

**Universidad de las Ciencias Informáticas
Facultad 4**



**Título: SISTEMA GENERADOR DE MAPAS TEMÁTICOS Y
GRÁFICOS ESTADÍSTICOS**

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

Autores:

Maglema Ramona Palmero Sánchez
Yenier Vázquez Baños

Tutor:

Lic. Rafael Rodríguez Puente

Ciudad de la Habana, julio de 2007
"Año 49 de la Revolución"

DECLARACIÓN DE AUTORÍA

Declaramos ser autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

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

Maglema R. Palmero Sánchez.

Autor.

Yenier Vázquez Baños.

Autor.

Lic. Rafael Rodríguez Puente.

Tutor

DATOS DE CONTACTO

Lic. Rafael Rodríguez Puente.

Profesor de la Universidad de Ciencias Informáticas.

Correo electrónico: rafaelrp@uci.cu

AGRADECIMIENTOS COMPARTIDOS.

Para atravesar estos 5 años de estudio y finalmente elaborar este documento se ha necesitado de la ayuda de muchas personas, a las que agradecerles de todo corazón, por su tiempo y dedicación, nunca sería suficiente:

A **la Revolución, a la UCI, a Nuestro Comandante en Jefe**, por brindarnos la oportunidad de ser mejores y por confiarnos el futuro.

A **nuestro tutor Rafael** por su dirección, por sus ideas y los valiosos consejos que nos permitieron desarrollar este trabajo. Por su paciencia y dedicación.

A **los profesores** que nos acompañaron en estos 5 años, por compartir sus conocimientos, consejos y experiencia.

AGRADECIMIENTOS

De Maglema:

A mi **Madre, Virginia**, por ser la mejor del mundo, por permitirme todas las malcriadeces y apoyarme en cualquier decisión que yo tomase. Por dejarme aprender de los errores.

A **mi familia** porque, más que mío, el sueño fue suyo.

A **mi suegra y a mi novio** por su paciencia, por el amor que siempre tienen para ofrecer.

A mis amigas: **Haydee, Lisy, Mayté, Dinia, Dalkis, Yaislen, Nadia y Yari**, gracias por estar siempre, por compartir su amistad, por no dejarme sola, por apurarme. Gracias por los consejos y el cariño que nunca me faltó.

De Yenier:

A **mis padres**, que me dieron la vida, me educaron, nunca me dieron la espalda en cada momento difícil 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.

A **mi novia Yanisbei** 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 Jorge y Odalis** que siempre me apoyaron en todo.

A mi familia.

"Cuando se es joven, se crea. Cuando se es inteligente, se produce. No se adapta, se innova: la medianía copia; la originalidad se atreve."

José Martí

DEDICATORIA

Maglema:

A Ma, por su apoyo incondicional como madre y amiga que siempre ha sido.

A mis abuelos y a Keynier, porque los adoro.

A Arianna, porque la extraño.

A mi novio, Leonardito, porque nunca me dejó sola.

A Sebastián, porque un día crecerá...

Yenier:

A mis padres que me inspiran.

A mi novia que siempre estuvo ahí.

RESUMEN

Este trabajo de diploma se propone realizar el diseño de un sistema para proveer al Sistema de Información Geográfica (SIG) de la 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 realizar el diseño y la implementación de un módulo para el Sistema de Información Geográfica existente en la UCI que permita de una forma cómoda, facilitar a los usuarios la información real de un área determinada a través de gráficos estadísticos y mapas temáticos que transformaría al SIG de un sistema de consulta geográfica a una poderosa herramienta de consulta geoestadística que automatice la toma de decisiones en las diferentes áreas existentes en la UCI.

Este documento contiene los resultados de todo el trabajo investigativo realizado para realizar el módulo en cuestión. Se identifican y describen los procesos de las áreas de trabajo de la UCI que tienen como base la toma de decisiones, especialmente aquellos que se van a automatizar. 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 realiza la implementación de dicho sistema.

Palabras Claves:

Sistema de Información Geográfica, SIG, Gráficos Estadísticos, Mapas Temáticos, información geográfica, GIS.

ÍNDICE

AGRADECIMIENTOS COMPARTIDOS	I
AGRADECIMIENTOS	II
DEDICATORIA	IV
RESUMEN	V
INTRODUCCIÓN	1
CAPÍTULO 01: FUNDAMENTACIÓN TEÓRICA.....	4
1.1 Introducción.....	4
1.2 Objeto de estudio	4
1.3 Situación problemática. Flujo actual de los procesos	4
1.4 Propuesta de solución.....	5
1.5 Fundamentación de los objetivos propuestos.....	6
1.6 Objetivo general.....	6
1.7 Objetivos específicos.....	6
1.8 Los Gráficos estadísticos. Su representación gráfica.....	6
1.9 Presentación de datos: tipos de gráficos.....	7
1.10 Análisis descriptivo de la información.....	8
1.11 Los mapas, combinaciones de mapas: atlas.....	9
1.12 Los mapas temáticos.....	10
1.13 Tipos de cartografías temáticas.....	10
1.14 Evolución de los servicios de mapas en la Web.....	12
1.15 Sistemas automatizados existentes vinculados al campo de acción.....	13
1.16 Tendencias y tecnologías actuales a considerar.....	15
1.16.1 La política de migración hacia software libre.....	15
1.16.2 Internet	15
1.16.3 Internet y los GIS	16
1.16.4 Aplicaciones Web.....	16
1.16.5 Lenguajes de Programación para la Web.....	17
1.16.6 Selección del lenguaje a utilizar.....	20
1.16.7 Hojas de Estilo en Cascada (Cascade Style Sheets, CSS).....	21
1.16.8 Sistemas Gestores de Bases de Datos (SGBD).....	22
1.16.9 Características de los SGBD de la propuesta.....	24
1.16.10 Metodologías de Desarrollo de Software.....	25
1.16.11 Selección de la metodología a utilizar.....	28
1.16.12 Propuesta.....	29
1.16.13 Conclusiones.....	29
CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA.....	30
2.1 Introducción.....	30
2.2 Modelo del dominio propuesto.....	30
2.2.1 Glosario de términos.....	31

2.2.2	Modelo de objeto del negocio.....	31
2.3	Modelo del sistema.....	32
2.3.1	Requerimientos funcionales.....	33
2.3.2	Requerimientos no funcionales.....	35
2.3.3	Descripción del Sistema Propuesto.....	36
2.3.4	Actores del sistema.....	36
2.3.5	Casos de uso del sistema.....	37
2.3.6	Diagrama de casos de uso del sistema.....	40
2.3.7	Descripción de los casos de uso expandidos.....	41
2.4	Conclusiones.....	65
CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA.....		66
3.1	Introducción.....	66
3.2	Modelo de análisis.....	66
3.2.1	Diagramas de Clases de Análisis.....	66
3.3	Modelo de Diseño.....	67
3.3.1	Principios de diseño gráfico.....	68
3.3.2	Tratamiento de excepciones.....	68
3.3.3	Diagrama de Clases de Diseño.....	69
3.3.4	Descripción de las clases.....	76
3.3.5	Descripción de las tablas de la Base de Datos.....	79
3.3.6	Diagrama De interacción.....	80
3.4	Conclusiones.....	83
CAPÍTULO 4: IMPLEMENTACIÓN Y PRUEBA.....		84
4.1	Introducción.....	84
4.2	Modelo de despliegue.....	84
4.3	Implementación. Modelos de implementación.....	84
4.3.1	Diagrama de componentes.....	85
4.4	Descripción preliminar del modelo de pruebas.....	87
4.4.1	Modelo de prueba.....	88
4.5	Conclusiones.....	90
CONCLUSIONES GENERALES.....		91
RECOMENDACIONES.....		93
BIBLIOGRAFÍA.....		94
ANEXOS I.....		95
ANEXOS II.....		98
ANEXOS III.....		103
ANEXO IV.....		106
GLOSARIO.....		110

ÍNDICE DE FIGURAS

Fig.1. 1 Gráfico de pastel.....	95
Fig.1. 2 Gráfico de barras.....	95
Fig.1. 3 Mapa Temático	96
Fig.1. 4 Mapa de punto	96
Fig.1. 5 Cartograma.	97
Fig.1. 6 coropletas	97
Fig. 2. 1 modelo de dominio	32
Fig. 2. 2 Actor del sistema	36
Fig. 2. 3 Caso de Uso del Sistema.....	37
Fig. 2. 4 Diagrama de CUS	41
Fig.3. 1 Tratamiento de errores.....	69
Fig.3. 2 Arquitectura del sistema.	71

ÍNDICE DE TABLAS

Tabla 3. 1 Clases del Análisis.....	67
Tabla 3. 2 Diagrama de clases persistentes.....	72
Tabla 3. 3 Diagrama del diseño.	73
Tabla 3. 4 Diagrama de clases del Diseño.	73
Tabla 3. 5 Diagrama de clases del Diseño.	74
Tabla 3. 6 Diagrama de clases del Diseño.	74
Tabla 3. 7 Diagrama de clases del Diseño.	75
Tabla 3. 8 Diagrama de clases del Diseño.	75
Tabla 3. 9 Clase Gráfico _ estadístico.....	76
Tabla 3. 10 Clase Acceso_Datos.....	76
Tabla 3. 11 Clase C_Usuario.	77
Tabla 3. 12 Clase Grafico.	77
Tabla 3. 13 Clase Mapa.....	78
Tabla 3. 14 Clase Webservice.	78
Tabla 3. 15 Clase Usuario.....	78
Tabla 3. 16 Usuario.....	79
Tabla 3. 17 Gráfico.	79
Tabla 3. 18 Mapa.	79
Tabla 3. 19 grafico creado.....	79
Tabla 3. 20 mapa creado.	79
Tabla 3. 21 Diagrama de interacción.	80
Tabla 3. 22 Diagrama de interacción.	81
Tabla 3. 23 Diagrama de interacción.	81
Tabla 3. 24 Diagrama de interacción.	82
Tabla 3. 25 Diagrama de interacción.	82
Tabla 3. 26 Diagrama Entidad Relación de la BD.....	83
Tabla 3. 27 Diagrama de clases del análisis.....	98
Tabla 3. 28 Diagrama de clases del análisis.....	98
Tabla 3. 29 Diagrama de clases del análisis.....	99
Tabla 3. 30 Diagrama de clases del análisis.....	99
Tabla 3. 31 Diagrama de clases del análisis.....	99
Tabla 3. 32 Diagrama de clases del análisis.....	100
Tabla 3. 33 Diagrama de clases del análisis.....	100
Tabla 3. 34 Diagrama de clases del análisis.....	101
Tabla 3. 35 Diagrama de clases del análisis.....	101
Tabla 3. 36 Diagrama de clases del análisis.....	102
Tabla 3. 37 Diagrama de clases del análisis.....	102
Tabla 3. 38 Diagrama de clases del Diseño.	103
Tabla 3. 39 Diagrama de clases del Diseño.	103
Tabla 3. 40 Diagrama de clases del Diseño.	104

Tabla 3. 41 Diagrama de clases del Diseño.	104
Tabla 3. 42 Diagrama de clases del Diseño.	105
Tabla 3. 43 Diagrama de interacción.	106
Tabla 3. 44 Diagrama de interacción.	107
Tabla 3. 45 Diagrama de interacción.	108
Tabla 3. 46 Diagrama de interacción.	109
Tabla 4. 1 Diagrama de despliegue.	84
Tabla 4. 2 Capa de Presentación.	86
Tabla 4. 3 Capa de Acceso a Dato.	86
Tabla 4. 4 Seguridad.	86
Tabla 4. 5 Capa de Lógica del Negocio.	87
Tabla 4. 6 Autenticar usuario.	88
Tabla 4. 7 Gestionar Roles.	88
Tabla 4. 8 Generar Gráfico estadístico.	89
Tabla 4. 9 Generar Mapa Temático.	89
Tabla 4. 10 Gestionar Gráfico Estadístico.	89
Tabla 4. 11 Gestionar Mapa Temático.	89
Tabla 4. 12 Configurar Gráfico.	90
Tabla 4. 13 Crear Gráfico Estadístico.	90
Tabla 4. 14 Crear Mapa Temático.	90

INTRODUCCIÓN

La Universidad de las Ciencias Informáticas (UCI), creada al calor de la batalla de ideas y concebida para formar jóvenes con una excelente preparación tanto académica como profesional tiene como objetivo rector la informatización de toda su infraestructura y con ella fomentar al desarrollo de la Industria Cubana del Software.

La UCI 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 Geográfica (SIG, se conoce también como GIS ya que son las siglas del término en inglés de Geographic Information System).

El uso de los Sistemas de Información 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. [Guevara, 2007]

La universidad actualmente cuenta con un Sistema de Información Geográfica, 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 forma:

En las áreas de la UCI no existe una herramienta que permita de forma rápida y segura brindar la información de un área determinada a través de gráficos estadísticos y mapas temáticos para la toma de decisiones.

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 GIS 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 de la 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 es el Sistema de Información Geográfica 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 Geográfica con que cuenta la institución, basado en un lenguaje de programación rápido eficiente y multiplataforma es posible lograr la representación de información vigente en la Universidad.

El **objetivo general** del trabajo será: realizar el diseño y la implementación de un módulo para el Sistema de Información existente en la UCI que permita de una forma cómoda, facilitar a los usuarios la información de un área determinada a través de gráficos estadísticos y mapas temáticos.

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

- Determinar la tecnología a utilizar.
- Estudiar los diferentes tipos de mapas temáticos.
- Realizar el análisis y diseño.
- Investigar que información se desea mostrar en un mapa en las diferentes áreas de la UCI.
- Puesta a punta de la herramienta.
- Elaborar el documento de tesis y la defensa.

Con vistas al cumplimiento de los objetivos se propone la realización de las siguientes **tareas**:

- 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 del 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 (SIG) con que contamos, la representación y búsqueda de información presentes en La Universidad.

En nuestro país no se cuenta con una aplicación similar a esta. En la Universidad de las Ciencias Informáticas tampoco se cuenta con una por lo que se pretende obtener un producto a la altura de las exigencias actuales de la Universidad. El desarrollo de esta aplicación y su posterior explotación, minimizará el tiempo de trabajo necesario para llevar a cabo la toma de decisiones en cualquier área de la universidad.

El presente documento se estructura en cinco capítulos:

En el capítulo 1 se analizarán los principales conceptos relacionados con gráficos estadísticos y mapas temáticos. Se hace un análisis de cómo se encuentran 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.

En este capítulo 2 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.

En el capítulo 3 se aborda aspectos relacionados con la construcción de la solución propuesta, se realiza el diseño de la propuesta de solución, creando los artefactos que ayudan a la construcción del proyecto y la eliminación de todos los errores que puedan aparecer en el camino.

En el capítulo 4 y último, contiene la implementación y las pruebas que permiten llegar a la culminación satisfactoria del módulo.

CAPÍTULO1: FUNDAMENTACIÓN TEÓRICA.

1.1 Introducción.

En el presente capítulo se hace referencia a los objetivos que se persigue con el sistema y las razones que conllevan a su realización. Se analizarán los principales conceptos relacionados con la representación gráfica y su utilidad en los análisis estadísticos de los datos. Se hace referencia a los diferentes tipos de gráficos que se pueden utilizar. Se hace un análisis de las tecnologías y tendencias actuales y que pudieran ser útiles en el desarrollo de la propuesta de solución así como de las metodologías de desarrollo de software. Finalmente, se seleccionan las más apropiadas teniendo en cuenta que las que se utilicen deben garantizar el cumplimiento de los intereses de los usuarios finales.

1.2 Objeto de estudio

El uso de los Sistemas de Información 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 de bueno o malo el lugar en el que se encuentra?, ¿dónde estaría mejor? Estas interrogantes se responden de manera sencilla haciendo uso de los Sistemas de Información.

En la universidad se cuenta con una aplicación de este tipo, pero la misma sólo ofrece las funcionalidades básicas de un sistema de información, las cuales son: acercar, alejar, mover y pedir información de un punto determinado; además permite hacer una búsqueda por edificio, pero, no brinda la posibilidad de mostrar información de un área en forma de gráfico ni permite crear mapas temáticos, los cuales son de vital ayuda en el estudio de información para la toma de decisiones, por tanto este trabajo se basará en la implementación de un módulo que permita darle estas funcionalidades al GIS de la universidad.

1.3 Situación problemática. Flujo actual de los procesos

Actualmente en la Universidad de las Ciencias Informáticas existe un Sistema de Información, 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 a partir del uso del GIS. Y este es uno de los objetivos principales que se persigue cuando se implementa un sistema de este tipo.

¿Qué es lo que sucede hoy realmente en la UCI?

Hoy en todas las entidades o áreas pertenecientes a al UCI existe gran cantidad de información almacenada en las diferentes bases de datos con que cuenta la institución (Akademos, Servicios técnicos). Para obtener los datos, los encargados de analizarlos, trabajan mediante aplicaciones independientes y analizan la información, y luego con la ayuda de alguna herramienta automatizada que permita generar gráficos (generalmente se utiliza los componentes de office para la Web o sistemas elaborados por estos especialistas que dan una solución particular a su problema), elaboran los gráficos a partir de los datos obtenidos y lo incorporan al reporte.

Pero esto no siempre logra cubrir todas las necesidades informativas de los usuarios, debido al uso indistintamente de los diferentes métodos aplicados por cada uno de los usuarios de la información y su atraso en las actividades y la comprensión de los datos. Esto trae como consecuencia la necesidad de un sistema dinámico que permita la presentación de la información a través de gráficos estadísticos y mapas temáticos por sus usuarios, que muestren la información de forma más clara y eficaz, que permita facilitar la comprensión representada, por terceros y además un análisis flexible de la misma para todos los usuarios.

1.4 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 Geográfica 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, circulares, áreas, entre otros.

Además se podrán representar los resultados en forma de mapas temáticos de coropletas¹, donde los colores del mismo permitirán conocer la distribución de los datos en el mapa de la UCI, de manera inmediata.

¹ Colores o tramas para representar el área que ocupa un hecho

1.5 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.

1.6 Objetivo general.

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

1.7 Objetivos específicos.

Del objetivo general se derivan los siguientes objetivos específicos:

- ✓ Realizar un estudio de los diferentes tipos de gráficos estadísticos que permiten representar información.
- ✓ Estudiar las diferentes tipos de mapas temáticos con los cuales se puede representar información de determinada área.
- ✓ Implementar un módulo que permita crear gráficos estadísticos y mapas temáticos dentro del Sistema de Información Geográfica (GIS) de la UCI.

1.8 Los Gráficos estadísticos. Su representación gráfica.

La utilidad de los gráficos puede verse desde dos puntos de vista: 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 y llegar a un buen punto de partida para la toma de decisiones.

En los datos estadísticos, la claridad de su representación es de vital importancia para su comprensión e interpretación. La utilización de un gráfico puede ayudarnos a representar de un modo más eficiente nuestros datos. Los gráficos alcanzan un alto nivel de importancia y utilidad en el proceso de análisis del comportamiento de la información.

En estadística se denomina gráficos a aquellas imágenes que, combinando la utilización de sombreado, colores, puntos, líneas, símbolos, números y un sistema de referencia ²(coordenadas), permiten presentar información cuantitativa. [Molinero, 2007]

Los gráficos además tienen una función estética que atrae la atención del usuario y lo convence del mensaje que se quiere transmitir. En comparación con otras formas de presentación de datos, los gráficos estadísticos nos permiten, de una mirada, comprender el comportamiento de cualquier tipo de variable independientemente del grado de dificultad que esta presente, por lo que sería válido decir que ahorran un tiempo considerable al analista de información ya que los mismos permiten usar las habilidades visuales para procesar información en un tiempo relativamente considerable.

Los gráficos estadísticos se pudieran ver desde dos facetas:

1. Presentación de datos.
2. Análisis descriptivo de la información.

1.9 Presentación de datos: tipos de gráficos.

La representación de los datos estadísticos por medio de gráficos recibe la mayor parte de la atención en un documento, la primera mirada al documento va dirigida a esta parte por lo que es considerada una tarea importante en el proceso de comunicación de información. La presentación de datos mediante gráficos es usada en todas las profesiones diariamente. Podemos decir que actualmente es casi natural este tipo de representación.

Los gráficos son los más convenientes para presentar datos permitiendo una representación visual de la totalidad de la información. Presentan los datos en forma de dibujo de tal modo que se puede percibir fácilmente los hechos esenciales y compararlos con otros.

Existen varios tipos de gráficos. El hecho de que existan variedades se debe a que cada uno está especialmente indicado para representar los datos de una manera distinta. Para obtener la máxima eficiencia al crear los gráficos y presentar los datos de la mejor manera posible se debe tener en cuenta; que cada tipo de gráfico esta destinado para una labor específica.

Existen diferentes tipos de gráficos estadísticos:

- Gráficos de Barras: Representan valores usando trazos verticales u horizontales, aislados o no unos de otros, según la variable a graficar sea discreta o continua. Este gráfico sirve para

² Conjunto de convenciones usadas por un observador para poder medir la posición de un objeto físico en el tiempo y el espacio.

comparar datos entre diferentes segmentos. Pueden usarse para representar una serie³ o más (también llamado de barras comparativas)

- Gráficos de líneas, gráficos de puntos, gráficos de líneas-punto: En este tipo de gráfico se representan los valores de los datos en dos ejes cartesianos ortogonales⁴ entre sí. Ayudan a ver la evolución de los datos. Por lo general se usan para mostrar un mismo tipo de dato y su evolución. Además estos gráficos se utilizan para representar valores con grandes incrementos entre sí.
- Gráficos Circulares: También conocidos como gráficos de " tartas " o " pastel ", permiten ver la distribución interna de los datos que representan un hecho, en forma de porcentajes sobre un total. Después de definir nuestros valores a representar, obtenemos el porcentaje que corresponde a cada uno de los valores y con ello el ángulo de la rebanada correspondiente.
- Gráficos de Áreas: Se usa para mostrar la tendencia de la información generalmente en un período de tiempo.
- Gráficos Mixtos: Se representan dos o más series de datos, cada una con un tipo diferente de gráfico. Son gráficos más vistosos y se usan para resaltar las diferencias entre las series.
- Histogramas: Se utilizan para representa distribuciones de frecuencias. Algún software específico para estadística grafica la curva de gauss superpuesta con el histograma.
- Cartogramas: se utilizan para mostrar datos sobre una base geográfica. La densidad de datos se puede marcar por círculos, sombreado, rayado o color.

Existen otros tipos de gráficos como los dispersogramas y los pictogramas. En esta categoría se encuentran la mayoría de los gráficos utilizados en publicidad. Se los complementa con un dibujo que esté relacionado con el origen de la información a mostrar. [Molinero, 2007] **Error! No se encuentra el origen de la referencia.**

1.10 Análisis descriptivo de la información.

El análisis descriptivo de la información que se obtiene de cualquier documento no es más que la forma que se tiene de desglosar la información que está, hasta cierto punto, con un determinado nivel de complejidad. Lo mismo pasa con la descripción gráfica de la información.

La información descrita a través de gráficos tiene varias aristas. Por ejemplo, un primer paso para abordar análisis estadísticos más complejos, una vez que se tiene la información que se necesita, consiste en

³ Conjunto de resultados observados, generalmente ligados a una secuencia temporal de información.

⁴ El término ortogonal no es más que una generalización de la noción geométrica de perpendicularidad.

presentar esa información de forma que ésta se pueda visualizar de manera más sistemática y resumida. Siempre los datos que interesan van a depender, en cada caso, del tipo de variables que se esté manejando.

Una forma muy sencilla de representar gráficamente estos resultados es mediante diagramas de barras o diagramas de sectores. En los gráficos de sectores 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. [Broader, 2007] Un ejemplo se muestra en la Figura 1.1 donde se muestra la distribución de proyectos y recursos, en las diferentes áreas que opera Fundeso, una Fundación privada, sin ánimo de lucro, que opera como organización no Gubernamental para el desarrollo de los países.

La idea de los diagramas de barras se hace similar a la de los gráficos de sectores que se explican anteriormente. 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.

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.

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. Existen incluso algunos análisis concretos que están basados completamente en la representación gráfica.

1.11 Los mapas, combinaciones de mapas: atlas.

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. Los mapas dan la posibilidad de conocer lugares lejanos.

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. 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. [GEOTEC, 2007]

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.

1.12 Los mapas temáticos.

La Asociación Internacional de Cartografía plantea que "un mapa temático es aquél que está diseñado para mostrar características o conceptos particulares." [GEOTEC, 2007]

Un mapa temático está compuesto por dos elementos fundamentales: Una base geográfica (mapa base), y una capa de contenido temático Figura 1.3. Una vez que el usuario lo analice debe ser capaz de integrarlas, visual e intelectualmente, durante su lectura.

El mapa base consiste en una imagen del territorio que se quiere mostrar, su objetivo es representar geográficamente el contenido temático del mapa; es decir, proporciona información espacial para referenciar el contenido temático y la capa de contenido temático es una capa que contiene la información que se quiere representar sobre el mapa base.

1.13 Tipos de cartografías temáticas.

La cartografía es la ciencia que estudia los mapas y los mapas son la representación geográfica de la tierra o parte de ella representada en una superficie plana. Entonces se puede decir que la cartografía no es más que la representación geográfica de una parte de la superficie terrestre en la que se da información relativa a un tema determinado.

Existen dos tipos de cartografía temática: la cartografía temática cualitativa y la cartografía temática cuantitativa. Para que una información sea del tipo cualitativa debe ser una descripción de características. El objetivo que persigue la cartografía de este tipo 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. Sin embargo, la cartografía cuantitativa se obtiene si se describen valores en la información aportada por el mapa. Aquí se muestran aspectos espaciales de datos numéricos. Pudiera ser que la variable cartografiada sea única, y el mapa se centraría entonces 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).

Teniendo en cuenta estas razones, es decir, las características de este tipo de representación cartográfica, y a partir de los objetivos que le dan vida al módulo, entonces se ha decidido que se implementará este tipo de mapa en este trabajo, pues los cualitativos no muestran la información de una forma que sea útil para la toma de decisiones 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 propósito que tiene la presentación del mapa.
- 2.- El carácter del fenómeno que se esté representando.
- 3.- El tipo y la complejidad de la información cuantitativa.

Para la confección de mapas con información cuantitativa se utilizan algunas técnicas como son las siguientes:

Mapas de punto: en estos mapas la información cuantitativa se representa por puntos, y cada uno representa un valor unitario. Ver Figura 1.4. 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. Figura 1.5.

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 utilizan tramas o colores diferentes aplicados a estas zonas siguiendo el criterio de: a mayor cantidad, más oscuro será el color. Criterio este muy conocido y usado por todos, incluso sin saber su nombre. Esto nos da una idea de la ventaja del uso de este sistema ya que se comprende con facilidad, por parte del usuario, la distribución de los datos en un territorio. Ver Figura 1.6.

En estos mapas se representan normalmente datos ya clasificados mediante intervalos. Al realizar esta clasificación, es evidente que se perderá cierta cantidad de información, (muy poca comparada con los cartogramas) sin embargo permiten tomar del mapa una información general de forma sencilla y rápida. Debido a que esta técnica es utilizada para datos discretos, y este es el tipo de datos con el que se trabajará, se toman estos mapas como punto de partida para darle solución al problema planteado.

1.14 Evolución de los servicios de mapas en la Web.

Los servicios de mapas en la web son aplicaciones que permiten a los usuarios que se conectan a la red de redes, a través de una página web o a través de un software cliente, producir sus propios mapas seleccionando áreas específicas y un set de capas de datos a partir de un menú predefinido.

Estamos en la era en la que podemos decir que 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.

A lo largo de los últimos años se han desarrollado diferentes métodos para obtener, analizar, procesar, y mostrar la información geográfica. La falta de estándares ha provocado que se hayan desarrollado formatos muy diferentes para representar la información dentro de los Sistemas de Información Geográfica.

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. Luego recomenzó con los servicios WEB de mapas que 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 un mayor aprovechamiento en las capacidades de tratamiento de información como las que ofrecen los mapas vectoriales, sumado a la considerable mejora en la representación ante requerimientos de gráficos de alta calidad.

Este problema se puede resolver a través de las especificaciones de implantación desarrolladas por el consorcio openGis para el Web Feature Service (WFS), que les da la posibilidad a los clientes de obtener y actualizar datos codificados en lenguaje de marcado geográfico (GML) desde múltiples servidores que brinden WFS. La especificación define interfaces para el acceso y las operaciones de manipulación en los

objetos geográficos, utilizando el protocolo HTTP como la plataforma de cómputo distribuido. A través de estas interfaces, un usuario o un servicio web puede combinar, usar y administrar datos geográficos desde fuentes de datos diferentes.

Las diferencias que existen entre estas dos formas de representar la información son muchas. Por ejemplo, los recursos que usa en un ordenador un mapa convencional (tipo Ráster) son muchísimo más, a la hora tanto de guardarlos en disco duro como a la hora de usarlos en un programa, que cuando usamos el mismo mapa creado vectorialmente.

1.15 Sistemas automatizados existentes vinculados al campo de acción.

Debido a esta misma complejidad para su tratamiento, en el país no se ha implementado un servicio de este tipo. En la Universidad de las Ciencias Informáticas no existe ningún sistema que se dedique a resolver el problema al que se enfrenta 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. [Ortiz, 2007]

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.

También existen, a nivel mundial, diversos gestores para la recuperación de la información gráficamente como son:

Crystal reports: este es un producto de alta tecnología para la creación e integración de reportes con datos provenientes de múltiples fuentes de datos. Dispone de plantillas de usuario que le permiten rápidamente aplicar un aspecto y estilo uniformes sobre múltiples informes, así como crear atractivas vistas de datos de forma más rápida y sencilla.

Necesita determinados requisitos para su aplicación, los cuales se mencionan a continuación:

- Microsoft Windows XP o superior.
- Computador Pentium o superior.

- Memoria RAM 128 o superior.

Lefebre Iris: Con Lefebre Iris se puede obtener:

- Diseño asistido de cuadros de mando directamente desde MS-Excel.
- Integración de reglas de gestión contable y financiera.
- Explotación en modo conectado o desconectado del servidor.
- Integración y constitución de gráficos dinámicos.
- Generación de los resultados con la posibilidad de cruzar todas las dimensiones de la empresa:

Agata Report: Es un generador de reportes multiplataforma, una herramienta de consulta y generación de gráficos como el Crystal Reports que se conecta a varias Bases de Datos, como PostgreSQL, MySQL, Oracle, DB2, MS-SQL, Informix, InterBase, Sybase, o Frontbase y permite exportar los reportes en formatos como PostScript, plain text, HTML, XML, PDF o CSV (StarCalc, Excel).

Como se puede apreciar, existen una buena cantidad de sistemas automatizados que podría usarse, sin embargo, hay que tener presente que en ninguno de ellos se puede llegar a representar datos de la forma que se propone en el presente trabajo.

Estos gestores son sin dudas realmente efectivos, pero presentan inconvenientes que dificultan su utilización como por ejemplo, el hecho de que la mayoría de estos sistemas son software propietarios y para poder usarlos hay que pagar grandes sumas de dinero en sus licencias, y ninguno de ellos se adapta completamente a nuestras necesidades. Algunos de estos gestores como es el caso de Agata Report, presentan una interfaz complicada, cargada de operaciones para los que se necesita mucha preparación para utilizarlos y debemos tener en cuenta que este módulo será para el uso de las diferentes áreas de la UCI.

Después de un profundo análisis de varios gestores de información, concluimos que se hace necesario el desarrollo de una herramienta software dinámica que pueda recuperar toda la información existente en las bases de datos de las diferentes áreas de la universidad y representarlas en forma de gráficos estadísticos y mapas temáticos.

1.16 Tendencias y tecnologías actuales a considerar.

1.16.1 La política de migración hacia software libre.

Recientemente se ha venido observando una tendencia en la Universidad de las Ciencias Informáticas, y en el país en general, hacia la utilización en grado creciente de software libre. Cada vez se promociona más la migración desde los sistemas con licencia comercial que están en poder de unos pocos monopolios de la rama de la informática – y que se reservan el derecho de vender sus productos a quienes consideren pertinente y “conveniente” –, hacia aquellos que se denominan “libres”, es decir, aquellos cuyo uso por parte de cualquiera que esté interesado en hacerlo está exento de pago o se paga muy poco por el derecho a usarlo.

