionada.</p>
<p>Poscondiciones: El campo seleccionado ha sido personalizado.</p>

Caso de Uso:	Generar Vista Previa
Actor(es):	Recuperador (inicia)
Propósito:	Generar la visualización de un informe por listado, por estadísticas o por gráficos.
Resumen:	El caso de uso comienza cuando el recuperador acciona el tab “Vista Previa”. El caso de uso termina con la generación de la visualización de un informe por listado, por estadísticas o por gráficos.
Referencias:	R7
Precondiciones:	Debe estar autenticado como recuperador. Para visualizar un informe por listado debe definir al menos un campo, en el caso de estadística un campo por columna y al menos uno por fila, en el caso de gráfico debe estar definido al menos un campo de datos y un campo de leyenda.



Interfaz 2

Recuperaciones Estadísticas

Condiciones
Campos Encabezado
Campo Columna
Campos Fila
Formato Informe
Vista Previa

Personal por Órgano, Sexo y Nivel Escolar

ORGANOS	SEXO	TOTAL	MEDIO	MEDIO SUPERIOR	
SUPERIOR					
Órgano 1	Masculino	27	5	12	10
	Femenino	24	3	14	7
Total de Órgano1		51	8	26	17
Órgano 2	Masculino	25	8	3	14
	Femenino	24	5	15	4
Total de Órgano2		49	13	18	18
Total de General		100	21	44	35

Interfaz 3

Recuperaciones Gráficas

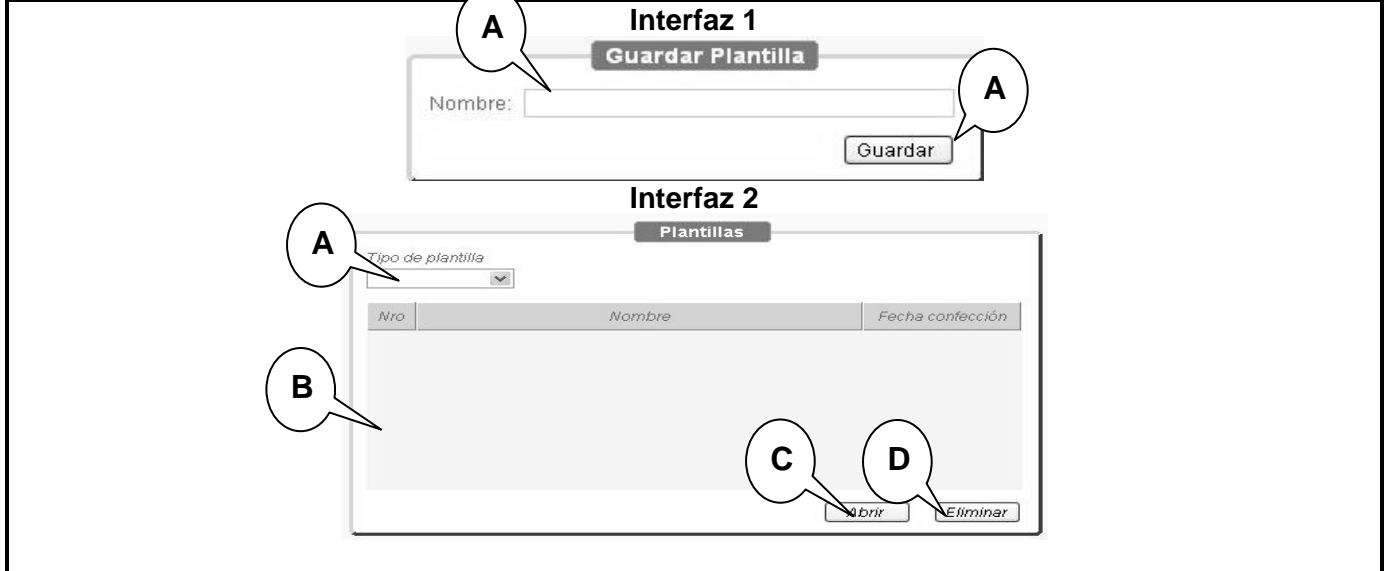
Tipos Gráficos
Condiciones
Formato Informe
Campo Dato
Campos Eje X
Campos Leyenda
Vista Previa

Category	Value
1	29.6
2	25.9
3	33.3

Acción del Actor	Respuesta del Sistema
1. En la interfaz de recuperaciones por listado, recuperaciones estadísticas o recuperaciones por gráficos, activa el tab "Vista previa"	. 2. El sistema muestra la visualización del informe por listado, por estadísticas o por gráficos.

3. Para guardar el informe generado deberá accionar el botón "Guardar"	4. El sistema guarda el reporte en el formato especificado (los formatos pueden ser: .doc, .pdf, .xls además para gráficos se podrá guardar en formato JPG y PNG).
Cursos Alternos	
-	
Poscondiciones: Se han se actualizado las diferentes interfaces que intervienen en los procesos de guardar, abrir o eliminar plantillas.	

Caso de Uso:	Gestionar Plantilla
Actor(es):	Recuperador (inicia)
Propósito:	Guardar como plantilla los procesos de recuperación de información que se hayan realizado o, abrir o eliminar procesos de recuperación de información que se hayan salvado como plantilla.
Resumen:	El caso de uso comienza cuando el recuperador desea salvar una recuperación como plantilla o, abrir o eliminar una plantilla guardada con anterioridad. El caso de uso finaliza cuando se actualizan las diferentes interfaces que intervienen en los procesos de guardar, abrir o eliminar plantillas.
Referencias:	R8
Precondiciones:	Debe estar autenticado como recuperador. Si se desea abrir o eliminar plantilla debe existir al menos una plantilla. Para salvar un informe por listado debe definir al menos un campo, en el caso de estadística un campo por columna y al menos uno por fila, en el caso de grafico debe estar definido al menos un campo de datos y un campo de leyenda.



Acción del Actor	Respuesta del Sistema
<p>1. Accede a la aplicación</p> <p>3. Para abrir una recuperación guardada como plantilla ir a Evento “Abrir Plantilla”</p> <p>4. Para eliminar una recuperación guardada como plantilla ir al Evento “Eliminar Plantilla”</p> <p>5. Para guardar una recuperación como plantilla ir al Evento “Salvar Plantilla”</p>	<p>.</p>
Evento “Salvar Plantilla”	
<p>1. Accede en el menú, en inicio, a la opción salvar plantilla</p> <p>3. Introduce el nombre con el que quiere guardar la plantilla. (Interfaz 1-A).</p> <p>4. Acciona el botón “Guardar” (Interfaz 1- B).</p>	<p>2. Muestra la interfaz Salvar plantilla</p> <p>5. Guarda las diferentes definiciones hechas en las tablas correspondientes de la base de datos.</p> <p>3. Muestra la interfaz Plantilla actualizada con la nueva plantilla salvada (Interfaz 2).</p>
Evento “Abrir Plantilla”	
<p>1. Accede en el menú, en inicio a la interfaz de Plantillas con el fin de abrir una plantilla previamente salvada</p> <p>2. Selecciona el tipo de plantilla que necesita abrir puede ser por listado, estadísticas o gráficos (Interfaz 2-A)</p> <p>4. Selecciona la plantilla que desea abrir dando clic sobre esta (Interfaz 2-B).</p> <p>5. Acciona el botón “Abrir” (Interfaz 2-C)</p> <p>7. Para visualizar la información contenida en</p>	<p>3. Muestra la interfaz actualizada por la selección anterior, con el nombre de la plantilla(s) y la fecha en que fue confeccionada (Interfaz 2-B)</p> <p>6. Muestra los datos en las interfaces de listado en caso de que la plantilla sea de recuperación por listado, muestra los datos en las interfaces de estadística en caso de que la plantilla sea de recuperación por estadística, muestra los datos en las interfaces de gráfico en caso de que la plantilla sea de recuperación por gráficos haciendo corresponder cada componente de la pagina con sus datos relacionados.</p>

la plantilla activa el tab "Vista Previa"	8. Invoca al caso de uso generar "Vista Previa"
Evento "Eliminar Plantilla"	
<p>1. Accede en menú, en inicio a la interfaz de Plantillas con el fin de eliminar una plantilla previamente salvada</p> <p>2. Selecciona el tipo de plantilla que necesita eliminar puede ser por listado, estadísticas o gráficos (Interfaz 2-A)</p> <p>4. Selecciona la(s) plantilla(s) que desea eliminar dando clic sobre esta(s) (Interfaz 2-B)</p> <p>5. Acciona el botón "Eliminar" (Interfaz 2-D)</p>	<p>3. Muestra la interfaz actualizada por la selección anterior, con el nombre de la plantilla(s) y la fecha en que fue confeccionada (Interfaz 2-B)</p> <p>6. Pregunta si esta seguro que desea eliminar la(s) plantilla(s) seleccionadas.</p> <p>7. Elimina la(s) plantilla(s) y muestra la lista de plantillas sin la(s) eliminada(s) (Interfaz 2-B).</p>
Cursos Alternos	
Evento "Eliminar Plantilla":	
Línea 6: Si el recuperador responde negativamente, no se elimina ninguna plantilla.	
Poscondiciones: Se han se actualizado las diferentes interfaces que intervienen en los procesos de guardar, abrir o eliminar plantillas.	

2.7 Conclusiones

En este capítulo se comenzó a profundizar en el desarrollo de la propuesta de solución, obteniéndose una lista de las funcionalidades que debe tener el sistema, las mismas fueron representadas mediante un Diagrama de Casos de Uso, finalmente fueron descritas todas las acciones que realiza el actor y el sistema en general. A partir de aquí se puede comenzar a construir el sistema, cumpliendo con todos los requerimientos y las funcionalidades que se consideraron en este capítulo.

CAPÍTULO 3 ANÁLISIS Y DISEÑO

3.1. Introducción

Las actividades contempladas en el análisis permiten adentrarse en el problema a resolver por lo que a través de ellas se representa una vista interna del sistema en la que, usando el lenguaje de los desarrolladores se refinan los requisitos y se estructuran en base a clases y paquetes. Este proceso continúa en el diseño hasta obtener los objetos que interactúan para cumplir los requisitos funcionales y no funcionales obtenidos.

En este capítulo se realiza el análisis y diseño de la propuesta de solución modelándose los artefactos que contribuyen al manejo de las complicaciones que puede implicar la construcción de aplicaciones Web. Además se obtiene el modelo de datos a partir de las clases especificadas como persistentes.

3.2. Modelo del Análisis

En el modelo del análisis hay un refinamiento de los requisitos del software, teniendo como objetivo comprender perfectamente los mismos. Esta estructura (basada en clases de análisis y paquetes) es independiente de la estructura que se dio a los requisitos (basada en casos de uso)

El modelo del análisis ayuda a estructurar y refinar los requisitos del software y proporciona una estructura centrada en el mantenimiento, en aspectos tales como la flexibilidad ante los cambios y la reutilización. El modelo del análisis no solo es útil para el mantenimiento de los requisitos, sino que también se utiliza como entrada en las actividades de diseño e implementación.

Este modelo ofrece además, un mayor poder expresivo y una mayor formalización.

3.2.1 Diagrama de clases del análisis

A través de los diagramas de clases del análisis se representan las clases que participan en un caso de uso y las relaciones entre estas.

Las clases del análisis se representan a través de 3 estereotipos estandarizados en UML, estos ayudan a los desarrolladores a distinguir el ámbito de las diferentes clases. A continuación se especifican brevemente estos estereotipos:

- Entidad: Modelan información que posee larga vida y que es a menudo persistente.

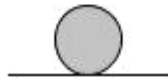


Figura 3.1: Clase entidad

- Interfaz: Modelan la interacción entre el sistema y sus actores.

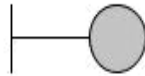


Figura 3.2: Clase Interfaz

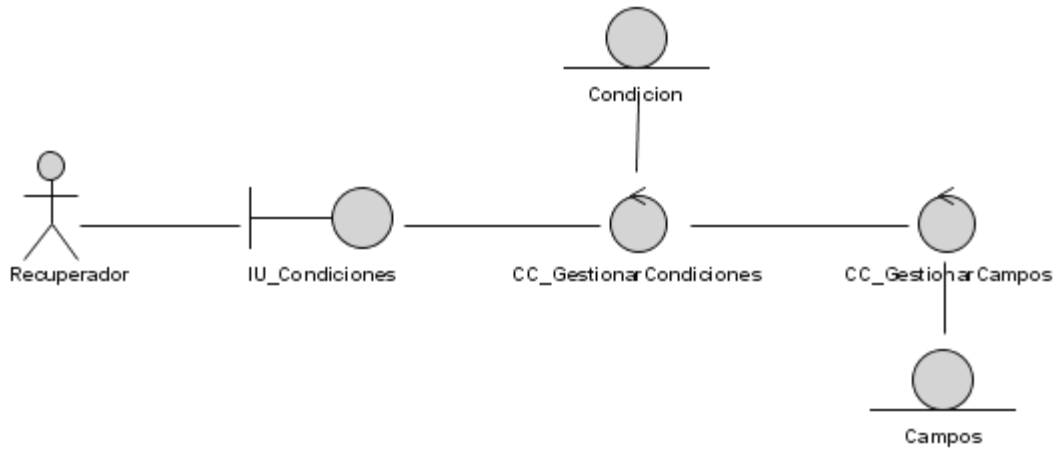
- Control: Coordinan la realización de uno o unos pocos casos de uso coordinando las actividades de los objetos que implementan la funcionalidad del caso de uso.



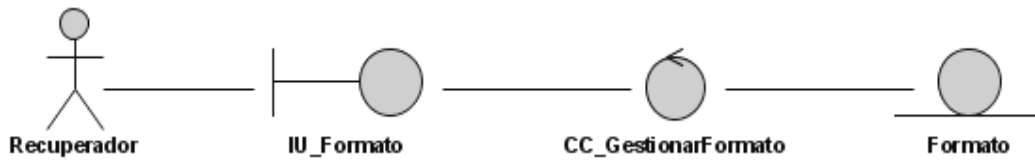
Figura 3.3: Clase Controladora

A continuación se representan los diagramas de clases del análisis por caso de uso:

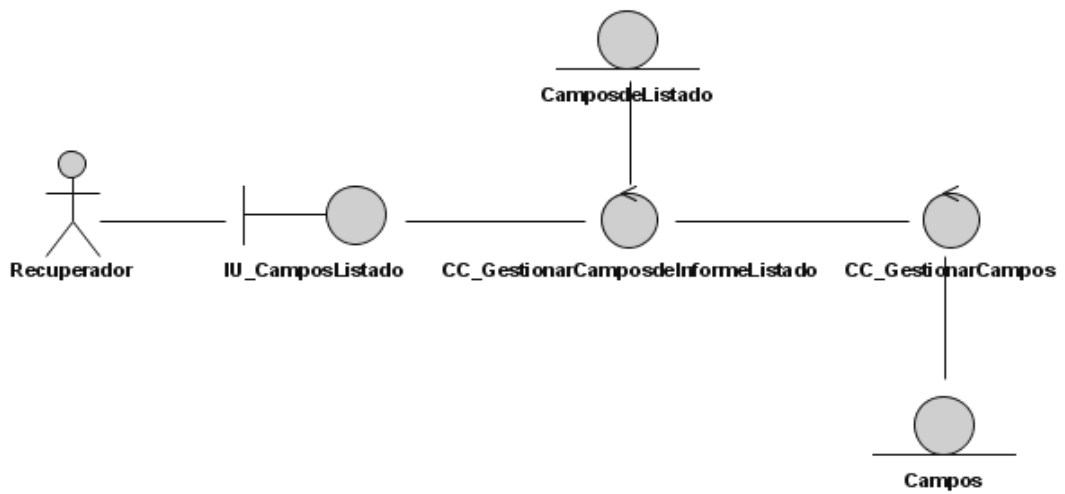
- Caso de uso Definir Condiciones del Informe



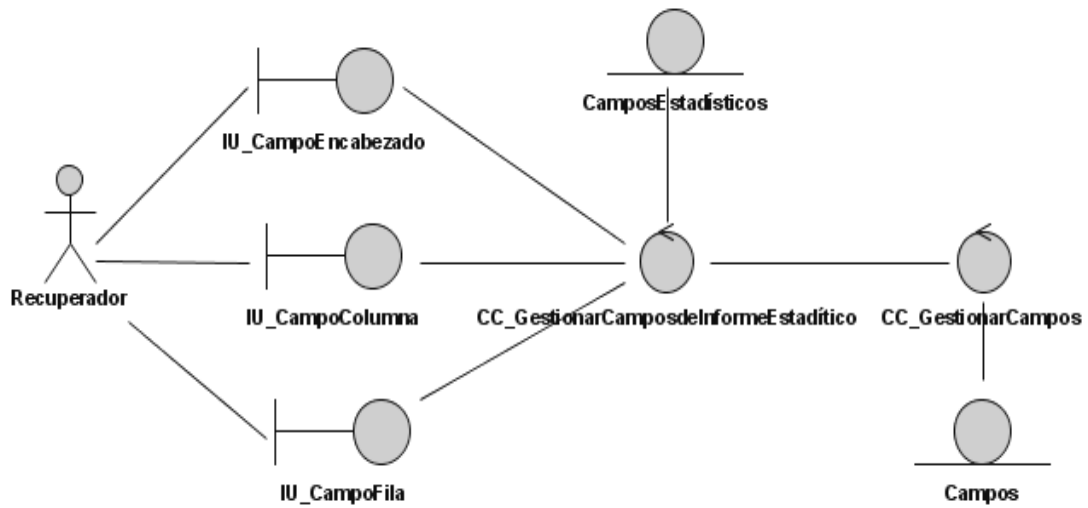
- Caso de uso Definir Formato del Informe



