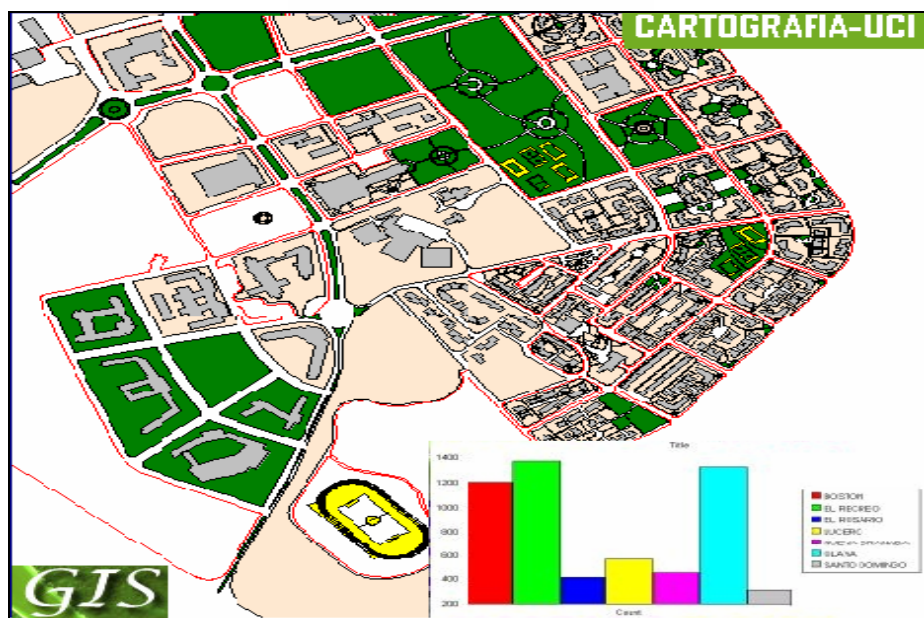
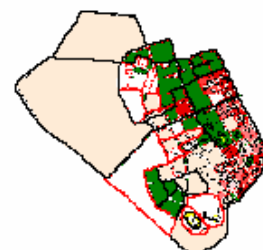


UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS
DIRECCIÓN DE INFORMATIZACIÓN



**Módulo Cartográfico del
Sistema de Información Geográfico de la UCI.
Gráficos Estadísticos y Mapas Temáticos**



**TRABAJO DE DIPLOMA PARA OPTAR POR EL TÍTULO DE
INGENIERÍA INFORMÁTICA**

**Autor: Edgar Rojas Ricardo
Tutor: Ing. Luís Lamela Fung**

Ciudad de La Habana, Abril 2006

DECLARACIÓN DE AUTORÍA

Por este medio se declara que Edgar Rojas Ricardo es el único autor de este trabajo y autoriza 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 6 días del mes de Abril del 2006.

Firma del Autor

Firma del Tutor

OPINIÓN DEL USUARIO DEL TRABAJO DE DIPLOMA.

El Trabajo de Diploma titulado “Módulo Cartográfico”, fue desarrollado en la Universidad de las Ciencias Informáticas. Esta entidad considera que, en correspondencia con los objetivos trazados, el trabajo realizado le satisface:

Totalmente _____

Parcialmente en un _____ %

Los resultados de este Trabajo de Diploma le reportan a esta entidad los beneficios siguientes:

Y para que así conste, se firma la presente a los ____ días del mes de _____ del año

Representante de la entidad Cargo

Firma Cuño

OPINIÓN DEL TUTOR DEL TRABAJO DE DIPLOMA

Título: Módulo Cartográfico

Autores: Edgar Rojas Ricardo

El tutor del presente Trabajo de Diploma considera que durante su ejecución el estudiante mostró las cualidades que a continuación se detallan.

<Aquí el tutor debe expresar cualitativamente su opinión y medir (usando la escala: muy alta, alta, adecuada) entre otras las cualidades siguientes:

- Independencia

- Originalidad

- Creatividad

- Laboriosidad

- Responsabilidad >

< Además, debe evaluar la calidad científico-técnica del trabajo realizado (resultados y documento) y expresar su opinión sobre el valor de los resultados obtenidos (aplicación y beneficios) >

Por todo lo anteriormente expresado considero que el estudiante está apto para ejercer como Ingeniero Informático; y propongo que se le otorgue al Trabajo de Diploma la calificación de <nota>. <Además, si considera que los resultados poseen valor para ser publicados, debe expresarlo también>

Firma

Fecha

“Los conceptos y principios fundamentales de la ciencia son invenciones libres del espíritu humano.”

Albert Einstein

AGRADECIMIENTOS

Para la realización de este trabajo he necesitado la ayuda de muchas personas, a las cuales quisiera agradecerles de todo corazón por su tiempo y dedicación, no solo agradecer a los que me ayudaron, también aquellos que me dieron las espaldas, gracias, porque me inspiraron para demostrarles que si se puede.

Le agradezco mucho a la revolución, a la UCI, y a mi invicto Comandante en Jefe, a todos los profesores y compañeros de aula que he tenido en estos 5 años.

Le agradezco a mi tutor que más que un tutor ha sido un amigo que he tenido cerca en todo momento y que me ha ayudado en todo.

Le agradezco a Yuniesky por todo el tiempo que le dedicó a este trabajo y a Maidileidis.

A mi novia Liudmila que estuvo siempre conmigo en las malas y en las buenas y siempre tuve en ella el apoyo que me hizo falta.

A mis suegros Santiago y Miriam que siempre me apoyaron en todo.

Le agradezco mucho a mi tío Roberto que siempre lo tuve ahí para lo que me hiciera falta y siempre me ayudo en todo sin tener reparo.

A mi hermano, a mi primo Ricardo, a mis abuelos a mi tío Papito, y a toda mi familia que siempre me ayudaron en lo que pudieron para que llegara hasta aquí.

A mis padres que no porque los haya dejado para último es a los que menos le agradezco, todo lo contrario desde el fondo de mi corazón, gracias a ustedes dos, que me dieron la vida, me educaron, aguantaron todas mis malcriadeces y nunca me dieron la espalda en nada y que hicieron todo lo que estaba a su alcance y más para que lo tuviera todo en estos 5 años, Por eso y por mucho más a ustedes dos gracias y felicidades que este es el fruto de su esfuerzo.

DEDICATORIA

A mis padres y a mi hermano

RESUMEN

Este trabajo se propone realizar un estudio para proveer al Sistema de Información Geográfico (GIS) de UCI de una mayor funcionalidad, debido a que en estos momentos no cuenta con una herramienta que permita a sus usuarios utilizarlo como ayuda en la toma de decisiones. Éste es uno de los objetivos claves en sistemas informáticos de este tipo.

El objetivo concreto de esta investigación es implementar un módulo que al incluirlo en el GIS brinde al usuario la posibilidad de realizar análisis estadísticos con la información real de un área determinada, de esta forma el GIS dejará de ser un sistema de consulta geográfica, para convertirse en una poderosa herramienta de consulta geoestadística con la cual se podrán tomar decisiones más rápidas y efectivas.

El presente documento recoge los resultados del estudio realizado para llevar acabo el módulo en cuestión. Se realiza un análisis de las formas con las cuales resultaría más óptima brindar información (Gráficos Estadísticos y Mapas Temáticos). Además se hace un análisis comparativo acerca de las tecnologías existentes para desarrollar este tipo de aplicación y se seleccionan las más apropiadas. Se muestran los resultados del diseño de la propuesta del módulo y finalmente se incluye el estudio de factibilidad del proyecto.

ÍNDICE

INTRODUCCIÓN	1
CAPÍTULO I. FUNDAMENTACIÓN TEÓRICA.....	5
1.1. INTRODUCCIÓN	5
1.2. REPRESENTACIÓN DE INFORMACIÓN MEDIANTE GRÁFICOS ESTADÍSTICOS.	5
1.3. ANÁLISIS DESCRIPTIVOS.	6
1.4. COMPARACIÓN DE DOS O MÁS GRUPOS.....	8
1.5. OTROS TIPOS DE GRÁFICOS.....	9
1.6. PRINCIPALES CARACTERÍSTICAS DE LOS MAPAS TEMÁTICOS.	10
1.6.1. <i>Componentes de un mapa temático.</i>	10
1.6.2. <i>Tipos de cartografías temáticas.</i>	10
1.7. ATLAS. COMBINACIÓN DE MAPAS.	12
1.8. SERVICIOS DE MAPAS EN WEB.	12
1.9. SISTEMAS AUTOMATIZADOS EXISTENTES VINCULADOS AL CAMPO DE ACCIÓN.....	13
1.10. OBJETO DE ESTUDIO.....	14
1.10.1 <i>Situación Problémica</i>	14
1.11. PROPUESTA DE SOLUCIÓN.....	14
1.12. FUNDAMENTACIÓN DE LOS OBJETIVOS PROPUESTOS.....	15
1.12.1. <i>Objetivo General.</i>	15
1.12.2. <i>Objetivos específicos.</i>	15
1.13. CONCLUSIONES.	16
CAPÍTULO 2. TENDENCIAS Y TECNOLOGÍAS ACTUALES UTILIZADAS	17
2.1 INTRODUCCIÓN	17
2.2 INTERNET	17
2.3 INTERNET Y LOS GIS.....	17
2.4 LENGUAJES DE PROGRAMACIÓN PARA LA WEB	18
2.4.1 <i>Lenguajes de lado del servidor:</i>	18
2.4.2 <i>Lenguajes del lado de cliente:</i>	25
2.5. FUNDAMENTACIÓN DE LA SELECCIÓN DE LENGUAJE A UTILIZAR.	26
2.6. SISTEMAS GESTORES DE BASES DE DATOS.	27
2.6.1. <i>SGBD de tipo comercial</i>	27
2.6.2. <i>SGBD libres</i>	28
2.7. CARACTERÍSTICAS DE LOS SGBD DE LA PROPUESTA.	29
2.8. FUNDAMENTACIÓN DE LA METODOLOGÍA A UTILIZAR.	29
2.9 PROPUESTA.....	31

2.10. CONCLUSIONES.	32
CAPITULO 3. DESCRIPCIÓN DE LA PROPUESTA DE SOLUCIÓN	33
3.1. INTRODUCCIÓN.	33
3.2. MODELO DE DOMINIO.	33
3.2.1. <i>Procesos del negocio propuesto.</i>	33
3.2.2. <i>Definiciones de los principales conceptos.</i>	34
3.3. REQUERIMIENTOS FUNCIONALES.	35
3.4. REQUERIMIENTOS NO FUNCIONALES.	38
3.5. DESCRIPCIÓN DEL SISTEMA PROPUESTO.	39
3.6. MODELO DE CASOS DE USO DEL SISTEMA.	39
3.7. EXPANSIÓN DE LOS CASOS DE USO.	41
3.8. CONCLUSIONES.	49
CAPITULO 4 CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA	50
4.1. INTRODUCCIÓN.	50
4.2. MODELO DE ANÁLISIS.	50
4.2.1. <i>Diagramas de Clases de Análisis.</i>	50
4.3. MODELO DE DISEÑO.	52
4.3.1. <i>Diagrama de Clases de Diseño.</i>	52
4.4. PRINCIPIOS DE DISEÑO DE LA INTERFAZ.	64
4.4.1 <i>Estándares de la interfaz de la aplicación.</i>	64
4.4.2 <i>Formato de los reportes.</i>	65
4.4.3 <i>Tratamiento de excepciones.</i>	65
4.5. ESTÁNDARES DE CODIFICACIÓN.	66
4.6. MODELO DE DESPLIEGUE.	67
4.7. MODELO DE IMPLEMENTACIÓN.....	68
4.8. DESCRIPCIÓN PRELIMINAR DEL MODELO DE PRUEBAS.	71
4.9. CONCLUSIONES	71
CAPÍTULO 5. ESTUDIO DE FACTIBILIDAD.	72
5.1. INTRODUCCIÓN	72
5.2. ESTIMACIÓN DEL ESFUERZO BASADA EN CASOS DE USO	72
5.3. PUNTOS DE CASOS DE USO	72
5.4. CÁLCULO DE PUNTOS DE CASOS DE USO SIN AJUSTAR	73
5.4.1. <i>Factor de Peso de los Actores sin ajustar (UAW).</i>	73
5.4.2. <i>Factor de Peso de los Casos de Uso sin ajustar (UUCW)</i>	74

5.5. CÁLCULO DE PUNTOS DE CASOS DE USO AJUSTADOS	74
5.5.1. <i>Factor de complejidad técnica (TCF)</i>	75
5.5.2. <i>Factor de ambiente (EF)</i>	76
5.6. DE LOS PUNTOS DE CASOS DE USO A LA ESTIMACIÓN DEL ESFUERZO	78
5.6.1. <i>Factor de conversión (CF)</i>	79
5.7. BENEFICIOS TANGIBLES E INTANGIBLES	81
5.8. ANÁLISIS DE COSTOS Y BENEFICIOS	81
5.9. CONCLUSIONES	82
CONCLUSIONES GENERALES	83
RECOMENDACIONES	84
GLOSARIO DE TÉRMINOS	85
REFERENCIAS BIBLIOGRÁFICAS	86
ANEXOS	88

ÍNDICE DE TABLAS

Tabla 3. 1. Actores del Sistema	40
Tabla 3. 2. Resumen del Caso de Uso 1.....	40
Tabla 3. 3. Resumen del Caso de Uso 2.....	40
Tabla 3. 4. Caso de Uso Crear Gráfico Estadísticos	41
Tabla 3. 5. Caso de Uso Crear Mapa Temático	46
Tabla 5. 1. Factores de Peso de los Actores sin Justificar	73
Tabla 5. 2. Factores de Peso de los Casos de Uso sin Justificar.....	74
Tabla 5. 3. Factores de Complejidad	75
Tabla 5. 4. Factores de Ambiente	77
Tabla 5. 5. Distribución del Esfuerzo.....	79
Tabla 5. 6. Valor del Esfuerzo	80
Tabla 1.1. Distribución de Frecuencia de la Edad de 100 Pacientes	89

ÍNDICE DE FIGURAS

Figura 3. 1. Modelo de Dominio	34
Figura 3. 2. Diagrama de Casos de Uso del Sistema.....	41
Figura 4. 1. Diagrama de Clases de Análisis: CU Crear Gráfico Estadístico	51
Figura 4. 2. Diagrama de Clase de Análisis: CU Crear Mapa Temático.....	52
Figura 4. 3. Arquitectura.....	53
Figura 4. 4. Paquete Páginas.....	55
Figura 4. 5. Paquete Filtrar.....	56
Figura 4. 6. Paquete Pintar	57
Figura 4. 7. Paquete Acceso a Datos.....	57
Figura 4. 8. Gestionar Gráfico Estadístico.....	58
Figura 4. 9. Gestionar Mapa Temático	59
Figura 4. 10. DCW: Crear Gráfico	60
Figura 4. 11. DCW: Crear Mapa Temático	61
Figura 4. 12. DS: Crear Gráfico Estadísticos	62
Figura 4. 13. DS: Crear Mapa Temático.....	63
Figura 4. 14. Diseño Interfaz	64
Figura 4. 15. Reporte	65
Figura 4. 16. Errores	66
Figura 4. 17. Modelo de Despliegue	68
Figura 4. 18. Diagrama de Componentes de la Capa Presentación	69
Figura 4. 19. Diagrama de Componentes de la Capa Acceso a Datos	69
Figura 4. 20. Diagrama de Componentes de la Capa Negocio	70
Figura 4. 21. Mapa de Navegación del Sitio.....	95
Figura 4. 22. Diagrama de Secuencia CU: Gráfico Estadístico (General X Sexo)	96
Figura 4. 23. Diagrama de Secuencia CU: Mapas Temáticos (Manzana X Provincias) 97	
Figura 1.1. Gráfico de Sectores	88
Figura 1.2. Gráfico de Barra.....	88
Figura 1.3. Histograma Correspondiente a los Datos de la Tabla 1.....	89
Figura 1.4. Polígono de Frecuencia Correspondiente a los Datos de la Tabla 1.	90
Figura 1.5. Diagrama de Caja Correspondiente a los Datos de la Tabla 1.	90
Figura 1.6. Gráfico P-P Correspondiente a los Datos de la Tabla 1.12.....	91
Figura 1.7. Diagrama de Barras Dobles.....	91

Figura 1.8. Barra del Error.....	92
Figura 1.9. Gráfico de Líneas.....	92
Figura 1.10. Gráfico de Líneas Superpuestas.....	93
Figura 1.11. Curvas ROC.....	93
Figura 1.12. Componentes de un Mapa Temático	94
Figura 1.13. Mapa de Coropletas	94

INTRODUCCIÓN

La Universidad de las Ciencias Informáticas (UCI), creada al calor de la batalla de ideas, ha sido concebida para formar jóvenes con una excelente preparación tanto académica como profesional. Para conseguirlo, ha puesto sus manos los recursos más novedosos y necesarios, no sólo para su preparación intelectual, sino para su bienestar de forma general.

Esta institución, que tiene como objetivo rector la informatización de toda su infraestructura y con ella fomentar al desarrollo de la Industria Cubana del Software, no se mantiene al margen del desarrollo de las Tecnologías de la Información y las Comunicaciones (TIC) en el mundo. El marcado avance de las TIC ha conllevado a la creación de nuevas herramientas en los más diversos campos, las cuales permiten mejorar el proceso de toma de decisiones en toda la sociedad. Uno de los más importantes en este sentido son los Sistemas de Información Geográfico (GIS).

El uso de los Sistemas de Información Geográfica se ha expandido de forma increíble por todo el mundo y han pasado del total desconocimiento a la práctica cotidiana. Ellos son los encargados de capturar, almacenar, integrar, manipular, analizar y visualizar todos los datos que están espacialmente referenciados a un área determinada.[1]

La universidad actualmente cuenta con un Sistema de Información Geográfico, que da la posibilidad de navegar por el mapa de la institución pero no cuenta con la opción de tener conocimiento de la información concreta existente en determinado espacio.

Esta investigación surge como necesidad de dar solución a las situaciones antes expuestas; por lo que nuestro **problema** queda formulado en la siguiente interrogante:

¿Cómo implementar un módulo que permita al Sistema de Información Geográfico existente en la UCI, brindar toda la información necesaria en un área determinada a través de gráficos estadísticos y mapas temáticos?

Históricamente, los mapas de propósito general o de referencia, han sido el objetivo de la cartografía hasta mediados del siglo XVIII. El ánimo de geógrafos, exploradores y cartógrafos hasta entonces, había sido el conocimiento geográfico del mundo. Sólo cuando esta necesidad fue satisfecha, los cartógrafos tuvieron la posibilidad de comenzar a expresar en los mapas datos sociales y científicos, naciendo así la cartografía temática. Su objetivo es la representación gráfica de estos datos, transformándolos en símbolos cartográficos y sus relaciones en todo lo que afecte al espacio geográfico.

Con este trabajo se pretende propiciar un incremento de las funcionalidades del sistema actual y que permita un mayor aprovechamiento de la información almacenada, convirtiéndose en una ayuda indispensable y útil para todos.

Por tanto el **objeto de estudio** de la presente investigación es la gestión información presente en la Universidad de las Ciencias Informáticas y su visualización por medio de gráficos estadísticos y mapas temáticos, para conocimiento de trabajadores y estudiantes de la misma.

De aquí se deriva que el **campo de acción** que abarca este trabajo, el Sistema de Información Geográfico existente en la UCI y la automatización de un módulo que permita manejar información de la misma

Como **hipótesis** de este trabajo partimos de la idea de que si se desarrolla un módulo para el Sistema de Información Geográfico con que cuenta la institución, basado en un lenguaje de programación rápido eficiente y multiplataforma como java, es posible lograr la representación de información vigente en la Universidad

El **objetivo general** del trabajo será: implementar un módulo que permita al Sistema de Información Geográfico existente en la UCI, brindar toda la información necesaria en un área determinada a través de gráficos estadísticos y mapas temáticos.

De este objetivo general se derivan los siguientes **objetivos específicos**.

- i. Realizar un estudio de los diferentes tipos de gráficos estadísticos que nos

- permiten representar información.
- ii. Estudiar los diferentes tipos de mapas temáticos con los cuales se puede representar información de determinada área.
 - iii. Crear un módulo que permita realizar gráficos estadísticos y mapas temáticos dentro del Sistema de Información Geográfico de la UCI.

Para cumplir con nuestros objetivos y resolver el problema planteado, se proponen las siguientes **acciones**:

- 1- Realizar un estudio teniendo en cuenta los antecedentes de esta forma de representación de la información y su impacto en el desarrollo de las TIC.
- 2- Analizar como se encuentran en el ámbito internacional las tecnologías que se utilizan para representar información de la forma que se plantea en el presente trabajo.
- 3- Selección de la metodología de Análisis y Diseño de sistemas Informáticos, que facilite la creación y garantice la calidad de el módulo.
- 4- Selección de las herramientas para implementar el módulo y la elección de la plataforma en la que se desarrollará la aplicación. fundamentando la elección.
- 5- Implementación del módulo que permita al Sistema de Información Geográfico (SIG) con que contamos, la representación y búsqueda de información presentes en La Universidad.

El presente trabajo no está antecedido en la Universidad de las Ciencias Informáticas por ninguna aplicación por lo que se pretende obtener un producto a la altura de las exigencias actuales de la Universidad.

El presente documento se estructura en cinco capítulos:

Capítulo 1. En este capítulo se analizarán los principales conceptos relacionados con gráficos estadísticos y mapas temáticos. Se explicará la combinación de mapas con gráficos estadísticos, se hará referencia a los servicios de mapas en Web y a los

sistemas automatizados existentes vinculados al campo de acción, y se plantean los objetivos generales y específicos además de la propuesta de solución.

Capítulo 2. En este capítulo se hace un análisis de cómo se encuentran internacionalmente las tecnologías adecuadas para llevar a cabo el módulo que se pretende desarrollar. Se analizan temas como; Internet y en que ha ayudado en el desarrollo de los (SIG). Se fundamentan las tecnologías con las cuales se desarrolla el módulo, así como la metodología a utilizar para el análisis y diseño del sistema. Finalmente se obtiene como resultado la propuesta final del módulo.

Capítulo 3. En este capítulo se describe la propuesta de esta investigación. Se describen los procesos del negocio que tienen que ver con el objeto de estudio. Y se enumeran los requisitos funcionales y No funcionales que debe tener el módulo para obtener de él óptimos resultados mostrándose en forma de diagramas los casos de usos que ellos se derivan.

Capítulo 4. En este capítulo se realiza el diseño de la propuesta de solución, creando los artefactos que ayudan a la construcción del proyecto, tratar con esto de eliminar todas las complicaciones que puedan aparecer en el camino hasta llegar a la culminación satisfactoria del módulo.

Capítulo 5. Este capítulo se realiza la estimación del esfuerzo del proyecto, basados en Casos de Uso. Se hace un estudio de los beneficios tangibles e intangibles que brinda la aplicación explicándose los costos y beneficios del mismo.

CAPÍTULO I. FUNDAMENTACIÓN TEÓRICA.

1.1. Introducción

En este capítulo se abordarán conceptos de la representación gráfica de los resultados de un estudio, constatando su utilidad en el proceso de análisis estadístico y la presentación de datos. Se describen los distintos tipos de gráficos que se pueden utilizar y su correspondencia con las distintas etapas del proceso de análisis. Se muestran las principales características de los mapas temáticos, se explica la combinación de mapas con gráficos estadísticos y por último se refiere a los servicios de mapas en Web.

1.2. Representación de Información mediante Gráficos Estadísticos.

En estadística se denomina gráficos a aquellas imágenes que, combinando la utilización de sombreado, colores, puntos, líneas, símbolos, números y un sistema de referencia (coordenadas), permiten presentar información cuantitativa. La utilidad de los gráficos es doble, ya que pueden servir no sólo como sustituto a las tablas, sino que también constituyen por sí mismos una poderosa herramienta para el análisis de los datos, siendo en ocasiones el medio más efectivo no sólo para describir y resumir la información, sino también para analizarla.[2]

La presentación de datos estadísticos por medio de gráficos es considerada una tarea importante en el proceso de comunicación de información. Usualmente cuando alguien recibe en sus manos un documento con gráficos, la primer mirada se dirige a estos. La exposición de datos mediante gráficos es algo que se realiza a diario y en forma casi natural por personas de las más disímiles profesiones. En comparación con otras formas de presentación de datos, los gráficos nos permiten, de una mirada, comprender el comportamiento de una variable, aún de variables muy complejas, por lo tanto ahorran tiempo al analista de información. Los gráficos estadísticos permiten usar las habilidades visuales para procesar información.

1.3. Análisis Descriptivos.

Cuando se dispone de datos de una población, y antes de abordar análisis estadísticos más complejos, un primer paso consiste en presentar esa información de forma que ésta se pueda visualizar de manera más sistemática y resumida. Los datos que interesan dependen, en cada caso, del tipo de variables que se esté manejando. Para variables categóricas, como el sexo, TipoPersona, etc., se quiere conocer la frecuencia y el porcentaje del total de casos que "caen" en cada categoría. Una forma muy sencilla de representar gráficamente estos resultados es mediante diagramas de barras o diagramas de sectores. En los gráficos de sectores, también conocidos como diagramas de "tartas", se divide un círculo en tantas porciones como clases tenga la variable, de modo que a cada clase le corresponde un arco de círculo proporcional a su frecuencia absoluta o relativa.[3] Un ejemplo se muestra en la Figura 1.1.

Como se puede observar, la información que se debe mostrar en cada sector hace referencia al número de casos dentro de cada categoría y al porcentaje del total que estos representan. Si el número de categorías es excesivamente grande, la imagen proporcionada por el gráfico de sectores no es lo suficientemente clara y por lo tanto la situación ideal es cuando hay alrededor de diez categorías. En este caso se pueden apreciar con claridad dichos subgrupos.

Los diagramas de barras son similares a los gráficos de sectores. Se representan tantas barras como categorías tiene la variable, de modo que la altura de cada una de ellas sea proporcional a la frecuencia o porcentaje de casos en cada clase Figura. 1.2. Estos mismos gráficos pueden utilizarse también para describir variables numéricas discretas. (Cantidad de estudiantes, profesores, etc.).

Para variables numéricas continuas, tales como la edad, el tipo de gráfico más utilizado es el histograma.

Para construir un gráfico de este tipo, se divide el rango de valores de la variable en

intervalos de igual amplitud, representando sobre cada intervalo un rectángulo que tiene a este segmento como base. El criterio para calcular la altura de cada rectángulo es el de mantener la proporcionalidad entre las frecuencias absolutas o relativas de los datos en cada intervalo y el área de los rectángulos. Como ejemplo, la Tabla 1.1 muestra la distribución de frecuencias de la edad de 100 pacientes, comprendida entre los 18 y 42 años.

Si se divide este rango en intervalos de dos años, el primer tramo está comprendido entre los 18 y 19 años, entre los que se encuentra el 4% del total. Por lo tanto, la primera barra tendrá altura proporcional a 4. Procediendo así sucesivamente, se construye el histograma que se muestra en la Figura. 1.3. Uniendo los puntos medios del extremo superior de las barras del histograma, se obtiene una imagen que se llama polígono de frecuencias.

Dicha figura pretende mostrar, de la forma más simple, en qué rangos se encuentra la mayor parte de los datos. Un ejemplo, utilizando los datos anteriores, se presenta en la Figura. 1.4. Otro modo habitual, y muy útil, de resumir una variable de tipo numérico es utilizando el concepto de percentiles, mediante diagramas de cajas. La Figura. 1.5 muestra un gráfico de cajas correspondiente a los datos de la Tabla 1.1. La caja central indica el rango en el que se concentra el 50% de los datos. Sus extremos son, por lo tanto, el 1er y 3er cuartil de la distribución. La línea central en la caja es la mediana. De este modo, si la variable es simétrica, dicha línea se encontrará en el centro de la caja. Los extremos de los "bigotes" que salen de la caja son los valores que delimitan el 95% de los datos, aunque en ocasiones coinciden con los valores extremos de la distribución. Se suelen también representar aquellas observaciones que caen fuera de este rango (valores extremos). Esto resulta especialmente útil para comprobar, gráficamente, posibles errores en nuestros datos.

Por último, en lo que respecta a la descripción de los datos, suele ser necesario, para posteriores análisis, comprobar la normalidad de alguna de las variables numéricas de las que se dispone. Un diagrama de cajas o un histograma son gráficos sencillos que permiten comprobar, de un modo puramente visual, la simetría y el "apuntamiento" de la distribución de una variable y, por lo tanto, valorar su desviación de la normalidad. Existen otros métodos gráficos específicos para este propósito, como son los gráficos

P-P o Q-Q. En los primeros, se confrontan las proporciones acumuladas de una variable con las de una distribución normal. Si la variable seleccionada coincide con la distribución de prueba, los puntos se concentran en torno a una línea recta. Los gráficos Q-Q se obtienen de modo análogo, esta vez representando los cuantiles de distribución de la variable respecto a los cuantiles de la distribución normal. En la Figura. 1.6 se muestra el gráfico P-P correspondientes a los datos de la Tabla 1.1 que sugiere, al igual que el correspondiente histograma y el diagrama de cajas, que la distribución de la variable se aleja de la normalidad.

1.4. Comparación de dos o más grupos.

Cuando se quieren comparar las observaciones tomadas en dos o más grupos de individuos una vez más el método estadístico a utilizar, así como los gráficos apropiados para visualizar esa relación, dependen del tipo de variables que estemos manejando.