Por la situación en la que se encuentra nuestro país está prácticamente obligado a llevar a cabo, y cuánto antes mejor, esta migración. Por tanto la UCI, 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, también está en la necesidad de migrar al software libre. El presente trabajo parte de esa premisa y se propone la construcción de un sistema que satisfaga las necesidades planteadas, haciendo uso de herramientas y tecnologías libres.

1.16.2 Internet

Muchas han sido las definiciones que se le han otorgado a Internet desde su surgimiento como por ejemplo: la red de redes, el Sistema Mundial de redes de computadoras interconectadas y otras. Internet tuvo su nacimiento 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* (WWW). En la actualidad Internet, es un espacio público utilizado por millones de personas en todo el mundo como herramienta de comunicación e información que permite tener todos los productos y servicios que requieran fronteras de espacio o tiempo a su alcance.

1.16.3 Internet y los GIS

Indudablemente Internet se ha convertido en el canal de comunicación más importante del mundo, ofrece mayores posibilidades increíbles para intercambiar todo tipo de información. Los Sistemas de Información Geográfica se han ido complementando con este desarrollo.

La rapidez en la evolución de la WWW ha posibilitado su conversión de un sistema hipermedia hasta una completa plataforma informática. Para los usuarios de información geográfica esto ha significado 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.

La forma de tomar decisiones antes de estar implementada la tecnología GIS no siempre era la más idónea, por ejemplo no era nada sencillo decidir cuál era el lugar más apto para ubicar un nuevo negocio o construir un aeropuerto o un centro comercial. La confianza estaba depositada en mapas tradicionales y en tablas estadísticas impresas lo que no quiere decir que estuviesen siempre actualizados. La actualización de alguno de estos documentos resultaba una tarea difícil ya fueran mapas, tablas o cartografía, y aún con todo actualizado, con las mejores versiones de ellos, se podía tener de forma exacta la información que se necesitaba analizar. Por tanto el resultado obtenido eran decisiones basadas en información pobre, al no tener todas las alternativas visualizadas en su conjunto, lo que trae consigo una mala planificación.

En la actualidad Internet nos ha posibilitado el acceso a Sistemas de Información Geográficas que nos permiten realizar consultas de tipo geográfico y cartográfico del mundo, con seguridad de obtener datos precisos y actualizados, ya sea durante un viaje, desde tu centro de trabajo o sencillamente desde tu propia casa.

1.16.4 Aplicaciones Web.

La plataforma Web tiene una arquitectura Cliente/Servidor. Funciona en modo “desconectado”, lo que significa que un usuario, usando un navegador o cliente, hace a través de la red, una petición de una página Web a un servidor Web, el servidor recepciona dicha petición, la procesa y le envía la respuesta al cliente (en formato HTML, también a través de la red), este la recepciona y se desconecta.

Se puede apreciar que los componentes de la arquitectura Web son: el servidor Web, la red física que permite la comunicación y un navegador o cliente.

Existen casos en que esta arquitectura es un poco más compleja, o sea, incluye un nuevo elemento: una aplicación que se ejecuta en el servidor. Este tipo de arquitectura permite manejar lógica de negocio a través de una "aplicación Web", que es como se denomina a este tipo de sistemas Web. La aplicación que se ejecuta en el servidor se encarga de controlar el estado del negocio y de gestionar los datos almacenados con ayuda de algún Sistema Gestor de Bases de Datos (SGBD).

En la actualidad se ha generalizado el uso de aplicaciones Web dada las grandes posibilidades que brindan, y dado que los clientes sólo necesitan un navegador, capaz de interpretar código con formato HTML, para hacer uso de ellas: no tienen que instalar ningún componente de software adicional. Por tanto se propone que el sistema a desarrollar consista en una aplicación Web.

1.16.5 Lenguajes de Programación para la Web.

Entre los distintos lenguajes de programación para la Web que existen en la actualidad, se destacan dos grupos, que se diferencian entre sí por el lugar que ocupan en la arquitectura Cliente / Servidor característica de los sistemas Web. El primer grupo está formado por los lenguajes que se ejecutan en el servidor (en inglés, server side languages). Como ejemplos más sobresalientes tenemos algunos como PERL, ASP, PHP, Java, JSP, los módulos CGI, etc. Estos se caracterizan por desarrollar la lógica de negocio dentro del Servidor, además de ser los encargados del acceso a Bases de Datos, tratamiento de la información, etc. Dentro del segundo grupo se encuentran aquellos lenguajes que se ejecutan en el cliente (client side languages). En este caso están el JavaScript y el Visual Basic Script, que son los encargados de aportar dinamismo a la aplicación en los navegadores.

Es importante seleccionar el lenguaje a utilizar, tanto del lado del servidor como del lado del cliente. A continuación se hace un breve análisis de los lenguajes de uso más común en la actualidad, con vistas a hacer una apropiada selección.

1.16.5.1 Lenguajes Del lado del Servidor:

El 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 funciones.

Perl (Practical Extracting and Reporting Language):

Es un lenguaje de programación muy utilizado para construir aplicaciones CGI para el Web, y es considerado el lenguaje perfecto para este fin dadas sus facilidades en cuanto a la manipulación de texto. Es un lenguaje de libre uso. 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, al igual que muchos otros lenguajes de Internet como JavaScript o ASP.

ASP (Active Server Pages):

Es una tecnología propietaria de Microsoft. Se utiliza exclusivamente en los servidores Web de Microsoft (Internet Information Server y Personal Web Server), lo cual constituye su principal desventaja. Los scripts ASP se ejecutan, por lo tanto, en el servidor y puede utilizarse conjuntamente con HTML y JavaScript para realizar tareas interactivas y en tiempo real con el cliente. Con ASP se pueden realizar fácilmente páginas de consulta de bases de datos, funciones sencillas como obtener la fecha y la hora actual del sistema servidor, cálculos matemáticos simples, etc. 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, y que presenta algunas diferencias en cuanto a sintaxis, de modo que se ha de tratar de distinta manera uno de otro. Para implementar ASP.NET es necesario instalar en el Servidor la Plataforma .NET.

PHP (Personal Home Page):

Es el acrónimo de Hypertext Preprocessor. Es un lenguaje de programación del lado del servidor gratuito, de código abierto, e independiente de plataforma, muy rápido, con una gran librería de funciones y mucha documentación. Es también un lenguaje interpretado y embebido en el HTML. Su sintaxis es muy parecida a la del lenguaje C, por lo que para cualquier programador que esté familiarizado con dicha sintaxis será fácil aprender a programar en PHP. Actualmente se encuentra en su última versión lanzada en febrero de 2007 que es la versión 5.2.1. [wikiprod, 2007]

Cuenta con una de las comunidades más grandes de Internet, lo cual facilita encontrar ayuda, documentación, y otros recursos relacionados.

PHP cuenta con un motor de plantillas denominado SMARTY que permite separar la lógica de la programación de la presentación, es decir, el código PHP del código HTML. Por tanto se puede modificar uno sin afectar el otro. SMARTY es sumamente rápido, entre otros aspectos porque cada plantilla se compila sólo una vez y sólo recompila aquellas que han sido modificadas. Permite que una misma plantilla

sea utilizada por varias páginas PHP que muestren el contenido en el mismo formato, independientemente de que los procedimientos usados para obtener la información a mostrar sean diferentes.

Otra de las ventajas de utilizar PHP, es poder usar la librería ADOdb, una capa de abstracción de base de datos, de alta velocidad, y que tiene características avanzadas, como la gestión de sesiones, generación automática del código SQL, simulación de SELECT LIMIT para todas las bases de datos y monitorización del rendimiento. Su aprendizaje resulta muy fácil, sobre todo si se está familiarizado con la programación de Windows, ya que utiliza muchas convenciones de ADO. Esta capa lleva siendo utilizada desde el año 2000, actualmente, con una comunidad amplia de usuarios. Su licencia es más que flexible (BSD). Esto significa que se puede incorporar (incluso compilar) en las aplicaciones libre de tasas, sin necesidad de solicitar el permiso del autor.

1.16.5.2 Lenguajes del lado del 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.

En sus inicios fue desarrollado por Netscape y la primera versión JavaScript 1.0 fue por primera vez introducida en su navegador Netscape Navigator 2.0. Luego surgió la versión JavaScript 1.1 que se introdujo en el Netscape Navigator 3.0. Sin embargo hasta este momento no era soportado por el navegador de Microsoft, Internet Explorer. Luego de algunos esfuerzos se logró total compatibilidad entre

JavaScript 1.3, que está incluido en Netscape Navigator 4.06 y posteriores versiones, y ECMA-262, un estándar para JavaScript introducido por Microsoft en el Internet Explorer.

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.

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 pre compilados, 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.

1.16.6 Selección del lenguaje a utilizar.

Del lado del servidor:

Aunque PHP es un lenguaje relativamente nuevo, fue diseñado desde cero con el único fin de diseñar aplicaciones web. Esto quiere decir que las tareas más habituales en el desarrollo de estas aplicaciones, pueden hacerse con PHP de forma fácil, rápida y efectiva. Otros lenguajes, como ASP, Perl o Java serán sin duda más completos y potentes, pero no fueron diseñados con este enfoque especializado. Es fácil dar los primeros pasos y ver los resultados rápidamente.

PHP combina excelentemente con otras buenas herramientas, como son el servidor Apache y la base de datos mysql (o sql, o posgres), todas ellas gratuitas y usadas en la Universidad de Ciencia Informáticas. ASP necesita, para algunas funcionalidades (algunas realmente básicas) acudir a programas y extensiones comerciales, de terceras empresas. Pagando, por supuesto.

PHP, con todas las ventajas que presenta por su propia esencia (es el más rápido de todos los analizados, es multiplataforma, con una sintaxis familiar a un gran número de usuarios, y cuenta con gran disponibilidad de recursos en Internet). Además, es fácil de aprender. Por ello se propone su uso como lenguaje del lado del servidor.

Del lado del cliente:

Como lenguaje del lado del cliente se propone el uso de JavaScript, que es un lenguaje orientado a eventos e interpretado, es decir, no requiere compilación. El navegador del usuario se encarga de interpretar las sentencias JavaScript contenidas en una página HTML y ejecutarlas adecuadamente.

Gracias a su compatibilidad con la mayoría de los navegadores modernos, es el lenguaje de programación del lado del cliente más utilizado. Con Javascript el programador se convierte en el verdadero dueño y controlador de cada cosa que ocurre en la página cuando la está visualizando el cliente.

1.16.7 Hojas de Estilo en Cascada (Cascade Style Sheets, CSS).

En términos de páginas Web se puede establecer una diferencia entre los términos estilo lógico y estilo físico. El estilo lógico se refiere a la lógica del documento: cabeceras, párrafos, etc., no se preocupa de la apariencia final, sino de la estructura del documento. Por el contrario, el estilo físico no se preocupa de la estructura del documento, sino por la apariencia final: párrafos con un cierto tipo de letra, tablas con un determinado color de fondo, etc.

La finalidad de las hojas de estilo es crear unos estilos físicos, separados de las etiquetas HTML (en lugar de como parámetros de las etiquetas), y aplicarlos en los bloques de texto en los que se quieran aplicar. Estos estilos podrán ser modificados en algunas ocasiones desde JavaScript, lo que brinda mayor

interactividad. En resumen, las hojas de estilo permiten separar el formato visual de las páginas de contenido. Por la ventaja que representan, se propone su utilización en la propuesta de solución.

1.16.8 Sistemas Gestores de Bases de Datos (SGBD).

“Los Sistemas Gestores de Bases de Datos son un tipo de software muy específico, dedicado a servir de interfaz entre las bases de datos y las aplicaciones que la utilizan. En los textos que tratan este tema, o temas relacionados, se mencionan los términos SGBD y DBMS, siendo ambos equivalentes, y acrónimos, respectivamente, de Sistema Gestor de Bases de Datos y Data Base Management System, su expresión inglesa.” [wikiprod, 2007].

En la actualidad existe una gran variedad de SGBD, tanto de tipo comercial como libre. Entre los más usados dentro del grupo de los comerciales se encuentra **Oracle**, el cual es considerado el SGBD más completo que existe. Sus características más destacadas son el soporte de transacciones, su gran estabilidad y seguridad, su escalabilidad, así como que es un sistema multiplataforma, entre otras ventajas. “Su mayor defecto es su enorme precio, que es de varios miles de euros (según versiones y licencias).” [wikiprod, 2007].

En sus inicios fue muy revolucionario dado que usaba la filosofía de bases de datos relacionales, algo que por los años 70, fecha en que surge Oracle, era todavía desconocido. Hasta hace poco su dominio en el mercado de los servidores de bases de datos empresariales era casi total, pero recientemente está sufriendo la competencia del MS SQL Server de Microsoft y de la oferta de otros SGBD libres.

SQL Server es un potente SGBD que está totalmente habilitado para Web. “Ostenta marcas de referencia en cuanto a escalabilidad y confiabilidad, que son críticas para el éxito de bases de datos de gran tamaño. El SQL Server permite lograr una gran velocidad en el procesamiento de transacciones, y agilidad en todas sus operaciones.” [wikiprod, 2007].

A pesar de todas las ventajas que presenta este SGBD, tiene el inconveniente de que, al igual que Oracle, no es un sistema libre.

MySQL “implementa funcionalidades Web que permiten un acceso a los datos, seguro y fácil, desde Internet.”[González, 2007]. Es el sistema gestor de bases de datos “Open Source” más popular, o sea que puede ser bajado de Internet y usarlo sin tener que pagar, además que cualquiera puede estudiar su código y adecuarlo a las necesidades que requiera. [González, 2007].

La licencia GPL de MySQL obliga a distribuir cualquier producto derivado (aplicación) bajo esa misma licencia. Por tanto MySQL tiene sus restricciones: 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.

MySQL opera en una arquitectura cliente/servidor. "MySQL es muy rápido, fiable y fácil de usar, surge para manipular bases de datos muy grandes. Es un sistema multiplataforma de base de datos relacionales, lo que da velocidad y flexibilidad, cuenta con un sistema de privilegios contraseñas muy seguro que permite la autenticación básica para el acceso al servidor". [wikiprod, 2007].

El lenguaje PHP es altamente compatible con MySQL, por el amplio conjunto de comandos definidos para el tratamiento de este.

Por otra parte está PostgreSQL que está considerado como el SGBD de código abierto más avanzado del mundo. PostgreSQL proporciona un gran número de características que normalmente sólo se encontraban en las bases de datos comerciales de alto calibre tales como DB2 u Oracle. [González, 2007].

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).

Es altamente extensible: soporta operadores y tipos de datos definidos por el usuario. Soporta la especificación SQL99 e incluye características avanzadas tales como las uniones (joins) SQL92. Cuenta con una API (del inglés Application Program Interface) flexible lo cual ha permitido dar soporte para el desarrollo con PostgreSQL en diversos lenguajes de programación entre los que se incluyen: Object Pascal, Python, Perl, PHP, ODBC, Java/JDBC, Ruby, TCL, C/C++, y Pike. Tiene soporte para lenguajes procedurales internos, incluido un lenguaje nativo denominado PL/pgSQL, el cual es comparable con el lenguaje procedural de Oracle PL/SQL. Es totalmente libre.

1.16.9 Características de los SGBD de la propuesta.

Como principal objetivo del módulo que se presenta está el manejo de toda la información posible de un área a la que se vaya a ser referencia geográficamente. Este es el motivo por el cual se trabaja con más de una base de dato, pues la información solicitada puede ser de diferentes naturalezas, por ejemplo computadoras reportadas por edificios, que está disponible en la base de datos del área de soporte técnico; estudiantes suspensos en una asignatura dada se encuentra en “Akademos”, entre otros.

Con el objetivo de 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 que sería la encargada de gestionar toda la información que se requiera de las mismas y que se pueda tener acceso a esta clase desde cualquier aplicación que desee hacer uso de ellas.

Este es hoy en día un elemento muy importante para el software que se realice 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 servicios técnicos, 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 en este momento no está resuelto ese problema. Actualmente se está trabajando con el protocolo estándar SOAP (siglas de Simple Object Access Protocol). Uno de los protocolos utilizados en los servicios Web para permitir un mejor intercambio de información. Es decir, permite que dos programas se comuniquen de una manera muy similar, técnicamente, a la invocación de páginas Web, permitiendo que los servicios se puedan consumir desde cualquier plataforma.

Esta peculiaridad con la que cuenta en la actualidad 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.

El diseño de este módulo permitirá tener acceso a SGBD como: MySQL, PostgreSQL, SQL Server debido a que estos son los que con más frecuencia se usan en la universidad y la base de datos que se encargaría del sistema estará sobre MySQL debido a que es un sistema multiplataforma de base de datos relacionales, lo que da velocidad, flexibilidad y además, cuenta con un sistema de privilegios y contraseñas, muy seguro que permite la autenticación básica para el acceso al servidor.

1.16.10 Metodologías de Desarrollo de Software.

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. [wikiprod,2007].

Uno de los temas más comunes en el mundo de la informática hoy en día es el de las metodologías de desarrollo de software: cómo trabajar eficientemente evitando las catástrofes que conllevan al fracaso de un gran porcentaje de proyectos. Una metodología tiene como objetivo aumentar la calidad del software que se produce en todas y cada una de sus fases de desarrollo, por medio de una mayor transparencia y control sobre el proceso; “producir lo esperado en el tiempo esperado y con el coste esperado” [Jacobson, 2007].

Durante los últimos años se han desarrollado dos corrientes en lo referente a las metodologías de desarrollo de software, las llamadas “pesadas” y las llamadas “ligeras o ágiles”. Las primeras se basan en la idea de conseguir el objetivo común por medio de orden y documentación, mientras que las segundas tratan de lograrlo por medio de la comunicación directa e inmediata entre aquello que intervienen en el proceso. Analizaremos a continuación tres de las más conocidas, sus características, ventajas y desventajas.

El Proceso Unificado de Rational (RUP):

El Proceso Unificado Racional (RUP siglas en inglés de Rational Unified Process), es un proceso de desarrollo de software y junto con el Lenguaje Unificado de Modelado UML, constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos. Es una de las metodologías más generales de las que existen en la actualidad, pues está pensada para adaptarse a cualquier proyecto, no sólo de software. Es el producto final de treinta años de desarrollo y uso práctico. Esta metodología también tiene la ventaja de venir acompañada de una potente herramienta que soporta todos los procesos básicos de RUP: Rational Rose Enterprise Edition 2003.

Las principales características del proceso unificado de modelado son:

Guiado por lo casos de uso: Los casos de uso reflejan lo que los usuarios futuros necesitan y desean, lo cual se capta cuando se modela el negocio y se representan a través de los requerimientos. A partir de aquí los Casos de Uso guían el desarrollo ya que los modelos que se obtienen como resultado de los diferentes flujos de trabajo, representan la realización de los Casos de Uso.

Centrado en la arquitectura: La Arquitectura muestra la visión común del sistema completo en la que todos los participantes deben estar de acuerdo, por lo que describe los elementos del modelo que son más importantes para su construcción, los cimientos del sistema que son necesarios como base para comprenderlo.

Iterativo e incremental: RUP propone que cada fase se desarrolle en iteraciones. Una iteración involucra actividades de todos los flujos de trabajo, aunque desarrolla fundamentalmente algunos más que otros. Las iteraciones hacen referencia a pasos en los flujos de trabajo y los incrementos al crecimiento del producto.

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. Tiene como hito los dejar claros y bien especificados los objetivos.
- **elaboración:** se hace un plan de proyecto, se completan los casos de uso y se eliminan los riesgos. Tiene su hito en la arquitectura del sistema.
- **construcción:** se concentra en la capacidad operativa inicial de un producto y el manual de usuario.
- **transición:** Se centra en el realice del sistema.

Programación Extrema (XP):

La Programación Extrema, o eXtreme Programming, es otra de las metodologías de desarrollo de software que existen en la actualidad. “Mientras que el RUP intenta reducir la complejidad del software por medio de estructura y la preparación de las tareas pendientes en función de los objetivos de la fase y actividad actual, XP, como toda metodología ágil, lo intenta por medio de un trabajo orientado directamente al objetivo, basado en las relaciones interpersonales y la velocidad de reacción.” [Molpeceres, 2007].

Supone la disposición en todo momento, por parte del equipo de trabajo, de un representante competente del cliente, que debe estar en condiciones de dar una respuesta rápida y correcta a cualquier pregunta del equipo de desarrollo de forma que no se retrase la toma de decisiones.

La base para el desarrollo del software que usa esta metodología son las llamadas User Stories, historias escritas por el cliente en las que describe escenarios sobre el funcionamiento del sistema y que no sólo están limitados a la interfaz de usuario, sino que también pueden describir modelos, dominio, etc. Estas

User Stories junto a la arquitectura que se persigue, sirve de base para crear un plan de “entregas de software” entre el equipo de desarrollo y el cliente, para cada una de las cuales se definen objetivos y las iteraciones (generalmente cortas) necesarias para cumplirlos. Las User Stories y los casos de pruebas son la base sobre la que se asienta el trabajo del desarrollador.

Esta metodología apuesta por iteraciones cortas que generan software que el cliente puede ver.

La codificación del software se realiza siempre en parejas (dos programadores, un ordenador), las cuales no son fijas sino que rotan a lo largo del proyecto, y el código que escriben no les pertenece sólo a ellos sino al equipo completo. El objetivo ideal sería que cada integrante del equipo trabaje al menos una vez con cada uno de los demás integrantes y con cada componente software, de forma que el conocimiento de la aplicación completa lo posea el equipo entero y no unos pocos miembros. Se programa sólo la funcionalidad requerida para la entrega en curso, se trabaja en función de las necesidades del momento, por lo que no se le da importancia al análisis como fase independiente.

Desarrollo Guiado por la Funcionalidad (FDD):

Se podría considerar que FDD (por las siglas en inglés de Feature Driven Development) está a medio camino entre RUP y XP, aunque en realidad es más bien una metodología ligera. Está pensada para proyectos con un tiempo de desarrollo relativamente corto (menos de un año). Se basa en un proceso iterativo con iteraciones cortas de aproximadamente dos semanas que producen un software funcional, el cual puede ser examinado por el cliente y la dirección de la empresa. Cada iteración se define en términos de funcionalidades (de ahí su nombre) que son pequeñas partes del sistema con significado para el cliente.

Un proyecto que siga esta metodología estará dividido en cinco fases: desarrollo de un modelo general, construcción de la lista de funcionalidades, plan de entregas sobre la base de las funcionalidades a implementar, diseño basado en las funcionalidades e implementación basada en las funcionalidades. Todo el trabajo se realiza en grupo, aunque siempre hay un responsable, que generalmente tiene mayor experiencia, que dice la última palabra en caso de no llegar a un acuerdo.

Las funcionalidades de cada entrega se dividen entre los distintos subgrupos del equipo y se implementan. El código escrito (las clases) tiene propietario, o sea, sólo quien lo crea puede modificarlo, lo que no ocurre en XP. Es por eso que en un subgrupo deben estar todos los propietarios de las clases implicadas, pudiendo un desarrollador pertenecer a varios subgrupos. También se contemplan como parte

del proceso de implementación, la preparación y ejecución de pruebas, las revisiones de código y la integración de las partes que componen el software.

1.16.11 Selección de la metodología a utilizar.

Las tres metodologías analizadas tienen pocas similitudes entre sí, aunque XP y FDD tienen algunas más al ser ambas del tipo de metodologías ágiles, orientadas al cliente y de iteraciones cortas. Por otra parte, es importante mencionar que algunos autores consideran que dado el carácter general de RUP, todas las otras metodologías son casos particulares de esta. Pero siempre se pueden comparar si se definen determinados criterios.

En cuanto a la forma en que se capturan los requisitos, RUP y XP crean como base los Use Cases (casos de uso) y las User Stories respectivamente, los cuales describen los requerimientos de la aplicación desde el punto de vista del cliente, definen los requisitos técnicos sin abordar detalles de implementación. FDD, por el contrario, no define la parte del proyecto en la que se obtienen los requisitos y sólo define el proceder a partir del momento que ya estos han sido identificados. La captura de requisitos es una parte vital de cualquier proceso de desarrollo de software, puesto que garantiza la eficacia de éste: que el producto final sea lo que espera el cliente.

En cuanto a la relación del equipo con el cliente, XP se basa en la disposición de un representante competente que reúna una serie de condiciones, pero en la práctica, es poco probable que el cliente pueda prescindir de una persona con esas características por un tiempo, ya que le resulta poco menos que imprescindible.

Por otra parte, XP es un proceso muy orientado a la implementación, en el que se genera poca documentación y en que la funcionalidad exacta del sistema final no se define nunca formal y contractualmente. Es por eso que este método es más aplicable para desarrollos internos.

Hoy en día la calidad en el desarrollo y mantenimiento del software se ha convertido 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 todas las aplicaciones.

Por ello y para un mejor control y planificación de este trabajo, se decidió utilizar como metodología RUP (del inglés *Rational Unified Process*), por todas las ventajas de organización que brinda y porque las otras metodologías estudiadas presentan ciertas debilidades que, a juicio del equipo de trabajo, representan

riesgos considerables, como es el caso de una posible captura de requisitos no adecuada. Además esta viene acompañado de una herramienta muy buena que soporta cada uno de los procesos que necesitamos: Rational Rose Enterprise Edition 2003.

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.

1.16.12 Propuesta.

Después del minucioso análisis llevado a cabo se pueden plantear las herramientas que resultarían de mayor eficiencia para la implementación de la propuesta de solución.

Se diseñará este módulo 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. El lenguaje que se utilizará será PHP como lenguaje del lado del servidor por ser uno de los más populares actualmente y brindar facilidades a la hora de conectarse 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, paso por paso y sin muchos problemas en la corrección final, y culminación del trabajo se trabajará con la metodología de desarrollo RUP.

1.16.13 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 de la investigación y exponer, finalmente, las ideas que constituyen la base del desarrollo de este trabajo. Se analizaron las tecnologías principales que podrían ser usadas para la elaboración de un sistema de estas características. A partir de ahí 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.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA.

2.1 Introducción.

En el presente capítulo se describe la propuesta de solución. Se centrará en el modelo de dominio, capturando los tipos más importantes de objetos que existen o los eventos que suceden en el entorno que permitirán mostrar al usuario los principales conceptos que se manejan en el dominio del sistema en desarrollo y para que entiendan los objetivos de la investigación y se familiaricen con su vocabulario, así como los procesos del negocio propuesto. Se plantean los requisitos funcionales y no funcionales de la aplicación a desarrollar y se modela la misma en términos de casos de uso del sistema para obtener mejores resultados.

2.2 Modelo del dominio propuesto.

Teniendo en cuenta que uno de los objetivos fundamentales de la universidad es la informatización de la misma a partir de los procesos que en ella se llevan a cabo y que por esta razón existe gran cantidad de información almacenada en diferentes bases de datos, tendríamos como primer paso para el mejoramiento de los procesos vinculados al campo de acción que se propone la realización de algunas modificaciones al negocio.

Actualmente estos datos son procesados por los distintos usuarios que se conectan a diferentes aplicaciones. Estas aplicaciones se encuentran en funcionamiento en sus respectivas áreas dentro de la Universidad y brindan a sus usuarios reportes de los mismos a través de herramientas informáticas que buscan los datos que van a manipular en las Bases de Datos pertenecientes a dichas áreas. Con el resultado final de esta investigación se pretende lograr que el Sistema de Información Geográfica de la UCI sea capaz de brindar a los usuarios de un área determinada, la información que se pueda obtener de las bases de datos ya existentes permitiéndoles obtener reportes a través de gráficos estadísticos y mapas temáticos.

De este modo el usuario tendrá a su disposición una poderosa herramienta que permitirá obtener reportes en forma de gráficos estadísticos para una mejor comprensión de la información que está manejando. Además los usuarios podrán conocer la distribución en la universidad de la información solicitada a través de mapas temáticos. A continuación se modela el negocio propuesto.

2.2.1 Glosario de términos.

Para una mejor comprensión del trabajo se presentarán los principales conceptos que se tratan a lo largo de este capítulo.

- Llamaremos **Área** al área a la que pertenece el usuario.
- Un **Usuario** es la persona autorizada que manipula la información almacenada en formato digital, y que utiliza una herramienta informática para generar reportes estadísticos de los datos.
- Los **Datos Digitales** representan datos⁵ almacenados en formato digital.
- Las **Herramientas Informáticas** son componente office o sistema informático utilizado por el usuario para generar informes.
- Los **Reportes Estadísticos** son los datos procesados a través de la herramienta informática y que recibe el usuario finalmente.

Principales Eventos

- **Pertenece:** proceso que identifica al usuario especificando la entidad a la que pertenece.
- **Utiliza:** proceso mediante el cual el usuario obtiene una información determinada a través de una herramienta informática.
- **Busca:** proceso mediante el cual el usuario manipula la información necesaria para dar respuesta al pedido de un solicitante.
- **Genera:** proceso mediante el cual se genera el informe de la información especificada.
- **Recibe:** proceso de entrega del informe por listado al especialista.
- **Contienes:** proceso que encapsula el contenido de un reporte estadístico.
- **Consulta:** proceso que permite al usuario analizar los datos digitales a través de la observación.

2.2.2 Modelo de objeto del negocio

El objetivo principal del modelo del dominio es comprender y describir las clases más importantes dentro del contexto del sistema a través de diagramas⁶ UML. El hecho de que se cuenta, en el área que abarca este trabajo, con unos flujos de información difusos, es decir, tienen múltiples orígenes, o el hecho de que ocurran solo eventos o algunos sucesos, conllevan al uso de un modelo de dominio para modelar el negocio propuesto.

⁵ conjunto organizado e integrado de información almacenada y clasificada

⁶ Es la representación gráfica de un conjunto de elementos. Visualizan un sistema desde diferentes perspectivas.

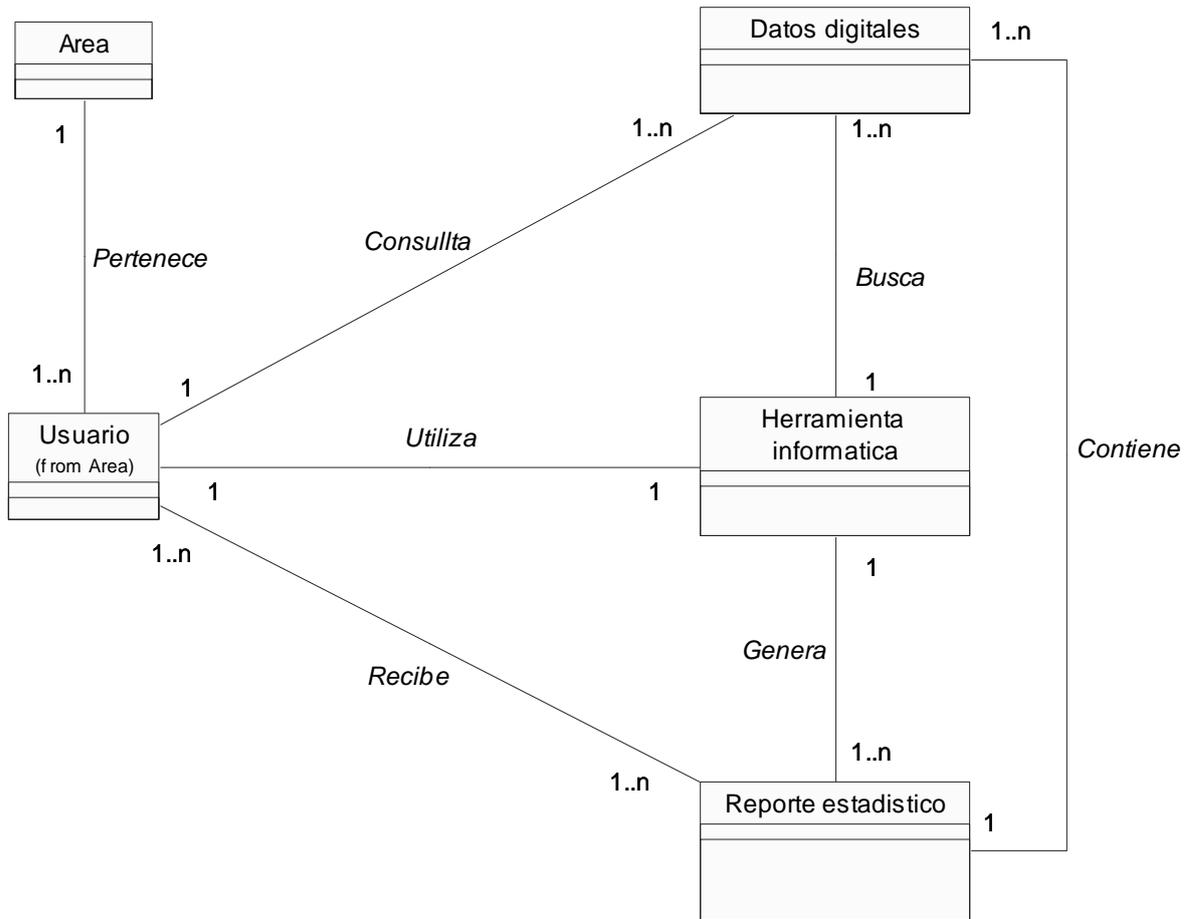


Fig. 2. 1 modelo de dominio

2.3 Modelo del sistema.

A partir de este punto se comienza a modelar el sistema que se va a construir. Para ello se identifican sus requisitos, tanto funcionales como no funcionales, y se modelan los funcionales en términos de casos de uso del sistema.