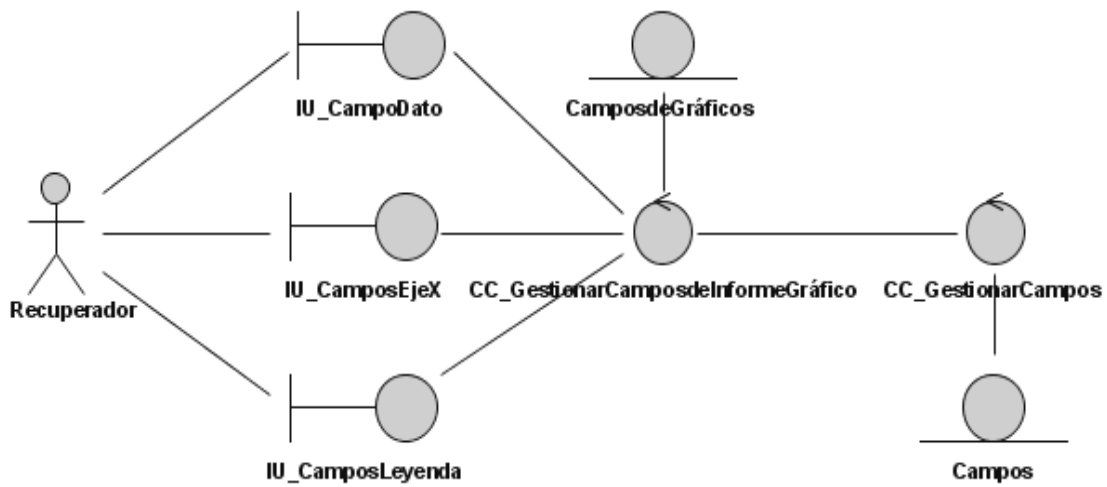
- Caso de uso Visualizar Informe por Listado



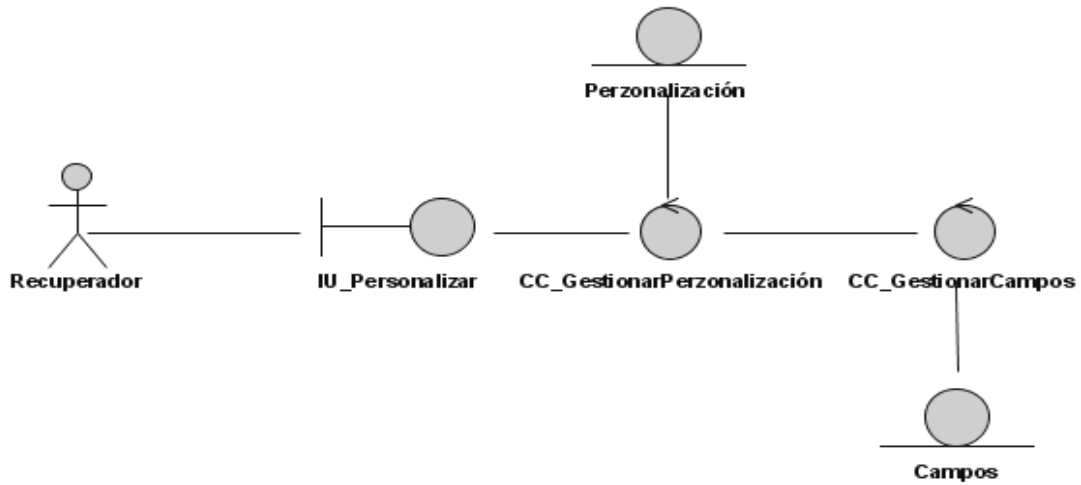
- Caso de uso Visualizar Informe Estadístico



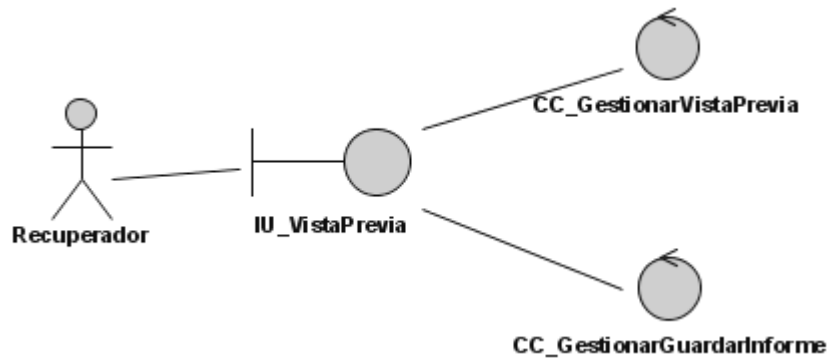
- Caso de uso Visualizar Informe Gráfico



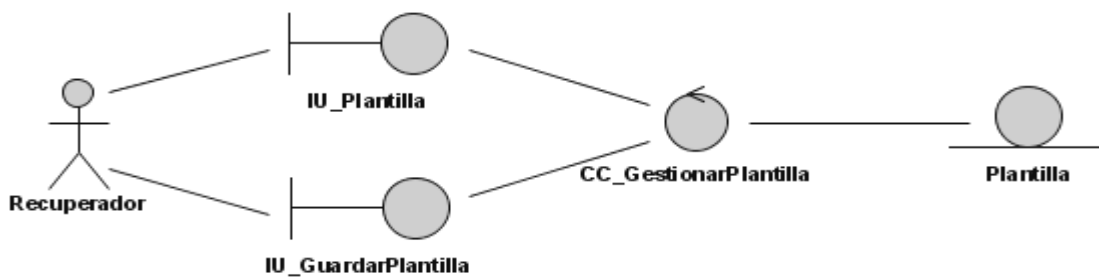
- Caso de uso Personalizar



- Caso de uso Generar Vista Previa



- Caso de uso Gestionar plantilla



3.3. Modelo del Diseño

En el diseño se modela el sistema y se encuentra su forma para que soporte todos los requisitos, incluyendo los no funcionales. El modelo del diseño es un modelo de objetos que describe la realización física de los casos de uso centrándose en como los requisitos funcionales y no funcionales tienen impacto en el sistema. En este modelo los casos de uso son realizados por clases de diseño y sus objetos.

Uno de los propósitos fundamentales del diseño es: “Crear una entrada apropiada y un punto de partida para actividades de implementación subsiguientes capturando los requisitos o subsistemas individuales, interfaces y clases”

3.3.1 Mecanismos de diseño

Un mecanismo de diseño se modela con el fin de comunicar mejor la manera de dar solución a problemas recurrentes en la aplicación, sirviéndonos de guía en la utilización de diferentes estilos de diseño. Estos mecanismos se identifican a partir de comportamientos comunes; con su elaboración, teniendo en cuenta las tecnologías de diseño e implementación disponibles, se modela un conocimiento que ayuda tanto al desarrollo de la aplicación actual como a construcciones futuras, además de las labores de mantenimiento.

Se pueden construir mecanismos de diseño para cada servicio que se necesite dentro de la aplicación. Relacionado con este sistema se utiliza uno para el acceso a datos concurrentes.

Este mecanismo se ha documentado con la descripción de las diferentes clases que participan en él, además con diagramas de vista estática (diagramas de clase) y diagramas de interacción que muestran las relaciones entre estas clases estas descripciones son partes de la documentación del proyecto.

El diagrama que representa la vista estática de la solución propuesta para manejar el acceso a datos del sistema se encuentra en el (ANEXO I).

3.3.2 Diagrama de clases del diseño

Esta aplicación ha sido diseñada usando la arquitectura n capas, la cual se trató en el capítulo 1. Obedeciendo esta arquitectura y teniendo en cuenta las funcionalidades que debe implementar el sistema que se está desarrollando se muestra a continuación un diagrama de la arquitectura del sistema:

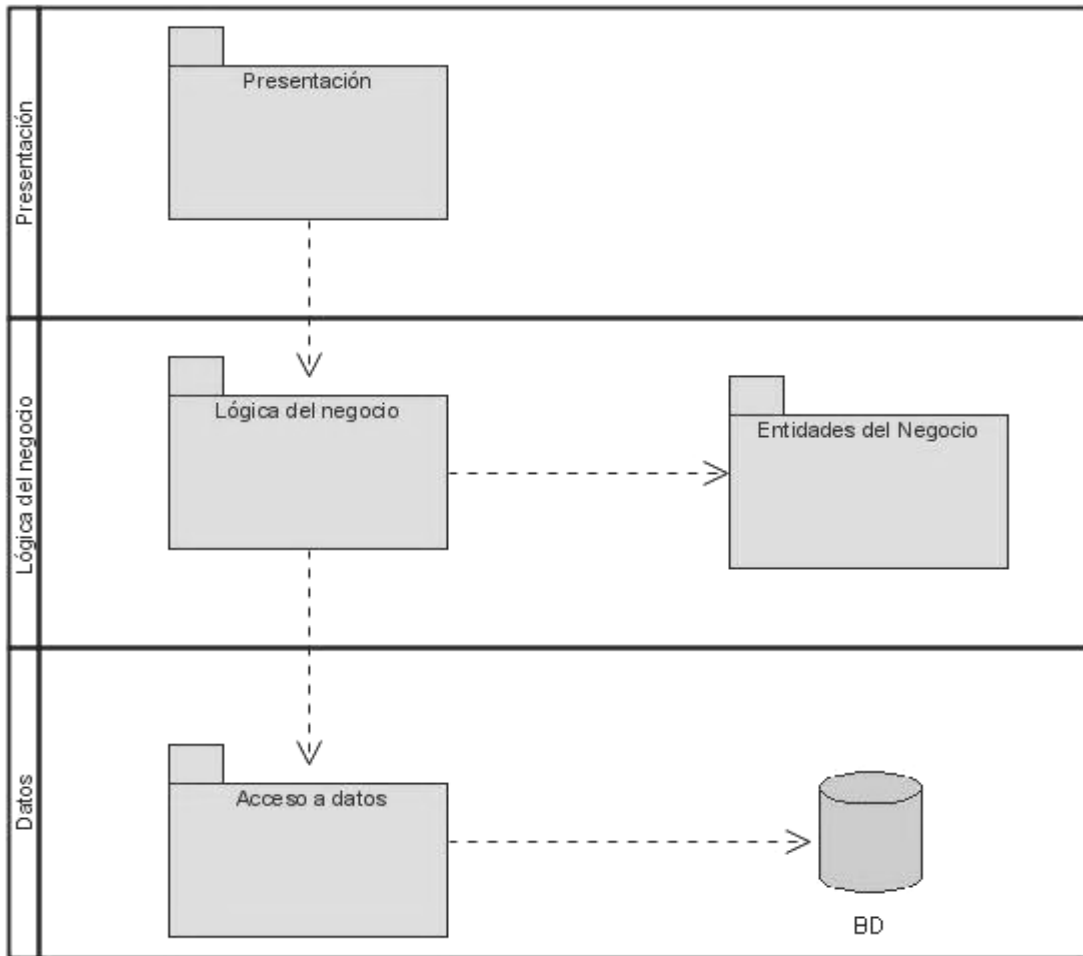
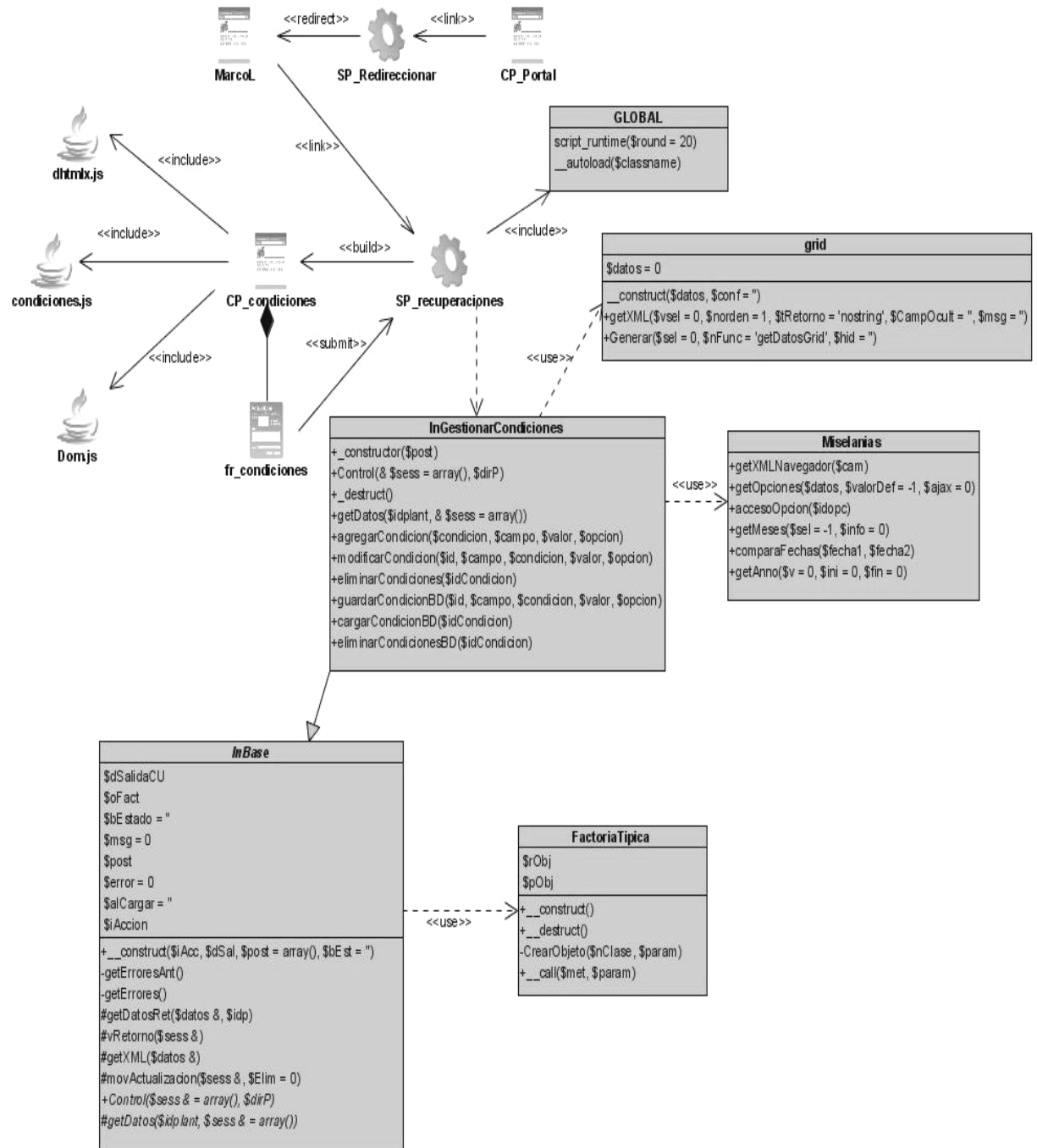


Figura 3.5.: Arquitectura del sistema

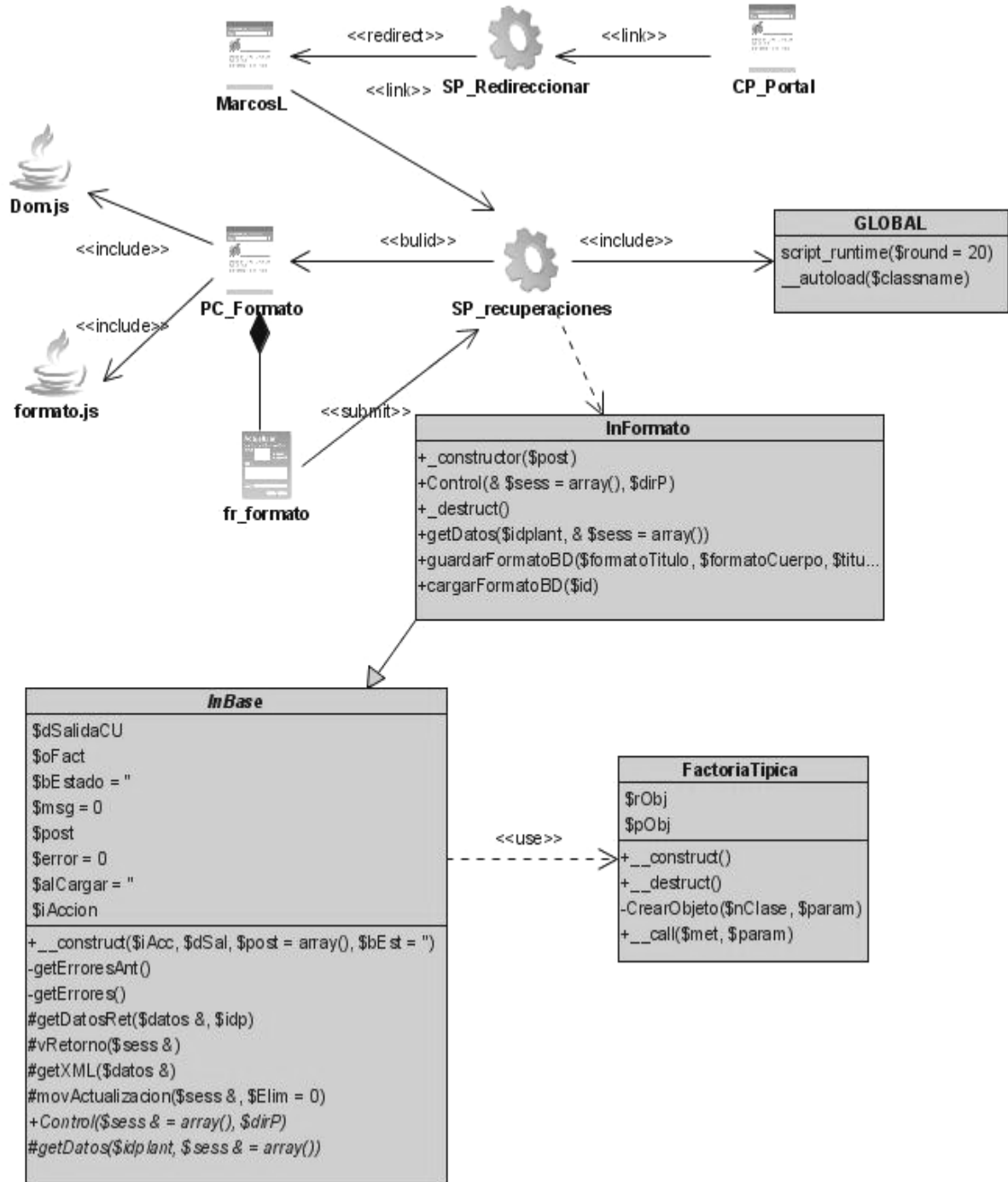
Teniendo en cuenta la arquitectura especificada se han realizado los diagramas de clases del diseño que muestran las clases participantes y sus relaciones, dividiéndolos por casos de uso con el fin de lograr una mejor reutilización y legibilidad de estos diagramas.

