

Universidad de las Ciencias Informáticas

Facultad 6



**MÓDULO DE ESTADÍSTICA DESCRIPTIVA PARA LA LPS
APLICATIVOS SIG**

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

Autores:

Noslen Filgueiras Trujillo

Arlet Torres Carballosa

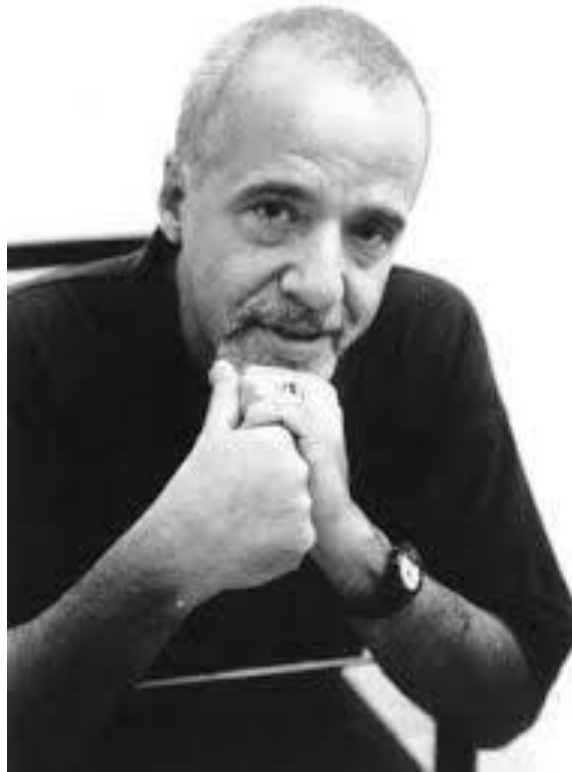
Tutores:

M.Sc. Vladimir Martell Fernández

Ing. Laritza Asán Caballero

Ing. Alain Leon Companioni

La Habana, junio de 2015
"Año 57 de la Revolución"



**“Cuando quieres realmente una cosa,
todo el Universo conspira para ayudarte
a conseguirla”.**

Paulo Coelho

DEDICATORIA

ARLET:

LE DEDICO EL RESULTADO DE MI ESFUERZO Y DEDICACIÓN:

A **Dios** POR ILUMINAR MI CAMINO HASTA EL DÍA DE HOY.

A LA PERSONA QUE CONSTITUYE EL PARADIGMA DE MI FAMILIA, **MI ABUELO** IGNACIO GUSTAVO CARBALLOSA PERERA, SIN SU EDUCACIÓN NO HABRÍA LLEGADO HASTA AQUÍ.

A **MI MAMÁ** IDALMIS CARBALLOSA HERNÁNDEZ POR SER EJEMPLO DE SUPERACIÓN Y SACRIFICIO EN TODO MOMENTO.

NOSLEN:

A **MIS HERMANAS:**

CUANDO SE PROPONGAN UNA META NO DEJEN DE LUCHAR POR ELLA, NINGÚN ESFUERZO ES EN VANO, TODO EN LA VIDA LLEGA LO QUE A SU DEBIDO TIEMPO.

MAMI TERE:

MIMA GRACIAS POR SER MI MAMÁ, MI PAPÁ, MI TODO, TU QUE MERECEES TODA LA FELICIDAD DEL MUNDO, NO TENGO PALABRAS PARA AGRADECERTE TODO LO QUE HAS HECHO POR MÍ, SIMPLEMENTE GRACIAS POR ILUMINARME DÍA A DÍA CON TU ETERNO Y DEDICADO AMOR.

AGRADECIMIENTOS

ARLET:

QUIERO AGRADECER DE MANERA MUY ESPECIAL A CADA PERSONA QUE PUSO SU GRANITO DE ARENA PARA QUE HOY ME CONVIERTA EN INGENIERA Y HAGA REALIDAD EL ANHELO MÍO Y DE MI FAMILIA:

A **Dios** POR SER EL GUÍA DE MI VIDA.

A **MI ABUELO** POR FORJAR TAN LINDA Y VALORADA FAMILIA, POR INCENTIVAR MI EDUCACIÓN SIENDO APENAS UNA NIÑA Y PORQUE A PESAR DE NO ESTAR CON NOSOTROS HOY SU RECUERDO PREVALECE ANTE TODO.

A **MI MAMÁ** POR DARMER TODO EL APOYO Y LA CONFIANZA NECESARIOS PARA CONVERTIRME EN LA MUJER INDEPENDIENTE QUE HE LLEGADO A SER.

A **MI ABUELA HILDA** (MIMA) LE QUIERO AGRADECER SIMPLEMENTE POR EXISTIR Y SER LA LUZ DE MIS OJOS, MI CONFIDENTE MÁS FIEL Y MI APOYO INCONDICIONAL.

A **MI TÍA HILDA**, (TITA) POR RECALCARMER TANTO LA NECESIDAD DE RESPETAR A LA FAMILIA Y LA IMPORTANCIA DEL HOGAR, POR CONTRIBUIR A MI EDUCACIÓN Y POR QUERERME SIN CONDICIONES COMO A UNA HIJA.

A **MI HERMANA**, (TATA) POR SER MI AMIGA, MI APOYO Y POR SIGNIFICAR PARA ELLA UN EJEMPLO A SEGUIR.

A **MI PAPÁ** POR ADORARME Y RESPETARME SIEMPRE.

A **MI PRIMO** LUISI Y **MI TÍO** ORLANDO (TATI) POR SER PARTE DE MI VIDA PARA HACER MIS DÍAS MÁS COMPLETOS.

AL RESTO DE **MI FAMILIA** LE AGRADEZCO SU AYUDA PARA QUE ESTO FUERA POSIBLE, DE MANERA MUY ESPECIAL A TETE Y SUS HIJAS, A LOS JIMAGUAS NILE E IVÁN Y SU MAMÁ MAYELÍN. LOS QUIERO MUCHOS A TODOS.

A SERGIO, **MI NOVIO**, POR LOS MARAVILLOSOS AÑOS QUE HEMOS COMPARTIDO Y POR TODO SU APOYO.

A **MIREYA** POR QUERERME COMO A UNA HIJA Y DARMER SIEMPRE LA RAZÓN Y AL RESTO DE LA FAMILIA POR ACOGERME.

A LA FAMILIA QUE LA VIDA ME DIO LA OPORTUNIDAD DE ESCOGER: **ALEXIS** Y **FABIÁN** POR SER MIS CUÑIS ADORADOS.

A LOS MEJORES AMIGOS QUE ALGUIEN PUEDA DESEAR: **CHIQUI**, **YOLA** Y **LITUAN** POR HACER MI VIDA MÁS GRATA CON SU PRESENCIA EN ELLA.

A **LUISMA** POR AYUDARME A SALIR HACIA ADELANTE TANTAS VECES Y POR SU INCONDICIONALIDAD.

A **LIS** Y **ELI** POR BATALLAR CONMIGO DURANTE TODA LA CARRERA EN LAS BUENAS Y MALAS.

A MI **COMPAÑERO DE TESIS**, NOSLEN, POR CONTRIBUIR CON SU ESFUERZO A QUE ESTO FUERA POSIBLE.

A **LOS TUTORES** ALAIN Y VLADIMIR POR SU DEDICACIÓN Y ESFUERZO, A LARITZA POR SU EMPEÑO EN QUE TODO SALIERA BIEN.

AL RESTO DE **MIS AMISTADES** Y CONOCIDOS POR FORMAR PARTE DE MI VIDA.

AGRADECIMIENTOS

NOSLEN:

ABUELO NELSON: GRACIAS POR HABERME FORMADO COMO LA PERSONA QUE SOY Y POR DEMOSTRARME QUE UNA VIDA NO ALCANZA PARA CONOCER A UNA PERSONA COMPLETAMENTE.

MAMI CHACHI: GRACIAS POR TODO EL AMOR, LOS REGAÑOS Y LOS CONSEJOS QUE ME HAS DADO, SIN LOS CUALES NO HUBIESE PODIDO SEGUIR ADELANTE Y HABER ALCANZADO ESTA META.

TÍO ROBE: GRACIAS POR APOYARME EN TODO, POR CREER EN MÍ Y POR SER EL MEJOR MAESTRO QUE HE TENIDO.

PAPÁ: GRACIAS PIPO POR SER EL PADRE QUE ERES, NUNCA CAMBIES, CUALQUIER HIJO ESTARÍA ORGULLOSO DE TENER UN PADRE COMO TÚ.

TATO: GRACIAS POR SER LA CHISPA Y LA ENERGÍA QUE LE FALTABA A ESTA FAMILIA, Y SOBRE TODO POR CONSIDERARME COMO UN HIJO.

ARY: MI VIDA GRACIAS POR QUERERME TODOS ESTOS AÑOS, POR APOYARME Y ESTAR AHÍ PARA MÍ EN TODO MOMENTO, POR COMPARTIR TU FELICIDAD Y TU TRISTEZA Y POR VIVIR A TU LADO LOS MOMENTOS MAS FELICES DE MI VIDA.

HECTI: GRACIAS POR ESTAR EN TODOS LOS MOMENTOS Y SITUACIONES POR LAS QUE PASE AQUÍ EN LA UNIVERSIDAD, GRACIAS POR SER MI COMPAÑERO DE ESTUDIO Y DE FIESTA, MI AMIGO, MI HERMANO Y MI ETERNO RIVAL: ¡HALAA MADRID!

ABUELO RICARDO: GRACIAS ABUELITO POR SER LA PERSONA QUE EN LOS MOMENTOS MAS DIFÍCILES Y ANGUSTIOSOS HACES SONREÍR A LOS DEMÁS, GRACIAS POR TENER ESA FUERZA Y ESAS GANAS DE SEGUIR VIVIENDO PARA PODER HACER FELIZ A TODO EL QUE TE RODEA.

ABUELA DEYSI: GRACIAS POR COMO ERES CONMIGO, POR TODO TU AMOR Y CARIÑO Y POR QUERER TANTO A ESTA FAMILIA.

AMISTADES: A TODOS AQUELLOS CON LOS QUE HE COMPARTIDO, CONVIVIDO, TODOS LOS QUE DE UNA FORMA U OTRA APORTARON UN GRANITO DE ARENA PARA QUE ESTE MOMENTO LLEGARA.

TUTORES: PORQUE FUERON MI MANO DERECHA Y POR ESTAR SIEMPRE DISPONIBLES PARA ENFRENTAR CUALQUIER ADVERSIDAD EN ESTE PROCESO DE INVESTIGACIÓN E IMPLEMENTACIÓN DE LA TESIS.

MI COMPAÑERA DE TESIS: GRACIAS POR SER TAN EXIGENTE Y POR TU ENTREGA Y DEDICACIÓN PARA LOGRAR ESTE SUEÑO.

DECLARACIÓN DE AUTORÍA

Declaro por este medio que nosotros, Arlet Torres Carballosa y Noslen Figueiras Trujillo, con números de carnet de identidad 92040424579 y 90121335065 respectivamente, somos los autores de este trabajo y que autorizamos a la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio, así como los derechos patrimoniales con carácter exclusivo.

Para que así conste, firmamos la presente declaración jurada de autoría en La Habana a los 27 días del mes de junio del año 2015.

Arlet Torres Carballosa
Autor

Noslen Figueira Trujillo
Autor

M.Sc. Ing. Vladimir Martell Fernández
Tutor

Ing. Laritza Asán Caballero
Tutor

Ing. Alain Leon Companioni
Tutor

RESUMEN

El surgimiento de la Estadística sucedió a través de un proceso largo de desarrollo y evolución, pasando, en un inicio, por la simple recolección de datos, hasta la compleja interpretación de los datos que se realiza hoy día. Esta ciencia mediante la recolección de datos cuantitativos brinda información que es empleada en las empresas para apoyar el proceso de toma de decisiones. La informática ofrece una variada gama de software para el procesamiento de grandes conjuntos de datos, uno de los más populares son los Sistemas de Información Geográfica. La Línea de Productos de Software Aplicativos SIG tiene como objetivo fundamental la producción de este tipo de software y para mejorar la calidad de sus productos necesita la utilización de técnicas Estadísticas. Para guiar el desarrollo del módulo de Estadística descriptiva se eligió la metodología RUP y para llevar a término el software se emplean las herramientas y tecnologías que constituyen la línea base de esta Línea. Se definió un modelo de dominio para esclarecer el contexto en el que se integrará el módulo y el proceso de desarrollo es centrado en Casos de Uso. La arquitectura seleccionada es compatible con la plataforma GeneSIG y se refleja en el Diagrama de Clases del Diseño. El proceso de pruebas arrojó varias no conformidades que se corrigieron dando como resultado un software con todas las funcionalidades deseadas.

Palabras claves: Estadística, GeneSIG, sistemas de información geográfica, toma de decisiones.

ABSTRACT

The emergence of the statistic came about through a long process of development and evolution, initially with the data collection and the data interpretation complex that is done today. This science by collecting quantitative data provides information which is used to support the process of decision-making in business. The informatics provides a variety of software's for processing large sets of data, one of the most popular are the GIS. The Software Product Lines Aplicativos SIG main purpose is the production of such software and to improve the quality of their products requires the use of statistical techniques. The RUP methodology was chosen to guide the development of descriptive statistic module and for the realization of it are used the tools and technologies that form the basis line of the Line. The domain model was defined to clarify the context in which the module is developed. The development process is focused on use cases. The GeneSIG platform supports the selected architecture and is reflected in the design class diagram. The testing process yielded several non-conformities and were corrected, of that way the software resulting have all desired features.

Key words: decision making, GeneSIG, geographic information systems, statistical.

ÍNDICE DE TABLAS

Tabla 1: Actores del Sistema.	35
Tabla 2: Descripción textual del Caso de Uso Calcular Parámetros estadísticos.	37
Tabla 3: Caso de Prueba del Caso de Uso Calcular Parámetros Estadísticos.	46
Tabla 4: Resultados de las Pruebas de rendimiento.	47
Tabla 5: Descripción textual del Caso de Uso Representar gráficamente parámetros estadísticos. ---	54
Tabla 6: Descripción textual del Caso de Uso Representar en el mapa parámetros estadísticos.	56
Tabla 7: Descripción textual del caso de Uso Gestionar capa.	57
Tabla 8: Descripción textual del Caso de Uso Gestionar atributo.	61

ÍNDICE DE FIGURAS

Figura 1: Análisis Estadístico en gvSIG.	18
Figura 2: Cálculo de la Estadística Básica en Quantum GIS.	19
Figura 3: Módulo de análisis estadístico de ARCGIS.	20
Figura 4: Salida gráfica del análisis exploratorio de datos en ARCGIS.	21
Figura 5: Modelo de Dominio.	30
Figura 6: Diagrama de Casos de Uso del Sistema.	36
Figura 7: Diagrama de clases persistentes.	41
Figura 8: Diagrama Entidad - Relación.	41
Figura 9: Diagrama de Clases del Diseño del Caso de Uso Calcular Parámetros Estadísticos.	43
Figura 10: Diagrama de despliegue.	44
Figura 11: Diagrama de componentes.	44
Figura 12: No conformidades detectadas durante el proceso de desarrollo.	48
Figura 13 Diagrama de Clases del diseño del Caso de Uso Representar en el mapa los Parámetros Estadísticos.	65
Figura 14: Diagrama de Clases del Diseño del Caso de uso Graficar Parámetros Estadísticos.	66
Figura 15: Diagrama de Clases del Diseño del Caso de Uso Gestionar atributo.	67
Figura 16: Diagrama de Clases del Diseño del Caso de uso Gestionar capa.	68

ÍNDICE DE CONTENIDO

INTRODUCCIÓN.....	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA.....	6
1.1 Introducción	6
1.2 Toma de Decisiones.....	6
1.2.1 Importancia de la Toma de Decisiones	7
1.2.2 La Estadística en la Toma de Decisiones	7
1.3 Estadística Descriptiva	8
1.3.1 ¿Qué es la Estadística Descriptiva?.....	8
1.3.2 Conceptos asociados a la Estadística descriptiva.....	8
1.3.3 Parámetros fundamentales de la Estadística descriptiva.....	10
1.4 Los SIG	12
1.4.1 Definiciones principales de SIG.....	12
1.4.2 Partes fundamentales de un SIG	13
1.4.3 Utilidad e importancia de los SIG	13
1.5 La plataforma GeneSIG.....	15
1.5.1 Descripción general de GeneSIG.....	15
1.5.2 Principales funcionalidades.....	16
1.6 Estudio de algunas soluciones existentes.....	17
1.7 Conclusiones.....	21
CAPÍTULO 2: HERRAMIENTAS Y TECNOLOGÍAS A UTILIZAR.....	22
2.1 Introducción	22
2.2 Metodología de desarrollo de Software.....	22
2.2.1 El Proceso Unificado de Desarrollo	22
2.3 Lenguaje de modelado: UML.....	23
2.4 Visual Paradigm for UML.....	24
2.4.1 Visual Paradigm for UML 8.0.....	25
2.5 Lenguajes de programación.....	25
2.5.1 PHP	25
2.5.2 PHP MapScript.....	26
2.5.3 ExtJS 3.0.....	26
2.5.4 Python	27

2.5.5	Lenguaje R	27
2.6	IDE NetBeans 8.0.....	27
2.7	Sistema Gestor de Base de Datos.....	28
2.7.1	PostgreSQL como SGBD	28
2.7.2	PostGIS: la extensión geográfica del gestor seleccionado	28
2.8	PGAdmin III para el manejo de PostgreSQL.....	29
2.9	Servidor de mapas: MapServer 6.0	29
2.10	Conclusiones parciales	29
CAPÍTULO 3: ANÁLISIS Y ARQUITECTURA DEL SISTEMA		30
3.1	Introducción	30
3.2	Modelo de dominio.....	30
3.2.1	Definición de clases del diagrama del dominio	31
3.3	Requisitos del Software	32
3.3.1	Requisitos funcionales.....	32
3.3.2	Requisitos no funcionales	33
3.4	Modelo del sistema	35
3.4.1	Actores del sistema	35
3.4.2	Patrones de Casos de Uso del Sistema	35
3.4.3	Diagrama de Casos de Uso del Sistema	36
3.4.4	Descripción textual de los Casos de Uso del Sistema.....	36
3.5	Arquitectura de software	38
3.5.1	Patrones arquitectónicos y de diseño.....	39
3.6	Estándar de codificación	40
3.7	Conclusiones parciales	40
CAPÍTULO 4: DISEÑO, IMPLEMENTACIÓN Y PRUEBAS.....		41
4.1	Introducción	41
4.2	Diseño de la Base de Datos.....	41
4.2.1	Diagrama de clases persistentes.....	41
4.2.2	Diagrama Entidad – Relación (DER).....	41
4.3	Modelo de Diseño.....	42
4.3.1	Diagrama de Clases del Diseño	42
4.4	Modelo de Implementación.....	42
4.4.1	Diagrama de Componentes.....	42

4.5	Modelo de Despliegue.....	43
4.6	El Proceso de Pruebas.....	45
4.6.1	Diseño de Casos de Prueba.....	45
4.6.2	Pruebas de Rendimiento.....	47
4.6.3	Ejecución de las pruebas.....	47
4.7	Conclusiones parciales.....	48
	CONCLUSIONES.....	49
	RECOMENDACIONES.....	50
	REFERENCIAS BIBLIOGRÁFICAS.....	51
	ANEXO I: DESCRIPCIÓN TEXTUAL DE LOS CASOS DE USO.....	54
	ANEXO II: DIAGRAMAS DE CLASES DEL DISEÑO.....	65

INTRODUCCIÓN

El surgimiento de la Estadística sucedió a través de un proceso largo de desarrollo y evolución, pasando, en un inicio, por la simple recolección de datos, hasta la compleja interpretación de los datos que se realiza hoy día. Se hace muy difícil definir cuáles son sus orígenes, pues desde el nacimiento mismo de la civilización ya se empleaban formas simples de Estadística, los hombres hacían representaciones gráficas y cuentas, tanto en paredes como en pieles de animales. Alrededor del año 3000 a.C. ya eran empleadas por los babilonios tablillas de arcilla para llevar el estado de sus producciones y ventas. La Estadística tal y como se le conoce hoy, así como la mayoría de los métodos de inferencia, se deben a Ronald Arnold Fisher (GONZÁLEZ 2005).

