

Universidad de las Ciencias Informáticas

Facultad 6



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

**COMPONENTE DE ESTADÍSTICA INFERENCIAL PARA LA PLATAFORMA
GENESIG**

Autor:

Eddy Rojas García

Tutores:

Ing. Arlet Torres Carballosa

Ing. Laritza Asán Caballero

La Habana, julio de 2016

“Año 58 de la Revolución”



He fallado más de 9000 tiros en mi carrera. He perdido casi 300 juegos. 26 veces han confiado en mí para tomar el tiro que ganaba el juego y lo he fallado.

He fracasado una y otra vez en mi vida y eso es por lo que tengo éxito.

Michael Jordan

DEDICATORIA

Dedico esta tesis a:

Mis padres, quienes han sido ejemplo de esfuerzo y sacrificio, todo logro alcanzado por mí se los debo a ellos.

Mi hermano, quien al ser mi único hermano de sangre y el mayor, siempre ha sido un paradigma a seguir para mí.

AGRADECIMIENTOS

Agradezco a:

Mis padres, por haberme inculcado desde pequeño sus principios y valores, por todo el amor que nos han brindado a mi hermano y a mí, de verdad no exagero, cuando digo que son los mejores padres que un hijo pudiera tener.

Mi hermano, por brindarme siempre su apoyo y ser mi guía, tanto en el estudio como en la vida.

Mis abuelas, a las que adoro y siempre han vigilado de cerca mis pasos, al igual que el resto de mi familia.

Mis hermanitas Alina y Laritza, y mi mejor amigo Jorge Félix, quienes siempre están disponibles cuando los necesito, al igual que yo para ellos.

Mis amigos del barrio: Daniel, Leodán, José Manuel, Juan Carlos, Yosvani, Yasiel, Arnaldo; que siempre me están preguntando como me va en la universidad, y puedo contar con ellos para cualquier cosa.

Mi clan en la UCI: Manuel, Darian, Alfredo, Yobanys, Yoni, Joice, Orlando y muchos más, la lista es demasiado larga, a quienes conocí hace 5 años y se han convertido en prácticamente hermanos para mí.

Las pocas mujeres que nos han acompañado a esta inmensa cantidad de varones, durante estos 5 años: Kielam, María, Ivonne, Dayana, Teresita, Grendys, Yeline, Ilianet; a ustedes muchas gracias.

Las personas de mi apartamento y a todas las personas de los grupos por los que he pasado, muchas gracias a los que no haya mencionado anteriormente.

Mis tutoras, Laritza y Arlet, quienes me han apoyado durante todo el proceso de tesis.

Todos los que han aportado su grano de arena para el desarrollo de esta tesis: Rosaida, Denis Luis, Mario Pupo, Ariel, Rafael, muchas gracias.

Yanary, que aunque no estaba desarrollando una tesis para su proyecto, me dió un lugar en su laboratorio.

Todos los profesores que me han formado a lo largo de estos 5 años.

Los miembros del tribunal y al oponente por la atención prestada y los consejos brindados, muchas gracias.

De forma general, a todos mis amigos, familiares y demás personas que se han visto involucradas en el desarrollo de esta tesis.

DECLARACIÓN DE AUTORÍA

Declaro por este medio que yo, Eddy Rojas García, con número de carné de identidad 92100924506, soy el autor de este trabajo y que autorizo 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 ___ días del mes de julio del año 2016.

Eddy Rojas García
Autor

Ing. Laritza Asán Caballero
Tutor

Ing. Arlet Torres Carballosa
Tutor

Resumen

La Línea de Producción de Software Aplicativos SIG forma parte de la estructura productiva del centro Geoinformática y Señales Digitales perteneciente a la Universidad de las Ciencias Informáticas, dicha línea tiene como objetivo el desarrollo de aplicaciones informáticas que contribuyan al seguimiento y control de información socioeconómica de cualquier sector de la población mediante su representación y análisis espacial, este tipo de aplicaciones son conocidas como Sistemas de Información Geográfica. A pesar de contar con más de 5 años de experiencia en el desarrollo estos sistemas, no poseen el apoyo a la toma de decisiones esperado desde el punto de vista estadístico, debido a que cuentan sólo con un Componente de Estadística Descriptiva para el tratamiento de los datos socioeconómicos. De tal forma, se establece la necesidad de fortalecer el apoyo a la toma de decisiones en los Sistemas de Información Geográfica, haciendo uso de la plataforma GeneSIG. Por tanto, se desarrolló un componente de Estadística Inferencial que fortalece el apoyo a la toma de decisiones en los Sistemas de Información Geográfica, empleando tecnologías libres con las que fue desarrollada GeneSIG, dentro de las que se destacan Netbeans 8.0, PostgreSQL 9.3, Mapserver 6.4, Visual Paradigm 8.0, y lenguajes como PHP 5.0 y ExtJS 3.0. Esta solución cuenta con diversos métodos inferenciales referentes a la estimación y verificación de hipótesis.

Palabras claves: estadística inferencial, estimación, información, métodos, toma de decisiones, verificación de hipótesis.

Abstract

Production Line Software Aplicativos SIG is part of the production structure of Geoinformatic and Digital Signals center belonging to the University of Informatics Sciences Digital Signal, this line aims at the development of applications that contribute to the monitoring and control of socioeconomic information of any section of the population by rendering and spatial analysis. Such applications are known as Geographic Information Systems, despite having more than 5 years of experience in developing these systems do not have support for decision-making that should have from a statistical point of view, because they have only a Component Descriptive Statistics for the treatment of socio-economic data. Therefore, is established the need to strengthen support for decision making in Geographic Information Systems, using the GeneSIG platform. Therefore, is developed a component of inferential statistics that strengthens support decision-making in Geographic Information Systems using free technologies that was developed GeneSIG, among which stand out Netbeans 8.0, PostgreSQL 9.3, MapServer 6.4 Visual Paradigm 8.0, and languages like PHP 5.0 and ExtJS 3.0. This solution has various inferential methods concerning estimation and hypothesis testing.

Key words: decision making, estimation, hypothesis testing, inferential statistics, information, methods.

Índice de Contenido

Introducción	1
Capítulo 1: Fundamentación Teórica	4
1.1 Introducción.....	4
1.2 Estadística Inferencial.....	4
1.2.1 Aspectos importantes que comprende la estadística inferencial.....	4
1.3 Sistemas de Información Geográfica.....	6
1.4 Descripción general de GeneSIG	7
1.4.1 Tecnologías y herramientas	8
1.5 Soluciones existentes	9
1.5.1 GvSIG.....	9
1.5.2 ArcGIS.....	10
1.5.3 QuantumGIS.....	11
1.6 Conclusiones parciales.....	12
Capítulo 2: Herramientas y tecnologías	13
2.1 Introducción.....	13
2.2 Proceso de desarrollo de software	13
2.2.1 Prodesoft	13
2.3 Lenguaje de Modelado UML	14
2.4 Visual Paradigm for UML	15
2.5 Lenguajes de programación	15
2.5.1 PHP.....	16
2.5.2 PHP MapScript.....	16
2.5.3 ExtJS 3.0.....	17
2.5.4 Python.....	17
2.5.5 Lenguaje R.....	17

2.6 IDE NetBeans 8.0	18
2.7 Sistema Gestor de Base de Datos	18
2.7.1 PostgreSQL como SGBD.....	19
2.7.2 PostGIS: la extensión geográfica del gestor seleccionado.....	19
2.8 PgAdmin III para el manejo de PostgreSQL.....	19
2.9 Servidor de mapas MapServer.....	20
2.10 Conclusiones Parciales.....	20
Capítulo 3: Análisis y arquitectura del sistema.	21
3.1 Introducción.....	21
3.2 Modelo de dominio	21
3.2.1 Definición de clases del diagrama del dominio	22
3.3 Requisitos del Software	22
3.3.1 Requisitos funcionales	22
3.3.2 Requisitos no funcionales.....	25
3.4 Arquitectura de software.....	27
3.4.1 Estilo arquitectónico.....	28
3.5 Estándar de codificación	28
3.6 Conclusiones parciales.....	29
Capítulo 4: Diseño, Implementación y pruebas.	30
4.1 Introducción.....	30
4.2 Diseño de la Base de Datos	30
4.2.1 Diagrama de Clases Persistentes	30
4.2.2 Diagrama Entidad - Relación.....	31
4.3 Modelo de Diseño.....	31
4.3.1 Patrones de diseño	31
4.3.2 Diagrama de Clases del Diseño	32
4.4 Modelo de Implementación.....	33

4.4.1 Diagrama de Componentes	33
4.5 Modelo de Despliegue	34
4.6 Proceso de Pruebas	35
4.6.1 Diseño de Casos de Prueba (DCP)	36
4.6.2 Pruebas de Rendimiento	38
4.6.3 Ejecución de las Pruebas	39
4.7 Conclusiones Parciales	40
Conclusiones	41
Recomendaciones	42
Bibliografía	43
Anexo I: Especificaciones de RF	45
Anexo II: Diagrama de Clases del Diseño	53
Anexo III: Diseño de casos de prueba	59

Índice de Figuras

Fig.1: Propiedades de raster de una capa en gvSIG	9
Fig.2: Ventana de estadísticas de una tabla	10
Fig.3: Agrupar características de un campo mediante opción Summarize	10
Fig.4: Tabla de estadísticas previamente seleccionadas en Summarize	11
Fig.5: Fases del ciclo de vida del proyecto en Prodesoft.	14
Fig.6: Modelo de Dominio de la solución.	21
Fig.7: Función getValueAttr de la clase ServerEstadistica.php	29
Fig.8: Diagrama de Clases Persistentes.....	30
Fig.9: Diagrama Entidad - Relación	31
Fig.10: DCD del RF Realizar verificación de hipótesis de varianza.....	33
Fig.11: Diagrama de Componentes del RF Realizar verificación de hipótesis de varianza.....	34
Fig.12: Diagrama de Despliegue	35
Fig.13: Resultados de aplicar las pruebas de caja negra.....	39
Fig.14: DCD del RF Realizar estimación puntual de media.....	53
Fig.15: DCD del RF Realizar estimación bayesiana.....	54
Fig.16: DCD del RF Realizar estimación por intervalos de media con varianza desconocida	55
Fig.17: DCD del RF Realizar verificación de hipótesis de media con varianza conocida.....	56
Fig.18: DCD del RF Realizar verificación de hipótesis de media con varianza desconocida.....	57
Fig.19: DCD del RF Realizar verificación de hipótesis de varianza.....	58

Índice de Tablas

Tabla 1: Resumen de soluciones existentes.....	11
Tabla 2: Especificación del RF Realizar verificación de hipótesis de varianza.....	24
Tabla 3: DCP para el RF Realizar verificación de hipótesis de varianza.....	37
Tabla 4: Resultados de las pruebas de rendimiento.....	40
Tabla 5: Especificación del RF Realizar estimación puntual de varianza.....	45
Tabla 6: Especificación del RF Realizar estimación puntual de media.....	45
Tabla 7: Especificación del RF Realizar estimación puntual de desviación típica.....	46
Tabla 8: Especificación del RF Realizar estimación por intervalos de varianza.....	47
Tabla 9: Especificación del RF Realizar estimación por intervalos de desviación típica.....	48
Tabla 10: Especificación del RF Realizar estimación bayesiana.....	49
Tabla 11: Especificación del RF Realizar verificación de hipótesis de media con varianza conocida.....	50
Tabla 12: Especificación del RF Mostrar el resultado de los cálculos en una gráfica.....	51
Tabla 13: DCP Realizar estimación puntual de varianza.....	59
Tabla 14: DCP Realizar estimación por intervalos de media con varianza conocida.....	60
Tabla 15: DCP Realizar estimación bayesiana.....	62
Tabla 16: DCP Realizar verificación de hipótesis de media con varianza conocida.....	63
Tabla 17: DCP Realizar test de K-S.....	65

Índice de Ecuaciones

Ecuación 1: Media aritmética.....	5
Ecuación 2: Varianza.....	5
Ecuación 3: Desviación típica.....	6

Introducción

En la actualidad cada vez son mayores los flujos de información que poseen las grandes empresas, instituciones gubernamentales y educativas, la banca, entre otros negocios; por tanto, es imprescindible el uso de la estadística como herramienta que permita la recolección, procesamiento, análisis e interpretación de datos. También es necesario el uso de aplicaciones informáticas, con las cuales se llevan a cabo los procesos mencionados anteriormente, debido a que sería lento y engorroso que se ocupase de estas tareas el personal humano. Entre dichas aplicaciones informáticas se encuentran los Sistemas de Información Geográfica (en lo adelante SIG), los cuales contribuyen a la gestión de la información socioeconómica de cualquier sector de la sociedad a partir de su representación y análisis espacial, lo cual es posible debido a que dicha información se encuentra referenciada geográficamente, apoyando de tal forma a la toma de decisiones.

La LPS (Línea de Producción de Software) Aplicativos SIG perteneciente al Centro Geoinformática y Señales Digitales de la Universidad de las Ciencias Informáticas se especializa en el desarrollo de Sistemas de Información Geográfica. La LPS Aplicativos SIG tiene como herramienta fundamental para el desarrollo de SIG la plataforma soberana GeneSIG que permite realizar representaciones y análisis de información referenciada geográficamente. Además, 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.

A pesar de que la plataforma GeneSIG posee un Componente de Estadística Descriptiva que permite calcular varios parámetros estadísticos, la toma de decisiones en ocasiones se ve afectada debido a la imposibilidad de ejecución de análisis estadísticos inferenciales que permitan la realización de inferencias sobre una población aleatoria, a partir de una muestra, lo cual podría afectar considerablemente las decisiones de los decisores al no contar con la disponibilidad de datos necesarios. Si luego de la representación de un fenómeno no es posible obtener los datos estadísticos suficientes, la decisión podría no ser rigurosamente acertada pues se desconocerían aristas importantes, tales como análisis referidos a estimación y verificación de hipótesis que son a menudo de interés para agentes decisores.

Lo anterior cobra particular interés cuando la información es mostrada sobre un mapa, siendo así una fuente de información muy efectiva, lo cual le proporciona 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 puede verse comprometida 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. Por si fuera poco, existen diversos SIG tales como ArcGIS, gvSIG y QuantumGIS que cuentan con estas funcionalidades desarrolladas lo cual limita mucho más las posibilidades reales de competencia con ellos.

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 en la LPS Aplicativos SIG, haciendo uso de la plataforma GeneSIG?

Se propone para la presente investigación como **Objetivo General** desarrollar un componente de estadística inferencial para fortalecer el apoyo a la toma de decisiones en los SIG desarrollados con GeneSIG, tomando como **Objeto de Estudio** los métodos de la estadística inferencial aplicados a los SIG y como **Campo de Acción** la informatización de los métodos de estimación y verificación de hipótesis de la Estadística Inferencial en la plataforma GeneSIG.

A continuación, se enumeran las **Preguntas Científicas** tomadas en cuenta para guiar el desarrollo de la investigación:

1. ¿Qué es la estadística inferencial?
2. ¿Cuáles de los métodos de la estadística inferencial se pueden aplicar a un SIG?
3. ¿Qué estructura posee la plataforma GeneSIG?
4. ¿Cómo diseñar y desarrollar un componente de estadística inferencial para la plataforma GeneSIG?
5. ¿Cuáles pruebas de validación deben ser aplicadas al componente desarrollado?

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

1. Analizar los métodos de estadística inferencial aplicados a los SIG.
2. Valorar las soluciones existentes que responden al campo de acción de la investigación, sus limitaciones y fortalezas.
3. Analizar las principales herramientas, tecnologías, lenguajes y metodologías a utilizar para la construcción de la propuesta de solución.
4. Implementar la solución propuesta.
5. Desarrollar el proceso de pruebas al módulo implementado.

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, 1997). Este método se emplea para analizar la evolución de los conceptos asociados a la Estadística Inferencial permitiendo formar una definición propia.

Analítico-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, 1997). 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 que se emplearán en el desarrollo de la solución propuesta.

Empíricos:

Análisis documental: El análisis documental es un trabajo mediante el cual por un proceso intelectual extraemos unas nociones del documento para representarlo y facilitar el acceso a los originales. Analizar, por tanto, es derivar de un documento el conjunto de palabras y símbolos que le sirvan de representación (Rubio Liniers, 1998). En la presente investigación se emplean los procedimientos sugeridos por el análisis documental para obtener los datos necesarios de la bibliografía consultada, plasmarlos en el presente documento, y dejar de la mano las fuentes de información para su posterior consulta.

El presente documento está estructurado de la siguiente forma:

Capítulo 1: Fundamentación Teórica. En este capítulo se exponen los principales conceptos asociados al problema en cuestión para un mejor entendimiento del mismo. Además, se lleva a cabo el análisis de las soluciones existentes.

Capítulo 2: Herramientas y tecnologías. En este capítulo se hace referencia a una serie de herramientas y tecnologías que se utilizan para obtener una solución al problema propuesto, tratando de garantizar de tal forma un producto con la mayor calidad posible.

Capítulo 3: Análisis y arquitectura del sistema. En este capítulo comienza el desarrollo de la solución propuesta según el proceso de desarrollo de software seleccionado. Se definen el modelo del dominio, los requisitos funcionales y no funcionales, y la arquitectura del sistema.

Capítulo 4: Diseño, implementación y pruebas. En este capítulo se realiza el modelado de la base de datos, los diagramas de clases del diseño, de componentes y de despliegue; finalmente se describe el proceso de pruebas realizado sobre la solución.

Capítulo 1: Fundamentación Teórica

1.1 Introducción

Este capítulo estará constituido por la fundamentación teórica de la solución propuesta, en la cual se definen varios conceptos asociados a la estadística inferencial, así como su importancia y otros aspectos que la comprenden. Además, se lleva a cabo un análisis general de la Plataforma Soberana GeneSIG en cuanto a sus principales módulos, funcionalidades y tecnologías sobre las cuales está desarrollada. Por último, queda reflejado un análisis sobre las posibles soluciones existentes.

1.2 Estadística Inferencial

La estadística inferencial es una técnica mediante la cual se obtienen generalizaciones o se toman decisiones en base a una información parcial o completa obtenida mediante técnicas descriptivas (García Mancilla, 2003).

Según otro criterio, la estadística inferencial se ocupa de las relaciones entre el modelo matemático y la práctica, constituyendo, en cierta forma, la rama aplicada de la estadística (Sironvalle, 2000).

Por tanto, se puede definir a la estadística inferencial como una rama de la estadística, que aplica sus métodos y técnicas sobre una muestra perteneciente a una población, para así realizar inferencias sobre esta última.

En caso de que no sea factible realizar un estudio completo por cuestiones de tiempo, recursos o costo, se puede calcular un tamaño de muestra para medir solo algunos elementos de la población, posteriormente se infiere que el resto de la población se comporta igual que la muestra tomada.

Para que una investigación pueda ser realmente válida se debe tener en cuenta las técnicas de muestreo, las cuales facilitan la obtención de resultados precisos e importantes para tener en cuenta al momento de darle solución a la situación que se estudia y de esta forma inferir en conclusiones sobre las poblaciones a partir del análisis de las muestras.

Las decisiones bajo incertidumbre y en general llevadas a cabo por los decisores, presentan una gran dependencia de los métodos estadísticos, evidenciando de tal forma el apoyo a la toma de decisiones. Además, en las empresas es utilizada la estadística para mejorar y optimizar los procesos de producción proporcionando de tal forma mayor calidad en los productos y servicios, a la vez que se ahorra tiempo y dinero.

1.2.1 Aspectos importantes que comprende la estadística inferencial

Con el objetivo de que el lector pueda tener una mayor comprensión de los temas abordados en el presente documento, se describen a continuación un grupo de conceptos generales asociados a los métodos inferenciales:

- Población: Conjunto de todos los individuos (personas, animales, plantas, cosas) de los que interesa estudiar ciertos datos.
- Muestra: Subconjunto de la población, que sirve para hacer inferencias sobre la misma.
- Muestreo aleatorio probabilístico: Consiste en elegir una muestra de una población al azar.