Cuando se trabaja con dos variables cualitativas se puede seguir empleando gráficos de barras o de sectores. Si se quiere determinar, por ejemplo, si en una muestra dada, la frecuencia de sujetos que padecen una enfermedad es más frecuente en aquellos que tienen algún familiar con antecedentes de la misma. A partir de dicha muestra se puede representar, como se hace en la Figura. 1.7, dos grupos de barras: uno para los sujetos con los antecedentes de la enfermedad en familiares y otro para los que no tienen este tipo de antecedentes. En cada grupo, se dibujan dos barras representando el porcentaje de pacientes que tienen o no alguna enfermedad de este tipo. No se debe olvidar que cuando los tamaños de las dos poblaciones son diferentes, es conveniente utilizar las frecuencias relativas, ya que en otro caso el gráfico podría resultar engañoso.

Por otro lado, la comparación de variables continuas en dos o más grupos se realiza habitualmente en términos de su valor medio, a través de análisis de la varianza o métodos no paramétricos equivalentes, y así se ha de reflejar en el tipo de gráfico utilizado. En este caso resulta muy útil un diagrama de barras de error, como en la Figura. 1.8. En él se compara el índice de masa corporal en una muestra de hombres y

mujeres. Para cada grupo, se representa su valor medio, junto con su 95% de intervalo de confianza. Conviene recordar que el hecho de que dichos intervalos no se solapen, no implica necesariamente que la diferencia entre ambos grupos pueda ser estadísticamente significativa, pero sí puede servir para valorar la magnitud de la misma. De esta forma, para visualizar este tipo de asociaciones, pueden utilizarse dos diagramas de cajas, uno para cada grupo. Estos diagramas son especialmente útiles aquí: no sólo permiten ver si existe o no diferencia entre los grupos, sino que además dan la posibilidad de comprobar la normalidad y la variabilidad de cada una de las distribuciones.

Por último, señalar que también en esta situación pueden utilizarse los ya conocidos gráficos de barras, representando en ellos, como altura de cada barra, el valor medio de la variable de interés. Los gráficos de líneas pueden resultar también especialmente interesantes, sobre todo cuando es necesario estudiar tendencias a lo largo del tiempo. Figura. 1.9. Este tipo de gráfico no es más que una serie de puntos conectados entre sí mediante rectas, donde cada punto puede representar distintas cosas según lo que se estudie en cada momento (el valor medio de una variable, porcentaje de casos en una categoría, el valor máximo en cada grupo, etc.).

1.5. Otros tipos de gráficos.

Los tipos de gráficos mostrados hasta aquí son los más sencillos que se pueden manejar, pero ofrecen grandes posibilidades para la representación de datos y pueden ser utilizados en múltiples situaciones, incluso para representar los resultados obtenidos por métodos de análisis más complicados. Se pueden utilizar, por ejemplo, dos diagramas de líneas superpuestos para visualizar los resultados de un análisis de la varianza con dos factores. Figura. 1.10. Un diagrama de dispersión es el método adecuado para valorar el resultado de un modelo de regresión logística. Existen incluso algunos análisis concretos que están basados completamente en la representación gráfica. En particular, la elaboración de curvas ROC. Figura. 1.11 y el cálculo del área bajo la curva constituyen el método más apropiado para valorar la exactitud de una prueba diagnóstica.

1.6. Principales características de los mapas temáticos.

Un mapa temático es aquel que está diseñado para mostrar características o conceptos particulares.[5]

1.6.1. Componentes de un mapa temático.

Todo mapa temático está compuesto por dos elementos fundamentales: Una base geográfica (mapa base), y una capa de contenido temático Figura 1.12. El usuario de un mapa temático tendrá que ser capaz de integrarlas, visual e intelectualmente, durante su lectura.

Mapa base: Es una imagen más o menos sintética del territorio, cuyo objeto es la representación geográfica del contenido temático del mapa; es decir, proporciona información espacial para referenciar el contenido temático.

Capa de contenido temático: Es una capa que contiene la información que se quiere representar sobre el mapa base.

1.6.2. Tipos de cartografías temáticas.

Existen dos tipos de cartografía temática: la cartografía temática cualitativa y la cartografía temática cuantitativa.

Una información cartografiada es cualitativa si es una descripción de características, mientras que si se describen valores la información aportada por el mapa es cuantitativa.

El fin de los mapas cualitativos es mostrar la distribución espacial o la situación de un grupo de datos nominales. De este tipo de mapas el usuario no puede determinar relaciones de cantidad.

Los mapas cuantitativos sin embargo, muestran aspectos espaciales de datos numéricos. A menudo la variable cartografiada es única, y el mapa se centra en la variación de ésta de un punto a otro del espacio geográfico. Estos mapas muestran los datos en una escala ordinal (más que, menos que), y en escalas de intervalo y proporción (cuánto más que). Es por esta razón que se implementará este tipo de mapa en este proyecto, pues los cualitativos no muestran la información de la manera que se quiere en la universidad.

Estos mapas informan el comportamiento de una variable atendiendo a criterios de cantidad. Para representar la información cuantitativa se dispone de diferentes técnicas. La elección de una técnica u otra depende de varios factores entre los que se pueden citar:

- 1.- El carácter del fenómeno que se esté representando.
- 2.- El tipo y la complejidad de la información cuantitativa.
- 3.- El propósito del mapa.

Algunas de las técnicas utilizadas para la confección de mapas con información cuantitativa son las siguientes:

Mapas de punto: en estos mapas la información cuantitativa se representa por puntos, y cada uno representa un valor unitario. Convencionalmente se utiliza el punto (la forma más simple de símbolo), aunque podría haber sido cualquier otro símbolo puntual. Para distinguir las cantidades a representar se colocan cierta cantidad de elementos en correspondencia con la ocurrencia del fenómeno.

Cartogramas: en estos mapas la representación geográfica no es proporcional a su tamaño geográfico (por ejemplo el comedor puede ser más grande que un docente), sino que éstos se representan en función de otra variable como cantidad de mesas, cantidad de computadoras, etc. En la aplicación de esta técnica, se pierden las relaciones de aproximación, de orientación y del área geográfica. El usuario ve una imagen distorsionada, que sólo se acerca muy relativamente a los mapas que está acostumbrado a ver.

Mapas de coropletas: estos son una forma de cartografiado cuantitativo, utilizada para la representación de fenómenos discretos y están asociados a unidades de enumeración, a las que se aplican símbolos superficiales de acuerdo con su valor. Para ello utiliza tramas o colores diferentes aplicados a estas zonas siguiendo el criterio de (a mayor cantidad, más oscuro será el color). Una buena ventaja de este sistema es que se comprende con facilidad por parte del usuario, la distribución de los datos en un territorio.

En estos mapas se representan normalmente datos ya clasificados mediante intervalos. Al realizar esta clasificación, es evidente que se perderá cierta cantidad de información, sin embargo permiten tomar del mapa una información general de forma sencilla y rápida. Esta técnica es utilizada para datos discretos, como en el proyecto se trabaja con este tipo de datos se toman estos mapas como punto de partida. Figura 1.13.

1.7. Atlas. Combinación de mapas.

Cuando se desconoce el significado de una palabra, se acude a un diccionario, cuando se habla de un lugar que no se sabe dónde está, se puede consultar el atlas para averiguarlo.

La geografía se interesa por describir y explicar las distintas regiones de la Tierra y permite comprender las costumbres de otros países, diferentes paisajes, climas, etc. Los mapas dan la posibilidad de conocer lugares lejanos. A lo largo de la historia se han utilizado para tener conocimiento de cualquier parte del mundo e información de todo tipo. Ellos pueden mostrar todas las características de una determinada región sin necesidad de interactuar físicamente con ella.

Existe diferentes tipos, y cada uno tiene determinado objetivo:

Mapas físicos: Tienen colores y símbolos que muestran las ciudades, ríos, montañas, etc.[6]

Mapas políticos: Muestran cómo está compuesto un país: regiones, comunas, etc.

Mapas estadísticos: Los más comunes son los demográficos, económicos, comerciales, etc. Y muestran las características particulares de una determinada región.

La combinación de todos ellos son los que forman un **Atlas**, y poseen, además de gran información geográfica, información estadística, la cual es necesaria para conocer el estado de cualquier región del mundo en cualquier aspecto, económico, social, político entre otros.

1.8. Servicios de mapas en Web.

En su base está la aparición del mundo XML como medio de intercambio de

información, que ha revolucionado el mundo de la informática.

De hecho, se ha pasado del concepto cliente/servidor (compartir datos y recursos en una red) al concepto de servicios WEB (compartir servicios a través de interfaces estándar que publican la información disponible y cómo acceder a ella) con una extensión virtualmente ilimitada.

En el mundo de los gráficos, y en particular en el mundo geográfico, no se había puesto en marcha esta estructura por la complejidad en el tratamiento de los datos de esta naturaleza, por la ausencia de estándares adecuados y software de base de suficiente fiabilidad. De hecho, hasta fechas recientes, los servicios WEB de mapas proporcionaban únicamente información ráster (JPEG, por ejemplo) como resultado a un requerimiento de una aplicación cliente.

Estos servicios WEB satisfacen las consultas entre sistemas que proporcionan información al usuario final. Sin embargo, en las relaciones de Productor a Productor se precisa el mayor aprovechamiento en las capacidades de tratamiento de información que ofrecen los gráficos vectoriales, sumado a la considerable mejora en la representación ante requerimientos de gráficos de alta calidad.

1.9. Sistemas automatizados existentes vinculados al campo de acción.

En la Universidad de las Ciencias Informáticas no existe ningún sistema que se dedique a resolver el problema que se enfrenta en el actual proyecto. En Internet existen varios visores de información geográfica y a continuación se presentan algunos de ellos.

Thuban: es un visualizador con las funciones básicas de control de leyenda, identificación y lectura de atributos de elementos geográficos, impresión, etc. Tiene funciones básicas de proyección. Es multiplataforma y multilenguaje, con versión disponible en español, aunque el manual sólo está en inglés.[7]

OpenMap: es un sencillo visor que permite manipular distintos tipos de información geográfica de forma limitada y a la vez proporciona un conjunto de JavaBeans que para desarrollar aplicaciones geográficas.

Global Mapper: un visualizador avanzado que empieza a incorporar algunas funciones de análisis y visualización 3D, tales como cálculo de cuencas de visibilidad y perfiles de visibilidad, cálculo de volúmenes sobre modelos digitales del terreno, y algunas otras funciones.

En ninguno de ellos se puede llegar a representar datos en la forma que se propone en el presente trabajo.

1.10. Objeto de Estudio

El uso de los Sistemas de Información Geográfica ha aumentado enormemente, ha pasado del total desconocimiento a la práctica cotidiana.

Siempre ha sido un deseo del hombre saber dónde está ubicado y qué lo rodea, para justamente poder tomar decisiones importantes en cuanto a: ¿para dónde es mejor trasladarse? ¿qué tiene bueno o malo el lugar en el que se encuentra?, ¿dónde estaría mejor? Estas interrogantes se responden de manera sencilla haciendo uso de un Sistema de Información Geográfica.

En la universidad se cuenta con una aplicación que ofrece la posibilidad de navegar por sus áreas, pero no se puede encontrar en ella información específica de determinado espacio, por tanto este trabajo se basará en la implementación de un módulo que permita darle esta funcionalidad.

1.10.1 Situación Problemática

Actualmente en la Universidad de las Ciencias Informáticas existe un Sistema de Información Geográfica, que aunque le brinda al usuario la posibilidad de tener conocimiento de determinadas áreas, como manzanas, edificios entre otras, no permite realizar análisis estadísticos con informaciones concretas de las mismas, como: capacidad real, capacidad faltante, etc. Esta razón constituye un obstáculo para la toma de decisiones utilizando el GIS. Y este es uno de los objetivos principales que se persigue cuando se implementa un sistema de este tipo.

1.11. Propuesta de solución.

Después de realizar un análisis del problema que se enfrenta, se concluye que se hace necesario implementar un módulo que al incluirlo dentro del Sistema de

Información Geográfico permite mostrarle al usuario información concreta de determinado espacio. Estos resultados se mostrarán en forma de gráficos estadísticos y mapas temáticos. La utilización de estos aumentará la calidad y percepción de cualquier reporte, dando la posibilidad de hacer un estudio exhaustivo por zonas geográficas de forma sencilla y rápida. Esta aplicación pone en manos de sus usuarios herramientas necesarias para la toma de determinadas decisiones, e incluirá fundamentalmente las siguientes funcionalidades:

- Crear gráficos estadísticos de un área determinada
- Visualizar la información requerida en gráficos de diferentes tipos. Ej: barra, pastel, puntos, líneas, áreas, entre otros.

Además se podrán representar los resultados en forma de mapas temáticos de coropletas, donde los colores del mismo permitirán conocer la distribución de los datos en el mapa de la UCI, de manera inmediata.

1.12. Fundamentación de los objetivos propuestos.

Después de haber hecho un análisis de los mecanismos con que cuenta la Universidad de las Ciencias Informáticas para el manejo de la información, este trabajo se propone un conjunto de objetivos para cumplimentar la propuesta de solución planeada anteriormente.

1.12.1. Objetivo General.

El objetivo general del trabajo será: implementar un módulo que permita al Sistema de Información Geográfico existente en la UCI, brindar toda la información necesaria en un área determinada a través de gráficos estadísticos y mapas temáticos.

1.12.2. Objetivos específicos.

Del objetivo general se derivan los siguientes objetivos específicos.

1. Realizar un estudio de los diferentes tipos de gráficos estadísticos que permiten representar información.
2. Estudiar las diferentes tipos de mapas temáticos con los cuales se puede

representar información de determinada área.

3. Implementar un módulo que permita realizar gráficos estadísticos y mapas temáticos dentro del Sistema de Información Geográfico (GIS) de la UCI.

1.13. Conclusiones.

En este capítulo se analizaron los principales conceptos necesarios para entender la importancia de este trabajo, se fundamentó el objetivo general del mismo y se detallaron cuestiones necesarias para comprender el verdadero significado del la naciente investigación.

CAPÍTULO 2. TENDENCIAS Y TECNOLOGÍAS ACTUALES UTILIZADAS

2.1 Introducción

En el presente capítulo se hace un análisis de cómo se encuentran actualmente las tecnologías para el desarrollo aplicaciones Web y se presenta, de forma especial, las herramientas seleccionadas para llevar a cabo el módulo que se pretende desarrollar.

Además se analizan temas como; Internet y cómo ha ayudado ésta en el desarrollo de los GIS, y finalmente se presenta la metodología a utilizar para el análisis y diseño del sistema, teniendo en cuenta las facilidades que puede aportar al trabajo.

2.2 Internet

Desde los inicios de Internet muchas han sido las definiciones que se han sugerido acerca de ella tales como: la red de redes, el Sistema Mundial de redes de computadoras interconectadas [8] y otras. Internet fue concebida a fines de la década de 1960 por la Agencia de Proyectos de Investigación Avanzada (ARPA) y se le nombró ARPAnet, que fue pensada para cumplir funciones de investigación. Su uso se popularizó a partir de la creación de la *World Wide Web*. Actualmente Internet, es un espacio público utilizado por millones de personas en todo el mundo como herramienta de comunicación e información; y permite tener a su alcance, todos los productos y servicios que requieran fronteras de espacio o tiempo.

2.3 Internet y los GIS

A medida que Internet se convierte día a día en el canal de comunicación más importante del mundo, ofrece mayores posibilidades para transmitir y recibir todo tipo de información. Los Sistemas de Información Geográfico se están complementando con este desarrollo.

En muy pocos años, la WWW ha evolucionado desde un sistema hipermedia hasta una completa plataforma informática. Para los usuarios de información geográfica eso

significa que gran parte del trabajo que se realiza en una computadora local se puede obtener a través de Internet. Esto ha posibilitado la implementación de aplicaciones que han permitido migrar de los documentos preparados y estáticos hasta una plataforma interactiva y dinámica.

Previo a la disponibilidad de la tecnología GIS, la forma en que se tomaban decisiones (ejemplo: cual era el lugar más apto para ubicar un nuevo negocio o construir un aeropuerto), no siempre era la más adecuada. Se confiaba en mapas tradicionales y en tablas estadísticas impresas. La actualización de alguno de estos documentos resultaba una tarea difícil ya fueran mapas, tablas o cartografía, y ni siquiera con las mejores versiones de ellos, se podía tener de forma exacta la información que se necesitaba analizar. El resultado obtenido eran decisiones basadas en información pobre, y solucionando sólo parte del problema o simplemente realizando una mala planificación. Todas las alternativas no podían tenerse en cuenta ya que no podían ser visualizadas en conjunto.

En la actualidad gracias a Internet tenemos acceso a Sistemas de Información Geográficos que permiten realizar consultas de tipo geográfico y cartográfico del mundo, con seguridad de obtener datos precisos y actualizados.

2.4 Lenguajes de Programación para la Web

En la actualidad existen varios lenguajes para la programación de aplicaciones Web. Cada uno de ellos posee características que los hace distintos unos de otro, y a la vez más o menos apropiado en determinadas momentos. Estos lenguajes se clasifican fundamentalmente en dos grandes grupos que reconocen la propia arquitectura Cliente / Servidor de esta plataforma de desarrollo: los lenguajes del lado del Servidor y los lenguajes del lado del Cliente.

2.4.1 Lenguajes de lado del servidor:

Un lenguaje del lado del servidor es aquel que se ejecuta en el servidor Web, y se le envía una respuesta al cliente a través de una página. Las páginas que se ejecutan en el servidor pueden realizar accesos a bases de datos, tratamiento de la información entre otras.

CGI

Es el sistema más antiguo que existe para programar páginas del lado del servidor. Actualmente esta tecnología no es la más utilizada por diversas razones entre las que destaca la dificultad con la que se desarrollan los programas y la pesada carga que supone para el servidor que los ejecuta. [9]

Los CGI se escriben habitualmente en el lenguaje Perl, sin embargo, otros lenguajes como C, C++ o Visual Basic pueden ser también empleados para construirlos. El funcionamiento básico de un programa CGI se describe a continuación.

1. Se realiza una petición http, a la que pueden acompañar datos llegados o bien por un formulario o bien a través de la URL.
2. El servidor ejecuta los programas CGI a los que se accede y trabaja con los recursos necesarios para llevar a cabo las acciones, como por ejemplo bases de datos.
3. El programa CGI va escribiendo en la salida estándar el resultado de la ejecución del CGI, que incluye etiquetas HTML, ya que lo que se escribe es una página web.

Algunas desventajas de la programación en CGI son las siguientes:

- Los resultados se escriben directamente con el CGI, así que el código del programa se mezcla con el del HTML haciendo difícil su comprensión y mantenimiento.
- Cada programa CGI que se pone en marcha lo hace en un espacio de memoria propio. Así, si tres usuarios ponen en marcha un CGI a la vez se multiplicará por tres la cantidad de recursos que ocupe ese CGI. Esto significa una grave ineficiencia.

Perl

Es un lenguaje de programación muy utilizado para construir aplicaciones CGI para la Web. Perl es un acrónimo de *Practical Extracting and Reporting Language*, que viene a indicar que se trata de un lenguaje de programación muy práctico para extraer

información de archivos de texto y generar informes a partir del contenido de los ficheros.

Es un lenguaje libre de uso, eso quiere decir que es gratuito. Antes estaba muy asociado a la plataforma Unix, pero en la actualidad está disponible en otros sistemas operativos como Windows.

Perl es un lenguaje de programación interpretado. Por tanto, el código de los *scripts* en Perl no se compila sino que cada vez que se quiere ejecutar se lee el código y se pone en marcha interpretando lo que hay escrito. Además es extensible a partir de otros lenguajes, ya que desde Perl se pueden hacer llamadas a subprogramas escritos en otros lenguajes. También desde otros lenguajes podremos ejecutar código Perl.

ASP

ASP (del inglés *Active Server Pages*) es la tecnología desarrollada por Microsoft para la creación de páginas dinámicas del servidor. ASP se escribe en la misma página Web, utilizando el lenguaje Visual Basic Script o Jscript (JavaScript de Microsoft).[11]

La mayor desventaja que presenta este lenguaje es que solo se puede implementar en los Servidores Web de su desarrollador: Microsoft. Actualmente se ha presentado ya la segunda versión de ASP: el ASP.NET, que comprende algunas mejoras en cuanto a posibilidades del lenguaje y rapidez con la que funciona. ASP.NET tiene algunas diferencias en cuanto a sintaxis con el ASP, de modo que se ha de tratar de distinta manera uno de otro. Para implementarlo es necesario montar en el Servidor la Plataforma .NET

ASP.NET

ASP.NET facilita el desarrollo de aplicaciones. Esta plataforma permite dotar de funciones adicionales a una aplicación Web y escribir una menor cantidad de código, entre otras características. Es independiente de la herramienta de desarrollo y lleva incorporada una rica biblioteca de clases. Además, brinda la Posibilidad de elección del lenguaje de programación, por defecto lleva integrado C#, VB.NET y J#, aunque se pueden usar otros lenguajes.

PHP

PHP (del inglés *Personal Home Page*). Es un lenguaje de programación del lado del servidor gratuito e independiente de plataforma, rápido, con una gran librería de funciones y mucha documentación. Es también un lenguaje interpretado y embebido en el HTML.[12]

Fue creado originalmente en 1994 por Rasmus Lerdorf, pero como PHP está desarrollado en política de código abierto, a lo largo de su historia ha tenido muchas contribuciones de otros desarrolladores. Actualmente PHP se encuentra en su versión 5, que utiliza el motor Zend, desarrollado con mayor meditación para cubrir las necesidades de las aplicaciones Web actuales.

XML

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 importantísimo 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 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 corren a cargo del lenguaje y está especificado por el estándar, de modo que el programador no tiene que preocuparse por ello.

Java

Java es un lenguaje de programación creado por un grupo de ingenieros de Sun a principios de los años 90. Casi desde el principio fue diseñado teniendo en mente las características que se esperaban de los lenguajes de última generación.

Orientado a objeto:

Java es un lenguaje totalmente orientado a objeto. Desde las aplicaciones individuales

hasta los efectos gráficos, pasando por la entrada/salida, la gestión de errores, etc. todos los elementos que forman el lenguaje son objetos.

Java está totalmente orientado a objeto debido a que el programador puede usar los objetos existentes, derivar otros objetos a partir de los anteriores, o crear objetos totalmente nuevos. Esta flexibilidad hace posible un sistema de programación menos propenso a errores y más fácil de depurar y extender.

Capaz de trabajar en la red:

Java no necesita bibliotecas externas para trabajar en red, sino que tiene integrado en el propio núcleo del lenguaje todos los elementos necesarios para abrir, usar y cerrar conexiones TCP o UDP. Los accesos a archivos remotos son transparentes al usuario y al programador. El intérprete Java puede leer archivos de un servidor HTTP remoto de la misma forma que lee un archivo de un directorio local.

Herramienta gráfica:

El programador de Java tiene a su disposición un conjunto de herramientas estándar que le permiten desarrollar aplicaciones complejas basadas en interfaces gráficas de usuario (GUI). Este conjunto de herramientas se denomina AWT (del inglés *Abstract Window Toolkit*) y facilita el uso de elementos tan típicos como ventanas, botones, cuadros de diálogo, etc.

Además, AWT tiene la suficiente flexibilidad como para que un programador pueda personalizar todos sus componentes, derivando nuevos objetos de los ya existentes.

Multitenhebrado:

Java es un entorno de ejecución multitenhebrado, lo que significa que un mismo programa Java puede subdividirse en varias unidades de proceso concurrentes, es decir, que se ejecutan al mismo tiempo, en el mismo espacio de direcciones y compartiendo las mismas variables.

Esto permite crear de forma sencilla programas que realicen más de una cosa a la vez, aunque exige un gran cambio de mentalidad a la hora de programar.

Interpretado:

Java es un lenguaje interpretado, el intérprete de Java, debe ir leyendo el código del programa instrucción a instrucción para descubrir cuál es la próxima acción a realizar.

¿Por qué Java no se compila en código máquina, como el resto de programas? Pues precisamente para permitir que pueda ejecutarse en cualquier ordenador de Internet,

tenga éste un microprocesador 80x86, 680x0, PowerPC, etc.

Además, el hecho de que sea un lenguaje interpretado se convierte en una medida de seguridad para el ordenador en que se ejecuta el applet, ya que el intérprete, a modo de sistema operativo, puede filtrar todas las operaciones de entrada/salida, como acceso a archivos o al hardware del sistema.

Todas estas medidas de seguridad impiden que se puedan transmitir virus escritos en Java a través de Internet, o que un desconocido pueda leer información privada de nuestros archivos para transmitirla a un servidor central.

Multiplataforma:

Java es compatible con todos los sistemas porque basa su funcionamiento en los Byte Codes, que no es más que una precompilación del código fuente de Java.

Estos Byte Codes no son el programa en Java propiamente dicho, sino un archivo que contiene un código intermedio que puede manejar la Máquina Virtual de Java. Cada sistema operativo dispone de una Máquina Virtual de Java que puede interpretar los Byte Codes y transformarlos a sentencias ejecutables en el sistema en cuestión.

JSP:

Es la tecnología de Java para la creación de páginas Web con programación en el servidor. JSP es un acrónimo de Java Server Pages, que en castellano vendría a decir algo como Páginas de Servidor Java. Es, pues, una tecnología orientada a crear páginas Web con programación en Java. [13]

Con JSP podemos crear aplicaciones Web que se ejecuten en variados servidores Web, de múltiples plataformas, ya que Java es en esencia un lenguaje multiplataforma. Las páginas JSP están compuestas de código HTML/XML mezclado con etiquetas especiales para programar scripts de servidor en sintaxis Java. Por tanto, las JSP se pueden escribir con el editor HTML/XML habitual.

En JSP se crean páginas de manera parecida a como se crean en ASP o PHP, dos tecnologías de servidor. Se generan archivos con extensión .jsp que incluyen, dentro de la estructura de etiquetas HTML, las sentencias Java a ejecutar en el servidor.

JSP sigue la filosofía de la arquitectura JAVA de "escribe una vez, ejecuta donde quieras". JSP se puede ejecutar en los sistemas operativos y servidores Web más

populares, como por ejemplo Apache.

Servlets:

Los *Servlets* se ejecutan en el servidor y no presentan ningún tipo de interfaz gráfica puesto que se encargan de hacer el trabajo oculto, un aspecto interesante por lo que muchos programadores que hasta ahora utilizaban CGI, están utilizando Servlets. De hecho, los CGI's eran el único medio de proporcionar interacción entre el cliente y el servidor. Un ejemplo muy común de uso de los CGI son los típicos formularios que el usuario llena con sus datos que posteriormente pasan a formar parte de una base de datos. [14]

Los servlets son programas que funcionan como los CGIs convencionales atendiendo peticiones de un cliente teniendo al servidor como el encargado, pero escritos en Java y con la ventaja de explotar todas las bondades de java. Por ejemplo, un servlet puede ser responsable de tomar los datos de un formulario HTML y enviarlos a una base de datos para actualización de la misma.

La API Servlet, usada para escribir servlets, no incluye nada acerca de cómo son cargados los servlets, ni el ambiente en el cual corren los servlets, ni el protocolo usado para transmitir los datos del usuario. Esto permite a los servlets poder ser usados por diferentes servidores Web.

Los *Servlets* son un sustituto eficaz de los CGI, pues proveen la forma de generar documentos dinámicos que son fáciles de escribir y ejecutar. También evitan el problema de desarrollar la programación según la plataforma utilizada. Los servlets son desarrollados con su propia API, una extensión estándar de Java.

El principal componente de la Servlet API es la interfaz Servlet. Todos los ellos implementan esta interfaz directamente, por medio de la extensión de la clase que la implementa, **HttpServlet**. Esta interfaz está provista de métodos que manipulan a los servlets y la comunicación con sus clientes.

2.4.2 Lenguajes del lado de cliente:

Los lenguajes de lado del cliente son los encargados de aportar dinamismo a la aplicación en los navegadores, pues se utilizan fundamentalmente para comprobar la autenticidad de los datos antes de ser procesados por el servidor.

Visual BasicScript:

Es un lenguaje de programación de scripts del lado del cliente, pero sólo compatible con Internet Explorer. Está basado en Visual Basic. El modo de funcionamiento de Visual Basic Script para construir efectos especiales en páginas Web es muy similar al utilizado en Javascript y los recursos a los que se puede acceder también son los mismos: el navegador.

JavaScript:

Se trata de un lenguaje de programación del lado del cliente, porque es el navegador el que soporta la carga de procesamiento. Gracias a su compatibilidad con la mayoría de los navegadores modernos, es el lenguaje de programación del lado del cliente más utilizado. Con Javascript se pueden crear efectos especiales en las páginas y definir interactividades con el usuario. El navegador del cliente es el encargado de interpretar las instrucciones Javascript y ejecutarlas para realizar estos efectos e interactividades, de modo que el mayor recurso, y tal vez el único, con que cuenta este lenguaje es el propio navegador.

Javascript es un lenguaje con muchas posibilidades, permite la programación de pequeños scripts, pero también de programas más grandes, orientados a objetos con funciones y estructuras de datos complejas. 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.

Con Javascript el programador, que se convierte en el verdadero dueño y controlador de cada cosa que ocurre en la página cuando la está visualizando el cliente.

Applets:

Es otra manera de incluir código a ejecutar en los clientes que visualizan una página Web. Se trata de pequeños programas hechos en Java, que se transfieren con las páginas Web y que el navegador ejecuta en el espacio de la página.

Los applets de Java están programados en Java y precompilados, es por ello que la manera de trabajar de éstos varía un poco con respecto a los lenguajes de script como Javascript. Los applets son más difíciles de programar que los scripts en Javascript y requerirán unos conocimientos básicos o medios del lenguaje Java.

La principal ventaja de utilizar applets consiste en que son mucho menos dependientes del navegador que los scripts en Javascript, incluso independientes del sistema operativo del ordenador donde se ejecutan. Además, Java es más potente que Javascript, por lo que el número de aplicaciones de los applets podrá ser mayor.