2.3.1 Requerimientos funcionales.

Los requerimientos funcionales van a ser las capacidades o condiciones que el debe cumplir. Por lo que, luego de conocer los conceptos que encierran al objeto de estudio, se pueden empezar a considerar los elementos con los que debe contar el sistema para que los objetivos planteados al inicio de este trabajo queden bien definidos en términos de la funcionalidad del sistema. Para ello enumeraremos las funciones que el sistema deberá ser capaz de llevar a cabo incluyendo las acciones que podrán ser ejecutadas por el usuario. Para cumplir los objetivos de esta aplicación la misma deberá ser capaz de:

R1 Autenticar Usuario.

- 1.1 Permitir al usuario introducir nombre de usuario y contraseña de dominio UCI.
- 1.2 Validar datos introducidos usando el directorio activo UCI.
- 1.3 Permitir al usuario hacer uso de las funcionalidades a las que tiene acceso.

R2 Generar Gráfico Estadístico

- 2.1 Permitir al usuario escoger el tipo de información a filtrar.
- 2.2 El sistema brinda la opción de seleccionar el tipo de gráfico que se desea graficar.
- 2.3 Generar gráfico estadístico.

R3 Definir formato del informe del gráfico estadístico.

- 3.1 Definir el título que se mostrará en el informe gráfico.

R4 Definir campo de Dato por gráficos.

- 4.1 Permitir especificar el titulo del eje Y para gráficos de tipo línea.
- 4.2 Permitir especificar el titulo del eje Y para gráficos de tipo barra.
- 4.3 Permitir especificar el título del eje X para gráficos de tipo barra.
- 4.4 Permitir especificar el título del eje X para gráficos de tipo línea.
- 4.5 Permitir especificar el color del gráfico de barra.
- 4.6 Permitir especificar el color del gráfico de línea.

R5 Generar Mapa Temático

- 5.1 Mostrar opciones para permitir al usuario escoger el tipo de información a filtrar. En este caso sólo se podrá mostrar la información por Manzanas.
- 5.2 Definir el título que se mostrará en el informe geográfico.
- 5.3 Mostrar opciones para escoger el Color de la escala. El color con que deseamos visualizar la información en el mapa.

- 5.4 Mostrar opciones para especificar los Niveles de escala. Para saber en cuanto quiero degradar el color del mapa.

R6 Gestionar gráfico.

- 6.1 Eliminar gráfico.
- 6.2 Modificar gráfico.

R7 Buscar gráfico.

- 7.1 Mostrar opciones para buscar un gráfico guardado anteriormente.
- 7.2 Muestra los resultados de la búsqueda.

R8 Gestionar mapa.

- 8.1 Eliminar mapa.
- 8.2 Modificar mapa.

R9 Buscar mapa.

- 9.1 Mostrar opciones para buscar un mapa guardado anteriormente.
- 9.2 Muestra los resultados de la búsqueda.

R10 Crear Gráfico.

- 10.1 El sistema brinda la opción, al usuario avanzado, de conectarse al Web Service.
- 10.2 El sistema devuelve la petición hecha por el usuario al Web Service en forma de gráfico estadístico.
- 10.3 Crear gráfico estadístico.

R11 Crear Mapa.

- 11.1 El sistema brinda la opción, al usuario avanzado, de conectarse al Web Service.
- 11.2 El sistema devuelve la petición hecha por el usuario al Web Service en forma de mapa temático.
- 11.3 Crear un mapa temático.

R12 Gestionar Roles.

Asignar Roles.

- 12.1 El Sistema permite al administrador asignar los roles para los usuarios que van a acceder al sistema.

Modificar Roles.

- 12.2 El sistema permite al administrador ver los usuarios y sus respectivos roles.

12.3 El Sistema permite al administrador cambiar los roles para los usuarios que ya tienen un rol definido.

Crear Roles.

12.4 El sistema permite al administrador crear un nuevo rol.

2.3.2 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, permitiendo la utilización del sistema sin mucho entrenamiento.
- Diseño encuadrado para resoluciones de 800x600, pero preparado para verse en otras resoluciones.

Portabilidad

- Sistema multiplataforma.

Seguridad

- Autenticación (Contraseña de acceso.)
- Autorización: Existencia de distintos roles que establezcan las acciones que pueden realizar los usuarios
- Verificación sobre acciones irreversibles (por ejemplo las eliminaciones).

Confiabilidad

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

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.

2.3.3 Descripción del Sistema Propuesto.

Asumiendo todos los requerimientos mencionados anteriormente, se tiene como objetivo darle cumplimiento y funcionalidad a lo planteado como propuesta de solución al sistema que se propone y que debe brindarle al usuario, (que será aquel que tenga acceso al GIS de la universidad: estudiantes, profesores y trabajadores en general), la posibilidad de crear reportes en forma gráficos estadísticos y mapas temáticos a través de consultas a las bases de datos correspondientes, para tener reportes de información concreta de las áreas de la universidad. Para ello se utilizan las bases de datos existente actualmente: Akademos y Servicios técnicos de las cuales se toman los datos que solicite el usuario a través de Web Service y se le dará a conocer la información de manera que pueda manejarla de la forma más adecuada para tomar decisiones, mediante gráficos permitiéndole al usuario darse cuenta con sólo una mirada de la distribución de la información solicitada en la universidad mediante los mismos y a través de un mapa temático también, en este caso particular de coropletas (por colores). Con el presente módulo el Sistema de Información Geográfica dejará de ser un simple visor de conocimiento geográfico, para convertirse en una poderosa herramienta que además de contar con la posibilidad de georreferenciar áreas de la Universidad de Ciencias Informáticas se puede conjuntamente, conocer información específica de la misma de una forma asequible para todos permitiendo una mayor facilidad para la toma de decisiones en las diferentes esferas o áreas de la universidad.

2.3.4 Actores del sistema.

Los actores de un sistema son agentes externos, es decir, aquellas personas o sistemas que interactúan con él. En la siguiente figura se puede observar la representación UML para el actor del sistema propuesto, que se describe en la siguiente tabla.

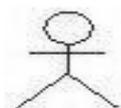


Fig. 2. 2 Actor del sistema

Tabla 2. 1 Actores del sistema

Actor	Justificación
Usuario	Generaliza a todos los usuarios del sistema. Se autentica.

Usuario Básico	Persona que interactúa directamente con la aplicación para generar gráficos estadísticos o mapas temáticos que estén predefinidos.
Usuario Avanzado	Persona que interactúa directamente con la aplicación para crear y gestionar gráficos estadísticos y mapas temáticos a través de Web Service.
Administrador	Persona que interactúa directamente con la aplicación para asignar los roles que tendrán los usuarios.

2.3.5 Casos de uso del sistema.

Los casos de uso del sistema (CUS) son artefactos que describen bajo la forma de acciones y reacciones, el comportamiento del sistema desde el punto de vista del usuario. Su representación en UML es la siguiente.



Fig. 2. 3 Caso de Uso del Sistema

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

Tabla 2. 2 Resumen del Caso de Uso 1

CU-1	Autenticar Usuario
Actor	Usuario
Descripción	El usuario envía al sistema sus datos (nombre de usuario y contraseña). Se le informa si los datos fueron correctos o no.
Referencia	R1

Tabla 2. 3 Resumen del Caso de Uso 2

CU-2	Generar _Gráficos Estadístico.
Actor	Usuario Básico
Descripción	Un usuario solicita la creación de gráficos estadísticos utilizando el sistema para darle respuesta a dicha solicitud.

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

Tabla 2. 4 Resumen del Caso de Uso 3

CU-3	Generar _Mapas Temático.
Actor	Usuario Básico
Descripción	Un usuario solicita la creación de mapas temáticos y utiliza el sistema para darle respuesta a dicha solicitud.
Referencia	R5

Tabla 2. 5 Resumen del Caso de Uso 4

CU-4	Configurar_ Gráfico
Actor	Usuario Básico
Descripción	El usuario especifica al sistema el formato y los campos de datos necesarios para generar el gráfico estadístico.
Referencia	R3, R4

Tabla 2. 6 Resumen del Caso de Uso 5

CU-5	Gestionar_ Gráfico
Actor	Usuario Avanzado
Descripción	El usuario especifica al sistema si desea, eliminar, guardar o modificar el gráfico estadístico.
Referencia	R6

Tabla 2. 7 Resumen del Caso de Uso 6

CU-6	Gestionar_ Mapa
Actor	Usuario Avanzado
Descripción	El usuario especifica al sistema si desea, eliminar, guardar o modificar el mapa.
Referencia	R8

Tabla 2. 8 Resumen del Caso de Uso 7

CU-7	Buscar_ Gráfico
Actor	Usuario Avanzado
Descripción	El usuario selecciona la opción de buscar algún gráfico guardado con anterioridad.
Referencia	R7

Tabla 2. 9 Resumen del Caso de Uso 8

CU-8	Buscar _ mapa
Actor	Usuario Avanzado
Descripción	El usuario selecciona la opción de buscar algún mapa.
Referencia	R9

Tabla 2. 10 Resumen del Caso de Uso 9

CU-9	Crear_ gráfico.
Actor	Usuario Avanzado
Descripción	El usuario Selecciona la opción de Crear un nuevo gráfico.
Referencia	R10

Tabla 2. 11 Resumen del Caso de Uso 10

CU-10	Crear_ mapa.
Actor	Usuario Avanzado
Descripción	El usuario Selecciona la opción de Crear un nuevo mapa.
Referencia	R11

Tabla 2. 12 Resumen del Caso de Uso 11

CU-11	Gestionar _ roles.
Actor	Administrador
Descripción	El usuario especifica al sistema si desea, modificar, asignar o crear un rol.
Referencia	R12

2.3.6 Diagrama de casos de uso del sistema.

Los casos de uso son fragmentos de funcionalidad del sistema. En ellos se describe la secuencia determinada de eventos que realiza un actor en interacción con la aplicación.

El diagrama donde se representa la relación existente entre el actor y los casos de uso del sistema propuesto se representa a continuación:

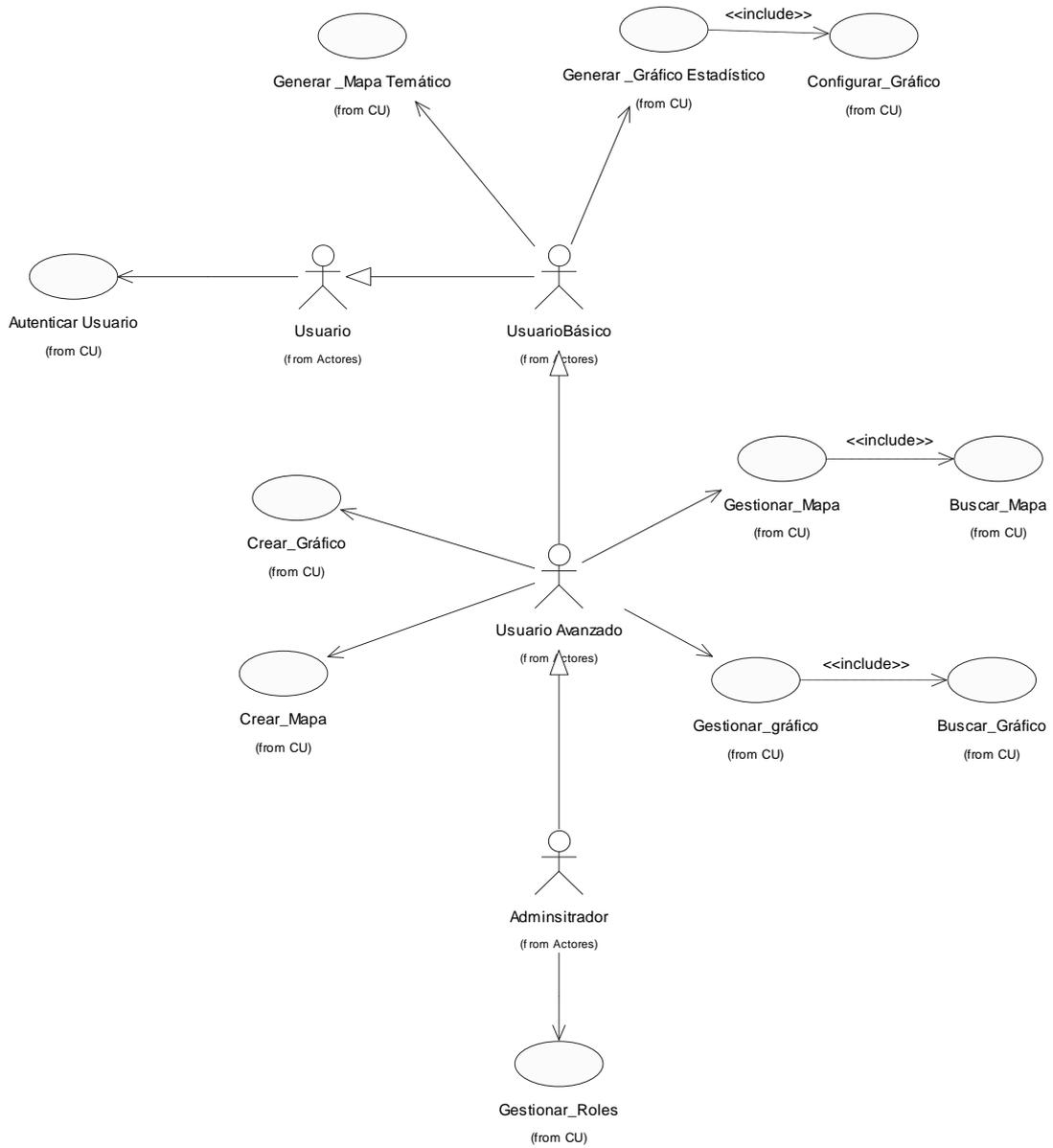


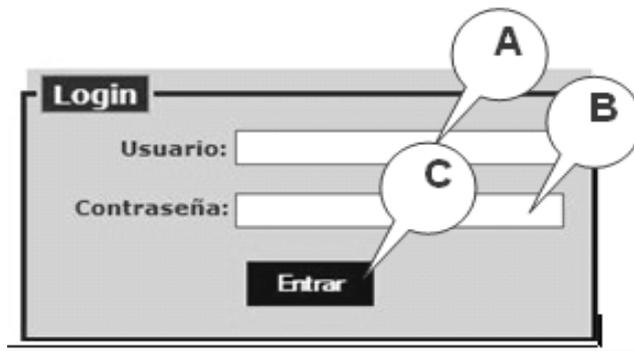
Fig. 2. 4 Diagrama de CUS

2.3.7 Descripción de los casos de uso expandidos.

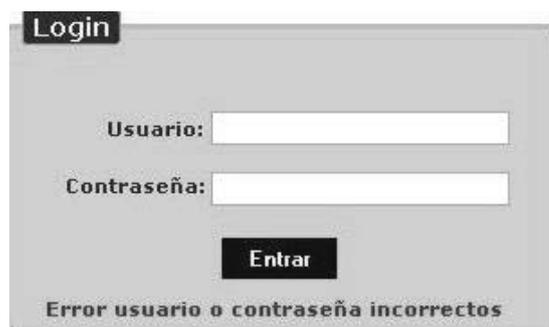
En este epígrafe se describen con un nivel más detallado los casos de uso del sistema.

Tabla 2. 13 Descripción de CU_ Autenticar Usuario.

Caso de Uso	Autenticar Usuario
Actores	Usuario
Resumen	El caso de uso se inicia cuando el usuario envía al sistema sus datos (nombre de usuario y contraseña). Se le informa si los datos fueron correctos o no y termina el caso de uso.
Propósito	Garantizar que en todo momento se conozca quién está accediendo al sistema.
Referencias	R1
Precondiciones	El usuario ha introducido sus datos y ha elegido la opción de autenticarse.



Pantalla 1

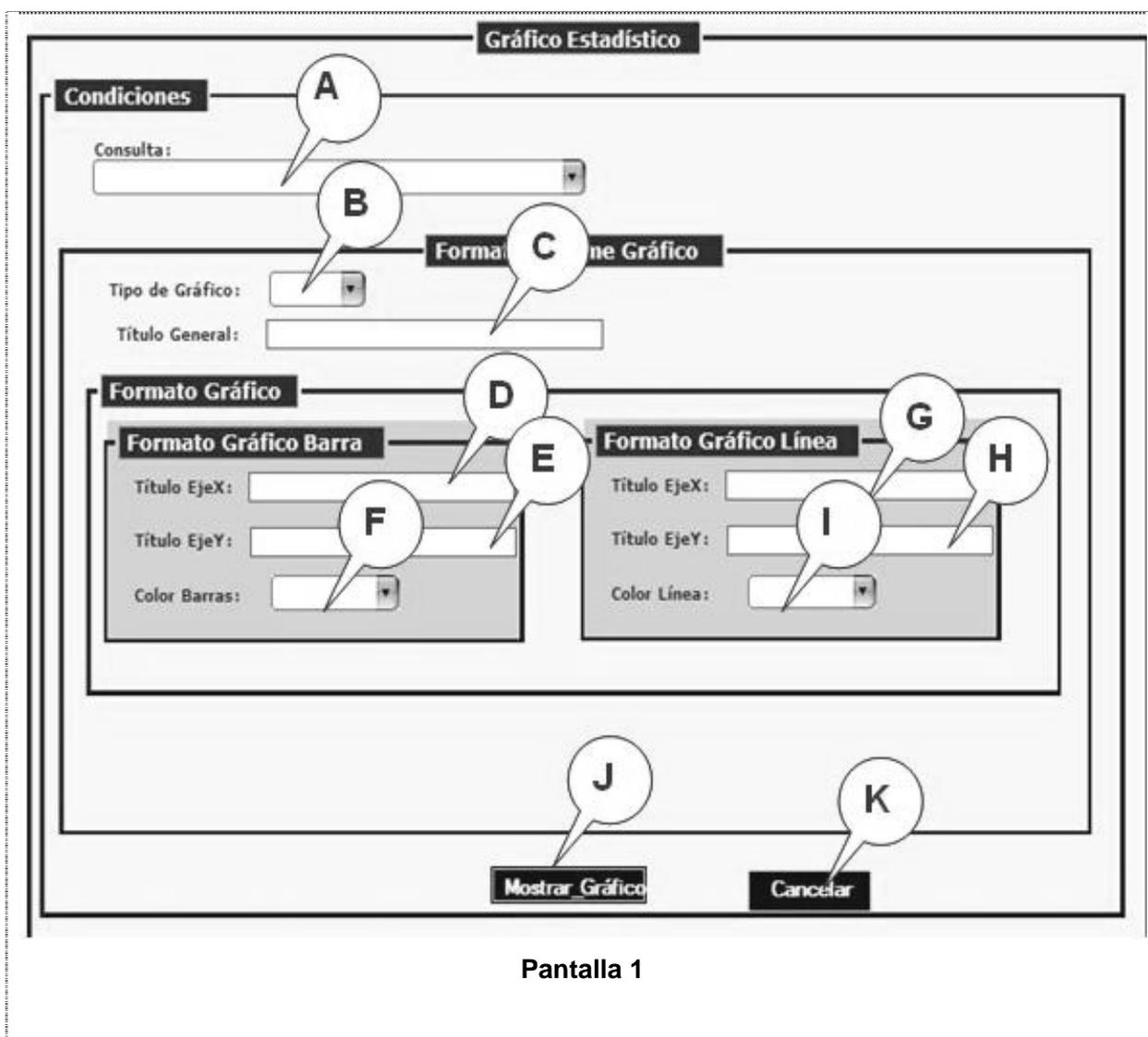


Pantalla 2

Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
	1. El sistema muestra los controles necesarios para que el usuario introduzca su usuario [A] y contraseña de dominio UCI [B]. [Pantalla 1]
2. Introduce los datos requeridos y acciona el botón entrar [C] para ejecutar la operación.	3. Valida la identidad del usuario y le informa que ya está autenticado mostrándole la página a la que tiene acceso el usuario. En caso de que los datos no sean correctos, ver CA1.
Cursos Alternos	
CA1	Muestra mensaje de error [Pantalla 2] .
Poscondiciones	Se ha autenticado un usuario en el sistema.

Tabla 2. 14 Descripción de CU_ Generar Gráficos Estadísticos.

Caso de Uso	Generar Gráficos Estadísticos.
Actores	Usuario básico.
Resumen	El caso de uso inicia cuando el usuario desea obtener una información y quiere visualizarla a través de un gráfico estadístico. Para ello selecciona la información a filtrar y un título general para el gráfico que quiere crear.
Propósito	Crear un gráfico estadístico con el propósito de mostrar la información.
Referencias	R2
Precondiciones	El usuario autenticado debe haber escogido la opción de Generar Gráfico Estadístico.



Pantalla 1



Pantalla 2



Pantalla 3



Pantalla 4

Flujo Normal de Eventos

Acción del Actor

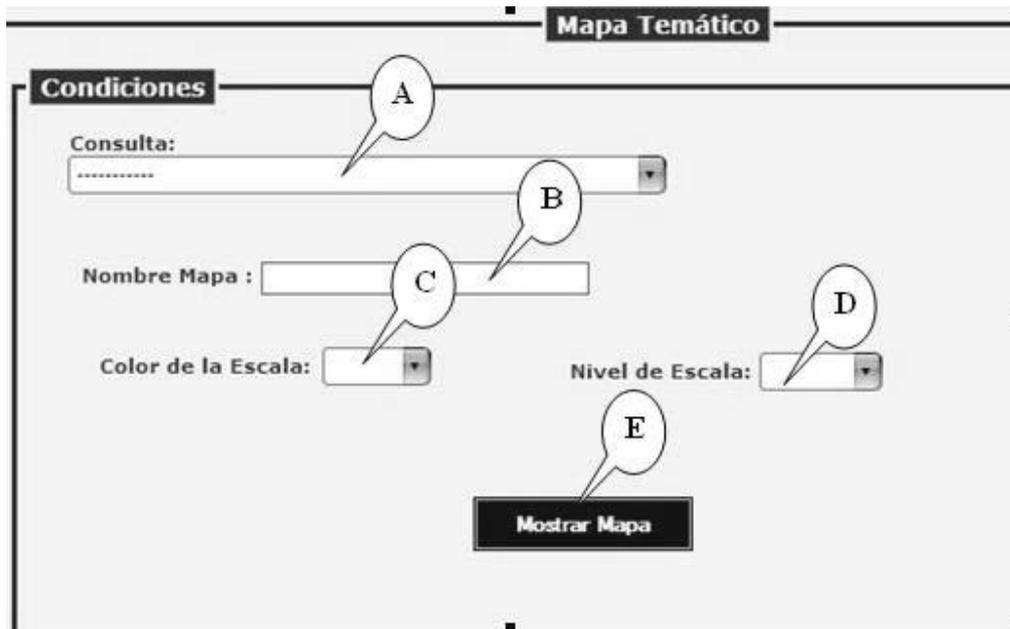
Respuesta del Sistema

	1. El sistema muestra la página de Generar gráfico estadístico. Pantalla 1
2. El usuario debe escoger las informaciones que desea visualizar a través de una consulta predefinida [A]. En caso de que el usuario no escoja ninguna consulta, ver CA1.	3. El sistema muestra la consulta seleccionada por el usuario. [A].
4. El usuario debe escribir un título general para el gráfico que desea mostrar [C]. En caso de que el usuario no escriba un título, ver CA2.	5. El sistema muestra el título general [C].
6. El usuario configura el tipo de gráfico que selecciona [Ver CU_Configurar_ Gráfico] y pulsa el botón mostrar gráfico [J]. Si el usuario no desea mostrar lo que ha seleccionado, Ver CA3.	7. El sistema muestra el gráfico creado. [Pantalla 2]
Cursos Alternos	
CA1	Muestra mensaje de error [Pantalla 3] y retorna al paso 2.
CA2	Muestra mensaje de error [Pantalla 4] y retorna al paso 4.
CA3	El usuario selecciona el botón cancelar [K].
Poscondiciones	Se ha generado el gráfico estadístico.

Tabla 2. 15 Descripción de CU_ Generar Mapa Temático.

Caso de Uso	Generar Mapa Temático.
Actores	Usuario.
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.

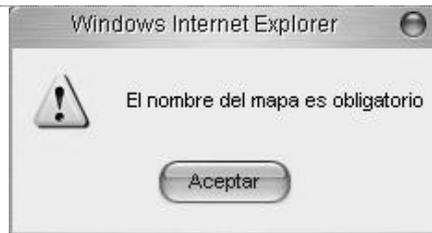
Propósito	Generar un mapa temático.
Referencias	R3
Precondiciones	El usuario autenticado debe haber escogido la opción de Generar Gráfico Estadístico.



Pantalla 1



Pantalla 2



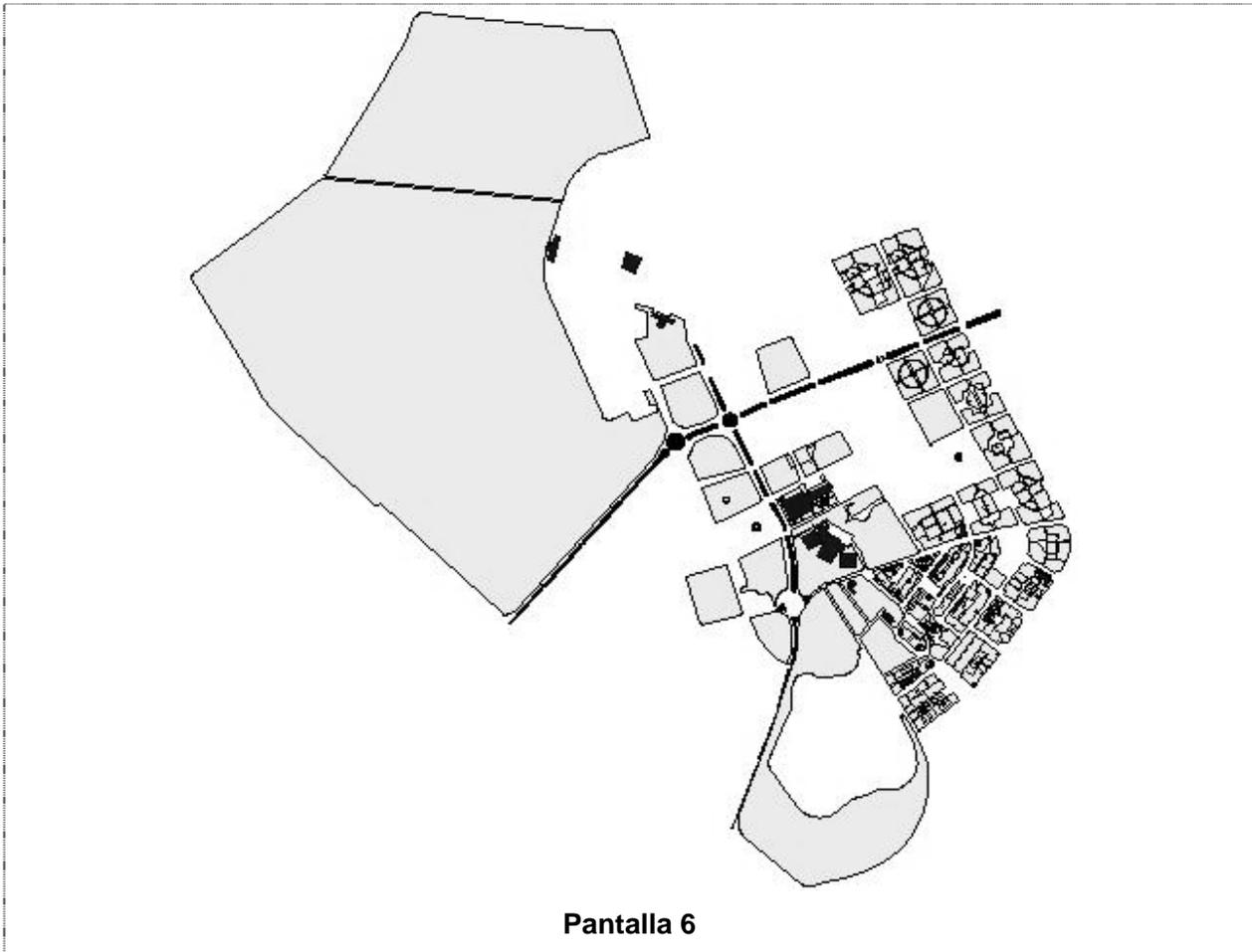
Pantalla 3



Pantalla 4



Pantalla 5



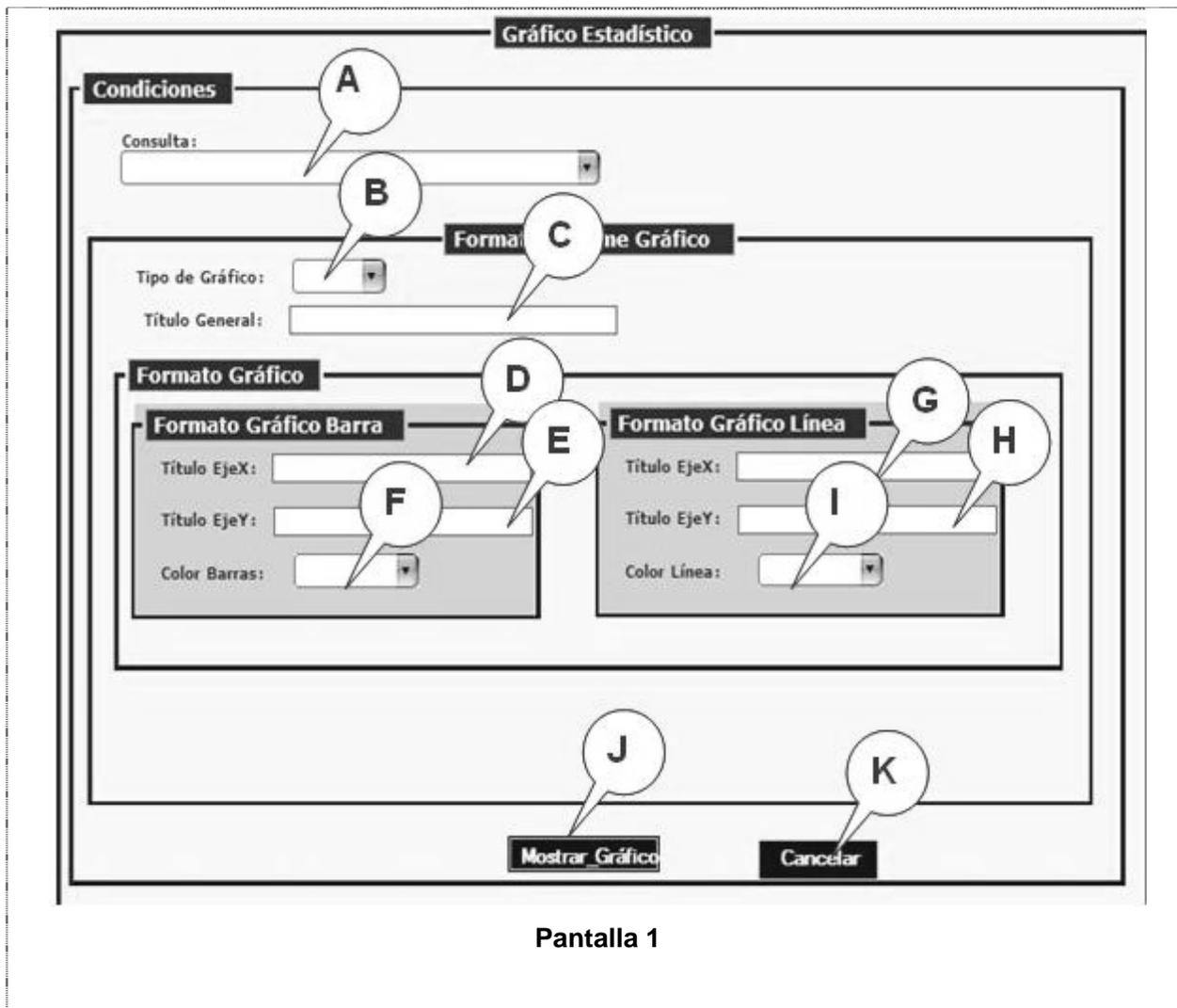
Pantalla 6

Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
	1. El sistema muestra la página de Generar Mapa Temático y carga las consultas. [Pantalla 1] .
2. El usuario escoge consulta [A] que desea mostrar en un mapa temático. En caso de que no elija una consulta ver CA1.	3. El sistema muestra en [A] la consulta que escogió el usuario.
4. El usuario escribe el nombre [B] con el que quiere llamar al mapa. En caso de	5. El sistema muestra en [B] el nombre.

que no elija una consulta ver CA2.	
6. El usuario escoge el color con el que desea mostrar la información en el mapa [C]. En caso de que no elija una consulta ver CA3.	7. El sistema muestra en [C] el color escogido por el usuario.
8. El usuario escoge el nivel de escala con el que va a degradar el color [D]. En caso de que no elija una consulta ver CA4.	7. El sistema muestra en [D] el nivel con el que va a degradar el color escogido por el usuario.
10. El usuario pulsa el botón mostrar mapa [E].	11. El sistema muestra el mapa generado. [Pantalla 6]
Cursos Alternos	
CA1	El sistema muestra un mensaje de error. [Pantalla 2]
CA2	El sistema muestra un mensaje de error. [Pantalla 3]
CA3	El sistema muestra un mensaje de error. [Pantalla 4]
CA4	El sistema muestra un mensaje de error. [Pantalla 5]
Poscondiciones	Se ha generado el mapa temático.