La Estadística dispone de cierta cantidad de métodos que se engloban dentro de la llamada Estadística Inferencial, los cuales se usan esencialmente para determinar la probabilidad de que una conclusión sacada a partir de los datos de una muestra sea cierta en la población muestreada. A continuación, se enumeran una serie de métodos con algunos de sus conceptos asociados, los cuales comprende la estadística inferencial:

1. Estimación de parámetros: Es el procedimiento utilizado para conocer las características de un parámetro poblacional, a partir del conocimiento de la muestra (Freund, y otros, 2006).

- Estimación puntual: se refiere a la elección de un estadístico, es decir, un número calculado a partir de datos muestrales, respecto al cual tenemos alguna esperanza o seguridad de que esté razonablemente cerca del parámetro que ha de estimar.
- Estimación por intervalo: se refiere a la elección de un estadístico, mediante el cual se obtiene un intervalo con un grado razonable de certeza, a partir de datos muestrales, y se espera que éste contenga al parámetro en cuestión.
- La estimación bayesiana plantea el uso de estimadores, dónde a partir de una distribución a priori (que son creencias previas sobre posibles valores) y evidencias muestrales directas, se conduce a una distribución a posteriori.
- Media aritmética: Es el valor medio ponderado de la serie de datos. Dada la muestra, la medida típica más comúnmente utilizada es la media definida por:

$$\text{Ecuación 1: Media aritmética. } \bar{X} = \frac{X_1 + X_2 + \dots + X_n}{n} = \frac{1}{n} \sum_{i=1}^n X_i$$

- Varianza: Mide la distancia existente entre los valores de la serie de datos y la media. Se define como:

$$\text{Ecuación 2: Varianza. } \delta^2 = \frac{1}{n-1} \sum_{i=1}^n (X_j - \bar{X})^2$$

- Desviación típica: se calcula como:

Ecuación 3: Desviación típica. $\delta = \sqrt{\delta^2}$ (Sironvalle, 2000)

- Intervalo de confianza: Se llama así a un intervalo en el que sabemos que está un parámetro, con un nivel de confianza específico.
 - Nivel de confianza: Probabilidad de que el parámetro a estimar se encuentre en el intervalo de confianza.
 - Error de estimación admisible: Que estará relacionado con el radio del intervalo de confianza.
2. Verificación de hipótesis: es un procedimiento que permite cuantificar la probabilidad del error que se cometería cuando se hace una afirmación sobre la población bajo estudio, es decir, permite medir la fuerza de la evidencia que tienen los datos a favor o en contra de alguna hipótesis de interés sobre la población (Freund, y otros, 2006).
- Hipótesis: Cualquier conjetura, a priori y evaluable, sobre la realidad o fenómeno de interés.
 - Error tipo I: se comete error tipo I, cuando se rechaza la hipótesis nula (H_0), siendo esta verdadera.
 - Error tipo II: se comete error tipo II, cuando se acepta la hipótesis nula (H_0), siendo esta realmente falsa.
3. Regresión Lineal: Método estadístico que aplica modelos matemáticos para aproximar la relación de dependencia entre variables independientes X_i , un término aleatorio ϵ y una variable dependiente Y (Tusell, 2011).
4. Análisis de Clúster: Se utiliza la información de una serie de variables para cada sujeto u objeto y, conforme a estas variables se mide la similitud entre ellos. Una vez medida la similitud se agrupan en: grupos homogéneos internamente y diferentes entre sí (Uriel, y otros, 2002).

1.3 Sistemas de Información Geográfica

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).

De forma similar 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 georreferenciados, como un conjunto de operaciones para trabajar con esos datos. En cierto modo, un SIG es un mapa de orden superior (Jeffrey Star, 1990).

A consideración del autor un SIG puede ser definido como un conjunto de software, hardware, métodos, datos y personal especializado encargados de la captura, almacenamiento y análisis de información referenciada geográficamente; la que puede ser representada de forma cartográfica. Además, es importante destacar que un SIG está compuesto por diversos subsistemas.

Un SIG está compuesto de tres subsistemas fundamentales:

- 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.
- Subsistema de visualización y creación cartográfica, que permite la interacción con los datos mediante mapas, leyendas, así como la edición de dichos datos.
- Subsistema de análisis, que provee los métodos y procesos para el análisis de los datos.

1.4 Descripción general de GeneSIG

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 georreferenciación y localización de recursos, así como la inclusión de datos y ubicación de nuevos objetos sobre mapas.

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 (Membrides Espinosa, 2013).

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.

Una característica muy peculiar de esta plataforma es que, la fuente del servidor se divide en dos subsistemas, uno encargado de la interacción con los datos provenientes del navegador el cual fue denominado Client y otro que sería responsable de administrar la información correspondiente al tratamiento espacial, así su manipulación a través de la MapScript, este último fue denominado Server (Membrides Espinosa, 2013).

Principales funcionalidades de GeneSIG:

- Permite la representación geoespacial de la información asociada a cualquier negocio que lo requiera.
- 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.
- Integra la información socioeconómica existente (recursos humanos, activos fijos, entidades de servicios, lugares de interés, entre otros) con la información geográfica asociada (Membrides Espinosa, y otros, 2012).

1.4.1 Tecnologías y herramientas

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

- Lenguajes de desarrollo: PHP 5 y ExtJS 3.0.
- Entornos Integrados de Desarrollo: Netbeans 8.0.
- Sistema Gestor de BD: PostgreSQL (9.3) con la extensión geográfica PostGIS 2.1.
- Sistemas Operativos: Ubuntu 14.04.
- Herramienta de Modelado: Visual Paradigm 8.0
- Frameworks: CartoWeb.
- Servidor de Mapas: MapServer (6.4).
- Servidor Web: Apache 2.5.6

1.5 Soluciones existentes

En la actualidad la tecnología SIG posee un impacto social muy importante en casi todos los sectores de la sociedad, por tanto, hacer uso de dicha tecnología en cualquier ámbito social, posibilita el seguimiento y control de la información socioeconómica de sus objetivos sobre una ubicación espacial, de manera que garantice una elevada eficiencia en la toma de decisiones a partir de su representación y análisis espacial. Además, en un sistema de este tipo donde se lleve a cabo análisis estadístico sobre la información socioeconómica obteniendo de tal forma salidas estadísticas, se proporciona una nueva variable a tener en cuenta en la toma de decisiones. A continuación, se describen algunas soluciones:

1.5.1 GvSIG

Es un sofisticado Sistema de Información Geográfica que permite gestionar datos espaciales y realizar análisis complejos sobre estos. La aplicación es de código abierto, con licencia GPL (*General Public License* o licencia publica general) y gratuita. Se ha hecho especial hincapié desde sus inicios, en que gvSIG sea un proyecto extensible, de forma que los desarrolladores puedan ampliar las funcionalidades de la aplicación fácilmente, así como desarrollar aplicaciones totalmente nuevas a partir de las librerías utilizadas en gvSIG (Association, 2011).

GvSIG permite ver las estadísticas básicas de una capa abierta a través de sus opciones de propiedades de raster, tal y como se muestra en la siguiente interfaz gráfica (Fig. 1).



Fig.1: Propiedades de raster de una capa en gvSIG (Association, 2011).

Además, permite el cálculo de los valores estadísticos más utilizados, sobre un campo de una tabla (Fig.2).

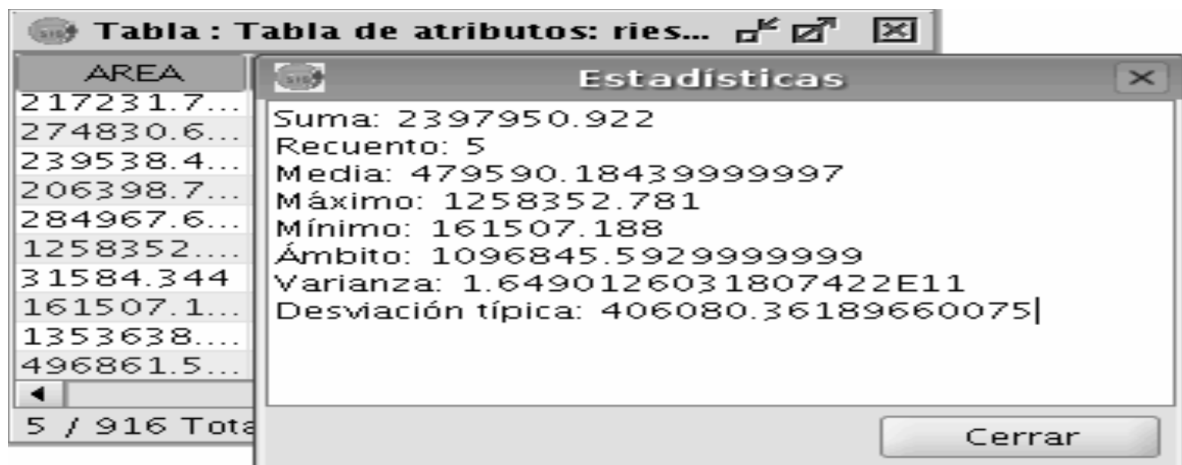


Fig.2: Ventana de estadísticas de una tabla (Association, 2011).

1.5.2 ArcGIS

Es un SIG integrado que consta de tres partes claves: el software ArcGIS Desktop como un conjunto integrado de aplicaciones SIG avanzadas, el ArcSDETM Gateway que es una interfaz para administrar las geodatabase (base de datos geográfica) en un sistema de administración de bases de datos (DBMS) y el software ArcIMS que es un SIG orientado a Internet para distribuir datos y servicios. La aplicación ArcGIS constituye una tecnología privativa, lo cual limita en gran medida las posibilidades de análisis sobre dicha herramienta (Caso Osorio, 2011).

ArcGIS permite agrupar un conjunto de estadísticas sobre un campo determinado, previamente seleccionado, para su posterior representación en una tabla.

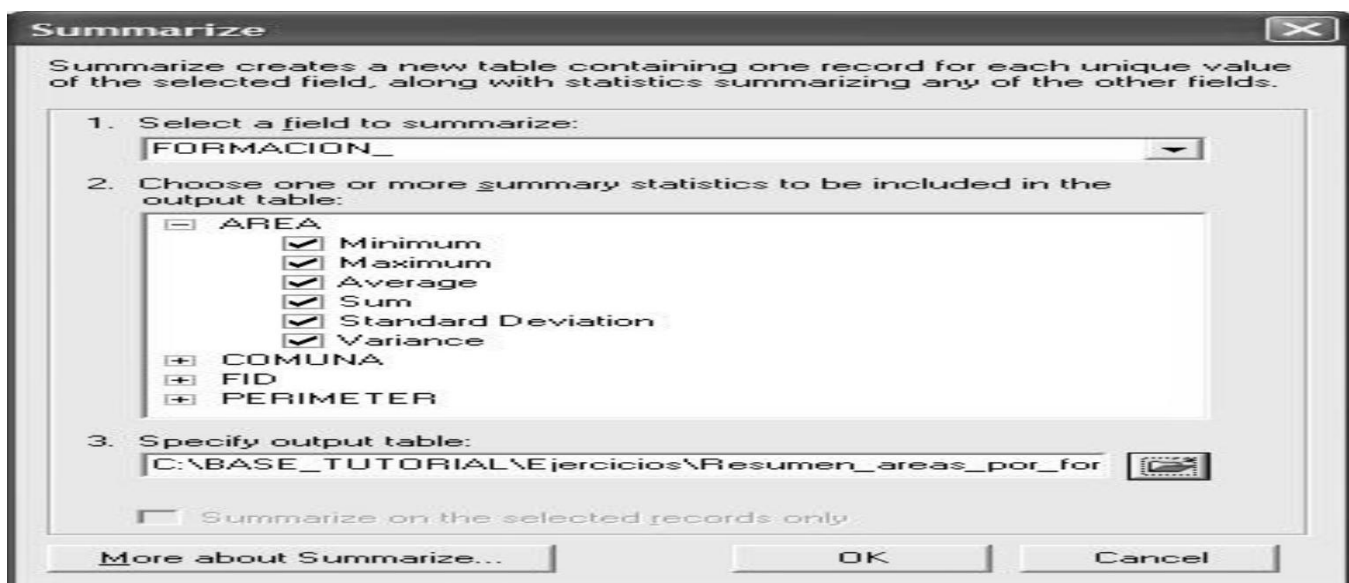


Fig.3: Agrupar características de un campo mediante opción Summarize (Caso Osorio, 2011).

OID	COMUNA	Count COMUNA	Minimum AREA	Maximum AREA	Average AREA	Sum AREA
0		19	53,785926	236259625,808	12604724,5598	239489766,6367
1	ALGARROBO	1	25836,103947	25836,103947	25836,1039	25836,1039
2	CASABLANCA	24	2292,184873	1434546,87094	135166,0202	3243984,4838
3	QUILPUE	15	7756,320419	455113,736261	96567,2418	1448508,6273
4	VALPARAISO	18	174,199807	580391,35578	85379,8945	1536838,1008
5	VILLA ALEMANA	9	5080,102387	81071,85003	21897,0958	197073,8626

Fig.4: Tabla de estadísticas previamente seleccionadas en Summarize (Caso Osorio, 2011).

1.5.3 QuantumGIS

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. La herramienta *fTools* es un *plugin* disponible en QGIS, el cual posee entre sus diversas funcionalidades: Estadística básica (Calcula estadísticas básicas (media, desviación típica, total, suma, varianza) en un archivo de entrada) (Athan, y otros, 2010).

Como resultado del análisis realizado sobre las distintas soluciones informáticas mencionadas anteriormente, se realiza la siguiente tabla resumen.

Tabla 1: Resumen de soluciones existentes.

Solución existente	GvSIG	ArcGIS	QGIS
Tecnología	Libre	Privativa	Libre
Entorno	Desktop	Desktop	Desktop
Motor estadístico	R	R	R
Estadísticos empleados	Media, suma, máximo, mínimo, desviación típica, varianza	Mínimo, máximo, media, suma, desviación típica, varianza	Media, varianza, desviación típica, suma

1.6 Conclusiones parciales

En el presente capítulo fueron analizados los principales conceptos asociados al campo de acción, lo que permitió tener un mayor entendimiento y comprensión de las temáticas que conforman la investigación, además, con el mismo fin, se realizó un análisis sobre los términos referentes a la estadística inferencial, los SIG y la plataforma GeneSIG. Luego de haber analizado diversas aplicaciones informáticas las cuáles proveen análisis estadísticos de datos, se puede llegar a la conclusión de que no satisfacen del todo las necesidades de un Componente Estadístico Inferencial como solución de la presente investigación, aunque se debe tener en cuenta el cálculo de diferentes estadísticos presentes en dichos sistemas; y la forma de activación de las funciones. Además, se debe tener presente que las tecnologías con que han sido desarrollados los sistemas analizados no se adecuan con las empleadas en la LPS Aplicativos SIG con que fue desarrollada GeneSIG, y el hecho de tomarlas como solución final implicaría un gasto excesivo de tiempo y esfuerzo, lo cual se puede evitar con el desarrollo de una solución que utilice solo el motor estadístico R y algunos aspectos de interés mencionados anteriormente que son empleados por dichas soluciones.

Capítulo 2: Herramientas y tecnologías

2.1 Introducción

En este capítulo se hace referencia a una serie de herramientas y tecnologías que se utilizan para obtener una solución al problema propuesto, tratando de garantizar de tal forma un producto con la mayor calidad posible. Serán argumentados en cuanto a su uso e importancia cada uno de los recursos seleccionados, entre los que se puede contar con lenguajes de programación y de modelado, herramientas CASE, sistema gestor de base de datos, así como una metodología de desarrollo de software.

2.2 Proceso de desarrollo de software

Para construir la ingeniería del software adecuadamente, se debe definir un proceso de desarrollo de software. En un proceso de software, *“se establece un marco común del proceso definiendo un pequeño número de actividades del marco de trabajo que son aplicables a todos los proyectos del software, con independencia de su tamaño o complejidad. Un número de conjuntos de tareas -cada uno es una colección de tareas de trabajo de ingeniería del software, hitos de proyectos, productos de trabajo, y puntos de garantía de calidad- que permiten que las actividades del marco de trabajo se adapten a las características del proyecto del software y a los requisitos del equipo del proyecto. Finalmente, las actividades de protección -tales como garantía de calidad del software, gestión de configuración del software y medición- abarcan el modelo de procesos”* (Pressman, 2005).

Un proceso de desarrollo de software tiene como objetivo la producción eficiente de un producto de software que satisfaga los requisitos de un especialista funcional con una planificación y una estimación de recursos predecibles. Los elementos de un proceso y sus relaciones deben responder *Quién* debe hacer *Qué*, *Cuándo* y *Cómo*. Esto se logra modelando las interacciones y relaciones que suceden entre las personas, las actividades que éstas desarrollan y los artefactos que se crean o actualizan durante el proceso (UCID, 2012).

2.2.1 Prodesoft

Como proceso de desarrollo de software a emplear para darle solución a la situación planteada queda definido Prodesoft v1.5 (Proceso de Desarrollo de Software) en correspondencia con las líneas bases de la Soberana Plataforma GeneSIG. Prodesoft describe la secuencia de actividades de alto nivel para la construcción y desarrollo de soluciones, mediante la combinación entre los modelos basado en Componentes, el Iterativo y el Incremental (UCID, 2012).

Según Prodesoft el ciclo de vida de un proyecto está compuesto por 5 fases: inicio, modelación, construcción, explotación experimental y despliegue (Fig.5); donde cada fase terminará en un hito con el objetivo fundamental de evaluar y decidir el paso a la siguiente fase de desarrollo.



Fig.5: Fases del ciclo de vida del proyecto en Prodesoft (UCID, 2012).

En la fase de Inicio queda definido el cronograma general con las actividades a desarrollar durante todo el ciclo de vida del proyecto. En la Modelación se desarrollan los siguientes artefactos: Modelo conceptual, Especificación de requisitos del software, Modelo de datos, Modelo de diseño, Modelo de implementación, Modelo de despliegue y Diseño de casos de prueba. En la fase de Construcción se obtiene el código fuente de la versión liberada del producto, y en la Explotación Experimental se obtiene la versión estable del producto. Por último en la fase de Despliegue se instala y configura el sistema para un ambiente de producción real.

2.3 Lenguaje de Modelado UML

Cualquier rama de ingeniería o arquitectura ha encontrado útil desde hace mucho tiempo la representación de los diseños de forma gráfica. Se necesitaba por tanto un lenguaje no sólo para comunicar las ideas a otros desarrolladores sino también para servir de apoyo en los procesos de análisis de un problema. Con este objetivo se creó el Lenguaje Unificado de Modelado (del inglés *Unified Modeling Language*, UML). UML se ha convertido en ese estándar tan ansiado para representar y modelar la información con la que se trabaja en las fases de análisis y, especialmente, de diseño (Hernández Oralo, 2003). Para el modelado del sistema será utilizado UML 2.0.

Los objetivos de UML son muchos, pero se pueden sintetizar sus funciones:

- Visualizar: UML permite expresar de una forma gráfica un sistema de forma que otro lo puede entender.

- Especificar: UML permite especificar cuáles son las características de un sistema antes de su construcción.
- Construir: A partir de los modelos especificados se pueden construir los sistemas diseñados.
- Documentar: Los propios elementos gráficos sirven como documentación del sistema desarrollado que pueden servir para su futura revisión.

2.4 Visual Paradigm for UML

“Visual Paradigm for UML” es una herramienta CASE (del inglés *Computer Aided Software Engineering*) que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, implementación y pruebas. Ayuda a una rápida construcción de aplicaciones de calidad, mejores y a un menor coste. Permite construir diagramas de diversos tipos, código inverso, generar código desde diagramas y generar documentación. La herramienta también proporciona abundantes tutoriales, demostraciones interactivas y proyectos UML (Carballé, y otros, 2013).