Como desventajas en relación con Javascript cabe señalar que los applets son más lentos de procesar y que tienen espacio muy delimitado en la página donde se ejecutan, es decir, no se mezclan con todos los componentes de la página ni tienen acceso a ellos. Es por ello que con los applets de Java no pueden hacer acciones directas, como abrir ventanas secundarias, controlar Frames, formularios, capas, etc.

2.5. Fundamentación de la selección de lenguaje a utilizar.

Luego de hacer un análisis de los diferentes lenguajes y teniendo en cuenta cada una de sus características, se presentarán las ventajas que ofrece el lenguaje seleccionado para esta investigación mediante el método comparativo, mostrando los aspectos que denotan mayor influencia en el desarrollo del trabajo. Todos los lenguajes analizados están soportados por múltiples plataformas, excepto el ASP que solo lo soporta la plataforma Windows. En el caso de la velocidad de ejecución podemos decir que el JSP y PHP son los más rápidos seguidos por PERL. Además de que son los más utilizados en Internet y podemos encontrar una gran cantidad de información, artículos, tutoriales y códigos de ejemplos. JAVA tiene una de las comunidades más grandes en Internet, al

igual que la de PHP. En la universidad JAVA es utilizado por una gran cantidad de desarrolladores que cada vez se hace mayor. Por todas las características que se han tenido en cuenta se concluye que la tecnología más apropiada a utilizar en este trabajo será JSP.

Por las ventajas que ofrecen los lenguajes del lado del cliente se propone el uso de JavaScript, que es un lenguaje orientado a eventos e interpretado, es decir, no requiere compilación.

2.6. Sistemas Gestores de Bases de Datos.

Actualmente son muchas las aplicaciones que requieren acceder a datos sin importar la magnitud de estos. Estos datos se deben almacenar en algún soporte permanente, y las aplicaciones deben disponer de un medio para acceder a ellos. Es aquí donde aparecen los Sistemas Gestores de Bases de Datos (SGBD) los cuales proporcionan una interfaz entre las aplicaciones y el sistema operativo. Esto proporciona, entre otras cosas, que el acceso a los datos se realice de una forma más eficiente, fácil de implementar y, sobre todo, mucho más segura.

En la actualidad existe una gran variedad de SGBD, tanto de tipo comercial como libre. En los epígrafes siguientes se presentará los más conocidos en cada grupo.

2.6.1. SGBD de tipo comercial

Oracle

Oracle es considerado el SGBD más completo que existe. Sus características más destacadas son el soporte de transacciones, su gran estabilidad, seguridad, escalabilidad, lo que lo pone a una altura de multiplataforma. Su gran desventaja es su precio, que es de varios miles de euros (según versiones y licencias).[15]

SQL Server

Se dice que SQL Server es uno de los SGBD más completos. Tanto por la capacidad para consultar la base de datos mediante un explorador como por la compatibilidad con el Lenguaje de marcado extensible (XML). SQL Server está totalmente habilitado para Web. Además, y ostenta marcas de referencia en cuanto a escalabilidad y confiabilidad, que

son críticas para el éxito de una base de datos empresarial. Este gestor al igual que Oracle no es libre. [15]

2.6.2. SGBD libres

MySQL

MySQL implementa funcionalidades Web que permiten un acceso a los datos, seguro y fácil, desde Internet. Es uno de los SGBD más populares, desarrollado bajo la filosofía de código abierto. La licencia GPL de MySQL obliga a distribuir cualquier producto derivado (aplicación) bajo esa misma licencia. Por tanto MySQL tiene sus restricciones, como por ejemplo: sólo es gratis si se está dispuesto a distribuir la aplicación que se quiere desarrollar bajo esa misma licencia GPL. Si se desea distribuir la aplicación comercialmente, entonces se debe pagar la licencia comercial de MySQL que permite hacer exactamente eso. MySQL tiene como una de sus principales ventajas la velocidad en la lectura de datos, pero a costa de eliminar un conjunto de facilidades que presentan otros SGBD: integridad referencial, bloqueo de registros, procedimientos almacenados, entre otros. En recientes versiones de MySQL (la versión 4 y la 5) se incluyen algunas de estas características, pero indudablemente esto va en detrimento de la velocidad. [15]

PostgreSQL

El PostgreSQL que está considerado como el SGBD de código abierto más avanzado del mundo, proporciona un gran número de características que normalmente sólo se encontraban en las bases de datos comerciales de alto calibre tales como Oracle. Es un SGBD objeto-relacional, aproxima los datos a un modelo objeto-relacional, y es capaz de manejar complejas rutinas y reglas. Su avanzada funcionalidad se pone de manifiesto con las consultas SQL declarativas, el control de concurrencia multiversión, soporte multiusuario, transacciones, optimización de consultas, herencia y valores no atómicos (atributos basados en vectores y conjuntos). Y es totalmente libre.[15]

2.7. Características de los SGBD de la propuesta.

El módulo que se presenta tiene como principal objetivo el manejo de toda la información posible de un área georeferenciada. Por este motivo se trabaja con más de una base de dato, pues la información solicitada puede ser de diferentes naturaleza, por ejemplo (estudiantes por provincias) que está disponible en la base de datos “Alojamiento”, (estudiantes suspensos en Programación) se encuentra en “Akademos”, entre otros.

Para lograr un mayor aprovechamiento de la información almacenada en las bases de datos existente en la UCI, es preciso que estas contengan una clase controladora encargada de gestionar toda la información que se requiera de las mismas y que se pueda tener acceso a esta clase desde cualquier aplicación.

Esta característica hoy se convierte en un elemento muy significativo debido al creciente auge del software libre y la migración hacia él por parte de las aplicaciones con las que cuenta la institución.

Las bases de datos Akademos y Alojamiento, a las que se hace referencia anteriormente, cuentan con esta clase controladora pero lo hacen a través de Web Service con autenticación integrada a Windows, lo que indica que sólo tienen acceso a ellos aquellos que trabajen en este Sistema Operativo.

Esta peculiaridad constituye una limitante para esta propuesta, porque de poderse utilizar su clase controladora nos haría el trabajo más fácil y mucho más dinámico. En estos momentos para obtener la información de estas bases de datos se tiene que conectar directamente a ellas lo que resultaría poco factible a la hora de una actualización de las mismas.

Este módulo estará diseñado para tener acceso a SGBD como: MySQL, PostgreSQL, SQL Server ya que son estos los mas utilizados en la universidad.

2.8. Fundamentación de la metodología a utilizar.

La calidad en el desarrollo y mantenimiento del software se ha convertido hoy en día en uno de los principales objetivos de la universidad, los principales servicios dependen de sistemas informatizados, y para su buen funcionamiento se debe tener en cuenta los principios que se deben seguir para lograr una unificación de todos las aplicaciones.

Todo desarrollo de software es riesgoso y difícil de controlar, pero si no se sigue una metodología para controlar los flujos de trabajos, probablemente se obtendrán clientes insatisfechos con el resultado. Por tanto uno de los principios de la institución es, a la hora de desarrollar un software seguir una metodología que nos garantice la satisfacción de los clientes.

Una metodología para el desarrollo de un proceso de software es un conjunto de filosofías, fases, procedimientos, reglas, técnicas, herramientas, documentación y aspectos de formación para los desarrolladores de Sistemas Informáticos.[16]

Para el control y planificación de este trabajo, se decidió utilizar como metodología RUP (del inglés *Rational Unified Process*), por sus características y las facilidades que aporta. Además esta viene acompañado de una herramienta muy buena que soporta cada uno de los procesos que necesitamos: Rational Rose Enterprise Edition 2003.

RUP

El Proceso Unificado Racional o RUP, 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.

Las principales características del proceso unificado de modelado son:

Centrado en los Modelos: Los diagramas son un vehículo de comunicación más expresivo que las descripciones en lenguaje natural. Se trata de minimizar el uso de descripciones y especificaciones textuales del sistema.

Guiado por lo casos de uso: Los casos de uso son el instrumento para validar la arquitectura del software y extraer los casos de prueba.

Centrado en la arquitectura: Los modelos son proyecciones del análisis y el diseño constituye la arquitectura del producto a desarrollar.

Iterativo e incremental: Durante todo el proceso de desarrollo se producen versiones incrementales (que se acercan al producto terminado) del producto en desarrollo.

Utilización de un único lenguaje de modelación: Utiliza *Unified Model Language (UML)* para describir todo el proceso.

El RUP divide el proceso de desarrollo en ciclos, teniendo un producto final al final de cada ciclo, estos se dividen a la vez en fases:

- **inicio:** se hace un plan de fases, se identifican los principales casos de uso y se identifican los riesgos.
- **elaboración:** se hace un plan de proyecto, se completan los casos de uso y se eliminan los riesgos.
- **construcción:** se concentra en la elaboración de un producto totalmente operativo y eficiente y el manual de usuario.
- **transición:** se implementa el producto en el cliente y se entrenan a los usuarios. Como consecuencia de esto suelen surgir nuevos requerimientos a ser analizados.

UML

El Lenguaje de Modelado Unificado UML es un lenguaje estándar para escribir planos de software. UML puede utilizarse para visualizar, especificar, construir y documentar los artefactos de un sistema que involucra gran cantidad de software.

UML proporciona una forma estándar de escribir los planos de un sistema, cubriendo tanto las cosas conceptuales, tales como procesos del negocio y funciones del sistema, como las cosas concretas: las clases escritas en un lenguaje de programación específico, esquemas de bases de datos y componentes de software reutilizables.

2.9 Propuesta.

Después de realizar el anterior análisis se pueden plantear las herramientas que resultarían de mayor eficiencia para la implementación de la propuesta.

Este módulo será diseñado bajo una plataforma Web, que le permita al usuario de GIS obtener información de un área determinada de la institución, en forma de gráfico estadístico o en un mapa temático. Se utilizará como tecnología, JSP pues esta permite utilizar como lenguaje del lado del servidor java. Este lenguaje es uno de los más potentes en la actualidad y brinda muchas facilidades a la hora de la conexión con varias

bases de datos, lo cual constituye un elemento fundamental en esta aplicación. Se utilizará el JavaScript para lograr una mejor interactividad con el usuario en el navegador. Y para garantizar un buen desarrollo y culminación del trabajo se seguirá como metodología de desarrollo RUP.

2.10. Conclusiones.

En este capítulo se analizaron las principales tecnologías candidatas para la elaboración de un sistema de este tipo. Se fundamentó cual será la tecnología que se utilizará para la realización de la propuesta, así como el lenguaje en la que será implementada la aplicación. Además se escogió la metodología de desarrollo de software por la cual se regirá esta investigación, y finalmente se planteó la propuesta que incluye dichos aspectos. A partir de este punto se comenzará el desarrollo de la propuesta de sistema como tal.

CAPITULO 3. DESCRIPCIÓN DE LA PROPUESTA DE SOLUCIÓN

3.1. Introducción.

En el presente capítulo se describe la propuesta de esta investigación. Se describen los procesos del negocio que tienen que ver con el objeto de estudio.

Se exponen los conceptos que puedan ser agrupados en un Modelo de Dominio, debido a la poca estructuración de estos. De esta forma se llegará correctamente a los requisitos y así el módulo propuesto cumplirá con los objetivos para los cuales ha sido concebido.

Además se enumeran los requisitos funcionales y No funcionales que debe tener el módulo para obtener de él óptimos resultados y se mostraran en forma de diagramas los casos de usos que ellos derivan.

3.2. Modelo de Dominio.

En este epígrafe la atención se centrará en el modelo de dominio, que permitirá mostrar al usuario los principales conceptos que se manejan en el dominio del sistema en desarrollo, así como los procesos del negocio propuesto. Esto servirá de gran ayuda para que los usuarios entiendan los objetivos de la investigación y se familiaricen con su vocabulario.

3.2.1. Procesos del negocio propuesto.

Uno de los objetivos rectores de la universidad es informatizar todos los procesos que se lleven a cabo en ella. Por esta razón existe gran cantidad de información almacenada en diferentes bases de datos. En la actualidad estos datos son procesados por diferentes aplicaciones que se encuentran en funcionamiento y que brindan a sus usuarios reportes de los mismos. Con el módulo que se propone como culminación de esta investigación. Se quiere que el Sistema de Información Geográfico de la UCI sea capaz de brindar, de un área determinada toda la información que se pueda obtener de las bases de datos ya existentes. De este modo el usuario tendrá a su disposición una herramienta, que le permitirá obtener reportes de diferentes tipos de información, para la mejor comprensión

de estos reportes se les darán en forma de gráficos estadísticos. Además los usuarios podrán conocer la distribución en la universidad de la información solicitada a través mapas temáticos.

3.2.2. Definiciones de los principales conceptos.

Para un mejor entendimiento de este trabajo se darán a continuación los principales conceptos que se manejarán en el modelo de dominio y a lo largo de la implementación del proyecto: Figura 3.1

- Se le denominará **usuario** a cualquier persona que estudie o trabaje en la Universidad.
- Se denominará **gráfico estadístico** al gráfico que sea capaz de mostrar el comportamiento de una variable matemáticamente. Ejemplo pastel, barra y otros
- Llamaremos **mapa temático** a los mapas que su fin es mostrar la distribución de una variable en un espacio geográfico. Ejemplo mapa de punto, coropletas entre otros.
- **Datos** se le denominará a todas las bases de datos de las cuales podremos obtener información, Avademos, Alojamiento y otras.

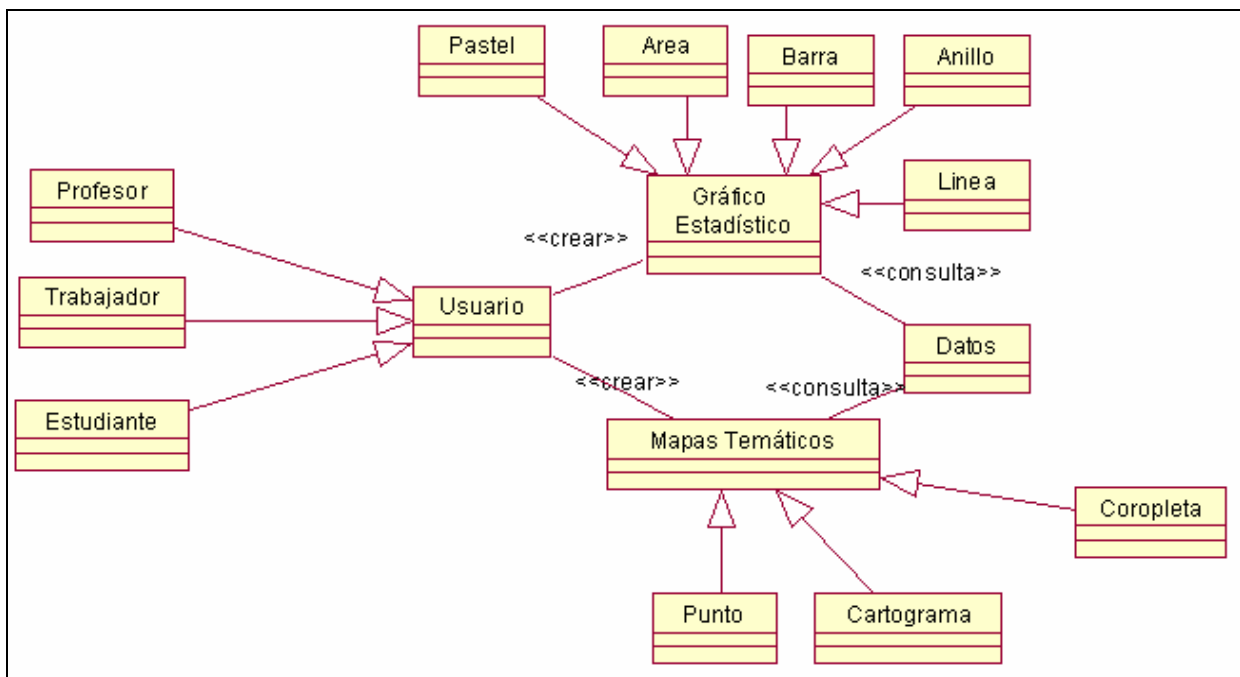


Figura 3. 1. Modelo de Dominio

El modelo del dominio se describe mediante diagramas UML, específicamente con un diagrama clases conceptuales significativas en el dominio del problema.

3.3. Requerimientos Funcionales.

Después de conocer los conceptos que encierran al objeto de estudio, se puede empezar a considerar los elementos con los que debe contar el sistema para que se cumplan los objetivos planteados al inicio de este trabajo. Para ello enumeraremos a través de requerimientos funcionales las funciones que el sistema deberá ser capaz de realizar. En ellos estarán presentes las acciones que podrán ser ejecutadas por el usuario. Para cumplir los objetivos de esta aplicación la misma deberá ser capaz de:

R1. Crear gráficos estadísticos.

1.1. Mostrar opciones para escoger el área de la cual se desea realizar un estudio cartográfico mediante gráficos estadísticos. Las áreas pudieran ser: General, Residencia, Manzanas y Edificios. Seleccione el área deseada.

1.1.1. General.

Para graficar la información correspondiente se necesita tener:

a) el tipo de información a filtrar que pudiera ser:

- Sexo
- Matrícula por provincia
- Capacidad total por Residencia.
- Capacidad total por Facultad
- Cantidad estudiante evaluados de B por residencia
- Cantidad estudiante evaluados de R por residencia
- Cantidad estudiante evaluados de M por residencia

b) el tipo de gráfico que se desea mostrar que pudiera ser(estos coinciden en todos los demás casos)

- Área
- Barra
- Barra3D
- Línea

- Linea3D
- StackedArea
- StackedBarra
- StackedBarra3D
- Mariposa
- Pastel
- Pastel3D
- Anillo

1.1.2. Residencia

Para graficar la información correspondiente a un área de la residencia se necesita tener:

- a) el área específica, que será en este caso el número de la residencia.
- b) el tipo de información a filtrar que pudiera ser:
 - Sexo
 - Matrícula por provincia
 - Capacidad total por Manzanas.
 - Capacidad total por Facultad
 - Cantidad estudiante evaluados de B por manzanas
 - Cantidad estudiante evaluados de R por manzanas
 - Cantidad estudiante evaluados de M por manzanas
- c) el tipo de gráfico que se desea mostrar.

1.1.3. Manzanas

Para graficar la información correspondiente a un área de la residencia se necesita tener:

- a) el área específica, que será en este caso el número de la manzana.
- b) el tipo de información a filtrar que pudiera ser:
 - Sexo
 - Matrícula por provincia
 - Capacidad total por Edificio.
 - Capacidad total por Facultad

- Cantidad estudiante evaluados de B por edificio
 - Cantidad estudiante evaluados de R por edificio
 - Cantidad estudiante evaluados de M por edificio
- c) el tipo de gráfico que se desea mostrar.

1.1.4. Edificios

Para graficar la información correspondiente a un área de la residencia se necesita tener:

- a) el área específica, que será en este caso el número del Edificio.
- b) el tipo de información a filtrar que pudiera ser:
- Sexo
 - Matrícula por provincia
 - Capacidad total por Aptos.
 - Capacidad total por Facultad
 - Cantidad estudiante evaluados de B por aptos
 - Cantidad estudiante evaluados de R por aptos
 - Cantidad estudiante evaluados de M por aptos
- c) el tipo de gráfico que se desea mostrar.

R2. Crear mapas temáticos

2.1. Mostrar opciones para escoger el área de la cual se desea realizar un estudio cartográfico mediante mapas temáticos. En este caso sólo se podrá seleccionar por Manzanas. Seleccione el área deseada.

2.1.1. Manzana.

Para mostrar en mapa de la universidad con la información correspondiente se necesita tener:

- a) El tipo de información a filtrar que pudiera ser:
- b. Sexo
 - c. Matrícula por provincia
 - d. Capacidad total
 - e. Capacidad total por Facultad
 - f. Cantidad estudiante evaluados de B

- g. Cantidad estudiante evaluados de R
- h. Cantidad estudiante evaluados de M
- b)** Rango del filtro. Por ejemplo en caso del sexo tenemos que especificar el sexo del cual queremos el mapa.
- c)** Color de la escala. El color con que deseamos visualizar la información en el mapa.
- d)** Niveles de escala. Para saber en cuanto quiero degradar el color del mapa.

3.4. Requerimientos No Funcionales.

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

Apariencia o interfaz externa:

- Diseño sencillo, con pocas entradas, permitiendo que no sea necesario mucho entrenamiento para utilizarlo.
- Empleo de los colores: azul, gris y blanco principalmente, que son los empleados en el (SIG) de la universidad.
- Diseño perfectamente encuadrado para resoluciones de 800x600, pero preparado para verse en otras resoluciones.

Portabilidad:

- Necesidad de que el sistema sea multiplataformas.

Funcionalidad:

- Capacidad de búsqueda y representación de información con velocidad menor que 10 segundos.
- Mínima cantidad de páginas para ejecutar todas las funciones posibles (preferentemente que estén relacionadas).
- Navegación fácil con el teclado.

3.5. Descripción del Sistema Propuesto.

Para darle cumplimiento a los objetivos propuestos al inicio de este trabajo, y asumiendo todos los requerimientos planteados, el módulo que se propone debe brindarle al usuario, (que será aquel que tenga acceso al GIS de la universidad), la posibilidad de tener reportes de información concreta de la residencia, ya sea en forma de gráfico estadístico o de mapa temático, para ello se utilizan las bases de datos existente actualmente: Akademos y Alojamiento de las cuales se tomaran los datos que solicite el usuario y se le dará a conocer la información de manera que pueda manejarla de la forma más adecuada para tomar decisiones, mediante gráficos. Asimismo el usuario podrá darse cuenta con sólo una mirada la distribución de la información solicitada en la universidad mediante un mapa temático en este caso particular de coropletas (por colores). Con el presente módulo el Sistema de Información Geográfico (GIS) dejará de ser un simple visor de conocimiento geográfico, para convertirse en una poderosa herramienta que además de contar con la posibilidad de georeferenciar áreas de la Universidad de Ciencias Informáticas se puede conjuntamente, conocer información específica de la misma de una forma asequible para todos.

3.6. Modelo de Casos de Uso del Sistema.

Utilizando las facilidades que brinda el UML, se capturarán los requisitos funcionales del sistema y se representarán mediante un diagrama de casos de uso. Primero se debe tener claro cuales son los actores que van a interactuar con el sistema, y los casos de uso que representarán todas las funcionalidades del sistema.

Un caso de uso es un documento narrativo que describe la secuencia de un actor (agente externo) que utiliza un sistema para completar un proceso. Un actor no es parte del sistema, es un rol de un usuario, que puede intercambiar información o puede ser un recipiente pasivo de información y representa a un ser humano, a un software o a una máquina que interactúa con el sistema. En este caso con el sistema interactúan cuatro actores. Tabla 3.1

Tabla 3. 1. Actores del Sistema

Actores del sistemas	Justificación
Usuario	Persona que interactúa directamente con la aplicación para crear gráficos estadísticos y mapas temáticos; puede ser un estudiante, un profesor o un trabajador de la Universidad.
BD_Alojamiento	Base de Datos que interactúa directamente con la aplicación proporcionando datos que son necesarios en el desarrollo del sistema.
BD_Mapa	Base de Datos que interactúa directamente con la aplicación proporcionando datos que son necesarios en el desarrollo del sistema.
BD_Akademoss	Base de Datos que interactúa directamente con la aplicación proporcionando datos que son necesarios en el desarrollo del sistema.

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

Tabla 3. 2. Resumen del Caso de Uso 1

CU-1	Crear Gráficos Estadísticos.
Actor	Usuario del Sistema, BD_Alojamiento, BD_Akademoss
Descripción	Un usuario solicita la creación de gráficos estadísticos y se utiliza el Módulo Cartográfico para darle respuesta a dicha solicitud.
Referencia	R1

Tabla 3. 3. Resumen del Caso de Uso 2

CU-2	Crear Mapas Temáticos.
Actor	Usuario del Sistema, BD_Alojamiento, BD_Mapa, BD_Akademoss
Descripción	Un usuario solicita la creación de mapas temáticos y se utiliza el Módulo Cartográfico para darle respuesta a dicha solicitud.
Referencia	R2

El diagrama donde se representa la relación existente entre los actores y los casos de uso se representa a continuación:

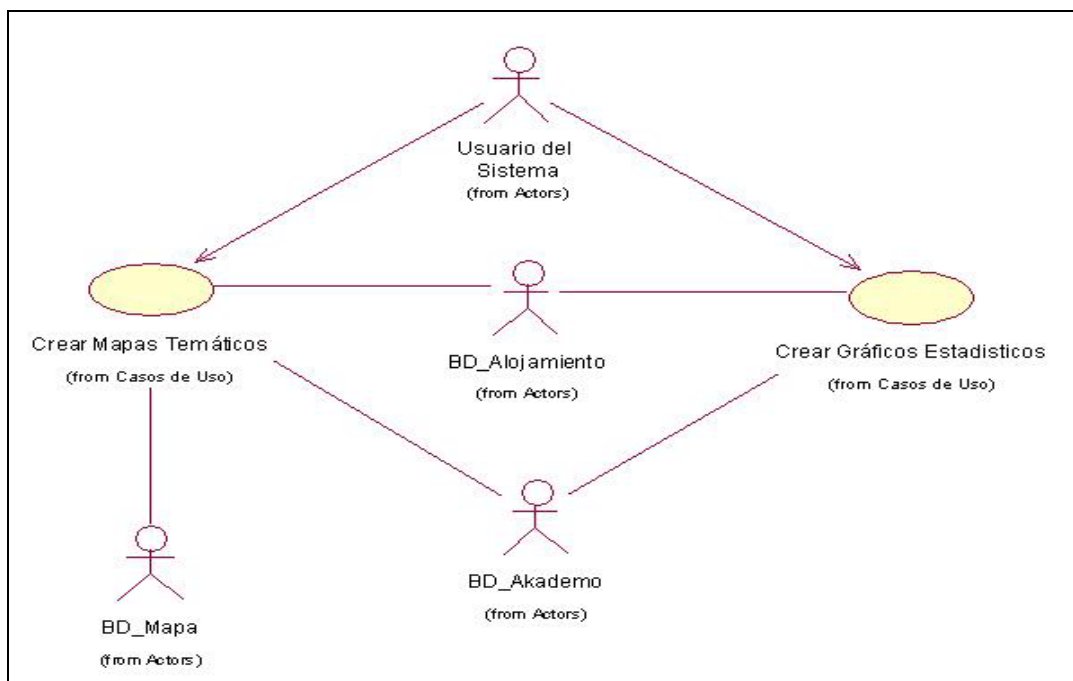


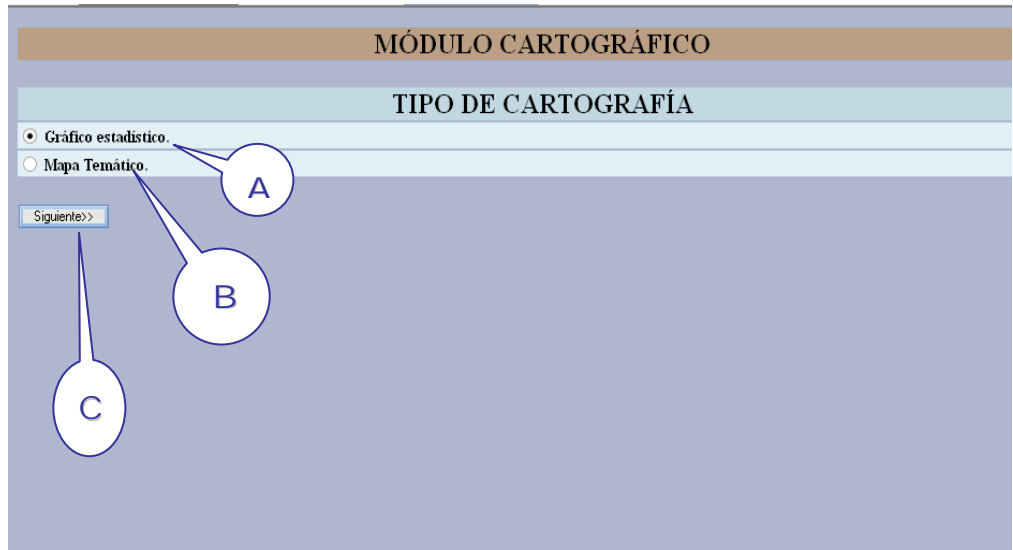
Figura 3. 2. Diagrama de Casos de Uso del Sistema

3.7. Expansión de los Casos de Uso.

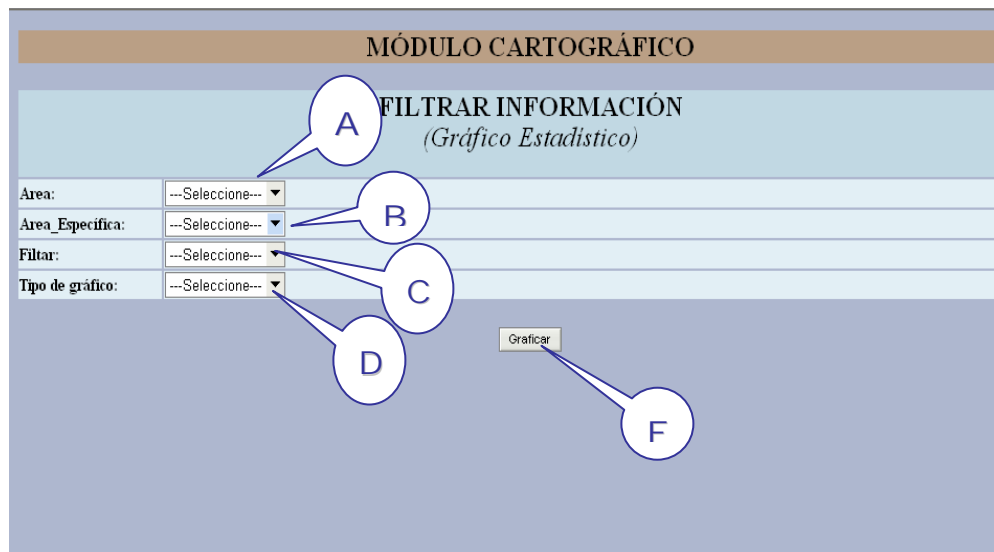
Tabla 3. 4. Caso de Uso Crear Gráfico Estadísticos

Caso de Uso:	Crear Gráfico Estadístico
Actor(es):	Usuario (inicia), BD_Alojamiento, BD_Akademo
Propósito:	Crear un Gráfico Estadístico
Resumen:	El caso de uso inicia cuando el usuario necesita obtener una información y desea visualizarla a través de gráficos estadísticos y para ello este puede seleccionar el área de donde este interesado, también permite seleccionar el tipo de gráfico que se desea crear y de esta forma quedará visualizada la información que el usuario desea.