Tabla 2. 16 Descripción de CU_ Configurar_ Gráfico.

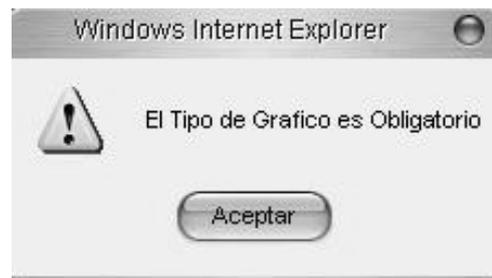
Caso de Uso	Configurar_ Gráfico.
Actores	Usuario Básico.
Resumen	El caso de uso inicia cuando el usuario escoge el tipo de gráfico en el que desea mostrar la información y desea elegir el formato del informe.
Propósito	Configurar el gráfico estadístico que se desea crear.
Referencias	R3, R4
Precondiciones	El usuario debe haber escogido alguna consulta.



Pantalla 1



Pantalla 2



Pantalla 3

Flujo Normal de Eventos

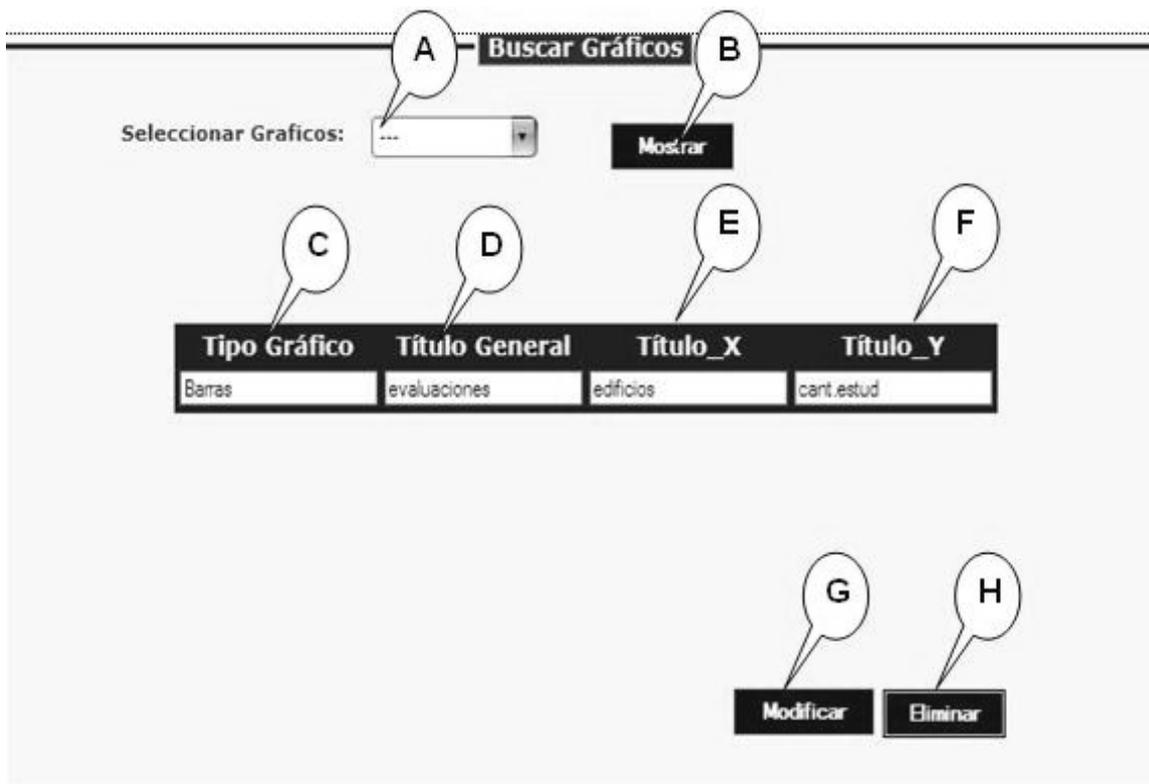
Acción del Actor	Respuesta del Sistema
	1. El sistema muestra la página de generar gráfico estadístico. Pantalla 1.
2. El usuario escoge el tipo de gráfico que desea generar [B] y su título general [C]. En caso de que no escoja un tipo de gráfico ver CA1.	3. Si el tipo de gráfico es: a) pastel, ir a sección Pastel. b) barras, ir a sección Barras. c) líneas, ir a sección Líneas.
4. El usuario llena los campos habilitados que desea. Si es: a) pastel, ir a sección Pastel.	5. El sistema muestra el gráfico estadístico.

b) barras, ir a sección Barras. c) líneas, ir a sección Líneas.	
Sección: Pastel.	
	1. El sistema muestra los controles de configurar gráfico estadístico de barras y de líneas deshabilitados.
Sección: Barras.	
	1. El sistema deshabilita los controles de configurar gráfico estadístico de líneas y deja habilitados los de gráficos de barra.
2. el usuario escribe el título del eje X [D], el título del eje Y [E] y el color con el que quiere mostrar el gráfico de barras [F].	3. El sistema muestra los campos seleccionados o escritos por el usuario.
Sección: Líneas	
	1. El sistema deshabilita los controles de configurar gráfico estadístico de barras y deja habilitados los de gráficos de líneas.
2. el usuario escribe el título del eje X [G], el título del eje Y [H] y el color con el que quiere mostrar el gráfico [I].	3. El sistema muestra los campos seleccionados escogidos o escritos por el usuario.
Cursos Alternos	
CA1	El sistema muestra un mensaje de error [Pantalla 3].
Poscondiciones	Se han llenado los campos de dato para configurar el gráfico estadístico.

Tabla 2. 17 Descripción de CU_ Gestionar_ Gráfico.

Caso de Uso	Gestionar_ Gráfico.
Actores	Usuario Avanzado.
Resumen	El caso de uso inicia cuando el usuario autenticado especifica al sistema

	si desea, eliminar o modificar el gráfico estadístico buscado.
Propósito	Gestionar la información en los gráficos.
Referencias	R6
Precondiciones	Debe existir al menos un gráfico creado en el sistema.



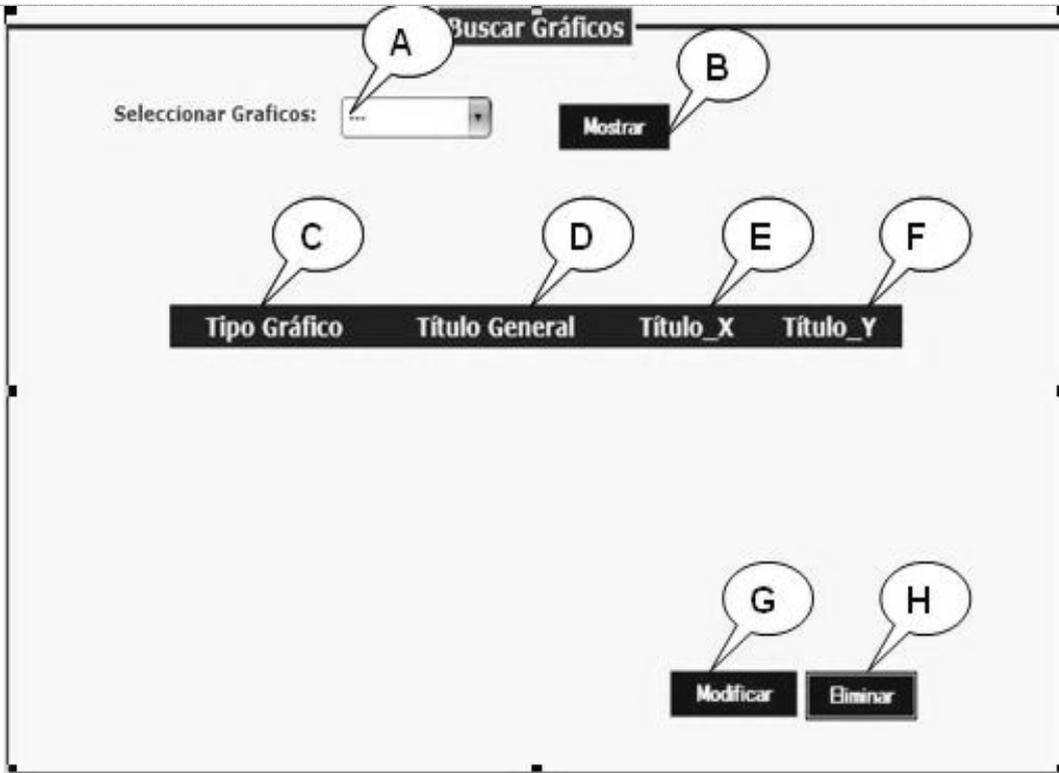
Pantalla 1

Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
	1. El sistema muestra la página de buscar gráfico estadístico. Pantalla 1.
2. El usuario, busca un gráfico creado anteriormente [Ver CU_Buscar Gráfico] y selecciona la acción que quiere realizar	3. Si es: a) Eliminar, ver Sección Eliminar. b) Modificar, ver Sección Modificar

pulsando el botón eliminar [H] o modificar [G],	
Sección: Eliminar.	
	1. El sistema muestra la tabla con los campos de datos del gráfico.
2. El usuario pulsa el botón eliminar [H].	3. El sistema elimina de la lista de gráficos el gráfico mostrado anteriormente.
Sección: Modificar.	
	1. El sistema muestra la tabla con los campos de datos del gráfico.
2. El usuario cambia el campo que desea modificar: Tipo de gráfico [C], título general [D], título en el eje X [E] y/o el título en el eje Y [F].	3. El sistema muestra la tabla con el (los) campo (s) modificado (s).
Poscondiciones	Se ha gestionado el gráfico estadístico seleccionado.

Tabla 2. 18 Descripción de CU_ Buscar _ gráfico.

Caso de Uso	Buscar _ gráfico.
Actores	Usuario Avanzado
Resumen	El caso de uso inicia cuando el usuario avanzado desea buscar un gráfico estadístico que haya sido creado con anterioridad.
Propósito	Buscar un gráfico estadístico.
Referencias	R7
Precondiciones	Debe existir al menos un gráfico creado con anterioridad.



Pantalla 1

Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
	1. El sistema muestra la interfaz para buscar los roles y carga los gráficos que hay en el sistema. [Pantalla 1]
2. El usuario escoge en Seleccionar Gráfico [A], el gráfico que desea mostrar.	3. El sistema muestra nombre del gráfico seleccionado por el usuario [A].
4. El usuario pulsa el botón mostrar [B].	5. El sistema muestra los campos de dato del gráfico seleccionado: Tipo de grafico [C], título general [D], título en el eje X [E] y el título en el eje Y [F].
Poscondiciones	Se muestra el gráfico buscado.

Tabla 2. 19 Descripción de CU_ Crear Gráfico.

Caso de Uso	Crear_ Gráfico
Actores	Usuario Avanzado.
Resumen	El caso de uso inicia cuando el usuario avanzado decide crear una nueva consulta para generar un gráfico estadístico.
Propósito	Crear una nueva consulta.
Referencias	R10
Precondiciones	El usuario avanzado debe haber escogido la opción de crear gráfico.

Pantalla 1

no operations defined in the WSDL document!

Pantalla 2



Pantalla 3



Pantalla 4



Pantalla 5

Flujo Normal de Eventos

Acción del Actor	Respuesta del Sistema
	1. El sistema muestra la página de crear gráfico [Pantalla 1].
2. El usuario escribe la URL del Web Service al que se quiere conectar [A]. En caso de que no escriba una, ver CA1. En caso de que la escriba incorrectamente, ver CA2.	

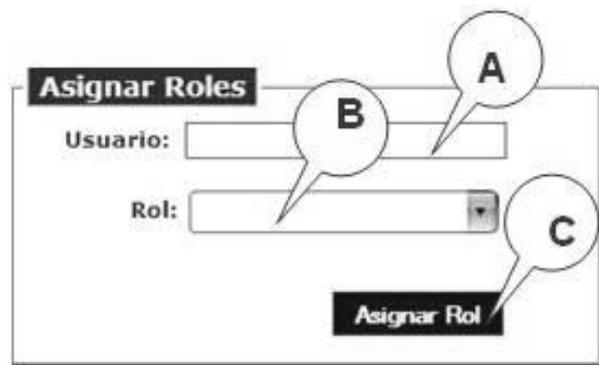
3. El usuario escribe la función del Web Service a la que le quiere hacer reportes en forma de gráfico estadístico [B]. En caso de que no escriba una, ver CA3.	
4. El usuario escribe una descripción breve de lo que mostrará el gráfico [C]. En caso de que no escriba una, ver CA4.	
5. El usuario adiciona [D] los parámetros de la función especificada.	
6. El usuario escribe los valores de los parámetros en su respectivo orden [F], [G].	
7. El usuario selecciona el botón Crear Gráfico [H]. En caso de que no desee mostrar los campos llenados, el usuario selecciona el botón cancelar [I], ver CA5.	
Cursos Alternos	
CA1	El sistema muestra un mensaje de error [Pantalla 3].
CA2	El sistema muestra un mensaje de error [Pantalla 2].
CA3	El sistema muestra un mensaje de error [Pantalla 4].
CA4	El sistema muestra un mensaje de error [Pantalla 5].
CA5	El sistema blanquea los campos que se habían llenado.
Poscondiciones	Se guarda el nuevo gráfico que genera la consulta.

Tabla 2. 20 Descripción de CU_ Gestionar Roles.

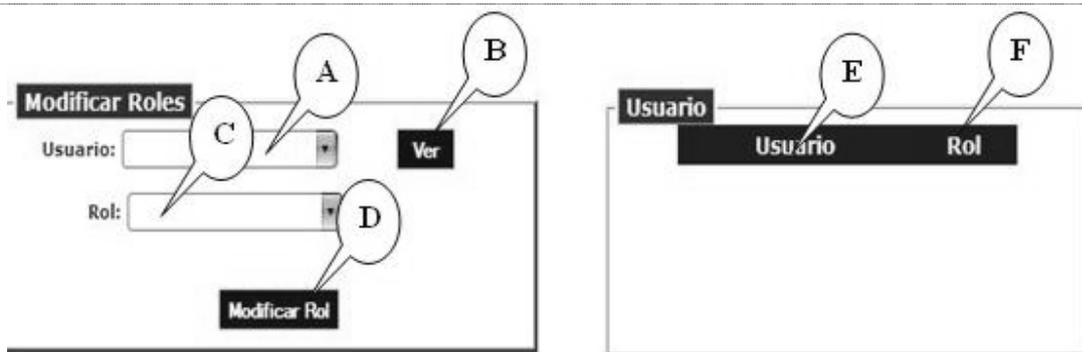
Caso de Uso	Gestionar _ roles
Actores	Administrador
Resumen	El caso de uso inicia cuando el usuario administrador desea modificar, crear o asignar un rol.
Propósito	Asignar roles a los usuarios del sistema.
Referencias	R12
Precondiciones	El usuario debe estar autenticado satisfactoriamente como administrador del sistema.



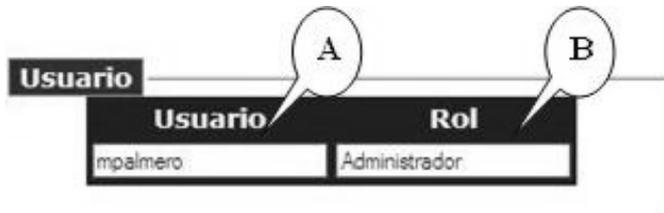
Pantalla 1



Pantalla 2



Pantalla 3



Pantalla 4

Flujo Normal de Eventos

Acción del Actor	Respuesta del Sistema
	1. El sistema muestra la página de opciones. [Pantalla 1]
3. Si el usuario escoge: a) Asignar rol, ver Sección: Asignar rol. b) Modificar rol, ver Sección: Modificar rol.	
Sección: Asignar rol.	
	1. El sistema muestra la interfaz para Asignar los roles y carga los roles que hay en el sistema. [Pantalla 2]

2. El usuario escribe el nombre del usuario (del dominio UCI) al que le va a asignar un determinado rol [A], y el rol que le corresponde [B]. Pulsa el botón asignar rol [C]	3. El sistema guarda los datos.
Sección: Modificar rol.	
	1. El sistema muestra la interfaz para modificar los roles y carga los usuarios que hay en el sistema. [Pantalla 3]
2. El actor selecciona el usuario al que desea modificar el rol o al que quiere ver que rol que tiene asignado. Si es : a) Ver el Rol que tiene asignado, ver Sección 2: Ver Rol. b) modificar el rol de un usuario, ver Sección 2: Modificar Rol.	3. El sistema muestra en [A] el usuario seleccionado.
Sección 2: Ver Rol.	
4. El actor pulsa el botón Ver [B].	5. El sistema muestra [Pantalla 4] el usuario seleccionado [A] y el Rol que tiene asignado [B].
Sección 2: Modificar Rol.	
1. El usuario escoge el nuevo rol que le desea asignar al usuario [C].	2. El sistema muestra en [C] el rol seleccionado.
3. El usuario pulsa el botón Modificar Rol [D].	4. El sistema carga el nuevo rol del usuario.
Poscondiciones	Se ha gestionado un rol para un usuario.

Tabla 2. 21 Descripción de CU_ Crear Mapa.

Caso de Uso	Crear_ Mapa
Actores	Usuario Avanzado.
Resumen	El caso de uso inicia cuando el usuario avanzado o el administrador

	deciden crear una nueva consulta para generar un mapa temático.
Propósito	Crear una nueva consulta.
Referencias	R11
Precondiciones	El usuario avanzado debe haber escogido la opción de crear mapa.
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
	1. El sistema muestra la página de crear mapa
2. El usuario escribe la URL del web service al que se quiere conectar [A]. En caso de que no escriba una, el sistema muestra un mensaje de error.	
3. El usuario escribe la función del web service a la que le quiere hacer reportes en forma de mapa temático.	
4. El usuario escribe una descripción breve de lo que mostrará el mapa. En caso de que no lo haga el sistema muestra un mensaje.	
5. El usuario agrega y escribe los parámetros de la función especificada.	
6. El usuario escribe los valores de los parámetros en su respectivo orden.	

Tabla 2. 22 Descripción de CU_ Buscar Mapa.

Caso de Uso	Buscar _ Mapa.
Actores	Usuario Avanzado
Resumen	El caso de uso inicia cuando el usuario desea buscar un mapa temático que haya sido creado con anterioridad.
Propósito	Buscar un mapa temático.
Referencias	R9

Precondiciones	Debe existir al menos un mapa creado con anterioridad.
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
	1. El sistema muestra la página de opciones.
2. El usuario púlsale botón Buscar Mapa	3. El sistema muestra la página Buscar Mapa.
4. El usuario selecciona el mapa de la lista de mapas existente.	5. El sistema muestra el mapa seleccionado en una tabla.
Poscondiciones	Se muestra el mapa buscado en una tabla.

Tabla 2. 23 Descripción de CU_ Gestionar Mapa.

Caso de Uso	Gestionar _ mapa.
Actores	Usuario avanzado
Resumen	El caso de uso inicia cuando el usuario autenticado especifica al sistema si desea, eliminar o modificar un mapa temático.
Propósito	Gestionar la información en de mapas.
Referencias	R8
Precondiciones	Debe existir un mapa creado en el sistema.
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
	1. El sistema muestra la página de Opciones.
2. El usuario, busca un mapa creado anteriormente [Ver CU_Buscar Mapa] y selecciona la acción que quiere realizar pulsando el botón eliminar o modificar	3. Si es: a) Eliminar, ver Sección Eliminar. b) Modificar, ver Sección Modificar
Sección: Eliminar.	
	1. El sistema elimina de la lista de mapas el mapa mostrado.
Sección: Modificar.	
	1. El sistema muestra la tabla con los campos de datos del mapa.

2. El usuario cambia el campo que desea modificar.	3. El sistema muestra la tabla con el (los) campo (s) modificado (s)
4. El usuario puede mostrar el nuevo mapa pulsando el botón graficar.	5. El sistema muestra el nuevo mapa modificado.
Cursos Alternos	
Poscondiciones	Se ha gestionado el mapa seleccionado.

2.4 Conclusiones.

En este capítulo se hizo una descripción de la propuesta de solución. Se comenzó a profundizar en el desarrollo de la propuesta de solución, obteniéndose una lista de las funcionalidades que debe tener el módulo que se va a desarrollar y la modelación de del mismo en términos de casos de uso de sistema. Estos fueron representados 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 que constituye la propuesta de solución, cumpliendo con todos los requerimientos y las funcionalidades que se consideraron en este capítulo.

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA.

3.1 Introducción.

En este capítulo se obtendrá el diseño de la propuesta de solución, creando los artefactos que ayudan a la construcción del sistema propuesto. Se especificarán los principios de diseño gráfico, los estándares de la interfaz de aplicación, así como una concepción general de la ayuda y el tratamiento de excepciones para finalmente obtener con esto el diseño de la interfaz gráfica para el módulo.

3.2 Modelo de análisis.

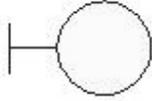
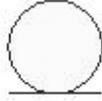
Mediante el flujo de trabajo de análisis se profundiza en el dominio de la aplicación. Se basa en un modelo de objetos conceptuales, que se denomina modelo de análisis y permite una mayor comprensión del problema para modelar la solución. Además, ofrece un mayor poder expresivo y una mayor formalización debido a que 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.

La importancia de este flujo reside en el hecho de que, como se podrá ver, permite un acercamiento a la lógica de cómo se resolverá el problema, profundizando en la comprensión de los requisitos y además, da la primera vista del sistema por dentro. Se puede plantear que es donde se encuentra la forma del sistema.

3.2.1 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 (asociación, agregación/composición, generalización/especialización, asociación) estos son mucho más conceptuales que sus contrapartidas de diseño e implementación. Siempre encajan en uno de estos tres estereotipos: [Larman, 2004]

Tabla 3. 1 Clases del Análisis.

Nombre de la clase	Características	Prototipo
Clase Interfaz	Modelan la interacción entre el sistema y sus actores.	
Clase Control	Coordinan la realización de uno o unos pocos casos de uso coordinando las actividades de los objetos que implementan la funcionalidad del caso de uso.	
Clase Entidad	Modelan información que posee larga vida y que es a menudo persistente.	

En el caso particular de este sistema estas clases tienen una peculiaridad y es el hecho de que este módulo cuenta con una base de datos muy pequeña y el resto de las bases de datos con las que cuenta no son fijas, por lo tanto no cuenta con muchas clases persistentes⁷, solo las necesarias para trabajar con los mapas y los gráficos que se generen. Los diagramas de clases del análisis (DCA), presentados por casos de uso, no son 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. Ver Anexo II.

3.3 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”.

[Larman, 2004]

La importancia del diseño del software se puede describir con una sola palabra: calidad. El diseño es el lugar en donde se fomenta la calidad en la ingeniería del software. Proporciona las representaciones del software que se pueden evaluar en cuanto a calidad y es la única forma de convertir exactamente los requisitos de un cliente en un producto o sistema de software finalizado. El diseño del software sirve como fundamento para todos los pasos siguientes del soporte del software y de la ingeniería del software. Sin un diseño, corremos el riesgo de construir un sistema inestable, un sistema que fallará cuando se lleven a cabo cambios; un sistema que puede resultar difícil de comprobar.

⁷ Son las clases que pasan como tablas a la base de datos.

Para lograr esto existen los patrones de diseño. Un patrón de diseño no es más que un par problema – solución que recibe un nombre en dependencia del problema que resuelve. Como por ejemplo: ¿Cuál es el principio general para asignar responsabilidades a los objetos? ¿Quién debería ser el responsable de conocer la información? o ¿Quién debería ser el responsable de la creación de una nueva instancia de alguna clase? Entre otras responsabilidades.

3.3.1 Principios de diseño gráfico.

El diseño de la interfaz de cualquier aplicación es un aspecto a tener en cuenta en cualquier sistema informático, requiere tacto y calidad 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 por qué tener un alto grado en el conocimiento informático, independientemente del lugar o el área a la que pertenezca. Una aplicación debe tener una interfaz con buena usabilidad, que tenga un buen diseño gráfico y que le posibilite al usuario alcanzar el objetivo por el cual está trabajando con nuestro sistema con un mínimo esfuerzo.

Aunque para este trabajo 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 página contiene el nombre del módulo, de forma que el cliente sepa donde está ubicado dentro del sistema.

3.3.2 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, en dependencia del rol que tenga asignado el usuario tiene que introducir algunos datos dato o seleccionarlos, por lo que se debe garantizar que siempre el usuario escoja o llene todos los campos que se le piden.

El hecho de que el usuario tenga que llenar otros campos, es decir, no solo los escoge, no facilita el trabajo a la hora de tratar los errores de entrada de datos al sistema. Es por esto que hay 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 o no llenara algún campo, la aplicación le emita un mensaje de alerta para que rectifique y así se garantiza que el sistema funcione y sus objetivos sean cumplidos.

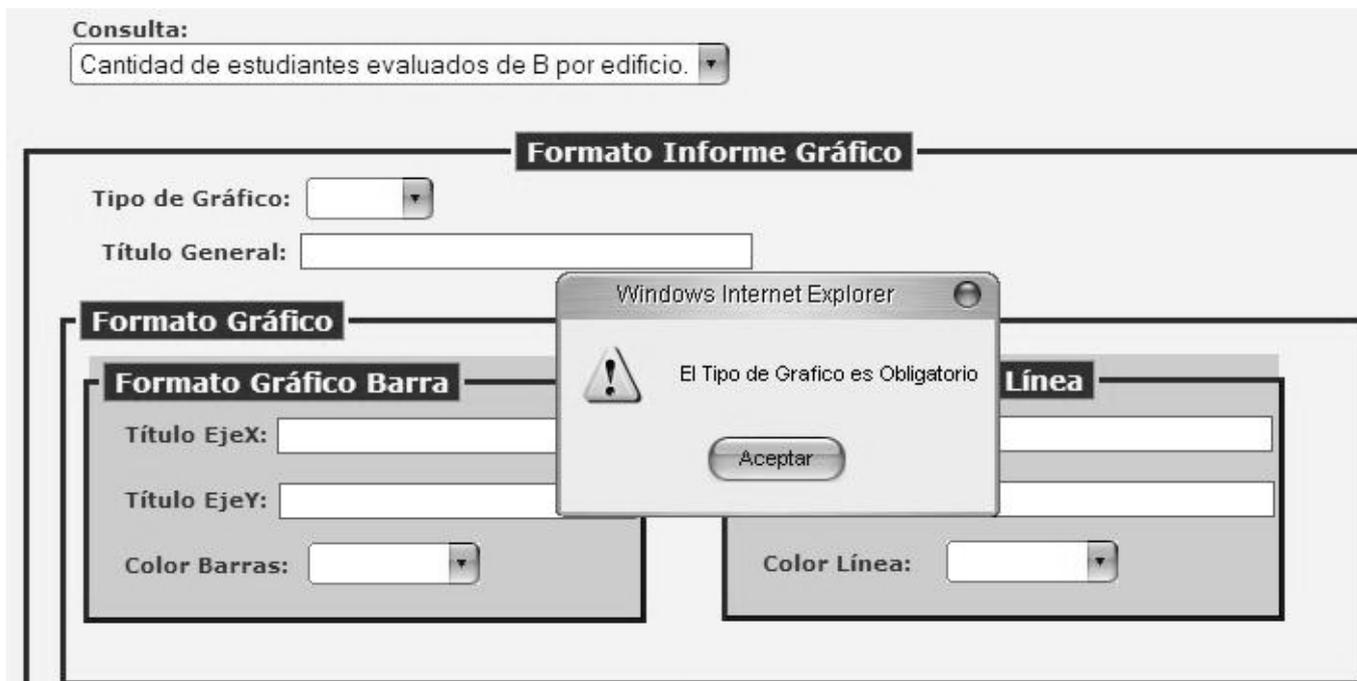


Fig.3. 1 Tratamiento de errores.

3.3.3 Diagrama de Clases de Diseño.

Los diagramas de clases⁸ del diseño son diagramas de estructura estática que muestran las clases del sistema y sus interrelaciones. Estos son considerados como el pilar básico del modelado con UML porque son utilizados, tanto para mostrar lo que el sistema puede hacer (análisis) como para mostrar como puede ser construido (diseño). Puede decirse entonces que 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 arquitectura⁹ de tres capas, que se han dividido además por paquetes para su mejor comprensión. Ver Fig. 4.2. La arquitectura en capas o multinivel, como también es conocida, es muy usada en sistemas informáticos actualmente, principalmente porque una de sus ventajas está en el hecho de que el desarrollo se puede llevar a cabo en varios niveles (como bien lo especifica su nombre) y en caso de algún cambio sólo se ataca al nivel requerido sin tener que revisar entre código mezclado. También permite el diseño de arquitecturas escalables, es decir, que pueden ampliarse con facilidad en caso de que las necesidades aumenten.

⁸ Conjunto de clases, interfaces y colaboraciones, así como sus relaciones.

⁹ Según Pressman, la arquitectura no es más que una descripción de los subsistemas y los componentes de un sistema informático y las relaciones entre ellos.

La capa de presentación es la que ve el usuario, le presenta el sistema, le comunica la información y captura la información dando un mínimo de proceso por ello se encuentran en esta capa todas las páginas que interactúan con el usuario y la lógica del negocio de cada una de estas páginas. Esta capa se comunica únicamente con la capa de negocio.