Entre las ventajas que proporciona Visual Paradigm for UML se encuentran las siguientes:

- Dibujo. Facilita el modelado de UML, ya que proporciona herramientas específicas para ello. Esto también permite la estandarización de la documentación, ya que la misma se ajusta al estándar soportado por la herramienta.
- Corrección sintáctica. Controla que el modelado con UML sea correcto.
- Reutilización. Facilita la reutilización, ya que disponemos de una herramienta centralizada donde se encuentran los modelos utilizados para otros proyectos.
- Generación de código. Permite generar código de forma automática, reduciendo los tiempos de desarrollo y evitando errores en la codificación del software.
- Generación de informes. Permite generar diversos informes a partir de la información introducida en la herramienta.

La versión Visual Paradigm for UML 8.0 fue lanzada en febrero de 2010 por *Visual Paradigm International Limited*. Debido a los beneficios que brinda para el desarrollo de software, especialmente referentes al modelado, se decidió utilizar esta versión para el modelado de los diagramas que se desarrollan a lo largo del ciclo de vida del proyecto.

2.5 Lenguajes de programación

Un lenguaje de programación es un conjunto de reglas o normas que permiten asociar a cada programa correcto un cálculo que será llevado a cabo por un ordenador (sin ambigüedades). Por tanto, un lenguaje de programación es un convenio o acuerdo acerca de cómo se debe de interpretar el significado de los programas de dicho lenguaje (Ureña Almagro, 2012).

Dentro de los lenguajes de programación se encuentran los de bajo nivel y los de alto nivel que se caracterizan por expresar los algoritmos necesarios para la creación de programas informáticos de manera que facilitan la comunicación entre un humano y la computadora mediante signos convencionales cercanos al lenguaje natural.

2.5.1 PHP

Es un lenguaje basado en C y en Perl, que se ha diseñado pensando en darle la máxima versatilidad y facilidad de aprendizaje, por encima de la rigidez y coherencia semántica. Además se trata indudablemente del lenguaje script de servidor más popular (Sánchez Asenjo, 2012). Para el desarrollo de la solución propuesta será empleado PHP 5.0 de lado del servidor.

Entre las ventajas que posee PHP se pueden citar las siguientes:

- Multiplataforma: Se trata de un lenguaje que se puede lanzar en casi todas las plataformas de trabajo.
- Abierto y gratuito. Pertenece al software licenciado como GNU, la licencia del sistema Linux; lo que permite su distribución gratuita y que la comunidad mejore el código.
- Gran comunidad de usuarios. La popularidad de PHP, junto con la gran defensa que de él hacen los defensores del código abierto, permite tener una comunidad amplia y muy dinámica a la que acudir en caso de necesidad.
- Puede interactuar con muchos motores de bases de datos tales como: MySQL, MSSQL, Oracle, Informix y PostgreSQL.
- PHP generalmente es utilizado como módulo de Apache lo que lo hace extremadamente veloz, además está completamente escrito en C, por lo que se ejecuta rápidamente utilizando poca memoria (Vázquez Mariño, 2008).

2.5.2 PHP MapScript

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.

La librería MapScript de PHP constituye la vía de comunicación de las aplicaciones SIG con el servidor de mapas MapServer y por otra parte rompe 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 (Membrides Espinosa, 2010).

2.5.3 ExtJS 3.0

ExtJS es una librería de Javascript que permite construir aplicaciones web complejas. La librería ExtJS incluye: componentes de interfaz de usuario de alto nivel y personalizables, modelo de componentes extensibles, un API fácil de usar, licencias *Open source* y comerciales.

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.

2.5.4 Python

Python es un proyecto de código abierto y posee soporte e integración con otros lenguajes y herramientas. Es un lenguaje con una sintaxis muy limpia y que favorece un código legible, debido a que su principal objetivo es resultar fácil en cuanto a lectura y diseño. Se trata de un lenguaje interpretado o de script, con tipado dinámico, multiplataforma y orientado a objetos (González Duque, 2012). Es empleado Python en su versión 2.7 para la integración con el lenguaje estadístico R.

2.5.5 Lenguaje R

R es un lenguaje de programación interpretado, de distribución libre, y se mantiene en un ambiente para el cómputo estadístico y gráfico. Este software corre en distintas plataformas Linux, Windows, MacOS, e incluso en PlayStation 3. El término ambiente pretende caracterizarlo como un sistema totalmente planificado y coherente, en lugar de una acumulación gradual de herramientas muy específicas y poco flexibles, como suele ser con otro software de análisis de datos (Santana, y otros, 2014). R constituye uno de los más útiles a la hora de estructurar y manipular datos, especialmente cuando se trata de grandes colecciones de ellos.

Algunas características importantes de R son:

- La funcionalidad de R consta de paquetes modulares. El sistema base de R contiene el paquete básico que se requiere para su ejecución y la mayoría de las funciones fundamentales.
- R es muy útil para el trabajo interactivo, pero también es un poderoso lenguaje de programación para el desarrollo de nuevas herramientas.

- Tiene una comunidad muy activa, por lo que, haciendo las preguntas correctas rápidamente encontrarás la solución a los problemas que se te presenten en el ámbito de la programación con R.
- R cuenta con una muy buena ayuda en el uso de funciones.

Se decide emplear R 3.0.2 para el tratamiento estadístico de los datos socioeconómicos.

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. Es un producto libre y gratuito sin restricciones de uso.

La plataforma NetBeans permite que las aplicaciones sean desarrolladas a partir de un conjunto de componentes de software llamados módulos. Un módulo es un archivo java que contiene clases de java escritas para interactuar con las APIs de NetBeans y un archivo especial (*manifest file*) que lo identifica como módulo. Las aplicaciones construidas pueden ser extendidas agregándole nuevos módulos. Debido a que los módulos pueden ser desplegados independientemente, las aplicaciones basadas en la plataforma NetBeans pueden ser extendidas fácilmente por otros desarrolladores de software.

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.

2.7 Sistema Gestor de Base de Datos

Si una base de datos son los datos almacenados, entonces el Sistema Gestor de Bases de Datos (en lo adelante SGBD) es el programa o conjunto de programas que gestionan y mantienen consistentes estos datos. Por tanto, un SGBD puede definirse como un sistema software de propósito general que facilita los procesos de definición, construcción y manipulación de la base de datos para distintas aplicaciones (Garsón Pérez, 2010).

Entre las múltiples funciones que cumplen los SGBD se destacan:

- Facilitar al administrador la descripción de los elementos de datos que integran la base de datos, su estructura y las relaciones que existen entre ellos, así como los controles a efectuar antes de realizar el acceso a la base de datos. Esta función es llevada a cabo por medio de Lenguajes de Definición de Datos (LDD).
- Facilitar la manipulación de datos, permitiendo a los usuarios de la base de datos añadir, buscar, suprimir o modificar los datos de la misma. Se lleva a cabo por medio de Lenguajes de Manipulación de Datos (LMD).
- Garantizar la seguridad y la confidencialidad de la base de datos.

- Garantizar la fiabilidad e integridad de los datos.
- Garantizar la compatibilidad de los datos. Permitir que varios usuarios puedan utilizar los mismos datos sin que se produzcan problemas en los accesos simultáneos (Tovar, 2003).

2.7.1 PostgreSQL como SGBD

PostgreSQL es un avanzado sistema de bases de datos relacionales basado en *Open Source*. Esto quiere decir que el código fuente del programa está disponible a cualquier persona libre de cargos directos, permitiendo a cualquiera colaborar con el desarrollo del proyecto o modificar el sistema para ajustarlo a sus necesidades (Denzer, 2002).

Entre los sistemas de bases de datos existentes hoy en día, PostgreSQL juega un papel muy importante ya que es un sistema que tiene muchas cualidades que lo hacen ser una muy buena alternativa para el manejo de las bases de datos de empresas, universidades y otros sitios en general que necesiten de una potente herramienta para la gestión de sus bases de datos. De las cualidades anteriormente mencionadas se puede destacar por su importancia: la robustez, escalabilidad y que su código fuente está disponible libremente publicado bajo la licencia BSD (*Berkeley Software Distribution*). Se emplea PostgreSQL 9.3 en correspondencia con la línea base de GeneSIG.

2.7.2 PostGIS: la extensión geográfica del gestor seleccionado

PostGIS está creado por *Refractions Research Inc*, como un proyecto de investigación de tecnologías de bases de datos espaciales. Está publicado bajo licencia GNU. PostGIS es una extensión al SGBD PostgreSQL que permite el uso de objetos SIG. PostGIS incluye soporte para índices GiST basados en R-Tree, y funciones básicas para el análisis de objetos SIG. Esta extensión soporta objetos SIG de características simples, tales como puntos, líneas, polígonos, multilíneas, multipuntos, y colecciones geométricas (Ramsey, 2011). Es utilizado PostGIS 2.1 como extensión geográfica del SGBD seleccionado.

2.8 PgAdmin III para el manejo de PostgreSQL

PgAdmin III es una herramienta de código abierto para la administración de bases de datos PostgreSQL y derivados (*EnterpriseDB Postgres Plus Advanced Server* y *Greenplum Database*) (Reingart, 2011).

PgAdmin III incluye:

- Interfaz administrativa gráfica.
- Herramienta de consulta SQL (con un EXPLAIN gráfico).
- Editor de código procedural.
- Agente de planificación SQL/shell/batch.
- Administración de Slony-I.

PgAdmin se diseña para responder a las necesidades de la mayoría de los usuarios, desde escribir simples consultas SQL hasta desarrollar bases de datos complejas. En el presente trabajo es utilizado PgAdmin III es su versión 1.18.

La interfaz gráfica soporta todas las características de PostgreSQL y hace simple la administración. Está disponible en más de una docena de lenguajes y para varios sistemas operativos, incluyendo Microsoft Windows, Linux, FreeBSD, Mac OSX y Solaris.

2.9 Servidor de mapas MapServer

MapServer es un entorno de desarrollo en código abierto para la creación de aplicaciones SIG con el fin de visualizar, consultar y analizar información geográfica a través de la red mediante la tecnología Web. Este puede ser utilizado como una aplicación CGI (*Common Gateway Interface*) o a través del acceso a la API de MapServer que proveen las librerías Mapscript (Membrides Espinosa, 2010). Se emplea MapServer en su versión 6.4.

2.10 Conclusiones Parciales

Tras realizar el análisis de los argumentos planteados a lo largo del presente capítulo sobre las herramientas y tecnologías seleccionadas, se puede concluir que dichos recursos al ser los mismos empleados para el desarrollo de la soberana plataforma GeneSIG y que están definidos por la LPS Aplicativos SIG, permiten el diseño e implementación de un componente perfectamente integrable con la mencionada plataforma. El empleo de estas herramientas y tecnologías permiten obtener un producto de calidad, además de apoyar la filosofía de explotación de software libre que sigue el país.

Capítulo 3: Análisis y arquitectura del sistema.

3.1 Introducción

En el presente capítulo se plantea el modelo de dominio de la solución, además se enumeran y especifican los Requisitos Funcionales (RF) y los Requisitos No Funcionales (RNF) que debe cumplir el sistema para considerarse exitoso. Se define la arquitectura del sistema y los patrones de diseño correspondientes para darle solución al problema planteado. Finalmente queda definido el estándar de codificación que será utilizado en la solución.

3.2 Modelo de dominio

La ingeniería del dominio explora un dominio de aplicaciones con la intención de encontrar específicamente los componentes de datos funcionales y de comportamiento candidatos para la reutilización. “La ingeniería del dominio crea un modelo de dominio de aplicación que se utiliza como base para analizar los requisitos del usuario en el flujo de la ingeniería del software” (Pressman, 2005).

El objetivo fundamental que se persigue al realizar el modelo de dominio es comprender y describir las clases más importantes dentro del contexto del sistema. Además, éste ayuda a los usuarios a mantener una terminología común, lo que les permite compartir el conocimiento de forma más consistente. El modelo de dominio de la solución planteada está representado mediante un diagrama de clases del dominio (Fig. 6).

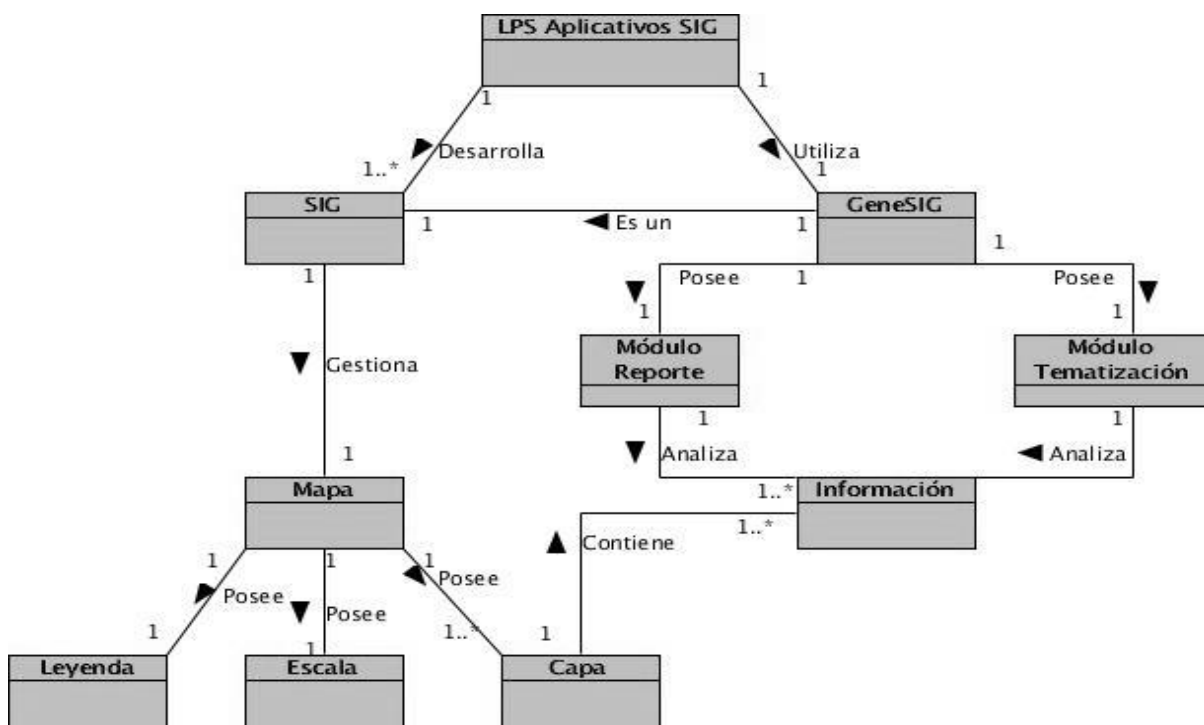


Fig.6: Modelo de Dominio de la solución.

La plataforma GeneSIG es utilizada por la LPS Aplicativos SIG para el desarrollo de SIG. Su función principal es la gestión de mapas, los cuales están conformados por leyendas, capas y escalas; y en ellos se representa e integra información. Esta información es analizada por los módulos de Reporte y Tematización, los cuales pertenecen a GeneSIG.

3.2.1 Definición de clases del diagrama del dominio

LPS Aplicativos SIG: Línea de Productos de Software especializada en el desarrollo de SIG haciendo uso de GeneSIG.

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

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

Mapa: Es una representación gráfica y métrica de una porción de territorio. El que el mapa tenga propiedades métricas significa que ha de ser posible tomar medidas de distancia, ángulos o superficies sobre él y obtener un resultado aproximadamente exacto (Membrides Espinosa, y otros, 2012).

Escala: Relación entre la distancia que separa dos puntos en un mapa y la distancia real de esos dos puntos en la superficie terrestre.

Capa: Conjunto de información temática descrita y almacenada en una base de datos, y estructurada según temas que se usan para elaborar los SIG.

Leyenda: Explicación de los símbolos, los colores, las tramas y los sombreados empleados en un mapa.

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

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

Información: Conjunto de datos tanto de carácter geográfico como socioeconómico que son gestionados por los SIG.

3.3 Requisitos del Software

Los requisitos para un sistema son la descripción de los servicios proporcionados por el sistema y sus restricciones operativas. Estos requisitos reflejan las necesidades de los clientes de un sistema que ayude a resolver algún problema como el control de un dispositivo, hacer un pedido o encontrar información. El proceso de descubrir, analizar, documentar y verificar estos servicios y restricciones se denomina ingeniería de requisitos (Sommerville, 2005). A menudo, los requisitos de sistemas de software se clasifican en funcionales y no funcionales.

3.3.1 Requisitos funcionales

Los RF son declaraciones de los servicios que debe proporcionar el sistema, de la manera en que éste debe reaccionar a entradas particulares y de cómo se debe comportar en situaciones particulares. En

algunos casos, los RF de los sistemas también pueden declarar explícitamente lo que el sistema no debe hacer (Sommerville, 2005).

El sistema debe cumplir con los siguientes RF:

RF1 Realizar estimación puntual de varianza: Esta funcionalidad permite realizar la estimación puntual de la varianza de un atributo dado y posteriormente tabular el resultado obtenido.

RF2 Realizar estimación puntual de media: Esta funcionalidad permite realizar la estimación puntual de la media de un atributo dado y posteriormente tabular el resultado obtenido.

RF3 Realizar estimación puntual de desviación típica: Esta funcionalidad permite realizar la estimación puntual de la desviación típica de un atributo dado y posteriormente tabular el resultado obtenido.

RF4 Realizar estimación por intervalos de varianza: Esta funcionalidad permite realizar la estimación por intervalos de la varianza de un atributo dado y posteriormente tabular el resultado obtenido.

RF5 Realizar estimación por intervalos de desviación típica: Esta funcionalidad permite realizar la estimación por intervalos de la desviación típica de un atributo dado y posteriormente tabular el resultado obtenido.

RF6 Realizar estimación por intervalos de media con varianza conocida: Esta funcionalidad permite realizar la estimación por intervalos de la media con varianza conocida de un atributo dado y posteriormente tabular el resultado obtenido.

RF7 Realizar estimación por intervalos de media con varianza desconocida: Esta funcionalidad permite realizar la estimación por intervalos de la media con varianza desconocida de un atributo dado y posteriormente tabular el resultado obtenido.

RF8 Realizar estimación bayesiana: Esta funcionalidad permite realizar la estimación bayesiana de la media de un atributo dado y posteriormente tabular el resultado obtenido.

RF9 Realizar verificación de hipótesis de varianza: Esta funcionalidad permite realizar la verificación de hipótesis de varianza de un atributo dado y posteriormente tabular el resultado obtenido, mostrando de tal forma si la hipótesis nula (H_0) es aceptada o rechazada.

RF10 Realizar verificación de hipótesis de media con varianza conocida: Esta funcionalidad permite realizar la verificación de hipótesis de media con varianza conocida de un atributo dado y posteriormente tabular el resultado obtenido, mostrando de tal forma si la hipótesis nula (H_0) es aceptada o rechazada.

RF11 Realizar verificación de hipótesis de media con varianza desconocida: Esta funcionalidad permite realizar la verificación de hipótesis de media con varianza desconocida de un atributo dado y

posteriormente tabular el resultado obtenido, mostrando de tal forma si la hipótesis nula (H_0) es aceptada o rechazada.

RF12 Realizar test de K-S: Esta funcionalidad permite realizar el test de K-S para un atributo especificado por el usuario y posteriormente tabular el resultado obtenido, mostrando de tal forma si la hipótesis nula (H_0) es aceptada o rechazada.

RF13 Realizar test de Chi-Cuadrado: Esta funcionalidad permite realizar el test de Chi-Cuadrado para un atributo especificado por el usuario y posteriormente tabular el resultado obtenido, mostrando de tal forma si la hipótesis nula (H_0) es aceptada o rechazada.