Referencias:	R1
Precondiciones:	El Usuario debe de estar interactuando directamente con (GIS) de la universidad, levantar el módulo cartográfico.



Pantalla 1



Pantalla 2



Pantalla 3

Acción del Actor	Respuesta del Sistema
	1. El sistema muestra la pagina principal Pantalla 1
2. El usuario escoge la opción de gráficos estadísticos (A) y pulsar el botón Siguiente (C)	3. El sistema le muestra la Pantalla 2 donde el usuario tendrá la posibilidad de escoger las informaciones que desea estudiar.
4. El usuario seleccionará el área de en la cual está interesado (A).	5. Si elige: a) General, ir a sección General b) Residencia, ir a sección Residencia c) Manzanas, ir a sección Manzanas d) Edificio, ir a sección Edificio
6. Escoge el Tipo de Gráfico (D), la información que desea filtrar (C). y de ser necesario el área específica (B) Pulsa el botón Graficar (E)	7. El sistema le muestra el gráfico deseado. Pantalla 3
Sección: "General"	
	1. El sistema da la posibilidad de escoger el tipo de información de la cual necesita un reporte, habilitando Filtrar (C) estas

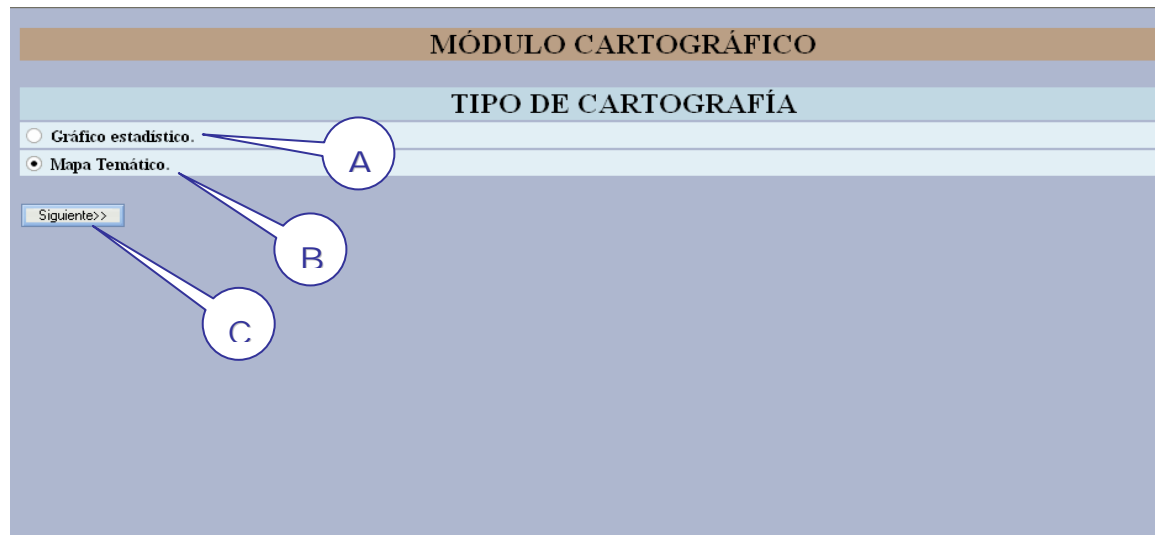
	<p>pueden ser:</p> <ul style="list-style-type: none"> • Sexo • Matrícula por provincia • Capacidad total por Residencia. • Capacidad total por Facultad • Cantidad estudiante evaluados de B por residencia • Cantidad estudiante evaluados de R por residencia • Cantidad estudiante evaluados de M por residencia
Sección: “Residencia”	
	<p>1. El sistema da la posibilidad de escoger de qué residencia específica desea obtener información (Número de Residencia) Área_Específica (B), el tipo de información de la cual necesita un reporte, Filtrar (C). que puede ser:</p> <ul style="list-style-type: none"> • Sexo • Matrícula por provincia • Capacidad total por Manzanas. • Capacidad total por Facultad • Cantidad estudiante evaluados de B por manzanas • Cantidad estudiante evaluados de R por manzanas • Cantidad estudiante evaluados de M por manzanas
Sección: “Manzanas”	

	<p>1. El sistema da la posibilidad de escoger de que manzana específicamente desea obtener información (número de manzana)Area_Especifica (B), el tipo de información de la cual necesita un reporte, Filtrar (C) esta puede ser:</p> <ul style="list-style-type: none"> • Sexo • Matrícula por provincia • Capacidad total por Edificios. • Capacidad total por Facultad • Cantidad estudiante evaluados de B por edificios • Cantidad estudiante evaluados de R por edificios • Cantidad estudiante evaluados de M por edificios
Sección: “Edificios”	
	<p>1. El sistema da la posibilidad de escoger de que edificio específicamente desea obtener información (número de edificio) Area_Especifica (B), el tipo de información de la cual necesita un reporte, Filtrar (C) que puede ser:</p> <ul style="list-style-type: none"> • Sexo • Matrícula por provincia • Capacidad total por Aptos. • Capacidad total por Facultad • Cantidad estudiante evaluados de B

	<p>por aptos</p> <ul style="list-style-type: none"> • Cantidad estudiante evaluados de R por aptos • Cantidad estudiante evaluados de M por aptos
--	---

Tabla 3. 5. Caso de Uso Crear Mapa Temático

Caso de Uso:	Crear Mapa Temático
Actor(es):	Usuario (inicia) Usuario del Sistema (inicia), BD_Alojamiento, BD_Mapas, BD_Akademias
Propósito:	Crear Mapas Temáticos.
Resumen:	El caso de uso inicia cuando el usuario necesita visualizar información en forma de mapa temático para conocer la distribución de la misma en el mapa de la UCI
Referencias:	R2
Precondiciones:	El Usuario debe de estar interactuando directamente con (GIS) de la universidad, levantar el módulo cartográfico.



Pantalla 1

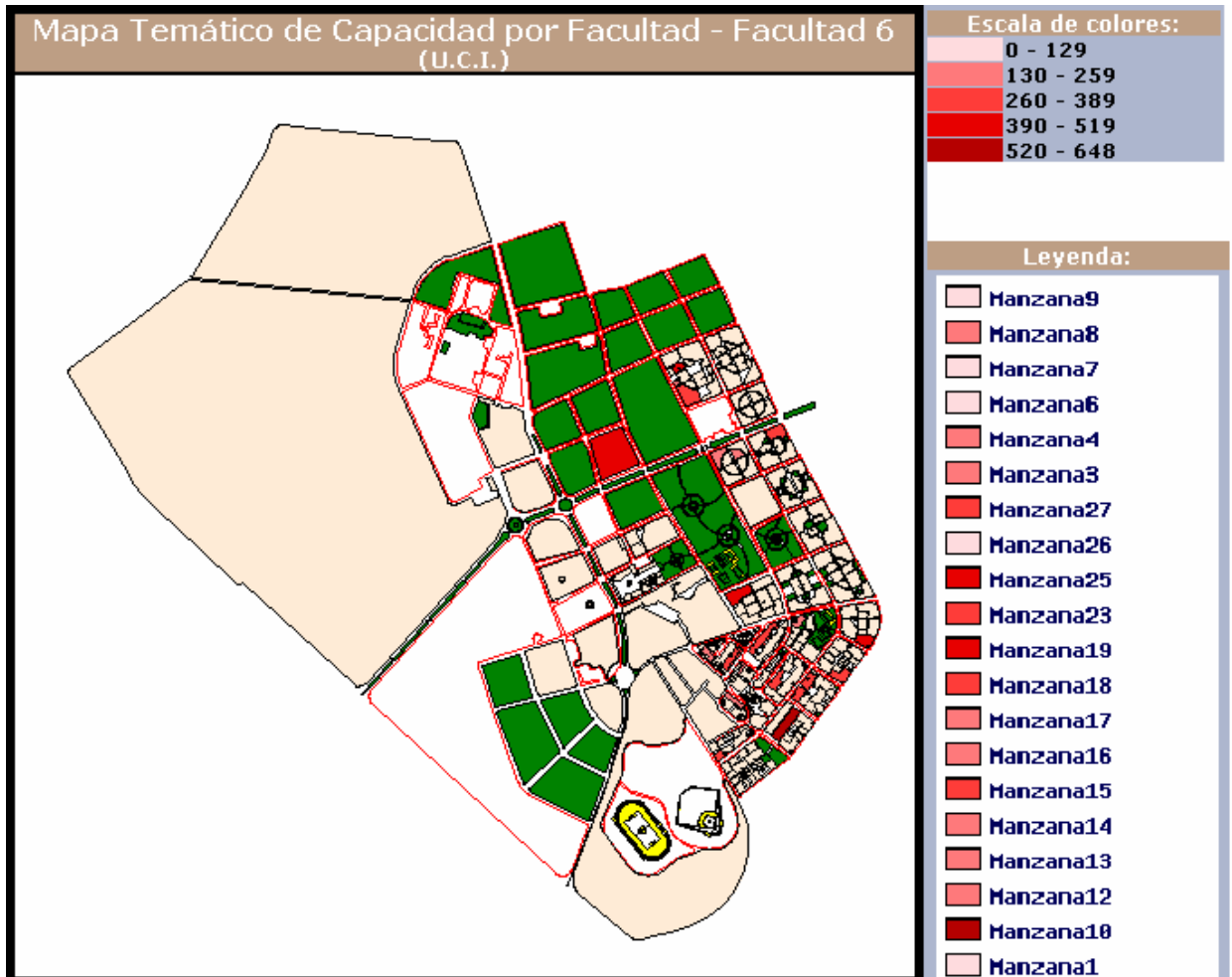
MÓDULO CARTOGRÁFICO

ALTRAR INFORMACIÓN

(Mapa Temático)

<u>Area:</u>	-Seleccione-
<u>Filtar:</u>	-Seleccione-
<u>Rango del filtro:</u>	-Seleccione-
<u>Color de escala:</u>	-Seleccione-
<u>Niveles de escala:</u>	-Seleccione-

Pantalla 2



Pantalla 3

Acción del Actor

Respuesta del Sistema

	1. El sistema muestra la pagina principas. Pantalla 1.
2. El usuario escoge la opción de mapa temático (B). y pulsa el botón siguiente(C)	3. El sistema muestra la pantalla 2.
4. El Usuario escoge por que áreas desea separar el Mapa temático (A).	5. El sistema mostrara las informaciones para realizar el mapa (B). En este caso podrán ser por: <ul style="list-style-type: none"> • Sexo • Matrícula por provincia • Capacidad total por Facultad • Cantidad estudiante evaluados de B • Cantidad estudiante evaluados de R • Cantidad estudiante evaluados de M
6. El usuario escoge la información que desea (B).	7. Si escoge: -Sexo -Matrícula por provincia -Capacidad por Facultad ir a sección Rango
8. El usuario selecciona el color con el cual quiere que se expresen sus resultados (D). y los niveles de escala (E) para saber la degradación y pulsa el botón Crear (F)	9. El sistema le mostrara el mapa temático reflejando la información solicitada. Pantalla 3
Sección: "Rango"	
	1. El sistema dará la posibilidad de escoger el rango de la información (C).

3.8. 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 módulo, estos fueron representadas mediante un Diagrama de Casos de Uso, y finalmente fueron descritas todas las acciones que realizan los actores. 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.

CAPITULO 4 CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA

4.1. Introducción.

En este capítulo se realiza el diseño de la propuesta de solución, creando los artefactos que ayudan a la construcción del proyecto, tratar con esto de eliminar todas las complicaciones que puedan aparecer en el camino hasta llegar a la culminación satisfactoria del módulo. Se presentan los principios a tener en cuenta para el diseño de la interfaz, se explica como se manejan lo errores. Se plantean los estándares de codificación que se van a utilizar y finalmente, se presenta el modelo de despliegue y el modelo de implementación para una mejor descripción de la solución.

4.2. Modelo de análisis.

El análisis se basa en un modelo de objetos conceptuales, que se denomina modelo de análisis. Este modelo estructura los requisitos de un modo que facilita su comprensión, su modificación, en general, su mantenimiento. Se describe utilizando el lenguaje de los desarrolladores, por tanto puede introducir un mayor formalismo y ser utilizado para razonar sobre los funcionamientos internos del sistema. [18]

Éste flujo como se puede ver es de gran importancia, porque permite un acercamiento a la lógica de cómo se resolverá el problema. Facilita una comprensión en profundidad de los requisitos y da la primera vista del sistema por dentro, se puede plantear que es donde se modela el sistema y se encuentra su forma.

4.2.1. Diagramas de Clases de Análisis.

Las clases de análisis se centran en el tratamiento de los requisitos funcionales y pospone los no funcionales. Tienen atributos y entre ellas se establecen relaciones estos son mucho más conceptuales que sus contrapartidas de diseño e implementación. Siempre encajan en uno de estos tres estereotipos:[18]

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

Clase de Controladoras: 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.

Clases Entidades: Modelan información que posee larga vida y que es a menudo persistente.

En el caso de este sistema este tipo de clases tienen una peculiaridad. Este módulo no cuenta con una base de datos fija, por lo tanto no cuenta con clases persistentes, como se trabaja con diferentes bases de datos, tomamos como clases entidades esas bases de datos para comprender mejor el modelado del sistema y que se entienda mejor como se desarrollan los casos de uso.

A continuación le presentamos el diagrama de clases del análisis que no es más que un artefacto en el que se representan los conceptos en un dominio del problema. Representa las clases a tener en cuenta, aunque no las que exactamente serán implementadas.

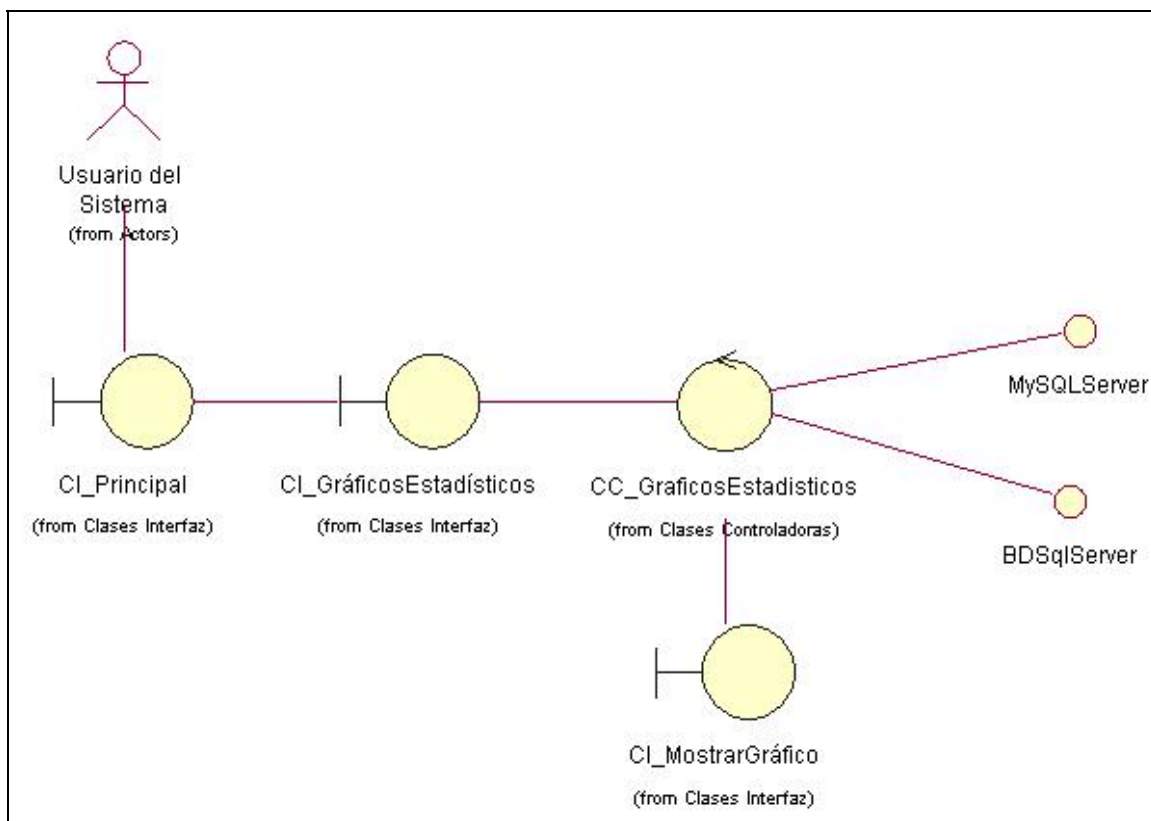


Figura 4. 1. Diagrama de Clases de Análisis: CU Crear Gráfico Estadístico

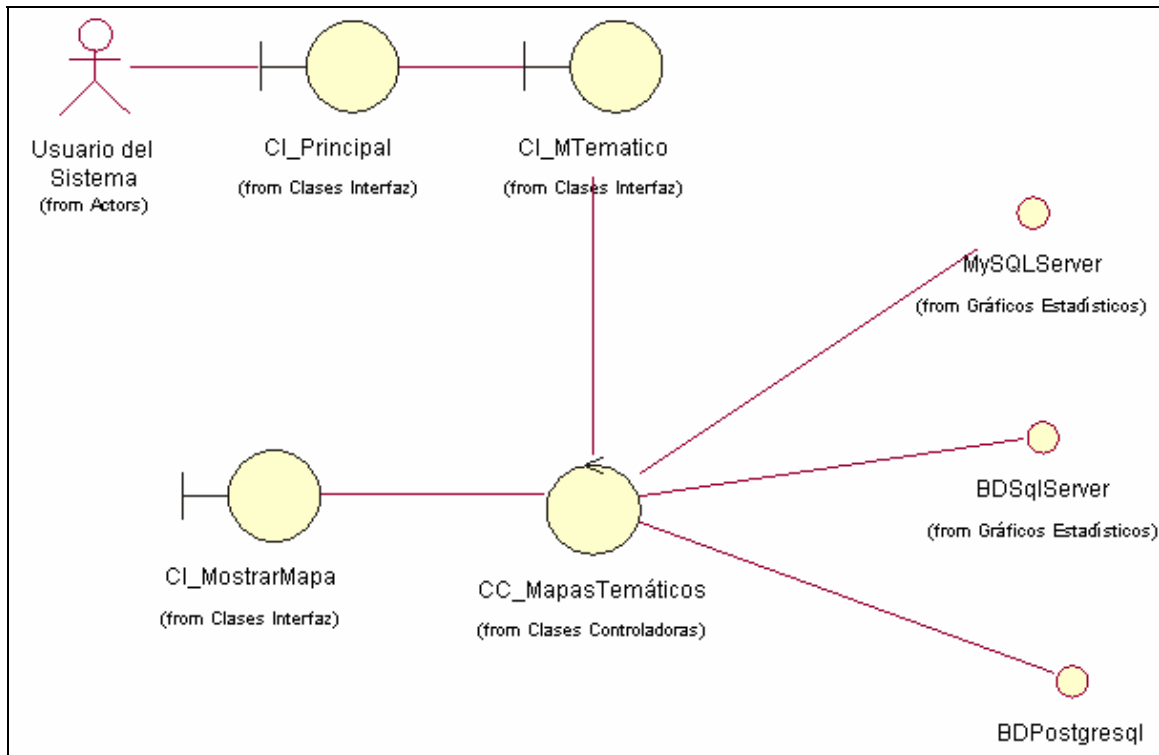


Figura 4. 2. Diagrama de Clase de Análisis: CU Crear Mapa Temático

4.3. Modelo de Diseño.

“El modelo de diseño es un modelo de objetos que describe la realización física de los casos de uso, centrándose en cómo los requisitos funcionales y no funcionales, junto con otras restricciones relacionadas con el entorno de implementación, tienen impacto en el sistema”. [18]

4.3.1. Diagrama de Clases de Diseño.

Una clase de diseño es una abstracción sin costuras de una clase o construcción similar en la implementación del sistema.

El diagrama de clase de diseño muestra la relación de estas clases en el sistema. Este módulo ha sido diseñado con una arquitectura de tres capas, que se han dividido además por paquetes para su mejor comprensión. Figura 4.3

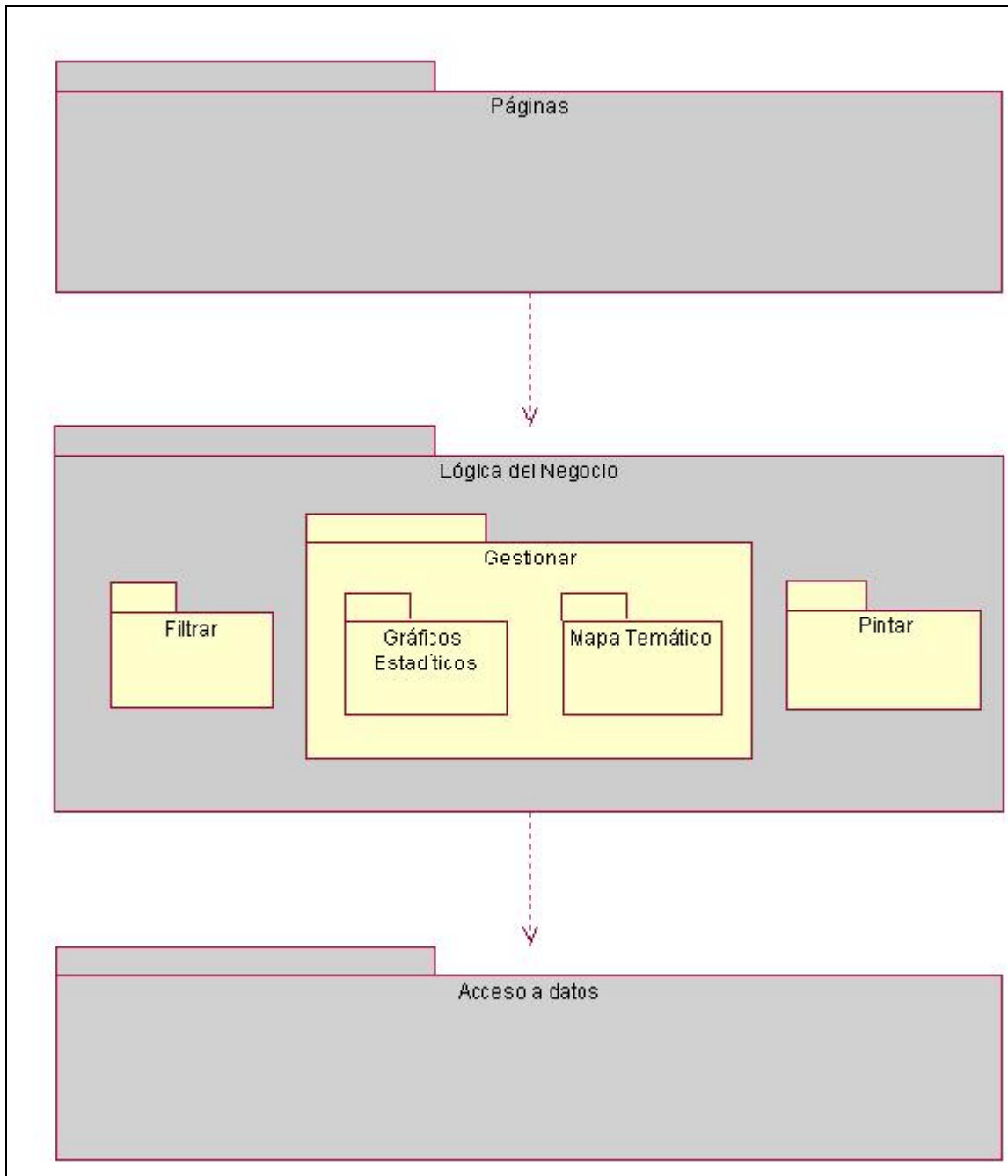


Figura 4. 3. Arquitectura

En la capa Páginas se encuentran todas las páginas que interactúan con el usuario.

Figura 4.4

La capa de Lógica del Negocio está dividida por su complejidad en tres paquetes:

El paquete Filtrar es el encargado de leer las áreas y los filtros desde el XML donde se encuentran estas. Que son las que darán la posibilidad al usuario de realizar estudios

estadísticos. Figura 4.5

El paquete Gestionar es que se encarga de gestionar toda la información necesaria para realizar los gráficos y los mapas interactúa con la capa de acceso a datos.

Figura 4.6 y 4.7

El paquete Pintar contiene las funciones que permite pintar un gráfico estadístico. Figura 4.8

La capa Acceso a Datos es la encargada de conectarse a las diferentes bases de datos y extraer de ellas las informaciones requeridas. Figura 4.9

Se confeccionó un diagrama de clases para cada paquete, con el objetivo de lograr un mayor desacoplamiento, reutilización y legibilidad de dichos diagramas.