Se puede definir la Estadística como la ciencia cuyo objetivo es reunir información cuantitativa concerniente a individuos, grupos, series de hechos, entre otros, y deducir de ello gracias al análisis de dicha información, unos significados precisos o unas previsiones para el futuro (MUÑOZ 2004). Esta ciencia, mediante la información cuantitativa que ofrece permite conocer mucho mejor a una sociedad, información que es utilizada por los gobiernos para mejorar la situación de la población; también es utilizada en el área de la investigación científica ya que los datos estadísticos son el resultado de varios casos de entre los cuales se toma un promedio, por lo que sirve para demostrar que determinado caso posee un resultado particular; se emplea con propósitos descriptivos (tabulación de datos, representaciones gráficas, entre otros.); otras aplicaciones de las técnicas Estadísticas son en mercadotecnia, contabilidad, control de calidad, análisis de resultados en deporte, la medicina, la educación, en la toma de decisiones, entre otros.

Esta última en la mayoría de las empresas se hace un proceso engorroso sin el apoyo de una aplicación informática, es por ello que destaca muchísimo la relación existente entre ambas disciplinas. Mediante el uso de computadoras y sistemas informáticos se obtienen resultados más precisos en numerosos cálculos que resultan complejos para el cerebro humano, también los investigadores pueden mejorar la calidad de los datos originales y la interpretación de los resultados.

La diversidad de aplicaciones informáticas que existen hoy día propone disímiles funcionalidades para el análisis de grandes conjuntos de datos, ejemplo de esto son los Sistemas de Información Geográfica,

INTRODUCCIÓN

tecnología que se ha fomentado en los últimos años y que cada vez se hace más popular. Diversos han sido los esfuerzos por desarrollar aplicaciones de este tipo en Cuba, cuyos objetivos fundamentales radican en apoyar y asegurar el éxito del proceso de toma de decisiones de sus negocios.

La Línea de Productos de Software (en lo adelante LPS) Aplicativos SIG forma parte de la estructura productiva del Centro Geoinformática y Señales Digitales de la Universidad de las Ciencias Informáticas. Esta LPS tiene como objetivo fundamental desarrollar aplicaciones informáticas de apoyo a la toma de decisiones, que contribuyan al seguimiento y control de la información socioeconómica de cualquier sector de la sociedad a partir de su representación y análisis espacial. Estas aplicaciones son conocidas como Sistemas de Información Geográfica; los mismos se desarrollan haciendo uso de la plataforma GeneSIG la cual entre sus funcionalidades permite la consulta, análisis y visualización de la información geográfica.

Aun cuando el desarrollo de Sistemas de Información Geográfica (en lo adelante SIG) en esta LPS cuenta con más de 5 años de experiencia, hoy los sistemas que se producen no suponen el apoyo en la toma de decisiones que se espera, redundando en varios inconvenientes para sus usuarios finales y para el equipo que los desarrolla. En primer lugar, el proceso se ve afectado por la imposibilidad de realización de análisis estadísticos sobre la información que se muestra, lo cual afecta considerablemente las decisiones de los decisores al extremo de tornarlas equivocadas. Si luego de la representación de un fenómeno no es posible obtener datos estadísticos, la decisión no es rigurosamente acertada pues desconoce aristas importantes, análisis referidos a promedio, moda, mediana, desviación y otros más específicos que son a menudo de interés para agentes decisores.

Lo anterior cobra particular interés cuando la información es mostrada sobre un mapa, lo cual le añade un valor adicional pues se compone de una ubicación espacial que también es susceptible de ser tenida en cuenta en la decisión. Utilizar esta componente espacial aumentaría la calidad de la información Estadística que se muestra.

De cara al equipo de desarrollo, la comercialización de estos sistemas se ha visto comprometida en más de una ocasión -y en algunos casos no ha sido posible- al no lograr satisfacer los requerimientos de algunos clientes interesados en salidas Estadísticas y análisis más profundos de la información que se maneja, limitando la visibilidad y la introducción del software en el mercado.

INTRODUCCIÓN

Debido a lo planteado anteriormente se identifica el siguiente **Problema de Investigación**: ¿Cómo fortalecer el apoyo a la toma de decisiones en los SIG que se desarrollan haciendo uso de la plataforma GeneSIG, en la LPS Aplicativos SIG?

Se propone para la presente investigación como **Objetivo General** desarrollar el módulo de Estadística descriptiva para la LPS Aplicativos SIG, tomando como **Objeto de Estudio** los parámetros fundamentales de la Estadística Descriptiva aplicados en los SIG y como **Campo de Acción** la informatización del cálculo de los parámetros fundamentales de la Estadística Descriptiva.

Para guiar el desarrollo de la presente investigación se definieron una serie de **Preguntas Científicas** las cuales se enumeran a continuación:

1. ¿Cómo definir el proceso de toma de decisiones?
2. ¿Qué es la LPS Aplicativos SIG, qué estructura tiene y cómo desarrolla cada uno de los SIG que implementa?
3. ¿Qué estudia la Estadística Descriptiva?
4. ¿Cómo diseñar y desarrollar de forma flexible un módulo de estadística de apoyo a la toma de decisiones en la LPS Aplicativos SIG?
5. ¿Cómo integrar el módulo de estadística a la plataforma GeneSIG?
6. ¿Cómo obtener una herramienta informática libre de errores?

Para dar cumplimiento al objetivo general propuesto anteriormente se desarrollan una serie de tareas, las cuales se enumeran a continuación:

1. Caracterizar el proceso de desarrollo de SIG en la LPS Aplicativos SIG.
2. Valorar las soluciones existentes que responden al campo de acción de la investigación en alguna medida, sus limitaciones y fortalezas.
3. Caracterizar las principales herramientas, tecnologías, lenguajes, y metodologías, a utilizar para la construcción de la propuesta de solución.
4. Realizar el análisis y el diseño de la solución propuesta.
5. Implementar la solución propuesta.
6. Desarrollar el proceso de pruebas al módulo implementado.

INTRODUCCIÓN

Durante el desarrollo de la presente investigación se decide utilizar los métodos científicos siguientes:

Teóricos:

Histórico-Lógico: El método histórico estudia la trayectoria real de los fenómenos y acontecimientos en el transcurso de su historia. El método lógico investiga las leyes generales del funcionamiento y desarrollo de los fenómenos (ZAYAS 1995). Este método se emplea para analizar la evolución de los conceptos asociados a la Estadística Descriptiva permitiendo formar una definición propia.

Análítico-Sintético: Permite la descomposición de un todo complejo en sus partes y cualidades. La síntesis, por su parte, establece la unión entre las partes, previamente analizadas y posibilita descubrir relaciones y características generales entre los elementos de la realidad (ZAYAS 1995). Este método se emplea para la valoración de las soluciones existentes que responden al campo de acción. Además se utiliza en la selección de las herramientas y tecnologías empleadas en el desarrollo de la solución.

Empíricos:

Análisis documental: Facilita dentro de la investigación científica el dominio de las técnicas empleadas para el uso de la bibliografía. Permite la creación de habilidades para el acceso a investigaciones científicas, reportadas en fuentes documentales de forma organizada (CASTILLO 2005). En la presente investigación se utilizan los procedimientos lógicos y mentales como análisis, síntesis, deducción e inducción; se hizo una recopilación adecuada de datos de varias fuentes permitiendo el planteamiento de nuevas interrogantes y la sugerencia de nuevas bibliografías para su consulta.

La presente investigación está estructurada en cuatro capítulos:

Capítulo 1: En este capítulo se exponen los fundamentos teóricos relacionados con los conceptos fundamentales de la Estadística Descriptiva, así como el análisis de las soluciones existentes en la actualidad que cuentan con módulos estadísticos.

Capítulo 2: En este capítulo se aborda la selección de las herramientas y tecnologías que se utilizan para la construcción de la propuesta de solución. Se argumenta su utilización así como sus principales características.

INTRODUCCIÓN

Capítulo 3: En este capítulo se comienza el desarrollo de la solución propuesta teniendo en cuenta la metodología seleccionada, se definen el modelo de dominio, los requisitos funcionales y no funcionales, se describen los Casos de Uso del sistema y se enuncian los elementos fundamentales de la arquitectura.

Capítulo 4: En este capítulo se definen los elementos de la Base de Datos, los diagramas de diseño, implementación y despliegue. Finalmente, se describe el proceso de pruebas de la solución.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

1.1 Introducción

En este capítulo se desarrolla lo que constituye la fundamentación teórica de la solución propuesta, se definen los principales conceptos asociados a los SIG, al proceso de toma de decisiones y la estadística descriptiva. Finalmente se analizan algunas posibles soluciones existentes.

1.2 Toma de Decisiones

La toma de decisiones es una actividad de extraordinaria importancia tanto en entornos administrativos como en la vida cotidiana del ser humano debido a su impacto económico, social y motivacional (VÉLIZ 2014). Al decir de (KEENEY and RAIFFA 1993), intenta ayudar a los individuos a tomar decisiones difíciles y complejas de una forma racional. Esta racionalidad implica el desarrollo de métodos y modelos que permitan representar fielmente cada problema y analizar las distintas alternativas con criterios objetivos. Sin embargo no todo problema de decisión se resuelve a través de un proceso racional.

Generalmente el proceso de toma de decisiones transita por las fases que a continuación se mencionan (CLEMEN 1996):

1. Identificación de decisión y sus objetivos.
2. Identificación de alternativas.
3. Definición del marco de trabajo.
4. Recolección de información.
5. Valoración de alternativas.
6. Elección de las mejores alternativas.
7. Análisis sensitivo.
8. Tomar decisión.

El padre de la Teoría de la Decisión ofrece otro concepto de toma de decisiones en el que plantea que es un proceso de selección entre cursos alternativos de acción, basado en un conjunto de criterios, para alcanzar uno o más objetivos (SIMON 1982). Se puede decir entonces que para efectuar un buen proceso

de toma de decisiones en primer lugar se debe entender y definir el problema, así como las alternativas, evaluar las mismas y finalmente escoger la que constituye la decisión.

1.2.1 Importancia de la Toma de Decisiones

Mediante el empleo de un buen juicio, la Toma de Decisiones indica que un problema o situación es valorado y considerado profundamente para elegir el mejor camino a seguir según las diferentes alternativas y operaciones. También es de vital importancia para la administración ya que contribuye a mantener la armonía y coherencia del grupo, y por ende su eficiencia (SIMON 1982).

En la Toma de Decisiones, considerar un problema y llegar a una conclusión válida, significa que se han examinado todas las alternativas y que la elección ha sido correcta. Dicho pensamiento lógico aumentará la confianza en la capacidad para juzgar y controlar situaciones. La toma de decisiones, se considera como parte importante del proceso de planeación cuando ya se conoce una oportunidad y una meta, el núcleo de la planeación es realmente el proceso de decisión (SIMON 1982).

1.2.2 La Estadística en la Toma de Decisiones

En todo proceso de decisión se necesita recabar información para dar respuesta a las indagaciones sobre los aspectos fundamentales que interfieren en la decisión; para que los resultados sean fiables tanto la recogida de los datos como su análisis deben ser realizados de forma objetiva. Las herramientas estadísticas permiten recolectar, analizar e interpretar de forma inteligente los datos relevantes en el proceso de toma de decisión (CABRIA 1994).

Los modelos estadísticos se emplean actualmente en varios campos de negocio y la ciencia, permitiendo la predicción e identificación de los factores más influyentes para el éxito o el fracaso de los mismos. En los procesos de toma de decisiones las variables son numerosas y difíciles de medir y controlar por lo que las nuevas tecnologías resultan imprescindibles para encontrar un modelo que responda a las necesidades de cada negocio. En este sentido se trata de utilizar la estadística como herramienta diferenciadora respecto a la competencia para aproximarse a la solución que satisfaga las necesidades de la empresa y así crear una oportunidad que permita posicionarla de manera estratégica en el mercado (CABRIA 1994).

Al decir de (VILLEGAS 2005) la estadística para la toma de decisiones puede dividirse en *Estadística Descriptiva* que describe las características de una serie de datos pertenecientes a una población o muestra y en *Estadística Inferencial* que dado el desconocimiento de la población se permite realizar inferencias para la toma de decisiones basándose en la información contenida en una muestra al azar de la población entera. La Estadística Descriptiva constituye la base de la Estadística Inferencial, motivo por el cual se selecciona la primera como objetivo de la presente investigación.

1.3 Estadística Descriptiva

1.3.1 ¿Qué es la Estadística Descriptiva?

La Estadística Descriptiva se ocupa del estudio de datos, los cuales se disponen en la forma más conveniente para su análisis o inspección (SIRONVALLE 2000). También puede definirse como aquellos métodos que incluyen la recolección, presentación y caracterización de un conjunto de datos con el fin de describir apropiadamente las diversas características de ese conjunto (LEVINE et al. 2006).

En palabras de los autores de esta investigación, puede definirse como la ciencia que analiza una serie de datos y trata de extraer conclusiones sobre el comportamiento de los mismos.

1.3.2 Conceptos asociados a la Estadística descriptiva

Varios son los términos que se manejan cuando se hace referencia a la Estadística descriptiva entre ellos se pueden mencionar la *población* que consiste en la totalidad de los elementos bajo consideración, la *muestra* que es parte de la población que se selecciona para su análisis. De igual forma la *tabla de distribución de frecuencias* es la clasificación de los datos en clases o categorías de acuerdo a sus valores y la *marca de la clase* (x_i) es el punto medio del intervalo de clase y se considera representativo de los datos en dicha clase.

Los *fenómenos aleatorios* son aquellos en que las mismas causas dan lugar a resultados diferentes y los *experimentos aleatorios* son las experiencias cuyos resultados dependen del azar, es decir, pueden variar cuando estas se repiten en condiciones supuestas, por otra parte el *resultado* se considera la información aportada por la realización de una experiencia. El conjunto de todos los resultados posibles de un experimento se llama *espacio muestral* y se designa por Ω (SIRONVALLE 2000).

Según el propio (SIRONVALLE 2000) también se encuentran conceptos a los que se asocia una fórmula para obtener su valor tales como los siguientes:

Frecuencia (f_i): Es el número de datos de la muestra que se corresponden con la clase en cuestión. En la determinación de la frecuencia es realizado un conteo del número de elementos presentes en la muestra, identificando aquellos datos que caen dentro del intervalo indicado por los límites de la clase (JUÁREZ 2003).

Frecuencia acumulada (F_i): Es el número de datos en la muestra cuyo valor es menor que el límite superior de la clase en cuestión. Se calcula contabilizando las frecuencias observadas en la clase de interés y las anteriores, se define de la siguiente manera:

$$F_i = \sum_{j=1}^i f_j$$

F_i : Frecuencia acumulada; f_j : Frecuencia.

Frecuencia relativa (f_i^*): Es la proporción de los datos en la muestra que pertenecen a la clase en cuestión. Si se denota por n al número de datos en la muestra e i como el número de la clase, la frecuencia relativa se expresa a través de la siguiente fórmula:

$$f_i^* = \frac{f_i}{n} = \frac{f_i}{\sum_{j=1}^m f_j}$$

f_i^* : Frecuencia relativa; $f_i = f_j$: Frecuencia; n : Número de datos en la muestra.

Frecuencia relativa acumulada (F_i^*): Es la proporción de los datos en la muestra que son menores que el límite superior de la clase en cuestión y se define de la siguiente forma:

$$F_i^* = \frac{F_i}{n} = \frac{F_i}{\sum_{j=1}^m f_j}$$

F_i^* : Frecuencia relativa acumulada; F_i : Frecuencia acumulada; n : Número de datos en la muestra; f_j : Frecuencia.

1.3.3 Parámetros fundamentales de la Estadística descriptiva

Un parámetro estadístico desde un enfoque descriptivo es considerado una medida poblacional, es una variable que define una familia en determinados modelos. Los parámetros estadísticos son consecuencia del objetivo fundamental de la Estadística: crear un modelo de la realidad, los mismos permiten tener una idea global de la población, realizar estimaciones desconocidas, sobre datos, de la misma y a tomar decisiones. Entre los parámetros fundamentales se definen los de tendencia central y las medidas de dispersión (ROSS 2007).

Los parámetros de *tendencia central* son aquellos que informan sobre los valores medios de la serie de datos y entre ellos clasifican los siguientes (SIRONVALLE 2000):

Media aritmética: Es el valor medio ponderado de la serie de datos. Dada la muestra $M = \{x_1, x_2, \dots, x_n\}$, la medida típica más comúnmente utilizada es la media definida por:

$$\bar{x} = \frac{x_1 + x_2 + \dots + x_n}{n} = \frac{1}{n} \cdot \sum_{i=1}^n x_i$$

\bar{x} : Media aritmética; n : Número de datos en la muestra; x_i : Valores de la muestra.

Mediana: Es el valor de la serie de datos que se sitúa justamente en el centro de la muestra una vez ordenada. Se define:

Si la cantidad de elementos en la serie de datos es impar: $M = y_{\frac{n+1}{2}}$

Si la cantidad de elementos en la serie de datos es par: $M = \frac{y_{\frac{n}{2}} + y_{\frac{n}{2}+1}}{2}$

M: Mediana; **n**: Cantidad de elementos de la serie de datos; **y**: Posición del elemento en la serie de datos.

Moda: Valor que más se repite en la muestra.

Las *medidas de dispersión* estudian la distribución de los valores de la serie, analizando si estos se encuentran más o menos concentrados. O más o menos dispersos y se mencionan a continuación los tenidos en cuenta para la presente investigación (SIRONVALLE 2000):

Varianza: La varianza mide la distancia existente entre los valores de la serie de datos y la media. Se define como:

$$\sigma^2 = \frac{1}{n} \sum_{i=1}^n x_i - \bar{x}^2$$

σ^2 : Varianza; **n**: Número de valores de la muestra; x_i : Media aritmética; \bar{x} : valores de la muestra.

Desviación estándar. La desviación estándar de una variable aleatoria X se define como la raíz cuadrada positiva de su varianza; se calcula como: $\sigma = \sqrt{\sigma^2}$.

σ : Desviación estándar; σ^2 : Varianza.

Utilidad e importancia de la Estadística descriptiva

A partir de la panorámica mostrada de los principales elementos que conforman la Estadística Descriptiva, es importante agregar que la misma permite obtener información, analizarla, elaborarla y simplificarla con el objetivo de que pueda ser interpretada rápida y cómodamente y así pueda utilizarse con eficiencia y agilidad.

Para poder obtener conclusiones de una determinada población a partir de la información contenida en la muestra es necesario primero conocer el comportamiento del conjunto de datos, y esto es posible mediante la Estadística (JUÁREZ 2003).

Así también se puede decir que todos los campos de la investigación científica se benefician del análisis estadístico ya que, al decir de (MINOTTA 2006), las técnicas Estadísticas se pueden utilizar en casi todos los campos de la vida, por ejemplo, en las campañas electorales, en la prestación de servicios, entre otros.

Los responsables de la toma de decisiones sobre la política económica, asesores presidenciales, ministeriales y de otros altos cargos públicos, tienen en la Estadística una herramienta muy valiosa. Los economistas consideran varios índices de la situación económica durante un determinado periodo de tiempo y utilizan la información para predecir la situación económica futura. Únicamente con la ayuda del análisis estadístico pueden tomarse decisiones inteligentes en relación con los tipos tributarios, programas sociales, gastos de defensas, políticas laborales, inversiones prioritarias (MINOTTA 2006).

1.4 Los SIG

1.4.1 Definiciones principales de SIG

Un SIG es un elemento que permite analizar, presentar e interpretar hechos relativos a la superficie terrestre. Es un conjunto de software y hardware diseñado específicamente para la adquisición, mantenimiento y uso de datos cartográficos (TOMLIN 1990).

Otro concepto los sitúa como sistemas de información diseñados para trabajar con datos referenciados mediante coordenadas espaciales o geográficas. En otras palabras, un SIG es tanto un sistema de Base de Datos con capacidades específicas para datos georeferenciados, como un conjunto de operaciones para trabajar con esos datos. En cierto modo, un SIG es un mapa de orden superior (STAR and ESTES 1990).