RF14 Realizar test de Carreras por encima y por debajo de la media: Esta funcionalidad permite realizar el test de Carreras por encima y por debajo de la media para un atributo especificado por el usuario y posteriormente tabular el resultado obtenido, mostrando de tal forma si la hipótesis nula (H_0) es aceptada o rechazada.

RF15 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.

RF16 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.

A continuación, se muestra la especificación del RF Realizar verificación de hipótesis de varianza, el resto de las especificaciones pueden ser consultadas en el Anexo I y en el artefacto “Especificación de requisitos”.

Tabla 2: Especificación del RF Realizar verificación de hipótesis de varianza.

Conceptos tratados	Conceptos	Atributos
	-	-
Precondiciones	Pre-condiciones	Pre-requisitos
	Abrir ventana “Formulario Estadístico”	No procede
Descripción	<ol style="list-style-type: none"> 1. Se selecciona la capa a la cual se aplicará la verificación de hipótesis de varianza. 2. Se selecciona el atributo al cual se aplicará la verificación de hipótesis de varianza. 3. Se selecciona el panel Verificación de Hipótesis. 4. Se selecciona como método de verificación de hipótesis la Varianza. 5. Se selecciona el caso de hipótesis, introduce el valor de la significación, y se da clic al botón Calcular. 6. Se muestra una nueva ventana que contiene Capa, Atributo, Función, Estimador, Valor Tabla y Decisión. 	

Validaciones	No se habilita el campo del atributo hasta que no sea seleccionada la capa. No se habilita el panel Verificación de Hipótesis hasta que no sea seleccionado el atributo. El valor de la significación es numérico entre 0 y 1.												
Post-condiciones	Será registrado el resultado de la verificación de hipótesis de varianza en la base de datos.												
Post-requisitos	No Procede												
Prototipo de Interfaz	<table border="1"> <thead> <tr> <th>Capa</th> <th>Atributo</th> <th>Funcion</th> <th>Estimador</th> <th>Valor_Tabla</th> <th>Decision</th> </tr> </thead> <tbody> <tr> <td>empresa</td> <td>cant_industriales</td> <td>VH_Varianza</td> <td>Est=689.282</td> <td>Chi1=26.757 & Ch...</td> <td>Rchz Ho</td> </tr> </tbody> </table>	Capa	Atributo	Funcion	Estimador	Valor_Tabla	Decision	empresa	cant_industriales	VH_Varianza	Est=689.282	Chi1=26.757 & Ch...	Rchz Ho
Capa	Atributo	Funcion	Estimador	Valor_Tabla	Decision								
empresa	cant_industriales	VH_Varianza	Est=689.282	Chi1=26.757 & Ch...	Rchz Ho								

3.3.2 Requisitos no funcionales

Los RNF, como su nombre sugiere, son aquellos requisitos que no se refieren directamente a las funciones específicas que proporciona el sistema, sino a las propiedades emergentes de éste como la fiabilidad, el tiempo de respuesta y la capacidad de almacenamiento (Sommerville, 2005). De forma alternativa, definen las restricciones del sistema como la capacidad de los dispositivos de entrada/salida y las representaciones de datos que se utilizan en las interfaces del sistema. Los RNF rara vez se asocian con características particulares del sistema; más bien, estos requisitos especifican o restringen las propiedades emergentes del sistema.

El sistema debe cumplir con los siguientes RNF agrupados según su clasificación:

RNF1 Usabilidad

COMPONENTE DE ESTADÍSTICA INFERENCIAL PARA LA PLATAFORMA GENESIG

- El componente podrá ser utilizado por personas con conocimientos básicos sobre computación, estadística inferencial y los SIG.
- El componente estará orientado a íconos para facilitar el reconocimiento de las funcionalidades por parte de los usuarios.

RNF2 Restricciones de diseño

- Diseño sencillo de la aplicación donde no sea necesario mucho entrenamiento para utilizar el sistema.
- El producto de software debe diseñarse sobre una arquitectura cliente – servidor.
- Se debe lograr un producto altamente configurable, teniendo en cuenta que se desarrollará sobre la plataforma GeneSIG, por tanto, debe permitir la incorporación de nuevas funcionalidades.

RNF3 Software

Para las computadoras clientes:

- Un navegador web.

Para los servidores:

- Sistema operativo GNU/Linux Ubuntu Server 14.04.
- Servidor Web Apache 2.0 o superior, con módulo PHP 5 configurado con la extensión pgsql incluida.
- PostgreSQL 9.3 como Sistema Gestor de Base de Datos.
- PostGIS 2.1 como soporte de datos espaciales.
- MapServer 6.4, con extensión PHP MapScript.
- Python 2.7.
- R - base 3.0.2 como lenguaje de tratamiento estadístico.
- Paquete rpy2.

RNF4 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.

RNF5 Restricciones de licencia

- El componente puede ser utilizado, modificado y distribuido, sin necesidad de obtener la autorización de sus respectivos titulares.

RNF6 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 Arquitectura de software

La arquitectura del software de un programa o sistema de cómputo es la estructura o estructuras del sistema, que incluyen los componentes del software, las propiedades visibles externamente de esos componentes y las relaciones entre ellos. La arquitectura es una representación que permite que un ingeniero de software: analice la efectividad del diseño para cumplir con los requisitos establecidos, considere opciones arquitectónicas en una etapa en que aún resulta relativamente fácil hacer cambios al diseño, y reduzca los riesgos asociados con la construcción del software (Pressman, 2005).

Entre las posibles razones que fundamentan la importancia de la arquitectura de software no deben dejar de mencionarse las siguientes, puesto que son esenciales:

- Las representaciones de la arquitectura del software permiten la comunicación entre todas las partes interesadas en el desarrollo de un sistema de cómputo.
- La arquitectura destaca las decisiones iniciales relacionadas con el diseño que tendrán un impacto profundo en todo el trabajo de la ingeniería del software que le sigue y, lo que también resulta importante, en el éxito final del sistema como entidad operacional.
- La arquitectura constituye un modelo relativamente pequeño e intelectualmente comprensible de cómo está estructurado el sistema y cómo trabajan juntos sus componentes.

Debido a que el componente a desarrollar será para la plataforma GeneSIG, han sido seleccionadas las siguientes arquitecturas:

- 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, y otros, 2004).
- Orientada a objetos: Los componentes del estilo se basan en principios orientados a objetos: encapsulamiento, herencia y polimorfismo. Son asimismo las unidades de modelado, diseño e implementación, y los objetos y sus interacciones son el centro de las incumbencias en el diseño de la arquitectura y en la estructura de la aplicación. Las interfaces están separadas de las

implementaciones. Entre las cualidades, la más básica concierne a que se puede modificar la implementación de un objeto sin afectar a sus clientes. Asimismo, es posible descomponer problemas en colecciones de agentes en interacción. Además, por supuesto, un objeto es ante todo una entidad reutilizable en el entorno de desarrollo (Reynoso, y otros, 2004).

3.4.1 Estilo arquitectónico

Un estilo arquitectónico es una transformación impuesta al diseño de todo un sistema. Cada estilo describe una categoría de sistemas que abarca: un conjunto de componentes que realizan una función requerida por el sistema; un conjunto de conectores que permiten la comunicación, cooperación y coordinación entre los componentes; restricciones que definen cómo se integran los componentes para formar el sistema; y modelos semánticos que permiten a un diseñador, mediante el análisis de las propiedades conocidas de las partes que lo integran, comprender las propiedades generales de un sistema. El objetivo de un estilo arquitectónico es establecer una estructura para todos los componentes del sistema (Pressman, 2005).

En el desarrollo de la solución propuesta se emplea el estilo Llamada y Retorno, esta familia de estilos enfatiza la modificabilidad y la escalabilidad; además son los estilos más generalizados en sistemas en gran escala. Miembros de la familia son las arquitecturas de programa principal y subrutina, los sistemas basados en llamadas a procedimientos remotos, los sistemas orientados a objetos y los sistemas jerárquicos en capas (Reynoso, y otros, 2004).

3.5 Estándar de codificación

En cuanto al estándar de codificación se decidió emplear como estilo de escritura el CamelCase el cual es aplicado a frases o palabras compuestas en la declaración de variables o en el nombre de las funciones, con el objetivo de garantizar una mayor legibilidad en el código para un mejor entendimiento por parte de los interesados que interactúen con el componente.

Existen dos tipos de CamelCase:

- UpperCamelCase: Se pone de manifiesto cuando la primera letra de cada palabra en las frases o palabras compuestas se escribe con mayúscula. Ejemplo: `MostrarGridConfig`.
- lowerCamelCase: Se pone de manifiesto cuando la primera letra de la primera palabra en las frases o palabras compuestas se escribe con minúscula y el resto de las primeras letras con mayúscula. Ejemplo: `mostrarGridConfig`.

Una vez identificados los tipos de CamelCase existentes, se procede a seleccionar el lowerCamelCase para la implementación de la solución propuesta. Además del estilo de escritura que es un aspecto importante del estándar de codificación se tendrá en cuenta también el uso de comentarios que deben ser

concisos y concretos añadiendo de tal forma claridad al código. Una vez establecido los aspectos del estándar de codificación, se puede apreciar en la Fig. 7 cómo son aplicados.

```
//Obtener una cadena con los valores del atributo seleccionado
public function getValueAttr($atr,$from){
    $query = "SELECT  $atr FROM ! ";
    $db = $this->getPluginConnection();
    $queryResults = $db->getAll($query, array($from));
    $string=null;
    for($i=0;$i<count($queryResults);$i++){
        $string.= $queryResults[$i][0] . ",";
    }
    return $string;
}
```

Fig. 7: Función *getValueAttr* de la clase *ServerEstadistica.php*

3.6 Conclusiones parciales

Al establecer el Modelo del Dominio, el cual identifica una serie de elementos presentes en el contorno que se investiga, se impone de tal forma un punto de partida que permitió identificar un conjunto de 16 funcionalidades del sistema, clasificadas en 14 de prioridad alta y 2 de prioridad media. De igual modo fueron identificados un total de 6 requisitos no funcionales. Se plantea el uso de la arquitectura Basada en Componentes en correspondencia con la estructura de GeneSIG y la arquitectura Orientada a Objetos para el componente a desarrollar en particular. El empleo de forma correcta de los estándares de codificación, permitirá una mayor legibilidad, limpieza y organización en el código para posteriores consultas o aclaraciones de dudas.

Capítulo 4: Diseño, Implementación y pruebas.

4.1 Introducción

En el presente capítulo se lleva a cabo el modelado de diversos diagramas tales como el de Clases Persistentes y Entidad – Relación para elaborar la base de datos, en cuanto al modelo de diseño se tiene en cuenta el diagrama de Clases del Diseño y para el modelo de implementación se desarrolla el diagrama de Componentes. También se desarrolla el modelo de despliegue para describir la distribución física del sistema. Por último, se lleva a cabo el proceso de pruebas, para determinar si el sistema cumple o no con todos los requerimientos anunciados.

4.2 Diseño de la Base de Datos

En la actualidad las empresas manejan un inmenso flujo de datos, por tanto, es imprescindible para su almacenamiento el uso de bases de datos, las cuales aportan mayor seguridad debido a que los datos almacenados en formato duro tienden a deteriorarse, además la información presente en las bases de datos puede ser obtenida de forma sencilla mediante consultas, ahorrando de tal forma tiempo y dinero. Un paso crucial en la construcción de una base de datos es la fase de diseño de la misma, en la cual se asegura entre otros aspectos que las tablas estén debidamente definidas, con el único propósito de asegurar que la base de datos esté correctamente diseñada para que tenga eficiencia, evitar problemas en las consultas y que se pueda alargar su vida útil. Para el diseño de la base de datos de la solución propuesta han sido utilizados los diagramas de Clases Persistentes y Entidad - Relación.

4.2.1 Diagrama de Clases Persistentes

Representa un diagrama que contiene las clases que almacenan los datos que persistirán más allá de la ejecución el software. En la Fig. 8 se muestra el Diagrama de Clases Persistentes de la solución propuesta.

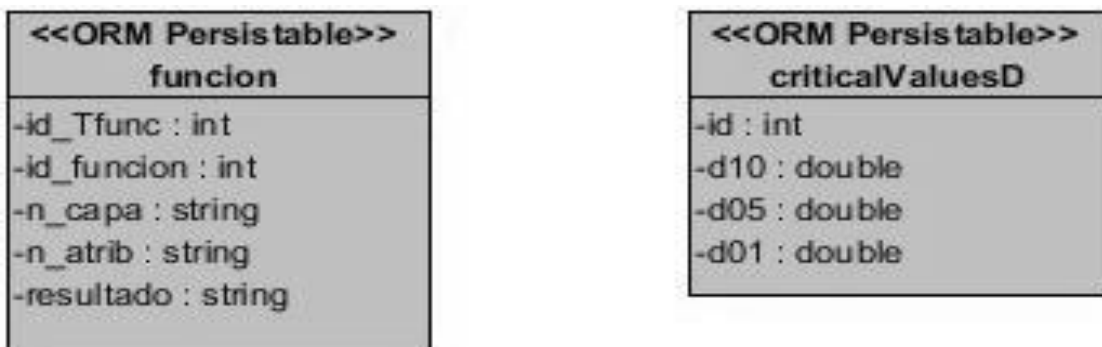


Fig.8: Diagrama de Clases Persistentes

4.2.2 Diagrama Entidad - Relación

Un Diagrama Entidad – Relación es una herramienta para el modelado de datos que permite representar un conjunto de objetos básicos de un sistema de información denominados entidades así como sus interrelaciones y propiedades (Storti, y otros, 2007). En la Fig. 9 se muestra el Diagrama Entidad – Relación de la solución propuesta.

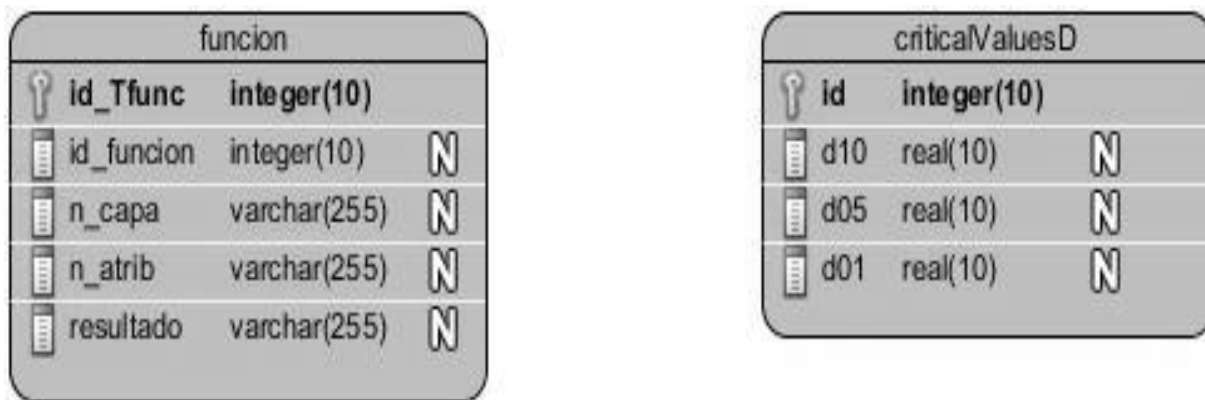


Fig.9: Diagrama Entidad - Relación

4.3 Modelo de Diseño

El modelo de diseño proporciona detalles acerca de las estructuras de datos, las arquitecturas, las interfaces y los componentes del software que son necesarios para implementar el sistema. El propósito del diseño es especificar una solución que trabaje y pueda ser fácilmente convertida en código fuente y construir una arquitectura simple y fácilmente extensible. Los diagramas en UML elaborados como parte del modelo de diseño proporcionan mayor detalle para la implementación específica y resaltan la estructura arquitectónica, los componentes que residen dentro de la arquitectura y las interfaces entre los componentes y con el mundo exterior (Pressman, 2005). Para la representación del modelo de diseño de la solución propuesta ha sido utilizado el Diagrama de Clases del diseño.

4.3.1 Patrones de diseño

Un patrón de diseño describe una estructura de diseño que resuelve un problema de diseño particular dentro de un contexto específico y en medio de fuerzas que pueden tener un impacto en la manera en que se aplica y utiliza el patrón. La finalidad de cada patrón de diseño es proporcionar una descripción que le permita al diseñador determinar si el patrón es aplicable al trabajo actual, si se puede reutilizar y si puede servir como guía para desarrollar un patrón similar, pero diferente en cuanto a la funcionalidad o estructura. Entre los patrones de asignación de responsabilidades se aplican los siguientes:

- Experto: Es un patrón que se usa más que cualquier otro al asignar responsabilidades; es un principio básico que suele utilizarse en el diseño orientado a objetos. Permite asignar la responsabilidad a la clase ServerEstadistica.php que tiene la información correspondiente para realizar todas las funcionalidades del sistema.

- Creador: El propósito fundamental de este patrón es encontrar un creador que se debe conectar con el objeto producido en cualquier evento. Se evidencia con la clase `formEstadistica.js` que hereda de `formEstadistica.ui.js`.
- Alta Cohesión: Es un patrón que se emplea con el fin de distribuir la información por clases mediante un conjunto de responsabilidades pequeño y enfocado, y aplica atributos y métodos de manera sencilla para implementar dichas responsabilidades, de forma tal que cada una maneje solo la información necesaria. Se pone de manifiesto en las clases `formEstadistica.js` y `formEstadistica.ui.js`, dónde se distribuye la lógica y la vista respectivamente, en vez de integrar toda la información en una sola clase.
- Bajo Acoplamiento: Soporta el diseño de clases más independientes, que reducen el impacto de los cambios, y también más reutilizables, que acrecienten la oportunidad de una mayor productividad. La gran mayoría de las relaciones empleadas en la aplicación son de tipo `<<use>>`, y existen pocas relaciones de tipo fuerte tales como composición y especialización.

Entre los patrones aplicados del Grupo de los cuatro (The Gang of Four, GOF) se encuentran los siguientes:

- Singleton: Patrón que garantiza la existencia de una única instancia para una clase y la creación de un mecanismo de acceso global a dicha instancia. Se evidencia en la clase `ServerContext` donde el método `getmapobject` crea una instancia del mapa, la cual es única en todo el sistema.
- Command: Patrón que permite encapsular las peticiones a través de un objeto, permitiendo ejecutar dicha operación sin necesidad de conocer el contenido de la misma. 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.

4.3.2 Diagrama de Clases del Diseño

Un diagrama de clases de diseño (DCD) corresponde a la realización de un RF y representa las especificaciones de las clases e interfaces software en una aplicación. Entre la información general obtenida mediante un DCD encontramos: clases, asociaciones y atributos, interfaces, con sus operaciones y constantes, métodos, información acerca del tipo de los atributos, navegabilidad y dependencias. En la Fig. 10 se muestra el DCD del RF Realizar verificación de hipótesis de varianza. El resto de los DCD pueden ser consultados en el Anexo II y en el artefacto "Modelo de Diseño".