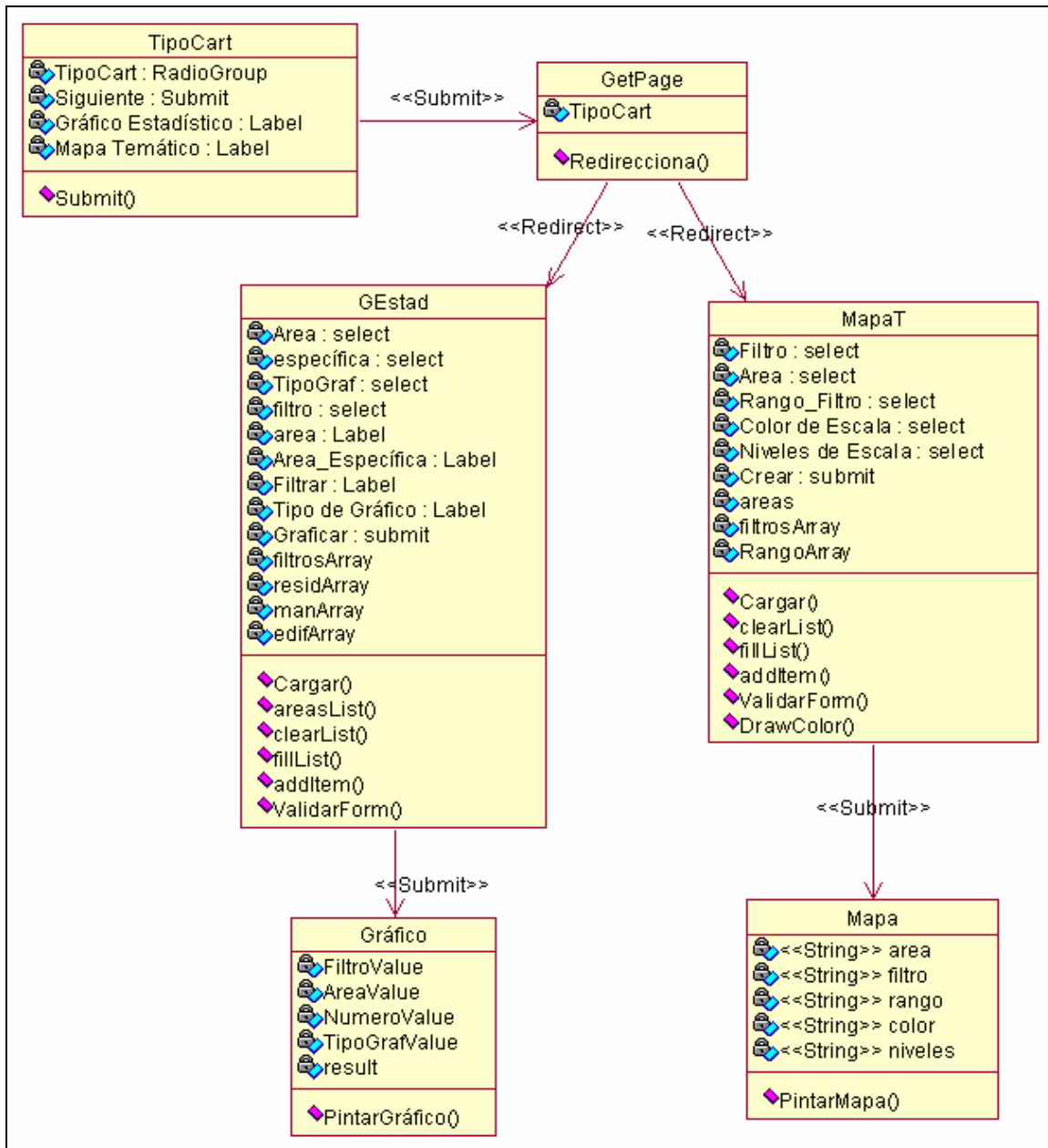


Figura 4. 4. Paquete Páginas

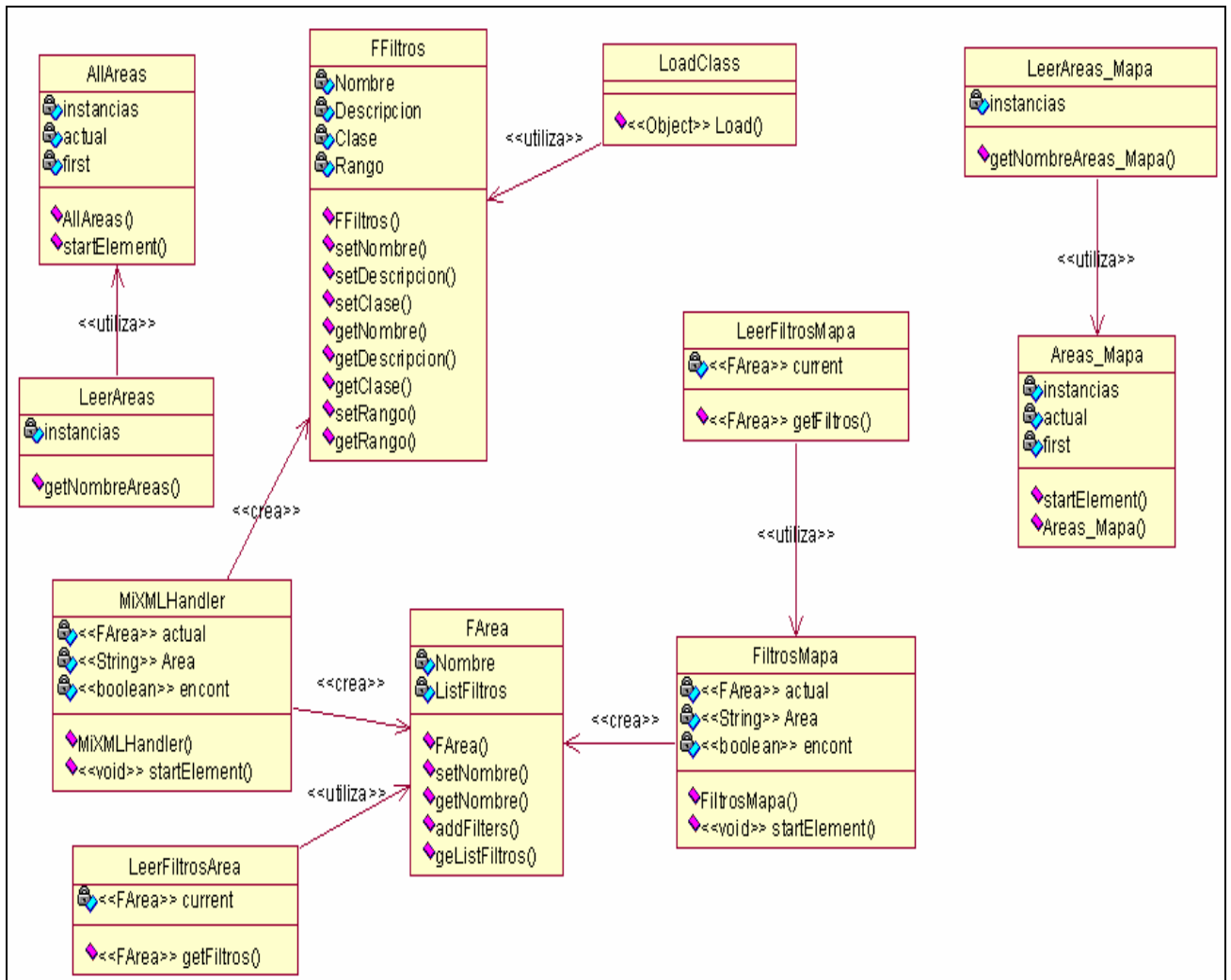


Figura 4. 5. Paquete Filtrar

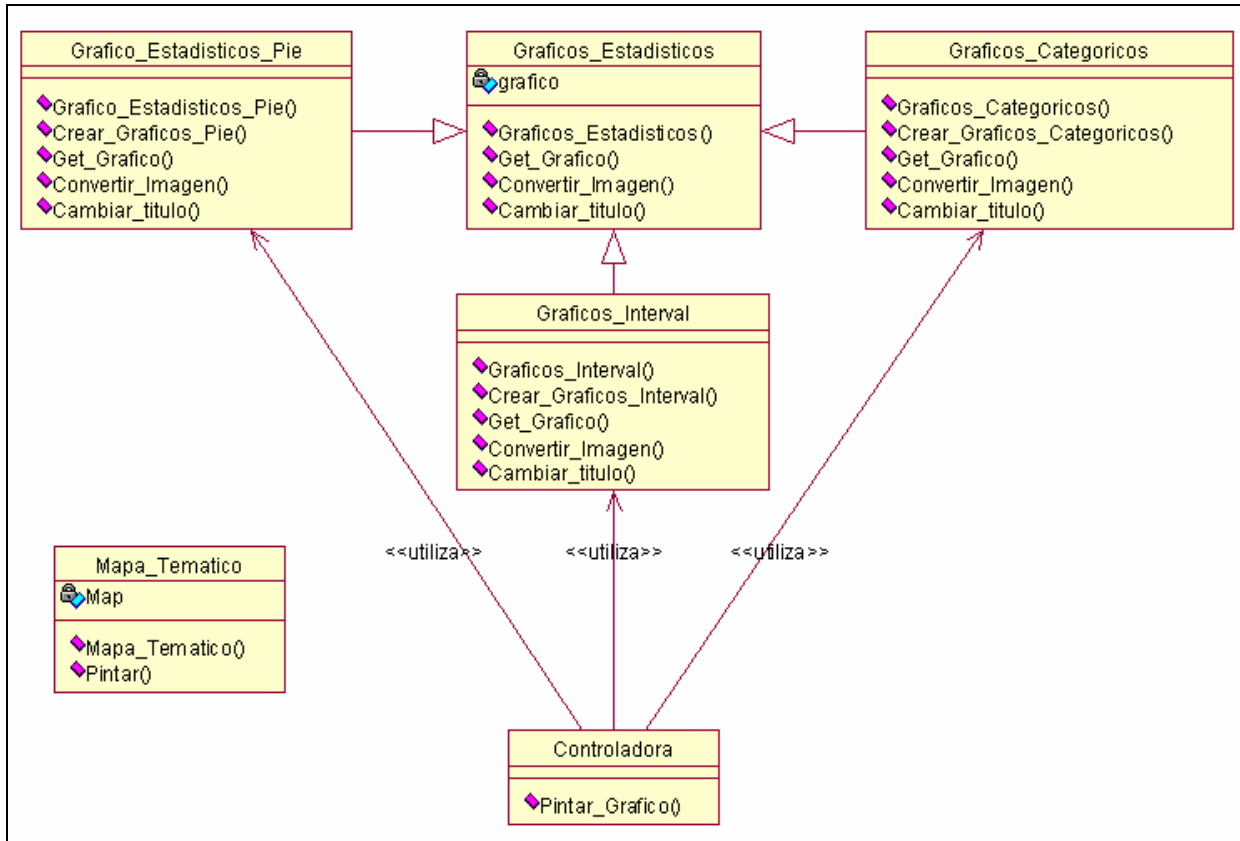


Figura 4. 6. Paquete Pintar

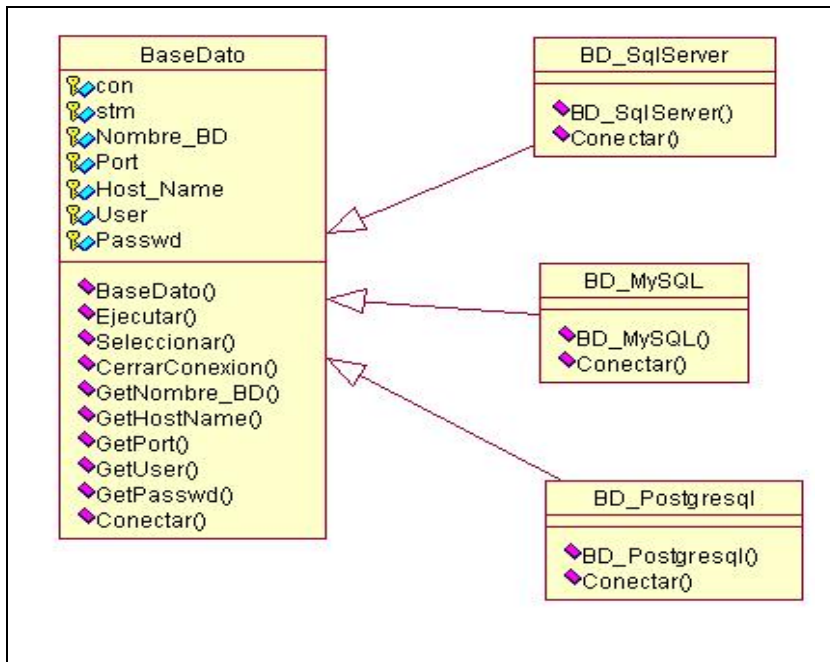


Figura 4. 7. Paquete Acceso a Datos

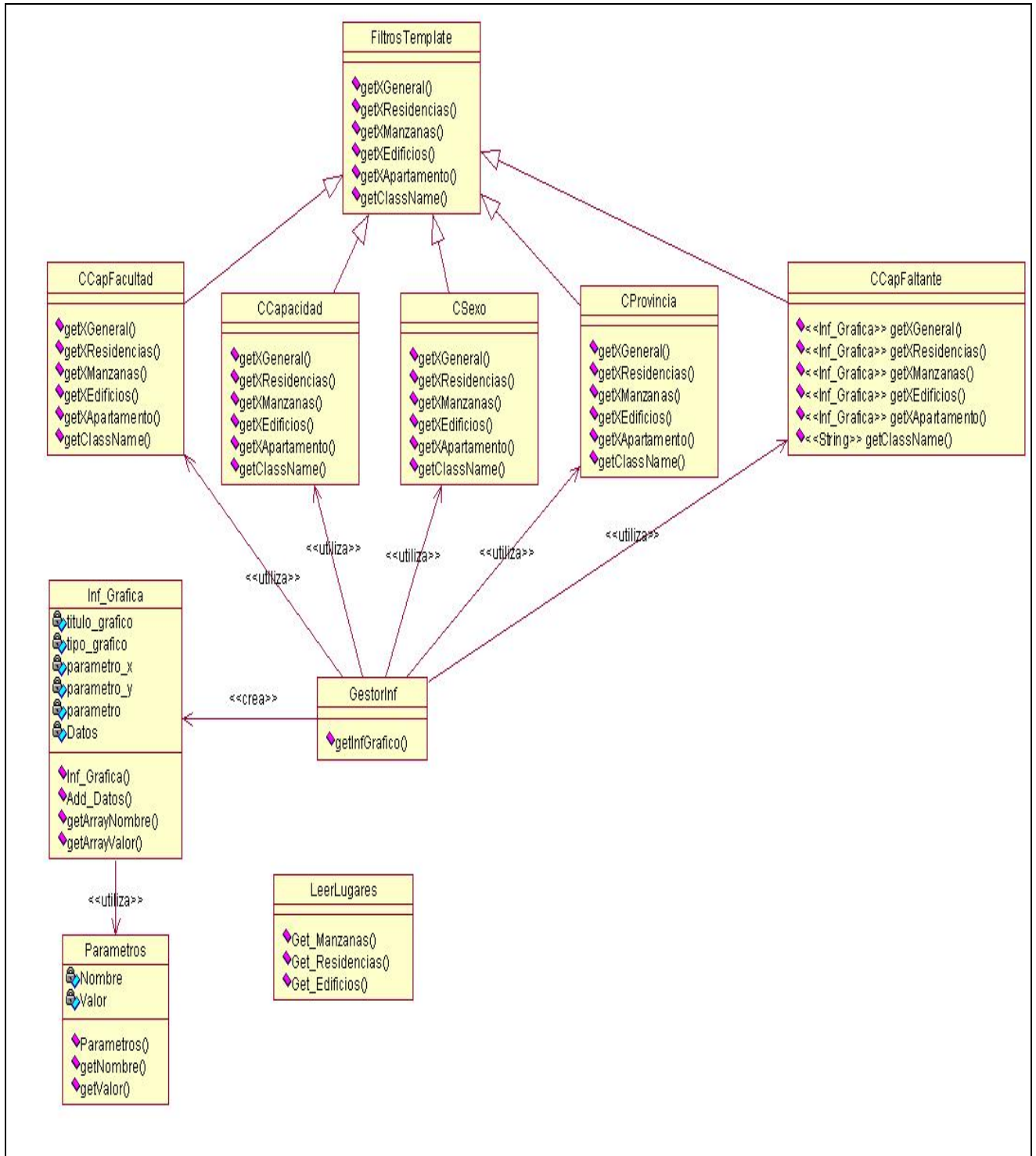


Figura 4. 8. Gestionar Gráfico Estadístico

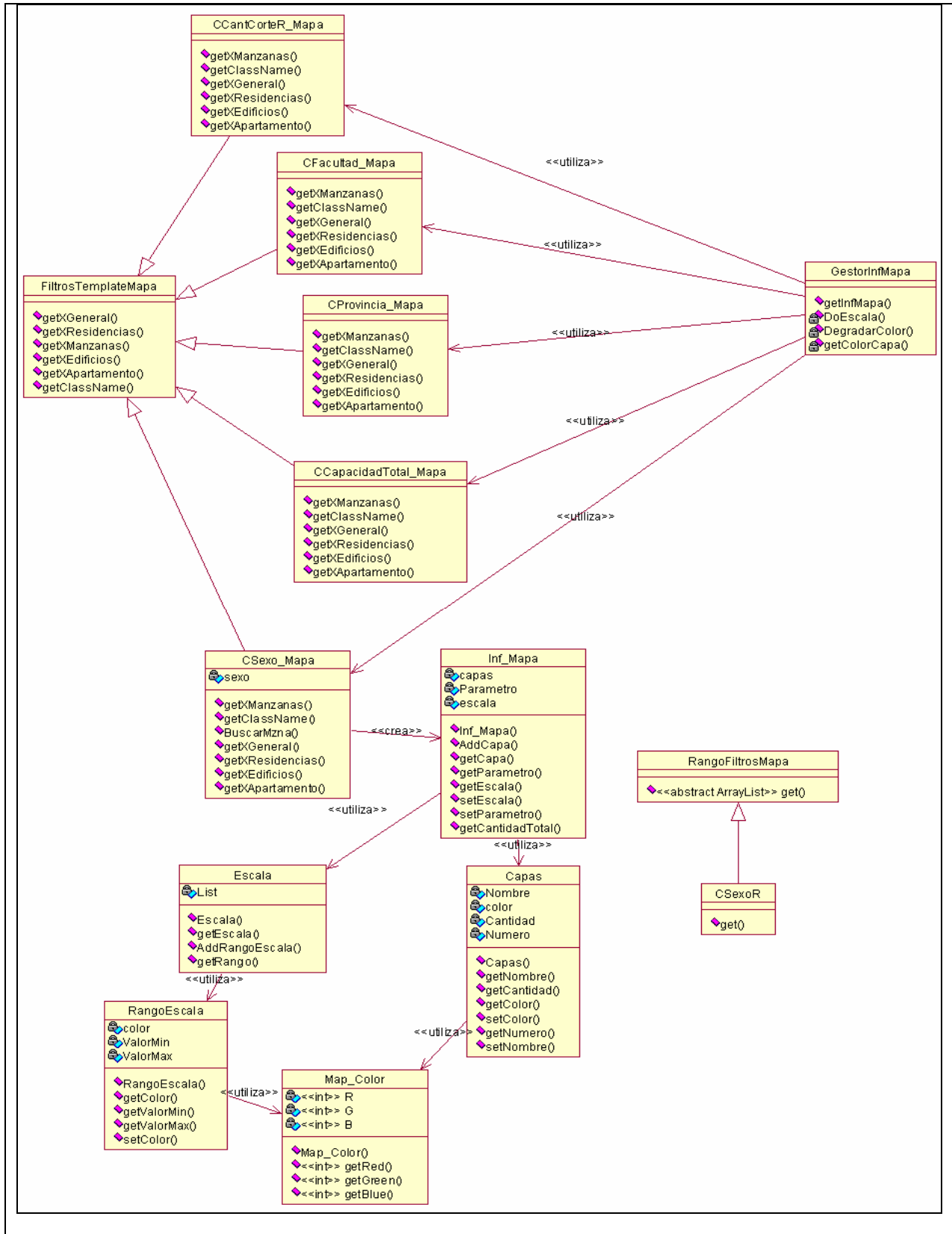


Figura 4. 9. Gestionar Mapa Temático

4.3.2. Diagrama de Clases Web.

La forma tradicional de modelar clases no es aplicable a la hora de diseñar una aplicación Web. Por ese motivo los especialistas de Rational crearon una extensión para UML que se adapta a la arquitectura de este tipo de sistemas.[18]

Para cada caso de uso del sistema se elaboró un diagrama de clases Web de forma tal que se facilite la comprensión de cómo se relacionan los distintos componentes en la realización de cada uno de ellos. Además de un mapa de navegación para que se vea por las páginas que se pasan hasta obtener el resultado buscado. Figura 4.21 de los Anexos.

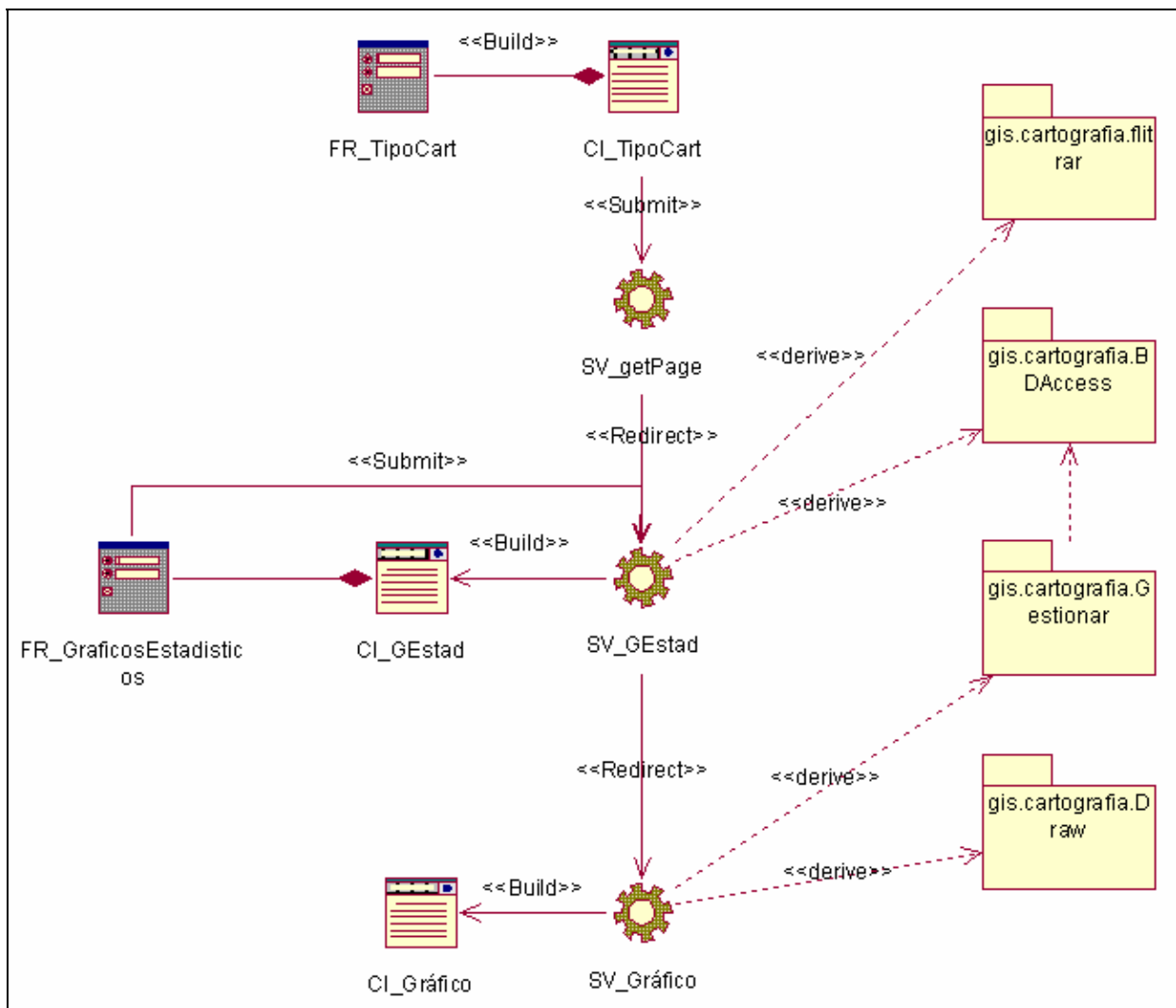


Figura 4. 10. DCW: Crear Gráfico

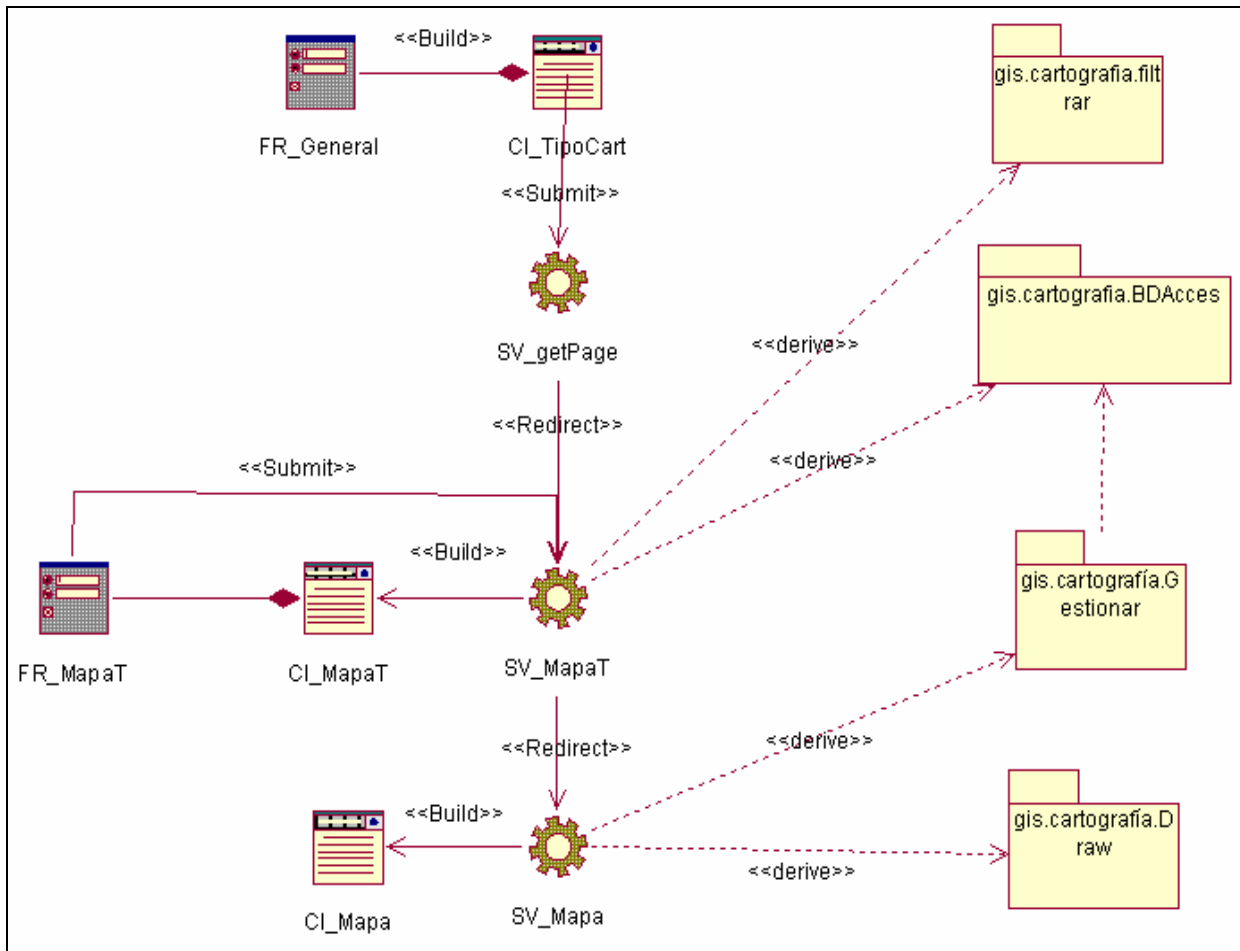


Figura 4. 11. DCW: Crear Mapa Temático

4.3.3. Diagrama de Interacción.

En el diseño, es recomendable representar las interacciones que los objetos que participan en la realización de un caso de uso, mediante diagramas de secuencias de esta forma podemos encontrar mejor las secuencias de interacciones detalladas y ordenadas en el tiempo. En los diagramas de secuencias, se muestran las interacciones entre objetos mediante transferencias de mensajes entre objetos o subsistemas. Cuando se dice que un subsistema recibe un mensaje, realmente es un objeto de una clase del subsistema el que recibe un mensaje, y lo mismo para el caso de enviar.[18]

Para un mejor entendimiento los diagramas se separaron por las diferentes situaciones que pueden tener lugar en cada caso de uso.

A continuación les presentamos uno de cada uno en forma de ejemplos los demás se pueden ver en el Anexo a partir de la Figura 4.22

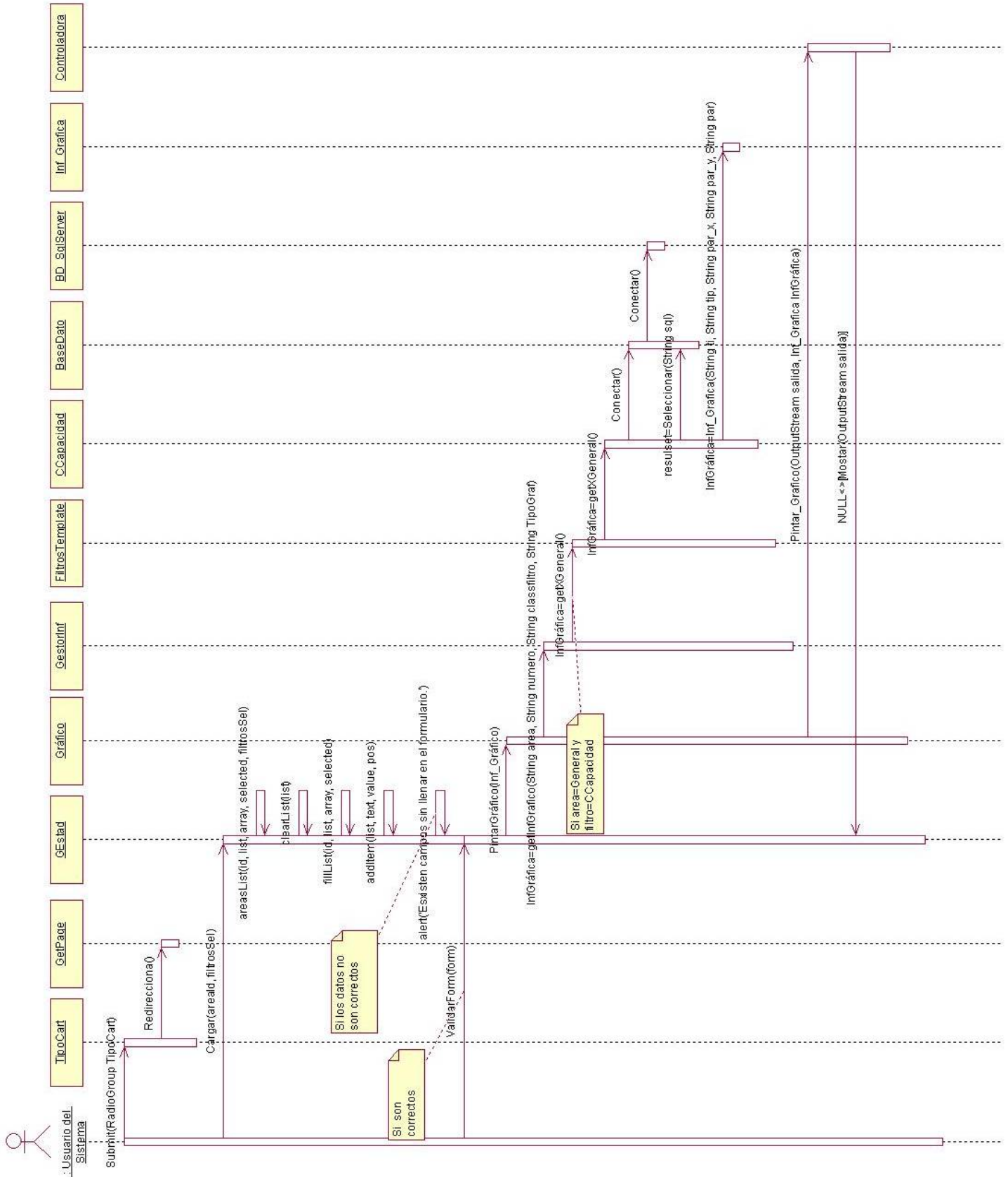


Figura 4. 12. DS: Crear Gráfico Estadísticos

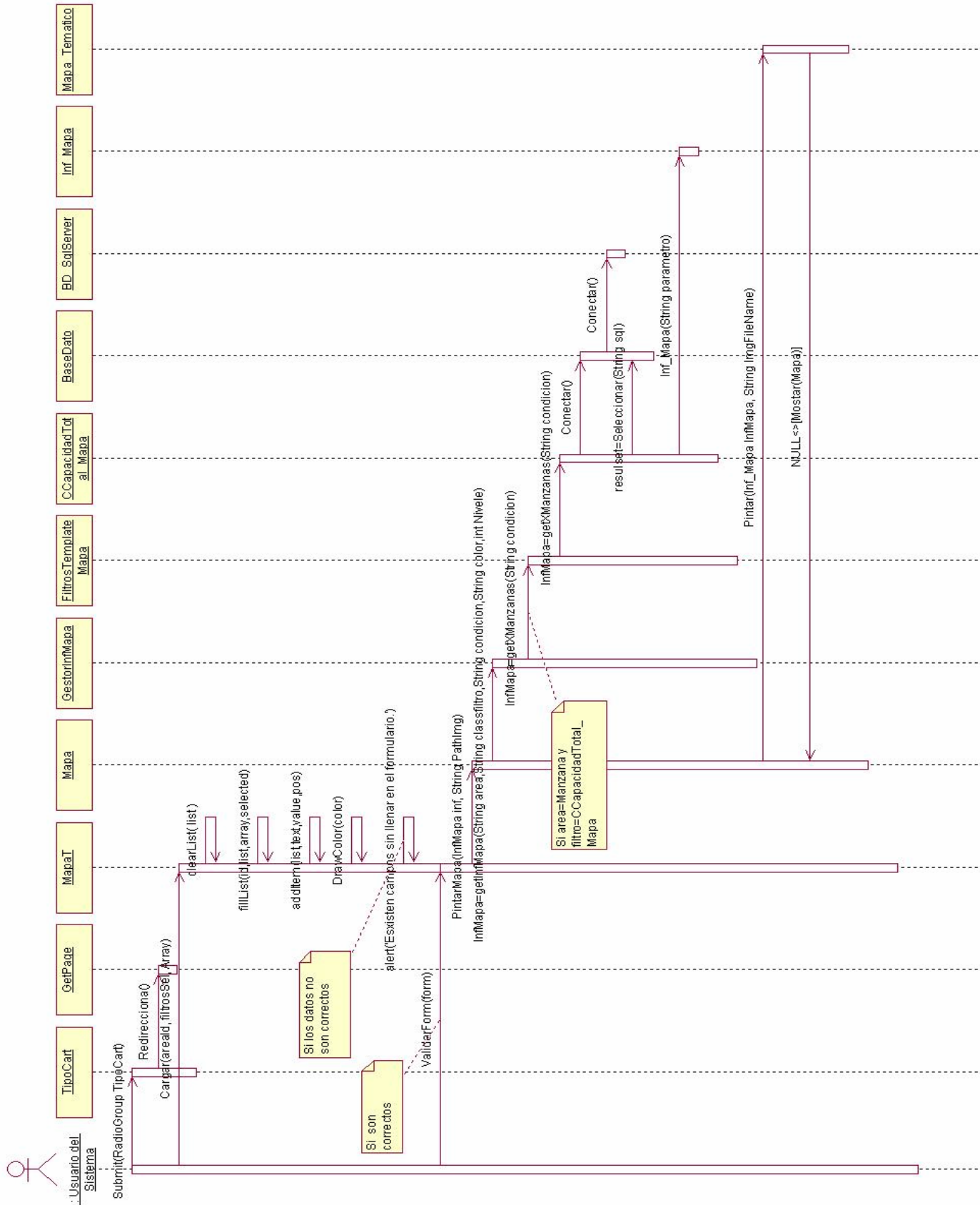


Figura 4. 13. DS: Crear Mapa Temático

4.4 Principios de Diseño de la Interfaz.

El diseño de la interfaz es un aspecto muy delicado, requiere mucho cuidado porque es la parte del sistema que interactúa directamente con el usuario. Por tanto debe de ser lo más amigable y entendible posible, pensando siempre que el usuario no tienen porque tener un alto grado informático. Una aplicación debe tener una interfaz con buena usabilidad, que tenga un buen diseño gráfico. Y que le brinde al usuario, alcanzar el objetivo por el cual esta trabajando con nuestro sistema con un mínimo esfuerzo. El diseño de la interfaz de este trabajo estará regido por las características anteriormente mencionadas.