Los SIG son sistemas computacionales que permiten consultar de manera interactiva información geográfica digital (latitud, longitud, altitud), facilitando la combinación e integración de múltiples cartografías, manejadas como capas superpuestas de datos digitales que se observan simultáneamente y como características de un mismo espacio, para la generación de información aplicable a proyectos o cuestiones específicas (REYNA 2005)

Al decir de los autores, los SIG son sistemas informáticos con múltiples componentes, todos creados con el fin de facilitar la representación de datos georeferenciados, además, estos sistemas cuentan con la capacidad de integrar información.

1.4.2 Partes fundamentales de un SIG

Un SIG está compuesto de tres subsistemas fundamentales:

1. Subsistema de datos, que permite la entrada, salida y gestión de datos, así como el acceso a los mismos por parte de los restantes subsistemas.
2. Subsistema de visualización y creación cartográfica, que permite la interacción con los datos mediante mapas, leyendas, así como la edición de dichos datos.
3. Subsistema de análisis, que provee los métodos y procesos para el análisis de los datos.

Además cuenta con cinco elementos básicos:

1. Datos: son los que propician toda la información geográfica necesaria para la creación de un SIG.
2. Métodos: son las acciones aplicadas sobre los datos, para llevar a cabo el análisis pertinente respecto a los mismos.
3. Software: es el instrumento que garantiza la implementación de los métodos y gestiona los datos.
4. Hardware: es el equipo necesario para ejecutar el software.
5. Personas: es el equipo informático capaz de diseñar y utilizar el software (OLAYA 2010).

1.4.3 Utilidad e importancia de los SIG

Los SIG permiten hacer un análisis del terreno en los más diversos ámbitos. Son herramientas versátiles, que pueden ser aplicados en cualquier actividad que requiera un componente espacial.

La tecnología de los SIG puede ser utilizada para investigaciones científicas, para la gestión de los recursos y activos, en arqueología, en evaluación del impacto ambiental, en cartografía, sociología, marketing, entre otros.

Los SIG se están convirtiendo en herramientas indispensables en la toma de decisiones en las que la información espacial tiene una especial relevancia. De alguna de estas decisiones depende en muchos casos el éxito o el fracaso de un negocio, o bien, la mejora considerable de la productividad de una empresa. Teniendo en cuenta esto resulta fácil comprender la relevancia que estas tecnologías están adquiriendo para el mundo empresarial (SIGCEA 2010).

Esta gran diversidad está dada porque el 80% de la información necesaria para realizar las actividades de cualquier organización está relacionada con la geografía ya que más de las tres cuartas partes de las actividades realizadas por el hombre pueden ser referenciadas a través de coordenadas geográficas. Si se acepta esta máxima se puede deducir la importancia que tiene el software capaz de gestionar y analizar la información espacial, en suma los datos geográficos (GARRIDO 2003).

El punto de partida de los SIG en América Latina fue la I Conferencia Latinoamericana de Informática en Geografía entre el 7 y el 9 de julio de 1987 en San José de Costa Rica donde muchos profesionales académicos de la región tomaron contacto inicial con las tecnologías digitales de automatización geográfica. El evento fue auspiciado por la Unión Geográfica Internacional (UGI) y las universidades de los países centrales participantes propiciaron la primera transferencia tecnológica a los países de América Latina (BUZAI and ROBINSON 2011).

Tal es la importancia y el interés que estos sistemas han generado que se crea en 1991 la Sociedad Iberoamericana de Sistemas de Información Geográfica (SIBSIG) encargada de organizar la Conferencia Iberoamericana de Sistemas de Información Geográfica (CONFIBSIG), que constituye la reunión científico-tecnológica de mayor alcance de la especialidad en América Latina y se desarrolla con un espacio de 2 años desde la fecha (BUZAI and ROBINSON 2011).

Los usuarios de los SIG en Cuba son en su mayoría geólogos, cartógrafos, geógrafos, arquitectos, ingenieros y gestores a diferentes niveles, que conocen, aplican y operan SIG en sus investigaciones y proyectos. Pese a las limitaciones de toda índole, el desarrollo de los especialistas cubanos en el campo de los SIG no se ha detenido, por el contrario, Cuba ha sido pionera en el desarrollo de estos sistemas siendo el único país del tercer mundo en la región que ha logrado desarrollar SIG propios, implementados y puestos en marcha por especialistas y para especialistas cubanos (CELADA *et al.* 2006).

1.5 La plataforma GeneSIG

1.5.1 Descripción general de GeneSIG

La plataforma GeneSIG es una herramienta que permite realizar representaciones y análisis de información referenciada geográficamente. Posee una estructura basada en plugins, lo que la convierte en una plataforma con un alto grado de interoperabilidad debido a que permite agregar o quitar componentes de manera sencilla. De igual modo permite separar los servidores de Bases de Datos y web en dos estaciones de trabajo diferentes, balanceando la carga del sistema, aumentando su disponibilidad y disminuyendo la posibilidad de fallas.

El tema de seguridad en GeneSIG es primordial, por lo que incluye un módulo encargado del tratamiento de la información teniendo en cuenta criterios de roles y permisos sobre los recursos que maneja. Es un producto totalmente desarrollado y logrado sobre software libre que permite la personalización de sus funcionalidades a cualquier negocio que lo requiera a través de la reutilización de sus componentes. Brinda servicios de georeferenciación y localización de recursos, así como la inclusión de datos y ubicación de nuevos objetos sobre mapas.

Cumple y respeta la estructuración y el diseño propuestos por CartoWeb, pero solo utiliza los paquetes a los cuales le realizará cambios o aportes funcionales. Presenta también un abanico bastante completo de características propias de un geoportal, con posibilidad de ir añadiendo o desarrollando nuevos plugins. Y es precisamente a través de estos plugins agregados de forma convencional que GeneSIG posee un amplio conjunto de funcionalidades, que actúan como herramientas de la misma plataforma y le brindan la posibilidad de ser altamente modular y escalable (ESPINOSA *et al.* 2012).

Entre sus principales módulos se encuentran:

- Módulo de Navegación.
- Módulo de Selección.
- Módulo de Consulta Espacial.
- Módulo de Análisis.
- Módulo de Configuración del Mapa.
- Módulo de Impresión.

- Módulo de Catálogo.
- Módulo de Servicios.
- Módulo de Edición.
- Módulo de Ayuda.

La arquitectura de la plataforma GeneSIG es una arquitectura distribuida, empleando como base cartográfica una información certificada por especialistas que laboran en su desarrollo, y sobre ella un conjunto de objetos representados geoespacialmente que contienen información asociada. El sistema se basa en la arquitectura cliente–servidor sobre plataforma Web, donde cada instancia del sistema en el cliente es independiente de la ejecución de otra.

Como parte de su organización interna, el sistema cuenta con tres capas lógicas: la capa de interfaz, la capa de negocio y la capa de datos.

- Interfaz: En esta capa están implementadas todas las interfaces gráficas con las que interactúa el usuario y las interfaces de interacción con otros sistemas. Estas interfaces se relacionan directamente con los módulos que se encuentran implementados en la capa de negocio.
- Negocio: En esta capa están incluidas todas las tareas y funcionalidades que realiza la plataforma e incluye al servidor de mapas MapServer.
- Datos: En esta capa se encuentran las Bases de Datos con las que trabaja la plataforma (una para datos cartográficos y otra para la información socio-económica, de configuración y los usuarios).

Es altamente modular y se emplea en el proyecto donde radica el equipo de desarrollo, donde se encuentra además abundante bibliografía, manuales de uso y personal con vasta experiencia en su uso (ESPINOSA *et al.* 2012).

1.5.2 Principales funcionalidades

1. Permite la representación geoespacial de la información asociada a cualquier negocio que lo requiera.
2. Proporciona servicios de acceso a la información geográfica, para su consulta, análisis y visualización, mediante una interfaz de usuario sencilla y de fácil manejo, que pueda ser utilizada por usuarios no especializados en tecnología SIG.

3. Integra la información socioeconómica existente (recursos humanos, activos fijos, entidades de servicios, lugares de interés) con la información geográfica asociada (ESPINOSA *et al.* 2012).

El producto GeneSIG está desarrollado sobre las siguientes tecnologías y herramientas:

- Lenguajes de desarrollo: PHP 5 y JavaScript.
- Entornos Integrados de Desarrollo: NetBeans 8.
- Sistema Gestor de BD: PostgreSQL 9.1.
- Sistemas Operativos: Ubuntu 12.04.
- Control de Versiones: RapidSVN.
- Herramienta de Modelado: Visual Paradigm 5.0.
- Frameworks: CartoWeb, ExtJS 3.0.
- Servidor de Mapas: MapServer 6.0.
- Servidor Web: Apache 2.5.6.

1.6 Estudio de algunas soluciones existentes

GvSIG

Proyecto de desarrollo de SIG en software libre, que incluye principalmente las aplicaciones gvSIG Desktop y gvSIG Mobile. La primera aplicación de las antes mencionadas se distribuye bajo la licencia GNU GPL v2. Permite el acceso a información vectorial y rasterizada así como a servidores de mapa. Está desarrollado en Java para los sistemas operativos Microsoft Windows, Linux y Mac OS X.

Es un SIG diseñado para capturar, almacenar, manipular, analizar y desplegar cualquier tipo de información geográfica referenciada con el fin de resolver problemas complejos de planificación y gestión. Proporciona las herramientas SIG más comunes como la carga de datos, mapa de navegación, mapa de consulta de información, función de selección, tablas de datos con Estadísticas, entre otros (GVSIG 2015).

En la investigación *Uso de SIG para la toma de decisiones* se evidencia el uso de gvSIG para la elaboración de los mapas temáticos y posteriormente se realizó un análisis estadístico y gráfico haciendo uso del lenguaje R (STANECKA and DÍAZ 2013).

La solución que se analiza propone el empleo de varias funciones estadísticas como por ejemplo máximo, mínimo, media, entre otras, calculando las mismas a partir de las capas del mapa. Los elementos antes mencionados son tenidos en cuenta para la solución del problema de investigación planteado, así como la interfaz gráfica que posee gvSIG.

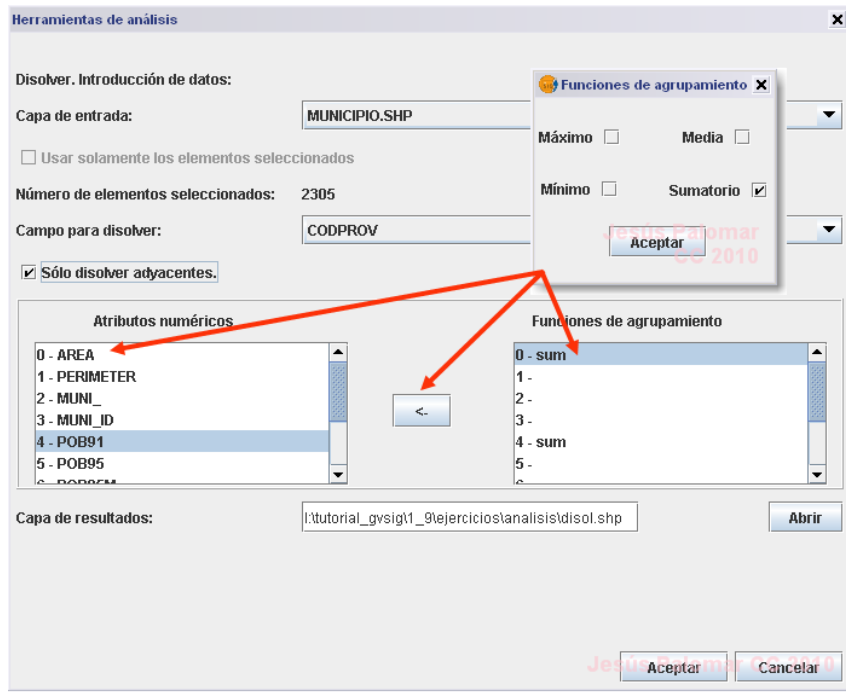


Figura 1: Análisis Estadístico en gvSIG.

Quantum GIS

Es un SIG de código libre para plataformas GNU/Linux, Unix, Mac OS y Microsoft Windows. Permite manejar formatos raster y vectoriales a través de las bibliotecas GDAL y OGR, así como Base de Datos. Está desarrollado en C++ y emplea el marco de trabajo Qt para la interfaz gráfica de usuario. Permite la integración de plugins tanto en C++ como en Python.

Algunas de sus características son: soporte para la extensión espacial PostGIS de PostgreSQL; manejo de archivos vectoriales Shapefile, ArcInfo coverages, Mapinfo, GRASS GIS, entre otros; soporte para un importante número de tipos de archivos raster (GRASS GIS, GeoTIFF, TIFF, JPG, entre otros). Quantum GIS incluye muchas herramientas estándar para análisis estadísticos útiles para el entendimiento de las Estadísticas espaciales que a su vez permiten entender un conjunto dado de datos vectoriales. Además

proporciona la opción de calcular las Estadísticas básicas de una determinada capa, motivo por el cual se tiene en cuenta para la presente investigación, entre las funciones estadísticas que presenta se encuentran, la media, mediana, máximo y mínimo (PROJECT 2014).

Quantum GIS no se emplea para dar solución al problema planteado a pesar de ser una tecnología libre ya que no es compatible con la arquitectura de GeneSIG. Se tuvieron en cuenta algunas características como la entrada y salida de datos en la interfaz de la aplicación.

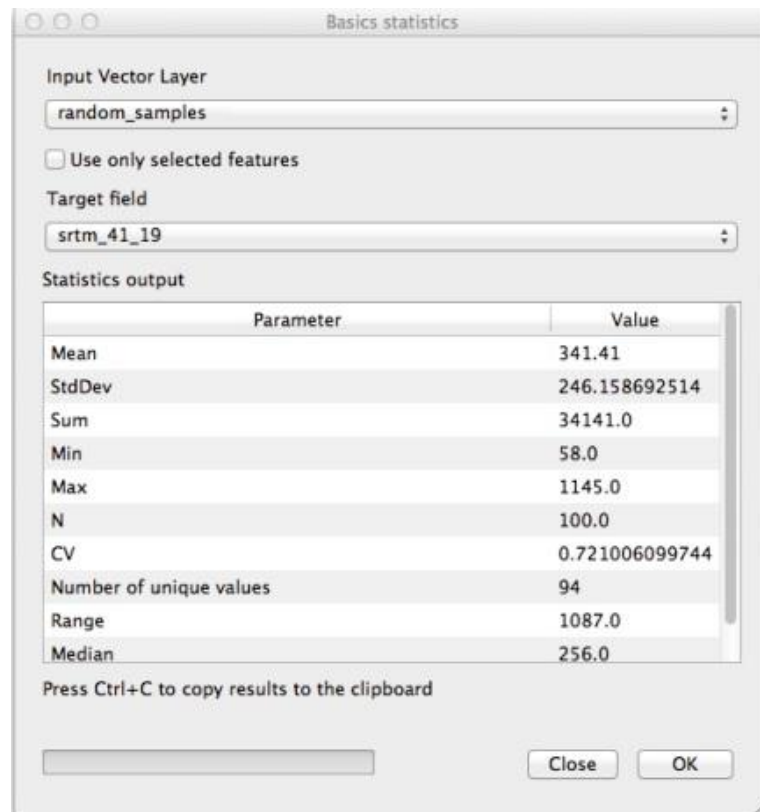


Figura 2: Cálculo de la Estadística Básica en Quantum GIS.

ArcGIS

ArcGIS es una completa plataforma de información que permite crear, analizar, almacenar y difundir datos, modelos, mapas y globos en 3D, poniéndolos a disposición de todos los usuarios según las necesidades de la organización. Como sistema de información, ArcGIS es accesible desde clientes de escritorio, navegadores web, y terminales móviles que se conectan a servidores de departamento,

corporativos o con arquitecturas de computación en la nube. Para los desarrolladores, ArcGIS proporciona herramientas que les permitirán crear sus propias aplicaciones.

Producido y comercializado por ESRI, posee herramientas como ArcMap, Maplex, Network Analyst, Schematics, Spatial Analyst, Tracking Analyst, ArcScan y Geostatistical Analyst que es una completa suite de modelos y herramientas estadísticas para la exploración de datos y la generación de superficies. Utilizando esta extensión se pueden crear superficies de predicción Estadísticamente válidas a partir de un número limitado de datos y mediciones. Dicha extensión cuenta con tres componentes, los mismos son:

- Conjunto de gráficas de análisis exploratorio de datos espaciales.
- Asistente Geoestadístico.
- Caja de herramientas de análisis geoestadístico que alberga herramientas específicamente diseñadas para ampliar las capacidades del Asistente Geoestadístico y permitir un análisis más detallado de las superficies que genera (ESRI 2015).

ArcGIS constituye una tecnología privada, lo que resulta en un inconveniente a la hora de su análisis y utilización, por lo cual, solo se toman ideas de como representar gráficamente los parámetros estadísticos definidos, a través de su interfaz gráfica.

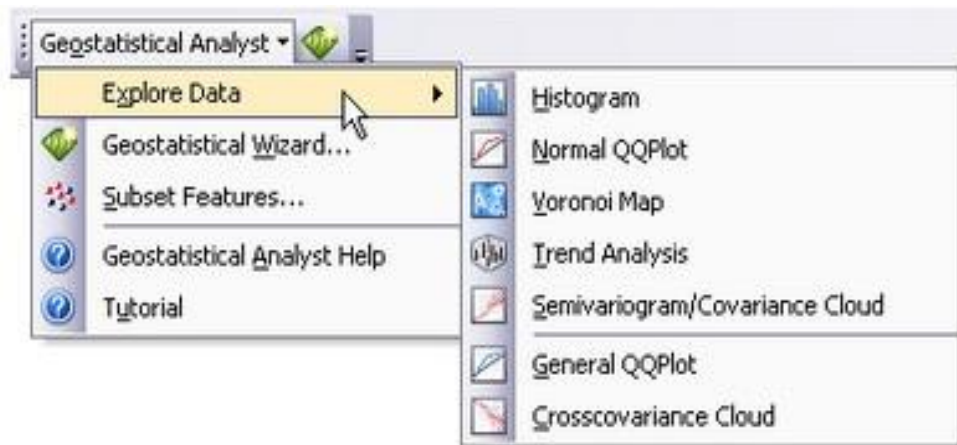


Figura 3: Módulo de análisis estadístico de ARCGIS.

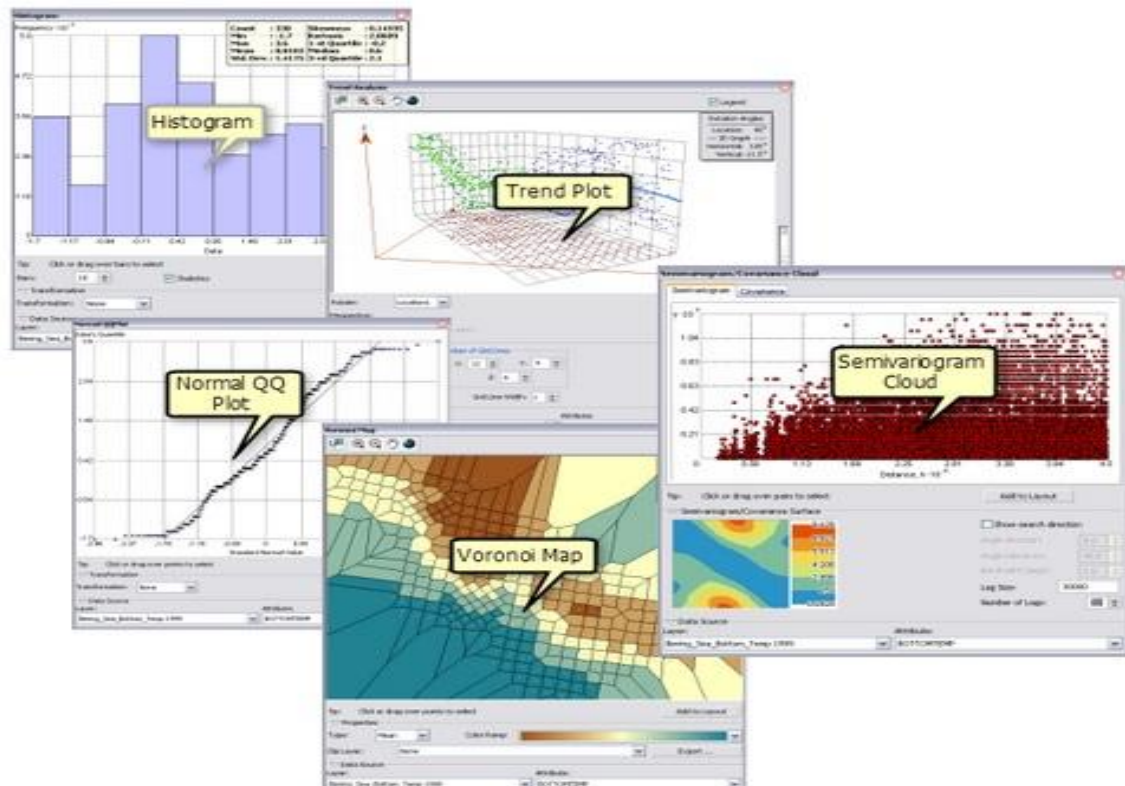


Figura 4: Salida gráfica del análisis exploratorio de datos en ARCGIS.