La capa de Lógica del Negocio (o de Negocio) es donde residen los programas que se ejecutan, recibiendo las peticiones del usuario y enviando las respuestas tras el proceso. Se comunica con la con la capa de presentación, para recibir las solicitudes y presentar los resultados, y con la capa de acceso a datos, para hacer las solicitudes al Web service para almacenar o recuperar datos de él. Esta capa está dividida por su complejidad en tres paquetes:

- El paquete Accesibilidad. Que contiene los componentes que darán la posibilidad al usuario de acceder a las páginas a las que se le está permitido según el rol que se les haya asignado.
- El paquete Gráfico es que se encarga de gestionar toda la información necesaria para realizar los gráficos.
- El paquete Mapa es que se encarga de gestionar toda la información necesaria para realizar los mapas.

La capa Acceso a Datos es la encargada de conectarse a las diferentes bases de datos y extraer de ellas las informaciones requeridas. Es la capa donde residen los datos. Está formada por uno o más gestor de bases de datos que realiza todo el almacenamiento de datos y reciben solicitudes de almacenamiento o recuperación de información desde la capa de negocio.

Se confeccionó un diagrama de clases para cada caso de uso, en el Anexo II se muestran el resto de los diagramas de clases del diseño.

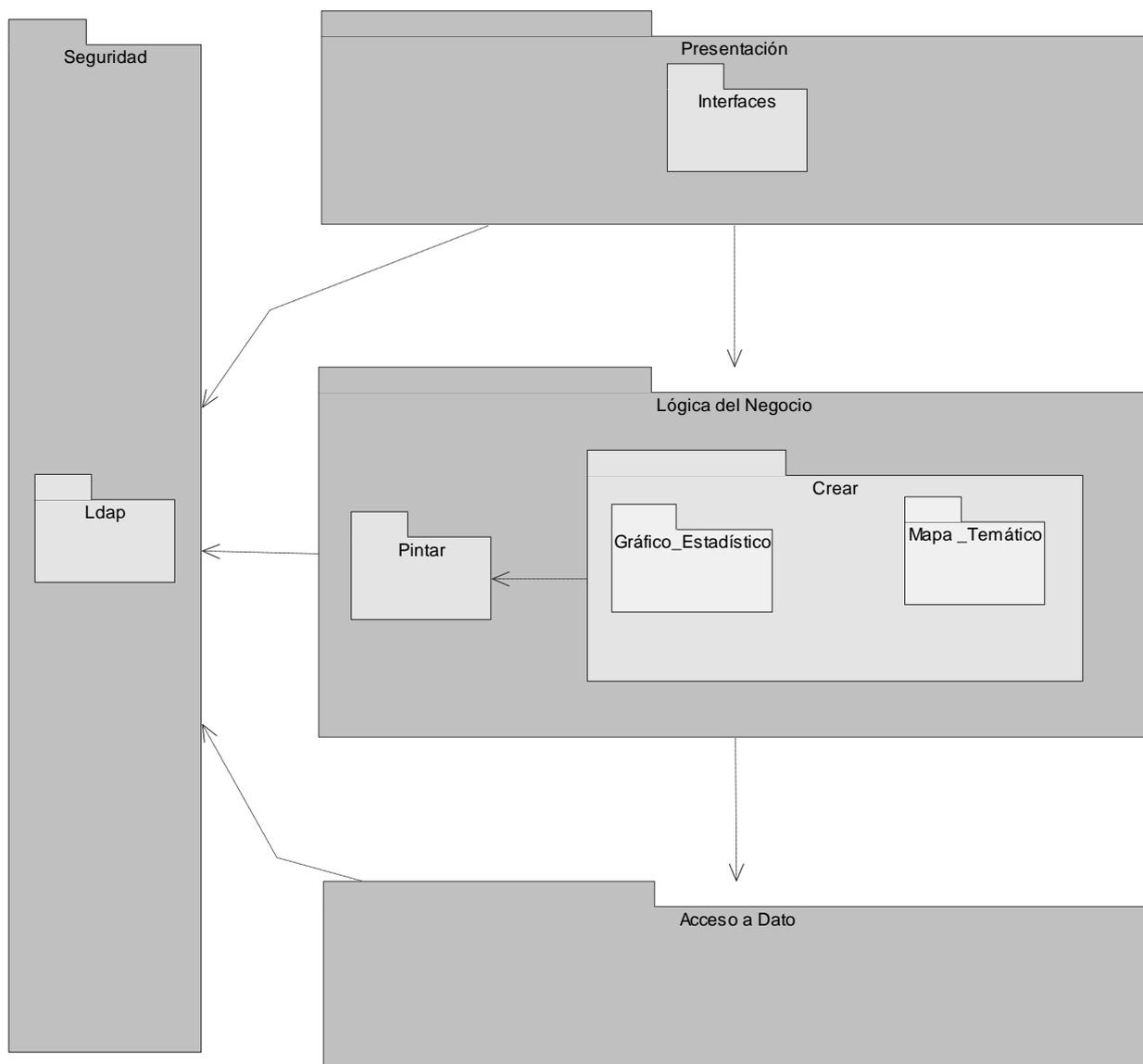


Fig.3. 2 Arquitectura del sistema.

Tabla 3. 2 Diagrama de clases persistentes.

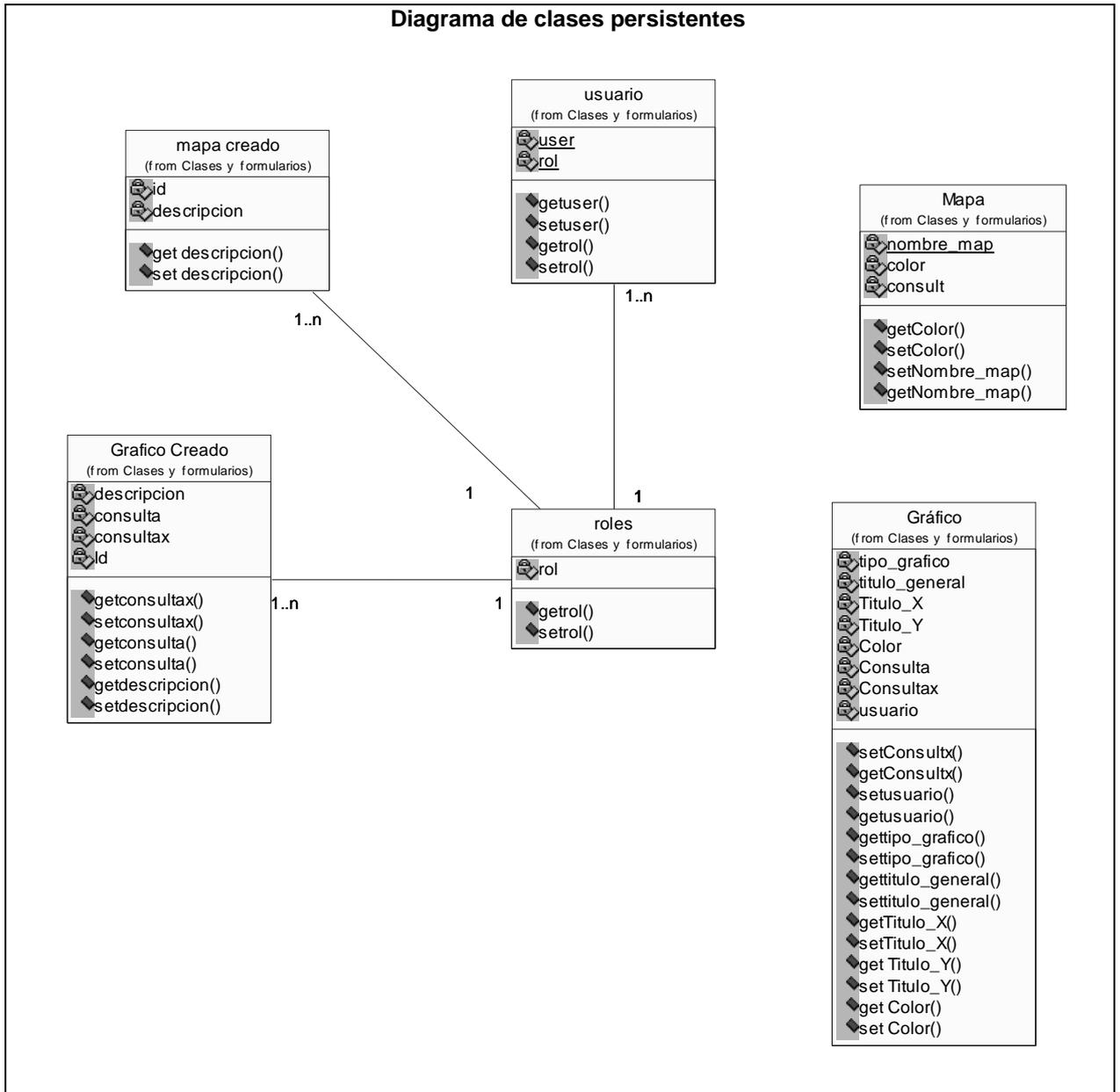


Tabla 3. 3 Diagrama del diseño.

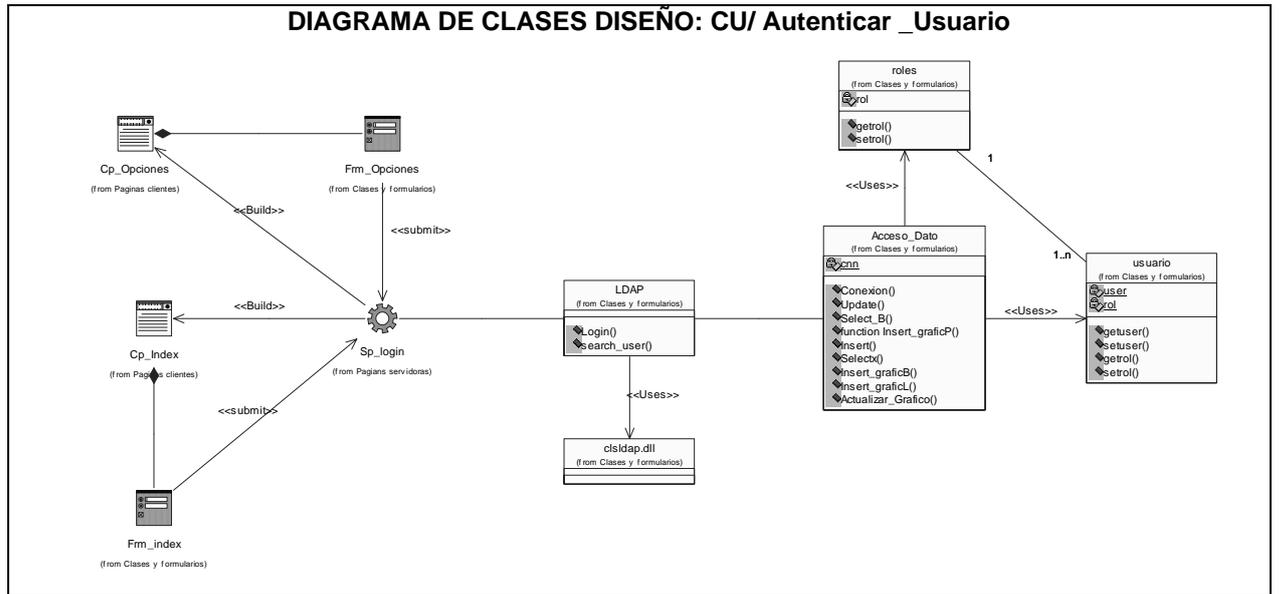


Tabla 3. 4 Diagrama de clases del Diseño.

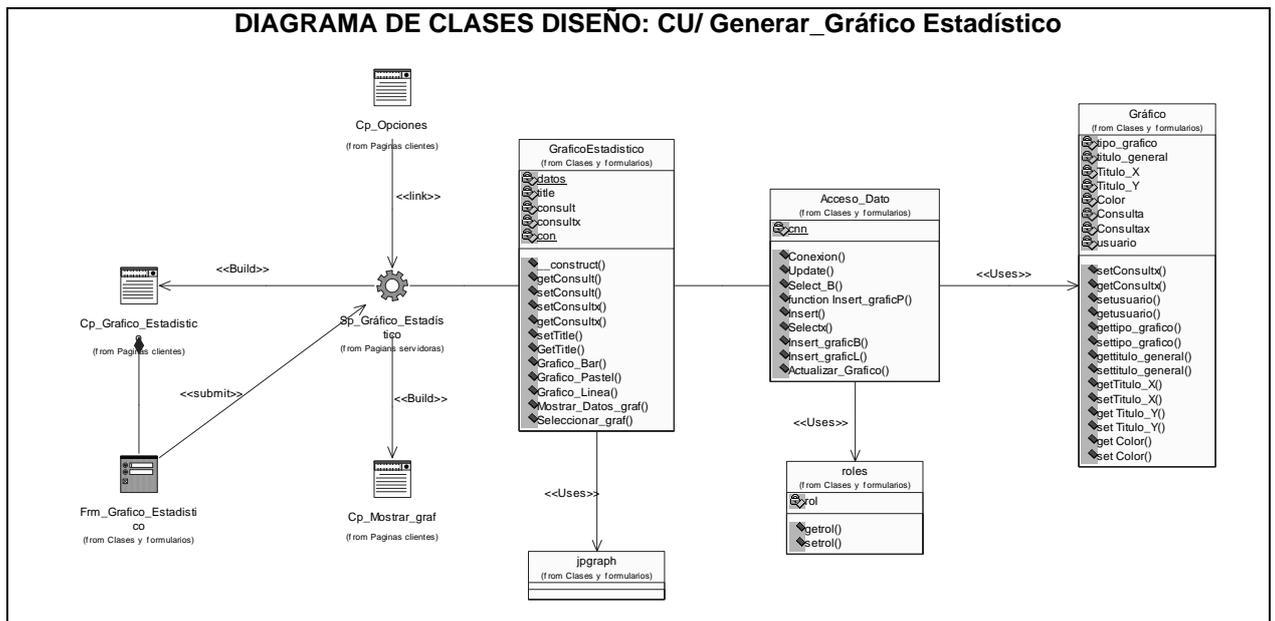


Tabla 3. 5 Diagrama de clases del Diseño.

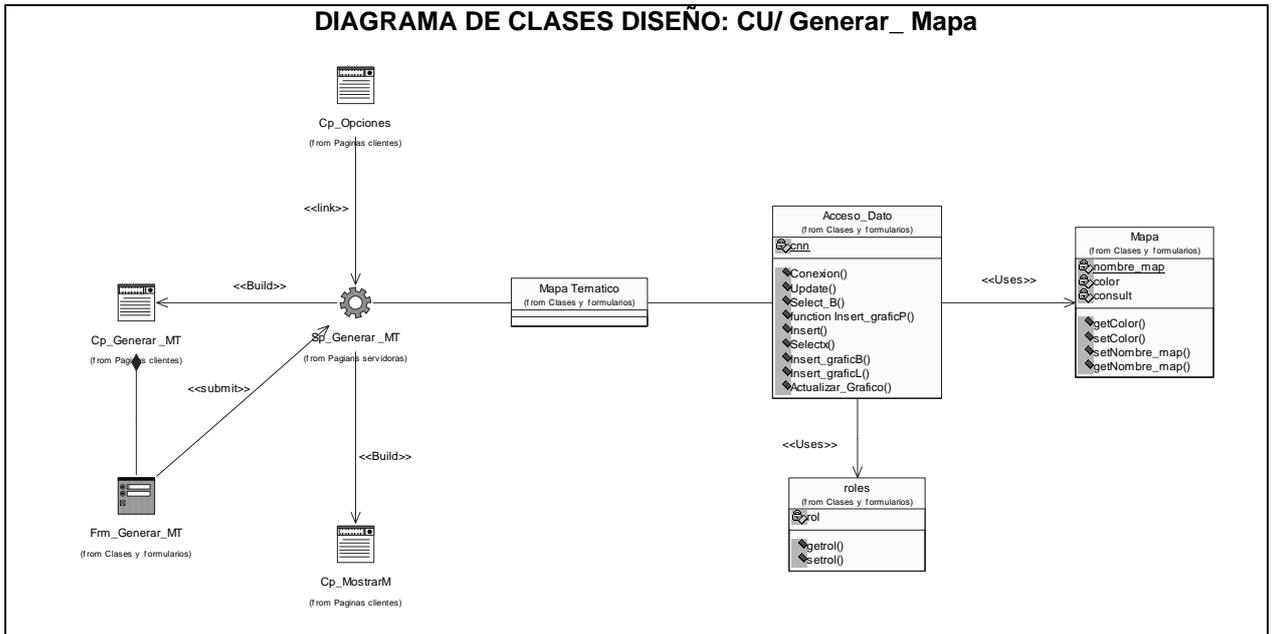


Tabla 3. 6 Diagrama de clases del Diseño.

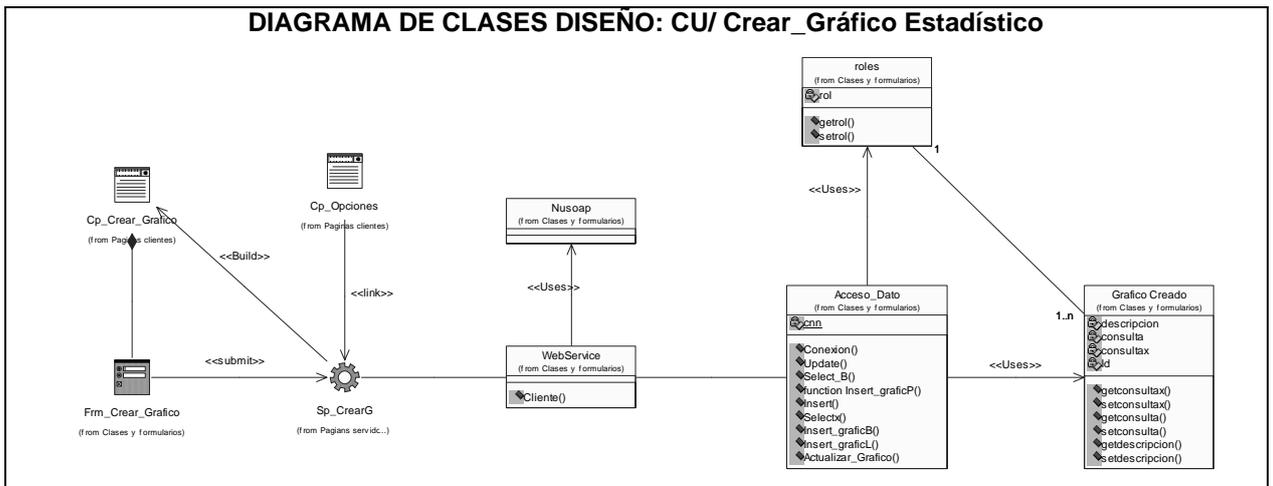


Tabla 3. 7 Diagrama de clases del Diseño.

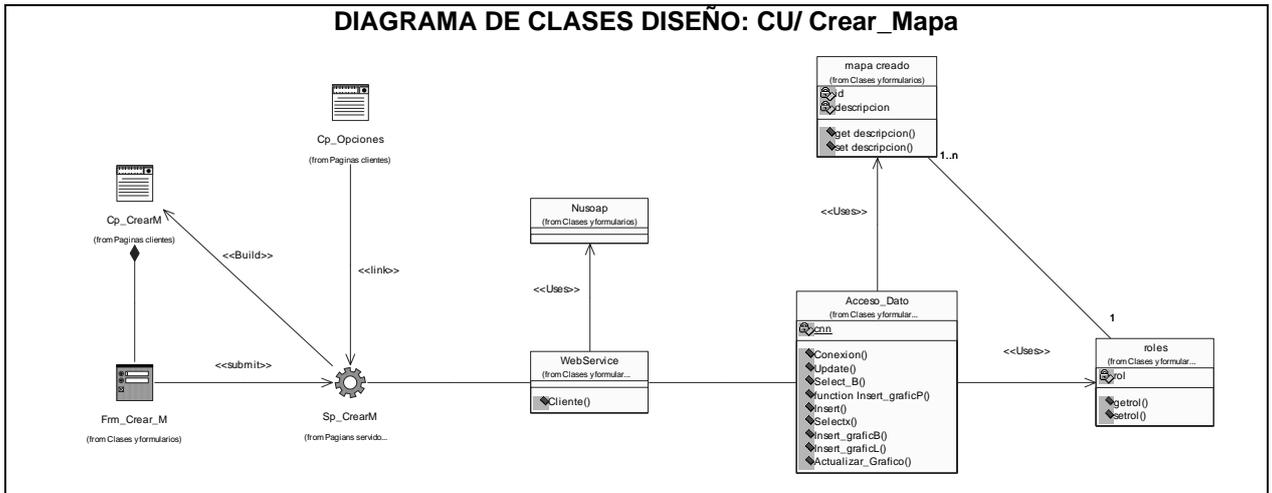
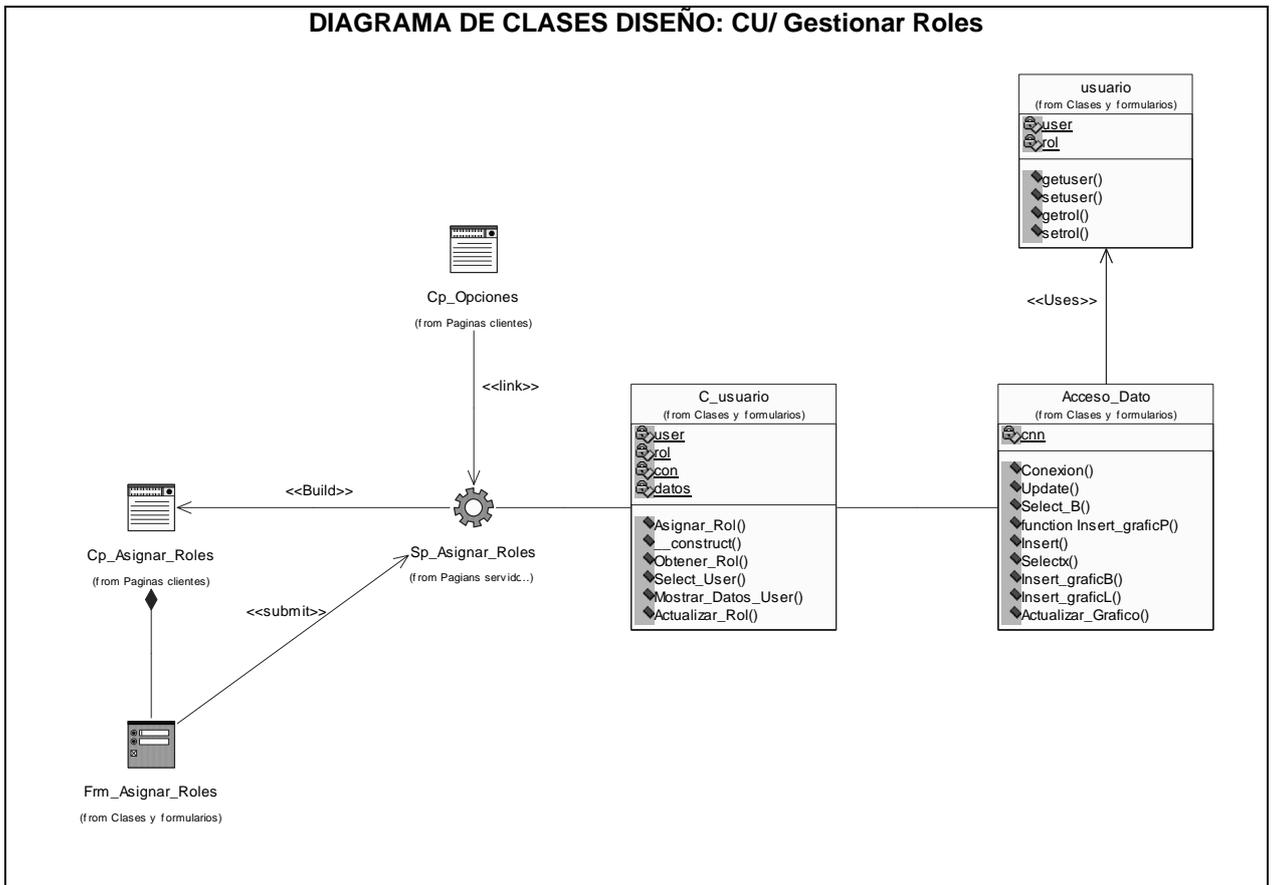


Tabla 3. 8 Diagrama de clases del Diseño.



En el Anexo III se podrán ver resto de los diagramas de clases del diseño para los casos de usos secundarios.

3.3.4 Descripción de las clases.

A continuación se presentará la descripción de las clases.

Tabla 3. 9 Clase Gráfico _ estadístico.

Nombre: Gráfico _ estadístico	
Tipo de clase: controladora (lógica del negocio)	
Atributo	Tipo
\$title	String
\$consult	String
\$con	String
\$datos	bool
\$consultx	string
responsabilidad:	
Nombre:	__construct()
Descripción:	Constructor de la clase.
Nombre:	Grafico_Bar(\$titulo_general,\$Titulo_X,\$Titulo_Y,\$Color_Barras,\$consult,\$consultx)
Descripción:	Se grafica el grafico de barra.
Nombre:	Grafico_Pastel(\$titulo_general,\$consult,\$consultx)
Descripción:	Se grafica el grafico de pastel
Nombre:	Grafico_Linea(\$titulo_general,\$Titulo_XL,\$Titulo_YL,\$Color_Linea,\$consult,\$consultx)
Descripción:	Se grafica el grafico de línea.
Nombre:	Insert(\$query)
Descripción:	Se inserta el resultado de una consulta en la base dato.
Nombre:	Select_B(\$consulta)
Descripción:	Retorna el resultado de una consulta.
Nombre:	Selectx(\$consultax)
Descripción:	Retorna el resultado de una consulta.
Nombre:	Insert_graficB(\$tipo_grafico,\$titulo_general,\$Titulo_X,\$Titulo_Y,\$Color_Barras,\$consulta,\$consultax)
Descripción:	Inserta el grafico de Barra.
Nombre:	Insert_graficL (\$tipo_grafico, \$titulo_general, \$Titulo_XL, \$Titulo_YL, \$Color_Linea,\$consulta,\$consultax)
Descripción:	Inserta el grafico de linea
Nombre:	Insert_graficP (\$tipo_grafico, \$titulo_general,\$consulta,\$consultax)
Descripción:	Inserta el grafico linea
Nombre:	Seleccionar_graf()
Descripción:	Retorna los graficos por el titulo general
Nombre:	Mostrar_Datos_graf(\$nombre)
Descripción:	Muestra los datos del grafico buscado.
Nombre:	Actualizar_Grafico(\$Tipo_Grafico,\$Titulo_General,\$Titulo_X,\$Titulo_Y)
Descripción:	Actualiza los datos del grafico.

Tabla 3. 10 Clase Acceso_Datos.

Nombre: Acceso_Datos	
Tipo de clase: controladora (acceso a dato)	
Atributo	Tipo
\$cnn	string

Responsabilidades:	
Nombre:	Conexion()
Descripción:	Función que establece la conexión con la BD.
Nombre:	Update(\$query)
Descripción:	Se conecta a la base datos.

Tabla 3. 11 Clase C_Usuario.

Nombre: C_Usuario	
Tipo de clase: controladora (lógica del negocio)	
Atributo	Tipo
\$user	string
\$datos	bool
\$con	string
\$rol	string
Responsabilidades:	
Nombre:	__construct(\$auser,\$arol)
Descripción:	Constructor de la clase.
Nombre:	Asignar_Rol()
Descripción:	Función que asigna un rol a un usuario.
Nombre:	Obtener_Rol()
Descripción:	Función que obtiene un rol que tiene asignado un usuario.
Nombre:	Select_User()
Descripción:	Función que Selecciona un usuario determinado.
Nombre:	Mostrar_Datos_User(\$nombre)
Descripción:	Función que muestra los datos de un usuario determinado.
Nombre:	Actualizar_Rol(\$rol,\$user)
Descripción:	Función que actualiza los roles en la base de datos.

Tabla 3. 12 Clase Grafico.

Nombre: Grafico	
Tipo de clase: entidad	
Atributo	Tipo
\$title	string
\$consult	string
\$consultx	string
Responsabilidades:	
Nombre:	getConsult()
Descripción:	Retorna la consulta
Nombre:	setConsult (\$consulta)
Descripción:	Asignar consulta
Nombre:	setConsultx (\$consultax)
Descripción:	Asignar consultax
Nombre:	getConsultx()
Descripción:	Retorna la consultax
Nombre:	setTitle (\$titulo_general)
Descripción:	Asigna el titulo
Nombre:	GetTitle ()
Descripción:	Retorna el titulo

Tabla 3. 13 Clase Mapa.

Nombre: Mapa	
Tipo de clase: entidad	
Atributo	Tipo
\$nombre_map	string
\$color	string
\$consult	string
responsabilidades:	
Nombre:	__construct()
Descripción:	Constructor de la clase
Nombre:	setNombre_map(\$nombre_map)
Descripción:	Asigna el nombre del mapa
Nombre:	getNombre_map()
Descripción:	Retorna el nombre
Nombre:	setColor(\$color)
Descripción:	Asigna el color
Nombre:	getColor()
Descripción:	Retorna el color

Tabla 3. 14 Clase WebService.

Nombre: WebService	
Tipo de clase: controladora	
responsabilidades:	
Nombre:	__construct()
Descripción:	Constructor de la clase
Nombre:	cliente(\$webservice,\$function,\$valores)
Descripción:	Retorna los valores que devuelven las funciones de los webservice

Tabla 3. 15 Clase Usuario.

Nombre: Usuario	
Tipo de clase: entidad	
Atributo	Tipo
user	string
rol	string
responsabilidades:	
Nombre:	getuser()
Descripción:	Retorna el usuario
Nombre:	Setuser()
Descripción:	Asigan usuario
Nombre:	Setrol()
Descripción:	Asigan el rol
Nombre:	Getrol()
Descripción:	Retorna el rol

3.3.5 Descripción de las tablas de la Base de Datos.

Tabla 3. 16 Usuario.

Nombre: Usuario		
Descripción: Almacena los usuarios y roles del sistema.		
Atributo	Tipo	Descripción
user	VARCHAR	Usuario del sistema
rol	INTEGER	Rol del usuario

Tabla 3. 17 Gráfico.

Nombre: Grafico.		
Descripción: Almacena los gráficos que se consultan en el sistema		
Atributo	Tipo	Descripción
id	INTEGER	Id del grafico
Tipo_grafico	VARCHAR	Tipo de grafico
Titulo_general	VARCHAR	Titulo general del grafico
tituloX	VARCHAR	Titulo por el eje X
tituloy	VARCHAR	Titulo por eje Y
color	VARCHAR	Color de las barras o de las líneas
consult;	VARCHAR	Consulta por el eje Y
consultx;	VARCHAR	Consulta por el eje X

Tabla 3. 18 Mapa.

Nombre: Mapa		
Descripción: Almacena los mapas que se consultan en el sistema.		
Atributo	Tipo	Descripción
nombre_map	VARCHAR	Nombre del mapa
color	VARCHAR	Color de la capa
consult	VARCHAR	Consulta del mapa

Tabla 3. 19 grafico creado

Nombre: grafico creado		
Descripción: Almacena los gráficos que se crean en el sistema.		
Atributo	Tipo	Descripción
descripcion	VARCHAR	Descripción del gráfico.
consulta	VARCHAR	Consulta que devuelve el webservice
consultax	VARCHAR	Consulta que devuelve el webservice en X
id	INTEGER	Id del gráfico creado.

Tabla 3. 20 mapa creado.

Nombre: mapa creado		
Descripción: Almacena los mapas que se crean en el sistema.		
Atributo	Tipo	Descripción
id	INTEGER	Id del mapa creado
descripcion	VARCHAR	Descripción del mapa
consult	VARCHAR	Consulta del mapa.

3.3.6 Diagrama De interacción.

El diagrama de interacción es un diagrama de comportamiento, dentro del vocabulario gráfico de UML, que incluye a los diagramas de colaboración y secuencia y no son más que un conjunto de objetos y sus relaciones, incluyendo los mensajes que se pueden enviar entre ellos. Por su importancia para el diseño se ha decidido trabajar con los diagramas de interacción.

A continuación se presentará el diagrama de secuencia (DS) de cada uno de los casos de uso críticos que se presentaron anteriormente y el resto de los diagramas se podrán ver en el Anexo IV.