4.4.1 Estándares de la interfaz de la aplicación.

Aunque por particularidad del módulo no es un objetivo principal, el diseño de una interfaz muy poderosa, se respetó en todas las páginas un formato para que el usuario pueda utilizar el sistema de manera cómoda. La cabecera de cada pagina contiene el nombre del módulo, de forma que el cliente sepa donde está, más abajo separado con otro color se le especifica el tipo de cartografía que puede hacer en cada momento y con mucha claridad se le explica más abajo los datos a los que puede acceder.

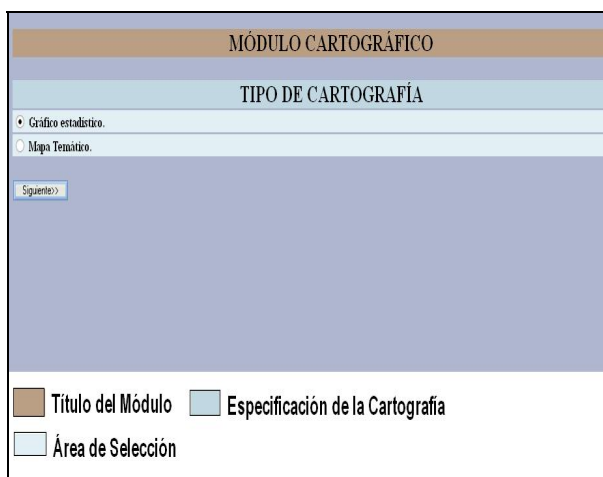


Figura 4. 14. Diseño Interfaz

Aunque el entorno de desarrollo que se utilizó durante la implementación del módulo no brinda muchas facilidades para el diseño de interfaz. Se trató en todo momento de mantener un único formato en todas las páginas. Teniendo en cuenta que este módulo será utilizado por el Sistema de Información Geográfico, se trabajó pensando en que la

interfaz del mismo no se separará mucho de la del (GIS), es por esta razón que se utilizan colores tan parecidos a los que tiene éste. Así se garantiza que cuando el usuario acceda a este módulo no se sienta alejado del Sistema de Información geográfico.

4.4.2 Formato de los reportes.

El módulo es en su esencia es un sistema de reportes, que permitirá a los usuarios del Sistema de Información Geográfico de la universidad, puedan tener conocimientos concreto de diferentes áreas de la misma. Como se ha venido explicando desde el capítulo 1 los reportes se darán en forma de gráficos estadísticos y mapas temáticos. Lográndose con estas herramientas una mejor comprensión de los datos solicitados por los usuarios.

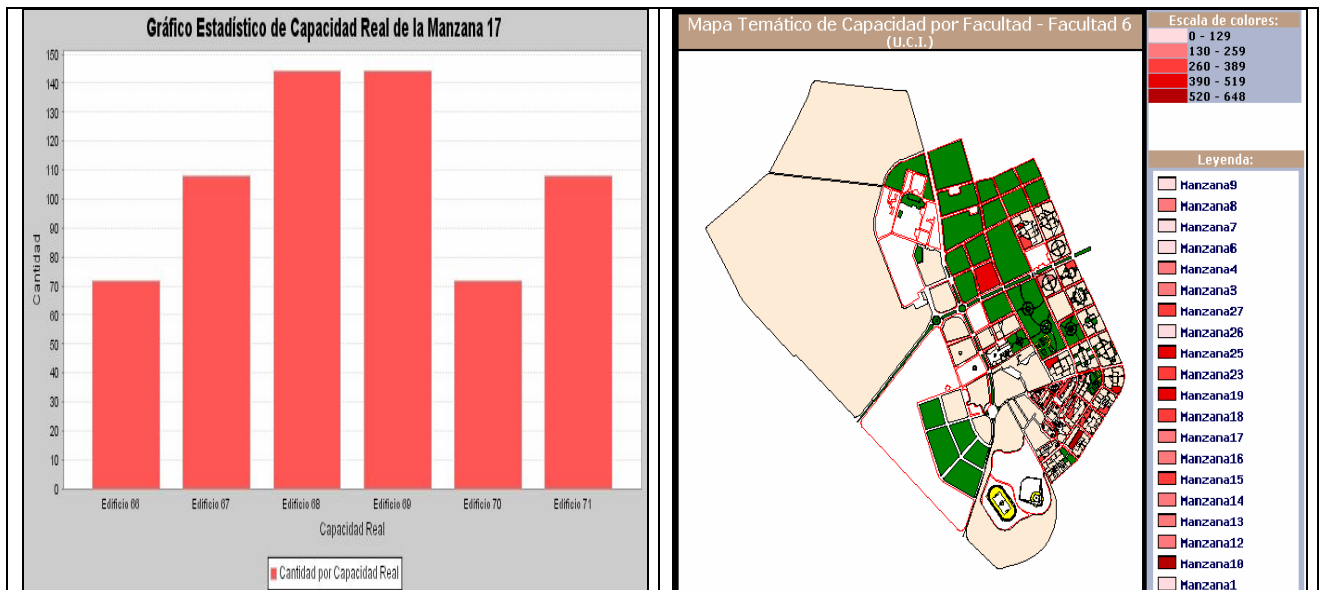


Figura 4. 15. Reporte

4.4.3 Tratamiento de excepciones.

El buen funcionamiento de cualquier aplicación está dado por la capacidad de la misma de detectar errores y recuperarse de ellos. En el este módulo el usuario no tiene necesidad de introducir ningún dato, además no se le da la posibilidad de hacerlo, lo único que tiene es que escoger, facilitándonos el trabajo a la hora de tratar los errores de entrada de datos al sistema. Es por esto que sólo tenemos que garantizar que el usuario escoja todos los datos que necesita el sistema.

Utilizando el lenguaje JavaScript, se logra que si el cliente por error no escogiera todos los datos necesarios, la aplicación le emite un mensaje de alerta para que rectifique y así

el sistema funcionará y sus objetivos serán cumplidos.

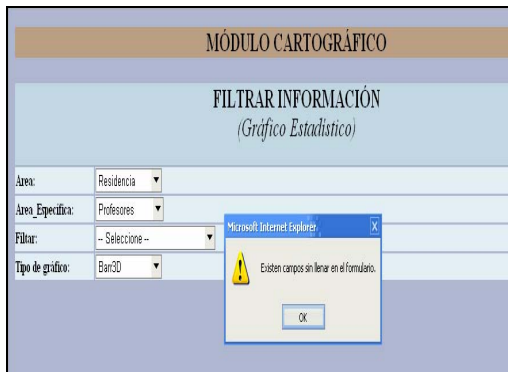


Figura 4. 16. Errores

4.5. Estándares de codificación.

Con vista a garantizar la homogeneidad del código del proyecto, se utiliza un estándar para escribirlo, con esto además se logra una menor posibilidad de cometer errores. Y se logra que el código se mucho más fácil de leer y comprender por otros desarrolladores. Para cumplir con estos paradigmas se establece el estilo siguiente:

Comentarios: Para los comentarios de más de una líneas los comentarios se escriben comenzando con los por */** y terminando con **/*, y *//* para los de una sola línea.

Declaraciones:

- Las constantes se declaran en mayúsculas todas las letras que la componga ejemplo: **float** *PI* = 3.14;
- Las variables deben ser explícitas para saber realmente lo que significan con minúscula
- Para declarar una clase se comienza con letra mayúscula en caso de necesitar escribir otro sustantivo para hacer la clase más entendible se comienza también con Mayúscula y en algunos casos se separa con `_` para no hacer una palabra engorrosa. Ejemplo: **class** *Inf_Grafica*
- Para una mejor organización del proyecto las clases se han guardado en paquetes los cuales se nombran con el siguiente prototipo `gis`(porque es para él)+`cartografía`(porque es el nombre del módulo que se desarrolla)+el nombre del

paquete(que indica el tipo de clases que en el se encuentran). Ejemplo *gis.cartografia.Draw* que por su nombre indica que sus clases son para pintar.

Espacios en blanco: Colocar espacios en blanco entre operadores lógicos-aritméticos y sus operandos.

Miscelánea: Poner al mismo nivel del bloque al que pertenecen Las llaves para abrir y cerrar un método o un bloque de control de flujo. Ejemplo:

```
if(a != 1)
{
...
}
```

4.6. Modelo de 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 nodos de cómputo. Se utiliza como entrada fundamental en las actividades de diseño e implementación debido a que la distribución del sistema tiene una influencia principal en su diseño. [18]

A continuación le mostramos el diagrama de despliegue correspondiente al sistema de esta investigación:

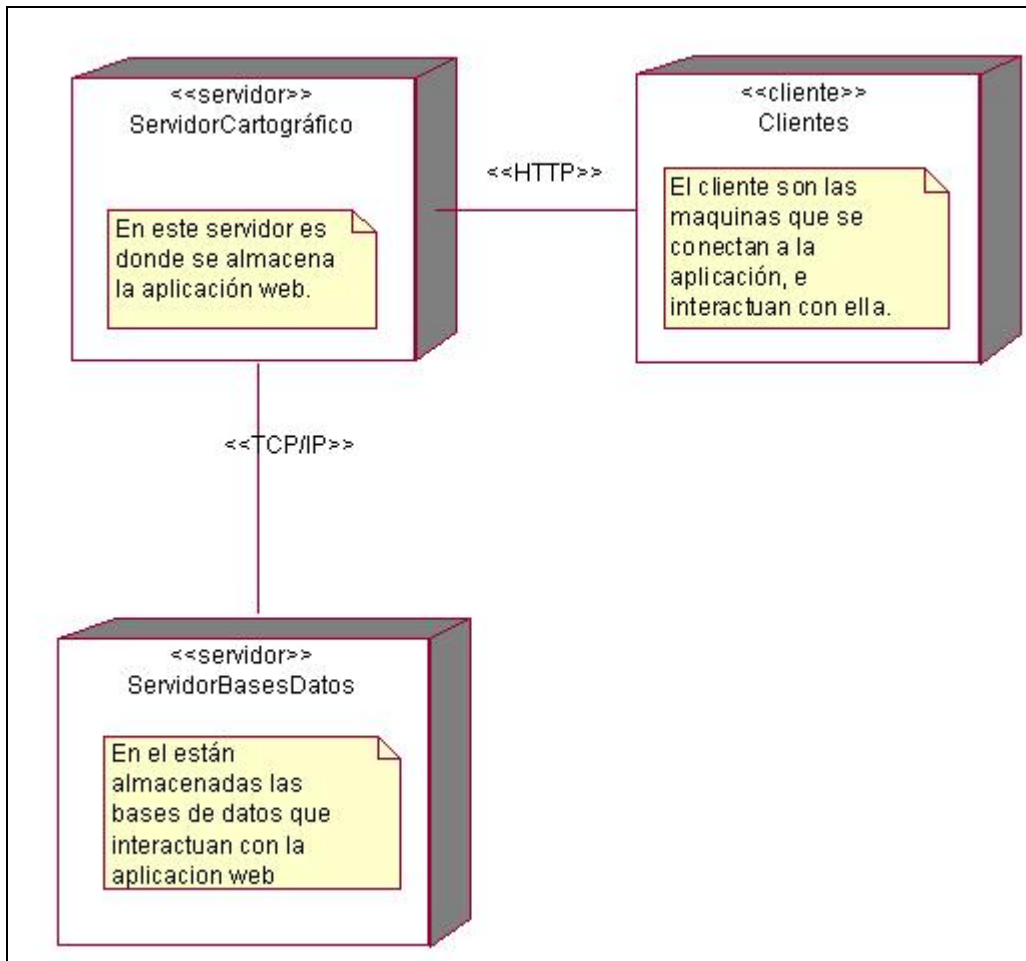


Figura 4. 17. Modelo de Despliegue

4.7. Modelo de Implementación.

El modelo de implementación describe, como se implementan las clases en término de componentes, como ficheros de código fuente, ejecutables entre otros. Describe también como se organizan los componentes de acuerdo con los mecanismos de estructuración y modularización disponible en el entorno de implementación y en el lenguaje de programación utilizado. Además muestra las dependencias entre componentes.[18]

Este es el modelo de implementación de este módulo separado por la capaz de la arquitectura para su mejor comprensión:

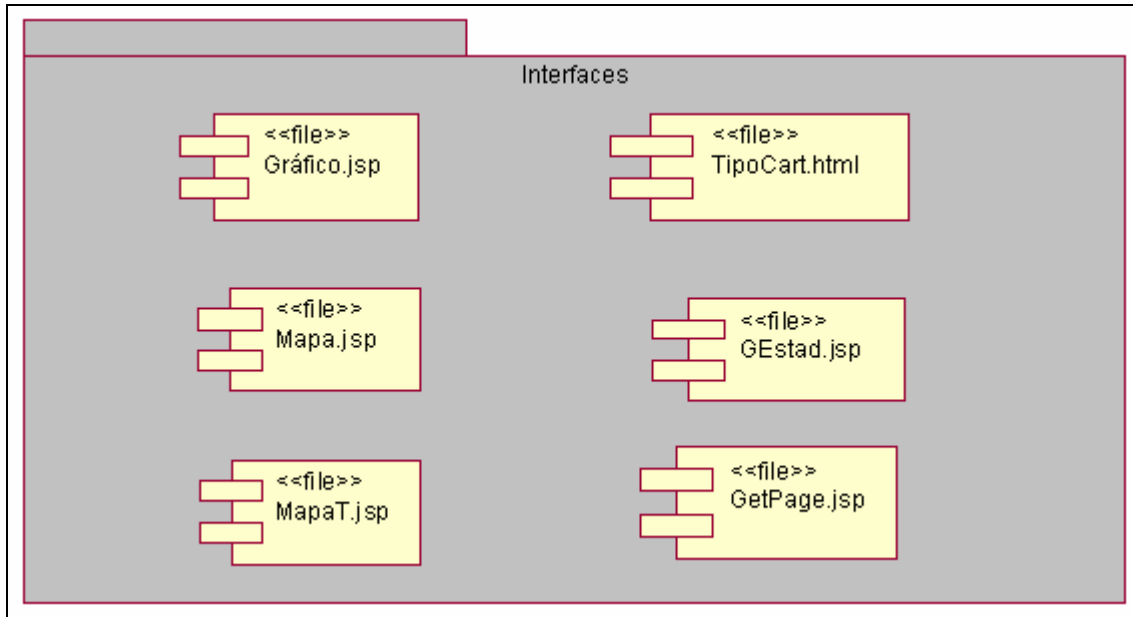


Figura 4. 18. Diagrama de Componentes de la Capa Presentación

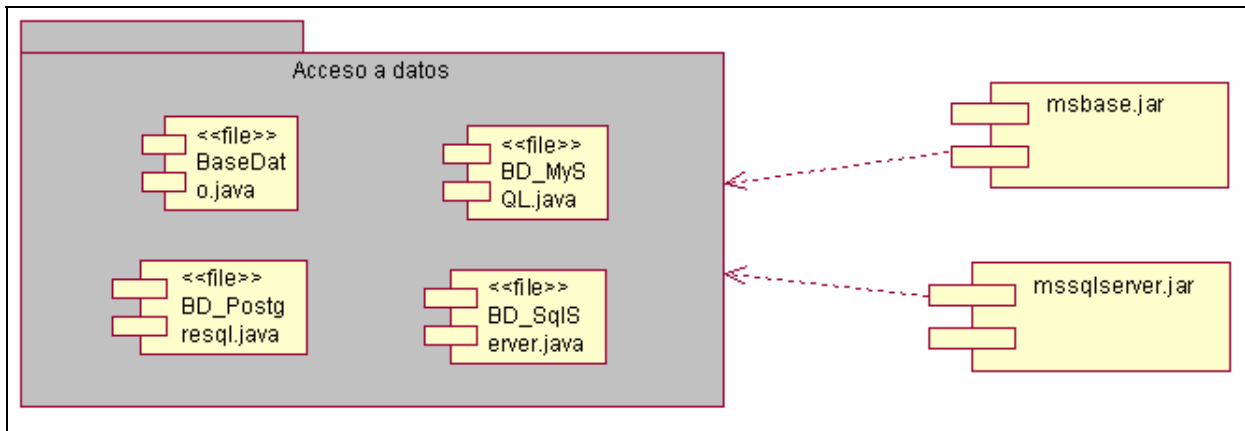


Figura 4. 19. Diagrama de Componentes de la Capa Acceso a Datos

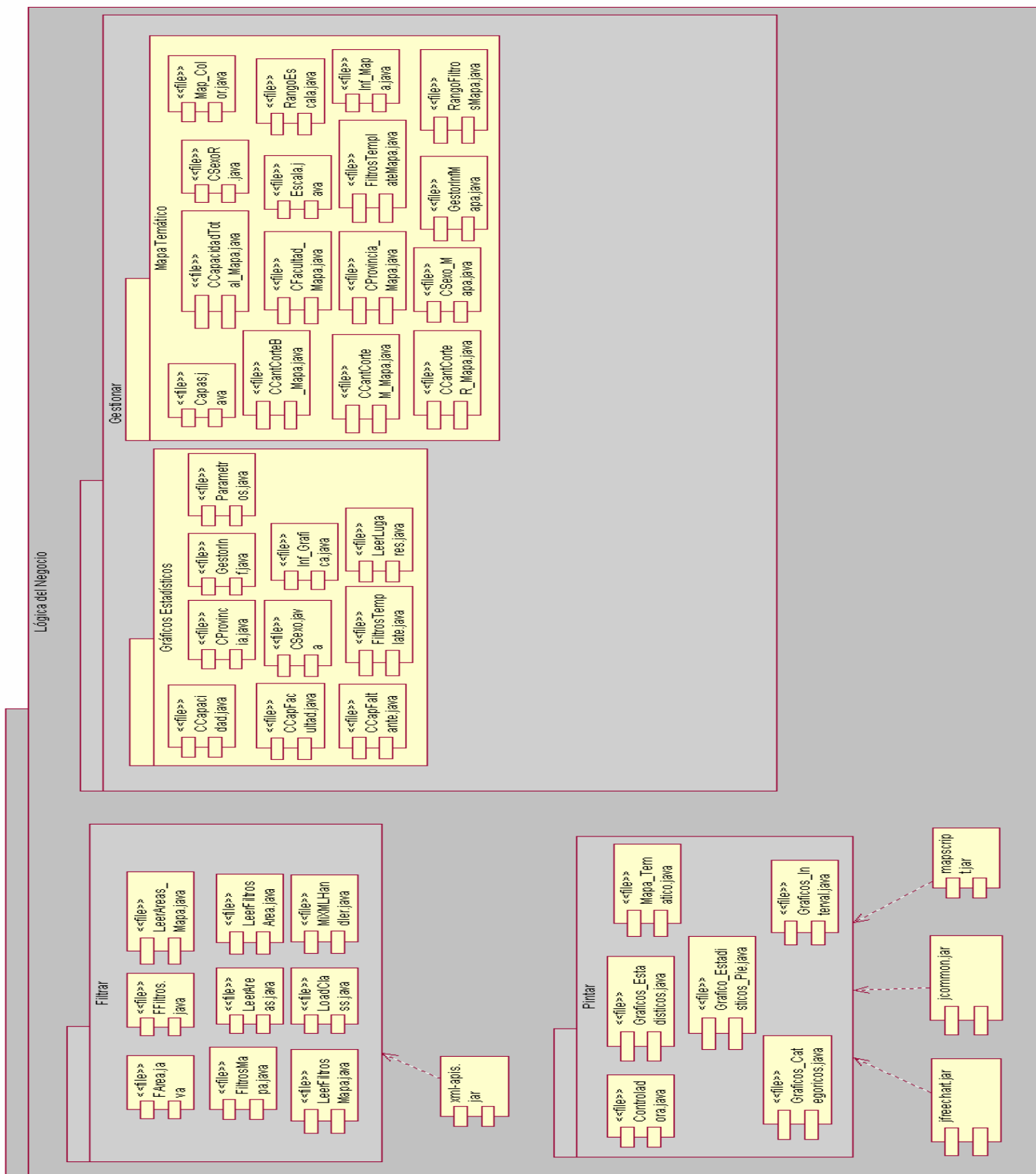


Figura 4. 20. Diagrama de Componentes de la Capa Negocio

4.8. Descripción preliminar del modelo de pruebas.

El modelo de pruebas permite principalmente, probar los componentes ejecutables del modelo de interacción. También se puede probar aspectos específicos como, la interfaz. Contiene una gran cantidad de casos de prueba, procedimientos de prueba y componentes de prueba. Entre los casos de prueba se puede distinguir dos tipos comúnmente utilizados: las llamadas pruebas de “caja negra” y las de “caja blanca”.

Las pruebas de caja negra verifican el resultado de la interacción entre los usuarios y el sistema, comprobando que se cumplan las precondiciones y poscondiciones especificadas para cada caso de uso siguiendo la secuencia de acciones previstas para el mismo.

Las pruebas de caja blanca son las encargadas de comprobar el comportamiento de las interacciones de los componentes internos del sistema.

En el caso del módulo que se entrega como culminación de esta investigación, se propone que se le realice una prueba de caja negra a cada caso de uso para que se pueda ver que los casos de usos cumplen con las precondiciones y poscondiciones especificadas para cada uno.

Además para comprobar el funcionamiento de los componentes interno se le pueden realizar pruebas de caja blanca.

4.9. Conclusiones

En este capítulo se ha llevado a cabo la descripción de las clases necesarias para la implementación del sistema, entre otros elementos. Se obtuvo el diagrama de clases de análisis, así como el de clase de diseño y los de clase Web. Se expusieron los principios seguidos para el diseño de la interfaz y la forma de tratar los errores. Se expuso la estructura de la aplicación, mediante los modelos de despliegue y de implementación. De esta forma cualquier desarrollador tendrá una idea de como esta implementado internamente este proyecto.

CAPÍTULO 5. ESTUDIO DE FACTIBILIDAD.

5.1. Introducción

En el presente capítulo se plantea un método para la estimación del esfuerzo en proyectos basados en Casos de Uso, utilizándose Análisis de Puntos de Casos de Uso, la cual es en cierta medida similar al Análisis de Puntos de Función.

Además se realiza un estudio de los beneficios tangibles e intangibles que brinda la aplicación y se hace un análisis beneficios donde se plantean las razones por las cuales queda justificado el tiempo que demoraría el desarrollo de este proyecto.

5.2. Estimación del esfuerzo basada en casos de uso

La especificación de los requerimientos mediante Casos e Uso ha probado ser uno de los métodos más efectivos para capturar la funcionalidad de un sistema. Este hecho se puede apreciar en algunas metodologías actuales ampliamente difundidas, como el Proceso Unificado de Racional.

Los Casos de Uso permiten especificar la funcionalidad de un sistema bajo análisis, pero no permiten por sí mismos efectuar una estimación del tamaño que tendrá el sistema o del esfuerzo que tomaría implementarlo. Para la estimación del tamaño de un sistema a partir de sus requerimientos, una de las técnicas más difundidas es el Análisis de Puntos de Función. Ésta técnica permite cuantificar el tamaño de un sistema en unidades independientes del lenguaje de programación, las metodologías, plataformas y/o tecnologías utilizadas, denominadas Puntos de Función.[19]

5.3. Puntos de Casos de Uso

La estimación mediante el análisis de Puntos de Casos de Uso es un método propuesto por muchos otros autores. Se trata de un método de estimación del tiempo de desarrollo de un proyecto mediante la asignación de "pesos" a un cierto número de factores que lo

afectan, para inicialmente, contabilizar el tiempo total estimado para el proyecto a partir de esos factores.