Los diagramas de clases se utilizan para modelar la vista de diseño estática de un sistema y constituyen la base para los diagramas de componentes y el diagrama de despliegue. A continuación se muestran los diagramas de clases del diseño especificados en nuestro sistema:

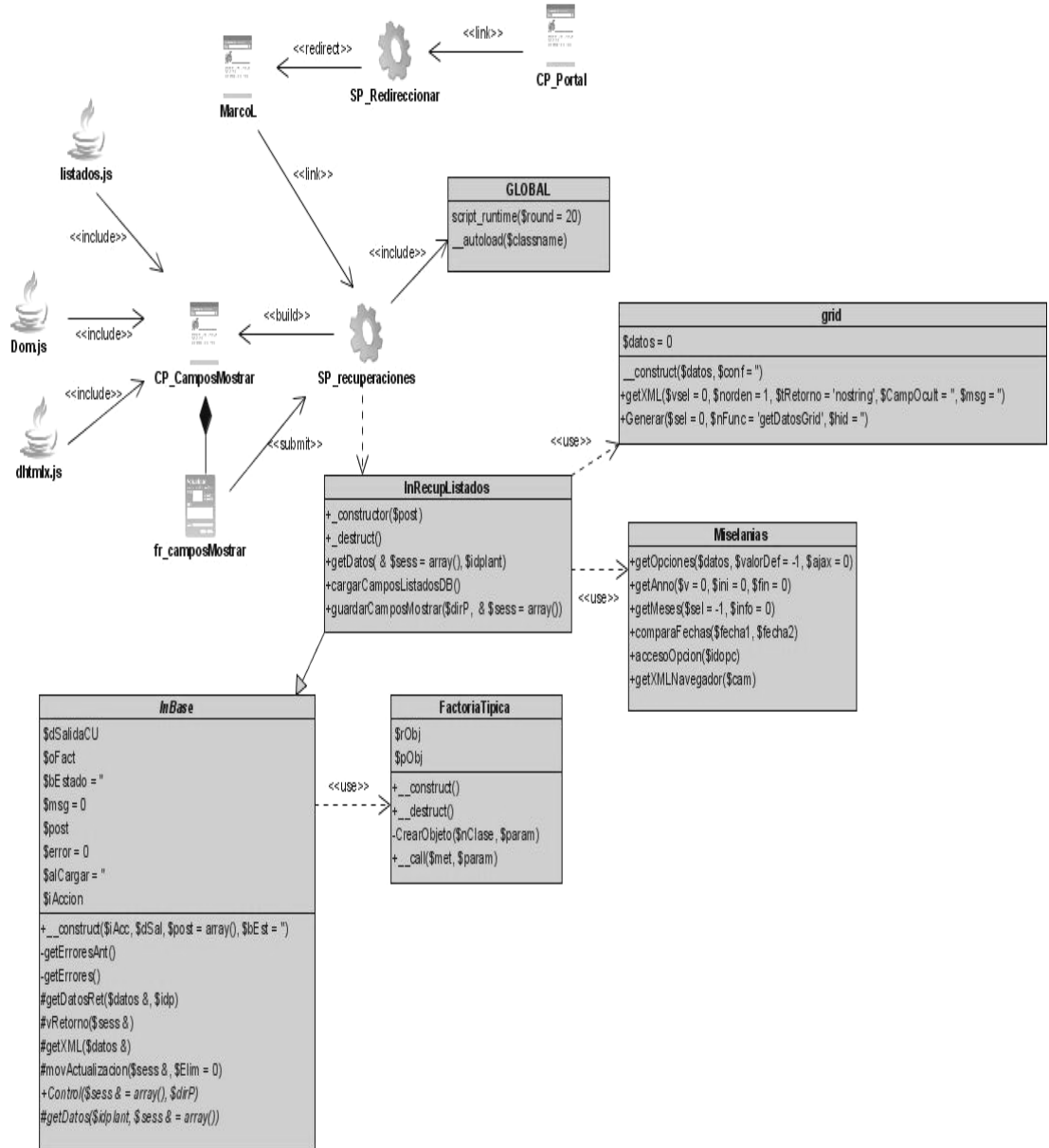
• Caso de uso Definir Condiciones



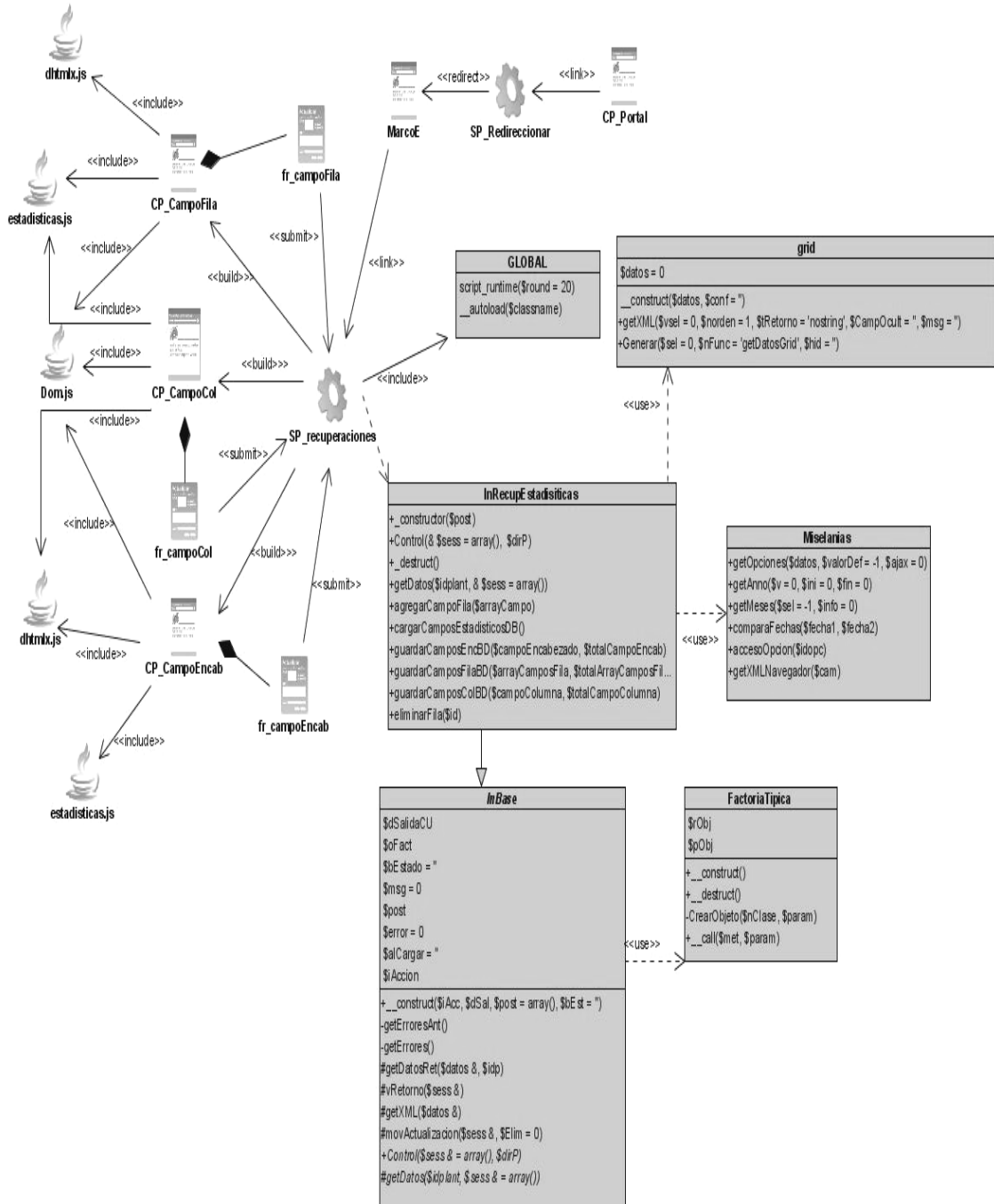
• Caso de uso Definir Formato



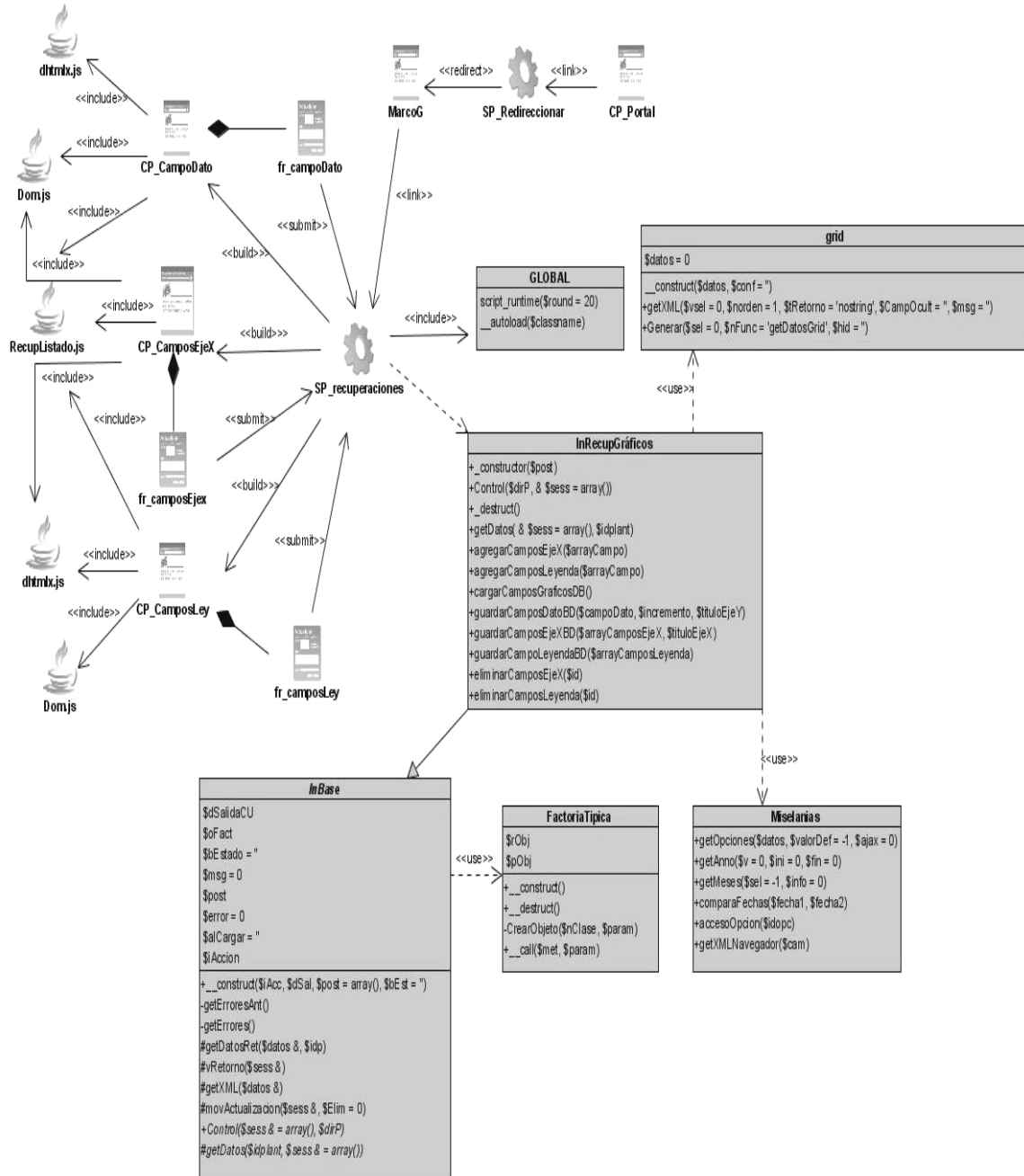
- Caso de uso Visualizar Informe por Listado



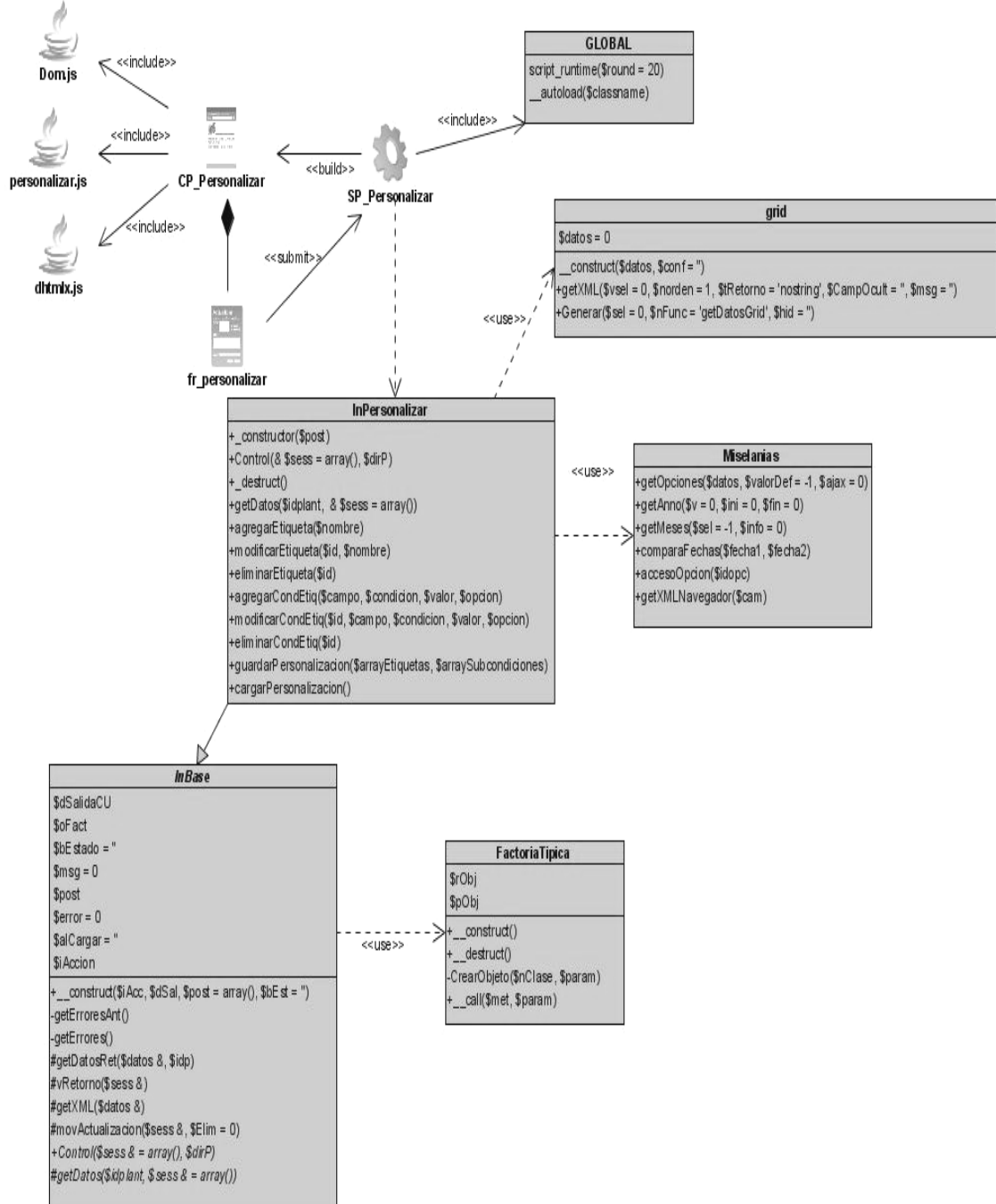
- Caso de uso Visualizar Informe Estadístico.