1.7 Conclusiones

A pesar de existir varios SIG en el mundo que proveen análisis estadístico de los datos, no se adecuan a las tecnologías definidas en la LPS Aplicativos SIG con que fue desarrollada GeneSIG, es por ello que se decide realizar el módulo de Estadística para la mencionada plataforma.

No obstante lo anterior, se considerarán las funcionalidades referidas a la Estadística descriptiva presentes en las soluciones, haciendo mayor énfasis en las que tienen en común como la media aritmética, el máximo y el mínimo.

Como resultado del análisis de las soluciones existentes se decidió emplear el lenguaje de tratamiento estadístico R.

CAPÍTULO 2: HERRAMIENTAS Y TECNOLOGÍAS A UTILIZAR

2.1 Introducción

Un sistema con calidad se obtiene, entre otras, a partir de la selección y utilización eficaz de varias herramientas y tecnologías. En el presente capítulo se describen las herramientas y tecnologías que se utilizan para obtener la solución propuesta. Se argumenta la utilización de la metodología de desarrollo de software, lenguajes de programación, herramienta CASE, lenguaje de modelado, frameworks y otros.

2.2 Metodología de desarrollo de Software

En la actualidad el software de computadora es la tecnología individual más importante en el ámbito mundial. A medida que la importancia del software crece, la comunidad del software ha intentado de manera continua desarrollar tecnologías que hagan más fácil, más rápida y menos cara la construcción y el mantenimiento de programas de computadora de alta calidad (PRESSMAN 2010).

Para la obtención de un software exitoso y con la calidad requerida es necesaria la selección de una metodología de desarrollo, la cual provee procedimientos, técnicas, herramientas y un soporte documental que ayuda a los desarrolladores a realizar un nuevo software. Las metodologías se clasifican en dos grupos conocidos como las ágiles y las tradicionales; éstas últimas se focalizan en la documentación, planificación y procesos (FIGUEROA *et al.* 2013).

2.2.1 El Proceso Unificado de Desarrollo

El proceso unificado de desarrollo (del inglés Rational Unified Process, RUP) es un proceso de desarrollo de software, o sea, el conjunto de actividades necesarias para transformar los requisitos de un usuario en un sistema software. Es considerado un marco de trabajo genérico que puede especializarse para una gran variedad de sistemas software. Emplea el Lenguaje Unificado de Modelado (UML) el cual es una parte esencial de RUP. Se define por tres aspectos claves: dirigido por Casos de Uso, centrado en la arquitectura, e iterativo e incremental (JACOBSON *et al.* 2000).

- *Dirigido por Casos de Uso:* Un Caso de Uso es un fragmento de funcionalidad del sistema que proporciona al usuario un resultado importante. Los Casos de Uso representan los requisitos funcionales del sistema. También guían su diseño, su implementación y el proceso de pruebas. En fin, el proceso de desarrollo avanza a través de una serie de flujos de trabajo que parten de los Casos de Uso.
- *Centrado en la arquitectura:* La arquitectura es una vista del diseño completo con las características más importantes resaltadas, dejando de lado los detalles. Debe existir interacción entre la arquitectura y los Casos de Uso, deben evolucionar en paralelo.
- *Iterativo e incremental:* El desarrollo de un producto de software puede durar un año o más. Es práctico dividir el trabajo en mini proyectos. Cada mini proyecto es una iteración que termina en un incremento. Las iteraciones hacen referencia a pasos en el flujo de trabajo y los incrementos al crecimiento del producto.

Una de las ventajas fundamentales de RUP radica en que funciona bien en proyectos de innovación y es sencillo ya que sigue los pasos intuitivos a la hora de desarrollar el software (FIGUEROA *et al.* 2013).

2.3 Lenguaje de modelado: UML

Según (LARMAN 1999) UML es un lenguaje para especificar, visualizar, construir y documentar los artefactos de los sistemas software, así como para el modelado del negocio. (ORALLO, 2003) lo considera un lenguaje que permite modelar lo que el software deberá hacer; posibilitando un mayor entendimiento y comprensión del sistema por parte del equipo de desarrollo.

Características y ventajas de UML:

- Es un lenguaje de modelado visual de propósito general orientado a objetos.
- Agrupa notaciones y conceptos provenientes de distintos tipos de métodos orientados a objetos.
- Posee vocabularios y reglas que se centran en la representación conceptual y física de un sistema, para crear y leer modelos bien formados y además constituyen los planos de un sistema de software.
- Es independiente del proceso de desarrollo; su uso óptimo se consigue en procesos como es el caso de RUP.
- Se basa en una notación gráfica concisa y fácil de aprender y utilizar.

Sus objetivos son varios, pero se pueden resumir en sus funciones:

- Visualizar: Posibilita expresar de una forma gráfica un sistema, de manera que pueda ser entendido fácilmente.
- Especificar: UML permite especificar cuáles son las características de un sistema antes de su construcción.
- Construir: Permite, a partir de los modelos especificados construir cada uno de los sistemas diseñados.
- Documentar: Los propios elementos gráficos sirven como documentación del sistema desarrollado, los que pueden ser empleados en su futura revisión.

2.4 Visual Paradigm for UML

Es una herramienta CASE (del inglés Computer Aided Software Engineering) que emplea UML como lenguaje de modelado. Propicia un conjunto de ayudas para el desarrollo de programas informáticos, desde la planificación, a través de la representación de todo tipo de diagramas. Proporciona la posibilidad de realizar diagramas de clases pasando por el análisis y el diseño, hasta la generación del código fuente de los programas y la documentación. Visual Paradigm ha sido concebida para soportar el ciclo de vida completo del proceso de desarrollo del software (PRESSMAN 2008).

Entre sus características se encuentran las siguientes:

- Entorno de creación de diagramas para UML 2.0.
- Diseño centrado en Casos de Uso y enfocado al negocio, lo cual genera un software con mayor calidad.
- Uso de un lenguaje estándar común a todo el equipo de desarrollo, lo cual facilita considerablemente la comunicación.
- Disponibilidad de múltiples versiones, para cada necesidad.
- Disponibilidad en múltiples plataformas.

Alguna de las ventajas que proporciona esta herramienta son la corrección sintáctica y la coherencia entre diagramas (PARADIGM 2015).

2.4.1 Visual Paradigm for UML 8.0

Fue lanzado en febrero de 2010 por Visual Paradigm International Limited y fue empleado para la realización de los diagramas que describen el ciclo de vida del sistema, creando un diseño centrado en Casos de Uso de acuerdo a lo planteado por RUP (PARADIGM 2015).

2.5 Lenguajes de programación

Un lenguaje de programación es un lenguaje artificial que puede ser utilizado para controlar el comportamiento de una computadora. Estos se componen de un conjunto de reglas sintácticas y semánticas que permiten expresar instrucciones que luego serán interpretadas.

Debe distinguirse de lenguaje informático que es una definición más amplia. Los lenguajes de programación pueden clasificarse según el paradigma que usan en procedimentales, orientado a objetos, funcionales, lógicos, híbridos (ALEGSA 1998 - 2015).

2.5.1 PHP

Es un lenguaje de código abierto muy popular, adecuado para desarrollo web y que puede ser incrustado en HTML. Es popular porque hay un gran número de páginas y portales web desarrollados con PHP. Código abierto significa que es de uso libre y gratuito para todos los programadores que quieran usarlo. Incrustado en HTML significa que en un mismo archivo se puede combinar código PHP con HTML. El lenguaje PHP se procesa en servidores, que son potentes ordenadores con un software y hardware especial (GONZÁLEZ 2015).

Dentro de sus principales características se encuentran:

- Implementa el paradigma orientado a objetos mediante la utilización de clases las cuales poseen propiedades y métodos y pueden ser instanciadas.
- El código PHP, el cual generalmente define la lógica del funcionamiento interno de un sistema, es invisible al navegador web pues el intérprete PHP es encargado de ejecutarlo y enviar solo el resultado HTML al cliente, esto hace que la programación sea segura y confiable.

- Posee capacidad de conexión con varios gestores de Bases de Datos como son MySQL y PostgreSQL.
- Posee extensa y detallada documentación la cual puede ser consultada desde su página web (ACHOUR *et al.* 2006).

2.5.2 PHP MapScript

Es un módulo para PHP que permite acceder a la API (Application Programming Interface) de MapServer. Carga las capas configuradas en un mapfile al inicializar y permite modificar, cambiar e incluso agregar más capas.

Las librerías MapScript de PHP constituyen la vía de comunicación de las aplicaciones SIG con el servidor de mapas MapServer y por otra parte rompen en cierta medida ésta rigidez de la representación de mapas a través de los ficheros "Mymapfile.map" permitiendo modificar el mapfile en tiempo de ejecución, a los cuales se les conoce como mapfile dinámicos, facilitando de esta forma la creación de aplicaciones con un grado de personalización mayor (NOZAWA 2007).

2.5.3 ExtJS 3.0

ExtJS es una librería JavaScript que permite construir aplicaciones complejas en internet además de flexibilizar el manejo de componentes de la página como el DOM, Peticiones AJAX, DHTML, tiene la gran funcionalidad de crear interfaces de usuario bastante funcionales (FREDERICK *et al.* 2008).

Esta librería incluye componentes UI personalizables, modelo de componentes extensibles, un API fácil de usar y licencias de código abierto. Entre sus ventajas se cuentan:

- Código reutilizable.
- Independiente o adaptable a frameworks diferentes.
- Orientada a la programación de interfaces tipos desktop en la web.
- El API es homogenizado independientemente del adaptador usado. Los controles siempre se verán igual.
- Una extensa comunidad de Usuarios (PÉREZ 2011).

2.5.4 Python

Es un lenguaje de programación dinámico y multiparadigma, su principal objetivo es resultar fácil en cuanto a lectura y diseño. Es un proyecto de código abierto y posee soporte e integración con otros lenguajes y herramientas. Tiene integrada una amplia biblioteca estándar. Es fácilmente extensible con nuevas funcionalidades. Es un lenguaje ideal para scripting y para el desarrollo rápido de aplicaciones en diferentes áreas y plataformas (ROSSUM 2009).

Ofrece varias ventajas, tales como la rapidez en su desarrollo, su agilidad y sencillez, la utilización de robustas bibliotecas y el soporte para varias Bases de Datos.

2.5.5 Lenguaje R

R es un lenguaje de programación especialmente orientado al **análisis estadístico** y a la **representación gráfica** de los resultados obtenidos. Es un proyecto de código abierto, por lo tanto, los usuarios son **libres** de modificarlo y extenderlo (ROMERO-CAMPERO 2013). Surgió en 1991, fue creado por Ross Ihaka y Robert Gentleman. Entre sus ventajas se cuentan las siguientes: una gran similitud sintáctica con S-Plus, lo que ha facilitado la migración de muchos usuarios, sus frecuentes versiones (ya que es un lenguaje activo, con más de una liberación anual) y el hecho de que puede ejecutarse sobre cualquier sistema operativo convencional. R constituye uno de los más útiles a la hora de estructurar y manipular datos, especialmente cuando se trata de grandes colecciones de ellos (BOJAEN 2013).

2.6 IDE NetBeans 8.0

NetBeans IDE es un entorno de desarrollo, una herramienta para que los programadores puedan escribir, compilar, depurar y ejecutar programas. Está escrito en Java pero puede servir para cualquier otro lenguaje de programación. Existe además un número importante de módulos para extender el NetBeans IDE. Es un producto libre y gratuito sin restricciones de uso (NETBEANS 2015b).

NetBeans IDE 8.0 proporciona algunas ventajas como un gran conjunto de herramientas para PHP y mejoras en la compatibilidad con dicho lenguaje, además, su código se encuentra bien organizado (NETBEANS 2015a).

2.7 Sistema Gestor de Base de Datos

Un Sistema Gestor de Base de Datos (en lo adelante SGBD) es una agrupación de programas que sirven para definir, construir y manipular una Base de Datos (ALEGSA 1998 - 2015). (MARTÍNEZ 2013) los considera los tipos de software que se encargan de brindar un sistema de almacenamiento de datos de manera ordenada y transparente al usuario, que sirve de interfaz entre las Bases de Datos y las aplicaciones que la utilizan.

Según (BERTINO and MARTINO 1993), un SGBD es un sistema de software que permite la definición de Bases de Datos, así como la elección de las estructuras de datos necesarios para el almacenamiento y búsqueda de los datos, ya sea de forma interactiva o a través de un lenguaje de programación. Un SGBD relacional es un modelo de datos que facilita a los usuarios describir los datos que serán almacenados en la Base de Datos junto con un grupo de operaciones para manejar los datos.

2.7.1 PostgreSQL como SGBD

PostgreSQL es un SGBD Relacional y Orientado a Objetos, desarrollado en la Universidad de California. Es un gestor de Bases de Datos de código abierto, brinda un control de concurrencia multi-versión (del inglés multi version concurrent control, MVCC) que permite trabajar con grandes volúmenes de datos; soporta gran parte de la sintaxis SQL y cuenta con un extenso grupo de enlaces con lenguajes de programación.

Posee características significativas del motor de datos, entre las que se pueden incluir las subconsultas y los valores por defecto. El código fuente se encuentra disponible sin costo. Está disponible para 34 plataformas con la última versión estable. Posee una integridad referencial e interfaces nativas para lenguajes como C, C++, PHP y PYTHON. Funciona en todos los sistemas operativos como Linux y Windows. Debido a la liberación de la licencia, PostgreSQL se puede usar, modificar y distribuir de forma gratuita para cualquier fin, ya sea privado, comercial o académico (GROUP 1996 - 2015).

2.7.2 PostGIS: la extensión geográfica del gestor seleccionado

Es una extensión de datos espaciales para PostgreSQL. Añade soporte para objetos geográficos que permite consultas de ubicación para ser ejecutadas en SQL (RAMSEY *et al.* 2015).

PostGIS 1.5.2: Es una versión estable y liberada de PostGIS, extensión de PostgreSQL que la convierte en una Base de Datos Espacial. Posee soporte de datos raster y vectoriales. Es compatible con el servidor de mapas MapServer (RAMSEY *et al.* 2015).

2.8 PGAdmin III para el manejo de PostgreSQL

PGAdmin III es una plataforma de administración y desarrollo de código abierto para PostgreSQL. Está diseñado para responder a las necesidades de todos los usuarios, desde escribir simples consultas SQL hasta crear Bases de Datos complejas. La interfaz gráfica soporta todas las características de PostgreSQL y facilita su administración. La aplicación también incluye un resaltado de sintaxis SQL y un editor de código del lado del servidor, apoyo para el motor de replicación Slony (PGADMIN 2015).

La conexión con el servidor se puede hacer a través de TCP / IP y puede utilizar encriptado SSL para la seguridad. No se requieren controladores adicionales para comunicarse con el servidor de Base de Datos. Desarrollado por una comunidad de expertos de PostgreSQL en todo el mundo, está disponible en más de una docena de idiomas y publicado bajo la licencia PostgreSQL (POSTGRESQL 2012).

2.9 Servidor de mapas: MapServer 6.0

MapServer es un motor de procesamiento de datos geográficos, de código abierto y creado en C. Permite la creación de mapas de imágenes geográficas. Entre sus principales características cuenta con la automatización de los elementos del mapa (barra de escala, mapa de referencia y leyenda), soporte para lenguajes como PHP y Python y es multiplataforma.

Se empleó por la fortaleza que posee para el graficado de imágenes y se tiene en cuenta además que la plataforma GeneSIG también utiliza MapServer ya que trabaja con formatos raster y vectoriales, es rápido y ligero (MAPSERVER 2015).

2.10 Conclusiones parciales

La utilización de las herramientas propuestas facilita el trabajo y aporta rapidez y eficacia a lo largo del ciclo de vida del desarrollo, además, garantiza una mayor usabilidad de la aplicación. Se utilizan las herramientas y tecnologías que forman parte del ambiente de desarrollo y la arquitectura de la LPS.

CAPÍTULO 3: ANÁLISIS Y ARQUITECTURA DEL SISTEMA

3.1 Introducción

En este capítulo se describe la fase inicial de la solución propuesta, para ello se ofrece la descripción del respectivo modelo de dominio, así como los requisitos que debe cumplir el sistema para considerarse exitoso ante los usuarios finales. Se definen la arquitectura con sus patrones y estilos correspondientes establecidos por la LPS.

3.2 Modelo de dominio

El modelo de dominio representa las clases más significativas del sistema. Su objetivo fundamental es, a través del Diagrama de clases del dominio, facilitar la comprensión del contexto y de los requisitos del sistema. Se utilizan diagramas UML para su descripción (JACOBSON et al. 2000).

En la Figura 5 se presenta el modelo de dominio de la solución.

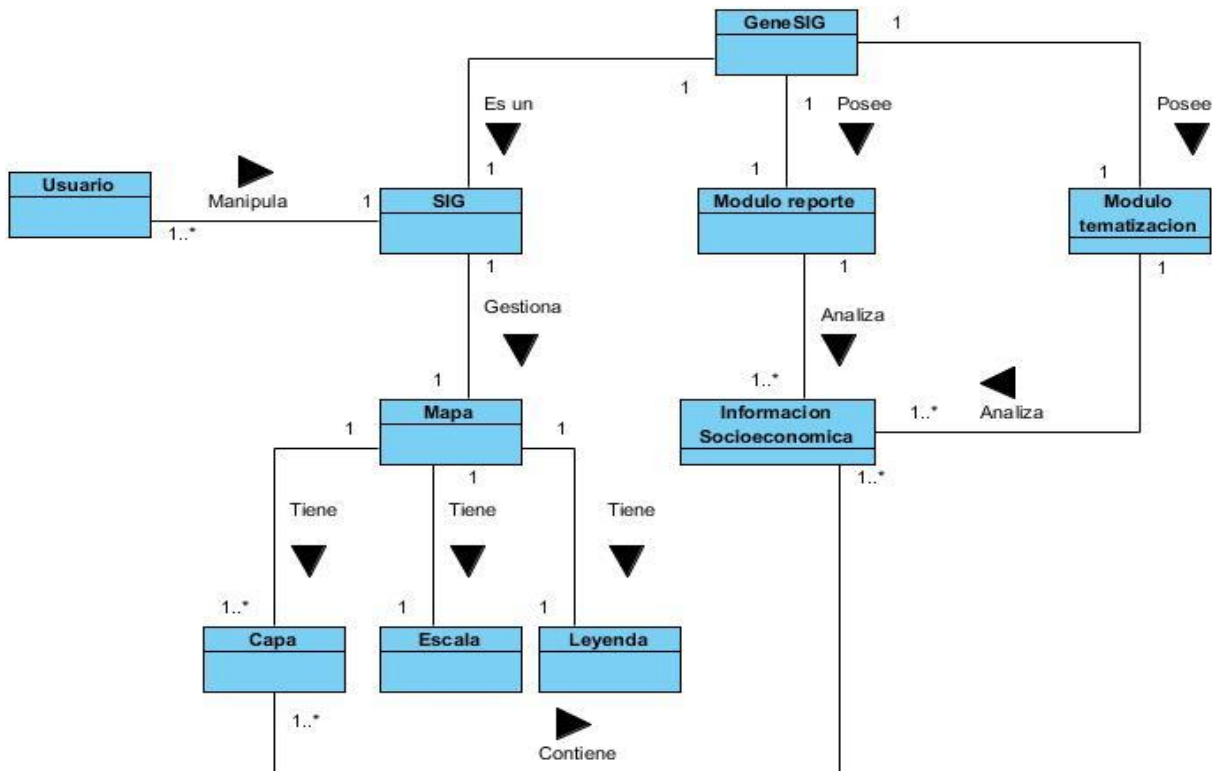


Figura 5: Modelo de Dominio.