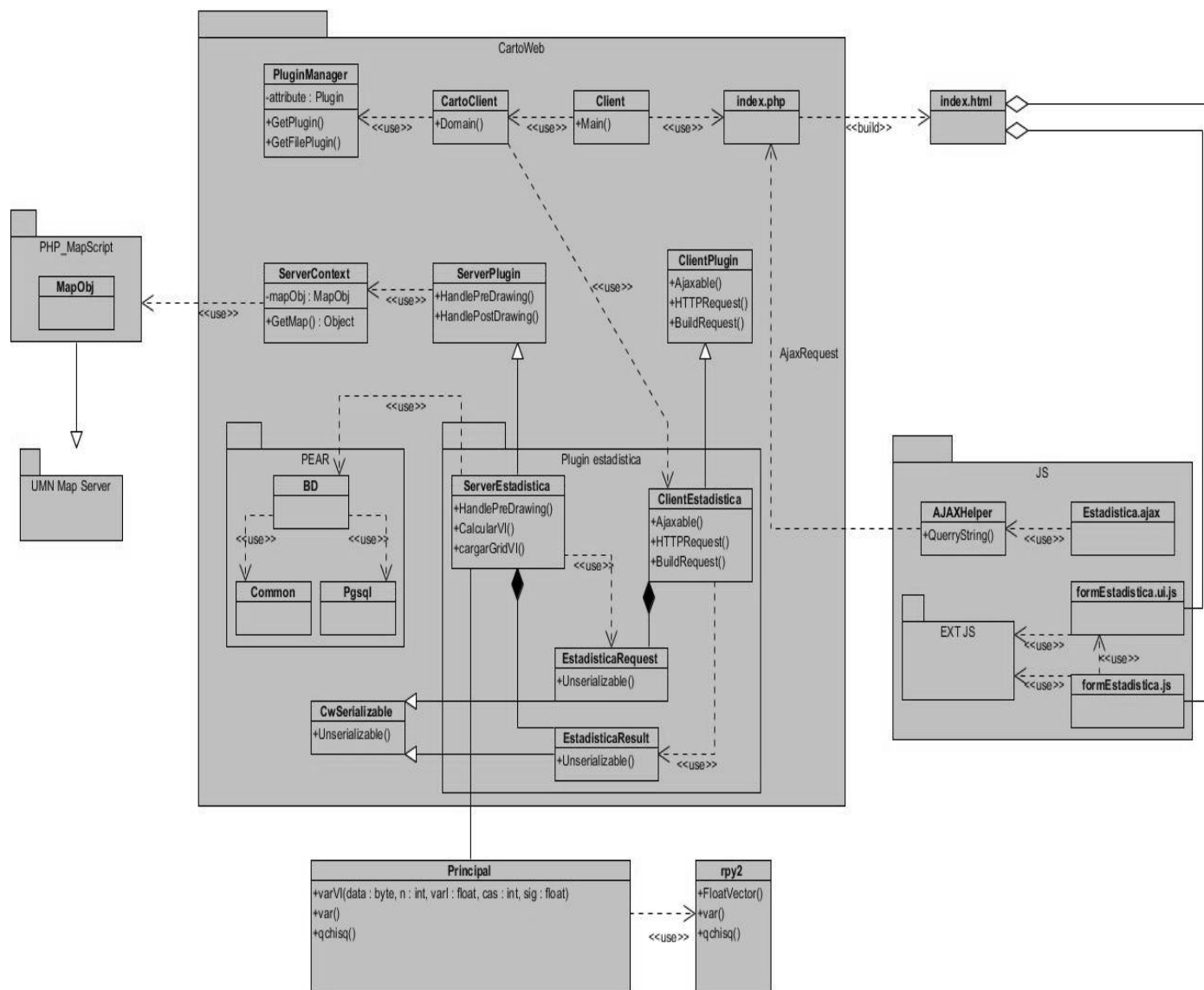


Fig.10: DCD del RF Realizar verificación de hipótesis de varianza

4.4 Modelo de Implementación

El modelo de implementación consiste en el código fuente de los objetos especificados en el modelo de diseño (Jacobson, 2000). En otras palabras, el modelo de implementación describe cómo los elementos del modelo del diseño se implementan en términos de componentes, ya sean ficheros ejecutables, de código fuente u otros. Por tanto, el objetivo principal de dicho modelo es guiar la implementación del sistema.

4.4.1 Diagrama de Componentes

Un diagrama de componentes representa cómo un sistema de software es dividido en componentes y muestra las dependencias entre estos componentes. Los componentes físicos incluyen archivos,

cabeceras, bibliotecas compartidas, módulos, ejecutables, o paquetes. Los diagramas de Componentes prevalecen en el campo de la arquitectura de software, pero pueden ser usados para modelar y documentar cualquier arquitectura de sistema. En la Fig.11 se muestra el Diagrama de Componentes del RF Realizar verificación de hipótesis de varianza.

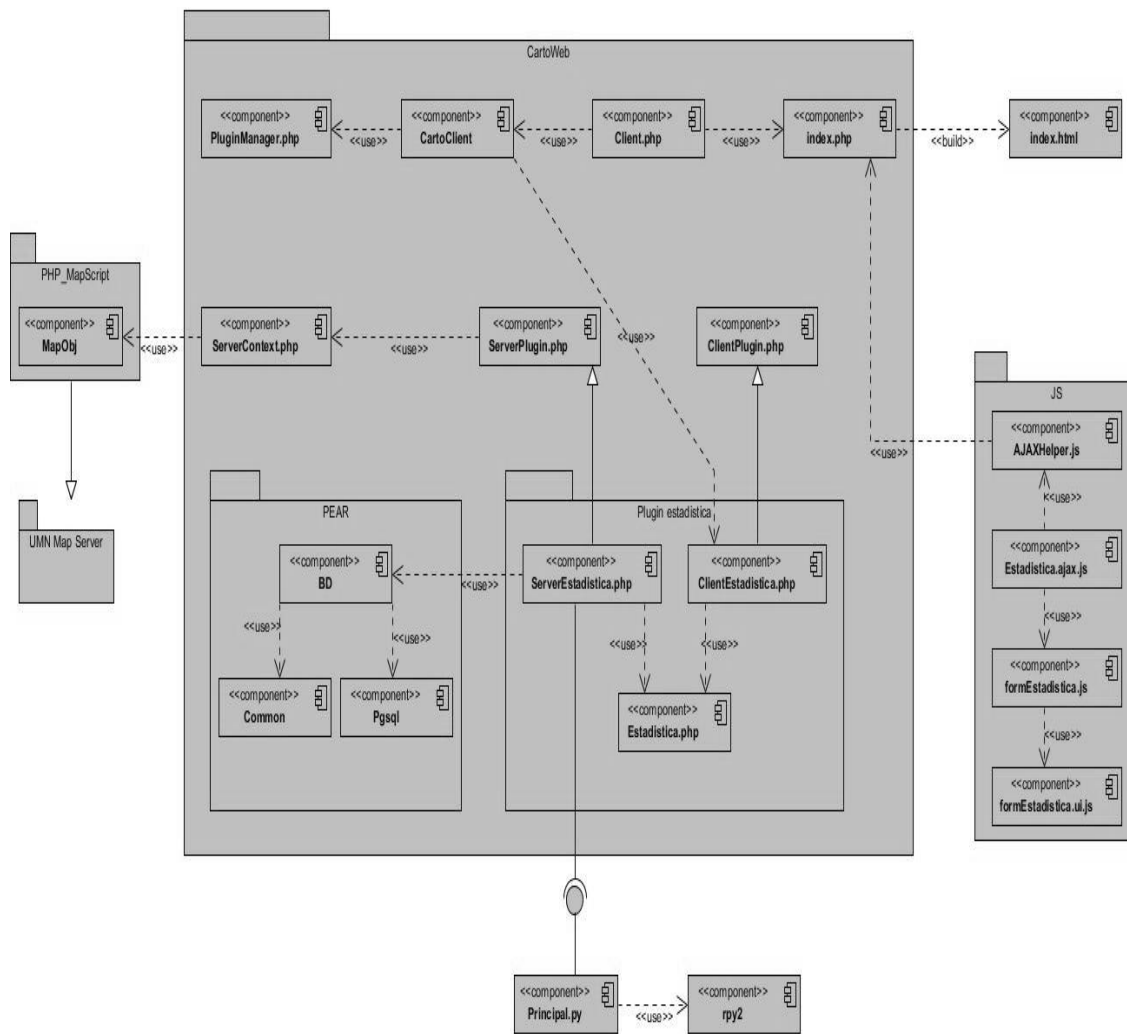


Fig.11: Diagrama de Componentes del RF Realizar verificación de hipótesis de varianza

4.5 Modelo de Despliegue

El modelo de despliegue es un modelo de objetos que describe la distribución física del sistema en términos de cómo se distribuye la funcionalidad entre los nodos de cómputo. El modelo de despliegue se utiliza como entrada fundamental en las actividades de diseño e implementación debido a que la distribución del sistema tiene una influencia principal en su diseño. En la Fig.12 se puede apreciar el diagrama de Despliegue correspondiente a la solución propuesta.

Entre los elementos que suelen componer un diagrama de despliegue, se encuentran:

- Nodos: Elementos de procesamiento con al menos un procesador, memoria, y posiblemente otros dispositivos.
- Dispositivos: Nodos estereotipados sin capacidad de procesamiento en el nivel de abstracción que se modela.
- Conectores: Expresa el tipo de conector o protocolo utilizado entre el resto de los elementos del modelo.

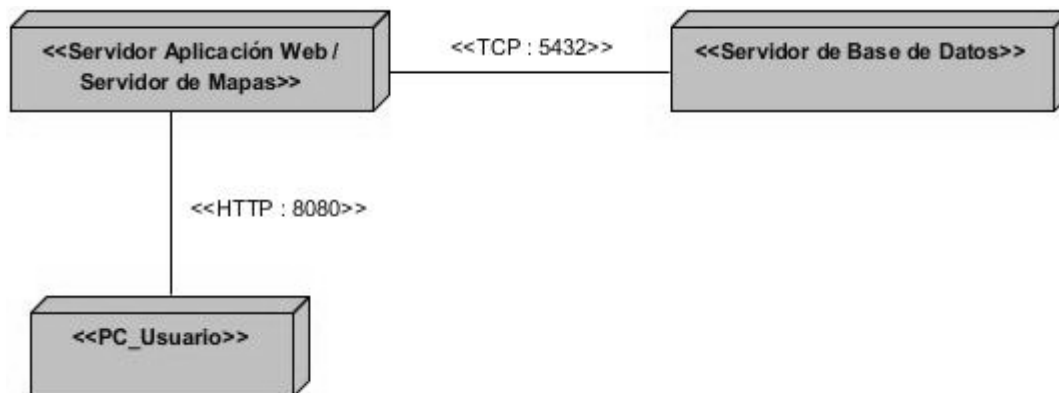


Fig.12: Diagrama de Despliegue

4.6 Proceso de Pruebas

Las pruebas de software son actividades que se desarrollan para detectar errores cometidos sin darse cuenta al realizar su diseño y construcción. Su objetivo es diseñar casos de prueba que, sistemáticamente, saquen a la luz diferentes clases de errores, haciéndolo con la menor cantidad de tiempo y de esfuerzo (Pressman, 2005). La prueba no puede asegurar la ausencia de errores; sólo puede demostrar que existen defectos en el software.

A continuación se definen un conjunto de características que debe poseer una buena prueba (Pressman, 2005):

- Una buena prueba tiene una elevada probabilidad de encontrar un error. Alcanzar este objetivo requiere que la persona que aplica la prueba comprenda el software y trate de desarrollar una imagen mental de la manera en que puede fallar.
- Una buena prueba no es redundante. El tiempo y los recursos destinados a las pruebas son limitados. No hay razón para realizar una prueba que tenga el mismo propósito que otra. Cada prueba debe tener un propósito diferente (aunque las diferencias sean sutiles).
- Una buena prueba debe ser “la mejor de su clase”. En un grupo de pruebas con un objetivo similar y recursos limitados, podría optarse por la ejecución de un solo subconjunto de ellas. En este caso, debe usarse la prueba que tenga la mayor probabilidad de descubrir un tipo completo de errores.

- Una buena prueba no debe ser ni muy simple ni demasiado compleja. Aunque a veces es posible combinar una serie de pruebas en un caso de prueba, los posibles efectos colaterales asociados con este enfoque podrían enmascarar errores. En general, cada prueba debe ejecutarse por separado.

4.6.1 Diseño de Casos de Prueba (DCP)

Un caso de prueba establece una forma de probar el sistema, incluyendo la entrada o resultado con la que se ha de probar y las condiciones bajo las que ha de probarse (Jacobson, y otros, 2000). Sin embargo, este puede estar orientado a encontrar defectos en la interfaz de usuario, en la ejecución de estructuras de datos, o un determinado requisito funcional. En este sentido, un producto de ingeniería puede probarse de dos formas según (Pressman, 2005):

- Conociendo la función específica para la cual fue diseñado el producto, se pueden llevar a cabo pruebas que demuestren que cada función es completamente operativa y al mismo tiempo, buscando errores en cada función.
- Conociendo el funcionamiento interno del producto, se pueden desarrollar pruebas que aseguren que todas las piezas encajan, o sea que la operación interna se ajusta a las especificaciones y que todos los componentes internos se han probado de forma adecuada.

Al primer enfoque de prueba se le denomina prueba de caja negra; al segundo, prueba de caja blanca.

La prueba de caja blanca es un método de diseño que usa la estructura de control descrita como parte del diseño al nivel de componentes para derivar los casos de prueba. Al emplear los métodos de prueba de caja blanca, el ingeniero del software podrá derivar casos de prueba que: garanticen que todas las rutas independientes dentro del módulo se han ejercitado por lo menos una vez, ejerciten los lados verdadero y falso de todas las decisiones lógicas, ejecuten todos los bucles en sus límites y dentro de sus límites operacionales, y ejerciten estructuras de datos internos para asegurar su validez.

Las pruebas de caja negra se concentran en los requisitos funcionales del software. Es decir, permiten al ingeniero de software derivar conjuntos de condiciones de entrada que ejercitarán por completo los requisitos funcionales de un programa. La prueba de caja negra no es una opción frente a las técnicas de caja blanca. Es, en cambio, un enfoque complementario que tiene probabilidades de descubrir una clase diferente de errores de los que se descubrirían con los métodos de caja blanca (Pressman, 2005).

Las pruebas de caja negra tratan de encontrar errores en las siguientes categorías:

- Funciones incorrectas o faltantes.
- Errores de interfaz.
- Errores en estructuras de datos o en acceso a bases de datos externas.
- Errores de comportamiento o desempeño.

➤ Errores de inicialización y término.

A diferencia de las pruebas de caja blanca que se realizan al inicio del proceso de prueba, las de caja negra tienden a aplicarse durante las últimas etapas de la prueba.

Para el proceso de pruebas ha sido seleccionada la técnica de partición equivalente perteneciente a las pruebas de caja negra. Dicha técnica divide el campo de entrada de un programa en clases de datos de los que se pueden derivar casos de prueba para de tal forma ejercitar y probar determinadas funciones del software.

A continuación, se presenta el DCP para el RF Realizar verificación de hipótesis de varianza. El resto de los DCP pueden ser consultados en el Anexo III y en el artefacto “Diseño de casos de pruebas”.

Tabla 3: DCP para el RF Realizar verificación de hipótesis de varianza.

Nombre de la sección	Escenarios de la sección	Capa	Atributo	Método	Caso	Varianza	Significación	Respuesta del Sistema	Flujo Central		
SC1: Realizar verificación de hipótesis de varianza.	EC1.1: Realizar verificación de hipótesis de varianza correctamente.	V	V	V	V	V	V	Muestra una tabla con los resultados de la verificación de hipótesis y los guarda en la base de datos.	El usuario selecciona la opción estadística inferencial para seleccionar la capa y el atributo, luego se selecciona la opción de Verificación de Hip. de DP y se selecciona el método Varianza, posteriormente se selecciona el caso, y se introducen los valores de varianza y significación. Por última se le da al botón Calcular.		
		empresa	cant_industriales	Varianza	1	451.2	0.01				
	EC1.2: Campos Obligatorios vacíos.		I	V	V	V	V	V	El sistema no habilita el siguiente paso.	El usuario selecciona la opción estadística inferencial.	
				N/A	N/A	N/A	N/A	N/A		N/A	El usuario selecciona la opción estadística inferencial e introduce la capa.
			V	I	V	V	V	V			
			empresa		N/A	N/A	N/A	N/A			
			V	V	I	V	V	V			El usuario selecciona la opción estadística inferencial e introduce la capa y el atributo.
			empresa	cant_industrial							

		V	V	V	V	I	V	El sistema notifica que existen campos vacíos.	El usuario selecciona la opción estadística inferencial, selecciona capa, atributo, la opción de Verificación de Hip. de DP y se selecciona el método Varianza, luego selecciona el caso e introduce valores dejando la varianza o la significación vacíos. Luego le da al botón Calcular.
		empresa	cant_industriales	Varianza	2		0.05		
		V	V	V	V	V	I		
		empresa	cant_industriales	Varianza	1	72.4			
EC1.3: Valor de la significación fuera de rango.		V	V	V	V	V	I	El sistema notifica que el valor de la significación está entre 0 y 1.	El usuario realiza toda la secuencia de pasos para ejecutar la funcionalidad de forma correcta, pero introduce un valor de significación no válido.
		empresa	cant_industriales	Varianza	3	94.14	1.02		
EC1.4: El atributo no corresponde a la capa.		V	I	V	V	V	V	El sistema notifica que el atributo no corresponde a la capa.	El usuario le da al botón Calcular habiendo seleccionado un atributo que no corresponde a la capa.
		empresa	cant_estanques	Varianza	2	561.2	0.01		
EC1.5: Cerrar Ventana.		V	V	V	V	V	V	El sistema oculta la ventana.	El usuario selecciona la opción estadística inferencial y luego da clic en cerrar.
		N/A	N/A	N/A	N/A	N/A	N/A		

4.6.2 Pruebas de Rendimiento

Las pruebas de rendimiento son pruebas que se realizan para determinar cuán rápido realiza una tarea un sistema en condiciones particulares de trabajo (Molyneaux, 2009). Aplicar este tipo de pruebas puede permitir también validar y verificar otros atributos de la calidad del sistema, tales como la escalabilidad, fiabilidad y uso de los recursos. Entre los diversos tipos de pruebas de rendimiento existentes serán aplicadas a la solución las pruebas de carga y estrés como parte del proceso de pruebas.

Las pruebas de carga son aplicadas para observar el comportamiento de una solución informática bajo una cantidad de peticiones esperada. Este tipo de prueba es la más sencilla de las pruebas de rendimiento, y su carga puede ser el número de usuarios concurrentes utilizando la aplicación y que realizan cierta cantidad de transacciones durante el tiempo que dura la carga.

Las pruebas de estrés se realizan para determinar la solidez de la aplicación en momentos de carga extrema y ayuda a los administradores a determinar si la aplicación rendirá lo suficiente en caso sobrecarga. Son utilizadas con el objetivo de observar el punto de ruptura del sistema, o determinar el número esperado de usuarios en concurrencia para hacer una simulación lo más cercana posible a la realidad. Estas pruebas permiten tomar decisiones sobre configuraciones de hardware, ajustes de software y selección de arquitecturas.

4.6.3 Ejecución de las Pruebas

A partir del diseño de los Casos de Prueba desarrollados para los RF se procede a la ejecución de los mismos, dónde se desarrollaron tres iteraciones de acuerdo a como se muestra en la Fig. 13. En la primera iteración se obtuvieron 5 no conformidades, en la segunda iteración se corrigen las no conformidades encontradas anteriormente y surgen 2 nuevas no conformidades las cuales fueron corregidas en la tercera iteración, donde no se obtuvieron nuevas no conformidades.