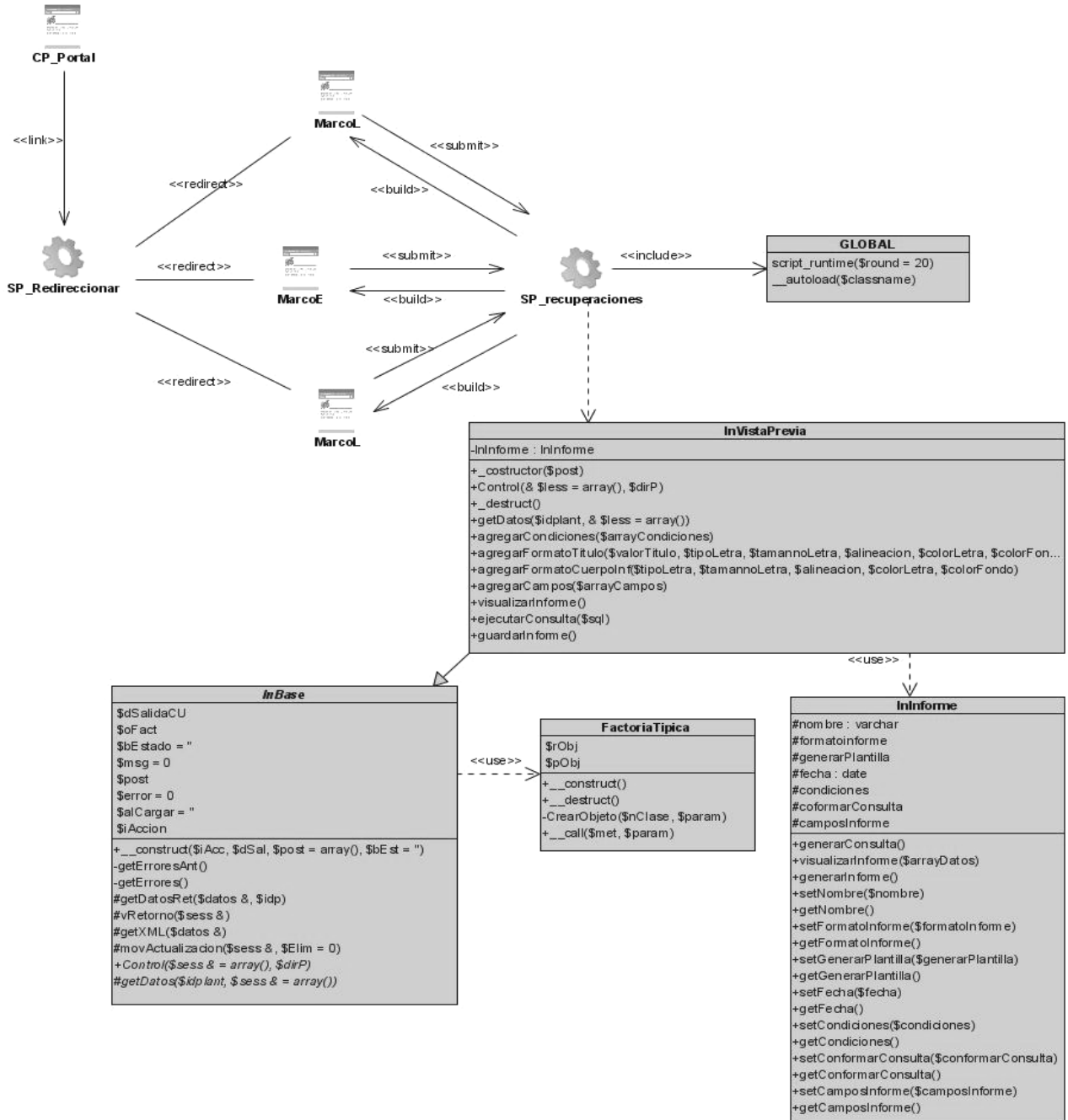
- Caso de uso Visualizar Informe Gráfico

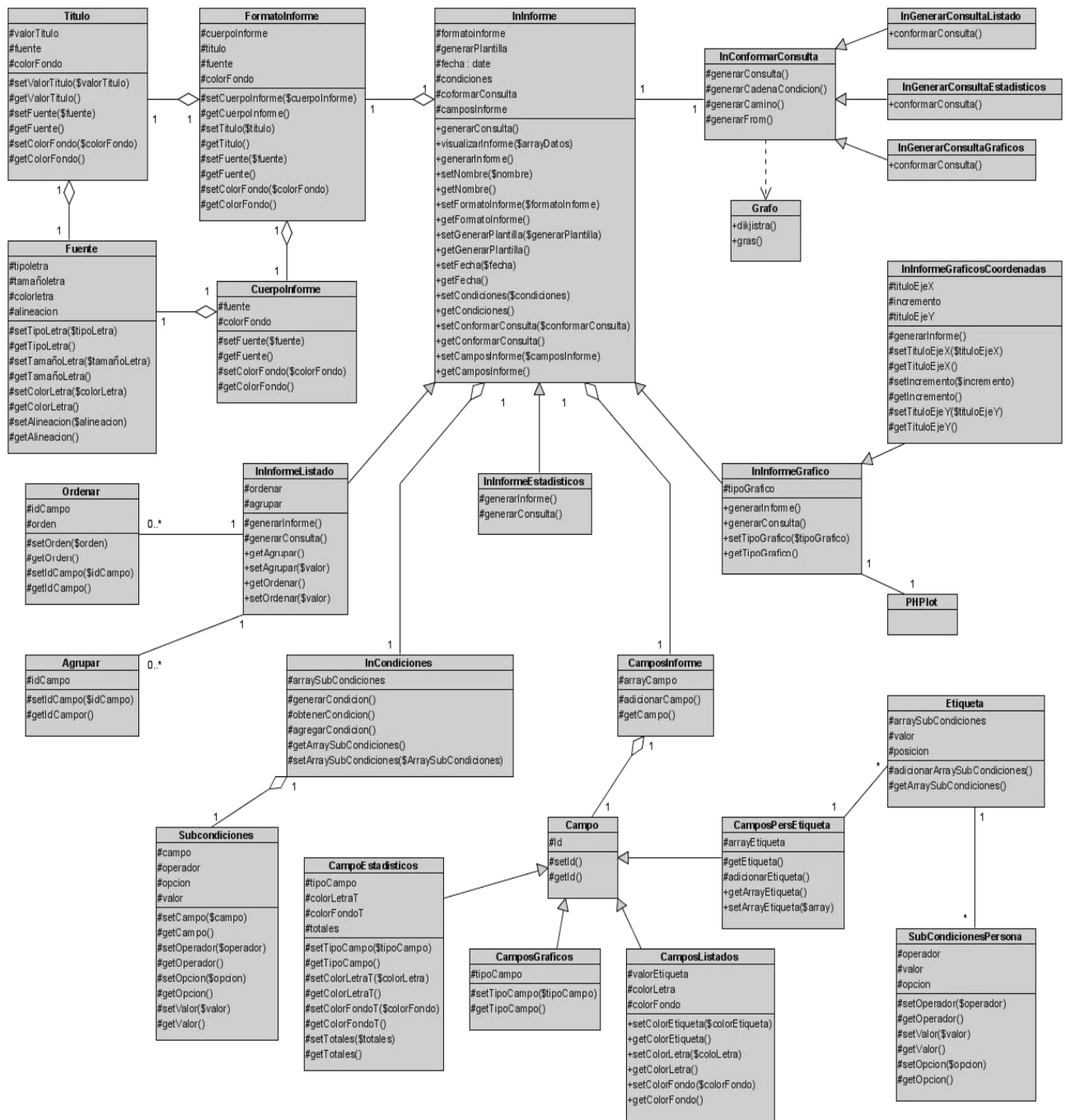


• Caso de uso Personalizar

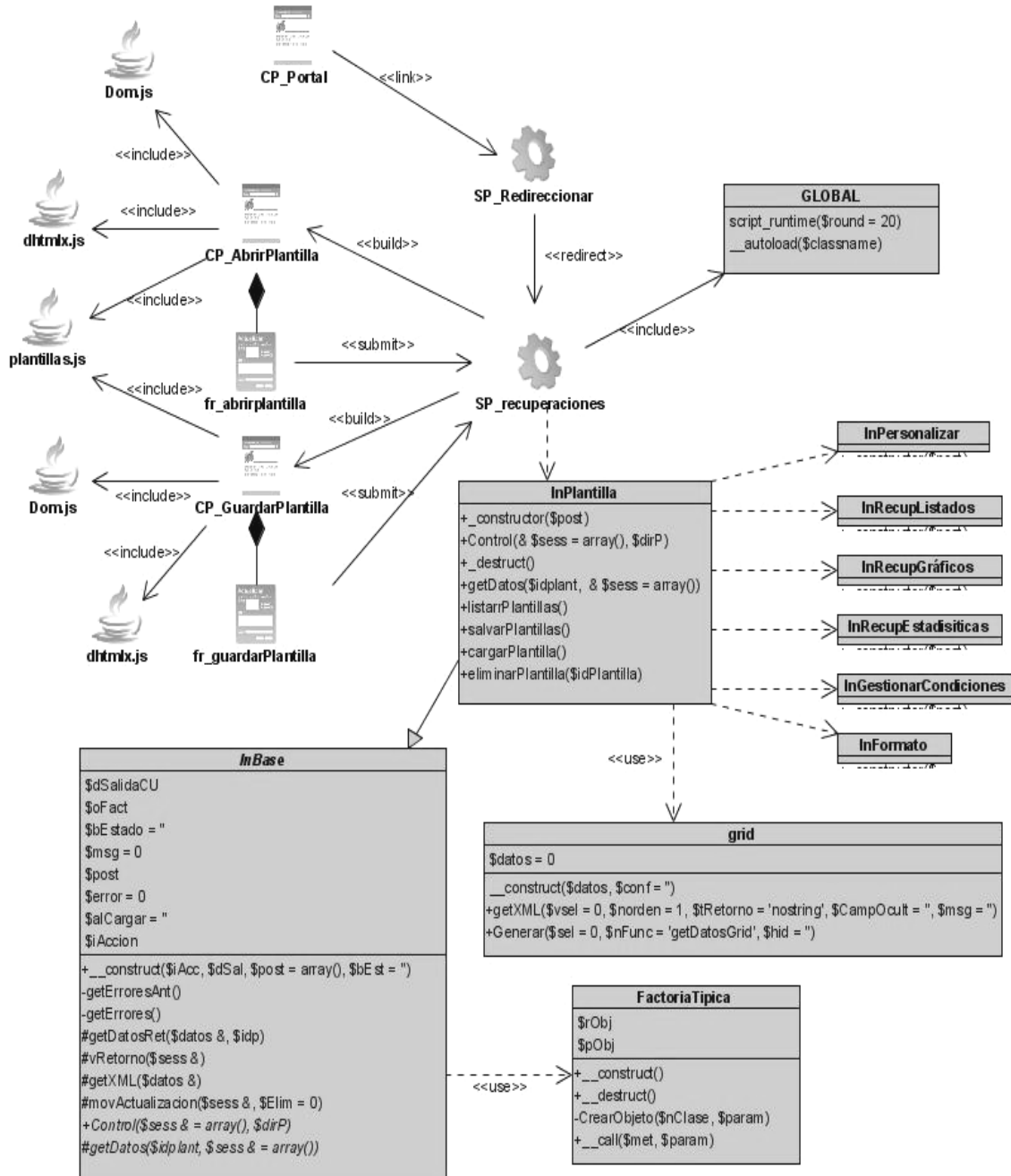


• Caso de uso Generar Vista previa





• Caso de uso Gestionar Plantilla



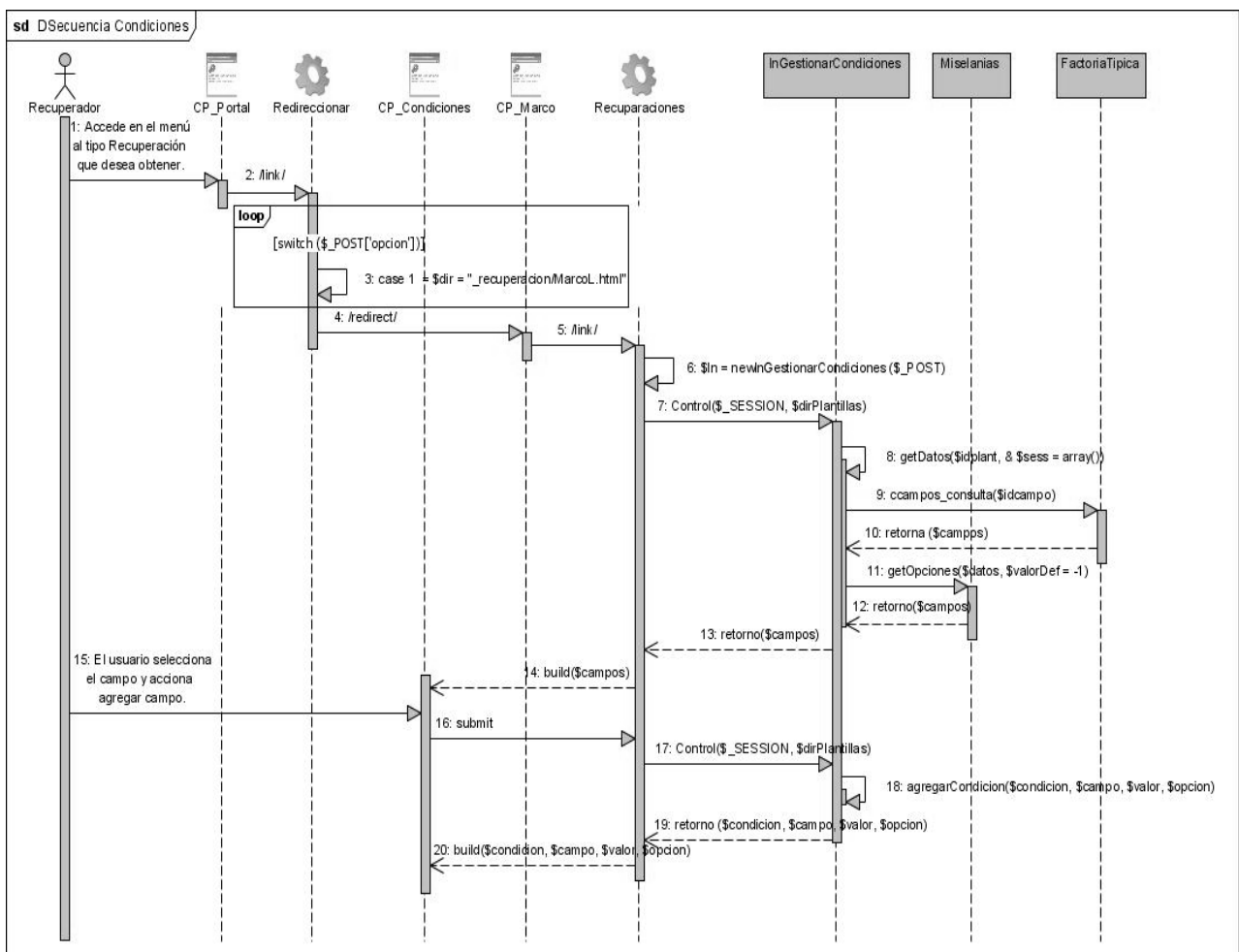
3.3.3 Diagramas de Interacción

Los diagramas de secuencia se utilizan para modelar los aspectos dinámicos del sistema. Un diagrama de secuencia muestra un conjunto de objetos y sus relaciones, incluyendo los mensajes que se pueden enviar entre ellos, estos destacan la ordenación temporal de los mensajes.

A continuación se muestran los diagramas de secuencia de algunos escenarios de cada caso de uso:

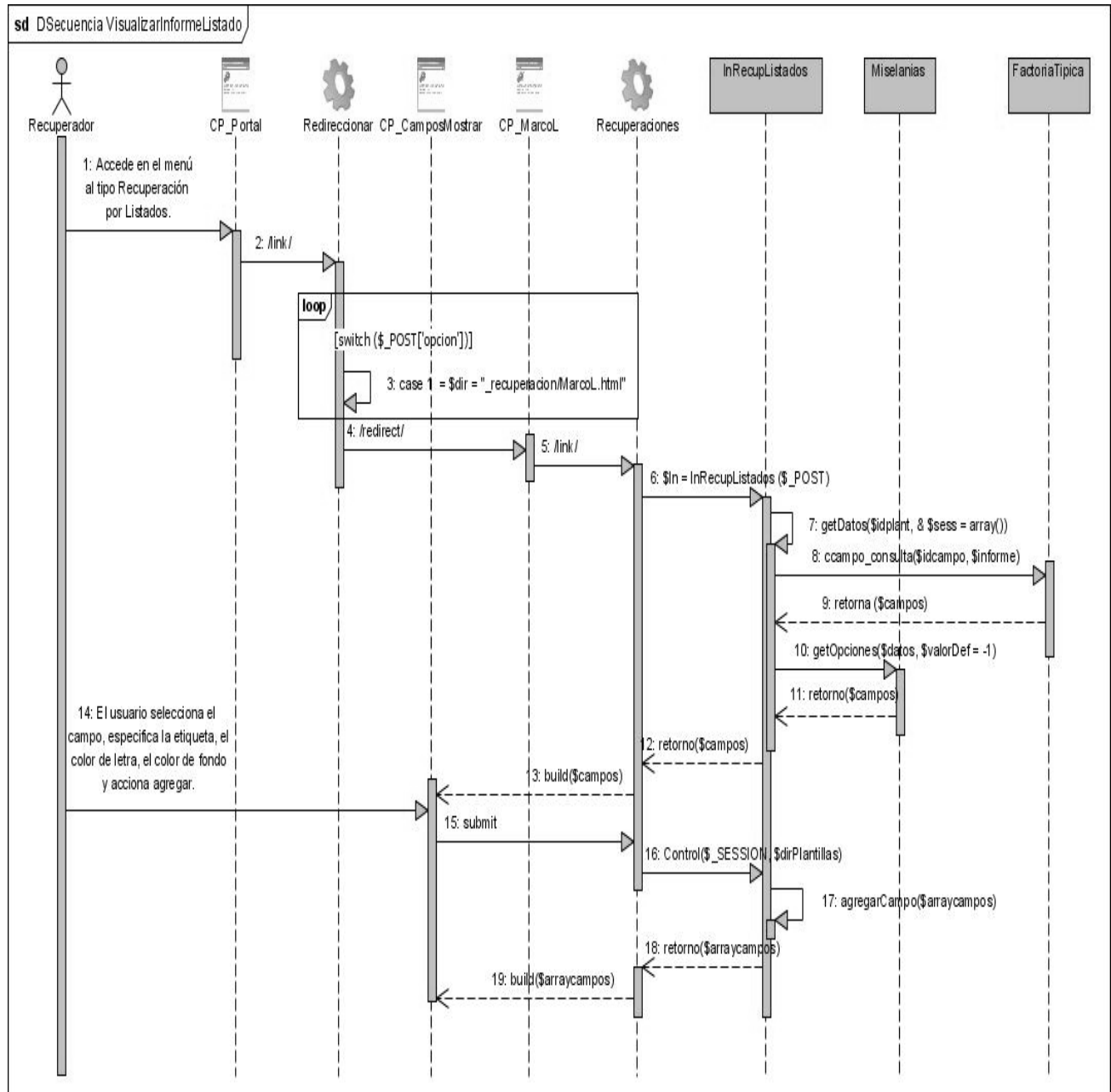
- **Caso de Uso Definir Condiciones**

- Escenario Agregar Condiciones:



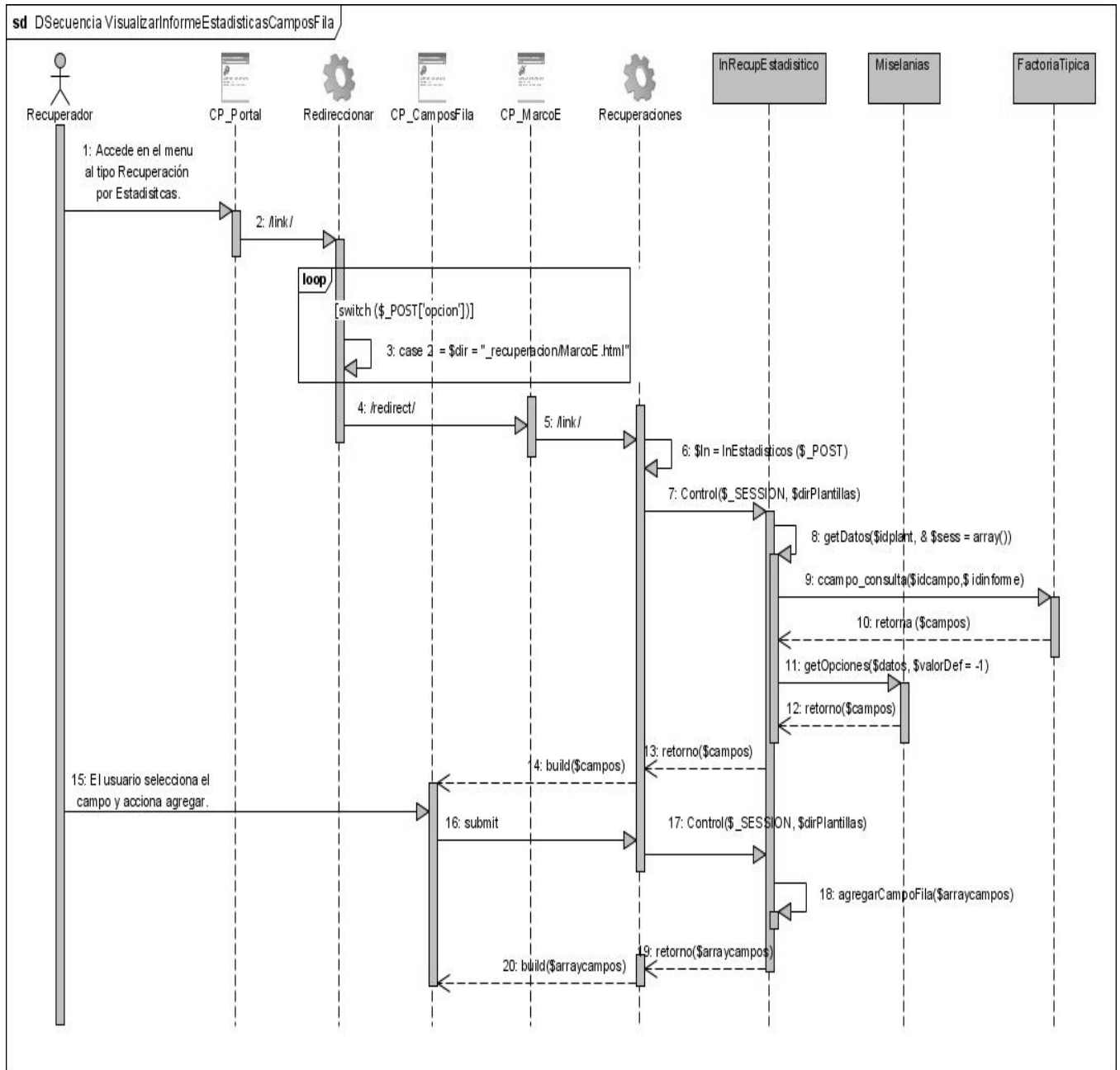
- Caso de Uso Visualizar Informe por Listado

- Escenario Agregar Campos Mostrar



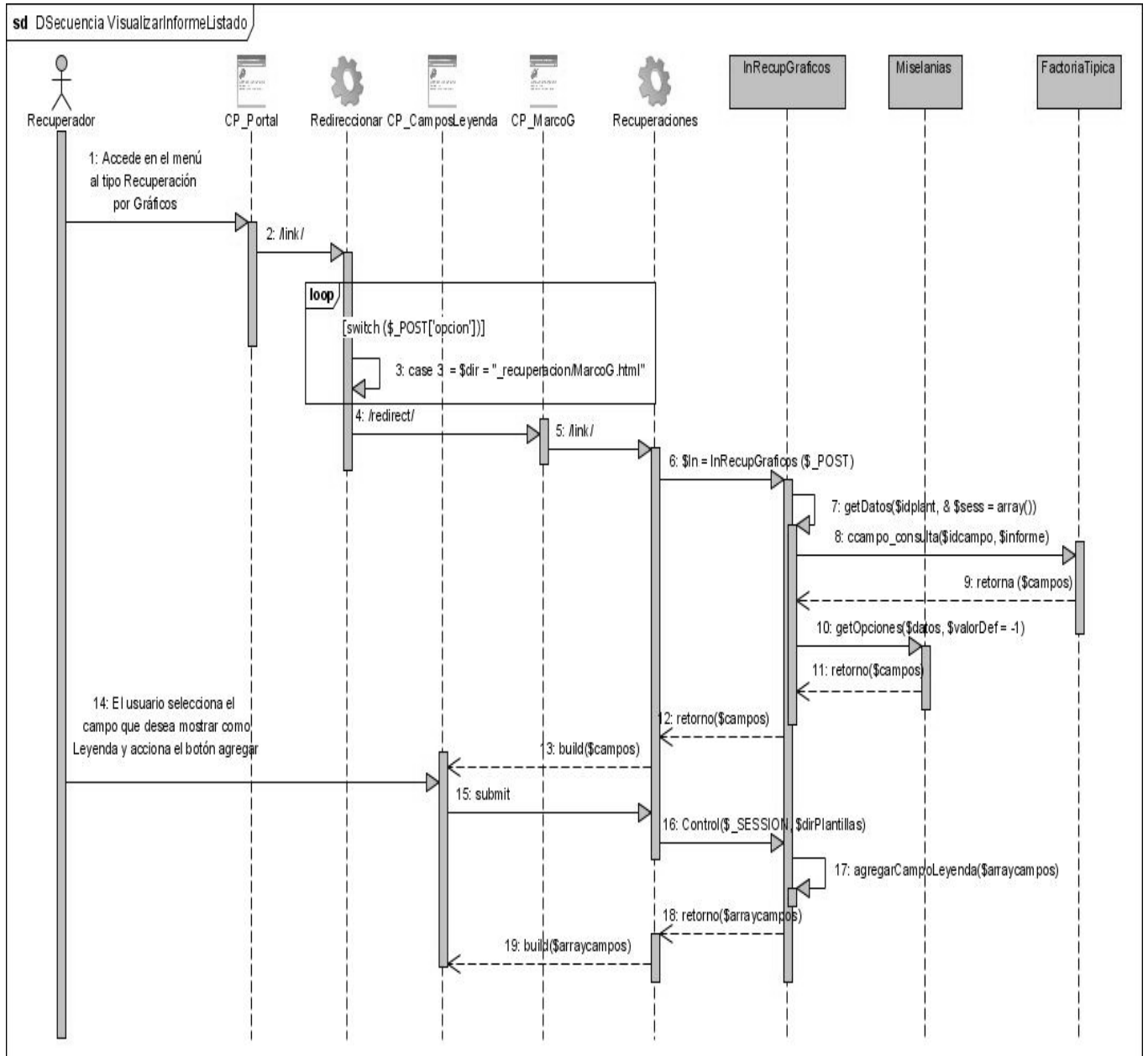
- Caso de Uso Visualizar Informe por Estadísticos

- Agregar Campos Fila



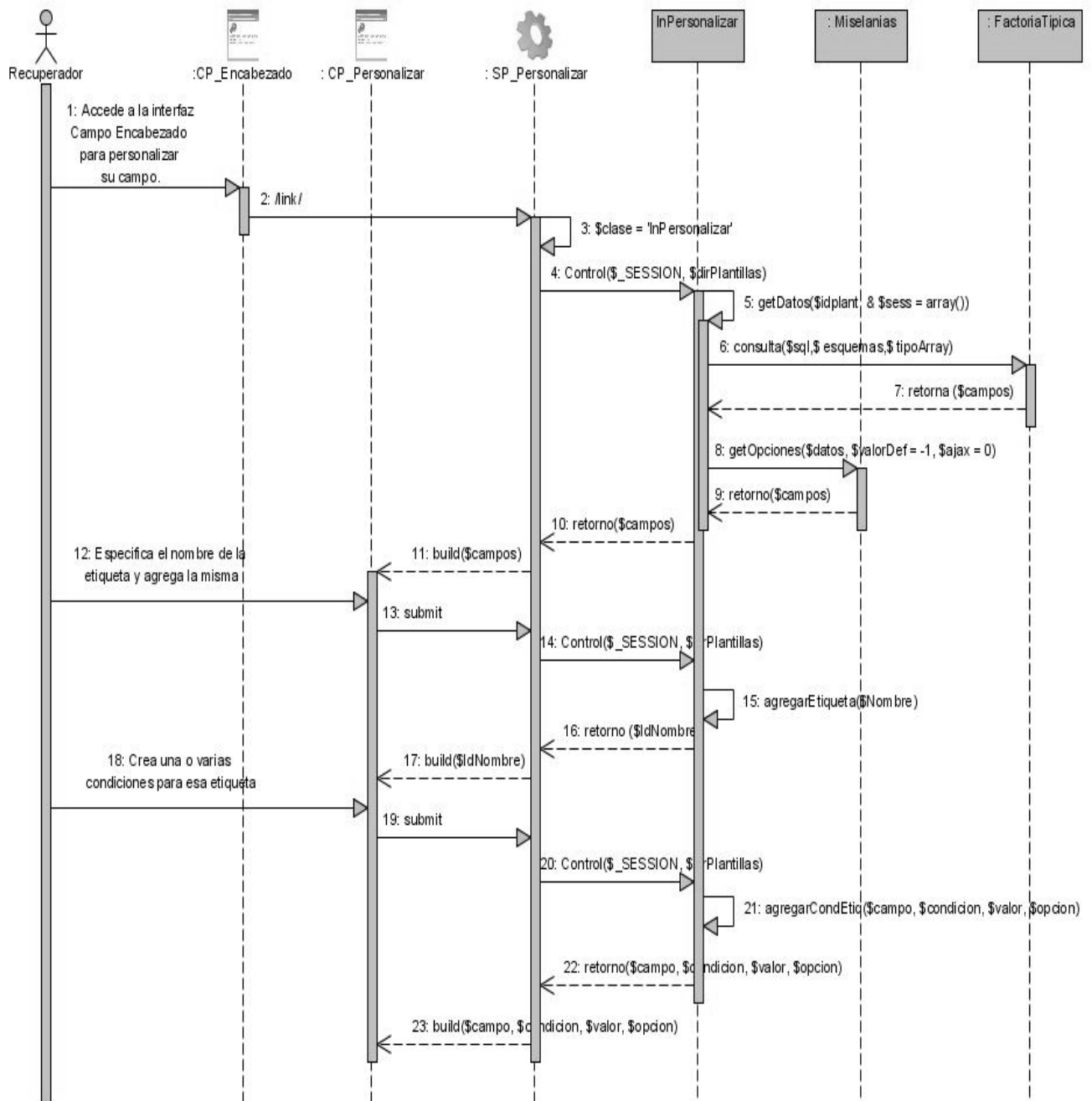
- Caso de Uso Visualizar Informe por Gráficos

- Escenario Agregar Campo Leyenda



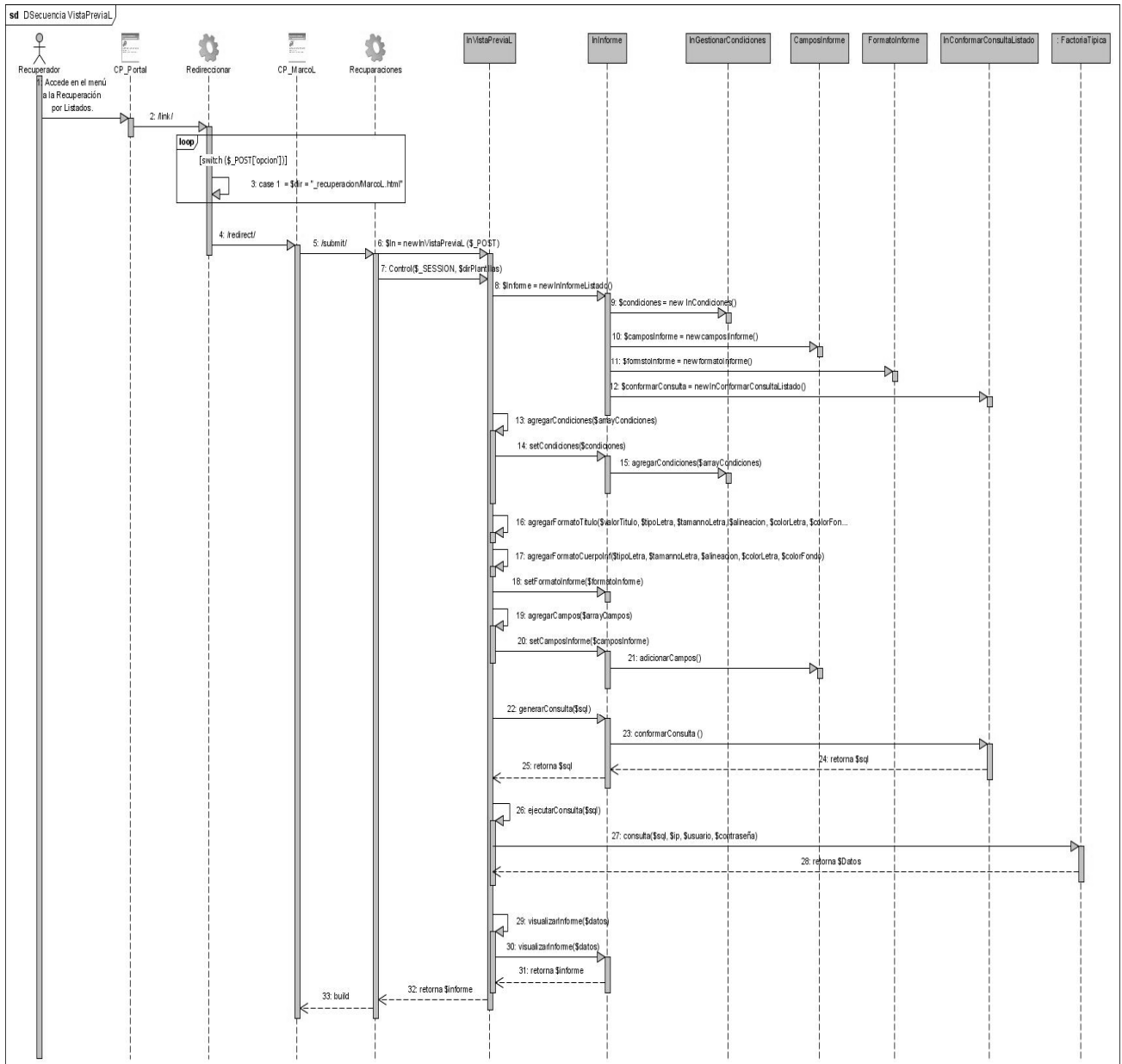
• Caso de Uso Personalizar

○ Escenario Agregar Personalización:

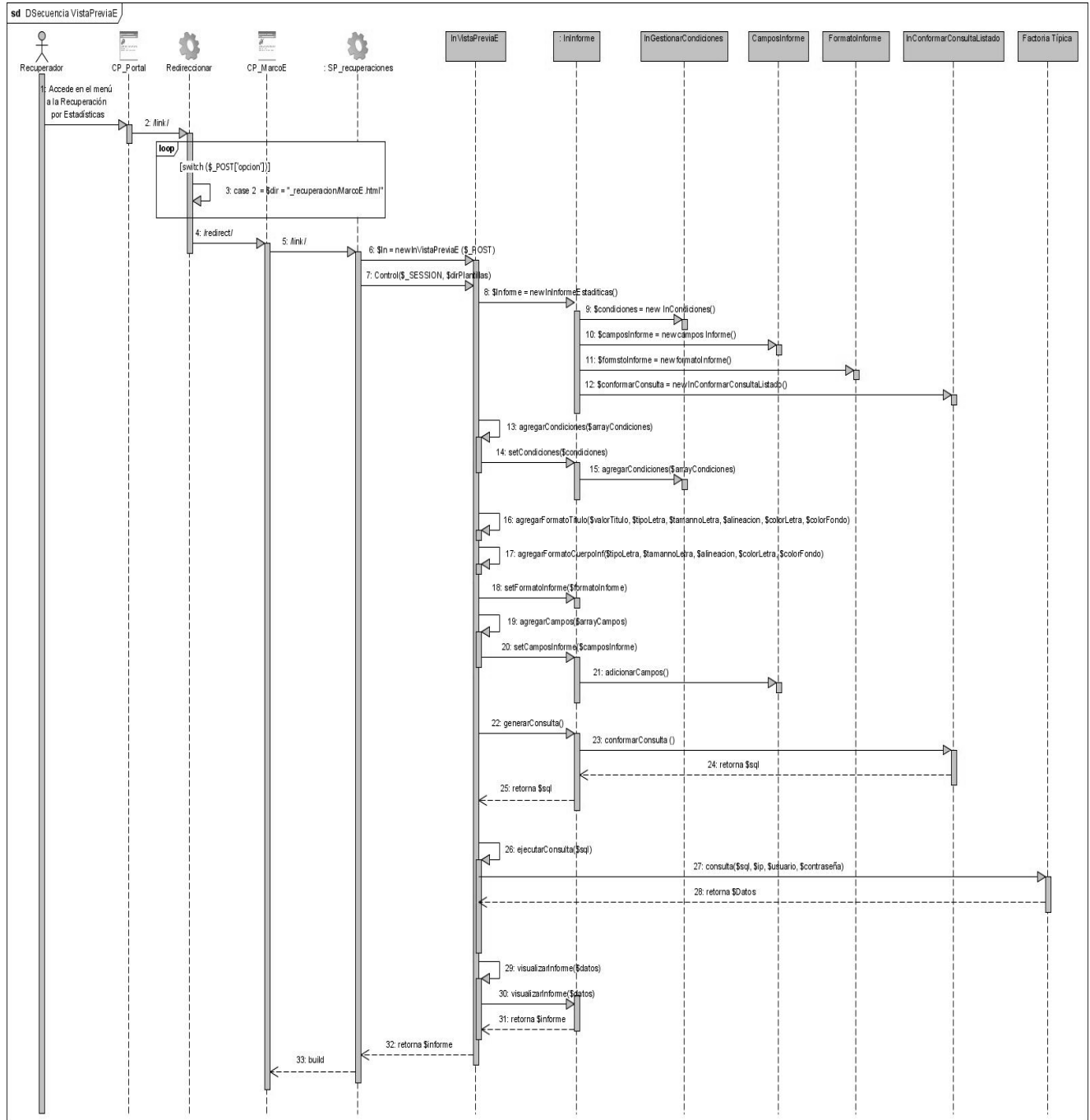


- Caso de Uso Generar Vista Previa

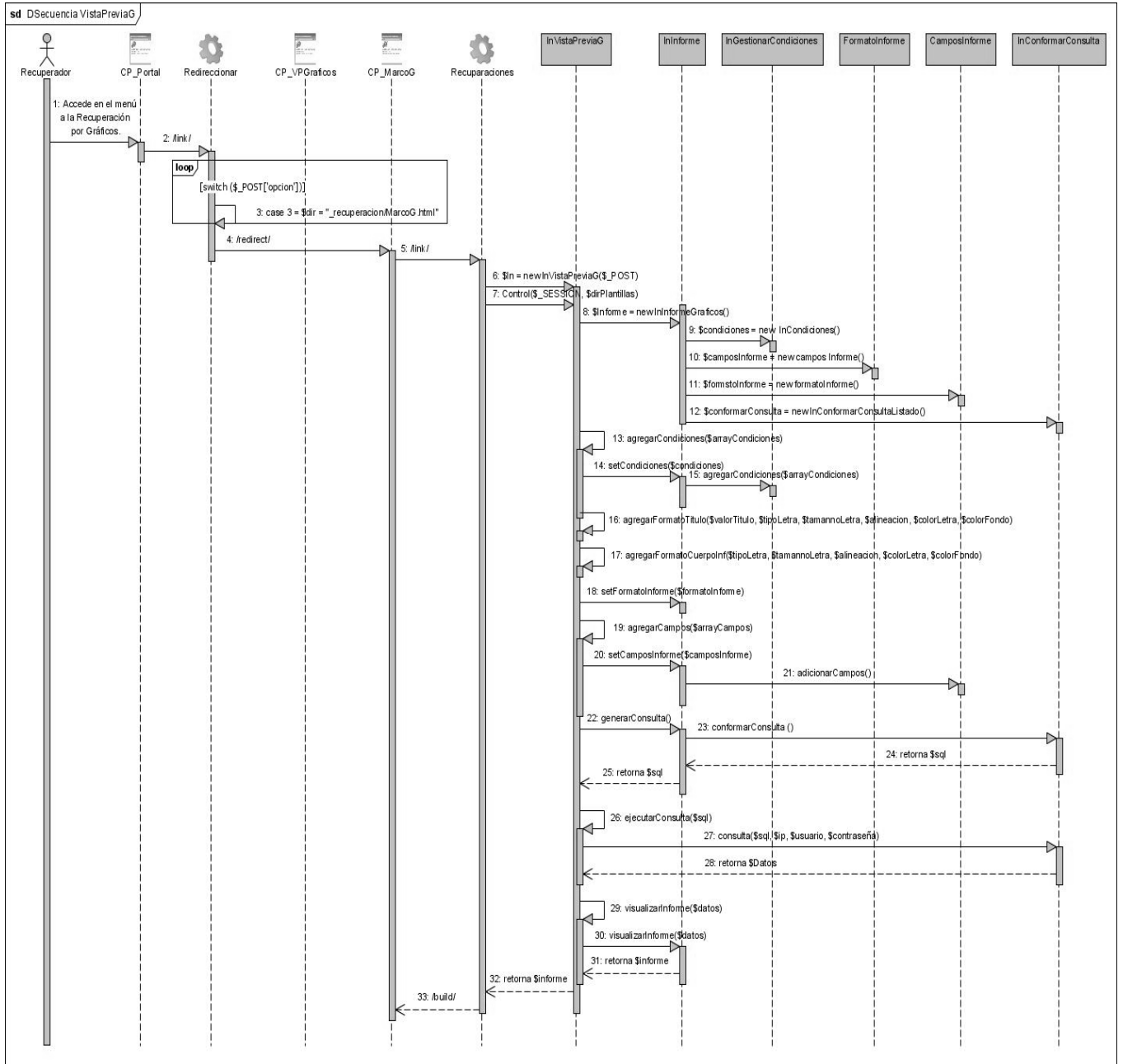
- Generar Vista Previa por Listado



o Generar Vista Previa por Estadísticas

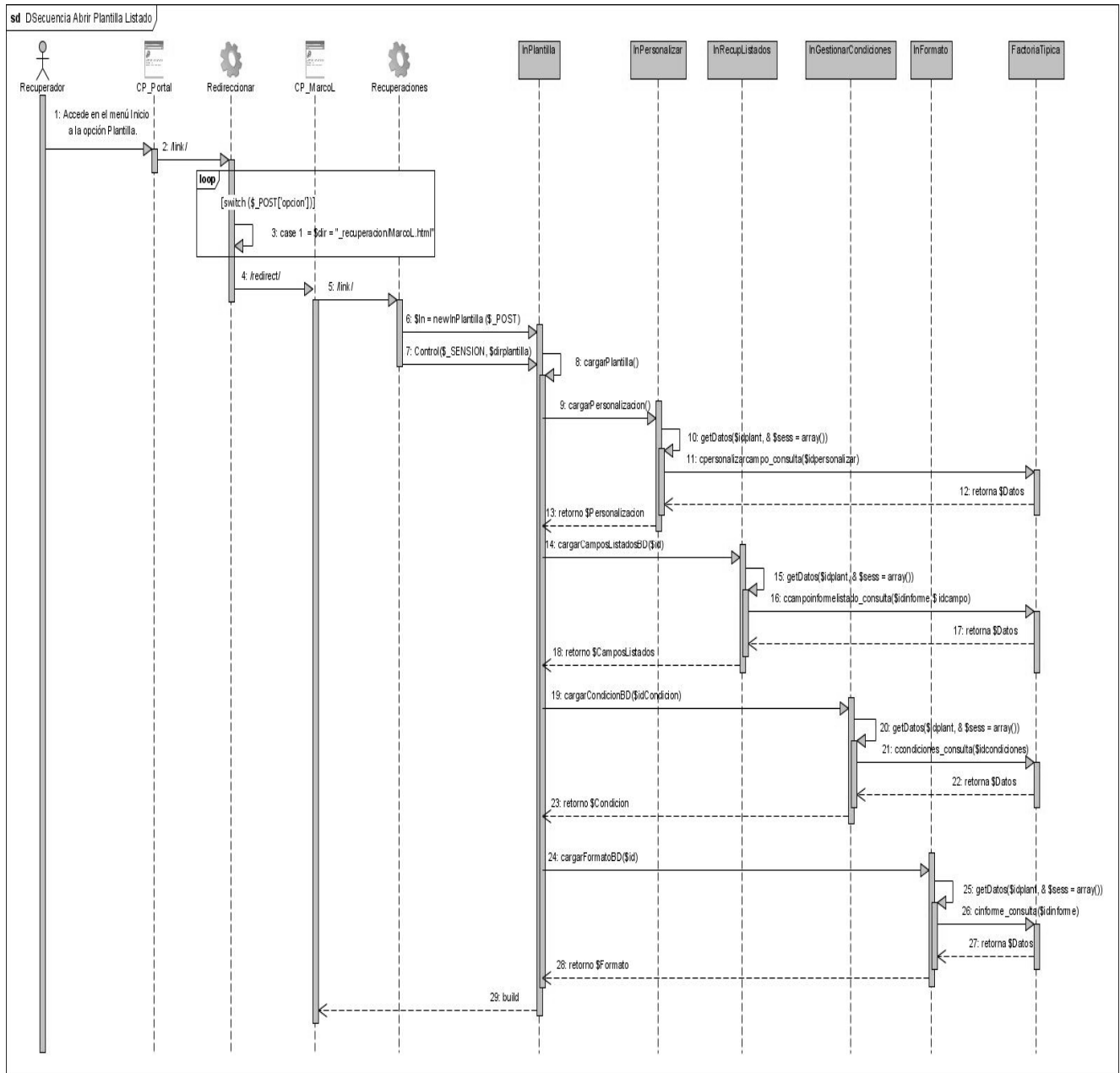


o Generar Vista Previa por Gráficos

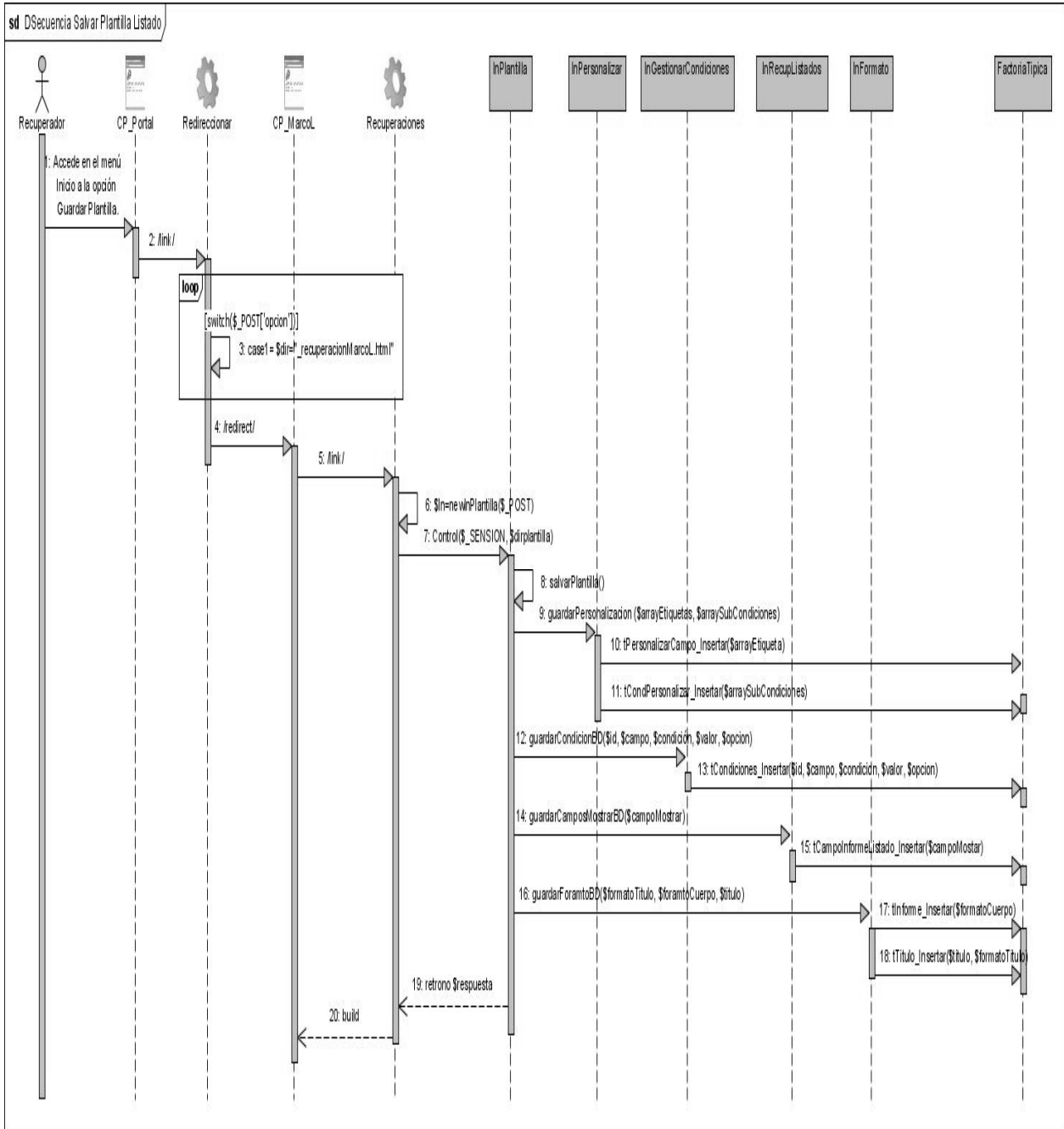


• Caso de Uso Gestionar Plantilla

○ Escenario Abrir Plantilla de Listado



o Escenario Salvar Plantilla de Listado



Nombre: dat_tabla		
Descripción: Se almacenan todas las tablas de una Base de Datos con sus respectivas características.		
Atributo	Tipo	Descripción
idtabla	integer	Atributo identificador de las tablas de la Base de Datos. (Llave primaria)
idesquema	integer	Atributo identificador que representa el esquema al que pertenece la tabla almacenada (llave foránea).
idtipotabla	integer	Atributo identificador que representa la relación entre la tabla y el tipo de tabla (llave foránea).
nombretabla	varchar	Nombre de la tabla que se almacenó.
descripcion	varchar	Breve información sobre la tabla que se almacenó.

Nombre: dat_relacion		
Descripción: Se almacenan todas las relaciones de una Base de Datos, además de las definidas por el usuario.		
Atributo	Tipo	Descripción
idcampo1	integer	Atributo identificador del primer campo que compone una relación (llave primaria).
idcampo2	integer	Atributo identificador del segundo campo que compone una relación, (llave primaria).
idtiporelacion	integer	El atributo identificador que representa la correspondencia entre la relación y el tipo de relación (llave foránea).
original	boolean	Si la relación fue guardada de la Base de Datos almacenada o fue creada por el usuario.

Nombre: dat_campo		
Descripción: Se almacenan todos los campos de una Base de Datos con sus respectivas características.		
Atributo	Tipo	Descripción
idcampo	integer	Atributo identificador de los campos de la Base de Datos (llave primaria).
idtabla	integer	Atributo identificador que representa la tabla a la que pertenece el campo

		almacenado (llave foránea).
idtipocampo	integer	Atributo identificador que representa la relación entre el campo y el tipo de campo (llave foránea).
nombrecampo	varchar	Nombre del campo que se almacenó.
descripción	varchar	Breve información sobre el campo que se almacenó.
etiqueta	varchar	Almacena el nombre que tomará el atributo campo.
longitud	varchar	Cantidad de caracteres que puede admitir el atributo campo.
posicion	varchar	Es el lugar que va a tener ese atributo campo en la tabla.
campodescriptor	varchar	El nombre del campo descriptor de la tabla nomenclador que se utiliza para decodificar el idcampo.
llave	boolean	Si el atributo campo es o no el identificador de una tabla.
dependencianomenclador	boolean	Si depende o no de un nomenclador para ser decodificado.

Nombre: dat_dependencia		
Descripción: Se almacenan todas las dependencias definidas por el usuario para facilitar la decodificación de la información.		
Atributo	Tipo	Descripción
idcampo1	integer	Atributo identificador del primer campo que compone la dependencia (llave primaria).
idcampo2	integer	Atributo identificador del segundo campo que compone la dependencia (llave primaria).

Nombre: dat_rol		
Descripción: Se almacenan todos los roles que pueden acceder a las aplicaciones existentes.		
Atributo	Tipo	Descripción
idrol	bigint	El atributo identificador de los roles que existen (llave primaria).
idmodulo	integer	El atributo identificador que representa el módulo al que pertenece el rol almacenado (llave foránea).

actual	bit(1)	Representa el estado en que se encuentra el rol.
--------	--------	--

Nombre: dat_modulos		
Descripción: Se almacenan todos los módulos que existen en las aplicaciones del proyecto.		
Atributo	Tipo	Descripción
idmodulo	bigint	Atributo identificador de los módulos que existen (llave primaria).
descripcion	varchar(50)	La descripción del módulo.
abreviatura	varchar(10)	La abreviatura del nombre del módulo.
padre	varchar(10)	Identificador del módulo superior.

Nombre: nom_tipotabla		
Descripción: Se almacenan los tipos de tablas que existen.		
Atributo	Tipo	Descripción
idtipotabla	integer	Atributo identificador del tipo de tabla (llave primaria).
nombretipotabla	varchar	Nombre del tipo de tabla que puede ser.

Nombre: nom_tipocampo		
Descripción: Se almacenan los tipos de campos que existen.		
Atributo	Tipo	Descripción
idtipocampo	integer	Atributo identificador del tipo de campo (llave primaria).
nombretipocampo	varchar	Nombre del tipo de campo que puede ser.

Nombre: nom_tiporelacion		
Descripción: Se almacenan los tipos de relaciones que existen.		
Atributo	Tipo	Descripción
idtiporelacion	integer	Atributo identificador del tipo de relación (llave primaria).
nombretiporelacion	varchar	Nombre del tipo de relación que puede ser.