La plataforma GeneSIG constituye un SIG que es manipulado por los usuarios. Como SIG gestiona mapas los cuales se conforman de capas, escalas y leyendas; las capas poseen información socioeconómica que es analizada mediante los módulos reporte y tematización que posee GeneSIG.

3.2.1 Definición de clases del diagrama del dominio

GeneSIG: Plataforma sobre la cual la LPS Aplicativos SIG desarrolla los SIG.

SIG: Sistemas de Información Geográfica desarrollados sobre la plataforma GeneSIG.

Usuario: Es la persona que interactúa con los SIG.

Mapa: Es la representación gráfica de una porción de territorio; tiene propiedades métricas y permite medir distancias, ángulos y otros.

Escala: Relación que existe entre las medidas de distancia expresadas en el mapa y las mismas de manera real; relación matemática entre las dimensiones en el mapa y la porción de terreno que representan.

Capa: La información temática que se usa para elaborar los SIG está estructurada mediante capas.

Leyenda: Descripción para entender las simbologías que aparecen en el mapa cuyo significado el usuario desconoce.

Información Socioeconómica: Es un conjunto de datos de carácter social o económico gestionados por el SIG referente al negocio en cuestión.

Módulo reporte: Permite el análisis de la información socioeconómica generando reportes según el parámetro definido por el usuario.

Módulo tematización: Permite la creación de capas temáticas en el mapa de acuerdo al método seleccionado.

3.3 Requisitos del Software

Según (PRESSMAN 2010) los requisitos de software constituyen las necesidades de los clientes, las funcionalidades que debe cumplir el sistema software y las restricciones que debe cumplir. Los mismos se clasifican en funcionales y no funcionales.

3.3.1 Requisitos funcionales

RF1 Calcular media aritmética: Esta funcionalidad permite calcular la media aritmética de una entidad o una capa (siempre que esta acción sea posible) luego de seleccionada por el usuario. Posteriormente permite tabular el resultado obtenido.

RF2 Calcular la mediana: Esta funcionalidad permite calcular la mediana de una entidad o una capa (siempre que esta acción sea posible) luego de seleccionada por el usuario. Posteriormente permite tabular el resultado obtenido.

RF3 Calcular la moda: Esta funcionalidad permite calcular la moda de una entidad o una capa (siempre que esta acción sea posible) luego de seleccionada por el usuario. Posteriormente permite tabular el resultado obtenido.

RF4 Calcular la desviación estándar: Esta funcionalidad permite calcular la desviación estándar de una entidad o una capa (siempre que esta acción sea posible) luego de seleccionada por el usuario. Posteriormente permite tabular el resultado obtenido.

RF5 Calcular la varianza: Esta funcionalidad permite calcular la varianza de una entidad o una capa (siempre que esta acción sea posible) luego de seleccionada por el usuario. Posteriormente permite tabular el resultado obtenido.

RF6 Calcular el máximo: Esta funcionalidad permite calcular el máximo de una entidad o una capa (siempre que esta acción sea posible) luego de seleccionada por el usuario. Posteriormente permite tabular el resultado obtenido.

RF7 Calcular el mínimo: Esta funcionalidad permite calcular el mínimo de una entidad o una capa (siempre que esta acción sea posible) luego de seleccionada por el usuario. Posteriormente permite tabular el resultado obtenido.

RF8 Mostrar el resultado de los cálculos en una gráfica: Esta funcionalidad permite representar a través de gráficas de barras, poligonal y circular, el resultado de los cálculos realizados.

RF9 Mostrar el resultado de los cálculos en el mapa: Esta funcionalidad permite representar mediante puntos específicos en el mapa el resultado de los cálculos realizados.

RF10 Insertar capa: Este requerimiento permite insertar una capa para poder seleccionar los atributos correspondientes a la misma.

RF11 Eliminar capa: Este requerimiento permite eliminar una capa, luego de haber sido analizada Estadísticamente o en caso de haber insertado la misma de manera errónea.

RF12 Insertar atributo: Esta funcionalidad permite añadir los atributos correspondientes a una capa previamente seleccionada a los que se le pueden calcular los parámetros estadísticos.

RF13 Eliminar atributo: Esta funcionalidad permite eliminar los atributos previamente guardados en la Base de Datos, correspondientes a una capa determinada.

3.3.2 Requisitos no funcionales

RNF1 De Usabilidad: El módulo podrá ser utilizado por cualquier usuario con conocimientos básicos de computación, SIG y Estadística descriptiva. Estará orientado a íconos para facilitar el reconocimiento de las funcionalidades por parte de los usuarios.

RNF2 De Fiabilidad: La información que se gestiona en el módulo estará protegida del acceso no autorizado.

RNF3 Restricciones de diseño: El diseño de la aplicación debe ser sencillo de manera tal que no sea necesario mucho entrenamiento a la hora de emplearla. La misma debe ser diseñada sobre una arquitectura cliente – servidor. Se debe lograr un producto configurable y extensible ya que será

desarrollado sobre la plataforma GeneSIG y debe permitir la incorporación de nuevas funcionalidades en el futuro.

RNF4 De Software:

Para las computadoras cliente:

- Un navegador como Mozilla Firefox 16.0 o superior.
- Sistema operativo: GNU/Linux, Windows 7 o superior.

Para los Servidores:

- Sistema operativo GNU/Linux Ubuntu Server 12.04.
- Servidor Web Apache 2.0 o superior, con módulo PHP 5 configurado con la extensión pgsql incluida.
- PostgreSQL 9.1 como Sistema Gestor de Base de Datos.
- PostGIS 1.5.2 como soporte de datos espaciales.
- MapServer 6.0, con extensión PHP MapScript.
- Python 2.7.
- R - base 2.14.1 como lenguaje de tratamiento estadístico.

RNF5 De Hardware:

Para las computadoras cliente:

- 512 MB de RAM como mínimo.
- 40 GB de capacidad en disco duro.
- 512 MHz de Procesador como mínimo.

Para los Servidores:

- 2 GB de RAM y 40 GB de disco duro para el Servidor de mapas como mínimo.
- 2 GB de RAM y 40 GB de disco duro para el servidor de Base de Datos como mínimo.
- 3.0 GHz de Procesador como mínimo.

RNF6 Restricciones de licencia: El módulo puede ser utilizado, modificado y distribuido, sin necesidad de obtener la autorización de sus respectivos titulares.

RNF7 Requisitos legales, de Derecho de autor y otros: El sistema debe cumplir con las leyes, decretos, resoluciones y manuales establecidos por la República de Cuba y la Dirección de Calidad de la UCI.

3.4 Modelo del sistema

El Modelo de Casos de Uso del Sistema ayuda al cliente, a los usuarios y a los desarrolladores para acordar cómo usar el sistema. Cada tipo de usuario se representa mediante un actor. Todos los actores y Casos de Uso conforman el Modelo de Casos de Uso del Sistema (CUS) (JACOBSON *et al.* 2000).

3.4.1 Actores del sistema

Los actores del sistema suelen ser personas, sistemas o hardware externo que interactúan con el sistema. En el desarrollo de la solución propuesta se definió el siguiente actor:

Tabla 1: Actores del Sistema.

Actor	Descripción
Usuario	Usuario con permisos para acceder a las funcionalidades del sistema.

3.4.2 Patrones de Casos de Uso del Sistema

Los patrones de CUS ayudan a describir lo que el sistema debe hacer, describen el uso del sistema y cómo interactúa con los usuarios. Son utilizados generalmente como plantillas que describen cómo deben ser estructurados y organizados los CUS. El uso de estos patrones constituye una buena práctica en el modelado de CUS (LARMAN 1999). En la presente investigación se emplearon para el modelado de los CUS los siguientes patrones:

Extensión concreta o Inclusión: Está dividido en dos partes: extensión concreta e inclusión. La extensión concreta consiste en una relación de extensión entre dos Casos de Uso como se evidencia en la Figura 6 en los Casos de Uso *Graficar parámetros estadísticos* y *Representar en el mapa los parámetros*

estadísticos, los cuales extienden del Caso base *Calcular parámetros estadísticos*. La inclusión se pone de manifiesto en la relación existente entre el Caso de Uso *Gestionar atributo* que está incluido en el Caso de Uso *Gestionar capa*.

CRUD Parcial: Es una variación del patrón CRUD Completo, se emplea para agrupar las operaciones en los Casos de Uso para administrar información. En la presente investigación se utiliza para modelar las funcionalidades añadir y eliminar las capas y los atributos.

3.4.3 Diagrama de Casos de Uso del Sistema

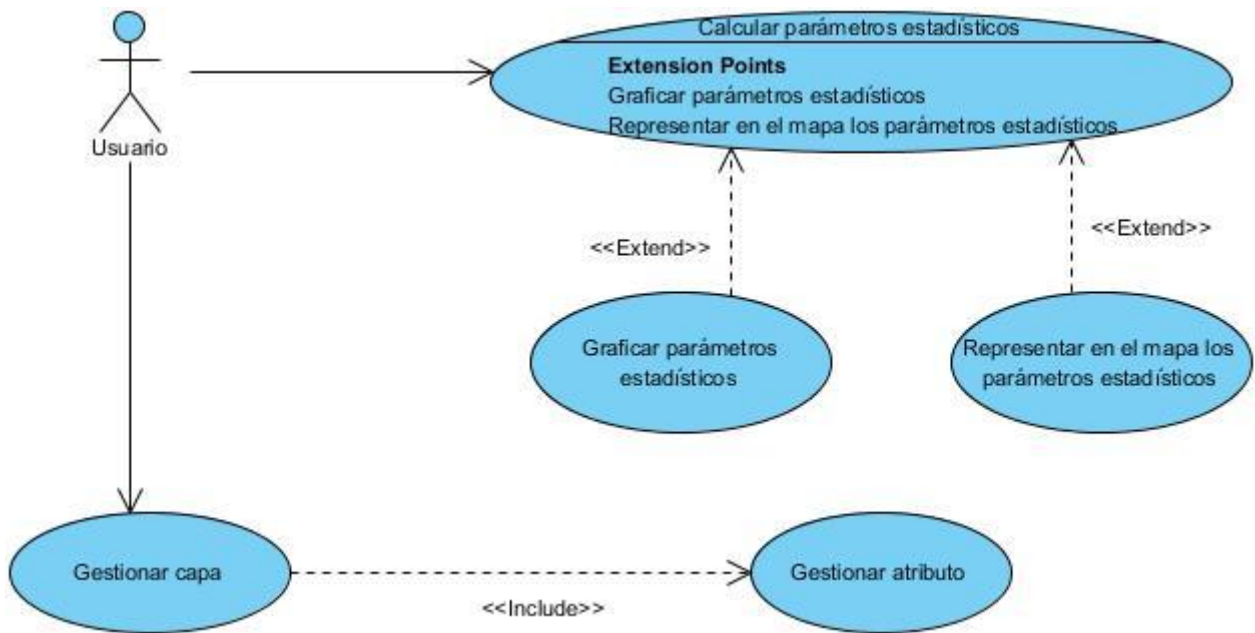


Figura 6: Diagrama de Casos de Uso del Sistema.

3.4.4 Descripción textual de los Casos de Uso del Sistema

La Descripción del Caso de Uso ayuda a identificar los objetos y operaciones en los sistemas. Cada Caso de Uso se describe utilizando una descripción en lenguaje natural. Esto ayuda a los diseñadores a identificar los objetos en el sistema y les permite comprender el comportamiento que tendrá el sistema (SOMMERVILLE 2005).

A continuación se muestra la descripción textual del Caso de Uso Calcular parámetros estadísticos. El resto de los Casos de Uso pueden ser consultados en el Anexo I.

Tabla 2: Descripción textual del Caso de Uso Calcular Parámetros estadísticos.

Caso de Uso:	Calcular parámetros estadísticos.	
Actores:	Usuario.	
Propósito:	Obtener los valores de los parámetros estadísticos.	
Resumen:	Este Caso de Uso se inicia cuando el usuario desea conocer el valor de determinado parámetro y finaliza cuando se tabula el resultado de dicho cálculo.	
Precondiciones:	El usuario debe haber guardado la configuración de las capas y los atributos previamente.	
Referencias:	RF1, RF2, RF3, RF4, RF5	
Prioridad:	Crítico.	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1. Selecciona la opción Funciones Estadísticas.	2. El sistema muestra la ventana con los campos de entrada capa y atributos, los parámetros estadísticos y las opciones de calcular, graficar y representar en el mapa.	
3. Selecciona una capa.	4. El sistema habilita la opción de seleccionar un atributo correspondiente a la capa seleccionada.	
5. Selecciona el atributo al que se le calculará los parámetros estadísticos.	6. El sistema habilita los parámetros estadísticos disponibles para ese atributo.	
7. Seleccionar el parámetro estadístico que se va a calcular.	8. El sistema muestra que la opción está seleccionada.	
9. Selecciona el botón calcular.	10. El sistema calcula el parámetro seleccionado y se muestra el resultado en una ventana que contiene una tabla con los campos función Estadística, capa, atributo y resultado. Además muestra los botones Graficar y Mapa.	
11. Selecciona la opción de cerrar la ventana que contiene la tabla.	12. El sistema oculta la ventana que contiene la tabla con los resultados del cálculo efectuado.	
	13. El sistema muestra la ventana Funciones Estadísticas.	
Flujo alterno de los eventos		
1. Cerrar ventana		
3. Selecciona la opción cerrar ventana.	4. Oculta la ventana de funciones Estadísticas.	
	5. El sistema muestra la interfaz principal de la	

	plataforma GeneSIG.	
2. Campos obligatorios vacíos		
3. El usuario no selecciona el valor de los campos obligatorios capa y atributo.	4. El sistema no habilita la opción siguiente (5) del flujo básico y muestra un mensaje de notificación.	
	5. El sistema muestra nuevamente la interfaz de las Funciones Estadísticas.	
Relaciones	CU incluidos	No procede
	CU extendidos	CU Graficar parámetros estadísticos. CU Representar en el mapa parámetros estadísticos.

Prototipo de Interfaz



Poscondiciones: El sistema guarda en la Base de Datos los resultados del cálculo de los parámetros.

3.5 Arquitectura de software

La arquitectura de un software es la estructura del sistema, que incluye los componentes del software, las propiedades visibles externamente de esos componentes y las relaciones entre ellos (PRESSMAN 2010).

Es importante definir una arquitectura de software dado que la misma permite la comunicación entre todas las partes interesadas en el desarrollo de la aplicación, también destaca las decisiones iniciales relacionadas con el diseño que tendrán un impacto profundo en todo el trabajo de la ingeniería del software y lo que también influye en el resultado final del software (PRESSMAN 2008).

El uso de la plataforma GeneSIG garantiza que las arquitecturas seleccionadas para regir el desarrollo del software son:

Basada en componentes: Se centra en el diseño y construcción de sistemas computacionales que utilizan componentes de software reutilizables. Define la composición de software como el proceso de construir aplicaciones mediante la interconexión de componentes de software a través de sus interfaces (REYNOSO and KICILLOF 2004).

Orientada a objetos: Los componentes del estilo se basan en principios orientados a objetos: encapsulamiento, herencia y polimorfismo. Las interfaces están separadas de las implementaciones. Las representaciones de los datos y las operaciones están encapsuladas en un tipo abstracto de datos u objetos. La comunicación entre componentes es a través de mensajes (REYNOSO and KICILLOF 2004).

3.5.1 Patrones arquitectónicos y de diseño

Un patrón arquitectónico impone una transformación en el diseño de una arquitectura, algunos de sus elementos fundamentales son: su alcance se enfoca en un aspecto determinado de la arquitectura, impone una regla sobre la arquitectura puesto que describe la manera en la que el software manejará algún aspecto de su funcionalidad a nivel de infraestructura, abarcan aspectos específicos dentro del contexto de la arquitectura. Estos patrones, por lo general, definen la estructura de un sistema (PRESSMAN 2010).

Un patrón de diseño describe una estructura de diseño que resuelve un problema de diseño particular dentro de un contexto específico. Permite al diseñador obtener una descripción la cual le facilita determinar si el patrón es aplicable al trabajo actual, si se puede reutilizar y así ahorrar tiempo de diseño, y si el patrón puede servir como guía para desarrollar un patrón similar pero con diferente estructura y funcionalidad (PRESSMAN 2010).

Patrones Generales de Asignación de Responsabilidades

Experto en Información: Permite asignar la responsabilidad a la clase `ServerEstadistica.php` que tiene la información correspondiente para realizar todas las funcionalidades del sistema.

Creador: Permite crear objetos de una instancia para poder acceder a la información de la misma. Se evidencia con la clase `formEstadistica` que hereda de `formEstadistica.ui.js`.

Alta Cohesión: Se emplea para distribuir la información por clases de forma tal que cada una maneje solo la información necesaria.

Bajo Acoplamiento: Este patrón se emplea con el objetivo de crear mínima dependencia entre las clases de forma tal que si se realizan cambios en alguna de ellas no implique grandes cambios en las restantes.

Por otra parte se emplearon dos de los patrones del **Grupo de los cuatro**:

Singleton: Establece la creación de una instancia de una clase como único punto de acceso global a esta, como es el caso de la clase `ServerContext` donde el método `getmapobject` crea una instancia del mapa, la cual es única en todo el sistema.

Command: Este patrón permite encapsular las peticiones a través de un objeto, lo que permite realizar operaciones como gestionar las acciones de dicho objeto. Se utiliza para la comunicación a través de las interfaces de usuario, específicamente mediante la clase `AjaxHelper` que es la encargada de comunicar las interfaces con el servidor.

3.6 Estándar de codificación

Como estándar de codificación se decidió emplear CamelCase para garantizar el entendimiento y mejor lectura del código por todos los desarrolladores de la LPS Aplicativos SIG. Este estilo de escritura es aplicable a palabras o frases unidas en la declaración de variables o en el nombre de las funciones y se puede traducir como Mayúsculas/Minúsculas. Según (BINKLEY *et al.* 2009) existen dos tipos de CamelCase:

- UpperCamelCase: Donde la primera letra de cada palabra se escribe con mayúscula.
- lowerCamelCase: Donde la primera letra de la primera palabra es con minúscula y el resto de las primeras letras quedan en mayúscula.

Se selecciona la variante lowerCamelCase para la implementación de la solución.

3.7 Conclusiones parciales

La modelación del dominio, la definición de los requisitos y los Diagramas de Casos de Uso del Sistema contribuyen al entendimiento del equipo de desarrollo lo que representa una ventaja para la correcta ejecución de las restantes etapas en la construcción de la solución propuesta.

CAPÍTULO 4: DISEÑO, IMPLEMENTACIÓN Y PRUEBAS

4.1 Introducción

En el presente capítulo se presentan los diagramas correspondientes a los modelos de diseño e implementación, así como el diseño de la Base de Datos. Finalmente, se presenta el proceso de pruebas a la solución propuesta.

4.2 Diseño de la Base de Datos

4.2.1 Diagrama de clases persistentes

El diagrama de clases persistentes muestra todas las clases capaces de mantener su valor en el espacio y en el tiempo. En la Figura 7 se muestra el diagrama de clases persistentes de la propuesta de solución.