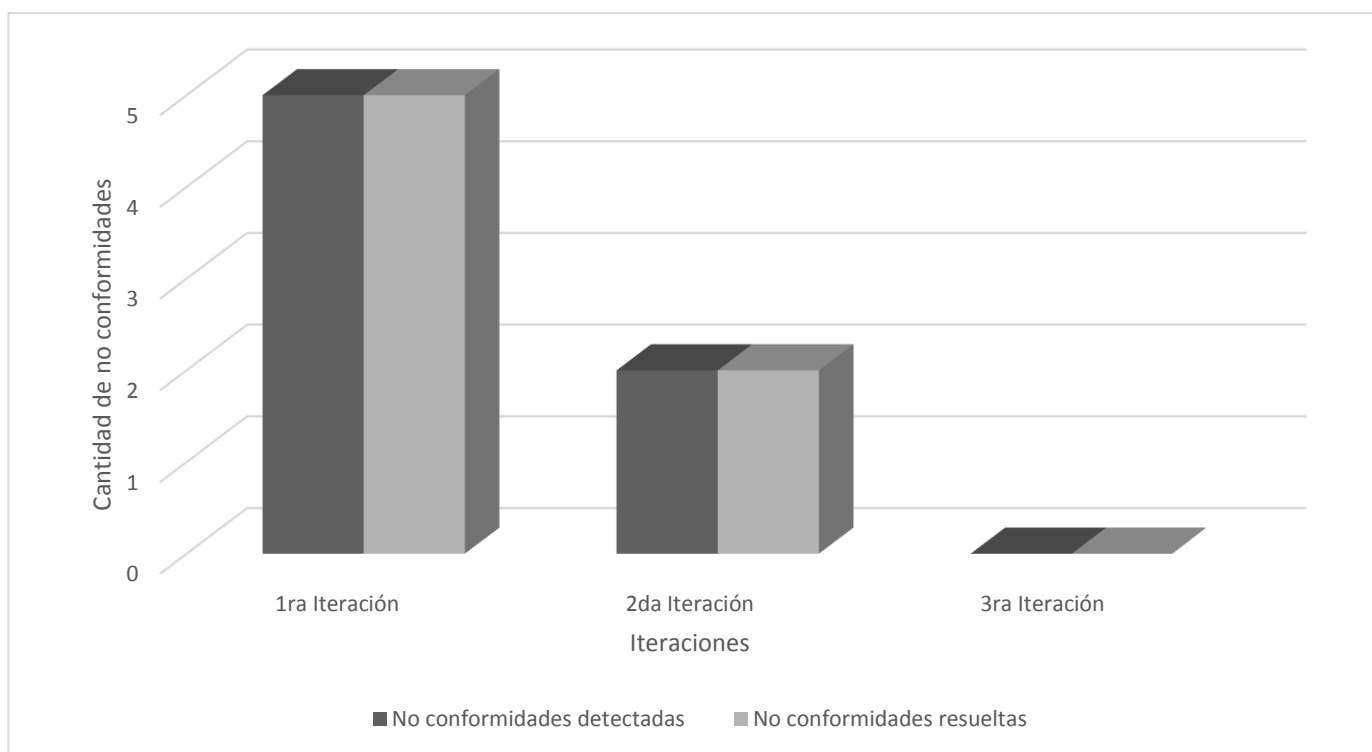


Fig. 13: Resultados de aplicar las pruebas de caja negra.

Al ser efectuadas las pruebas de carga y estrés para comprobar el rendimiento de la plataforma GeneSIG antes de contener el componente desarrollado, y una vez contenido el componente; se obtienen los resultados que demuestran las diferencias poco significativas entre ambas salidas, por tanto, es válido expresar que la incorporación del Componente de Estadística Inferencial a la plataforma GeneSIG no

influye de forma negativa en este último. Dichas pruebas de rendimiento fueron desarrolladas con la herramienta JMeter 2.8.

Tabla 4: Resultados de las pruebas de rendimiento.

Acción	Cantidad de hilos (usuarios)	Rendimiento (peticiones / segundos)
Ejecución de GeneSIG	25	162,8 / seg
Ejecución de GeneSIG con Componente de Estadística Inferencial	25	152,3 / seg
Ejecución de GeneSIG	50	151,7 / seg
Ejecución de GeneSIG con Componente de Estadística Inferencial	50	137,4 / seg

4.7 Conclusiones Parciales

El desarrollo del Diagrama de Clases del Diseño permitió observar en detalle cómo están estructurados los datos; y al implementar estos en forma de componente se pudo obtener el Diagrama de Componentes, donde se observa cómo queda definido el modelo de Implementación con sus archivos de código fuente. Por su parte, el Diagrama de Despliegue evidenció la relación existente entre el software y el hardware. Además, mediante el diseño de los casos de prueba fue posible detectar y corregir los errores de la etapa de implementación, para de tal forma obtener una solución final con un correcto funcionamiento. El empleo de las pruebas de rendimiento permitió determinar que al incluir el Componente de Estadística Inferencial en la plataforma GeneSIG no se afecta de forma significativa el rendimiento de dicha plataforma.

Conclusiones

Luego de culminada la investigación se puede afirmar que ha sido cumplido el objetivo general planteado, por tanto, se puede afirmar que:

- Mediante el análisis de las soluciones existentes se determina que la mejor opción es desarrollar el Componente de Estadística Inferencial para la plataforma GeneSIG, debido a que de otra forma representaría un gasto de tiempo y esfuerzo aún mayor. Aunque se tuvieron en cuenta diversos aspectos de los sistemas analizados para el desarrollo de la solución propuesta, tal como el empleo del motor estadístico R.
- Para el análisis, diseño y construcción de la solución propuesta fueron utilizadas las tecnologías y herramientas definidas por la LPS Aplicativos SIG, de tal forma que no existieran problemas para su incorporación a GeneSIG.
- La aplicación desarrollada alcanza cierto nivel de uniformidad, robustez y flexibilidad; debido a la utilización de un modelo arquitectónico, y el empleo de patrones, que permiten poner o quitar de forma sencilla el componente desarrollado en la plataforma GeneSIG, además de poseer la misma estructura que los *plugins* de la mencionada plataforma.
- Mediante el empleo de las pruebas de caja negra es posible detectar diversos errores, para así dar cumplimiento a los requisitos funcionales establecidos y obtener un producto con la calidad requerida. En cuanto a las pruebas de rendimiento permiten establecer el hecho de que la incorporación de la solución desarrollada a la plataforma GeneSIG, no afectará el rendimiento de este último.
- La incorporación del Componente de Estadística Inferencial a la plataforma GeneSIG, aporta cierto valor agregado a ésta e incrementa sus posibilidades de comercialización e introducción en nuevos mercados.

Recomendaciones

- Implementar el resto de los métodos estadísticos que forman parte de la Estadística Inferencial, tales como, regresión lineal y análisis de clúster.
- Garantizar la seguridad de la información que se maneja en el componente, protegiéndola del acceso no autorizado.

Bibliografía

- Association, GvSIG. 2011.** *GvSIG. Manual de Usuario.* 2011.
- Athan, Tara, y otros. 2010.** *Quantum GIS.* 2010.
- Carballé, Adaily Hernández, Guerra, Mayelin Timoteo y González, Ayeris Lizandra Navarro. 2013.** *EXTENSIÓN DE LA HERRAMIENTA “VISUAL PARADIGM FOR UML” PARA LA APLICACIÓN DEL MÉTODO DE ESTIMACIÓN UCI*. 2013.
- Caso Osorio, Edson E. 2011.** *Manual ArcGIS. Básico.* 2011.
- Denzer, Patricio. 2002.** *PosgreSQL.* 2002.
- Freund, John E., Miller, Irwin R. y Jonhson, Richard. 2006.** *Probabilidad y Estadística para ingenieros.* 2006.
- García Mancilla, H. 2003.** *Estadística Descriptiva e Inferencial (I).* 2003.
- Garsón Pérez, Teresa. 2010.** *Sistemas Gestores de Bases de Datos.* 2010.
- González Duque, Raúl. 2012.** *Python para todos.* 2012.
- Hernández Oralo, Enrique. 2003.** *El Lenguaje Unificado de Modelado (UML).* 2003.
- INTENCO. 2009.** *Ingeniería del Software: metodologías y ciclos de vida.* 2009.
- Jacobson, I. y Booch, G. 2000.** *El Proceso Unificado de Desarrollo de Software.* 2000.
- Jacobson, Ivar. 2000.** *Ingeniería de Software orientado a objetos.* 2000.
- Jeffrey Star, J. 1990.** *Geographic Information Systems.* 1990.
- Membrides Espinosa, Antonio. 2010.** *Fundamentos del Mapserver, Mapscript, PostGIS y su Integración con el Cartoweb.* 2010.
- Membrides Espinosa, Antonio. 2013.** *Manual de GeneSIG.* 2013.
- Membrides Espinosa, Antonio, y otros. 2012.** *Manual de GeneSIG.* 2012.
- Molyneaux, I. 2009.** *The Art of Application Performance Testing.* 2009.
- Pressman. 2005.** *Ingeniería de Software, un enfoque práctico.* 2005.
- Ramsey, Paul. 2011.** *PostGIS Manual.* 2011.
- Reingart, Mariano. 2011.** *Tutorial PgAdmin III.* 2011.
- Reyna. 2005.** *El uso de los sistemas de información geográfica (SIG) en el análisis demográfico de situaciones de desastre.* 2005.
- Reynoso, Carlos y Kicillof, Nicolás. 2004.** *Estilos y Patrones en la Estrategia de Arquitectura de Microsoft.* Buenos Aires : s.n., 2004.
- Rubio Liniers, María. 1998.** *EL ANÁLISIS DOCUMENTAL: INDIZACIÓN Y RESUMEN EN BASES DE DATOS ESPECIALIZADAS.* 1998.
- Sánchez Asenjo, Jorge. 2012.** *Introducción a PHP.* 2012.

- Santana, Julio Sergio y Mateos Farfán, Efraín. 2014.** *El arte de programar en R: un lenguaje para la estadística.* 2014.
- Sironvalle. 2000.** *Estadística.* 2000.
- Sommerville, Ian. 2005.** *Ingeniería del Software.* 2005.
- Storti, Guillermo, Ríos, Gladys y Campodónico, Gabriel. 2007.** *Base de Datos. Modelo Entidad - Relación.* 2007.
- Tovar, Manuel. 2003.** *Sistemas de Gestión de Bases de Datos.* 2003.
- Tusell, Fernando. 2011.** *Análisis de Regresión.* 2011.
- UCID. 2012.** *PRODESOF. Proceso de Desarrollo y Gestión de Proyectos de Software (Versión 1.5).* 2012.
- Ureña Almagro, Carlos. 2012.** *Lenguajes de Programación. Lenguajes de Programación.* 2012.
- Uriel, Ezequiel y Manzano, J. A. 2002.** *Análisis multivariante aplicado.* 2002.
- Vázquez Mariño, Carlos. 2008.** *Programación en PHP5. Nivel básico.* 2008.
- Zayas. 1997.** *Metodología de la Investigación Científica.* 1997.

Anexo I: Especificaciones de RF

Tabla 5: Especificación del RF Realizar estimación puntual de varianza

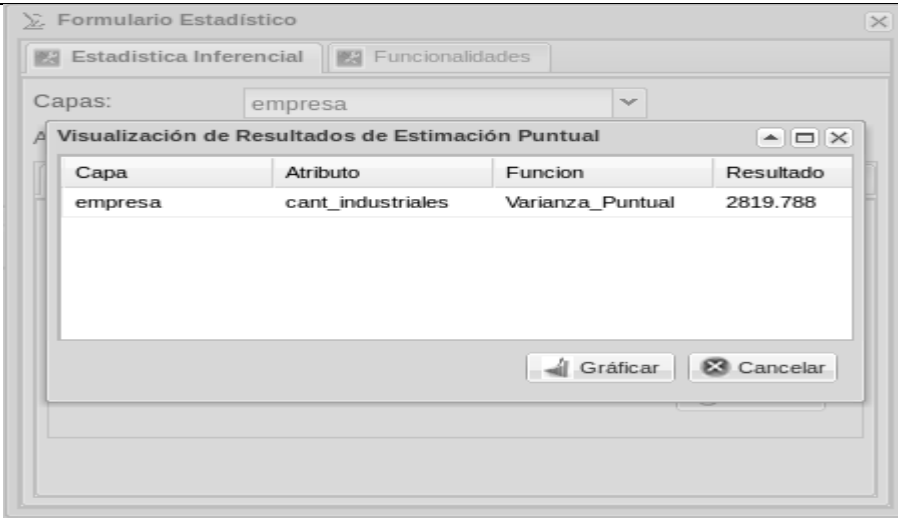
Conceptos tratados	Conceptos	Atributos
	-	-
Precondiciones	Pre-condiciones	Pre-requisitos
	Abrir ventana "Formulario Estadístico"	No procede
Descripción	<ol style="list-style-type: none"> 1. Se selecciona la capa a la cual se aplicará el estimador puntual de varianza. 2. Se selecciona el atributo al cual se aplicará el estimador puntual de varianza. 3. Se selecciona el panel Estimación. 4. Se selecciona como método de estimación el Puntual. 5. Se selecciona la Varianza como estimador Puntual y se da clic al botón Calcular. 6. Se muestra una nueva ventana que contiene Capa, Atributo, Función y Resultado. 	
Validaciones	No se habilita el campo del atributo hasta que no sea seleccionada la capa. No se habilita el panel Estimación hasta que no sea seleccionado el atributo.	
Post-condiciones	Será registrado el valor de la estimación puntual de varianza en la base de datos.	
Post-requisitos	Mostrar el resultado de los cálculos en una gráfica, Mostrar el resultado de los cálculos en el mapa.	
Prototipo de Interfaz		

Tabla 6: Especificación del RF Realizar estimación puntual de media

Conceptos tratados	Conceptos	Atributos
	-	-
Precondiciones	Pre-condiciones	Pre-requisitos

COMPONENTE DE ESTADÍSTICA INFERENCIAL PARA LA PLATAFORMA GENESISG

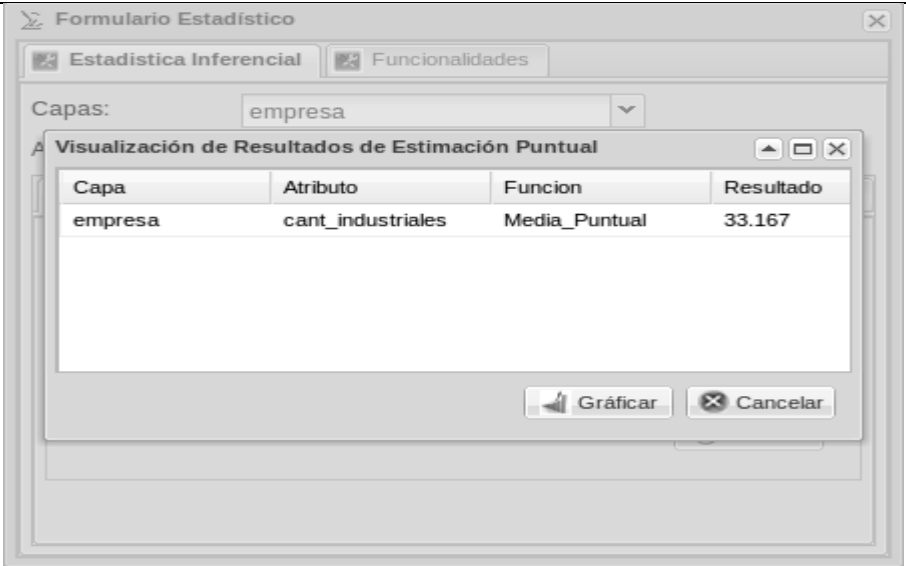
	Abrir ventana "Formulario Estadístico"	No procede
Descripción	<ol style="list-style-type: none"> 1. Se selecciona la capa a la cual se aplicará el estimador puntual de media. 2. Se selecciona el atributo al cual se aplicará el estimador puntual de media. 3. Se selecciona el panel Estimación. 4. Se selecciona como método de estimación el Puntual. 5. Se selecciona la Media como estimador Puntual y se da clic al botón Calcular. 6. Se muestra una nueva ventana que contiene Capa, Atributo, Función y Resultado. 	
Validaciones	No se habilita el campo del atributo hasta que no sea seleccionada la capa. No se habilita el panel Estimación hasta que no sea seleccionado el atributo.	
Post-condiciones	Será registrado el valor de la estimación puntual de la media en la base de datos.	
Post-requisitos	Mostrar el resultado de los cálculos en una gráfica, Mostrar el resultado de los cálculos en el mapa.	
Prototipo de Interfaz		

Tabla 7: Especificación del RF Realizar estimación puntual de desviación típica

Conceptos tratados	Conceptos	Atributos
	-	-
Precondiciones	Pre-condiciones	Pre-requisitos
	Abrir ventana "Formulario Estadístico"	No procede
Descripción	<ol style="list-style-type: none"> 1. Se selecciona la capa a la cual se aplicará el estimador puntual de la desviación típica. 2. Se selecciona el atributo al cual se aplicará el estimador puntual de la desviación 	

COMPONENTE DE ESTADÍSTICA INFERENCIAL PARA LA PLATAFORMA GENESISG

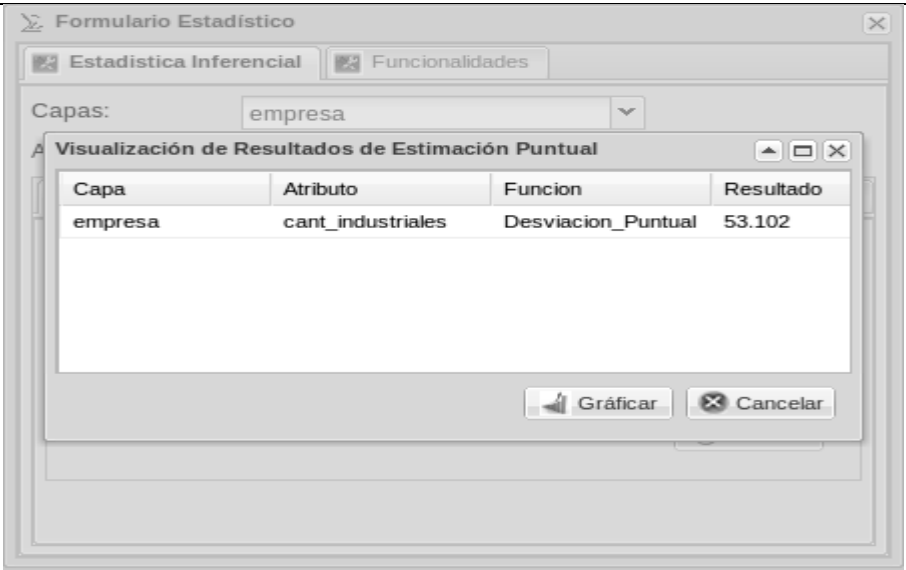
	<p>típica.</p> <ol style="list-style-type: none"> 3. Se selecciona el panel Estimación. 4. Se selecciona como método de estimación el Puntual. 5. Se selecciona la Desviación como estimador Puntual y se da clic al botón Calcular. 6. Se muestra una nueva ventana que contiene Capa, Atributo, Función y Resultado. 								
Validaciones	No se habilita el campo del atributo hasta que no sea seleccionada la capa. No se habilita el panel Estimación hasta que no sea seleccionado el atributo.								
Post-condiciones	Será registrado el valor de la estimación puntual de la desviación típica en la base de datos.								
Post-requisitos	Mostrar el resultado de los cálculos en una gráfica, Mostrar el resultado de los cálculos en el mapa.								
Prototipo de Interfaz	 <p>The screenshot shows a software window titled 'Formulario Estadístico'. Inside, there are tabs for 'Estadística Inferencial' and 'Funcionalidades'. A dropdown menu for 'Capas' is set to 'empresa'. A sub-window titled 'Visualización de Resultados de Estimación Puntual' is open, displaying a table with the following data:</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>Capa</th> <th>Atributo</th> <th>Funcion</th> <th>Resultado</th> </tr> </thead> <tbody> <tr> <td>empresa</td> <td>cant_industriales</td> <td>Desviacion_Puntual</td> <td>53.102</td> </tr> </tbody> </table> <p>At the bottom of the sub-window, there are two buttons: 'Gráficar' and 'Cancelar'.</p>	Capa	Atributo	Funcion	Resultado	empresa	cant_industriales	Desviacion_Puntual	53.102
Capa	Atributo	Funcion	Resultado						
empresa	cant_industriales	Desviacion_Puntual	53.102						

Tabla 8: Especificación del RF Realizar estimación por intervalos de varianza

Conceptos tratados	Conceptos	Atributos
	-	-
Precondiciones	Pre-condiciones	Pre-requisitos
	Abrir ventana "Formulario Estadístico"	No procede
Descripción	<ol style="list-style-type: none"> 1. Se selecciona la capa a la cual se aplicará el estimador por intervalos de la varianza. 2. Se selecciona el atributo al cual se aplicará el estimador por intervalos de la varianza. 	