5.4. Cálculo de Puntos de Casos de Uso sin ajustar

El primer paso para la estimación consiste en el cálculo de los Puntos de Casos de Uso sin ajustar. Este valor, se calcula a partir de la siguiente ecuación:

$$UUCP = UAW + UUCW$$

Donde,

UUCP: Puntos de Casos de Uso sin ajustar

UAW: Factor de Peso de los Actores sin ajustar

UUCW: Factor de Peso de los Casos de Uso sin ajustar

5.4.1. Factor de Peso de los Actores sin ajustar (UAW)

Este valor se calcula mediante un análisis de la cantidad de Actores presentes en el sistema y la complejidad de cada uno de ellos. La complejidad de los Actores se establece teniendo en cuenta en primer lugar si se trata de una persona o de otro sistema, y en segundo lugar, la forma en la que el actor interactúa con el sistema. Los criterios se muestran en la tabla #5.6: donde se especifican los tipos de actor, la descripción de cada uno de ellos, los pesos correspondientes y la cantidad de con la que cuenta el módulo en cuestión.[19]

Tabla 5. 1. Factores de Peso de los Actores sin Justificar

Tipo de Actor	Descripción	Peso	Cant *Peso
Simple	Otro sistema que interactúa con el sistema a desarrollar mediante una interfaz de programación	1	3*1
Medio	Otro sistema que interactúa con el sistema a desarrollar mediante un protocolo o una interfaz basada en texto	2	0*2
Complejo	Una persona que interactúa con el sistema mediante una interfaz gráfica	3	1*3
Total			6

5.4.2. Factor de Peso de los Casos de Uso sin ajustar (UUCW)

Este valor se calcula mediante un análisis de la cantidad de Casos de Uso presentes en el sistema y la complejidad de cada uno de ellos. La complejidad de los Casos de Uso se establece teniendo en cuenta la cantidad de transacciones efectuadas en el mismo, donde una transacción se entiende como una secuencia de actividades atómica, es decir, se efectúa la secuencia de actividades completa, o no se efectúa ninguna de las actividades de la secuencia.[19] Los criterios se muestran en la siguiente tabla:

Tabla 5. 2. Factores de Peso de los Casos de Uso sin Justificar

Tipo de Actor	Descripción	Peso	$\Sigma(\text{Peso} * \text{Valor})$
Simple	El Caso de Uso contiene de 1 a 3 transacciones	5	0*5
Medio	El Caso de Uso contiene de 4 a 7 transacciones	10	0*10
Complejo	El Caso de Uso contiene más de 8 transacciones	15	2*15
Total			30

Teniendo los resultados anteriores se puede completar la ecuación anterior

$$\text{UUCP} = \text{UAW} + \text{UUCW}$$

$$\text{UUCP} = 6 + 30$$

$$\text{UUCP} = 36$$

5.5. Cálculo de Puntos de Casos de Uso ajustados

Una vez que se tienen los Puntos de Casos de Uso sin ajustar, se debe ajustar éste valor mediante la siguiente ecuación:

$$\text{UCP} = \text{UUCP} \times \text{TCF} \times \text{EF}$$

Donde:

UCP: Puntos de Casos de Uso ajustados

UUCP: Puntos de Casos de Uso sin ajustar

TCF: Factor de complejidad técnica

EF: Factor de ambiente

5.5.1. Factor de complejidad técnica (TCF)

El Análisis de Puntos de Función plantea el ajuste de los Puntos de Función calculados a partir de las Transacciones y Archivos, mediante la evaluación de 14 características generales del sistema. A cada una de estas características se le asigna un factor de peso (un valor entre 0 y 5) que indica la importancia de la característica para el sistema bajo análisis.[19] El significado del valor asignado a cada característica es el siguiente:

0: No presente o sin influencia

1: Influencia incidental

2: Influencia moderada

3: Influencia media

4: Influencia significativa

5: Fuerte influencia

Este coeficiente se calcula mediante la cuantificación de los factores que determinan la complejidad técnica del sistema. Tabla # 5.8

Tabla 5. 3. Factores de Complejidad

Factor	Descripción	Peso	Valor	Comentario	Σ (Peso * Valor)
T1	Sistema distribuído	2	0	El sistema es centralizado	2*0
T2	Objetivos de performance o tiempo de respuesta	1	3	La velocidad es limitada por las entradas provistas por el usuario	1*3
T3	Eficiencia del usuario final	1	3	Escasas restricciones de eficiencia	1*3
T4	Procesamiento interno complejo	1	4	No hay cálculos Complejos	1*4
T5	El código debe ser reutilizable	1	3	No se requiere que el código sea reutilizable	1*3
T6	Facilidad de instalación	0.5	2	Escasos requerimientos de facilidad de instalación	0.5*2
T7	Facilidad de uso	0.5	3	Normal	0.5*3

T8	Portabilidad	2	1	No se requiere que el sistema sea portable	2*1
T9	Facilidad de cambio	1	2	Se requiere un costo moderado de mantenimiento	1*2
T10	Concurrencia	1	1	No hay concurrencia	1*1
T11	Incluye objetivos especiales de seguridad	1	1	Seguridad normal	1*1
T12	Provee acceso directo a terceras partes	1	5	Los usuarios web tienen acceso directo	1*5
T13	Se requieren facilidades especiales de entrenamiento a los usuarios	1	2	Pocos usuarios internos, sistema fácil de usar.	1*2
Total					28.5

El Factor de complejidad técnica se calcula mediante la siguiente ecuación:

$$TCF = 0.6 + 0.01 \times \Sigma (\text{Peso} \times \text{Valor})$$

$$TCF = 0.6 + 0.01 \times \Sigma (\text{Peso} \times \text{Valor})$$

$$TCF = 0.6 + 0.01 \times 28.5$$

$$TCF = 0.6 + 0.285$$

$$TCF = 0.885$$

5.5.2. Factor de ambiente (EF)

Las habilidades y el entrenamiento del grupo involucrado en el desarrollo tienen un gran impacto en las estimaciones de tiempo. Estos factores son los que se contemplan en el cálculo del Factor de ambiente. El cálculo del mismo es similar al cálculo del Factor de complejidad técnica, es decir, se trata de un conjunto de factores que se cuantifican con valores de 0 a 5.[19] En la siguiente tabla se muestra el significado y el peso de cada uno de éstos factores.

Tabla 5. 4. Factores de Ambiente

Factor	Descripción	Peso	Valor	Comentario	$\Sigma(\text{Peso} \cdot \text{Valor})$
E1	Familiaridad con el modelo de proyecto utilizado	1.5	4	El grupo está bastante familiarizado con el modelo	1.5*4
E2	Experiencia en la aplicación	0.5	2	La mayoría del grupo ha trabajado mucho tiempo en ésta aplicación	0.5*2
E3	Experiencia en orientación a objetos	1	3	La mayoría del grupo programa en Objetos	1*5
E4	Capacidad del analista líder	0.5	4	Se contrató a un Especialista	0.5*4
E5	Motivación	1	5	El grupo está altamente motivado	1*5
E6	Estabilidad de los requerimientos	2	2	Se esperan cambios	2*2
E7	Personal part-time	-1	0	Todo el grupo es full-time	-1*0
E8	Dificultad del lenguaje de programación	-1	3	Se usará lenguaje java	-1*3
Total					18

- Para los factores E1 al E4, un valor asignado de 0 significa sin experiencia, 3 experiencia media y 5 amplia experiencia (experto).
- Para el factor E5, 0 significa sin motivación para el proyecto, 3 motivación media y 5 alta motivación.

- Para el factor E6, 0 significa requerimientos extremadamente inestables, 3 estabilidad media y 5 requerimientos estables sin posibilidad de cambios.
- Para el factor E7, 0 significa que no hay personal part-time (es decir todos son full-time), 3 significa mitad y mitad, y 5 significa que todo el personal es part-time (nadie es full-time).
- Para el factor E8, 0 significa que el lenguaje de programación es fácil de usar, 3 medio y 5 que el lenguaje es extremadamente difícil.

El Factor de ambiente se calcula mediante la siguiente ecuación:

$$EF = 1.4 - 0.03 \times \Sigma (\text{Peso} \times \text{Valor})$$

$$EF = 1.4 - 0.03 \times \Sigma (\text{Peso} \times \text{Valor})$$

$$EF = 1.4 - 0.03 \times 18$$

$$EF = 1.4 - 0.54$$

$$EF = 0.86$$

Finalmente, los Puntos de Casos de Uso ajustados resultan:

$$UCP = UUCP \times TCF \times EF$$

$$UCP = 36 \times 0.885 \times 0.86$$

$$UCP = 27.39$$

5.6. De los Puntos de Casos de Uso a la estimación del esfuerzo

El esfuerzo en horas-hombre viene dado por:

$$E = UCP \times CF$$

Donde:

E: esfuerzo estimado en horas-hombre

UCP: Puntos de Casos de Uso ajustados

CF: factor de conversión

Se debe tener en cuenta que éste método proporciona una estimación del esfuerzo en

horas-hombre contemplando sólo el desarrollo de la funcionalidad especificada en los casos de uso. Finalmente, para una estimación más completa de la duración total del proyecto, hay que agregar a la estimación del esfuerzo obtenida por los Puntos de Casos de Uso, las estimaciones de esfuerzo de las demás actividades relacionadas con el desarrollo de software.

Para ello se puede tener en cuenta el siguiente criterio, que estadísticamente se considera aceptable. El criterio plantea la distribución del esfuerzo entre las diferentes actividades de un proyecto, según la siguiente aproximación: Tabla 5.10

Tabla 5. 5. Distribución del Esfuerzo

Actividad	% esfuerzo
Análisis	10%
Diseño	20%
Implementación	40%
Prueba	15%
Sobrecarga	15%
Total	100%

Obviamente, estos valores no son absolutos sino que pueden variar de acuerdo a las características de la organización y del proyecto. Con éste criterio, y tomando como entrada la estimación de tiempo calculada a partir de los Puntos de Casos de Uso, se pueden calcular las demás estimaciones para obtener la duración total del proyecto.

5.6.1. Factor de conversión (CF)

Según los siguientes criterios se procederá a calcular el esfuerzo total del proyecto:

- Se contabilizan cuántos factores de los que afectan al Factor de ambiente están por debajo del valor medio (3), para los factores E1 a E6.
- Se contabilizan cuántos factores de los que afectan al Factor de ambiente están por encima del valor medio (3), para los factores E7 y E8.
- Si el total es 2 o menos, se utiliza el factor de conversión 20 horas-hombre/Punto de Casos de Uso, es decir, un Punto de Caso de Uso toma 20 horas-hombre.
- Si el total es 3 ó 4, se utiliza el factor de conversión 28 horas-hombre/Punto de Casos de Uso, es decir, un Punto de Caso de Uso toma 28 horas-hombre.

- Si el total es mayor o igual que 5, se recomienda efectuar cambios en el proyecto, ya que se considera que el riesgo de fracaso del mismo es demasiado alto.

Cantidad de EF < 3 entre E1 – E6 para el módulo en cuestión es 3 y la cantidad de EF >3 de los factores E7 y E8 no existen

Por tanto:

Total EF = 3

Teniendo en cuenta los criterios anteriores:

CF= 28horas – hombres

Luego,

$$E = UCP \times CF$$

$E = UCP \times CF$

$E = 27.39 \times 28\text{horas} - \text{hombres}$

$E = 767.188$

Si además se considera que este esfuerzo representa un porcentaje del esfuerzo total del proyecto, de acuerdo a los valores porcentuales de la tabla anterior, se obtiene:

Tabla 5. 6. Valor del Esfuerzo

Actividad	% esfuerzo	Valor esfuerzo
Análisis	10%	192 horas-hombre
Diseño	20%	384 horas-hombre
Implementación	40%	767 horas-hombre
Prueba	15%	288 horas-hombre
Sobrecarga	15%	288 horas-hombre
Total	100%	1918 horas-hombre

Como el valor de esfuerzo calculado representa el esfuerzo del FT implementación, por comparación salen el resto de los esfuerzo y la suma de ellos es el esfuerzo total (ET).

Suponiendo que una persona trabaje 8 horas por día, y un mes tiene como promedio 24 días laborables; la cantidad de horas que puede trabajar una persona en 1 mes es 192 horas.

Si ET = 1918 horas-hombre y por cada 192 horas se tiene 1 mes eso daría un

ET de aproximadamente 10 mes-hombre.

Lo que significa que 1 persona puede realizar el problema analizado en más o menos 10 meses de trabajo.

5.7. Beneficios tangibles e intangibles

La aplicación Módulo Cartográfico resultaría de gran importancia en cualquier contexto en la que fuera implantada, bastaría con que las empresas desearan obtener en forma de gráficos estadísticos o mapas temáticos cualquier información almacenada en bases de datos.

Aunque esta aplicación no fue elaborada con un fin comercializable, pudiera resultar muy costosa por las funcionalidades que brinda.

El principal objetivo de la aplicación es ayudar a manipular grandes informaciones a la hora de tomar decisiones, por lo que sus beneficios inmediatos son mayormente intangibles.

- Aumentara la calidad y percepción de cualquier reporte.
- Da la posibilidad de hacer un estudio exhaustivo por zonas geográficas de forma sencilla y rápida.
- Visualizar la información requerida en gráficos de diferentes tipos
- Representa los resultados en forma de mapas temáticos de coropletas, donde los colores del mismo permitirán conocer la distribución de los datos en un mapa.

5.8. Análisis de costos y beneficios

El desarrollo de la aplicación Módulo Cartográfico no requiere de grandes gastos de recursos, ni tampoco de tiempo. El personal forma parte de la institución por lo que no se le tendría que hacer pagos adicionales. La información contenida en las bases de datos de la institución, puede ser alojada en los servidores de esta, gracias a su capacidad y rapidez en los servicios.

La tecnología utilizada para el desarrollo del módulo es java, la cual juega un papel protagónico en estos tiempos en los que la migración hacia el software libre se convierte en un hecho.

El sistema puede ser extendido para uso general, obteniéndose un producto comercializable que puede ser fuente de ingresos y se considera que es factible el desarrollo de la aplicación, por lo que el esfuerzo de desarrollo (10 Personas-Mes) está plenamente justificado.

5.9. Conclusiones

En este capítulo se ha aplicado el método de estimación por Puntos de Caso de Uso. Este método resulta muy efectivo para estimar el esfuerzo requerido en el desarrollo de los primeros Casos de Uso de un sistema, si se sigue una aproximación iterativa como el Proceso Unificado de Rational. Aunque en muchas ocasiones el método tiende a sobredimensionar el esfuerzo requerido, cabe aclarar que el uso de otras herramientas como COCOMO II no resultaría tan efectivo para estimar esfuerzos en proyectos como este. Además se realizó un estudio de los beneficios tangible e intangible, así como un análisis de costos y beneficios. El resultado de estas estimación justifica la el tiempo de desarrollo.

CONCLUSIONES GENERALES

En este trabajo se demostró la necesidad de diseñar e implementar un módulo que al incluirlo dentro del Sistema de Información Geográfico permita la posibilidad de mostrarle al usuario información concreta de determinado espacio. Estos resultados se mostrarán en forma de gráficos estadísticos y mapas temáticos. La utilización de estos aumentará la calidad y percepción de cualquier reporte, dando la posibilidad de hacer un estudio exhaustivo por zonas geográficas de forma sencilla y rápida. Esta aplicación pone en manos de los usuarios del GIS herramientas necesarias para la toma de determinadas decisiones.

Después de un análisis de las tecnologías más usadas en la actualidad para la construcción de sistemas informáticos, se llegó a la conclusión de utilizar la tecnología JSP, que nos permite utilizar como lenguaje del lado del servidor java, uno de los más poderosos en este sentido, así como otras tecnologías como el formato XML. Y para lograr una mejor interactividad con el usuario en el navegador se utilizó como lenguaje del lado del cliente JavaScript.

También se consideró que la metodología idónea para llevar a cabo el proceso de desarrollo es RUP (Proceso Unificado de Rational).

Se modeló el negocio propuesto, se definieron los requerimientos del sistema, tanto funcionales como no funcionales, se identificaron los actores del sistema, así como los casos de uso que deben de implementarse para que el módulo cumpla con los objetivos propuestos y se especificaron para un mejor entendimiento de los mismos.

Se diseñó el sistema, a través de diagramas de clases análisis, el diagrama de clases de diseño, entre otros. Posteriormente se elaboró el modelo de despliegue y el de implementación.

Se plantearon los principios a seguir en el diseño de la interfaz de usuario y algunas convenciones a respetar durante la escritura del código fuente.

Finalmente, se analizaron los beneficios y costos que genera el sistema. Luego de todo este proceso de trabajo se puede concluir que el módulo que se presenta da solución a la situación problemática que lo originó y que su explotación significará una mejora considerable en la calidad y eficiencia de la toma de decisiones en la institución.

RECOMENDACIONES

Se recomienda que se trate de hacer una clase controladora para cada base de datos de La Universidad, que se pueda acceder a ella desde cualquier aplicación en cualquier plataforma de esta manera se podrá utilizar información que ya esta digitalizada, y así hacer las aplicaciones más funcionales y dinámicas.

También se recomienda actualizar el mapa de la UCI, de esta forma se podrán hacer más Mapas Temáticos y con mejor calidad.

Que se integre lo más rápido posible este módulo al GIS para brindar en el caso de los mapas temáticos las mismas posibilidades que a un mapa normal.

Además que la dirección de la universidad defina que información le seria útil a la hora de la toma de decisiones filtrar, para implementarlos para próximas iteraciones.

GLOSARIO DE TÉRMINOS

- **Gráficos:** Gráficos, en informática, es el nombre dado a cualquier imagen generada por un ordenador.
- **Estadística:** Por estadística entendemos una batería de recursos científicos por los cuales podemos recolectar, organizar, resumir, presentar y analizar datos numéricos de un conjunto de observaciones.
- **Frecuencia absoluta:** Es el número de observaciones que la variable toma en determinada clase o categoría.
- **Frecuencia relativa:** Es la proporción del valor de la frecuencia absoluta de cada clase en relación al total de las observaciones de la variable.
- **Percentiles:** Los percentiles son valores que resultan de dividir la población (el N de las observaciones) en cien partes iguales
- **Cuartil:** La mediana separa en dos mitades el conjunto de observaciones. Podemos aún dividir cada mitad en dos de tal manera que resulten cuatro partes iguales. Cada una de esas divisiones se conoce como Cuartel.
- **Cuantiles:** la mediana divide a los datos en dos partes iguales. Los cuantiles son aquellos que dividen los datos de la distribución en función de otras cantidades.
- **Varianza:** es un estimador de la divergencia de una variable aleatoria de su valor esperado
- **Modelizarse:** Calcular la moda de un conjunto de observaciones.
- **Mapa:** Es la representación gráfica de una porción del espacio geográfico, proyectada sobre una superficie plana
- **Información ráster:** Información que se basa en tratar las imágenes como líneas y polígonos.
- **Hipermedia:** Páginas Web que integran información en distintos tipos de formatos: texto, gráficos, sonidos y vídeo.
- **Georeferenciada:** Es la forma de referenciar una área geográficamente.

REFERENCIAS BIBLIOGRÁFICAS

- [1]. Definición y algunas aplicaciones de los sistemas de información geográfica. Monografía.com(Octubre 2005)
<http://www.monografias.com/trabajos14/informageogra/informageogra.shtml>
- [2]. Molinero, Luís. Gráficos de Datos Estadísticos en Medicina. (Octubre 2005)
<http://www.seh-lha.org/graficos.htm>.
- [3]. Broader, Terms. Representación Gráfica. (Octubre 2005)
<http://www3.unileon.es/dp/abd/tesauro/pagina/tesdocumentacion/00000135.htm>
- [4]. Gráficos. (Noviembre 2005)
<http://www.bioestadistica.uma.es/libro/node10.htm>
- [5]. Nociones Básicas.
<http://nivel.euito.upm.es/~mab/htmls/intro.html#m.tematico>(Noviembre 2005)
- [6]. Taller de Biblioteca. (Noviembre 2005)
<http://www.geocities.com/crachilecl/atlas.pdf>
- [7]. Ortiz, Gabriel. Información Geográfica. (Diciembre 2005)
<http://recursos.gabrielortiz.com/index.asp>
- [8]. Internet. wikipedia.org (Diciembre 2005)
<http://es.wikipedia.org/wiki/Internet>
- [9]. Biblioteca de Rutina. (Diciembre 2005)
- [10]. Perl. <http://geneura.ugr.es/~jmerelo/tutoperl/>(Diciembre 2005)
- [11]. Tutoriales en la Red(Diciembre 2005)
<http://www.tutorialesenlared.com/manual926.html>
- [12]. PHP. WebEstilo(Diciembre 2005)
<http://www.webestilo.com/php/>
- [13]. JSP. wikipedia.org(Diciembre 2005)
<http://es.wikipedia.org/wiki/JSP>
- [14]. Servlet. wikipedia.org(Enero 2006)
<http://es.wikipedia.org/wiki/Servlet>
- [15]. Sistemas de Bases de datos: (Enero 2006)
<http://usuarios.lycos.es/cursosgbd/UD2.htm>
- [16]. RUP. . wikipedia.org (Enero 2006)
<http://es.wikipedia.org/wiki/RUP>
- [17]. Java en Castellano. (Enero 2006)
<http://www.programacion.com/java/tutorial/patrones/>
- [18]. Larman, Craig. UML y Patrones. Introducción al análisis y diseño orientado a objetos. Editorial Félix Varela. La Habana, 2004. (Enero 2006)

- [19].** Peralta, Mario. Estimación del esfuerzo basada en casos de usos. Centro de Ingeniería del Software e Ingeniería del Conocimiento (CAPIS),

ANEXOS



Figura 1.1. Gráfico de Sectores

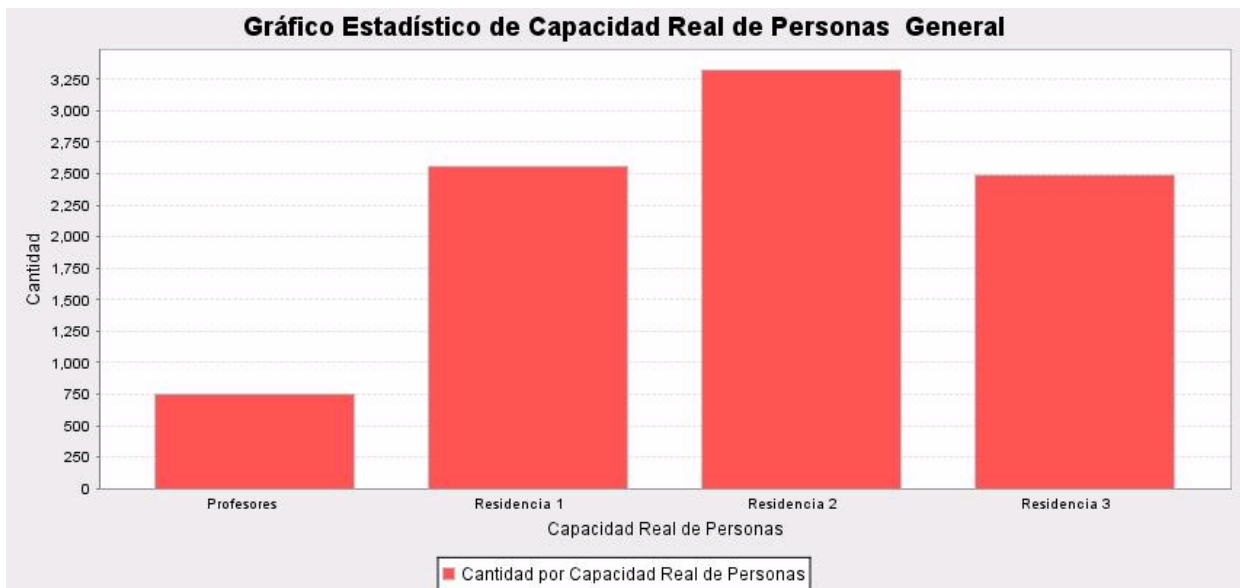


Figura 1.2. Gráfico de Barra

Tabla 1.1. Distribución de Frecuencia de la Edad de 100 Pacientes

Edad	Nº de pacientes
18	1
19	3
20	4
21	7
22	5
23	8
24	10
25	8
26	9
27	6
28	6
29	4
30	3
31	4
32	5
33	3
34	2
35	3
36	1
37	2
38	3
39	1
41	1
42	1

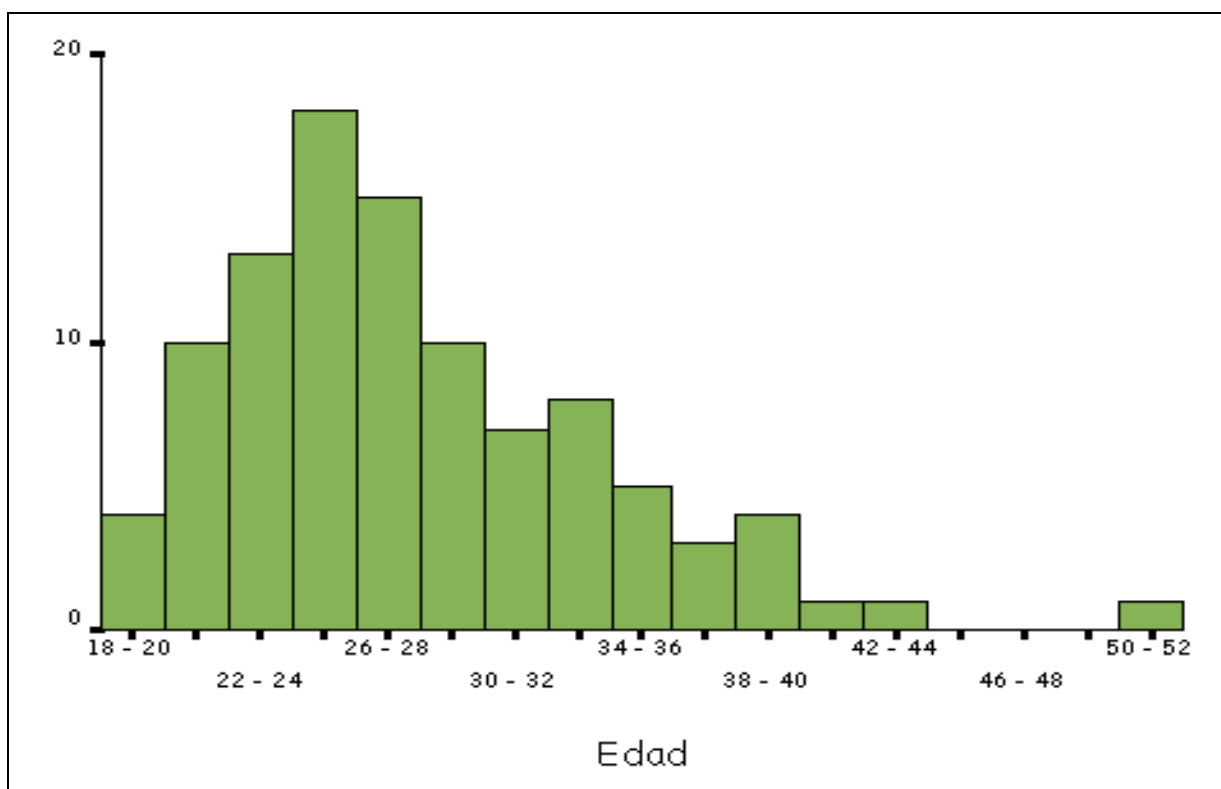


Figura 1.3. Histograma Correspondiente a los Datos de la Tabla 1.

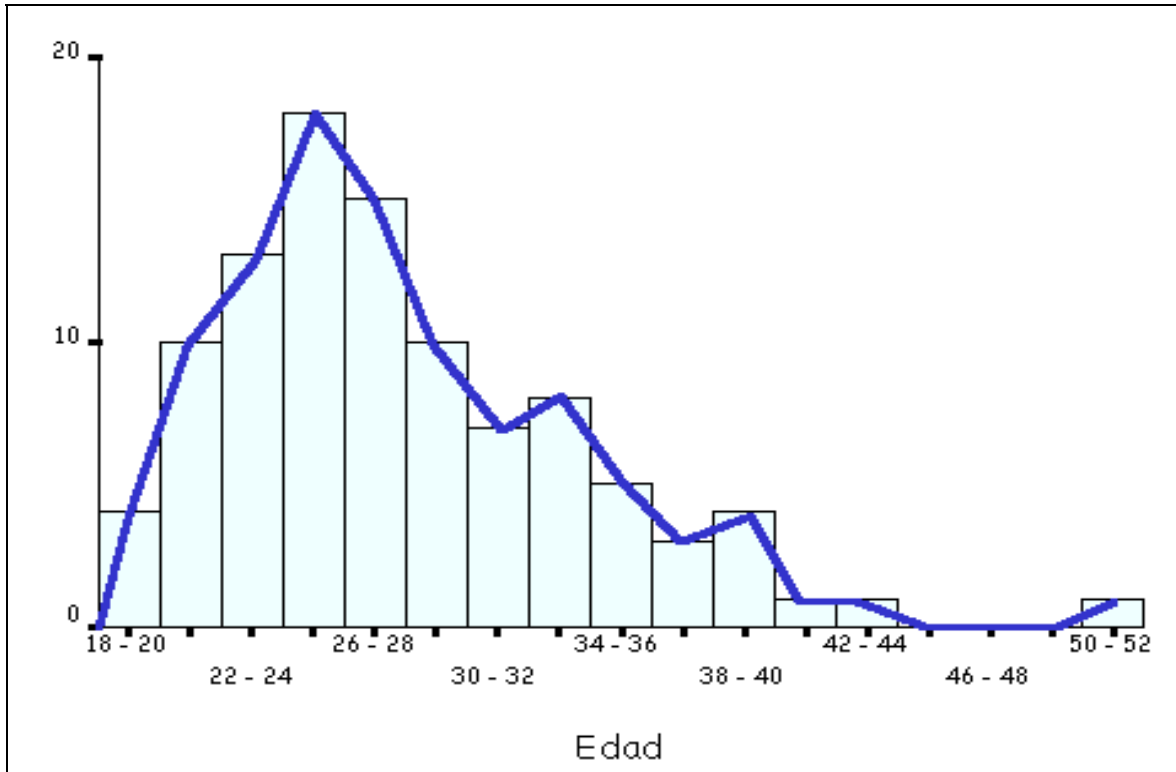


Figura 1.4. Polígono de Frecuencia Correspondiente a los Datos de la Tabla 1.

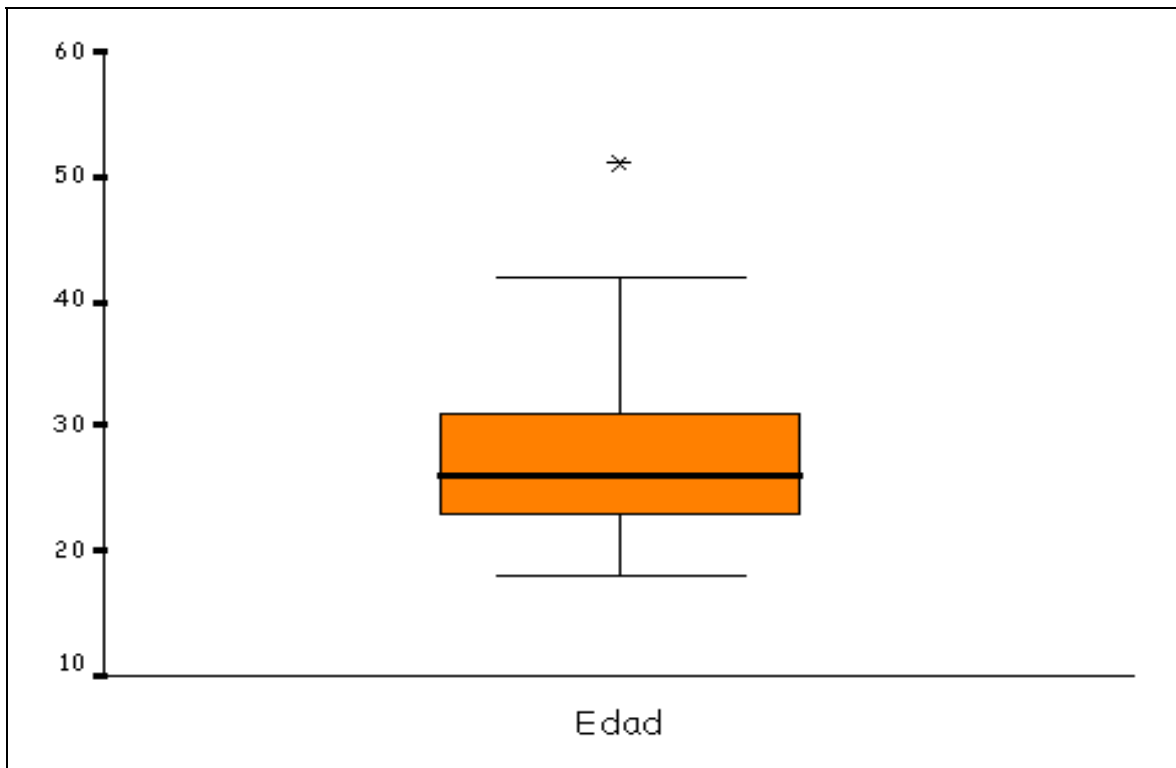


Figura 1.5. Diagrama de Caja Correspondiente a los Datos de la Tabla 1.