Nombre: dat_campo_rol		
Descripción: Contiene los identificadores de las tablas dat_rol y dat_campo, ya que entre estas existe una relación de muchos a muchos.		

Atributo	Tipo	Descripción
idrol	integer	Atributo identificador de los roles (llave primaria).
idcampo	integer	Atributo identificador de los campos (llave primaria).

Nombre: dat_tabla_rol		
Descripción: Contiene los identificadores de las tablas seg_rols y dat_tabla, ya que entre estas existe una relación de muchos a muchos.		
Atributo	Tipo	Descripción
idrol	integer	Atributo identificador de los roles (llave primaria).
idtabla	integer	Atributo identificador de las tablas (llave primaria).

Nombre: dat_informe		
Descripción: Se almacenan los atributos o características que debe tener un informe.		
Atributo	Tipo	Descripción
idinforme	int	El atributo identificador del informe (llave primaria).
nombre	varchar	Nombre que debe tener el informe, cuando sea guardado.
tipoletra	varchar	Tipo de letra para el cuerpo de informe.
colorletra	varchar	Color de letra para el cuerpo del informe
tamaño letra	varchar	Tamaño de letra para el cuerpo del informe.
colorfondo	varchar	Color de fondo para el informe.
alineacion	varchar	Alineación para las letras.
tipoinforme	varchar	Tipos de Informes que existen, pueden ser por listados, estadísticas, y graficas.
fecha	date	Fecha en que se realizó el informe

Nombre: dat_orden		
Descripción: Se almacena los atributos que permiten ordenar los campos en un informe de listado.		
Atributo	Tipo	Descripción
idorden	int	Atributo identificador de orden (llave primaria).

idinforme	int	Atributo identificador de informe (llave primaria).
idcampo	int	Atributo identificador de campo (llave primaria).
orden	varchar	Atributo orden es el que permite ordenar los campos en el informe de listados.

Nombre: dat_grupo		
Descripción: Se almacenan los campos por los que se quiere agrupar en el informe		
Atributo	Tipo	Descripción
idgrupo	int	El atributo identificador de grupo (llave primaria).
idinforme	int	El atributo identificador de informe (llave foránea).
idcampo	int	Atributo identificador de campo (llave foránea).

Nombre: dat_campoinformelistado		
Descripción: Se almacenan los atributos o características que va tener un campo del Informe Listado.		
Atributo	Tipo	Descripción
idinforme	int	El atributo identificador de informe (llave primaria).
idcampo	int	El atributo identificador de campo (llave primaria).
etiqueta	varchar	Nombre que se le puede poner a un campo seleccionado desde la misma Base de Datos.
colorletra	varchar	Color de la letra para un campo determinado.
colorfondo	varchar	Color de fondo para un campo determinado.
alineacion	varchar	Alineación para un campo determinado.

Nombre: dat_condicion		
Descripción: Se almacenan las condiciones que debe tener un campo seleccionado de la Base de Datos.		
Atributo	Tipo	Descripción
idinforme	int	Atributo identificador de informe (llave

		foránea).
idcondicion	int	El atributo identificador de condición (llave primaria).
idcampo	int	Atributo identificador de campo (llave foránea).
condicion	varchar	Condición que se le pone a un campo
valor	varchar	El valor de la condición
opcion	varchar	Opción de la condición (Y, O, FIN)
valorcodigo	varchar	Valor del identificador del campo cuando depende de un nomenclador

Nombre: dat_campoInformeestadistico		
Descripción: Se almacenan los atributos que permiten la selección de campos para obtener un Informe Estadístico.		
Atributo	Tipo	Descripción
idinforme	int	Atributo identificador de informe (llave primaria).
idcampo	int	Atributo identificador de campo (llave primaria).
tipocampo	varchar	Tipo de campo puede ser encabezado, columna y fila.
numerofila	byte	Se especifica la cantidad de filas que tendrá el informe
total	bool	Este atributo te permite calcular el total del campo seleccionado.
colorfondototal	varchar	Color de fondo para el total de algún campo seleccionado.
colorletratotal	varchar	Color de letra para el total de algún campo seleccionado.
personalizar	bool	Se especifica si un campo está personalizado o no.

Nombre: dat_personalizarcampo		
Descripción: Se almacenan los atributos que permiten renombrar un campo determinado.		
Atributo	Tipo	Descripción
idinforme	int	Atributo identificador de informe (llave primaria).
idcampo	int	Atributo identificador de campo (llave primaria).
posicion	byte	Posición que tendrá la etiqueta en el

		informe
etiqueta	varchar	El nombre que se le puede poner a un campo seleccionado desde la misma Base de Datos.

Nombre: dat_condicionpersonalizar		
Descripción: Se almacenan los atributos que permiten personalizar las condiciones de un campo determinado.		
Atributo	Tipo	Descripción
idpersonalizar	int	Atributo identificador (llave primaria).
condicion	varchar	La condición se le pone a un campo que se va a personalizar.
valor	varchar	El valor que se le pone a un campo que se va a personalizar.
opcion	varchar	La opción se le pone a un campo que se va a personalizar.
valorCodigo	varchar	Valor del identificador del campo cuando depende de un nomenclador

Nombre: dat_titulo		
Descripción: Se almacenan los atributos que permiten el formato del titulo del informe.		
Atributo	Tipo	Descripción
idinforme	int	Atributo identificador de informe (llave primaria).
titulo	varchar	Título del informe
tipoletra	varchar	El tipo de letra del titulo del informe.
colorletra	varchar	El color de la letra del titulo del informe.
tamaño	varchar	El tamaño del titulo del informe.
alineacion	varchar	La alineación del titulo del informe.
colorfondo	varchar	El color de fondo del titulo del informe.

Nombre: dat_campoinformegrafico		
Descripción: Se almacenan los atributos que permiten la selección de campos para obtener un Informe Gráfico.		
Atributo	Tipo	Descripción
idinforme	int	Atributo identificador de informe (llave primaria).
idcampo	int	Atributo identificador de campo (llave primaria).
tipocampo	varchar	Los tipos de campo de gráficos, como campo dato, Campos EjeX y campos

		Leyenda.
titulo	varchar	El titulo que se le puede poner al campo seleccionado en campo dato y campos Ejes.
incremento	varchar	Incremento de campo dato.

3.5 Principios de diseño

Este sistema esta orientado a los usuarios finales, por lo que consideramos que uno de los puntos fundamentales que tenemos que tener en cuenta es el diseño de la interfaz, ya que esta es la que va a representar la interacción del usuario con el sistema, por lo que debe ser lo más amigable y comprensible posible. Una aplicación con una interfaz bien diseñada debe tener, además de un buen diseño gráfico, una buena navegabilidad, usabilidad y distribución de los contenidos.

De manera general se puede decir que los usuarios finales no están familiarizados con conocimientos informáticos, por lo que no les interesa el funcionamiento interno de la aplicación, sino que están interesados en la manera de cómo se muestran las funcionalidades de esta y como se debe usar.

Para el diseño de la interfaz de usuario de este sistema se han seguido los siguientes principios:

- Permitir su utilización desde el primer momento, por cualquier persona que tenga un mínimo dominio de la informática.
- Requerir de los usuarios un mínimo esfuerzo para alcanzar sus objetivos.
- Cada elemento se diseñará siguiendo un patrón de tamaño, colores y formas
- No se utilizarán colores fuertes ni brillantes.
- Permitir al usuario acceder solamente a las opciones a las que, dado su rol, puede ejecutar.

3.6 Tratamiento de errores

El tratamiento de errores posibilita el buen funcionamiento de una aplicación dándole una mejor apariencia ante los usuarios. En nuestro proyecto se ha realizado un fuerte trabajo con el fin de identificar y controlar los posibles errores que se pueden presentar a la hora de interactuar con el software.

Para prevenir errores por parte del usuario buscamos que en los formularios el usuario introduzca la menor cantidad posible de datos, evitando así incoherencias e incorrecciones, en el caso de la entrada de datos por parte del usuario se implementarán funciones que validen dicha entrada para que, de existir errores, se muestren mensajes que ilustren la incorrecta inserción, modificación o mala manipulación de datos en general.

Mediante la validación en el lado del cliente, utilizando el lenguaje JavaScript, se garantiza que los datos suministrados por los usuarios, se almacenen íntegros y no existan inconsistencias. Para esto, se verifican los campos obligatorios, y se revisa el tipo de datos, mostrándose en caso de algún error, mensajes de alerta.

Otro tipo de error que puede ocurrir son los que no pueden ser detectados en la parte del cliente, pues ocurren internamente en la aplicación. En este caso se muestra un mensaje que sea capaz de orientar al usuario en su próxima acción.

3.7 Conclusiones

En este capítulo se representaron los diagramas de clases y diagramas de interacción del diseño, donde se mostraron las relaciones de las diferentes clases que componen nuestro sistema. Se definieron las clases persistentes y a partir de esto se construyó el modelo de datos.

CAPÍTULO 4 IMPLEMENTACIÓN Y PRUEBA

4.1 Introducción

En este capítulo se continúa con la modelación de la solución para el sistema que se desea realizar. Se crea el modelo de despliegue que corresponde con la distribución física del sistema. Además se crean los diagramas de componentes que contribuirán a una mejor descripción de la solución y se diseñan los casos de prueba que servirán de guía en el refinamiento de los mismos y su futura ejecución.

4.2 Modelo del despliegue

El modelo de despliegue es un modelo de objeto que describe la distribución física del sistema en términos de cómo se distribuye la funcionalidad entre los nodos de cómputo. [R1]

El modelo de despliegue representa la correspondencia de entre el software y el hardware. A continuación se representa el modelo de despliegue definido para este sistema.

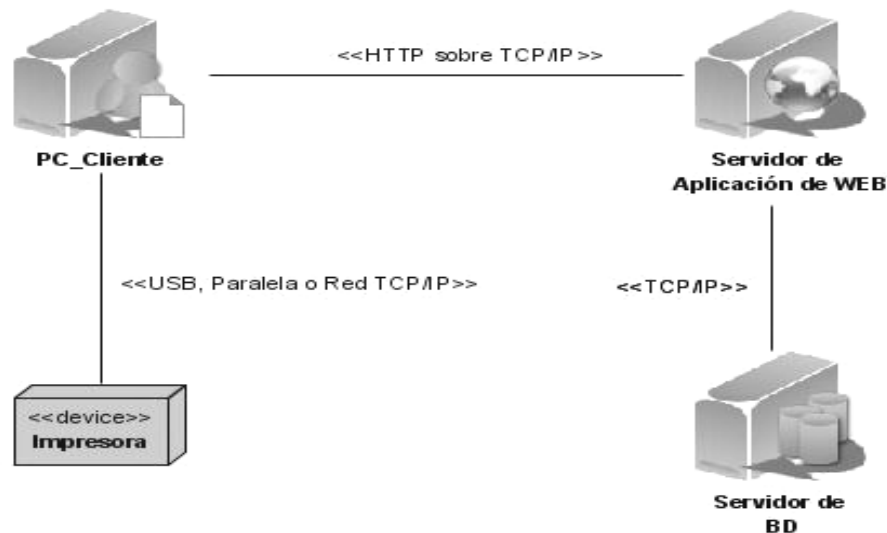


Figura 4.1: Modelo del Despliegue.

4.3 Modelo de implementación

El modelo de implementación describe cómo se implementan los elementos del modelo de diseño en términos de componentes. Describe también como se organizan y se relacionan unos con otros, definiendo un componente como el empaquetamiento físico de los elementos de un modelo, como es el caso de las clases del modelo de diseño, además sirve de base en las labores de construcción del sistema.

Para lograr una mejor comprensión de los componentes que forman nuestro sistema, presentaremos primeramente un diagrama de paquetes de componentes los cuales representan una división lógica de varios componentes según su funcionalidad.

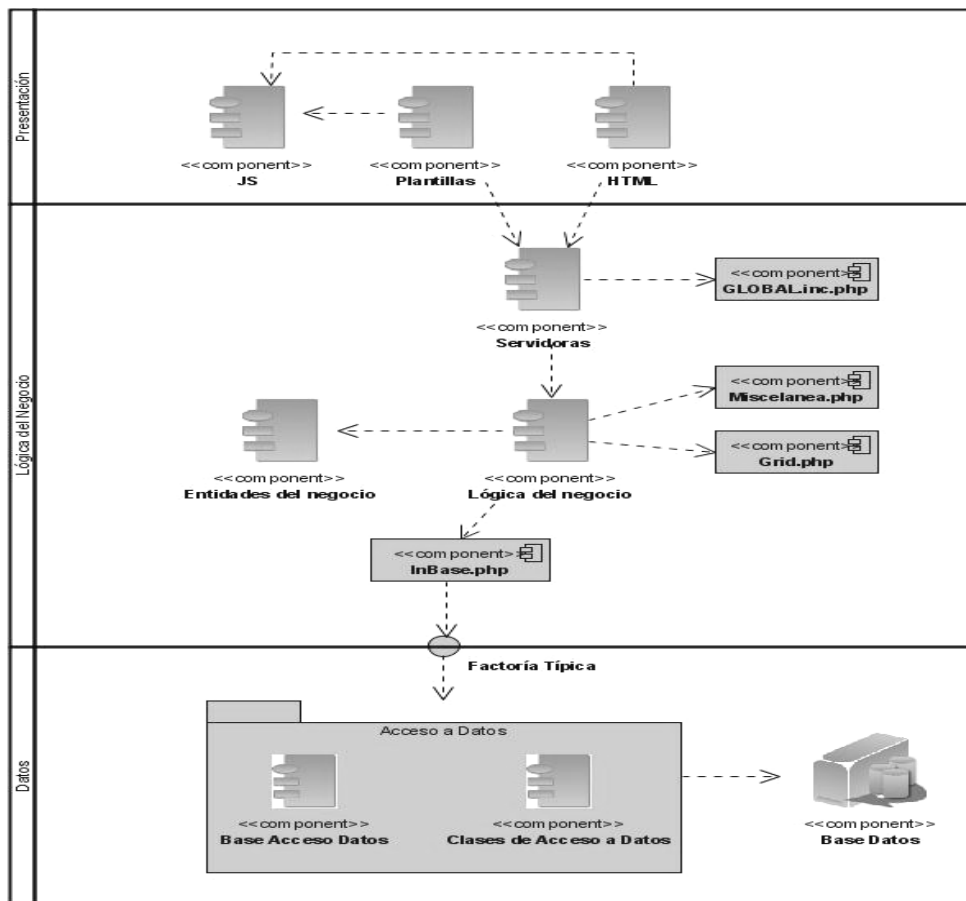


Figura 4.2: Diagrama de Componente General

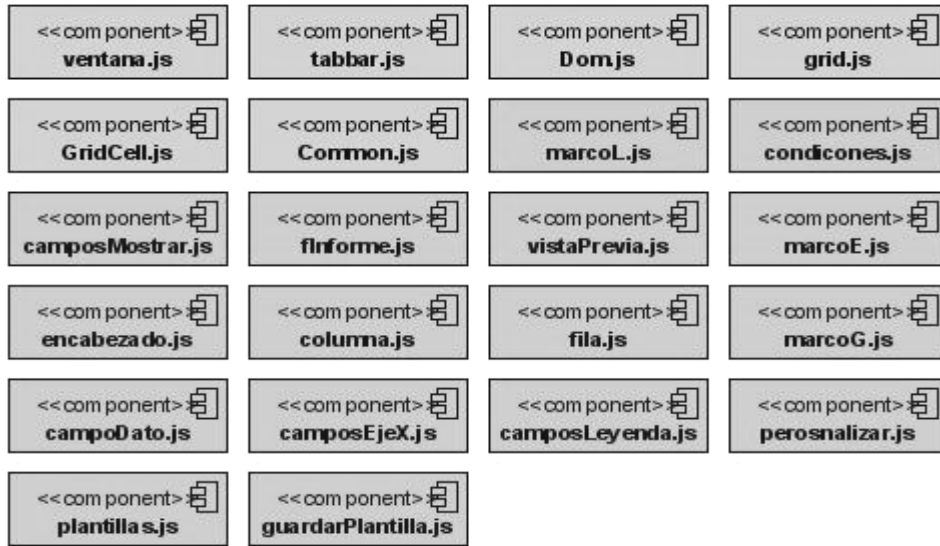


Figura 4.3: Diagrama del paquete de los js.

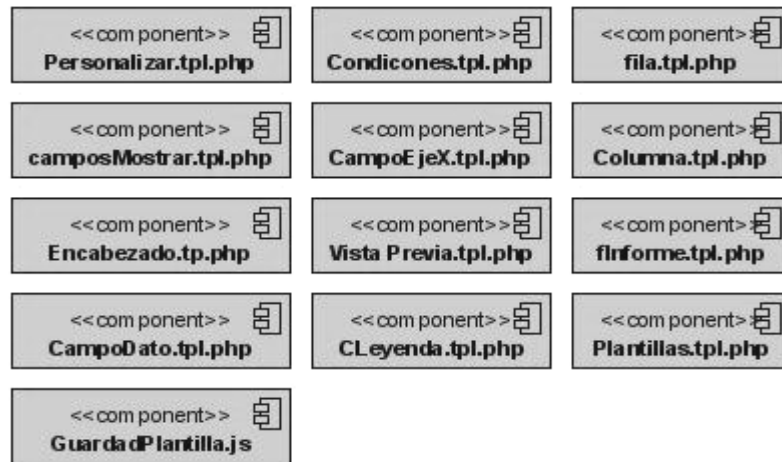


Figura 4.3: Diagrama del paquete de las plantillas.

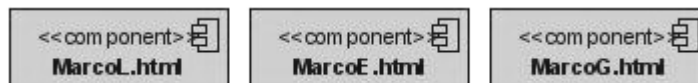


Figura 4.4: Diagrama del paquete de las html.

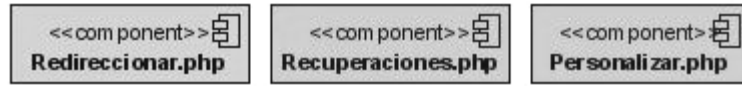


Figura 4.5: Diagrama del paquete de las Servidoras.