COMPONENTE DE ESTADÍSTICA INFERENCIAL PARA LA PLATAFORMA GENESISG

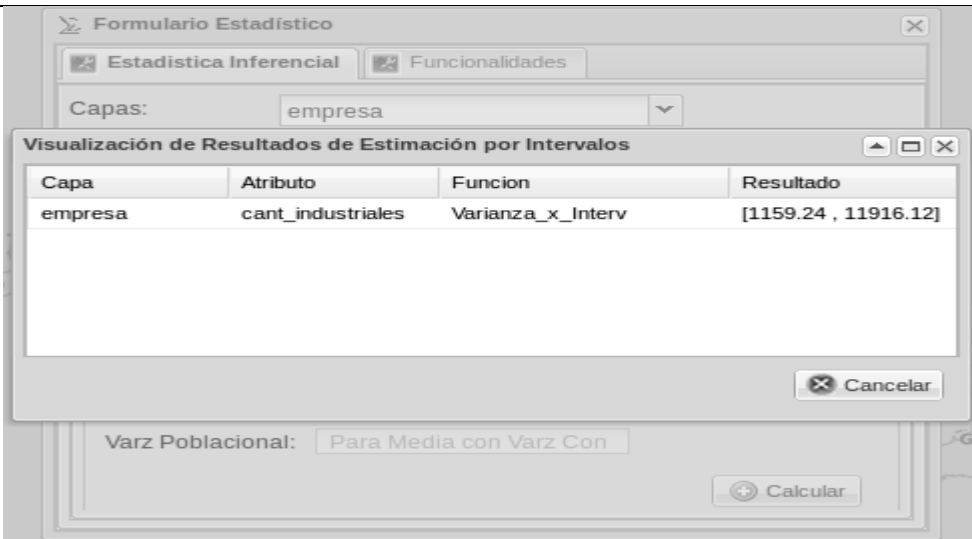
	<ol style="list-style-type: none"> 3. Se selecciona el panel Estimación. 4. Se selecciona como método de estimación el Intervalo. 5. Se selecciona la Varianza como estimador por Intervalo, se introduce el valor de la significación y se da clic al botón Calcular. 6. Se muestra una nueva ventana que contiene Capa, Atributo, Función y Resultado.
Validaciones	No se habilita el campo del atributo hasta que no sea seleccionada la capa. No se habilita el panel Estimación hasta que no sea seleccionado el atributo. El valor de la significación es numérico entre 0 y 1.
Post-condiciones	Será registrado el valor de la estimación por intervalos de la varianza en la base de datos.
Post-requisitos	No Procede
Prototipo de Interfaz	

Tabla 9: Especificación del RF Realizar estimación por intervalos de desviación típica

Conceptos tratados	Conceptos	Atributos
	-	-
Precondiciones	Pre-condiciones	Pre-requisitos
	Abrir ventana "Formulario Estadístico"	No procede
Descripción	<ol style="list-style-type: none"> 1. Se selecciona la capa a la cual se aplicará el estimador por intervalos de la desviación típica. 2. Se selecciona el atributo al cual se aplicará el estimador por intervalos de la desviación típica. 3. Se selecciona el panel Estimación. 4. Se selecciona como método de estimación el Intervalo. 	

COMPONENTE DE ESTADÍSTICA INFERENCIAL PARA LA PLATAFORMA GENESISG


	<p>5. Se selecciona la Desviación como estimador por Intervalo, se introduce el valor de la significación y se da clic al botón Calcular.</p> <p>6. Se muestra una nueva ventana que contiene Capa, Atributo, Función y Resultado.</p>
Validaciones	No se habilita el campo del atributo hasta que no sea seleccionada la capa. No se habilita el panel Estimación hasta que no sea seleccionado el atributo. El valor de la significación es numérico entre 0 y 1.
Post-condiciones	Será registrado el valor de la estimación por intervalos de la desviación típica en la base de datos.
Post-requisitos	No Procede
Prototipo de Interfaz	

Tabla 10: Especificación del RF Realizar estimación bayesiana

Conceptos tratados	Conceptos	Atributos
	-	-
Precondiciones	Pre-condiciones	Pre-requisitos
	Abrir ventana "Formulario Estadístico"	No procede
Descripción	<ol style="list-style-type: none"> 1. Se selecciona la capa a la cual se aplicará la estimación bayesiana de la media. 2. Se selecciona el atributo al cual se aplicará la estimación bayesiana de la media. 3. Se selecciona el panel Estimación. 4. Se selecciona como método de estimación la Bayesiana. 5. Se introduce el valor de la media a priori y de la desviación típica a priori, y se da clic al botón Calcular. 	

COMPONENTE DE ESTADÍSTICA INFERENCIAL PARA LA PLATAFORMA GENESISG


	6. Se muestra una nueva ventana que contiene Capa, Atributo, Función, Media a Posteriori y Desviación a Posteriori.
Validaciones	No se habilita el campo del atributo hasta que no sea seleccionada la capa. No se habilita el panel Estimación hasta que no sea seleccionado el atributo. El valor de la media a priori y de la desviación típica a priori es numérico.
Post-condiciones	Será registrado el valor de la estimación bayesiana de la media en la base de datos.
Post-requisitos	No Procede
Prototipo de Interfaz	

Tabla 11: Especificación del RF Realizar verificación de hipótesis de media con varianza conocida

Conceptos tratados	Conceptos	Atributos
	-	-
Precondiciones	Pre-condiciones	Pre-requisitos
	Abrir ventana "Formulario Estadístico"	No procede
Descripción	<ol style="list-style-type: none"> 1. Se selecciona la capa a la cual se aplicará la verificación de hipótesis de media con varianza conocida. 2. Se selecciona el atributo al cual se aplicará la verificación de hipótesis de media con varianza conocida. 3. Se selecciona el panel Verificación de Hipótesis. 4. Se selecciona como método de verificación de hipótesis la Media con varianza conocida. 5. Se selecciona el caso de hipótesis, introduce el valor de la significación y la varianza poblacional, y se da clic al botón Calcular. 6. Se muestra una nueva ventana que contiene Capa, Atributo, Función, Estimador, 	

COMPONENTE DE ESTADÍSTICA INFERENCIAL PARA LA PLATAFORMA GENESISG

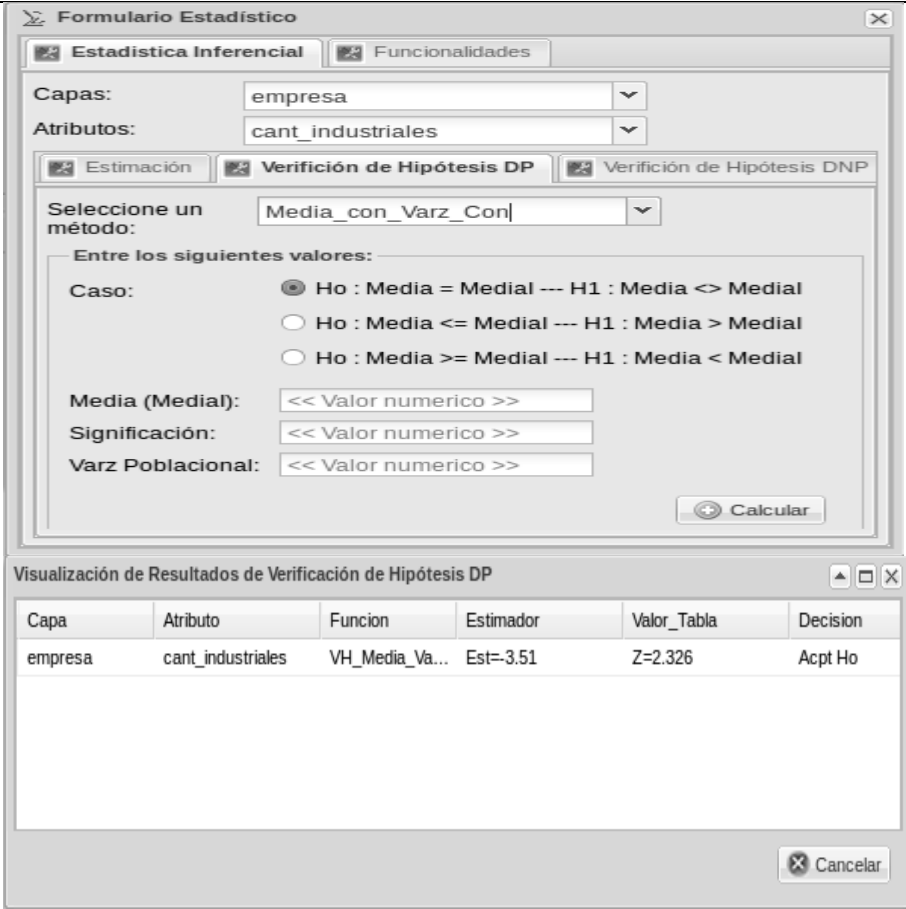
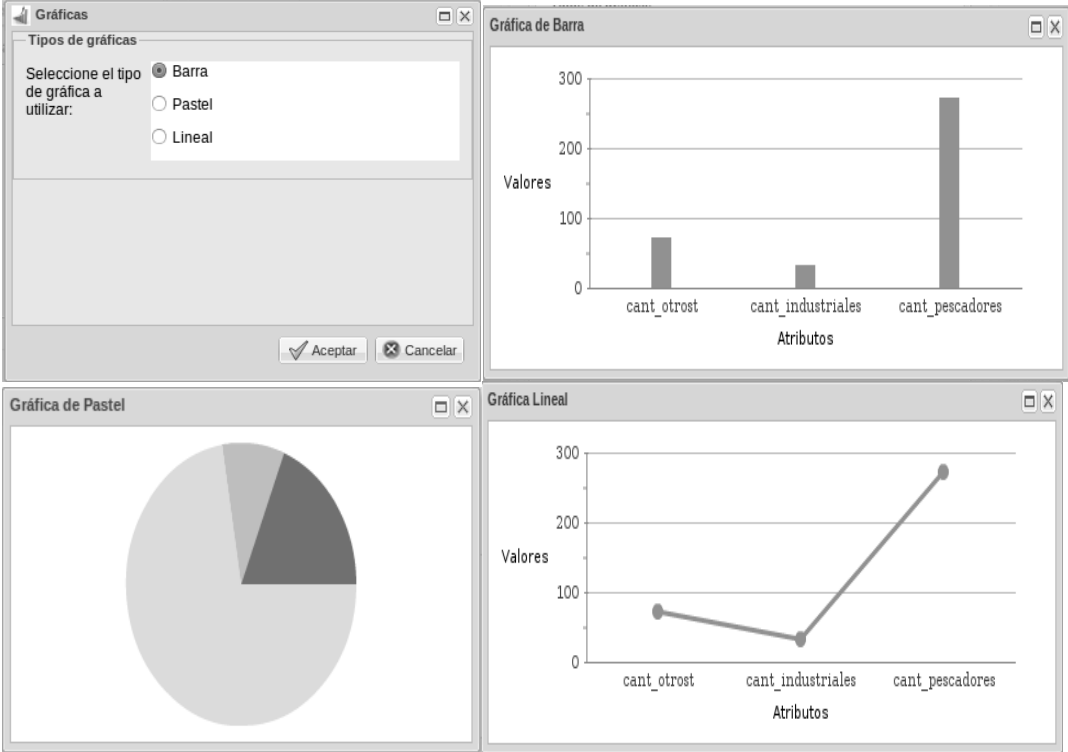
	Valor Tabla y Decisión.												
Validaciones	No se habilita el campo del atributo hasta que no sea seleccionada la capa. No se habilita el panel Verificación de Hipótesis hasta que no sea seleccionado el atributo. El valor de la significación es numérico entre 0 y 1. El valor de la varianza poblacional es numérico.												
Post-condiciones	Será registrado el resultado de la verificación de hipótesis de media con varianza conocida en la base de datos.												
Post-requisitos	No Procede												
Prototipo de Interfaz	 <table border="1" style="margin-top: 10px;"> <caption>Visualización de Resultados de Verificación de Hipótesis DP</caption> <thead> <tr> <th>Capa</th> <th>Atributo</th> <th>Funcion</th> <th>Estimador</th> <th>Valor_Tabla</th> <th>Decision</th> </tr> </thead> <tbody> <tr> <td>empresa</td> <td>cant_industriales</td> <td>VH_Media_Va...</td> <td>Est=-3.51</td> <td>Z=2.326</td> <td>Acpt Ho</td> </tr> </tbody> </table>	Capa	Atributo	Funcion	Estimador	Valor_Tabla	Decision	empresa	cant_industriales	VH_Media_Va...	Est=-3.51	Z=2.326	Acpt Ho
Capa	Atributo	Funcion	Estimador	Valor_Tabla	Decision								
empresa	cant_industriales	VH_Media_Va...	Est=-3.51	Z=2.326	Acpt Ho								

Tabla 12: Especificación del RF Mostrar el resultado de los cálculos en una gráfica

Conceptos tratados	Conceptos	Atributos
	-	-
Precondiciones	Pre-condiciones	Pre-requisitos
	Haber ejecutado algún método de estimación puntual.	RF1 Realizar estimación puntual de varianza, RF2 Realizar estimación puntual de desviación, RF3 Realizar estimación puntual de media.
Descripción	1. Se selecciona el botón Gráfica.	

COMPONENTE DE ESTADÍSTICA INFERENCIAL PARA LA PLATAFORMA GENESIG

	<p>2. Se selecciona el tipo de gráfica, ya sea lineal, de barras o pastel, y se da clic en el botón Aceptar.</p> <p>3. Se muestra la gráfica según el tipo seleccionado.</p>
Validaciones	No procede
Post-condiciones	No procede
Post-requisitos	No Procede
Prototipo de Interfaz	 <p>El prototipo de interfaz muestra un menú de selección de tipos de gráficos con tres opciones: Barra (seleccionada), Pastel y Lineal. Debajo del menú se encuentran los botones 'Aceptar' y 'Cancelar'. A la derecha se muestran tres ventanas de visualización de gráficos:</p> <ul style="list-style-type: none"> Gráfica de Barra: Muestra tres barras representando los atributos 'cant_otrost', 'cant_industriales' y 'cant_pescadores'. El eje vertical 'Valores' va de 0 a 300. Gráfica de Pastel: Muestra un gráfico de pastel con tres sectores de diferentes tamaños y tonos de gris. Gráfica Lineal: Muestra una línea con tres puntos conectados por segmentos rectos, representando los valores de los atributos. El eje vertical 'Valores' va de 0 a 300.