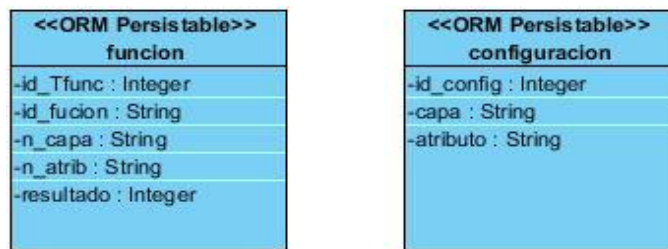


Figura 7: Diagrama de clases persistentes.

4.2.2 Diagrama Entidad – Relación (DER)

Este diagrama se emplea para identificar un conjunto de componentes primarios como objetos de datos, atributos, relaciones e indicadores de varios tipos, su propósito principal es representar objetos de datos y sus relaciones (PRESSMAN 2008). La Figura 8 muestra el DER de la propuesta de solución.

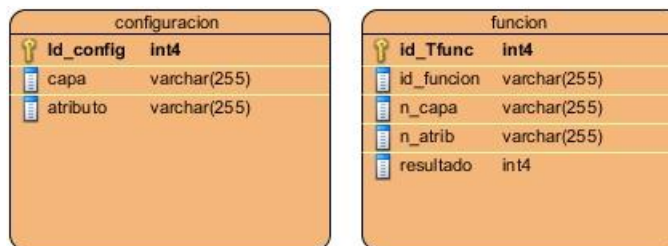


Figura 8: Diagrama Entidad - Relación.

4.3 Modelo de Diseño

Es la representación física de los Casos de Uso mediante un modelo de objetos; centra su atención en el impacto que tiene en el sistema los requisitos tanto funcionales como no funcionales y algunas otras restricciones del entorno de implementación (JACOBSON *et al.* 2000), sirve de abstracción para la implementación y como entrada fundamental de la misma.

4.3.1 Diagrama de Clases del Diseño

Según (JACOBSON *et al.* 2000), un diagrama de clases corresponde a la realización de un Caso de Uso, mostrando las clases, subsistemas y las relaciones involucradas en el mismo, también tiene en cuenta los requisitos que influyen sobre determinada clase, objetos o subsistemas. En la Figura 9 se muestra el Diagrama de Clases del Diseño del Caso de Uso Calcular Parámetros Estadísticos. El resto de los Diagramas de Clases del Diseño pueden ser consultados en el Anexo II.

4.4 Modelo de Implementación

El modelo de implementación describe cómo los elementos del modelo del diseño, se implementan en términos de componentes (archivos de código fuente, ejecutables, entre otros). El modelo de implementación también describe cómo se organizan los componentes de acuerdo con los mecanismos de estructuración y modularización disponibles en el entorno de implementación y en el lenguaje o los lenguajes de programación utilizados, y cómo dependen los componentes unos de otros (JACOBSON *et al.* 2000).

4.4.1 Diagrama de Componentes

El diagrama de componentes muestra el sistema de software dividido en componentes y las relaciones existentes entre estos componentes, forma parte del modelo de implementación. Un diagrama de componentes permite visualizar con más facilidad la estructura general del sistema y el comportamiento del servicio que estos componentes proporcionan y utilizan a través de las interfaces (JACOBSON *et al.* 2000). La Figura 11 muestra el Diagrama de Componentes del Caso de Uso Calcular Parámetros Estadísticos.

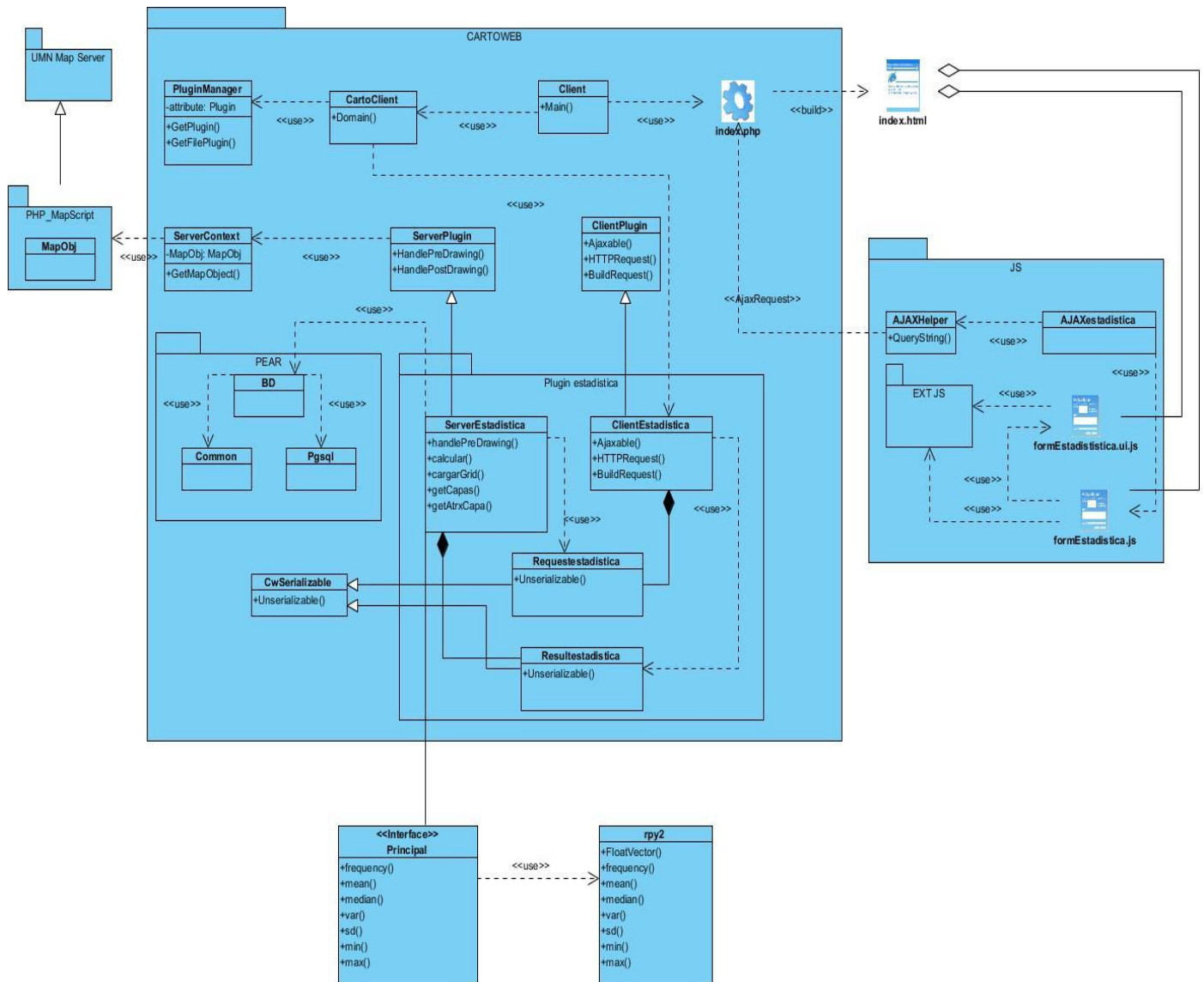


Figura 9: Diagrama de Clases del Diseño del Caso de Uso Calcular Parámetros Estadísticos.

4.5 Modelo de Despliegue

El modelo de despliegue consiste en un modelo de objetos que tiene como objetivo describir la distribución física del sistema en términos de cómo se distribuye la funcionalidad entre las estaciones de trabajo (denominados nodos). Es la entrada fundamental en las actividades de diseño e implementación debido a que la distribución del sistema tiene una influencia principal en su diseño (JACOBSON *et al.* 2000). El modelo de despliegue de la solución propuesta puede consultarse a continuación:

CAPÍTULO 4

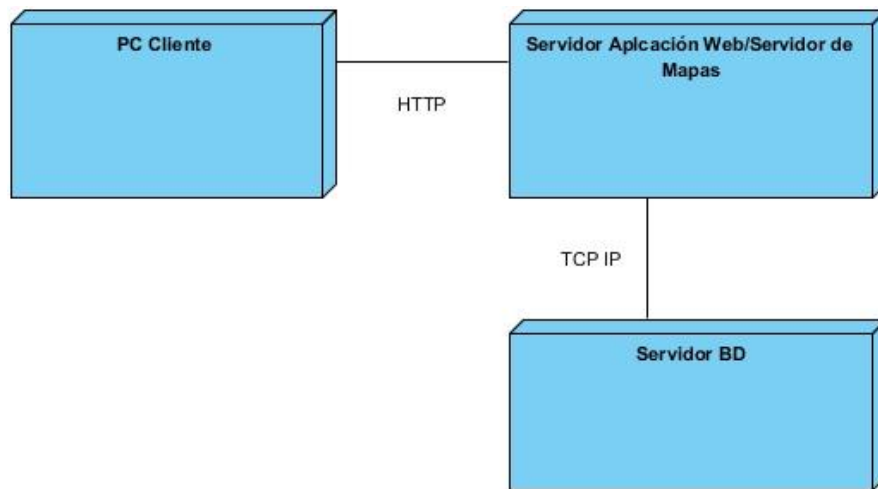


Figura 10: Diagrama de despliegue.

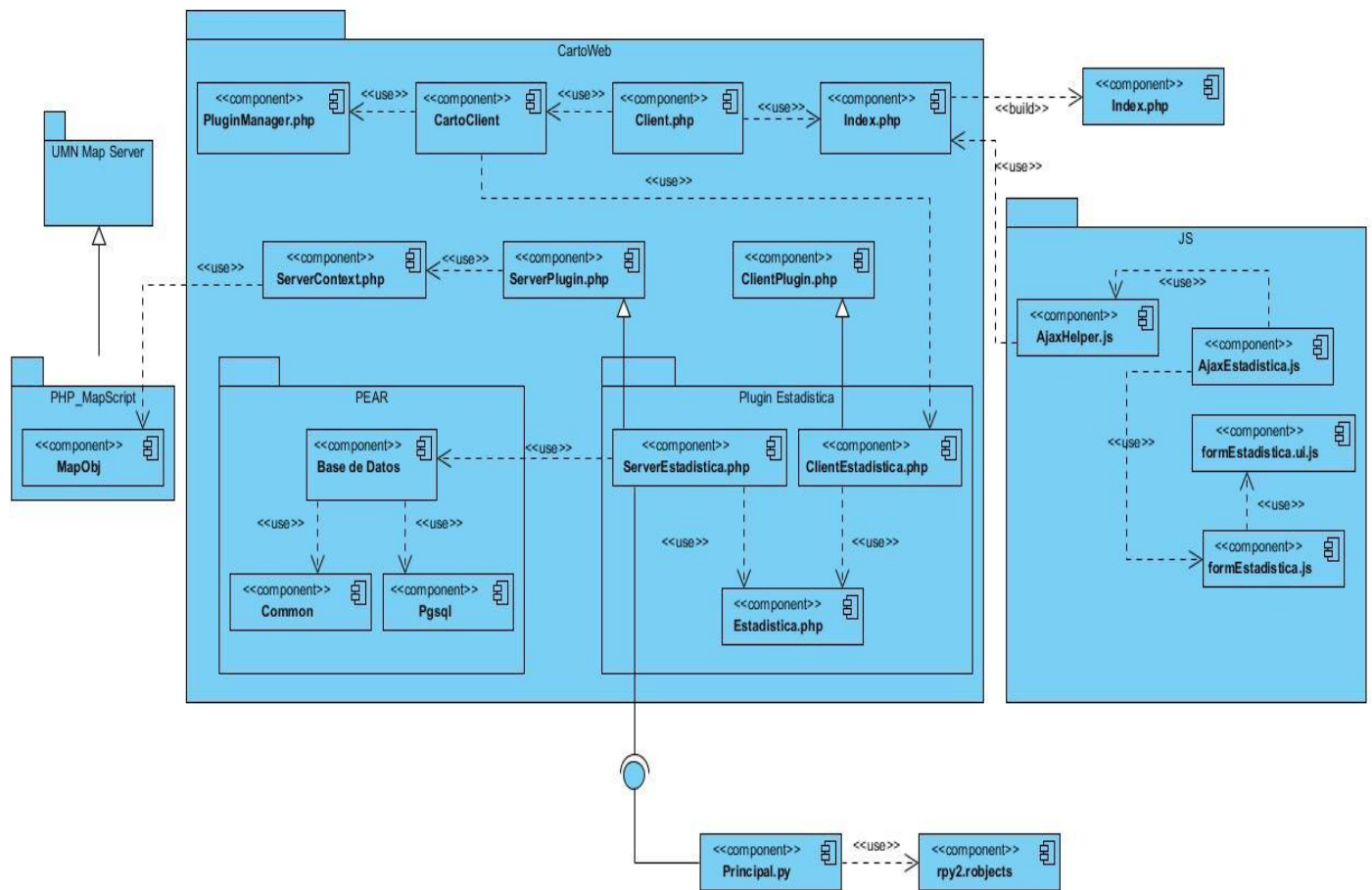


Figura 11: Diagrama de componentes.

4.6 El Proceso de Pruebas

Las pruebas de software son el mecanismo para comprobar que el sistema cumpla con los requerimientos enunciados y además que responda de acuerdo a lo esperado.

Según (PRESSMAN 2010) las pruebas son un conjunto de actividades que se planean con anticipación y se realizan de manera sistemática y para descubrir errores cometidos a la hora de realizar el diseño y construcción de un software. Se ejecutan utilizando un conjunto de pasos en que se pueden incluir técnicas y métodos específicos del diseño de casos de pruebas.

El objetivo de las pruebas es asegurarse de que los subsistemas de todos los niveles y de todas las capas funcionen. Comenzando por las capas más bajas de la arquitectura se prueban los mecanismos de distribución, almacenamiento, recuperación y concurrencia de objetos, así como otros mecanismos de las capas inferiores del sistema lo que significa que no solo se comprueba la funcionalidad sino también si el rendimiento es aceptable (JACOBSON *et al.* 2000).

4.6.1 Diseño de Casos de Prueba

El diseño de las pruebas se basa en la creación de Casos de Prueba cuya ejecución permitirá observar posibles síntomas de defectos. Se puede definir un Caso de Prueba como “el conjunto de entradas, condiciones de ejecución y resultados esperados desarrollados para un objetivo particular como, por ejemplo, ejercitar un camino concreto de un programa o verificar el cumplimiento de un determinado requisito” (LARA and FERNÁNDEZ 2001).

Las pruebas de caja negra son un método de prueba en el que los datos de prueba se derivan de los requisitos funcionales especificados sin tener en cuenta la estructura final del programa. Estas pruebas permiten obtener un conjunto de condiciones de entrada que ejerciten completamente todos los requisitos funcionales de un programa. En ellas se ignora la estructura de control, concentrándose en los requisitos funcionales del sistema y ejercitándolos (PRESSMAN 2008).

La técnica seleccionada para el proceso de pruebas es la partición de equivalencia, que divide el campo de entrada en clases de datos que tienden a ejercitar determinadas funciones del software (JURISTO 2005).

A continuación se presenta el Diseño del Caso de Prueba para el Caso de Uso Calcular Parámetros Estadísticos.

Tabla 3: Caso de Prueba del Caso de Uso Calcular Parámetros Estadísticos.

Nombre de la sección	Escenarios de la sección	Capa	Atributo	Funciones	Respuesta del sistema	Flujo central
SC1: Calcular parámetros estadísticos	EC 1.1: Calcular parámetros estadísticos correctamente	V	V	V	Muestra una tabla con los resultados del cálculo y guarda los mismos en la Base de Datos.	El usuario selecciona la opción Funcionalidades para escoger la capa y el atributo así como la opción del parámetro que desea calcular y posteriormente oprime el botón Calcular, el sistema muestra el resultado del cálculo mediante una tabla.
		empresa	Cant_ind ustriales	Media		
		V	V	V		
		estación_ alevinaje	id_ueba	Moda		
	EC 1.2: Campos obligatorios vacíos	I	V	V	El sistema no habilita las opciones de seleccionar el atributo y el parámetro estadístico y muestra un mensaje de notificación.	El usuario selecciona la opción Funcionalidades y oprime el botón Calcular.
			N/A	N/A		
		V	I	V		
		empresa		N/A		
	EC 1.3: Cerrar ventana	V	V	V	El sistema oculta la ventana Funcionalidades y no guarda los cambios realizados.	El usuario selecciona la opción Funcionalidades y oprime el botón Cerrar.
		N/A	N/A	N/A		

4.6.2 Pruebas de Rendimiento

Según (MOLYNEAUX 2009) son las pruebas que se realizan para determinar cuán rápido realiza una tarea un sistema en condiciones particulares de trabajo. También resultan útiles para verificar atributos como la escalabilidad y fiabilidad del sistema. Existen varios tipos de pruebas de rendimiento, se ejecutarán las pruebas denominadas de carga y estrés como parte del proceso de pruebas a la solución.

Pruebas de carga: Se realiza para observar el comportamiento de una aplicación bajo una cantidad de peticiones esperada. Esta carga puede ser la cantidad de usuarios concurrentes que utiliza la aplicación y las transacciones que realizan. Esta prueba puede mostrar el tiempo de respuesta de cada transacción.

Prueba de estrés: Se emplea normalmente para romper la aplicación. Se va duplicando el número de usuarios que se agregan a la aplicación y se ejecutan pruebas de carga hasta que se rompe. Este tipo de prueba se realiza para determinar la solidez de la aplicación en los momentos de carga extrema y ayuda a los administradores para determinar si la aplicación rendirá lo suficiente en caso de que la carga real supere la carga esperada.

4.6.3 Ejecución de las pruebas

Una vez diseñados todos los Casos de Prueba, a partir de las descripciones textuales de los CUS se desarrollaron tres iteraciones de acuerdo a como se muestra en la Figura 12. En la primera iteración se obtuvieron seis no conformidades, en la segunda iteración se corrigieron las no conformidades encontradas anteriormente y surgieron dos nuevas no conformidades las cuales fueron corregidas en la tercera iteración en la cual no se obtuvieron nuevas no conformidades.

Luego de efectuadas las pruebas de carga y estrés para comprobar el rendimiento del sistema los resultados obtenidos fueron los siguientes:

Tabla 4: Resultados de las Pruebas de rendimiento.

Acción	Cantidad de Usuarios	Rendimiento
Ejecución de GeneSIG	25	830,8 sec
Ejecución de GeneSIG + Módulo Desarrollado	25	1,4 sec
Ejecución de GeneSIG	50	1007,6 sec
Ejecución de GeneSIG + Módulo Desarrollado	50	1318,5 sec

La Tabla anterior muestra los resultados de las pruebas de rendimiento a la plataforma GeneSIG antes de contener el módulo desarrollado y una vez obtenido el módulo y ejecutado el mismo. Los resultados demuestran diferencias poco significativas entre los dos resultados por lo cual es posible concluir que la implementación del módulo de estadística descriptiva no incide negativamente en el rendimiento de la plataforma GeneSIG.

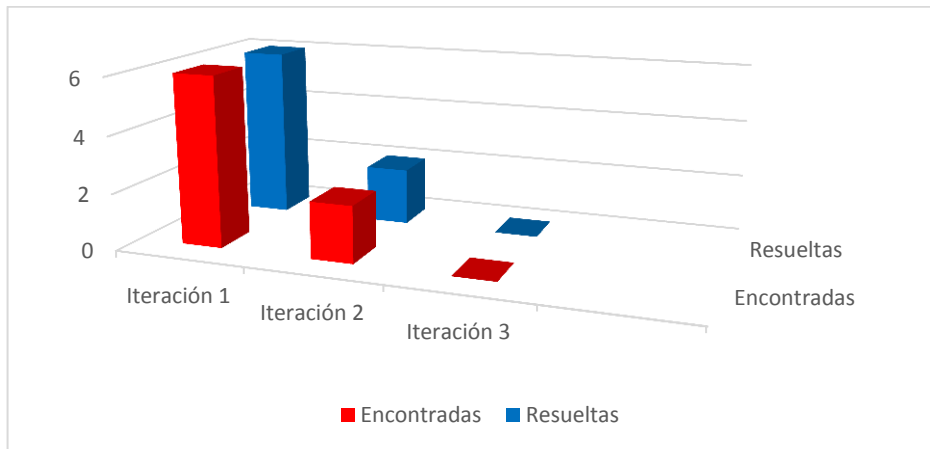


Figura 12: No conformidades detectadas durante el proceso de desarrollo.