Tabla 3. 21 Diagrama de interacción.

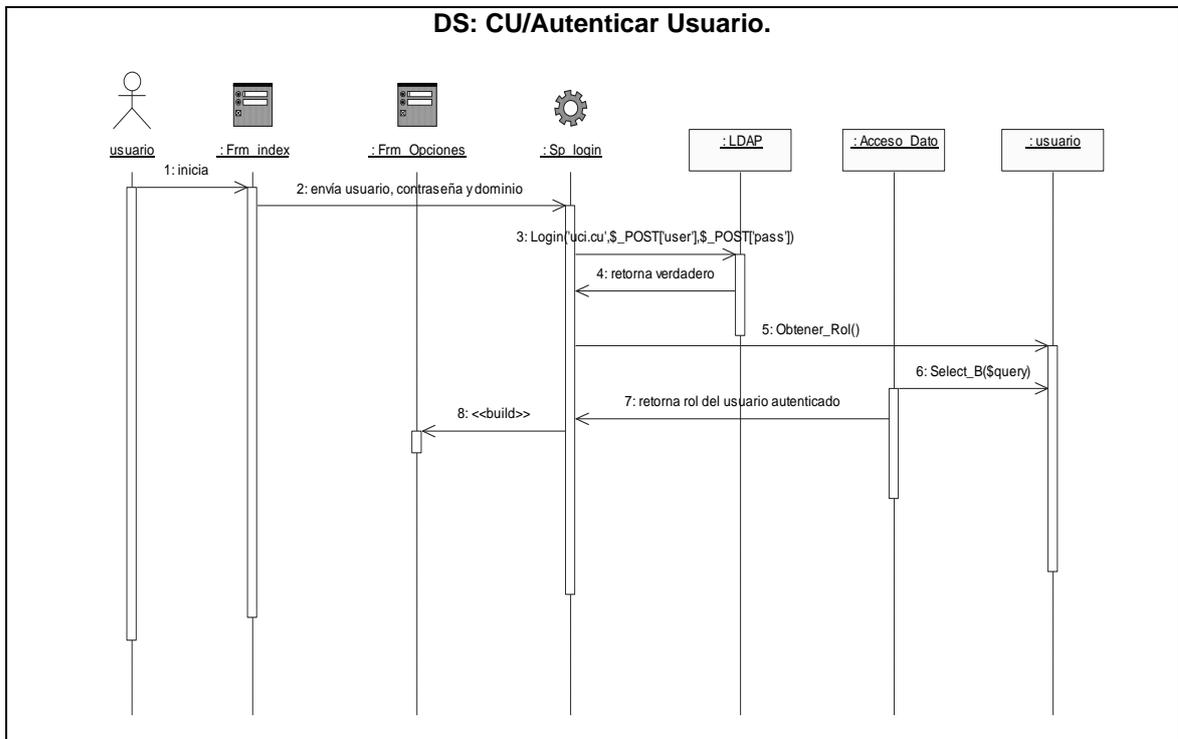


Tabla 3. 22 Diagrama de interacción.

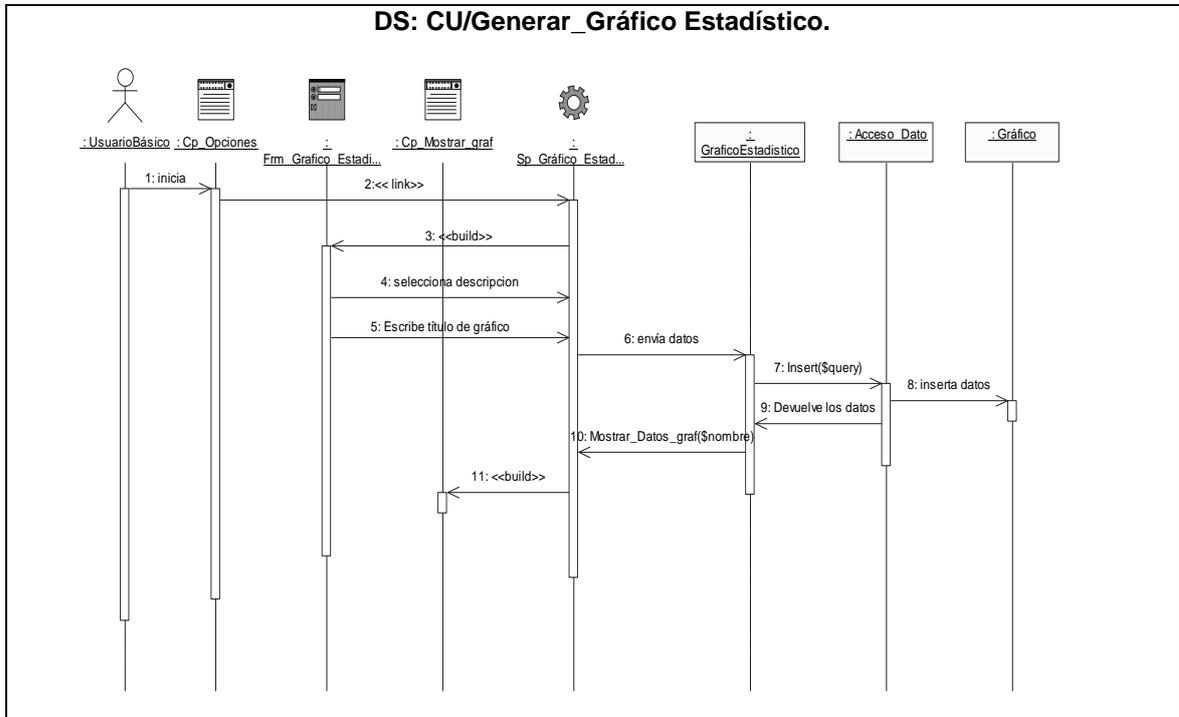


Tabla 3. 23 Diagrama de interacción.

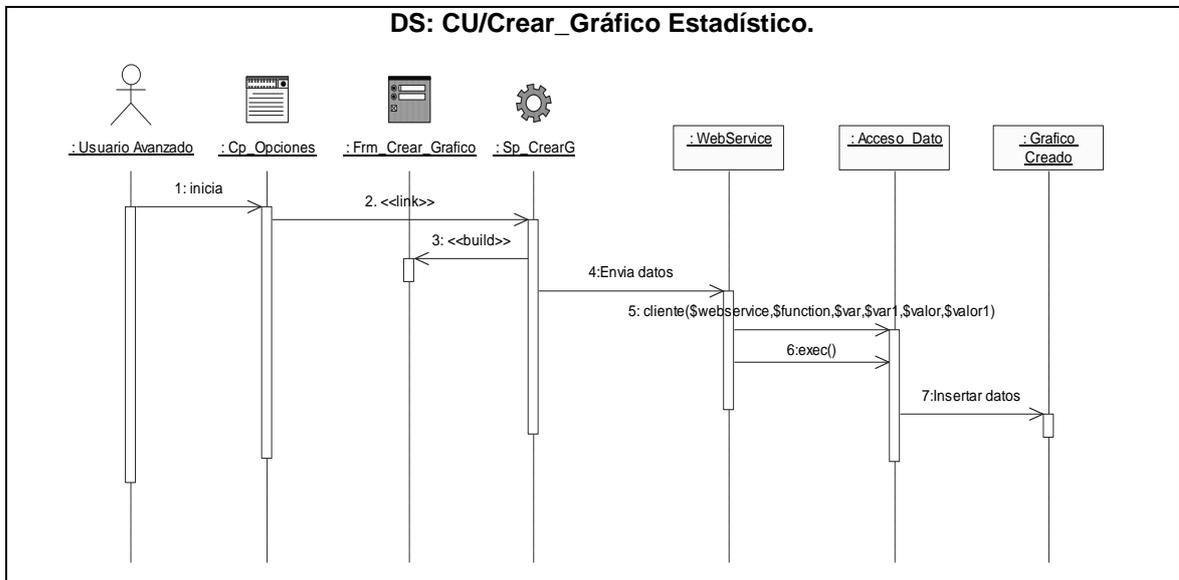


Tabla 3. 24 Diagrama de interacción.

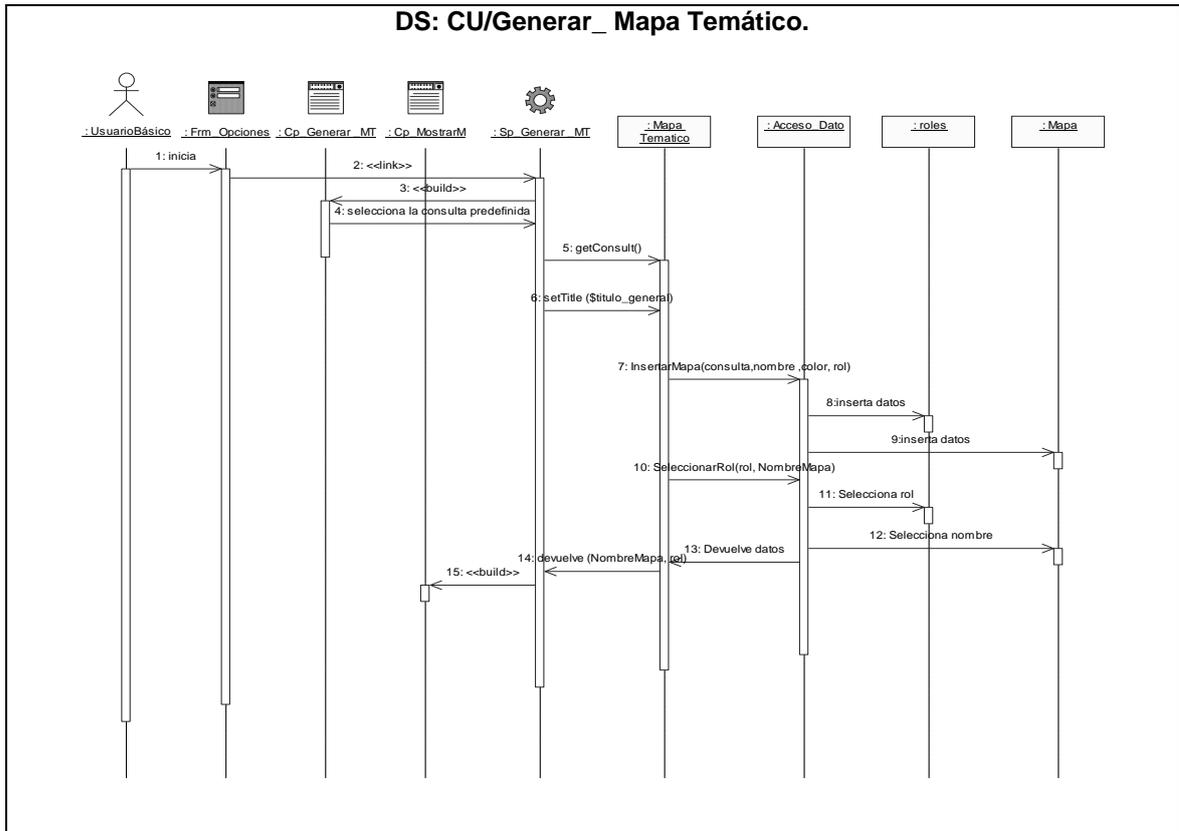


Tabla 3. 25 Diagrama de interacción.

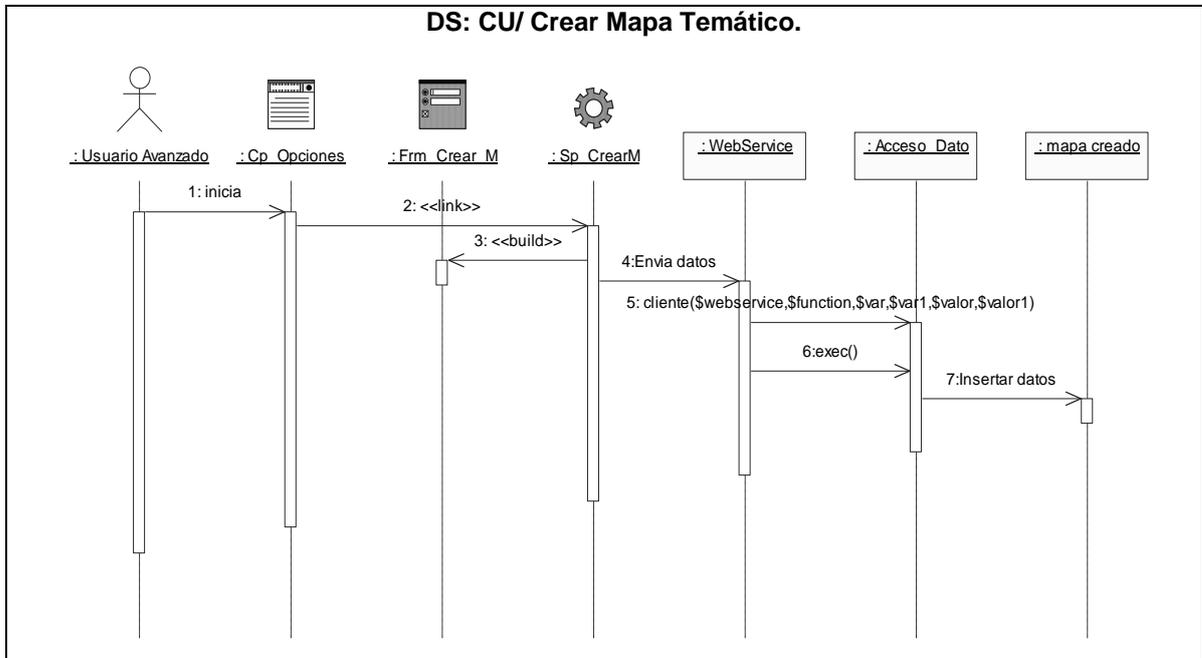
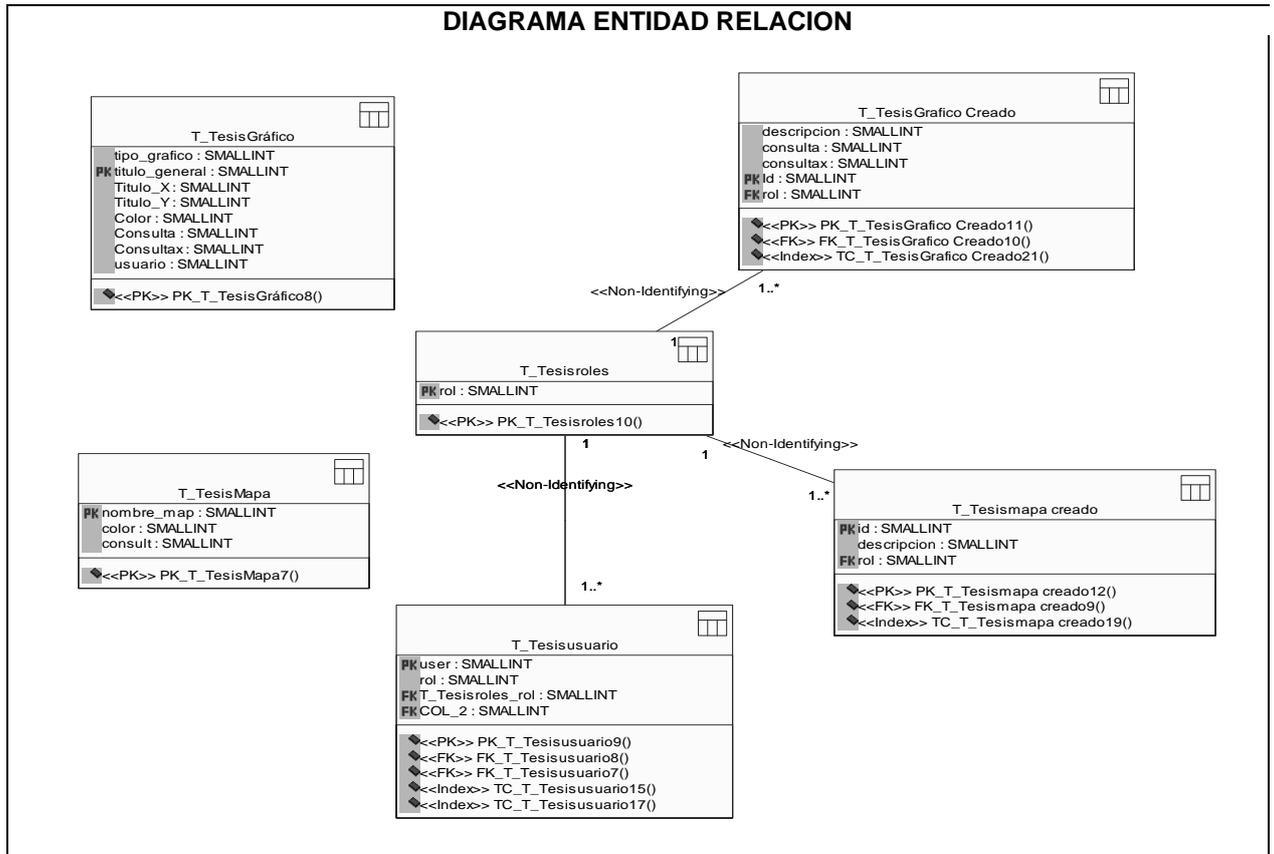


Tabla 3. 26 Diagrama Entidad Relación de la BD.



3.4 Conclusiones

En este capítulo se ha llevado a cabo el análisis y el diseño para el sistema propuesto a partir de los casos de usos del sistema. Se hizo la descripción de las clases necesarias para llegar la implementación partiendo de los artefactos obtenidos como los diagramas de clases Web y los diagramas de interacción. Se expusieron los principios para el diseño de la interfaz y la forma de tratar los errores. Además se expuso la arquitectura de la aplicación, permitiendo que de esta forma cualquier desarrollador tenga una idea de como será implementado este proyecto.

CAPÍTULO 4: IMPLEMENTACIÓN Y PRUEBA.

4.1 Introducción

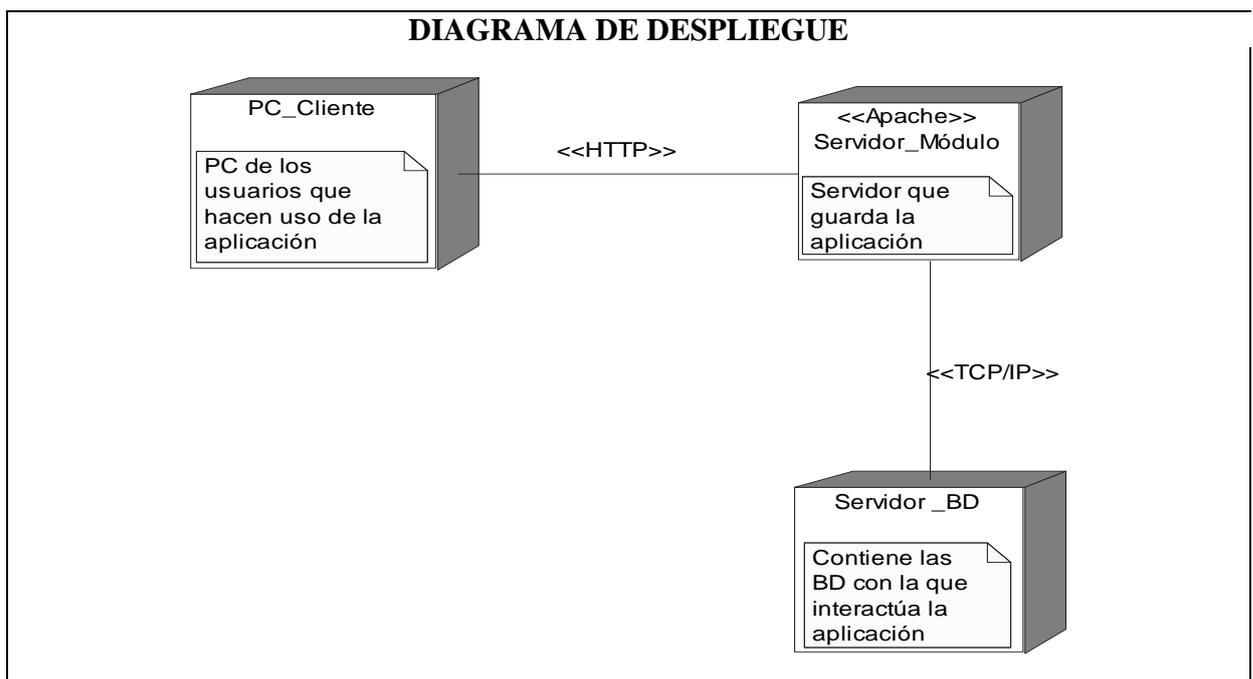
En este capítulo se realiza el modelo de despliegue y el modelo de implementación para una mejor descripción de la solución. Se presentan además los tipos de prueba a realizarle al sistema para verificar su funcionamiento.

4.2 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. [Larman, 2004]

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

Tabla 4. 1 Diagrama de despliegue.



4.3 Implementación. Modelos de implementación.

El modelo de implementación describe como los elementos del diseño, como las clases, se implementan en términos de componentes, como ficheros de código fuente, ejecutables etc.

En el subepígrafe que se presenta a continuación se especificarán los diagramas de implementación relacionados a este sistema.

4.3.1 Diagrama de componentes.

Los diagramas de componentes representan todos los tipos de elementos software que entran en la fabricación de aplicaciones informáticas y las dependencias entre ellos. Incluyen componentes de código fuente, componentes del código binario, y componentes ejecutables. El diagrama de componente hace parte de la vista física de un sistema, la cual modela la estructura de implementación de la aplicación por sí misma

Esta vista proporciona la oportunidad de establecer correspondencias entre las clases y los componentes de implementación.

Tabla 4. 2 Capa de Presentación.

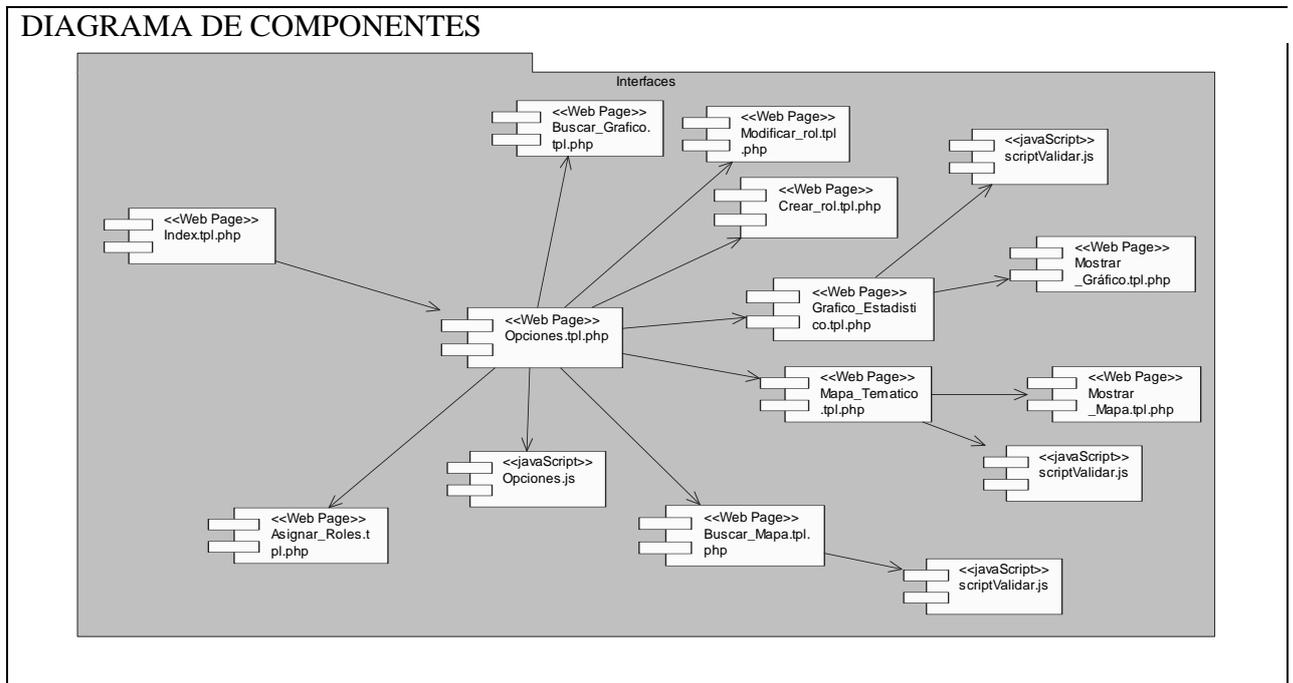


Tabla 4. 3 Capa de Acceso a Dato.

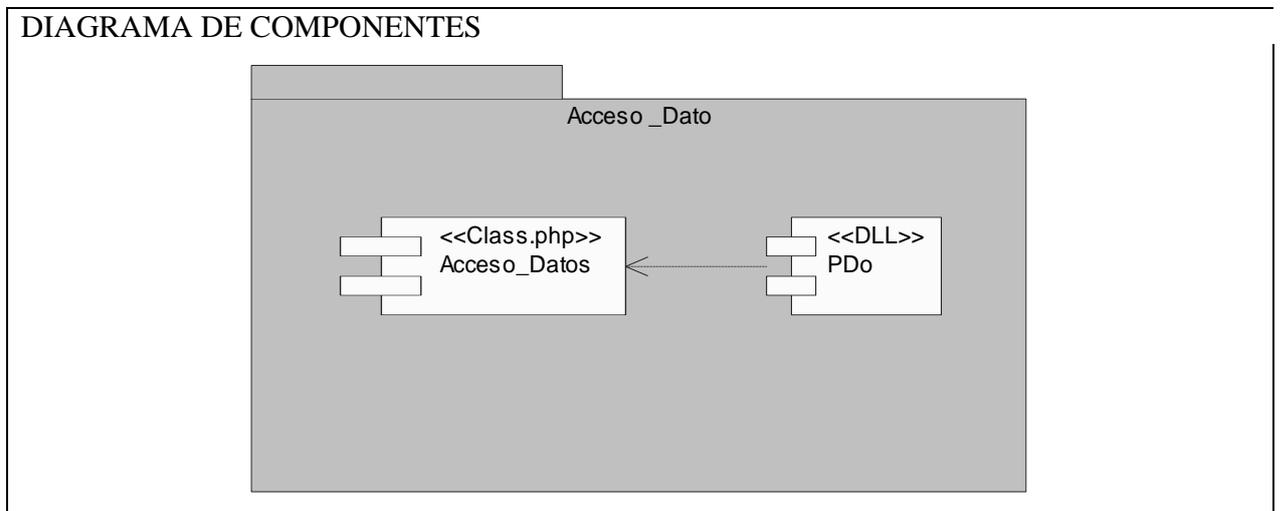


Tabla 4. 4 Seguridad.



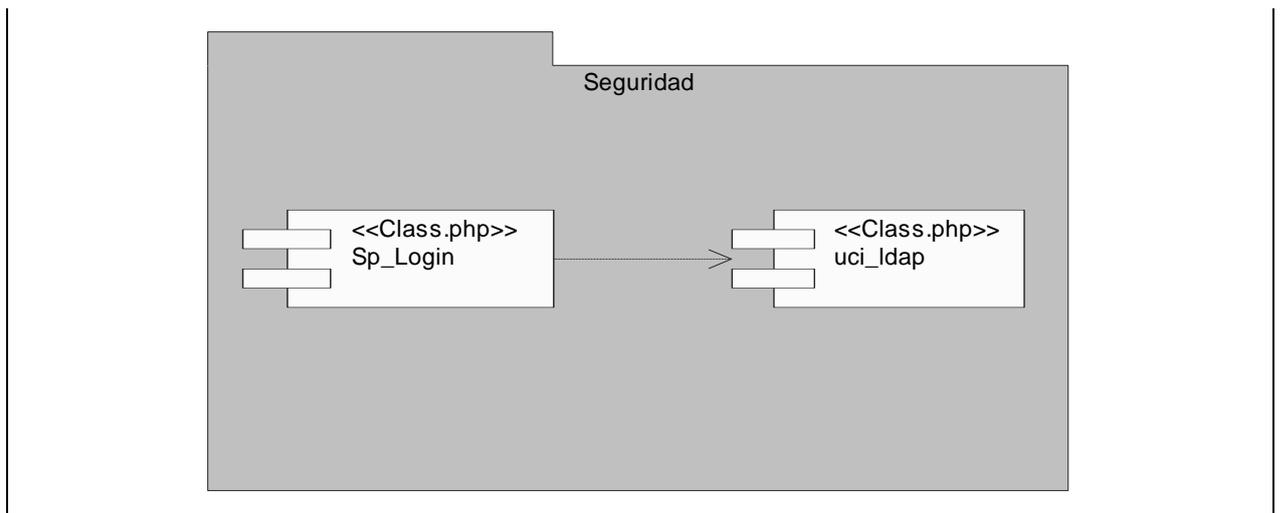
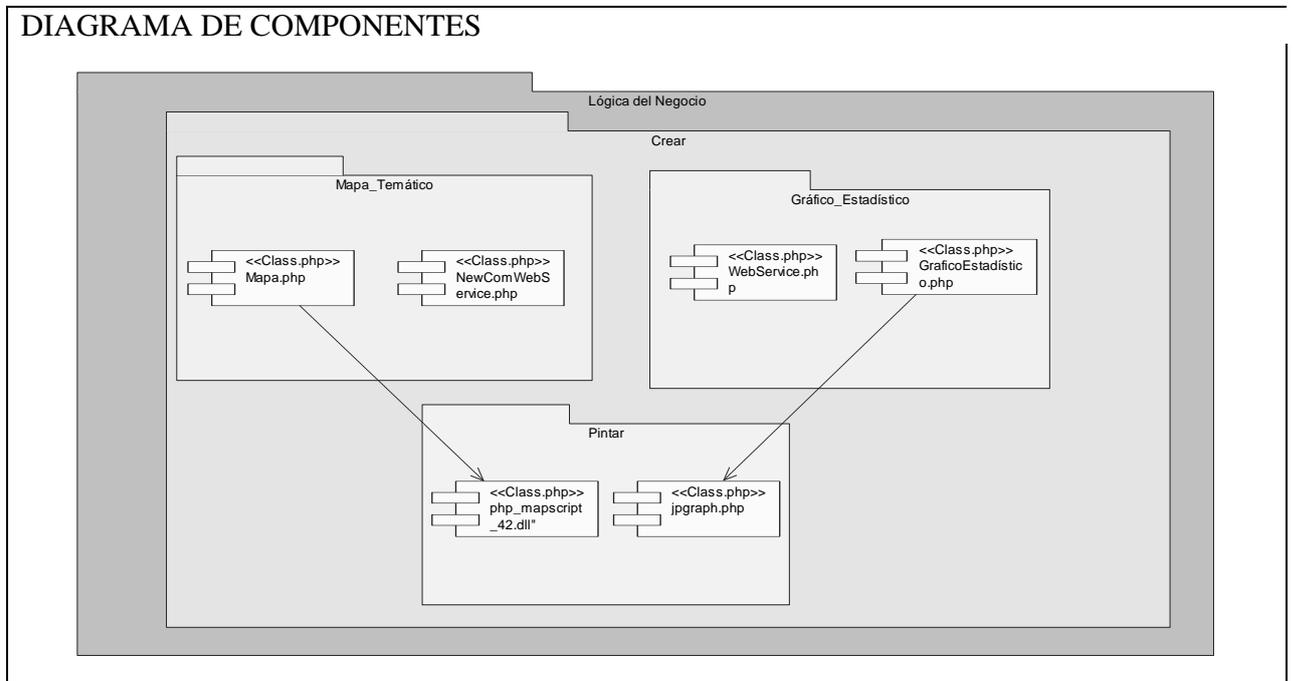


Tabla 4. 5 Capa de Lógica del Negocio.



4.4 Descripción preliminar del modelo de pruebas.

El modelo de pruebas permite principalmente, probar los componentes ejecutables del modelo de interacción. También se pueden 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.

4.4.1 Modelo de prueba

A continuación se presenta la descripción de los casos de prueba de integración, para cada caso de uso.

Tabla 4. 6 Autenticar usuario.