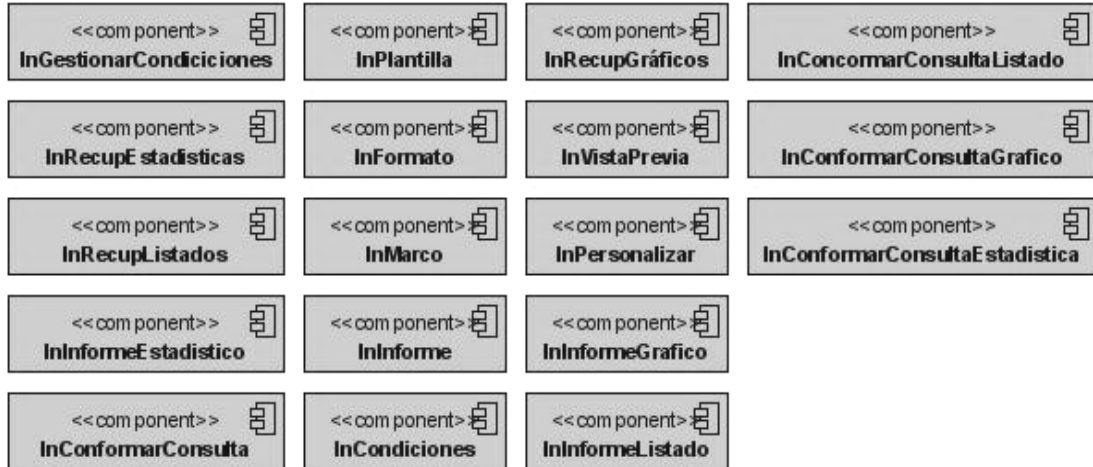


Figura 4.6: Diagrama del paquete de las Lógicas del Negocio.

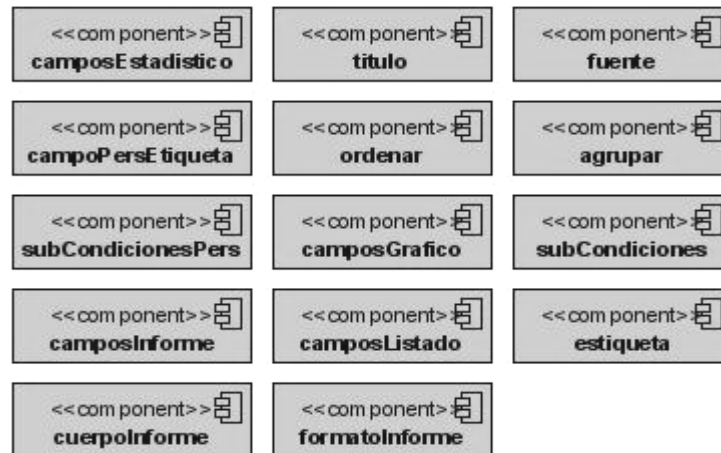


Figura 4.7: Diagrama del paquete de las Entidades del Negocio

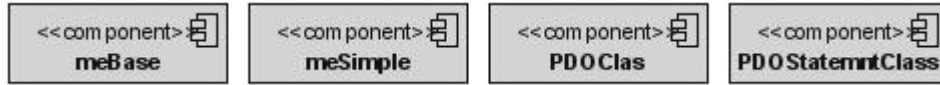


Figura 4.7: Diagrama del paquete de la Base de Acceso a Datos.

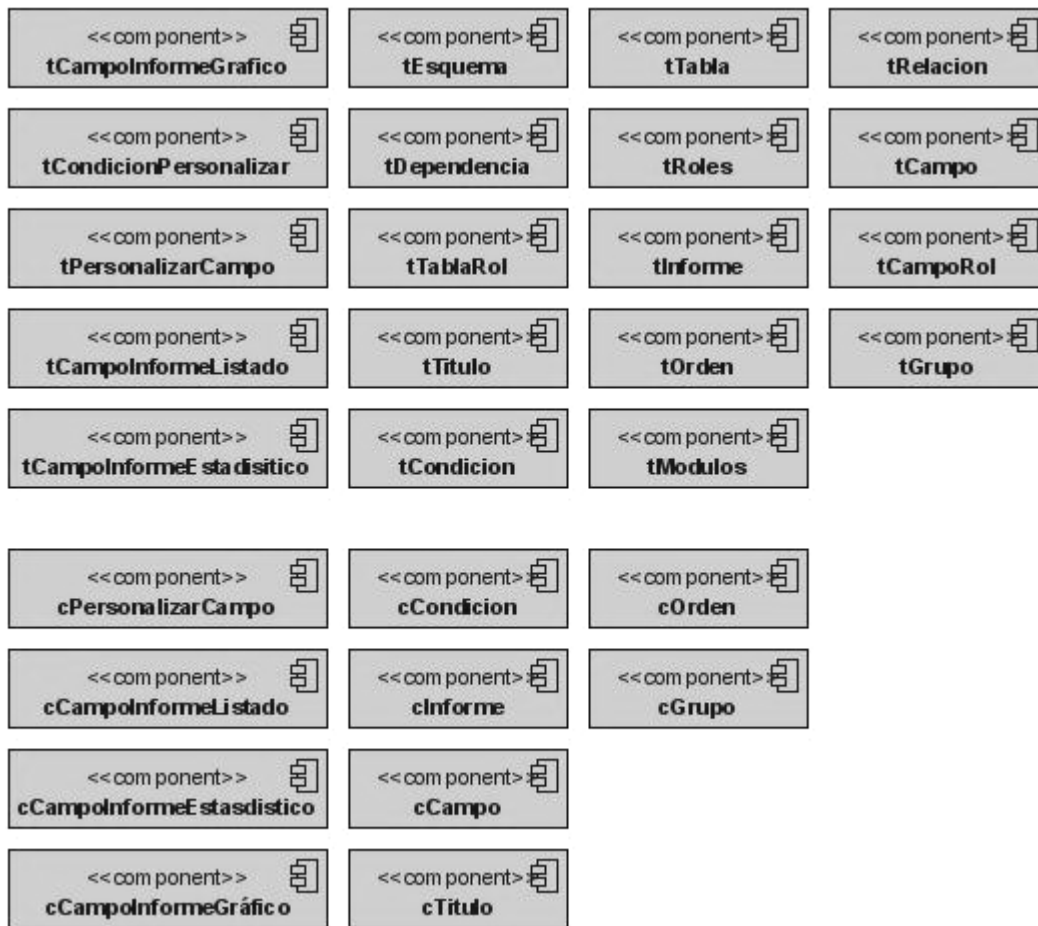


Figura 4.8: Diagrama del paquete de las Clases de Acceso a Datos

4.4 Diseño de Casos de Prueba

Las pruebas deben planificarse mucho antes de que empiecen. La planificación de las pruebas puede empezar tan pronto como se haya completado la captura de requisitos, donde será posible la realización de un diseño de casos de prueba que aunque todavía rústico servirá como base para el refinamiento de estos y su futura ejecución. La definición detallada de los casos de prueba podrá empezar entonces tan pronto como el modelo de diseño se haya consolidado. Por tanto, se pueden planificar y diseñar todas las pruebas antes de generar algún código.

A continuación presentamos algunos de los diseños de casos de prueba, lo restantes se pueden encontrar en el (ANEXO II).

Diseño de los Casos de Pruebas: Definir Formato

Descripción General

El caso de uso consiste en definir el formato con el que se mostrará el informe de la recuperación de información que se va a realizar. Es iniciado por el Recuperador y termina con la definición correcta de un formato para el informe de la recuperación.

Las pruebas realizadas a este caso de uso son las siguientes:

1. Definir el título que se mostrara en el informe y su formato
2. Definir formato del cuerpo del informe que se mostrará como resultado de la recuperación

1. CPR1: Definir el título que se mostrará en el informe y su formato

1.1 Descripción

Este caso de prueba permite especificar un título general para el informe y su formato

1.2 Flujo Central

- El sistema muestra la interfaz de la aplicación

- El Recuperador selecciona en el menú la interfaz de recuperación (ya sea por listado, por gráfico o por estadística)
- El Recuperador accede al **tab** correspondiente a la interfaz de Formato
- El usuario debe seleccionar los siguientes datos:
 - Título
 - Tipo de Letra
 - Color de Letra
 - Tamaño de letra
 - Color de Fondo
 - Alineación

1.3 Condiciones de Ejecución

- Debe estar autenticado como Recuperador del sistema

Clases Válidas	Clases Inválidas	Resultado Esperado
Acceder a la interfaz de recuperaciones, ya sea recuperaciones por listado, por estadística o por gráfico		El sistema muestra la interfaz de las recuperaciones
Acceder al tab de Formato		El sistema muestra la interfaz de Formato
Si el recuperador no selecciona alguno de los valores siguientes: Título Tipo de Letra Color de Letra Tamaño de letra Color de Fondo Alineación		El sistema especificará un valor predeterminado.
	Si no se especifica el título	

2. CPR1: Definir el título que se mostrará en el informe y su formato

2.1 Descripción

Este caso de prueba permite especificar el formato del cuerpo del informe que se mostrará después de una recuperación por listado, por estadística o por gráfico.

2.2 Flujo Central

- El Recuperador debe seleccionar los siguientes datos en la interfaz Formato, en la parte de formato del cuerpo del informe:
 - Título
 - Tipo de Letra
 - Color de Letra
 - Tamaño de letra
 - Color de Fondo
 - Alineación

2.3 Condiciones de Ejecución

- Debe estar autenticado como Recuperador del sistema

Clases Válidas	Clases Inválidas	Resultado Esperado
Acceder al tab de Formato		El sistema muestra la interfaz de Formato
Si el recuperador no selecciona alguno de los valores siguientes: Título Tipo de Letra Color de Letra Tamaño de letra Color de Fondo Alineación		El sistema especificará un valor predeterminado.

4.5 Conclusiones

En este capítulo se modeló la estructuración de la aplicación físicamente, mediante el diagrama de despliegue, que representa la distribución física del sistema, y el diagrama de componentes, que jugará un papel fundamental en la construcción del mismo, además se realizó una primera iteración del diseño de casos de prueba que servirán de guía en el refinamiento de los mismos y su futura ejecución.

Conclusiones Generales

Luego del estudio realizado se logró el diseño del sistema que servirá como base para futuras labores de implementación, además, con este trabajo, se brinda una guía para familiarizar a implementadores, clientes y usuarios finales con los conceptos y actividades relacionadas con la recuperación de información de forma dinámica.

Para la realización de este trabajo se investigaron los procesos que se llevan a cabo en la recuperación y visualización de información, además de realizarse un análisis de las tecnologías de desarrollo utilizadas.

La modelación del sistema se desarrolló siguiendo la metodología RUP, y se utilizó el lenguaje de modelado UML para la modelación de las diferentes fases por las que transitó el desarrollo de este trabajo. Se modeló el dominio, se definieron los requerimientos del sistema, tanto funcionales como no funcionales, estructurándose además, el modelo de casos de uso del sistema.

Se realizó el análisis y diseño del sistema a través de diagramas de clases del análisis, diagramas de clases del diseño, de interacción, de clases persistentes, entre otros. Finalmente se elaboraron el modelo de datos, el de despliegue y el de implementación.

Los objetivos propuestos al comienzo de este trabajo se cumplieron satisfactoriamente, además se han incluido una serie de recomendaciones que deben tenerse en cuenta para el trabajo futuro.

Recomendaciones

Al concluir este trabajo se recomienda lo siguiente para versiones futuras:

- Continuar el estudio de los mecanismos de recuperación de información de forma dinámica
- Continuar el desarrollo de este sistema y agregarle nuevas funcionalidades adecuándolo aun más a las demandas de los usuarios haciéndolo más útil y provechoso
- Implementar lo que se ha modelado en este trabajo con el fin de acoplar el resultado final a las aplicaciones Web que necesiten recuperar información

Bibliografía

Citada

- [1] KRONOS, J. T. Concepto de recuperación de información, 2007.
<http://tramullas.com/documatica/3-1.html>
- [2] López, Carlos. Qué es la estadística, qué tipos tiene, qué objetivos persigue y algunas definiciones de sus parámetros, 2006.
<http://www.gestiopolis.com/recursos/experto/catsexp/pagans/eco/44/estadistica.htm>
- [3] Fermín, Yoniel. Representación gráfica de estadísticas. Cartogramas. Histogramas. Dispersogramas. Pictogramas, 2006.
<http://html.rincondelvago.com/graficos-estadisticos.html>
- [4] Lozano Terrazas, Víctor. ALGORITMO DE DIJKSTRA, abril 2007
<http://www.dma.fi.upm.es/java/matematicadiscreta/Dijkstra/home.html>
- [5] Manuel Tupia, D. M., Un algoritmo voraz para resolver el problema de la programación de tareas dependientes en máquinas diferentes. Facultad de Ingeniería de Sistemas e Informática. 2007, Universidad Nacional Mayor de San Marcos.
- [6] Corresponde a: Microsoft Office Excel 2003. ¿Qué es Microsoft Query?, 2007
<http://office.microsoft.com/es-es/excel/HP052747513082.aspx?pid=CH062528423082>
- [7] Mañón Ramírez, Enriquillo. Lo que debo saber sobre Crystal Reports y VB .NET, 2007.
<http://www.microsoft.com/spanish/msdn/comunidad/mtj.net/voices/art192.asp>
- [8] Jhon, R. Agata Report: un reporteador libre para las DB más comunes, diciembre 2006
<http://alts.homelinux.net/libreapp.php?id=408>
- [9] SPSS, Reportes OLAP: Opciones Tecnológicas Para Quienes Toman Decisiones, enero 2007.
<http://www.spss.com/la/whitepapers/Opciones%20Tecnologicas%20para%20Tomadores%20de%20Decisiones.pdf>
- [10] Vegas, Jesús. Desarrollo de Aplicaciones Web, enero- 2007.
<http://www.infor.uva.es/~jvegas/cursos/buendia/pordocente/node17/>
- [11] Marco, Cristina. La importancia del Software Libre en el entorno educativo, agosto 2003
<http://bulimages.bulma.net/articulos.php>
- [12] Álvarez, Miguel A. Qué es javascript, enero 2007

<http://www.desarrolloweb.com/articulos/25.php>

[13] Web dinámicas con PHP, noviembre 2006

<http://ascii.eii.us.es/docs/2002-03/php/php4.html>

[14] Sistemas De Bases de Datos, abril 2007

<http://usuarios.lycos.es/cursosgbd/UD2.htm>

[15] Van Der Henst, Christian. Las novedades de Dreamweaver 8, abril 2007

<http://www.maestrosdelweb.com/actualidad/2334/>

Consultada

- Adad, M.T., *Algoritmos Voraces*. 2005
- Chappell, David. Desarrollo de aplicaciones distribuidas con .NET, mayo 2007
<http://www.microsoft.com/spanish/msdn/articulos/archivo/081102/voices/dncapas.asp>
- Kon, Marcos D. El Software Libre, marzo 2007.
<http://www.monografias.com/trabajos12/elsoflib/elsoflib.shtm>
- Nitot, Tristan. FireFox 2, marzo 2007
<http://www.mozilla-europe.org/es/products/firefox/>
- Visual Paradigm, mayo 2007
<http://www.visual-paradigm.com>
- PostgreSQL, enero 2007
<http://manuales.gfc.edu.co/postgres/es/postgres.html>
- N. M. Soto López, Y.S.R., Sistema de Catalogación y Recuperación de Recursos de Información, HUBBLE. 2007, Instituto Superior Politécnico “José Antonio Echeverría” Ciudad de la Habana.
- Pressman, Roger S. Ingeniería del Software. Un enfoque práctico, 1999, pp. : 601
- Larman, C. UML Y PATRONES, Introducción al análisis y diseño orientado a objetos, 2007.
- Booch, G., Rumbaugh, J., Jacobson, I. El Proceso Unificado de Desarrollo de Software, 2007.
- Álvarez, Miguel Ángel. *¿Qué es XML?*, enero 2007.
<http://www.desarrolloweb.com/articulos/449.php>.

Glosario de términos

Recuperación por Listado: recuperación de datos en forma de listas de elementos.

Recuperación Estadística: recuperación de datos, generalmente numéricos, que se representarán en tablas.

Recuperación Gráfica: recuperación de datos que se mostrará n en un gráfico (gráficos pastel, barras, líneas, áreas).

Campo encabezado: campo que se recuperará de una base de datos y que nos permite mostrar los resultados por nivel de agrupamiento.

Campo Columna: campo que se recuperará de una base de datos y que sus valores se mostrarán como columnas de una tabla.

Campo Fila: campo que se recuperará de una base de datos y que sus valores se mostrarán como filas de una tabla.

Campo Dato: campo que se mostrará en el área de datos, en el caso de gráficos de barra, línea y área se corresponde con el eje Y, y en el caso de un gráfico de pastel se corresponde con lo que se muestra como área circular.

CSS: Referencia al lenguaje de hojas de estilo en cascada (Cascading Style Sheets). Es utilizado para definir la presentación de un documento HTML o XML.

HTML: Acrónimo inglés de HyperText Markup Language (lenguaje de marcas hipertextuales), lenguaje de marcación diseñado para estructurar textos y presentarlos en forma de hipertexto, que es el formato estándar de las páginas web.

PDF: Del inglés **P**ortable **D**ocument **F**ormat, Formato de Documento Portátil, es un formato de almacenamiento de documentos, desarrollado por la empresa Adobe Systems. Está especialmente ideado para documentos posibles a ser impresos, ya que especifica toda la información necesaria para la presentación final del documento.

CSV: Del inglés *comma-separated values*, son un tipo de documento sencillo para representar datos en forma de tabla, en las que las columnas se separan por comas, o punto y coma, y las filas por saltos de línea.