4.7 Conclusiones parciales

El proceso de pruebas desarrollado permitió detectar y corregir los errores de la etapa de implementación, lo cual aumenta la calidad del producto final obtenido. Las pruebas de rendimiento arrojaron que no existen diferencias significativas en cuanto al rendimiento al implementar y ejecutar el módulo de estadística descriptiva sobre GeneSIG.

CONCLUSIONES

Luego de finalizada la investigación los autores concluyen:

1. El análisis de las soluciones existentes arrojó que no existen propuestas de implementación que satisfagan los requerimientos del problema de la investigación actual. Aun así, varias de sus funcionalidades fueron contempladas en la propuesta actual.
2. El modelo arquitectónico utilizado, y la utilización de cada patrón de diseño proporciona uniformidad, robustez y flexibilidad a la solución desarrollada, al mismo tiempo que favorecieron la rigurosidad en su implementación.
3. La aplicación de las pruebas de caja negra permitió detectar los errores obtenidos durante el proceso de implementación aumentando la calidad del producto final obtenido.
4. La aplicación de las pruebas de rendimiento demostró que el módulo desarrollado no incide negativamente en el rendimiento de la plataforma GeneSIG pues los valores de tiempo de respuesta del sistema se comportan similares con el módulo en ejecución y sin él.
5. La obtención del módulo de Estadística descriptiva aumentará las potencialidades de la plataforma GeneSIG y sus posibilidades de comercialización e introducción en el mercado nacional e internacional.

RECOMENDACIONES

1. Implementar el resto de los parámetros estadísticos que forman parte de la Estadística descriptiva.
2. Ampliar el campo de acción de la solución que incluya la geoestadística además de la Estadística descriptiva.

REFERENCIAS BIBLIOGRÁFICAS

1. ACHOUR, M.; F. BETZ, *et al.* *PHP Manual* PHP Documentation Group, 2006. [Disponible en: <http://php.net/manual/en/index.php>]
2. ALEGA. *Definición de lenguaje de programación*, 1998 - 2015. [Disponible en: <http://www.alegsa.com.ar/Dic/lenguaje%20de%20programacion.php>]
3. BERTINO, E. and L. MARTINO. *Sistemas de bases de datos orientadas a objetos: conceptos y arquitecturas*. Massachusetts, Madrid, Addison Wesley Iberoamericana S. A. y Ediciones Díaz de Santos S. A., 1993. 287 p.
4. BINKLEY, D.; M. DAVIS, *et al.* *To CamelCase or Under_score*. 17th IEEE International Conference on Program Comprehension Vancouver, Canadá, 2009. 167 p.
5. BOJAEN, J. *Introducción a R: historia de un lenguaje de computación para el análisis de datos*, 2013. [Disponible en: <http://www.genbetadev.com/lenguajes-y-plataformas/introduccion-a-r-historia-de-un-lenguaje-de-computacion-para-el-analisis-de-datos>]
6. BUZAI, G. D. and D. J. ROBINSON. *Sistemas de Información Geográfica en América Latina (1987 - 2010). Un análisis de su evolución académica basado en la CONFIGSIG. XIII Conferencia Iberoamericana de Sistemas de Información Geográfica*. Toluca, México, Departamento de Ciencias Sociales, Universidad Nacional de Luján, 2011. 18.
7. CABRIA, S. G. *Filosofía de la Estadística*. Valencia, Casa del Libro, 1994. 320 p.
8. CASTILLO, L. G. *Análisis Documental Biblioteconomía*, 2005: 18.
9. CELADA, M. S.; T. G. ROFFE, *et al.* *Estado del arte del uso de la tecnología SIG en la gestión del riesgo de desastres naturales en la agricultura a escala nacional*, 2006. 43.
10. CLEMEN, R. T. *Making Hard Decisions: An Introduction to Decision Analysis*. Duxbury Press, 1996. 664 p.
11. ESPINOSA, A. M.; E. P. RAMOS, *et al.* *Desarrollo de Sistemas de Información Geográficos sobre una plataforma soberana*, Universidad de las Ciencias Informáticas, 2012. 241.
12. ESRI. *ArcGIS Help 10.1*, Environmental Systems Research Institute, 2015. [Disponible en: <http://resources.arcgis.com/es/help/main/10.1/index.html#//00qn0000001p000000>]
13. FIGUEROA, R. G.; C. J. SOLÍS, *et al.* *Metodologías tradicionales vs. metodologías ágiles.*, Universidad Técnica Particular de Loja, 2013. 9.
14. FREDERICK, S.; C. RAMSAY, *et al.* *Learning Ext JS*. Birmingham, Mumbai, Packt Publishing, 2008. 324 p.
15. GARRIDO, M. A. *SIG y medio ambiente: principios básicos*. España, Cádiz : Servicio de Publicaciones de la Universidad de Cádiz, 2003. 19 p.
16. GONZÁLEZ, E. *¿Qué es PHP? y ¿Para qué sirve? Un potente lenguaje de programación para crear páginas web.*, 2015. [Disponible en: http://www.aprenderaprogramar.com/index.php?option=com_content&id=492:ique-es-php-y-ipara-que-sirve-un-potente-lenguaje-de-programacion-para-crear-paginas-web-cu00803b&Itemid=193]
17. GONZÁLEZ, S. H. *Historia de la Estadística*, [Artículo]. 2005. [Disponible en: <http://www.uv.mx/cienciahombre/revistae/vol18num2/articulos/historia/>]
18. GROUP, T. P. G. D. *Tutorial de PostgreSQL*, The PostgreSQL Global Development Group 1996 - 2015. [2015]. Disponible en: <http://www.postgresql.org/docs/9.1/static/>
19. GVSIG. *gvSIG - Association*, 2015. [Disponible en: <http://www.gvsig.com/es>]

REFERENCIAS BIBLIOGRÁFICAS

20. JACOBSON, I.; G. BOOCH, *et al.* *El Proceso Unificado de Desarrollo de Software*. Edición en español. Madrid, Addison Wesley, Pearson Education, S. A., 2000. 464 p.
21. JUÁREZ, I. P. A. *Estadística Descriptiva*, 2003. 29.
22. JURISTO, N. Chapter 3 – VERIFICATION AND VALIDATION: CURRENT AND BEST PRACTICE en: *Validation, Verification and Certification of Embedded Systems*. 2005. 14.p.
23. KEENEY, R. L. and H. RAIFFA. *Decisions with Multiple Objectives: Preferences and Value Trade-Offs*. Cambridge University Press, 1993. 569 p.
24. LARA, P. and L. FERNÁNDEZ. *Generación de casos de prueba a partir de especificaciones UML*, 2001. 11.
25. LARMAN, C. *UML y Patrones: Introducción al análisis y programación orientada a objetos*. 2da Edición. Madrid, España, Prentice Hall, Pearson Education, 1999. 624 p.
26. LEVINE, D. M.; M. L. BERENSON, *et al.* *Estadística para administración*. Pearson Educación, 2006. 619 p.
27. MAPSERVER. *Acerca de MapServer*, 2015. [Disponible en: <http://www.mapserver.org/es/about.html#about>]
28. MARTÍNEZ, R. *Sobre PostgreSQL*, 2013.
29. MINOTTA, E. A. H. *Estadística Descriptiva*. 2006. 228 p.
30. MOLYNEAUX, I. *The Art of Application Performance Testing*. O'Reilly Media, 2009. 158 p.
31. MUÑOZ, D. R. *Manual de Estadística*. Juan Carlos Martínez Coll, Casa del Libro, 2004. 92 p.
32. NETBEANS. *NetBeans IDE 8.0.2 Information*, Oracle, 2015a. [2015]. Disponible en: <https://netbeans.org/community/releases/80/>
33. ---. *¿Qué es Netbeans?*, Oracle, 2015b. [2015]. Disponible en: https://netbeans.org/index_es.html
34. NOZAWA, J. M. T. *PHP-Mapscript de Mapserver. Parte 1: Conceptos* 2007. [Disponible en: <http://phpexperto.blogspot.com/2007/11/php-mapscript-de-mapserver-parte-2.html>]
35. OLAYA, V. *Sistemas de información geográfica*
36. 2010. 911 p.
37. PARADIGM, V. *Guía del usuario*, 2015. [Disponible en: <http://www.visual-paradigm.com/>]
38. PÉREZ, A. F. *Generación de consultas para la manipulación de Geontologías desde la Plataforma GeneSIG*. La Habana, Universidad de las Ciencias Informáticas, 2011. 69. p.
39. PGADMIN. *pgAdmin Introduction*, 2015 [Disponible en: <http://www.pgadmin.org/>]
40. POSTGRES SQL pgAdmin Introduction, 2012.
41. PRESSMAN, R. S. *Ingeniería de Software: Un enfoque práctico*. 5ta Edición. Madrid, McGraw Hill, 2008. 765 p.
42. ---. *Software Engineering: A Practitioner's Approach*. 7ma Edición. New York, McGraw-Hill, 2010. 870 p.
43. PROJECT, Q. *QGIS User Guide*, 2014. [Publicación 2.2]. Disponible en: <http://www.qgis.org/es/site/>
44. RAMSEY, P.; S. SANTILLI, *et al.* *Postgis Documentation*, PostGIS Project Steering Committee (PSC), 2015. [Disponible en: <http://postgis.net/documentation/>]
45. REYNA, A. El uso de los sistemas de información geográfica (SIG) en el análisis demográfico de situaciones de desastre. *Notas de Población*, 2005, (81): 129 - 162.
46. REYNOSO, C. and N. KICILLOF. *Estilos y patrones en la estrategia de arquitectura de Microsoft*, Versión 1.0. 2004. [2015]. Disponible en: <http://carlosreynoso.com.ar/archivos/arquitectura/Estilos.PDF>
47. ROMERO-CAMPERO, F. J. *Introducción al lenguaje de programación R*, Universidad de Sevilla, 2013. [2015]. Disponible en: http://www.cs.us.es/~fran/curso_unia/introduccion_R.html
48. ROSS, S. M. *Introducción a la Estadística*. Barcelona, España, Editorial Reverté S. A., 2007. 817 p.

REFERENCIAS BIBLIOGRÁFICAS

49. ROSSUM, G. V. *Tutorial de Python*, Python Software Education, 2009. [Disponible en: <http://python.org.ar/pyar/Tutorial>]
50. SIGCEA. *Sistemas de información geográfica, tipos y aplicaciones empresariales.*, 2010. [Disponible en: <https://sig.cea.es>]
51. SIMON, H. A. *La nueva ciencia de la decisión gerencial*. Buenos Aires, El Ateneo, 1982. 163 p.
52. SIRONVALLE, M. A. A. *Estadística*. Santiago de Chile, 2000. 114 p.
53. SOMMERVILLE, I. *Ingeniería del Software*. Séptima Madrid, Pearson Education S.A., 2005. 712 p.
54. STANECKA, N. and C. DÍAZ. *Uso de SIG para la toma de decisiones*. 10mas Jornadas Argentinas de Software Libre, Córdoba, Argentina, Instituto de Estadísticas y Demografía y Centro de Cómputos, Facultad de Ciencias Económicas, 2013. 10 p.
55. STAR, J. and J. E. ESTES. *Geographic Information Systems: An Introduction*. Prentice hall, 1990. 303 p.
56. TOMLIN, C. D. *Geographic Information Systems and cartographic modeling*. Prentice Hall, 1990. 249 p.
57. VÉLIZ, Y. Z. *Modelos de evaluación de la importancia del impacto ambiental en contextos complejos bajo incertidumbre.*: Departamento de Ciencias de la Computación e Inteligencia Artificial. Granada, Universidad de Granada, 2014. 240. p.
58. VILLEGAS, M. Á. G. *Inferencia Estadística*. Madrid, Ediciones Díaz de Santos, 2005. 536 p.
59. ZAYAS, C. Á. D. *Metodología de la Investigación Científica*. Santiago de Cuba, Centro de estudios de Educación Superior Manuel f. Gran, 1995. 80 p.

ANEXO I: DESCRIPCIÓN TEXTUAL DE LOS CASOS DE USO

Tabla 5: Descripción textual del Caso de Uso Representar gráficamente parámetros estadísticos.

Objetivo	Representar mediante una determinada gráfica los resultados del cálculo de los parámetros estadísticos.	
Actores	Usuario.	
Resumen	Inicia cuando el usuario selecciona la opción graficar en la tabla que muestra los resultados del cálculo realizado y termina cuando el sistema muestra la gráfica.	
Complejidad	Alta	
Prioridad	Crítica	
Precondiciones	Haber ejecutado el CU Calcular parámetros estadísticos.	
Poscondiciones	No procede.	
Flujo de eventos		
Flujo básico		
Actor	Sistema	
1. Selecciona la opción graficar.	2. El sistema muestra una ventana con los diferentes tipos de gráficas que se pueden representar.	
3. El usuario selecciona el tipo de gráfica que quiere representar.	4. El sistema muestra que la opción ha sido seleccionada.	
5. El usuario selecciona el botón aceptar.	6. El sistema muestra la gráfica con los datos, previamente calculados, representados.	
7. El usuario selecciona la opción cerrar ventana de la gráfica representada.	8. El sistema oculta la ventana de la gráfica representada.	
	9. El sistema muestra la interfaz con los resultados de los parámetros estadísticos calculados.	
Flujos alternos		

ANEXOS

1. Cerrar ventana		
Actor	Sistema	
2. El usuario selecciona la opción cerrar ventana.	3. El sistema oculta la ventana que contiene los tipos de gráficas.	
	4. El sistema muestra la interfaz con los resultados de los parámetros estadísticos calculados.	
2. Cancelar ventana		
Actor	Sistema	
2. El usuario selecciona el botón Cancelar.	3. El sistema oculta la ventana que contiene los tipos de gráficas y no guarda los cambios y regresa a la interfaz que muestra los resultados de los parámetros estadísticos calculados.	
	4. El sistema muestra la interfaz con los resultados de los parámetros estadísticos calculados.	
Relaciones	CU incluidos	No procede
	CU extendidos	No procede
<i>Prototipo de Interfaz</i>		
<p>The image displays four screenshots of a graphical user interface for data visualization. The first window, titled 'Gráficas', is a selection dialog with radio buttons for 'Barra' (selected), 'Pastel', and 'Lineal'. The second window, 'Gráfica de Pastel', shows a pie chart with a data label for 'cant_pescadores' at 10 (40%). The third window, 'Gráfica Lineal', shows a line graph with 'Valores' on the y-axis (0-50) and 'Atributos' on the x-axis (cant_pescadores, cant_otros, cant_industriales). The fourth window, 'Gráfica de Barra', shows a bar chart with the same axes and data points as the line graph.</p>		

ANEXOS

Tabla 6: Descripción textual del Caso de Uso Representar en el mapa parámetros estadísticos.

Objetivo	Representar mediante puntos específicos en el mapa los resultados del cálculo de los parámetros estadísticos.	
Actores	Usuario.	
Resumen	Inicia cuando el usuario selecciona la opción mapa en la tabla que muestra los resultados del cálculo realizado y termina cuando el sistema muestra la tematización.	
Complejidad	Alta	
Prioridad	Crítica	
Precondiciones	Haber ejecutado el CU Calcular parámetros estadísticos.	
Poscondiciones	El sistema muestra un árbol con la tematización guardada.	
Flujo de eventos		
Flujo básico		
Actor	Sistema	
1. Selecciona la opción mapa.	2. El sistema muestra una ventana con los diferentes colores para las cotas superior, inferior e igual respectivamente.	
3. El usuario selecciona los colores para tematizar.	4. El sistema muestra los colores seleccionados.	
5. El usuario selecciona el botón tematizar.	6. El sistema muestra en el mapa la tematización realizada.	
	7. El sistema muestra la interfaz principal de la plataforma GeneSIG.	
Flujos alternos		
1. Cerrar ventana		
Actor	Sistema	
2. El usuario selecciona la opción cerrar ventana.	3. El sistema oculta la ventana que contiene los	

ANEXOS

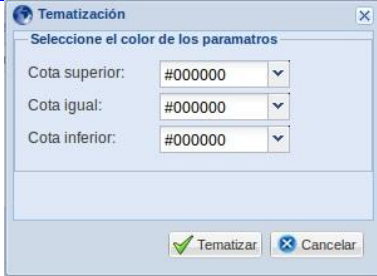
		colores para la tematización.
		4. El sistema muestra la interfaz con los resultados de los parámetros estadísticos calculados.
2. Cancelar ventana		
Actor		Sistema
2. El usuario selecciona el botón Cancelar.		3. El sistema oculta la ventana que contiene los colores para la tematización y regresa a la interfaz que muestra los resultados de los parámetros estadísticos calculados.
		4. El sistema muestra la interfaz con los resultados de los parámetros estadísticos calculados.
Relaciones	CU incluidos	No procede
	CU extendidos	No procede
Prototipo de Interfaz		
		

Tabla 7: Descripción textual del caso de Uso Gestionar capa.

Objetivo	Seleccionar la capa que se desea configurar.
Actores	Usuario.
Resumen	Inicia cuando el usuario selecciona la opción Configuración
Complejidad	Alta
Prioridad	Crítica

ANEXOS

Precondiciones	No procede.	
Poscondiciones	El sistema guarda en la Base de Datos la configuración realizada.	
Flujo de eventos		
Flujo básico		
Actor	Sistema	
1. Selecciona la opción Configuración.	2. El sistema muestra una ventana que contiene las opciones para seleccionar la capa y el atributo, además de las configuraciones previamente realizadas.	
3. El usuario selecciona la capa.	4. El sistema muestra que la capa ha sido seleccionada.	
	5. El sistema muestra la interfaz principal de Configuración.	
Flujos alternos		
1. Cerrar ventana		
Actor	Sistema	
2. El usuario selecciona la opción cerrar ventana.	3. El sistema oculta la ventana que contiene las configuraciones.	
	4. El sistema muestra la interfaz principal de la plataforma GeneSIG.	
2. Cancelar ventana		
Actor	Sistema	
2. El usuario selecciona el botón Cancelar.	3. El sistema oculta la ventana que contiene las configuraciones.	
	4. El sistema muestra la interfaz principal de la plataforma GeneSIG.	
Sección 1: “Guardar Configuración”		
Flujo básico		

ANEXOS

Actor	Sistema
1. El usuario selecciona el botón Guardar.	2. El sistema guarda los datos de la configuración en la Base de Datos, los muestra en la ventana Configuración y muestra un mensaje de notificación.
	3. El sistema muestra la interfaz Configuración.
Flujos alternos	
1. Cerrar ventana	
Actor	Sistema
1. El usuario selecciona la opción cerrar ventana.	2. El sistema cierra la ventana Configuración.
	3. El sistema muestra la interfaz principal de la plataforma GeneSIG.
2. Cancelar ventana	
Actor	Sistema
1. El usuario selecciona el botón Cancelar.	2. El sistema cierra la ventana Configuración y no guarda los cambios.
	3. El sistema muestra la interfaz principal de la plataforma GeneSIG.
3. Campos vacíos	
Actor	Sistema
1. El usuario selecciona la opción guardar.	2. El sistema un mensaje de notificación.
	3. El sistema muestra la interfaz Configuración.
4. Datos previamente guardados	
Actor	Sistema
1. El usuario selecciona la opción guardar.	2. El sistema un mensaje de notificación.