Entrada	Resultados	Condiciones
El usuario del sistema debe autenticarse (introducir usuario y contraseña del dominio UCI).	Si el usuario introduce correctamente los datos el sistema le muestra las páginas a las que tiene acceso según el rol que tiene asignado. Si los datos entrados son incorrectos el sistema te vuelve a pedir los datos.	1. El usuario del sistema debe pertenecer al dominio UCI.

Tabla 4. 7 Gestionar Roles.

Entrada	Resultados	Condiciones
Para que el caso de uso se ejecute el usuario autenticado debe tener el rol de administrador del sistema. Puede: 1. Eliminar un rol Selecciona uno de los usuarios que se cargan y acciona el botón eliminar 2. Cambiar un rol a un usuario determinado. Selecciona un usuario determinado y le asigna un nuevo rol.	El sistema cargan o elimina los roles que se hayan modificado.	El usuario el que se le hizo algún cambio en el rol tiene acceso a las páginas que perteneces al nuevo rol

Tabla 4. 8 Generar Gráfico estadístico.

Entrada	Resultados	Condiciones
El usuario selecciona una consulta predefinida y un título para el gráfico que desea mostrar.	Se muestra una consulta y un título para el gráfico	1. el usuario debe escoger siempre una consulta si desea generar un gráfico y ponerle su título de lo contrario el sistema muestra una alerta.

Tabla 4. 9 Generar Mapa Temático.

Entrada	Resultados	Condiciones
El usuario selecciona una consulta predefinida y un título para el mapa que desea mostrar.	Se muestra una consulta y un título para el mapa	1. el usuario debe escoger siempre una consulta si desea generar un mapa y ponerle su título de lo contrario el sistema muestra una alerta.

Tabla 4. 10 Gestionar Gráfico Estadístico

Entrada	Resultados	Condiciones
El usuario autenticado debe ser Avanzado para que pueda entrar a esta interfaz. Debe buscar el gráfico (previamente cargado por el sistema) para luego eliminarlo o modificarlo	Se muestra el gráfico cargado por el sistema en la tabla donde se pueden modificar los valores de los campos del gráfico o eliminarlo.	El usuario, si va a eliminar o modificar debe haber escogido antes un gráfico, de lo contrario el sistema muestra una excepción.

Tabla 4. 11 Gestionar Mapa Temático.

Entrada	Resultados	Condiciones
El usuario autenticado debe ser Avanzado para que pueda entrar a esta interfaz. Debe buscar el mapa (previamente cargado por el sistema) para luego eliminarlo o modificarlo	Se muestra el mapa cargado por el sistema en la tabla donde se pueden modificar los valores de los campos del mapa o eliminarlo.	El usuario, si va a eliminar o modificar debe haber escogido antes un mapa, de lo contrario el sistema muestra una excepción.

Tabla 4. 12 Configurar Gráfico.

Entrada	Resultados	Condiciones
El usuario debe escoger el tipo de gráfico que desea realizar. Debe accionar el botón graficar para mostrar el gráfico que se ha generado.	1. Según el tipo de gráfico de habilitan o deshabilitan los controles para configurar el gráfico 2. se deben llenar los campos para el color y los títulos de los ejes X y Y.	1. siempre se debe escoger algún tipo de gráfico.

Tabla 4. 13 Crear Gráfico Estadístico.

Entrada	Resultados	Condiciones
El usuario que desea crear un gráfico debe tener el Rol de avanzado o administrador. Debe hacerlo a través de una URL (Web Service)	El usuario debe escribir su propia función y agregarle los parámetros.	1. Se debe siempre llenar todos los campos. 2. El sistema reconoce si es una URL o no.

Tabla 4. 14 Crear Mapa Temático.

Entrada	Resultados	Condiciones
El usuario que desea crear un mapa debe tener el Rol de avanzado o administrador. Debe hacerlo a través de una URL (Web Service)	El usuario debe escribir su propia función y agregarle los parámetros.	1. Se debe siempre llenar todos los campos. 2. El sistema reconoce si es una URL o no.

4.5 Conclusiones

En este capítulo 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. Se hizo una disertación de los tipos de prueba que se le pueden hacer al sistema para comprobar su correcto funcionamiento.

CONCLUSIONES GENERALES

Este trabajo de diploma se propuso demostrar la necesidad de un módulo para el Sistema de Información Geográfica de la UCI, que brinde la posibilidad de mostrarle a un usuario determinado una información concreta de un área específica dentro de la institución. Estos resultados se mostrarán en forma de gráficos estadísticos y mapas temáticos.

Luego de haber cumplido con todos los objetivos propuesto para este trabajo de diploma se ha podido llegar a la conclusión de que existe una necesidad evidente de ponerlo en práctica. En la Universidad, hay una necesidad imperiosa de integrar los datos que se encuentran dispersos entre las diferentes bases de datos que usan las áreas para generar reportes.

La importancia que reviste el uso de estos reportes aumentará en calidad y percepción de cualquier información con el uso de este sistema, dada la importancia visual que transmite esta forma de representación de los datos, dando la posibilidad de hacer un estudio exhaustivo de forma sencilla y rápida. Además, la presentación de datos en forma de gráficos estadísticos y mapas temáticos brinda calidad y estética al tipo de información debido a que la primera mirada, en un documento, siempre va a parar a la imagen que se presenta, haciendo más efectivo el estudio de la información y agilizando la toma de decisiones. Esta aplicación pone en manos de los usuarios del GIS una herramienta necesaria y muy útil para la toma de decisiones según los estudios hechos en las diferentes áreas de la universidad.

Luego de un análisis riguroso de las tecnologías más usadas en la actualidad para la construcción de sistemas informáticos de este tipo, se determinó que se usaría PHP como lenguaje del lado del servidor, debido a que fue diseñado desde cero con el único fin de diseñar aplicaciones web por lo que las tareas habituales en el desarrollo de estas aplicaciones pueden hacerse con PHP de forma fácil, rápida y efectiva.. 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 de desarrollo de software idónea para llevar a cabo el proceso de desarrollo es RUP. A partir de ella se modeló el negocio propuesto partiendo de un modelo de dominio. Se especificaron los requerimientos funcionales y no funcionales con los que debía cumplir el sistema, se identificaron los actores del sistema, así como los casos de uso que debían implementarse para que el módulo cumpliera con los objetivos propuestos y se realizó una descripción detallada de los casos de uso para una mejor comprensión de los mismos.

CONCLUSIONES GENERALES

A través de UML se diseñaron los diagramas de clases del análisis y los diagramas de clases para el diseño web del sistema, entre otros artefactos. 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 realizaron los casos de pruebas pertinentes por casos de uso que permiten llegar a la conclusión de que el módulo que se presenta da solución a la situación problemática que le dio origen y que su explotación significará una mejora considerable en la calidad y eficiencia de la toma de decisiones en la Universidad de las Ciencias Informáticas.

RECOMENDACIONES

Se recomienda:

- Que cada área publique sus datos para lograr una mayor integridad dentro de las áreas de la universidad.
- Que se integre lo más rápido posible este módulo al GIS para que el mismo cuente con una mayor variedad de mapas.
- Que todas las áreas de la universidad definan que información le sería útil para la toma de decisiones, para incluir los gráficos estadísticos y/o mapas temáticos relacionados con dicha información en futuras versiones del sistema.

BIBLIOGRAFÍA

- [1]. Guevara, Roberto Tinoco. Definición y algunas aplicaciones de los sistemas de información geográfica. Monografía.com (Abril, 2007).
[Disponible en: <http://www.monografias.com/trabajos14/informageogra/informageogra.shtml>]
- [2]. Molinero, Luís. Gráficos de Datos Estadísticos en Medicina. (Abril, 2007)
[Disponible en: <http://www.seh-lelha.org/pdf/graficos.pdf>]
- [3]. Broader, Terms. Representación Gráfica. (Abril, 2007)
[Disponible en: <http://www3.unileon.es/dp/abd/tesauro/pagina/tesdocumentacion/00000135.htm>].
- [4]. Ortiz, Gabriel. Información Geográfica. (Abril, 2007)
[Disponible en: <http://recursos.gabrielortiz.com/index.asp>]
- [5.]. GEOTEC. Mapas Físicos (Abril, 2007)
[Disponible en: <http://www.geotecnet.com/fisico.html>]
- [6]. Larman, Craig. UML y Patrones. Introducción al análisis y diseño orientado a objetos. Editorial Félix Varela. La Habana, 2004. (Enero 2007)
- [7]. Wikiprod, Wikipedia de producción. (Marzo, 20007)
[Disponible en: <http://wiki.prod.uci.cu/index.php/Internet>]
- [8]. Álvarez Marañón, Gonzalo. Web Seguro. (Marzo, 20007)
[Disponible en: <http://www.iec.csic.es/criptonomicon/ssl.html>]
- [9]. CGI Group Inc. Common Gateway Interface. (Abril, 2007)]
[Disponible en: <http://hoohoo.ncsa.uiuc.edu/cgi/intro.html>.]
- [10]. Wikiprod, Wikipedia de producción. Sistemas Gestores de Bases de Datos. (Marzo, 20007)
[Disponible en: http://wiki.prod.uci.cu/index.php/Sistemas_gestores_de_bases_de_datos].
- [11]. Wikiprod, Wikipedia de producción. Oracle. (Marzo, 20007)
[Disponible en: <http://wiki.prod.uci.cu/index.php/Oracle>]
- [12]. Wikiprod, Wikipedia de producción. PHP. (Marzo, 20007)
[Disponible en <http://wiki.prod.uci.cu/index.php/Php>]
- [13]. González, Carlos D. Curso: Integral de diseño. Programación de sitios dinámicos con MySQL y PHP. (Abril, 2007)
[Disponible en: http://www.usabilidadweb.com.ar/x_int.php]
- [14]. Wikiprod, Wikipedia de producción. MySQL. . (Marzo, 20007)
[Disponible en <http://wiki.prod.uci.cu/index.php/MySQL>]
- [15]. González, Carlos D. Curso: Sitios Web dinámicos con Base de Datos PostgreSQL y PHP. . (Marzo, 20007)
[Disponible en: <http://www.usabilidadweb.com.ar/postgre.php>]
- [16]. Wikiprod, Wikipedia de producción. RUP(Enero, 2007)
[Disponible en: <http://wiki.prod.uci.cu/index.php/RUP>]
- [17]. Jacobson, Ivar; Booch, Grady y Rumbaugh, James. El Proceso Unificado de Desarrollo de Software. Editorial Félix Varela, La Habana, 2004. (Enero, 2007).
- [18]. Molpeceres, Alberto. Procesos de desarrollo: RUP, XP y FDD. (Marzo, 2007).
[Disponible en: <http://www.javahispano.org/articles.article.action?id=76>]

ANEXOS I

Distribución de proyectos y recursos de Fundeso.

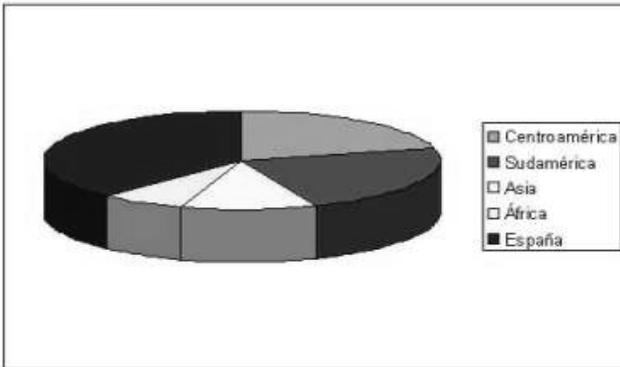


Fig.1. 1 Gráfico de pastel

Evolución del trabajo de Fundeso.

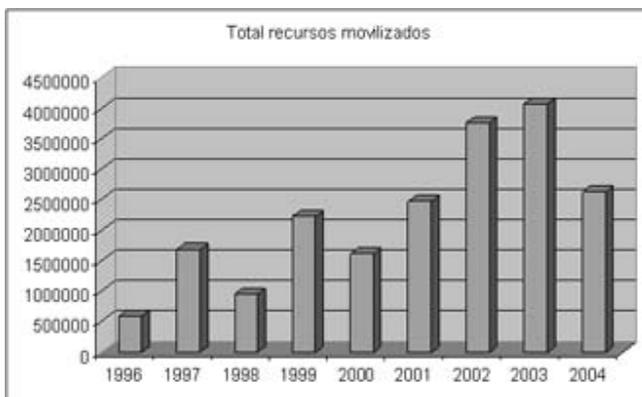


Fig.1. 2 Gráfico de barras.



Fig.1. 3 Mapa Temático

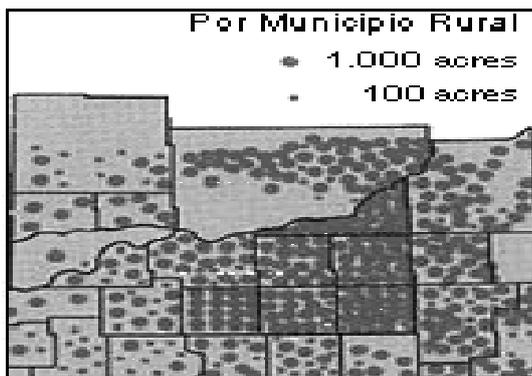


Fig.1. 4 Mapa de punto

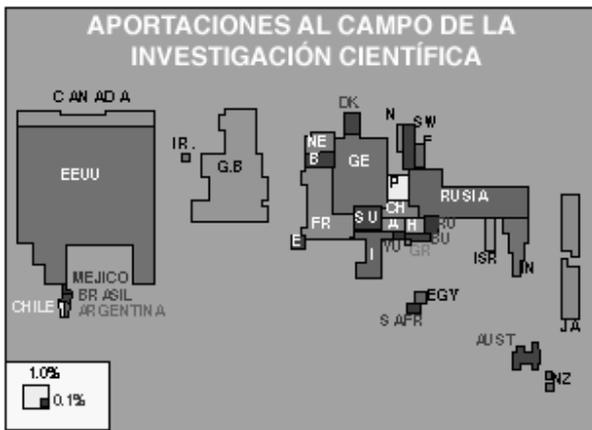


Fig.1. 5 Cartograma.

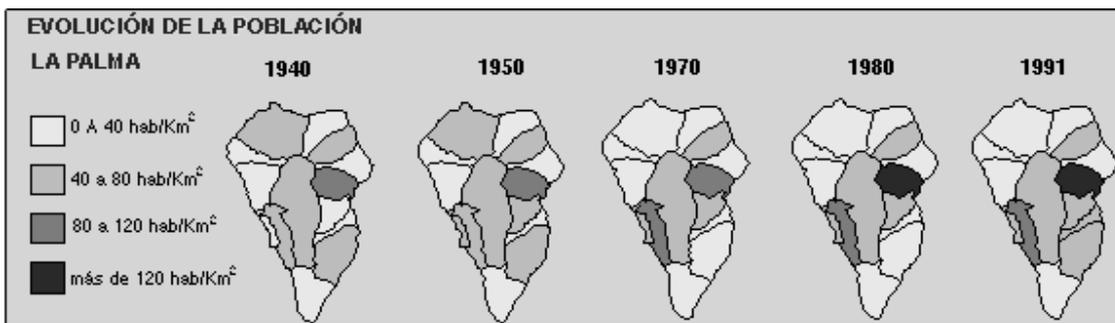


Fig.1. 6 coropletas

ANEXOS II

Tabla 3. 27 Diagrama de clases del análisis.

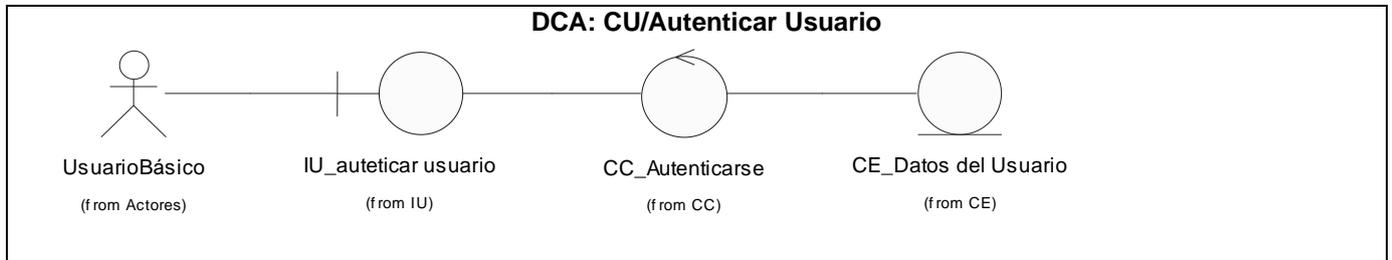


Tabla 3. 28 Diagrama de clases del análisis.

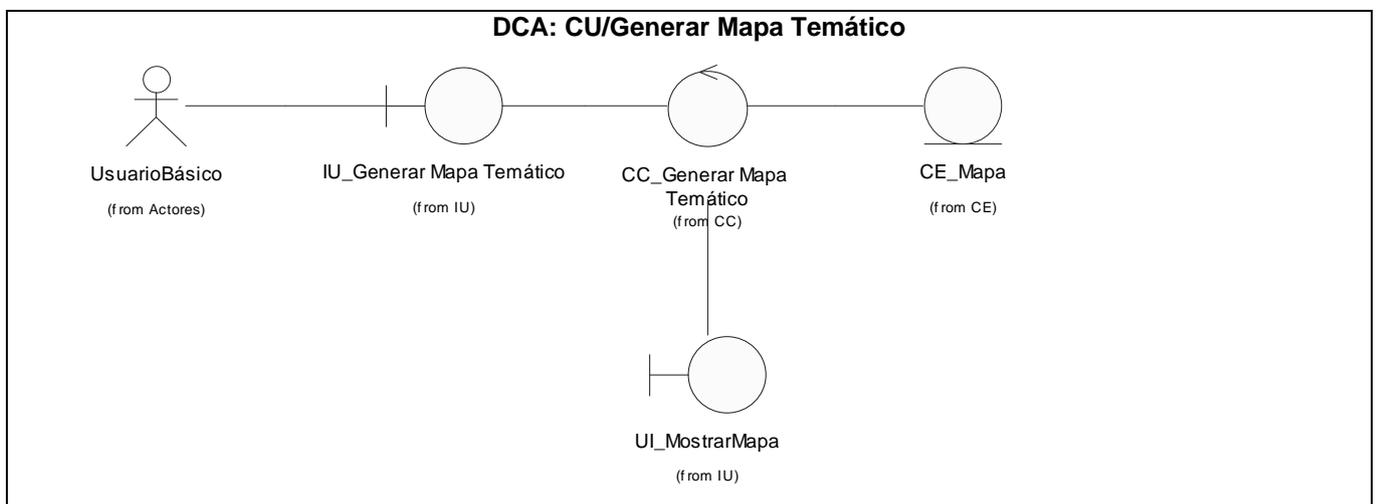


Tabla 3. 29 Diagrama de clases del análisis.

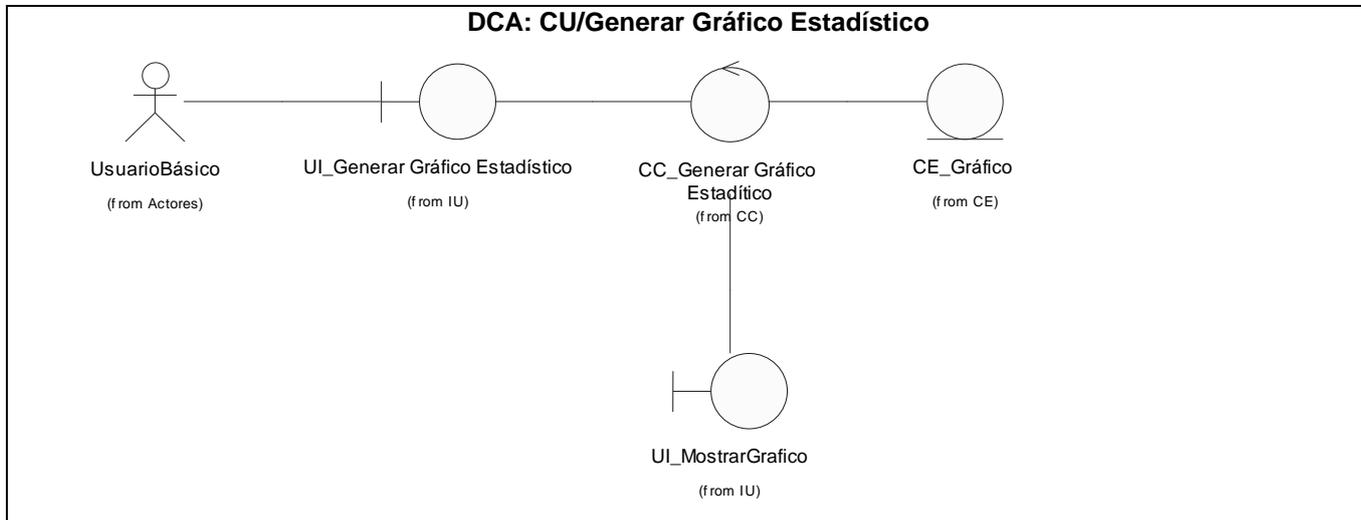


Tabla 3. 30 Diagrama de clases del análisis.

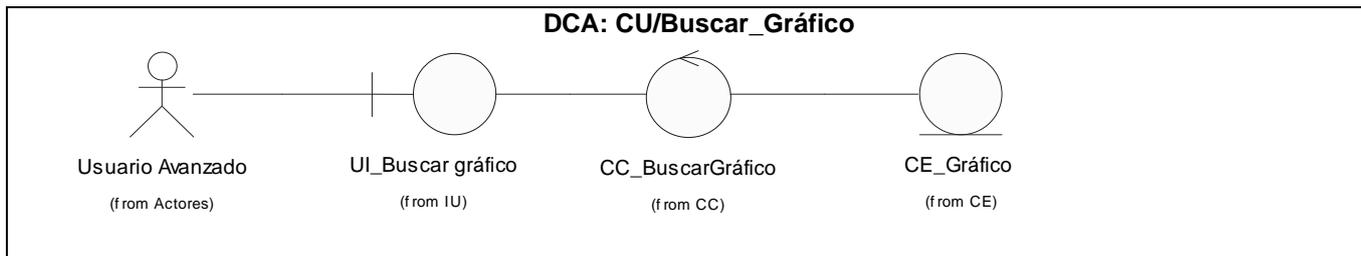


Tabla 3. 31 Diagrama de clases del análisis.

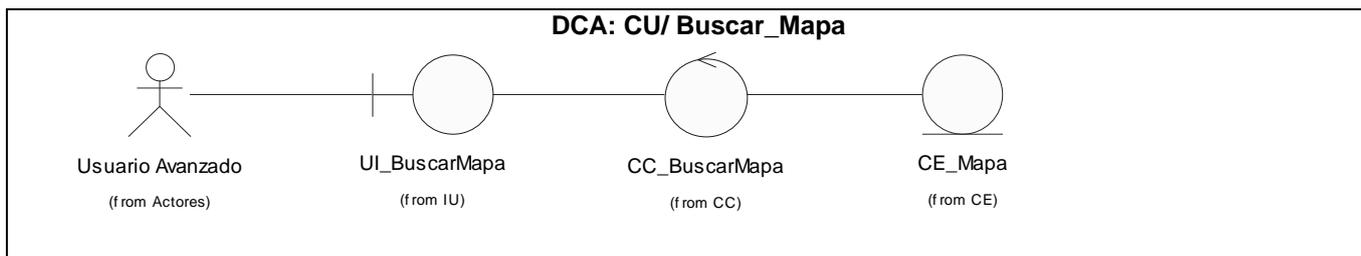


Tabla 3. 32 Diagrama de clases del análisis.

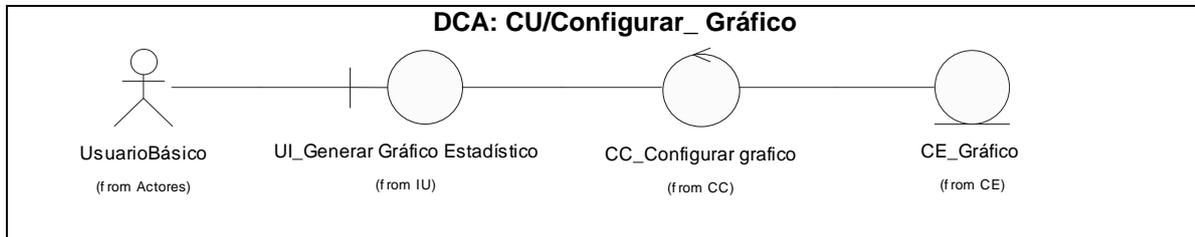


Tabla 3. 33 Diagrama de clases del análisis.

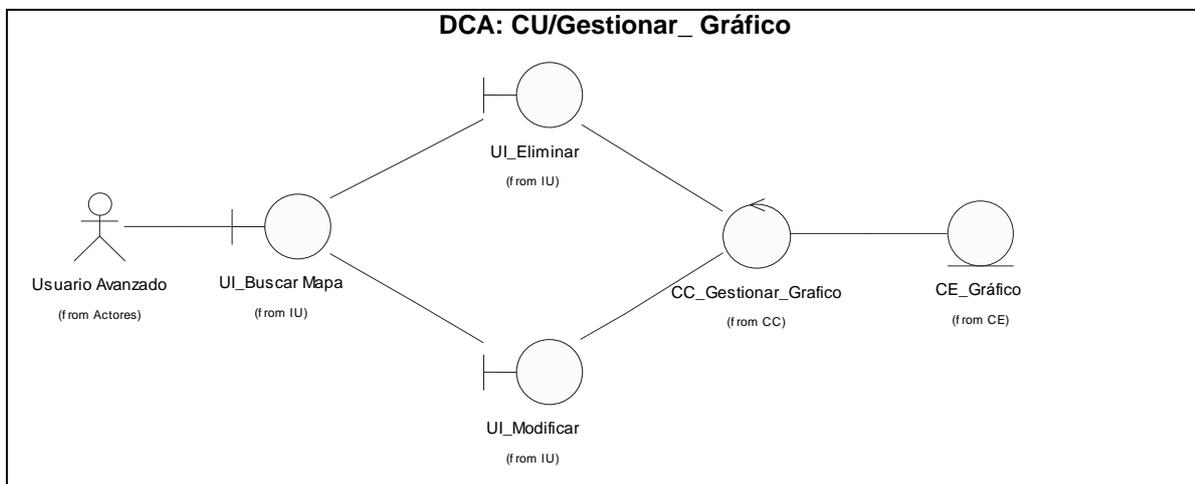


Tabla 3. 34 Diagrama de clases del análisis.

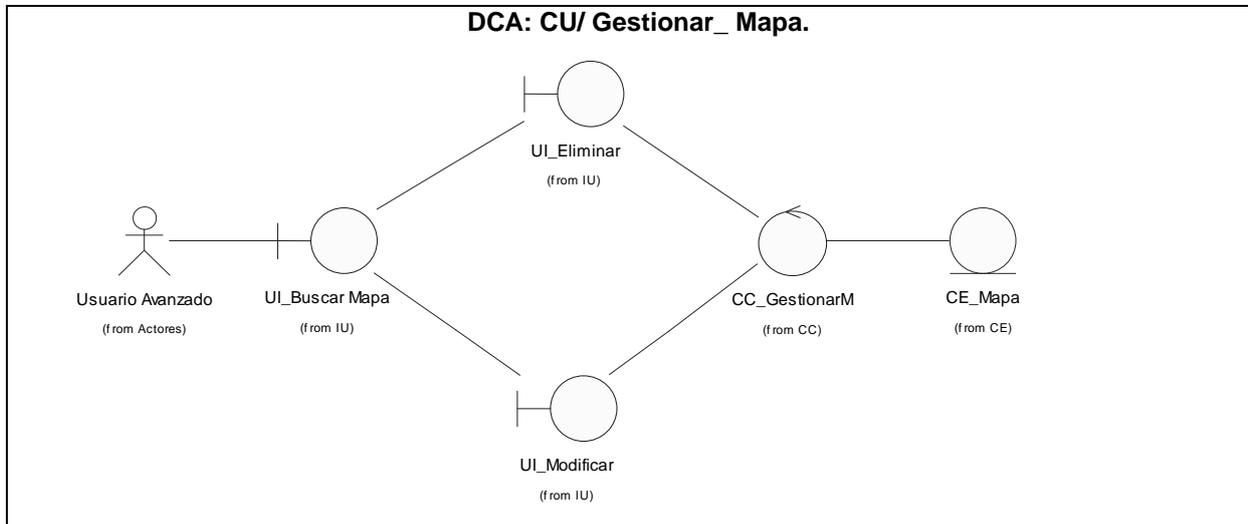


Tabla 3. 35 Diagrama de clases del análisis.

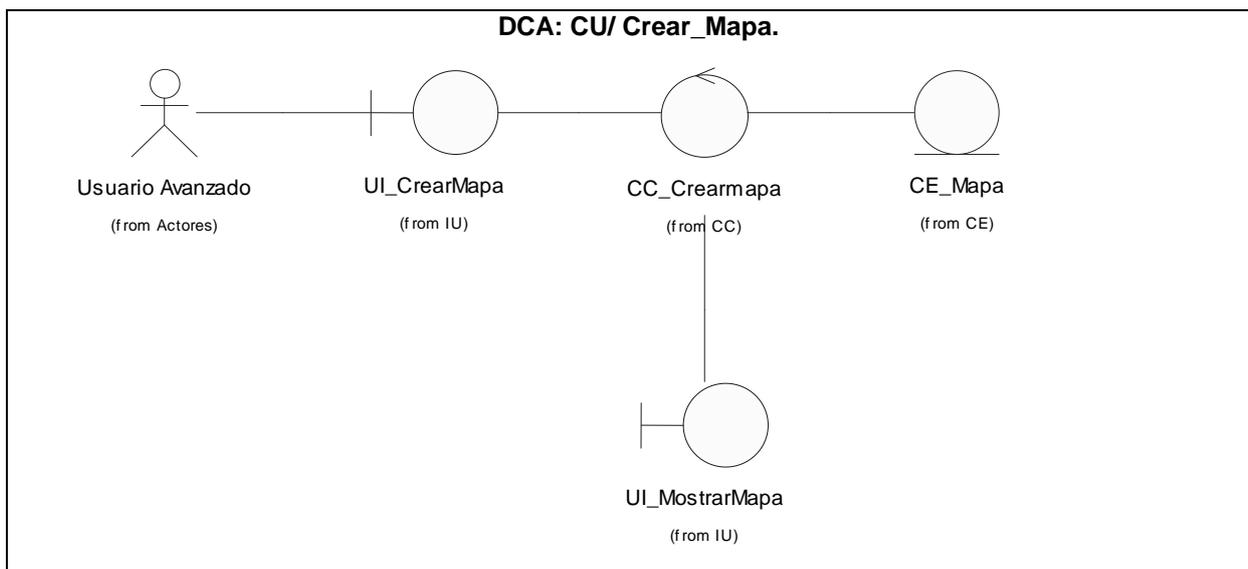


Tabla 3. 36 Diagrama de clases del análisis.