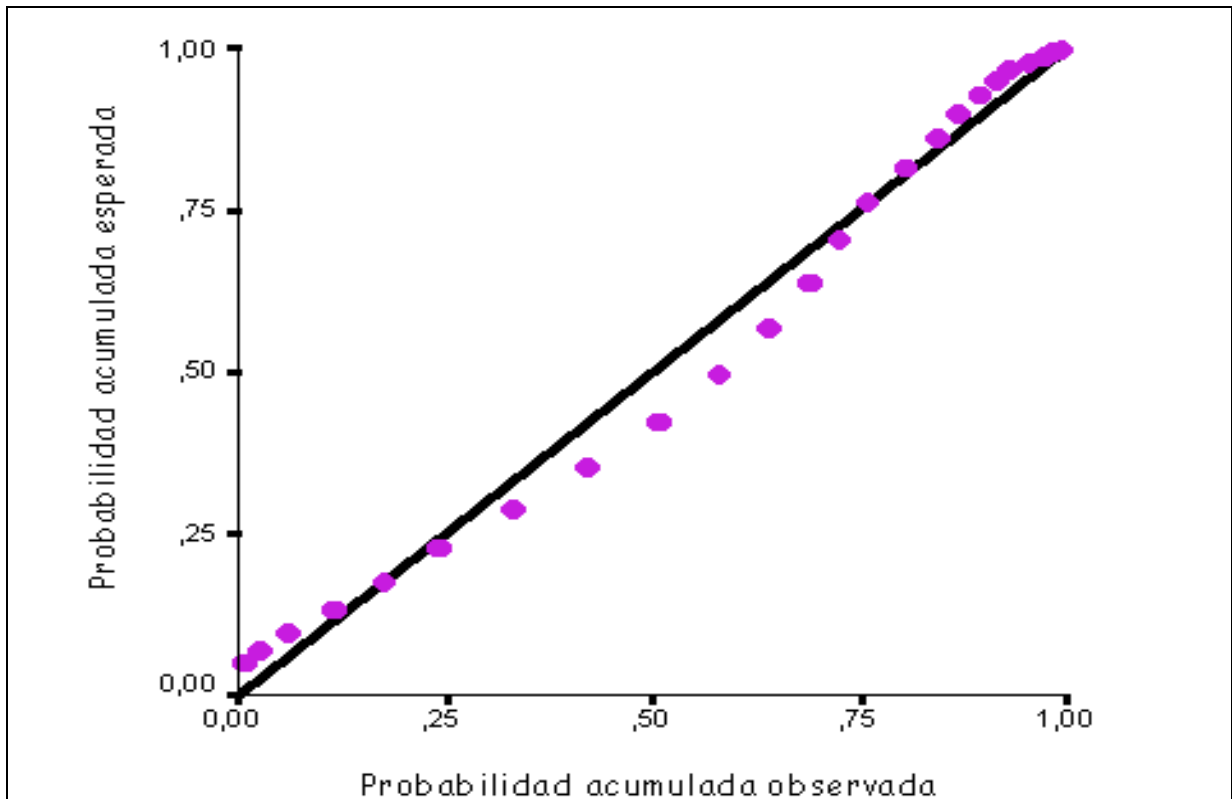


Figura 1.6. Gráfico P-P Correspondiente a los Datos de la Tabla 1.12

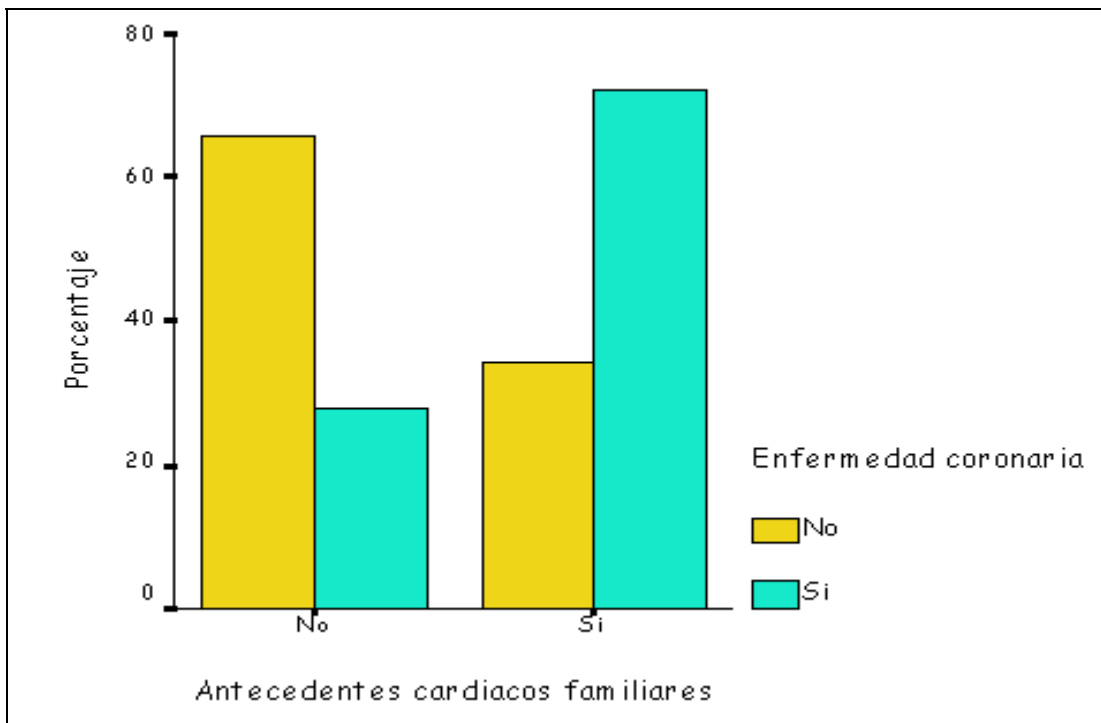


Figura 1.7. Diagrama de Barras Dobles

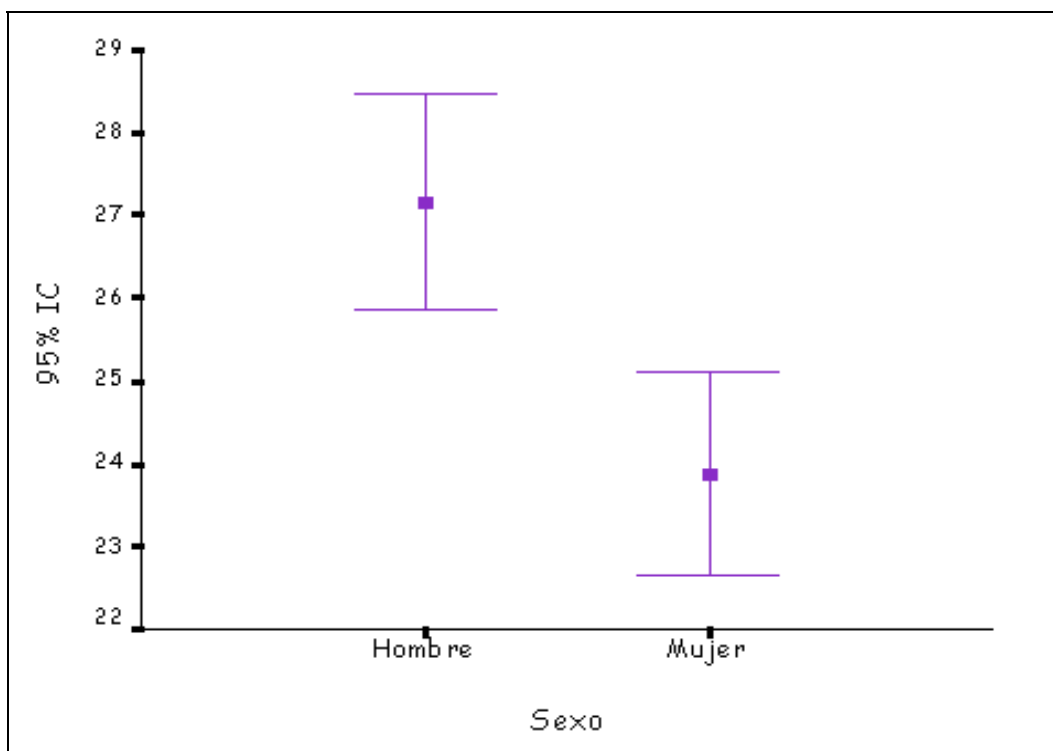


Figura 1.8. Barra del Error

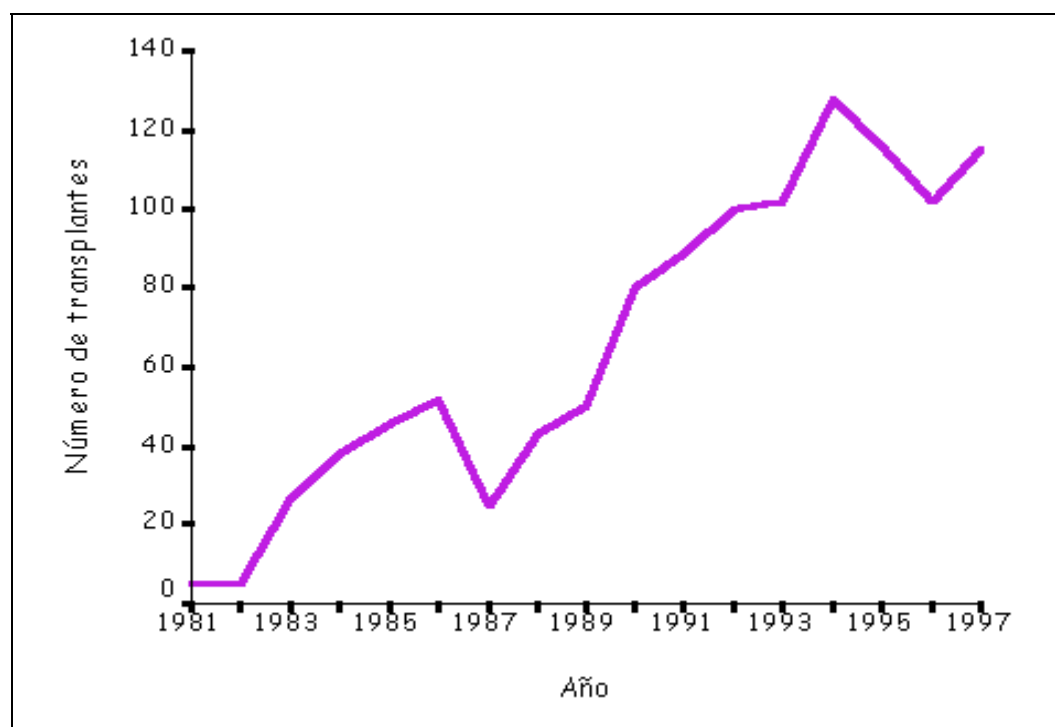


Figura 1.9. Gráfico de Líneas

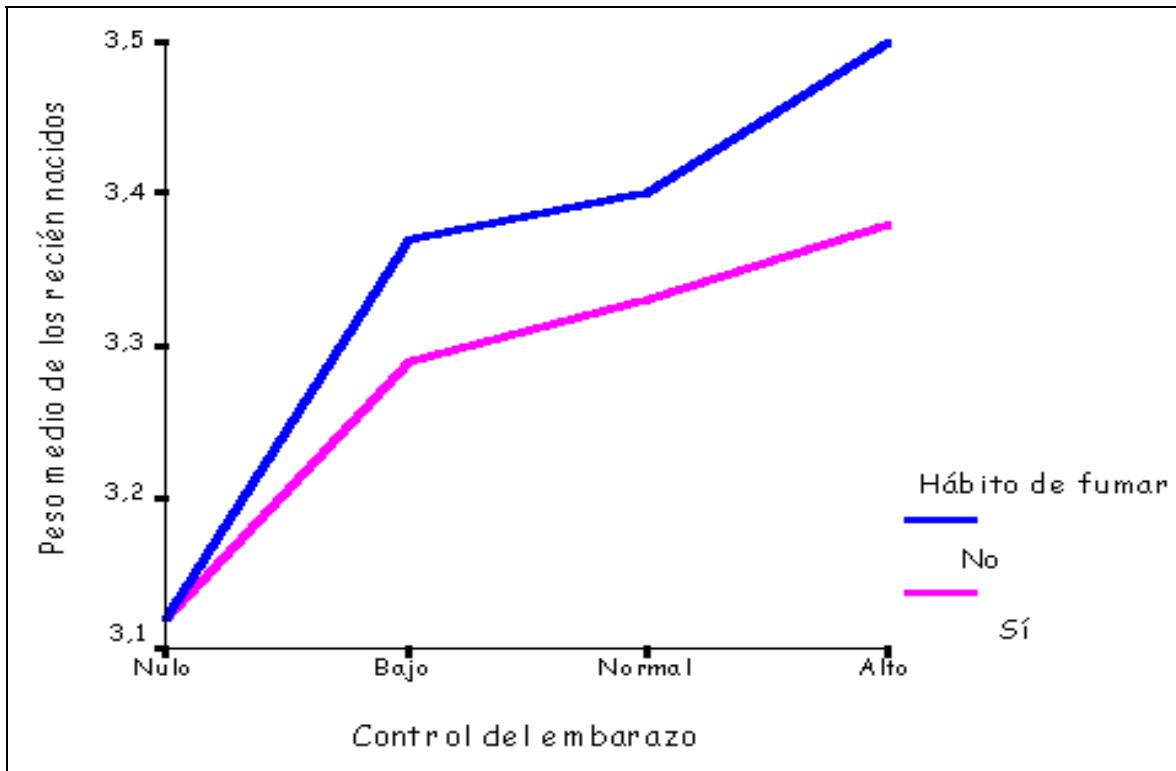


Figura 1.10. Gráfico de Líneas Superpuestas

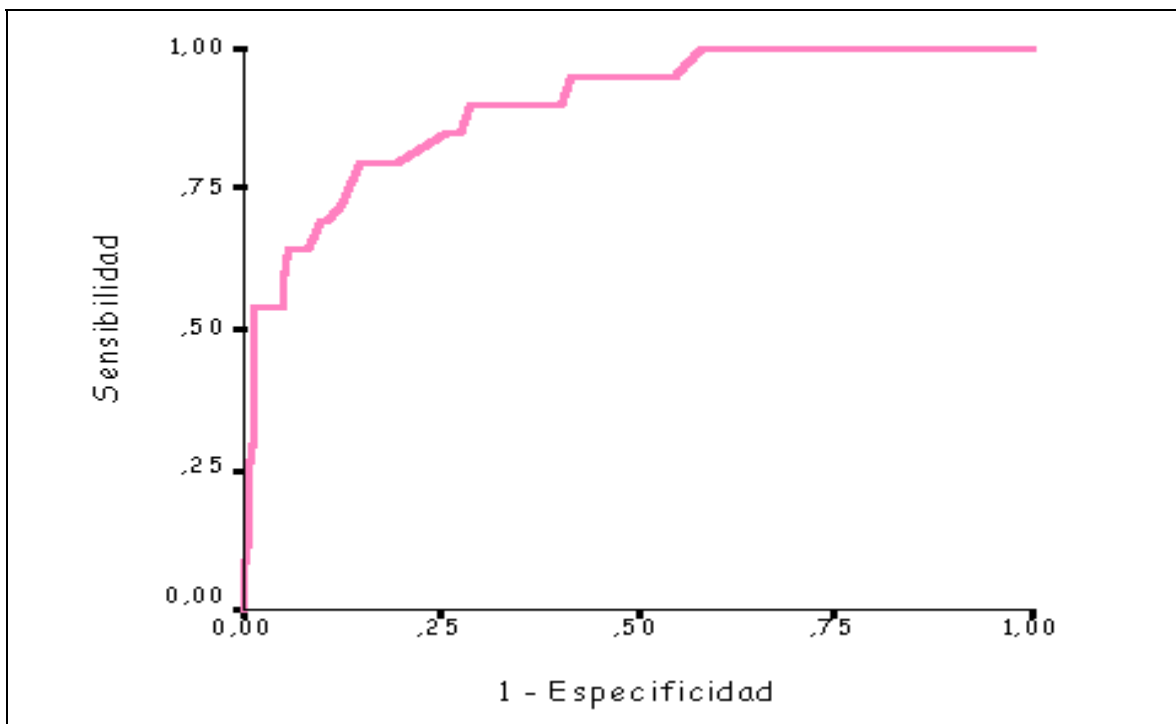


Figura 1.11. Curvas ROC

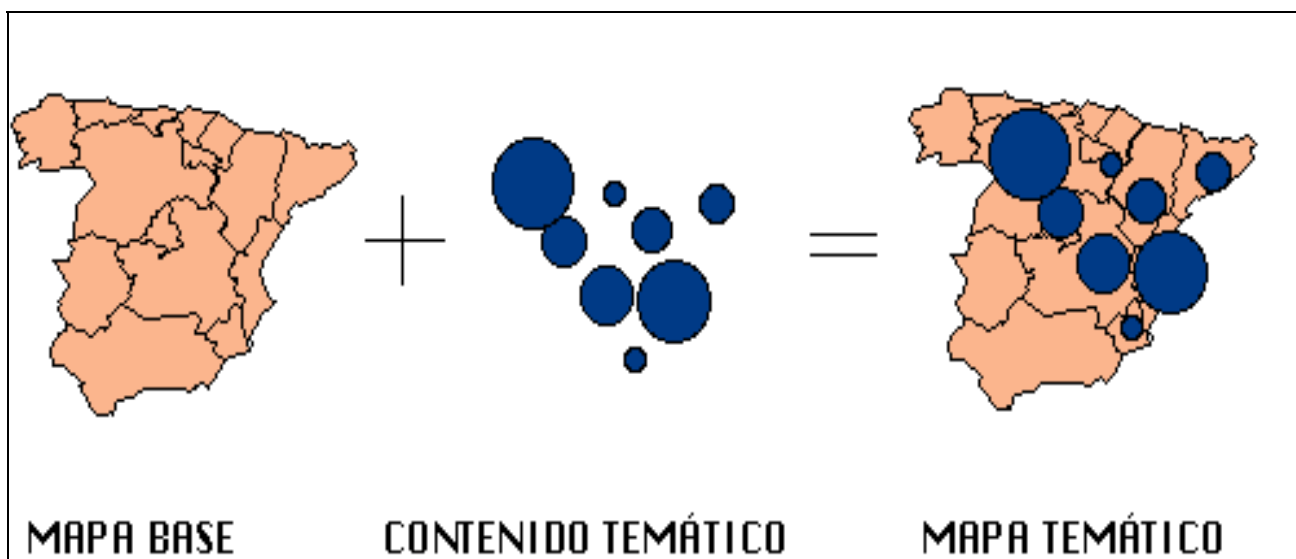


Figura 1.12. Componentes de un Mapa Temático

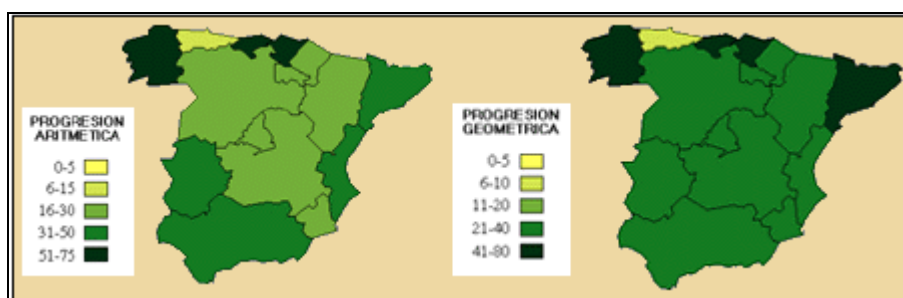


Figura 1.13. Mapa de Coropletas

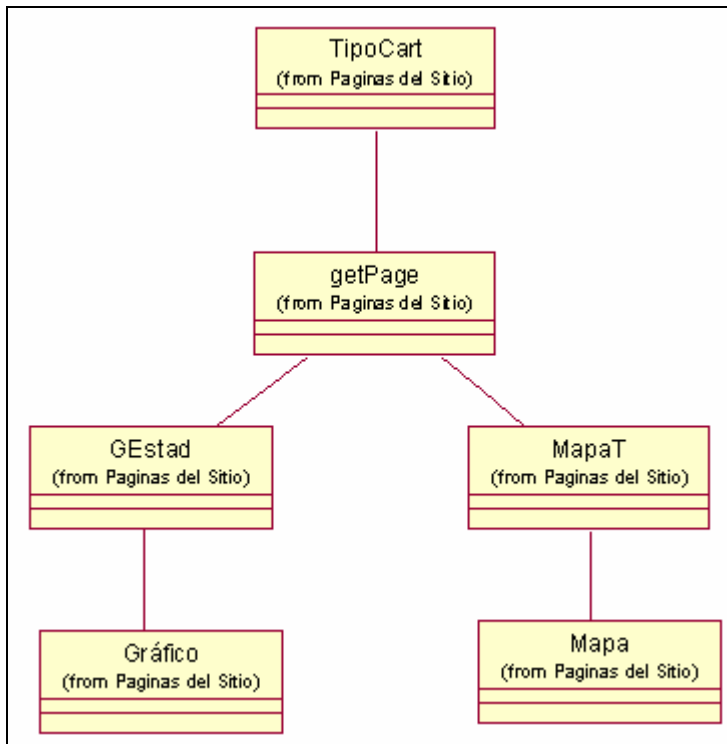


Figura 4. 21. Mapa de Navegación del Sitio

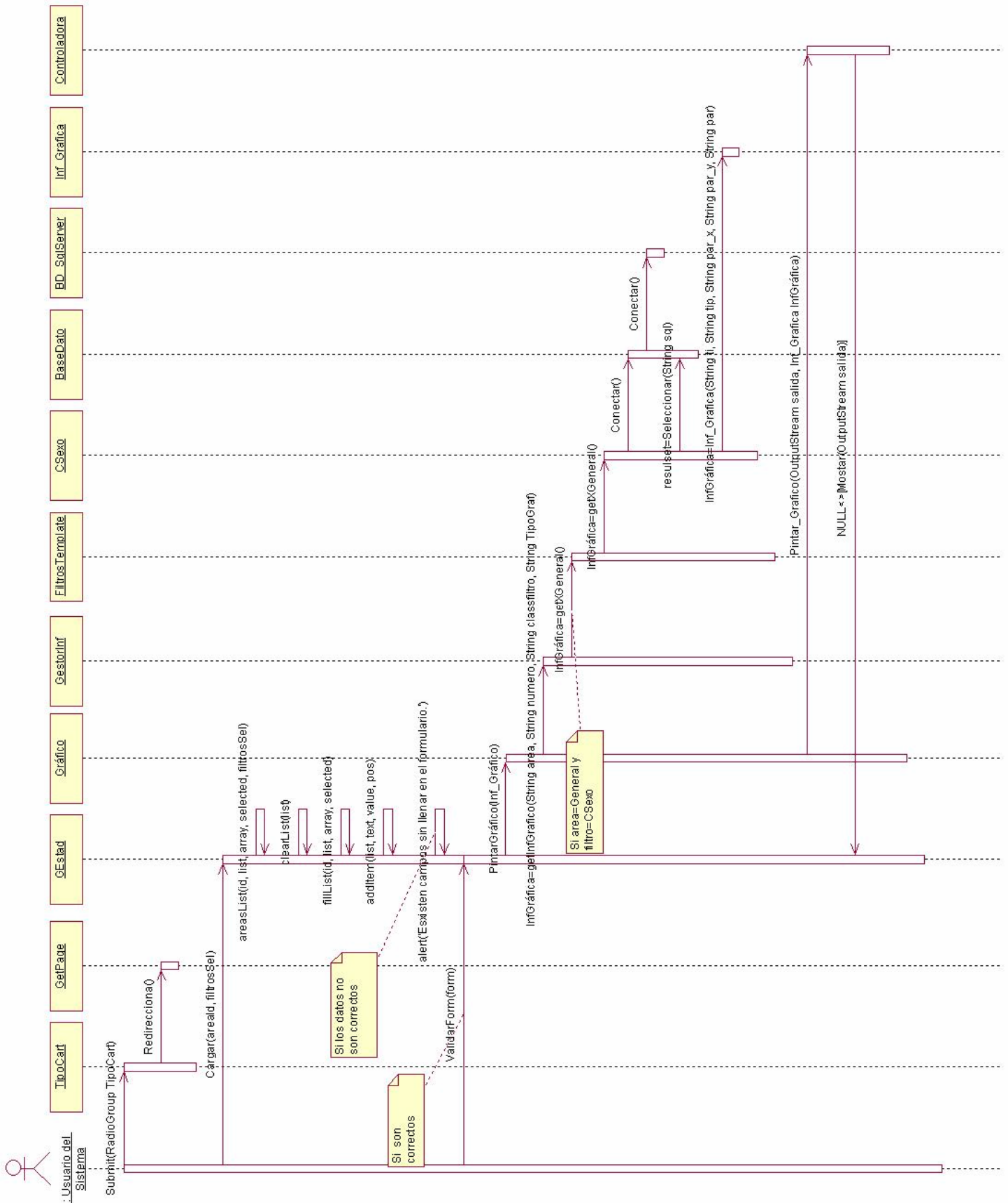


Figura 4. 22. Diagrama de Secuencia CU: Gráfico Estadístico (General X Sexo)

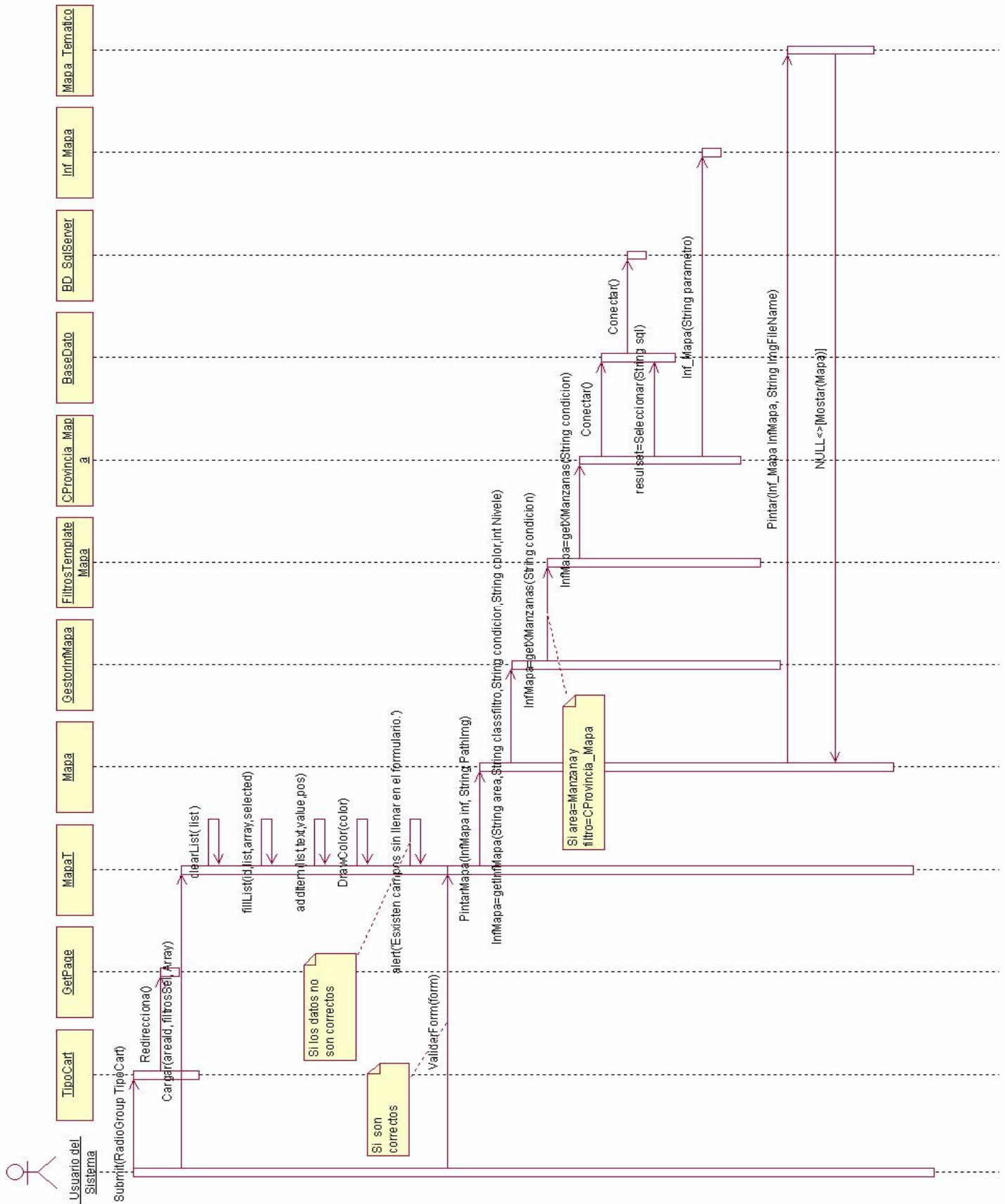


Figura 4. 23. Diagrama de Secuencia CU: Mapas Temáticos (Manzana X Provincias)

Nota: Los demás diagramas de interacción siguen la misma secuencia de acciones lo único que cambia en ellos son los filtros seleccionados por los usuarios.