ANEXOS

	3. El sistema muestra la interfaz Configuración.
Sección 1: “Eliminar Configuración”	
Flujos básico	
Actor	Sistema
1. El usuario selecciona el botón Eliminar.	2. El sistema elimina los datos de la configuración en la Base de Datos y muestra un mensaje de notificación.
	3. El sistema muestra la interfaz Configuración.
Flujos alternos	
1. Cerrar ventana	
Actor	Sistema
1. El usuario selecciona la opción cerrar ventana.	2. El sistema cierra la ventana Configuración.
	3. El sistema muestra la interfaz principal de la plataforma GeneSIG.
2. Cancelar ventana	
Actor	Sistema
1. El usuario selecciona el botón Cancelar.	2. El sistema cierra la ventana Configuración.
	3. El sistema muestra la interfaz principal de la plataforma GeneSIG.
3. Campos vacíos	
Actor	Sistema
1. El usuario selecciona la opción eliminar.	2. El sistema un mensaje de notificación.
	3. El sistema muestra la interfaz Configuración.
4. Datos inexistentes	

ANEXOS


Actor		Sistema
1. El usuario selecciona la opción eliminar.		2. El sistema un mensaje de notificación.
		3. El sistema muestra la interfaz Configuración.
Relaciones	CU incluidos	Gestionar atributo
	CU extendidos	No procede
Prototipo de Interfaz		
		

Tabla 8: Descripción textual del Caso de Uso Gestionar atributo.

Objetivo	Seleccionar los atributos correspondientes a una capa determinada que se desean configurar.
Actores	Usuario.
Resumen	Inicia cuando el usuario selecciona la capa deseada.
Complejidad	Alta
Prioridad	Crítica
Precondiciones	Haber seleccionado una capa o que la misma haya sido previamente guardada.
Poscondiciones	El sistema guarda en la Base de Datos la configuración realizada.


ANEXOS

Flujo de eventos	
Flujo básico	
Actor	Sistema
1. El usuario selecciona el atributo.	2. Muestra que la capa ha sido seleccionada.
	3. El sistema muestra la interfaz Configuración.
Flujos alternos	
1. Cerrar ventana	
Actor	Sistema
1. El usuario selecciona la opción cerrar ventana.	2. El sistema oculta la ventana que contiene las configuraciones.
	3. El sistema muestra la interfaz principal de la plataforma GeneSIG.
2. Cancelar ventana	
Actor	Sistema
1. El usuario selecciona el botón Cancelar.	2. El sistema oculta la ventana que contiene las configuraciones.
	3. El sistema muestra la interfaz principal de la plataforma GeneSIG.
Sección 1: “Guardar Configuración”	
Flujos básico	
Actor	Sistema
1. El usuario selecciona el botón Guardar.	2. El sistema guarda los datos de la configuración en la Base de Datos, los muestra en la ventana Configuración y muestra un mensaje de notificación.
	3. El sistema muestra la interfaz Configuración.
Flujos alternos	

ANEXOS

1. Cerrar ventana	
Actor	Sistema
1. El usuario selecciona la opción cerrar ventana.	2. El sistema cierra la ventana Configuración.
	3. El sistema muestra la interfaz principal de la plataforma GeneSIG.
2. Cancelar ventana	
Actor	Sistema
1. El usuario selecciona el botón Cancelar.	2. El sistema cierra la ventana Configuración y no guarda los cambios.
	3. El sistema muestra la interfaz principal de la plataforma GeneSIG.
3. Campos vacíos	
Actor	Sistema
1. El usuario selecciona la opción guardar.	2. El sistema un mensaje de notificación.
	3. El sistema muestra la interfaz Configuración.
4. Datos previamente guardados	
Actor	Sistema
1. El usuario selecciona la opción guardar.	2. El sistema un mensaje de notificación.
	3. El sistema muestra la interfaz Configuración.
Sección 1: “Eliminar Configuración”	
Flujo básico	
Actor	Sistema
1. El usuario selecciona el botón Eliminar.	2. El sistema elimina los datos de la configuración en la Base de Datos y muestra un mensaje de notificación.
	3. El sistema muestra la interfaz Configuración.
Flujos alternos	
1. Cerrar ventana	

ANEXOS

Actor		Sistema
1. El usuario selecciona la opción cerrar ventana.		2. El sistema cierra la ventana Configuración.
		3. El sistema muestra la interfaz principal de la plataforma GeneSIG.
2. Cancelar ventana		
Actor		Sistema
1. El usuario selecciona la opción cerrar ventana.		2. El sistema cierra la ventana Configuración.
		3. El sistema muestra la interfaz principal de la plataforma GeneSIG.
3. Campos vacíos		
Actor		Sistema
1. El usuario selecciona la opción eliminar.		2. El sistema un mensaje de notificación.
		3. El sistema muestra la interfaz Configuración.
4. Datos inexistentes		
Actor		Sistema
1. El usuario selecciona la opción eliminar.		2. El sistema un mensaje de notificación.
		3. El sistema muestra la interfaz Configuración.
Relaciones	CU incluidos	Gestionar atributo
	CU extendidos	No procede
Prototipo de Interfaz		
		

ANEXO II: DIAGRAMAS DE CLASES DEL DISEÑO

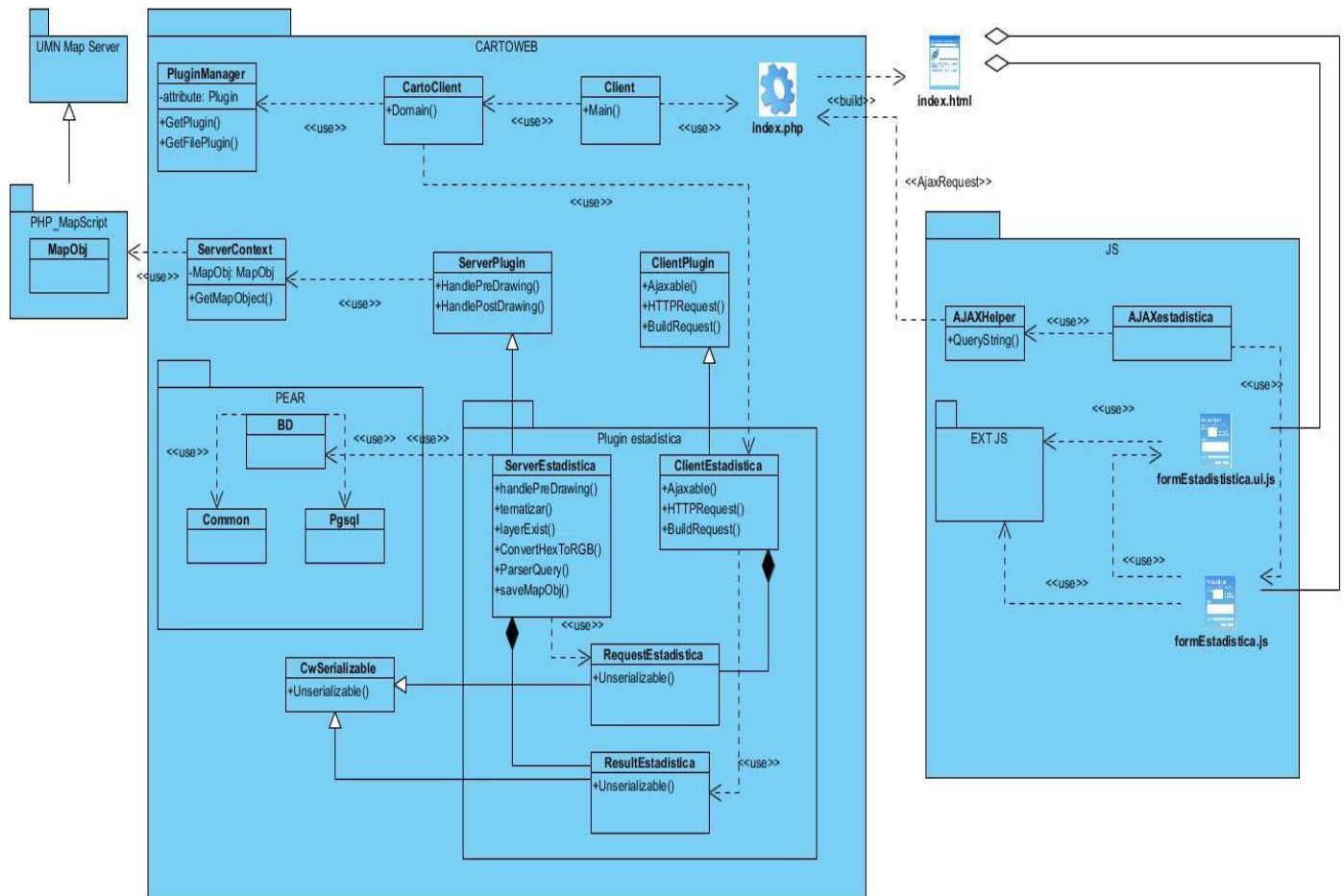


Figura 13 Diagrama de Clases del diseño del Caso de Uso Representar en el mapa los Parámetros Estadísticos.

ANEXOS

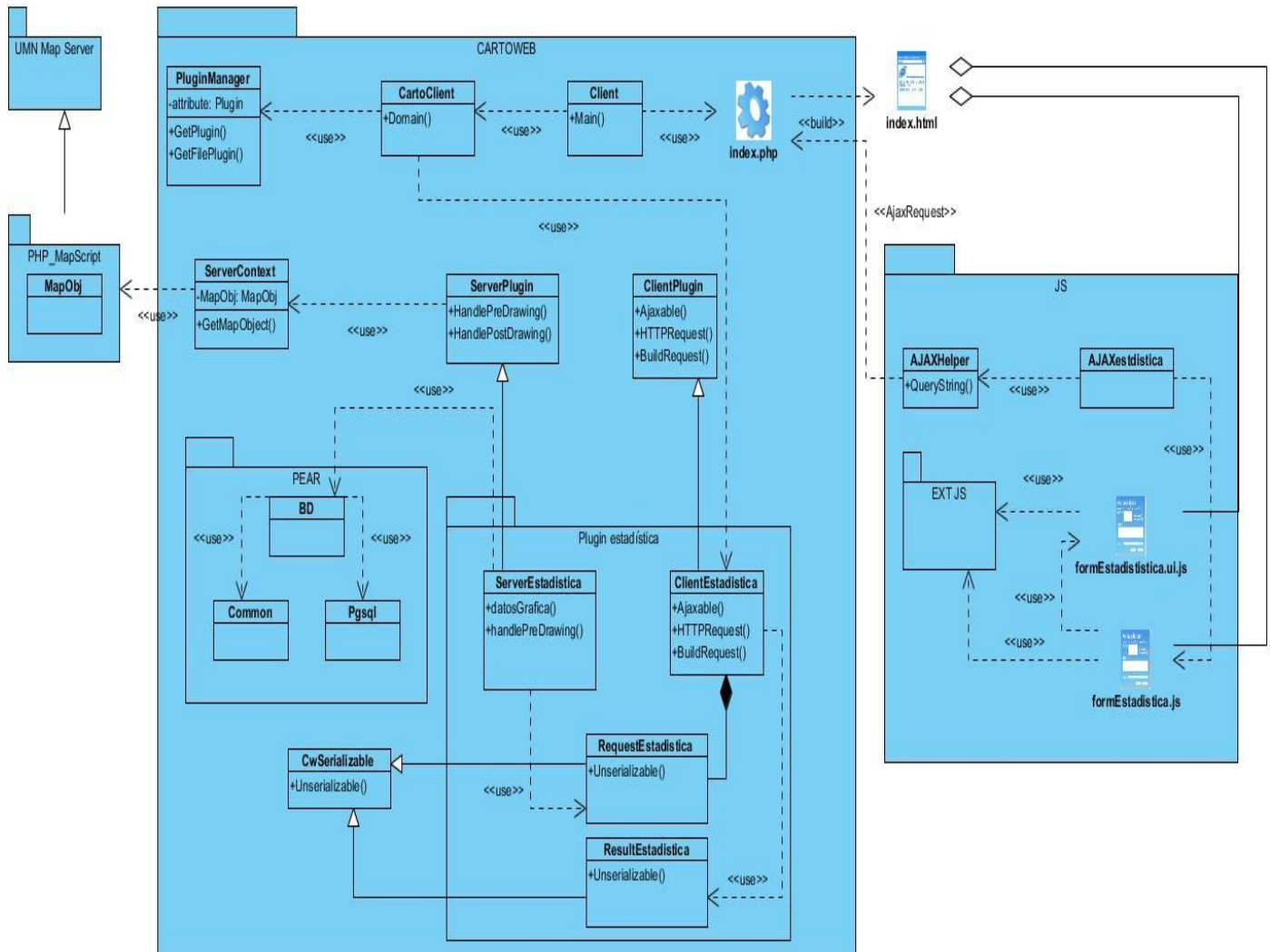


Figura 14: Diagrama de Clases del Diseño del Caso de uso Graficar Parámetros Estadísticos.

ANEXOS

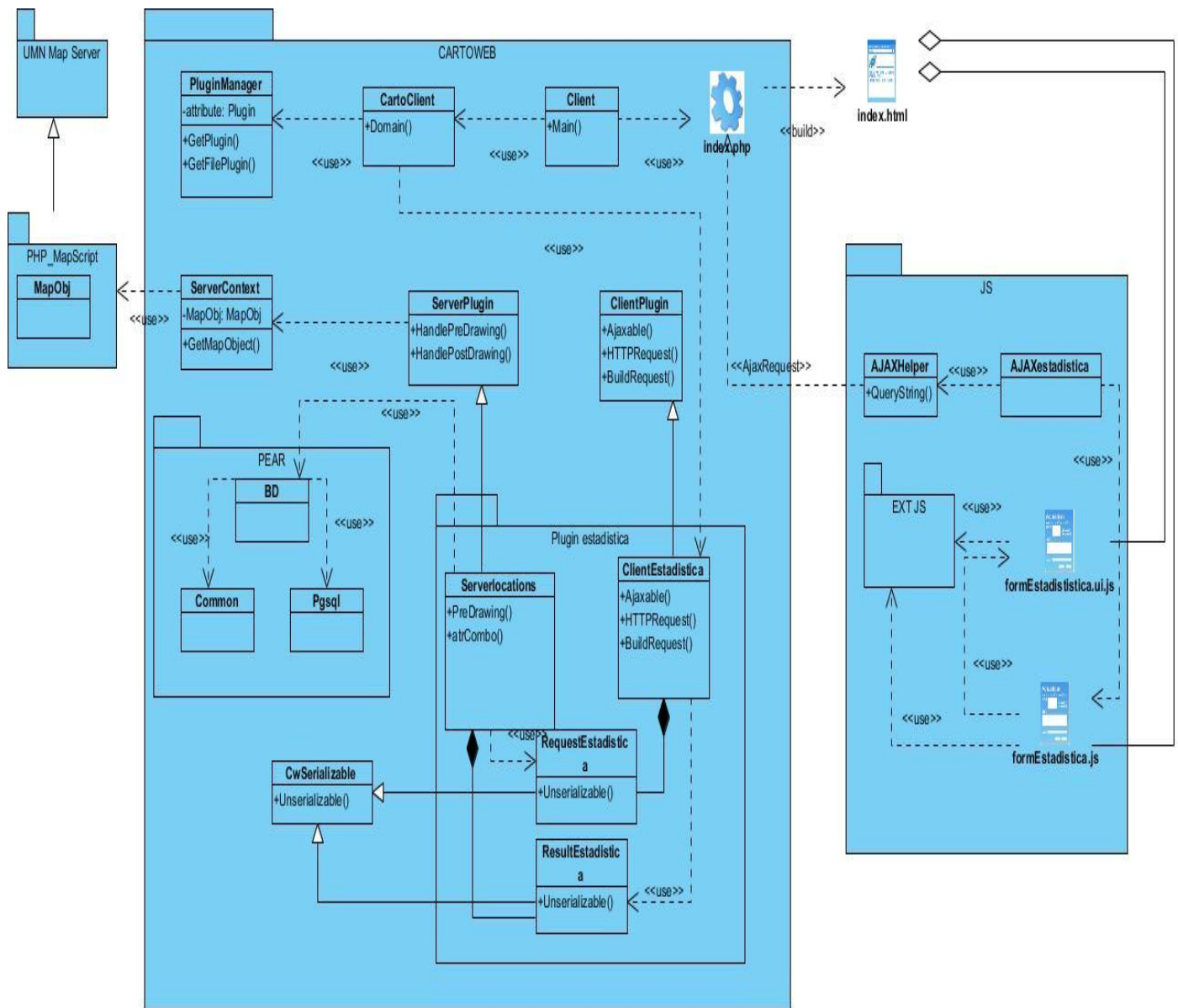


Figura 15: Diagrama de Clases del Diseño del Caso de Uso Gestionar atributo.

ANEXOS

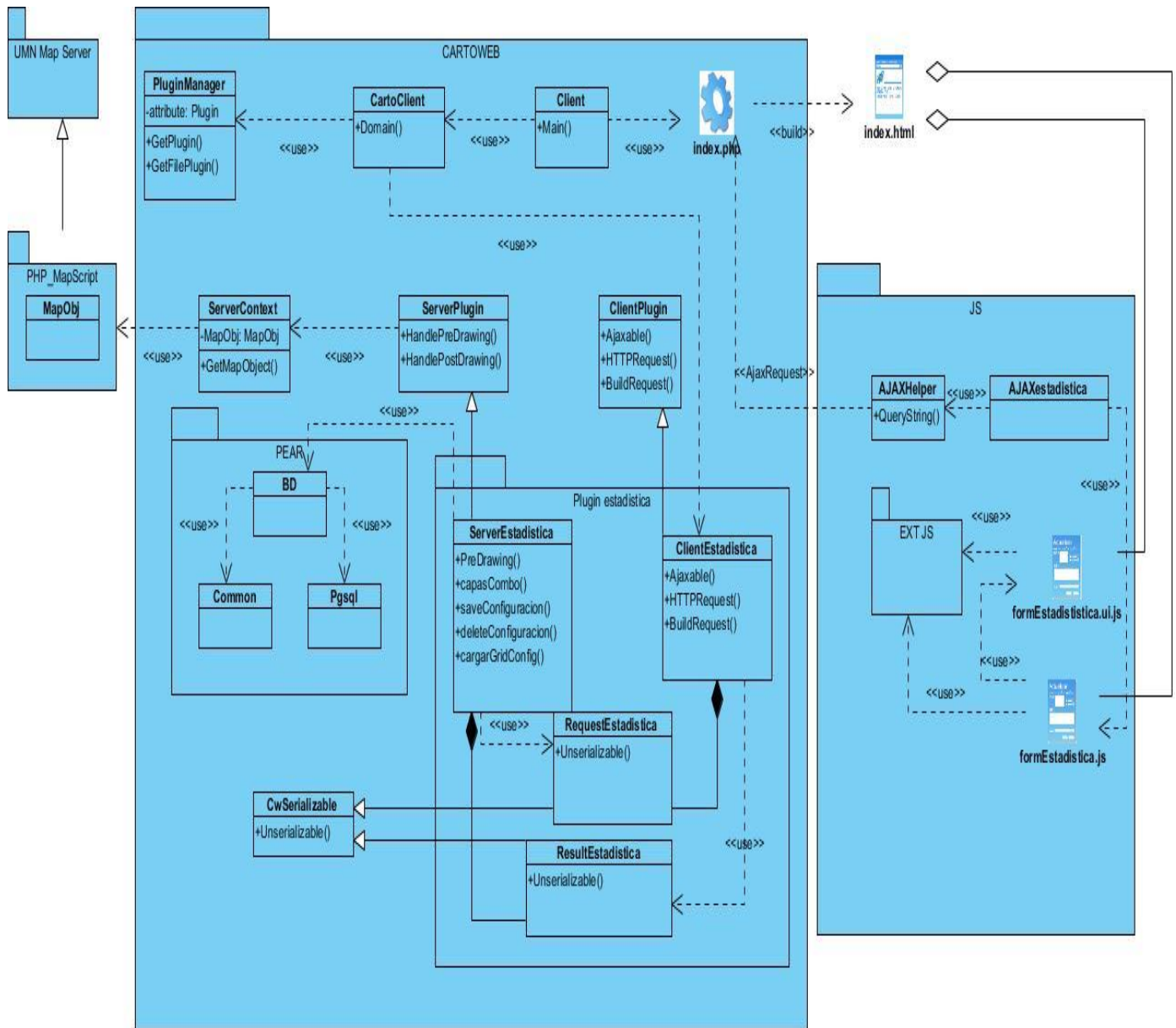


Figura 16: Diagrama de Clases del Diseño del Caso de uso Gestionar capa.