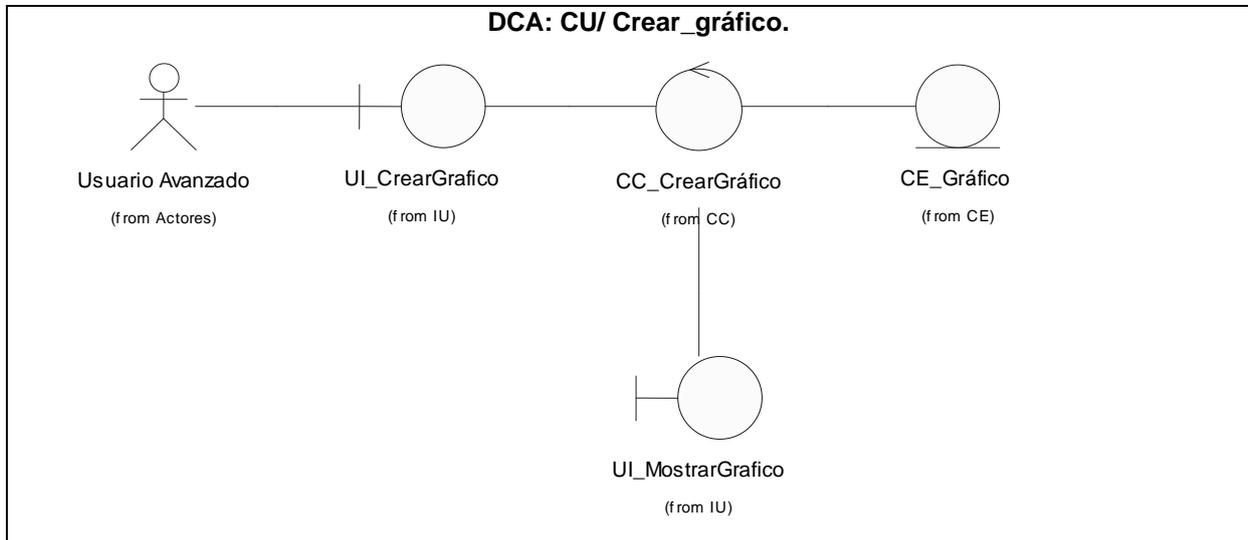
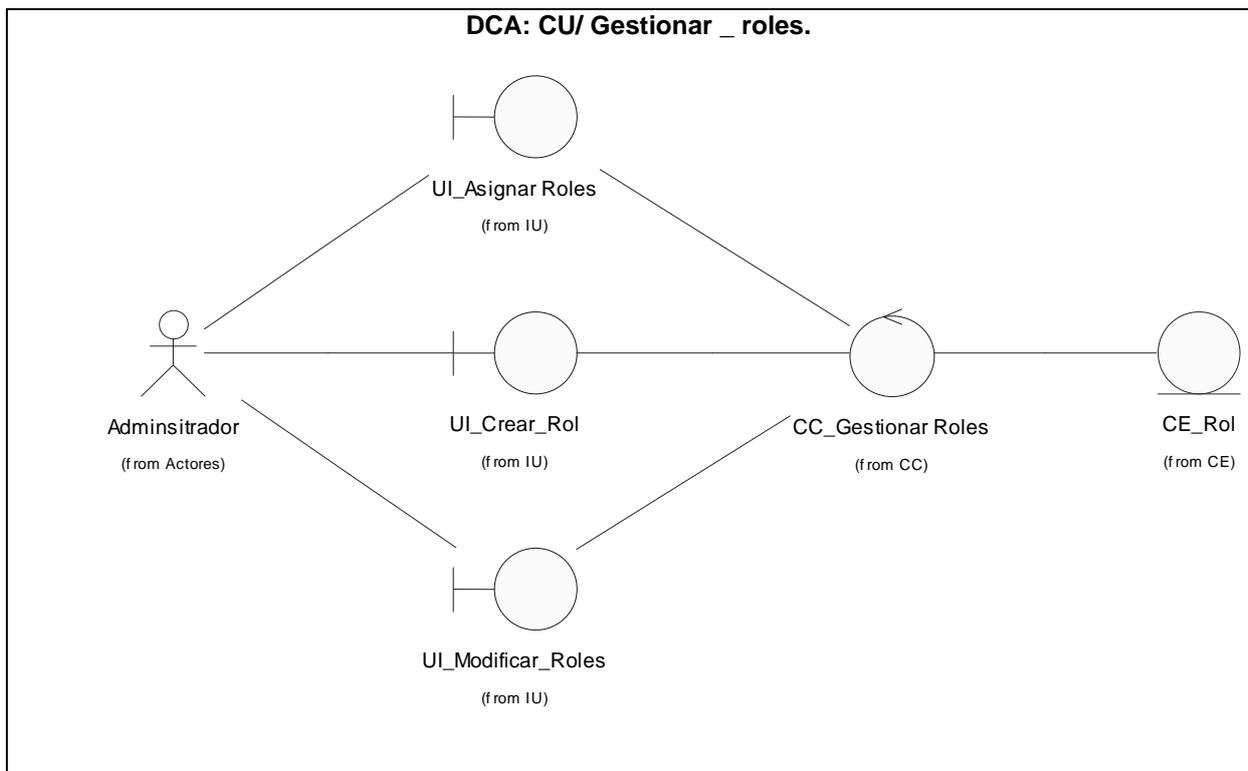


Tabla 3. 37 Diagrama de clases del análisis.



ANEXOS III

Tabla 3. 38 Diagrama de clases del Diseño.

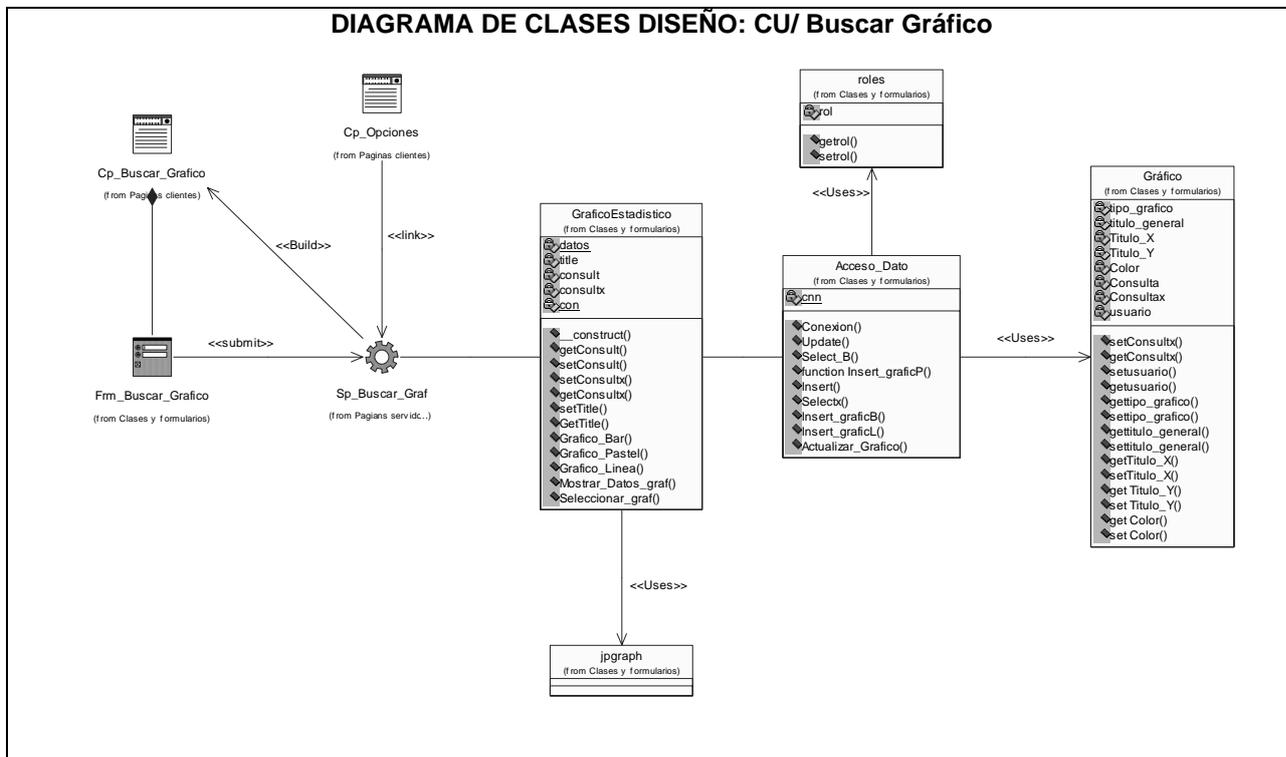


Tabla 3. 39 Diagrama de clases del Diseño.

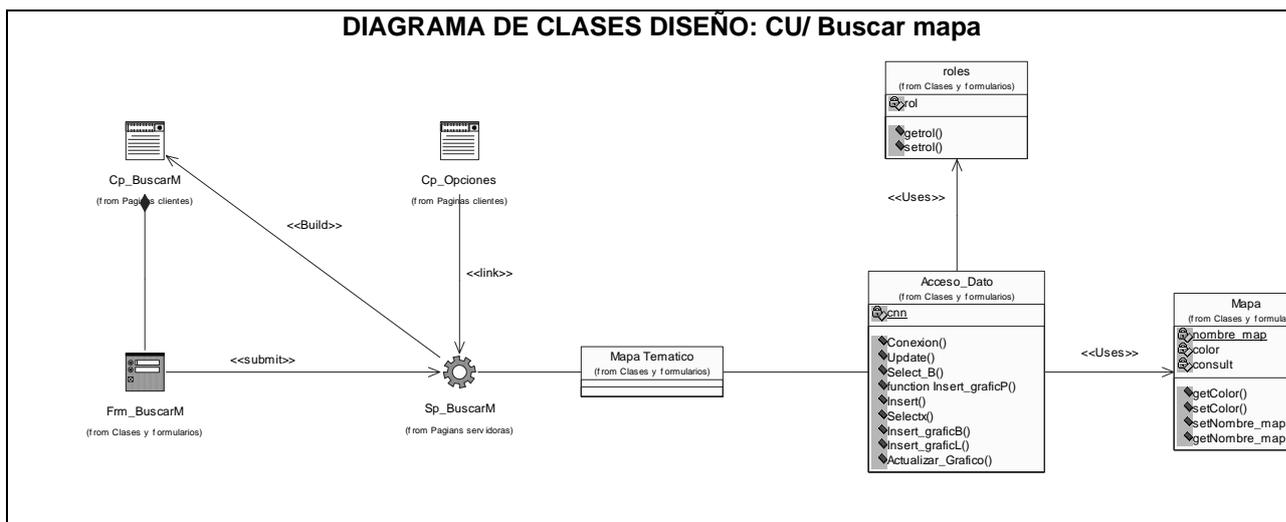


Tabla 3. 40 Diagrama de clases del Diseño.

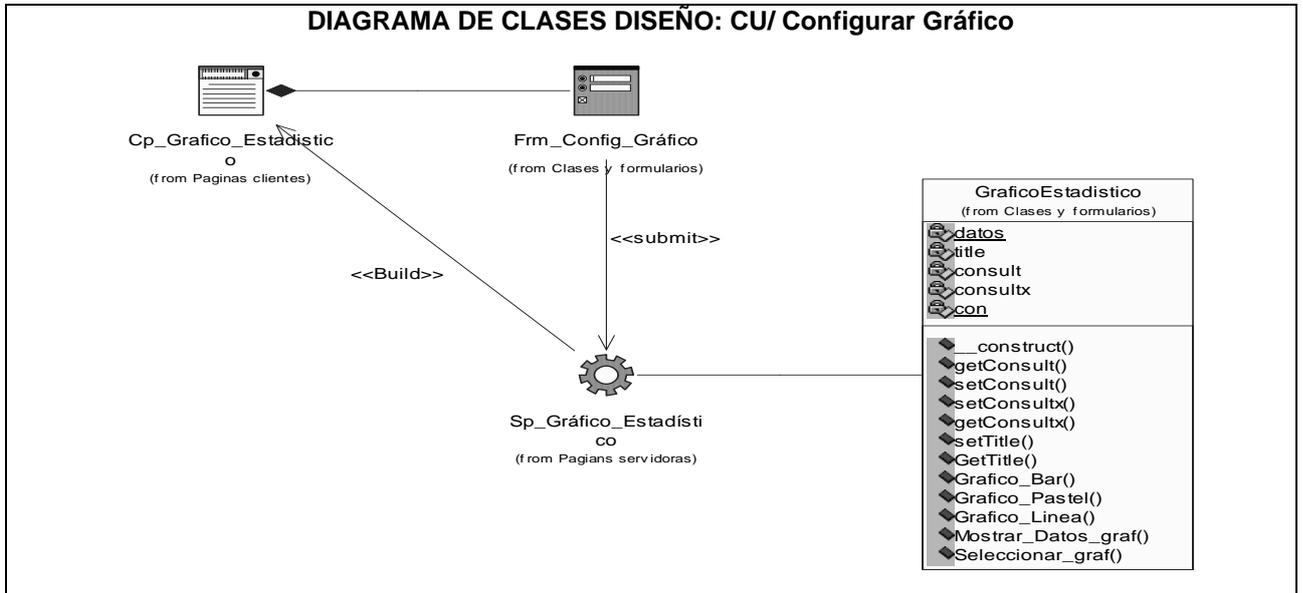


Tabla 3. 41 Diagrama de clases del Diseño.

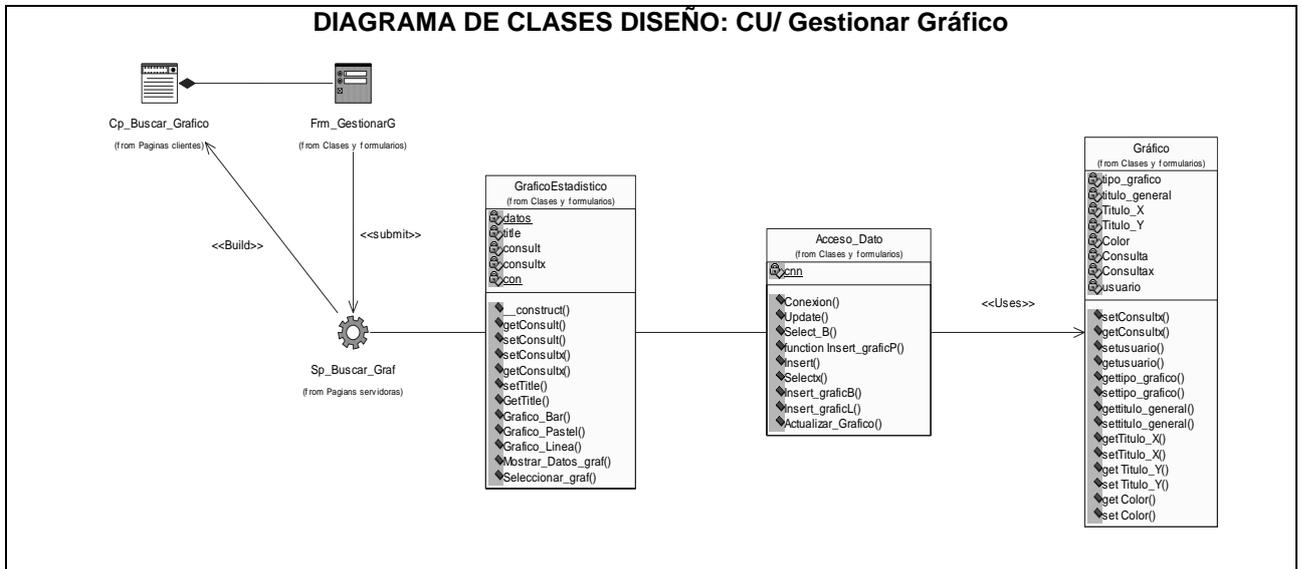
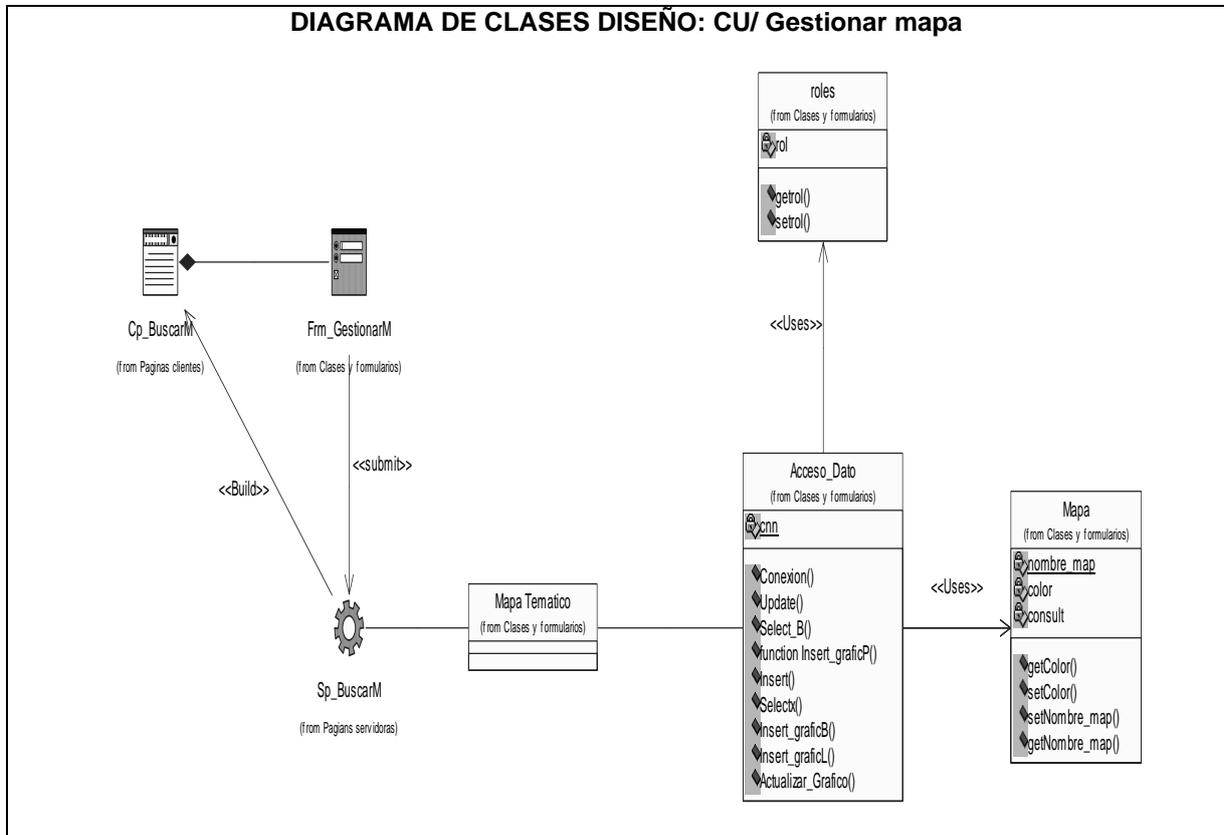


Tabla 3. 42 Diagrama de clases del Diseño.



ANEXO IV

Tabla 3. 43 Diagrama de interacción.

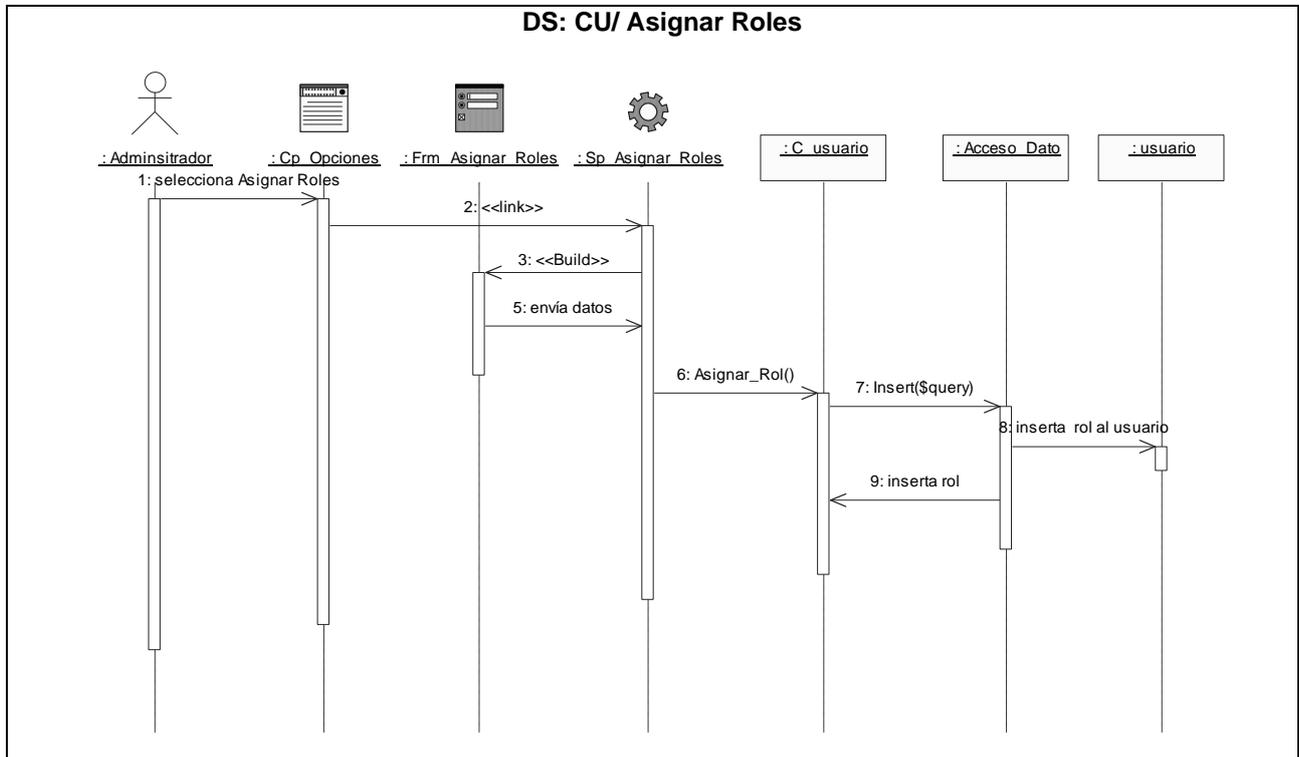


Tabla 3. 44 Diagrama de interacción.

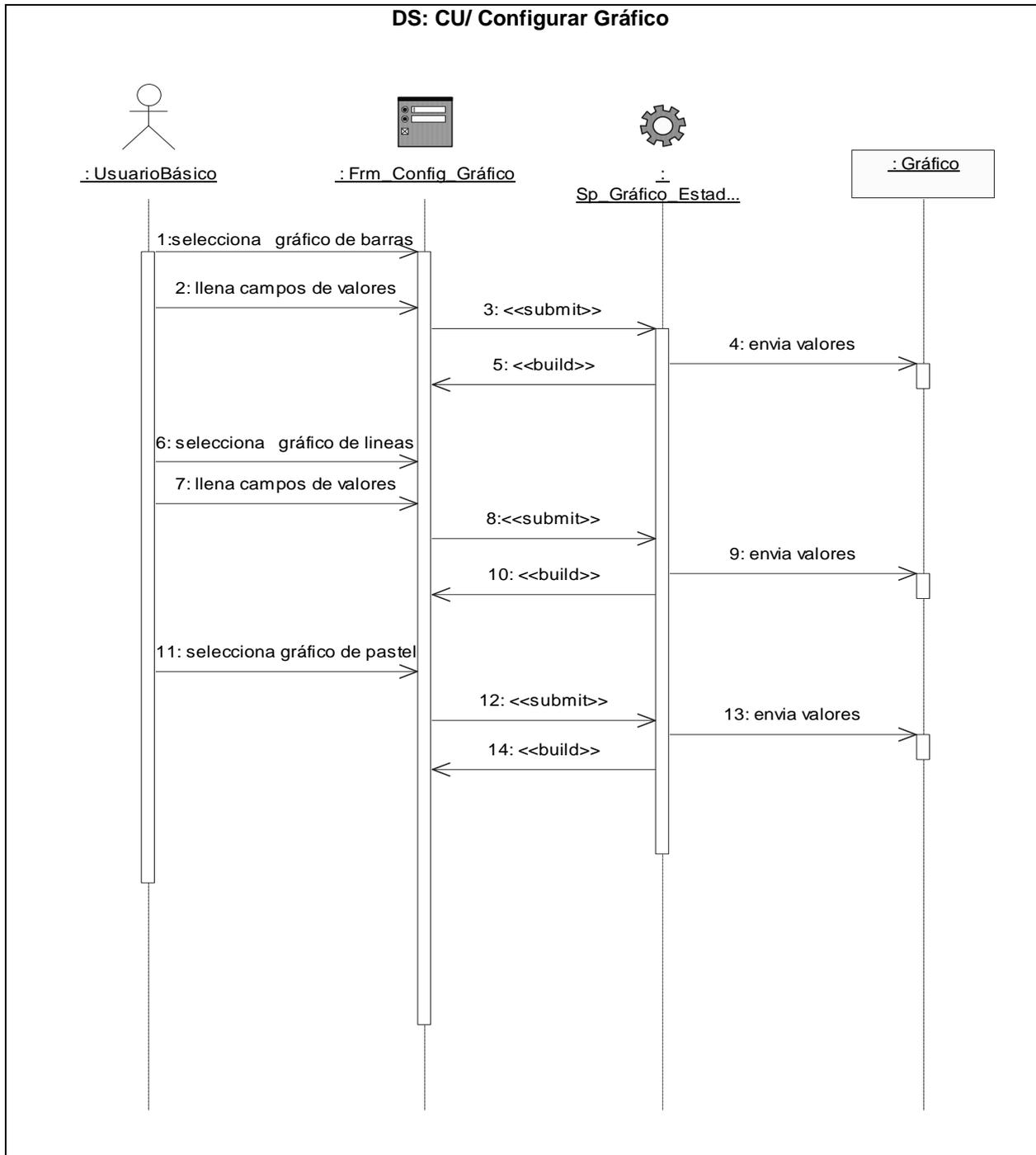


Tabla 3. 45 Diagrama de interacción.

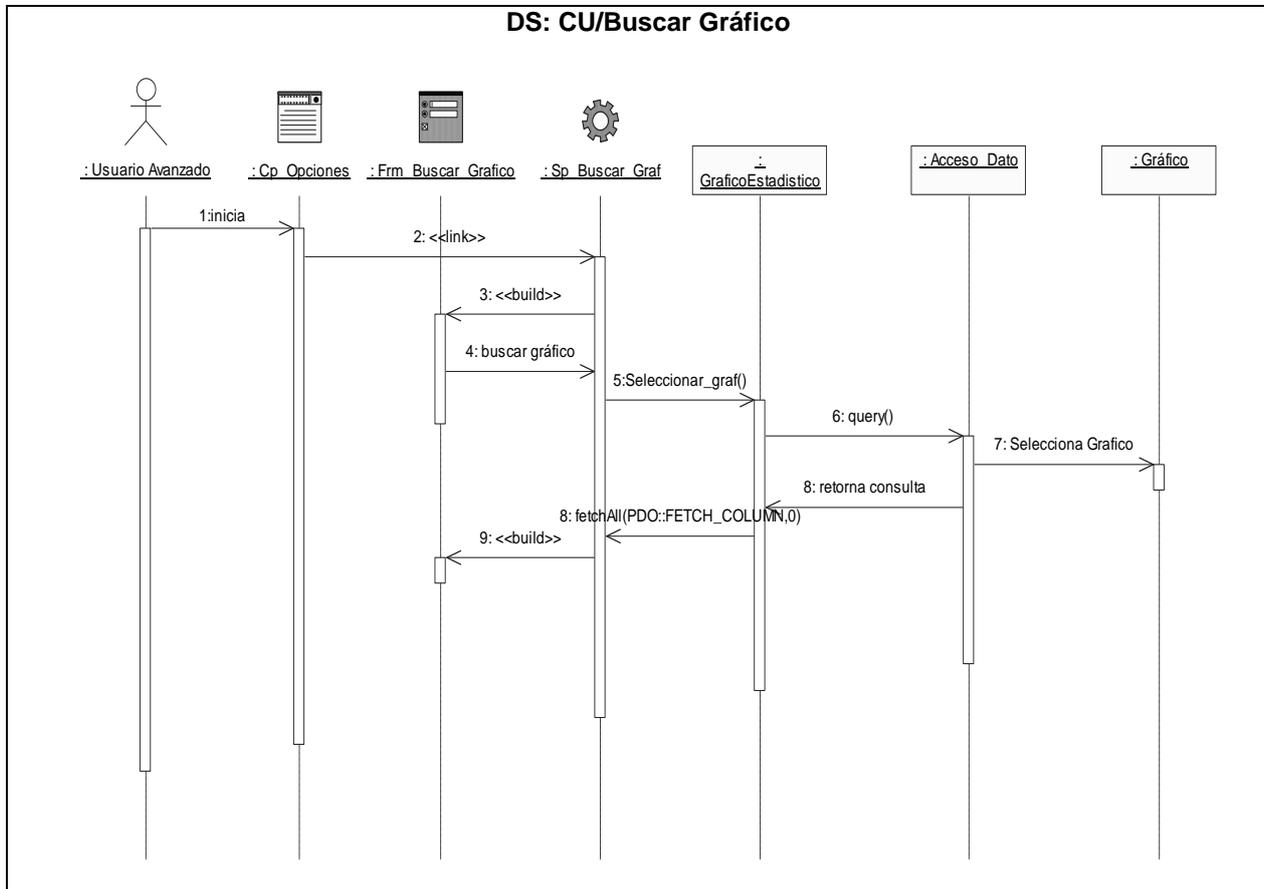
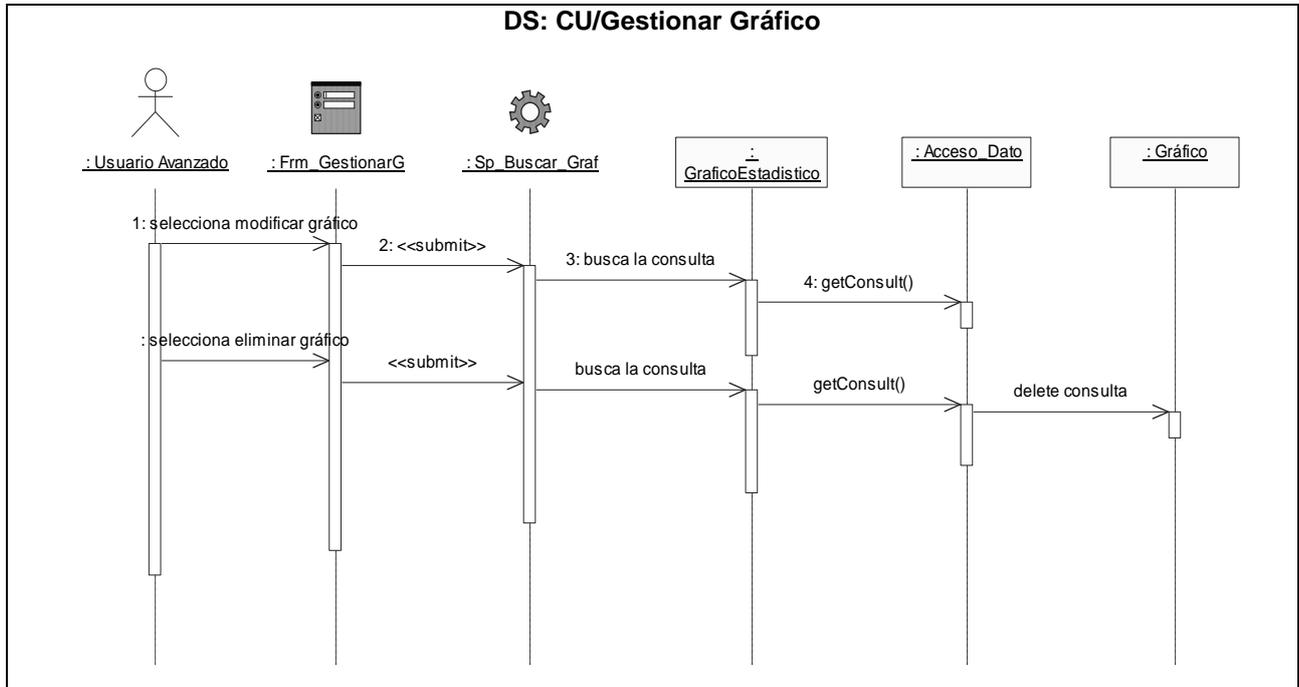


Tabla 3. 46 Diagrama de interacción.



GLOSARIO

- **Georeferenciada:** Es la forma de referenciar una área geográficamente.
- **Sistema de referencias:** Conjunto de convenciones usadas por un observador para poder medir la posición de un objeto físico en el tiempo y el espacio.
- **Coropletas:** Colores o tramas para representar el área que ocupa un hecho.
- **Gráficos:** Gráficos, en informática, es el nombre dado a cualquier imagen generada por un ordenador.
- **Mapa:** Es la representación gráfica de una porción del espacio geográfico, proyectada sobre una superficie plana.
- **Hipermedia:** Páginas Web que integran información en distintos tipos de formatos: texto, gráficos, sonidos y vídeo.