Anexo II: Diagrama de Clases del Diseño

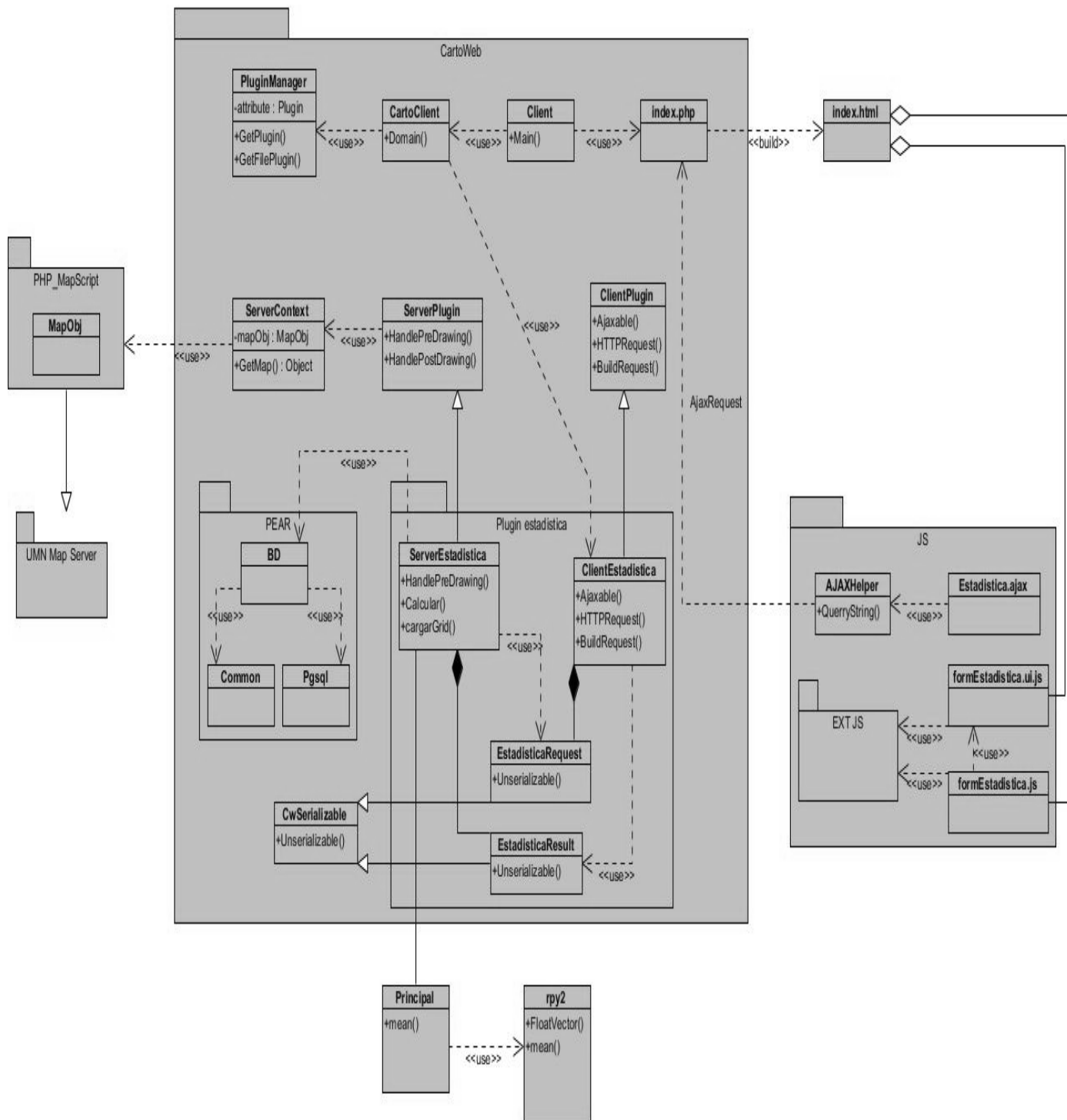


Fig.14: DCD del RF Realizar estimación puntual de media

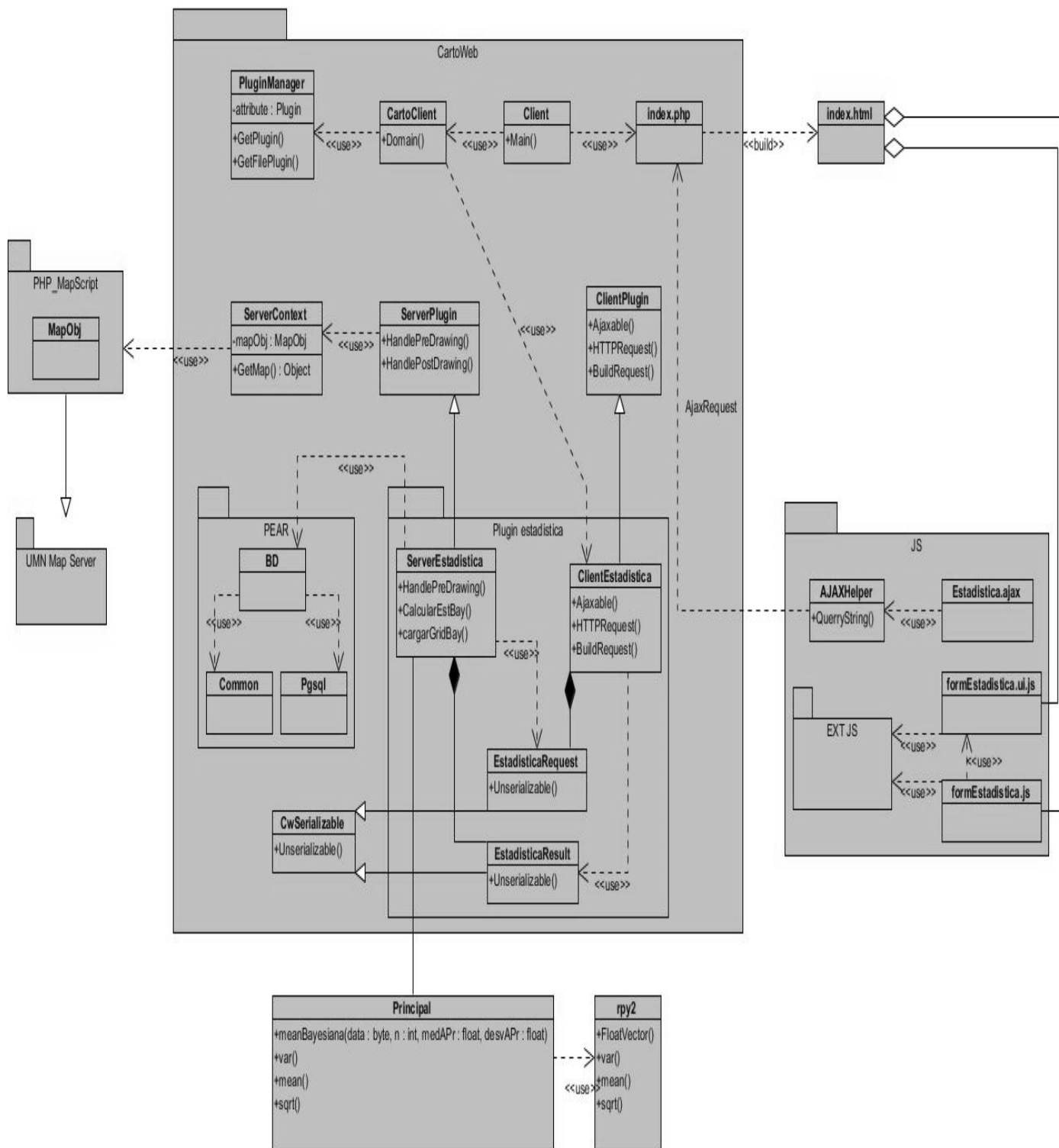


Fig.15: DCD del RF Realizar estimación bayesiana

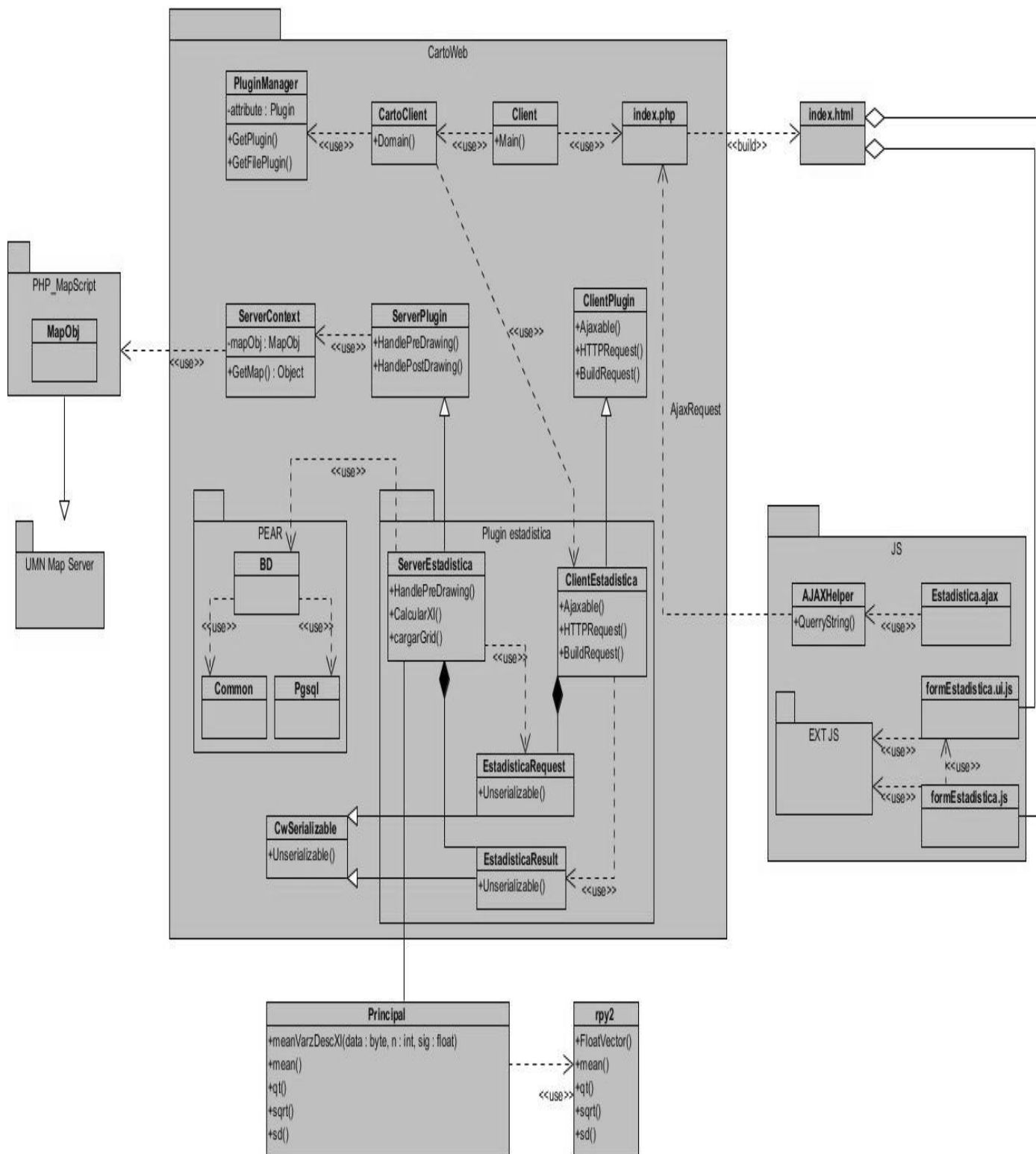


Fig.16: DCD del RF Realizar estimación por intervalos de media con varianza desconocida

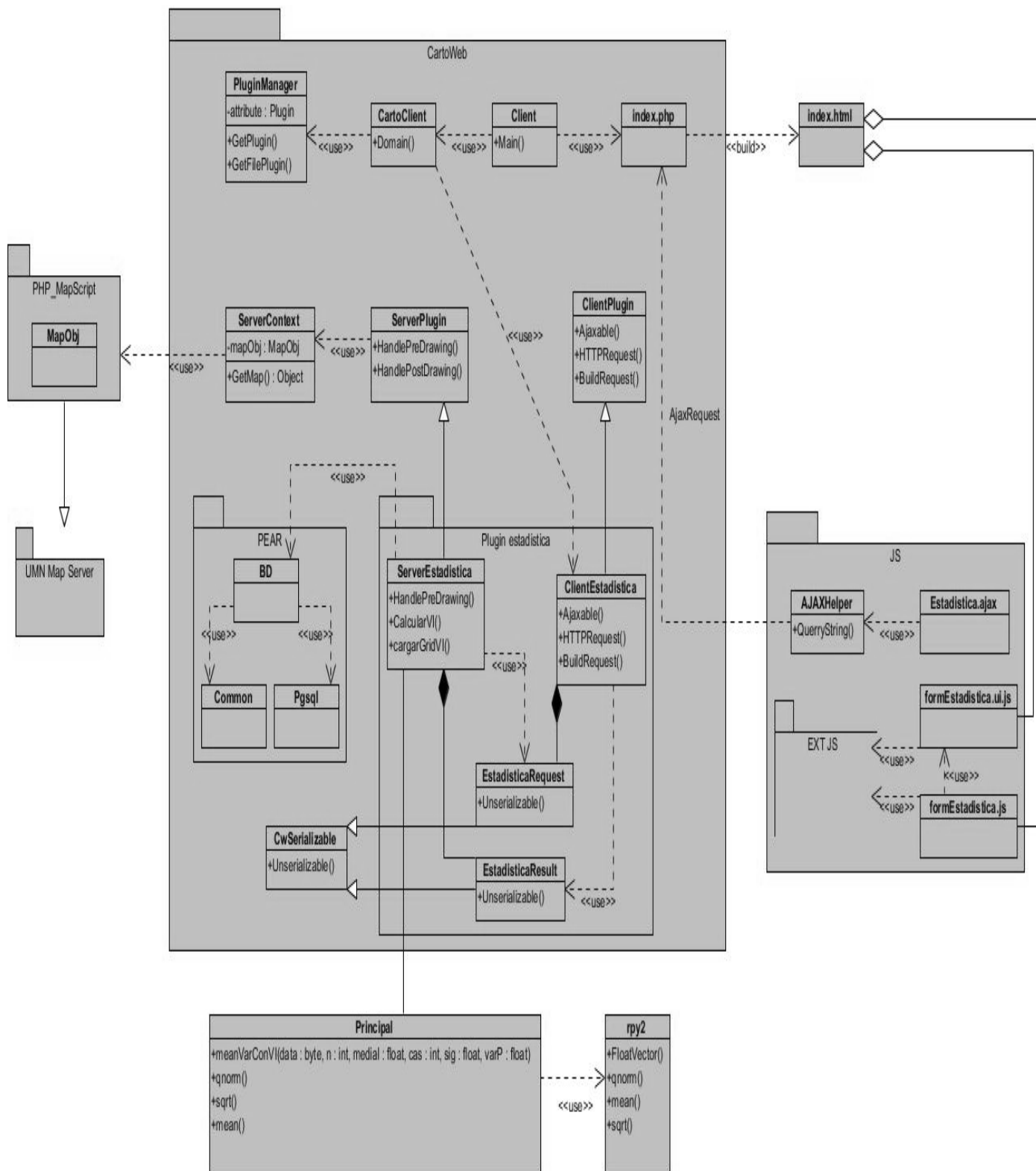


Fig.17: DCD del RF Realizar verificación de hipótesis de media con varianza conocida

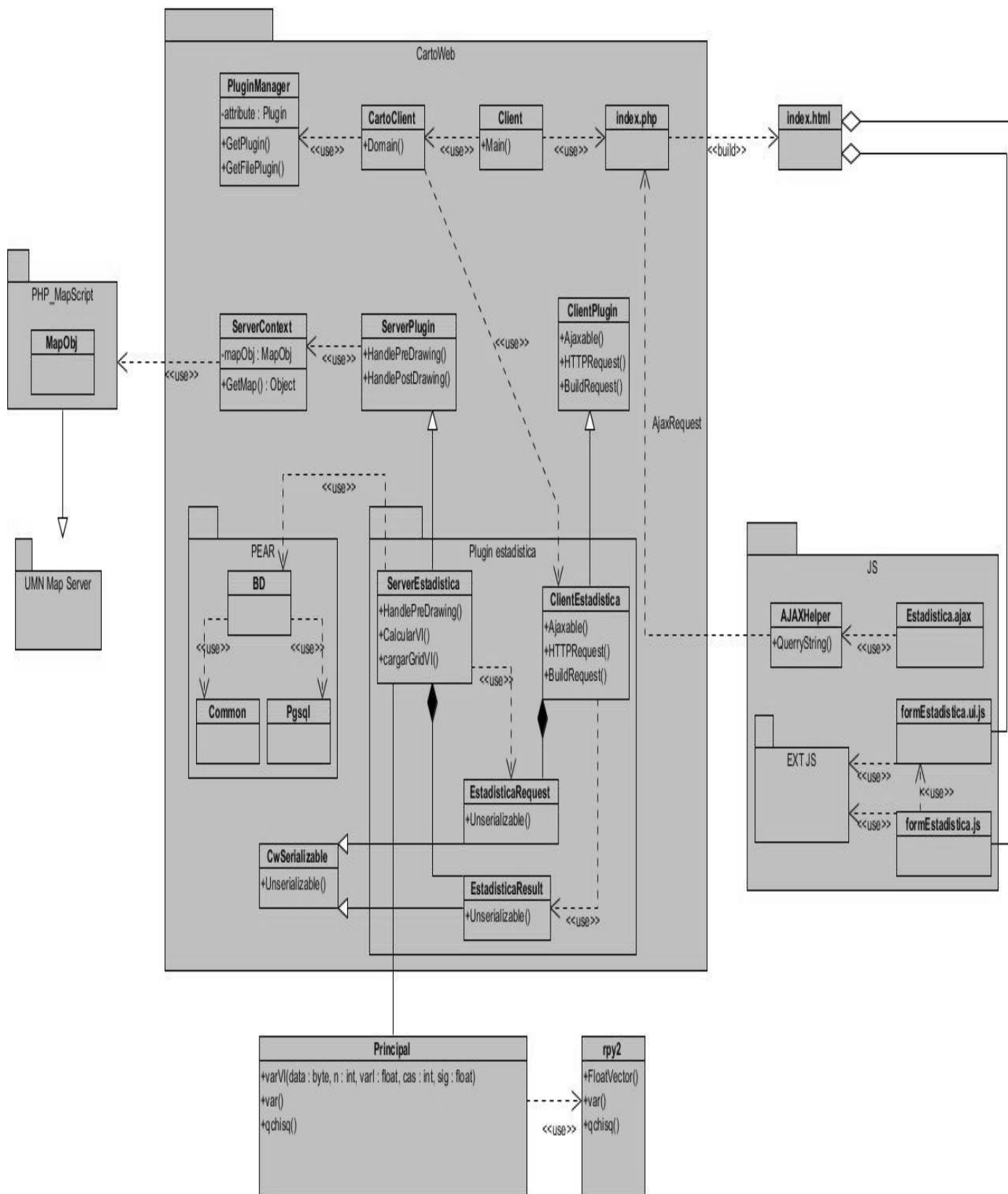


Fig.19: DCD del RF Realizar verificación de hipótesis de varianza

Anexo III: Diseño de casos de prueba

Tabla 13: DCP Realizar estimación puntual de varianza

Nombre de la sección	Escenarios de la sección	Capa	Atributo	Método	Estimador	Respuesta del Sistema	Flujo Central	
SC1: Realizar estimación puntual de varianza.	EC1.1: Realizar estimación puntual de varianza correctamente.	V	V	V	V	Muestra una tabla con los resultados de la estimación Puntual de varianza y los guarda en la base de datos.	El usuario selecciona la opción estadística inferencial para seleccionar la capa y el atributo, luego se selecciona la opción de Estimación y se selecciona el método Puntual, y luego el estimador de varianza. Por último se le da clic al botón Calcular.	
		empresa	cant_industriales	Puntual	Varianza			
	EC1.2: Campos Obligatorios vacíos.		I	V	V	V	El sistema no habilita el siguiente paso.	El usuario selecciona la opción estadística inferencial.
				N/A	N/A	N/A		El usuario selecciona la opción estadística inferencial e introduce la capa.
			V	I	V	V		
			empresa		N/A	N/A		
			V	V	I	V		
			empresa	cant_industriales		N/A		
	EC1.3: El atributo no corresponde a la capa.	El atributo no corresponde a la capa.	V	I	V	V	El sistema notifica que el atributo no corresponde a la capa.	El usuario le da al botón Calcular habiendo seleccionado un atributo que no corresponde a la capa.
			empresa	cant_estanques	Puntual	Varianza		
EC1.4:	Cerrar	V	V	V	V	El sistema oculta la ventana.	El usuario selecciona la opción	

COMPONENTE DE ESTADÍSTICA INFERENCIAL PARA LA PLATAFORMA GENESISG

	Ventana.	N/A	N/A	N/A	N/A		estadística inferencial y luego da clic en cerrar.
--	----------	-----	-----	-----	-----	--	--

Tabla 14: DCP Realizar estimación por intervalos de media con varianza conocida

Nombre de la sección	Escenarios de la sección	Capa	Atributo	Método	Estmador	Significación	Varz. Conoc	Respuesta del Sistema	Flujo Central	
SC1: Realizar estimación por intervalos de media con varianza conocida.	EC1.1: Realizar estimación por intervalos de media con varianza conocida correctamente.	V	V	V		V	V	Muestra una tabla con los resultados de la estimación por intervalos de media con varianza conocida y los guarda en la base de datos.	El usuario selecciona la opción estadística inferencial para seleccionar la capa y el atributo, luego se selecciona la opción de Estimación y se selecciona el método por Intervalos, y luego el estimador de media con varianza conocida posteriormente se introduce los valores de significación y varianza conocida. Por último se le da clic al botón Calcular.	
		empresa	cant_industriales	Por Intervalo	Media_Varz_Con	0.05	562.5			
	EC1.2: Campos Obligatorios vacíos.		I	V	V	V	V	V	El sistema no habilita el siguiente paso.	El usuario selecciona la opción estadística inferencial.
				N/A	N/A	N/A	N/A	N/A		
			V	I	V	V	V	V		
			empresa		N/A	N/A	N/A	N/A		
			V	V	I	V	V	V		El usuario selecciona la opción estadística inferencial e introduce la capa.
			empresa	cant_industriales		N/A	N/A	N/A		
			V	V	V	V	I	V		
			V	V	V	V	I	V		
							El sistema	El usuario selecciona la opción		

COMPONENTE DE ESTADÍSTICA INFERENCIAL PARA LA PLATAFORMA GENESISG

		empresa	cant_industriales	Por Intervalo	Media_Varz_Con			761.5	notifica que existen campos vacíos.	estadística inferencial, selecciona capa, atributo, la opción de Estimación y se selecciona el método por Intervalos, y luego el estimador de media con varianza conocida posteriormente se introduce valores dejando la significación o la varianza conocida vacíos. Luego le da clic al botón Calcular.
		V	V	V	V	V				
		empresa	cant_industriales	Por Intervalo	Media_Varz_Con			0.01		
EC1.3: El atributo no corresponde a la capa.	El no a	V	I	V	V	V	V		El sistema notifica que el atributo no corresponde a la capa.	El usuario le da al botón Calcular habiendo seleccionado un atributo que no corresponde a la capa.
		empresa	cant_estanques	Por Intervalo	Media_Varz_Con			0.05		
		V	I	V	V	V	V	612.5		
EC1.4: Valor de la significación fuera de rango.		V	V	V	V	I	V		El sistema notifica que el valor de la significación está entre 0 y 1.	El usuario realiza toda la secuencia de pasos para ejecutar la funcionalidad de forma correcta, pero introduce un valor de significación no válido.
		empresa	cant_industriales	Por Intervalo	Media_Varz_Con			2.05		
		V	V	V	V	V	V	452.14		
EC1.5: Cerrar Ventana.		V	V	V	V	V	V		El sistema oculta la ventana.	El usuario selecciona la opción estadística inferencial y luego da clic en cerrar.
		N/A	N/A	N/A	N/A	N/A	N/A			

COMPONENTE DE ESTADÍSTICA INFERENCIAL PARA LA PLATAFORMA GENESIG

Tabla 15: DCP Realizar estimación bayesiana

Nombre de la sección	Escenarios de la sección	Capa	Atributo	Método	Media a Priori	Desv. a Priori	Respuesta del Sistema	Flujo Central	
		empresa	cant_industriales	Bayesiana	35.2	62.5			
SC1: Realizar estimación bayesiana.	EC1.1: Realizar estimación bayesiana correctamente.	V	V	V	V	V	Muestra una tabla con los resultados de la estimación bayesiana y los guarda en la base de datos.	El usuario selecciona la opción estadística inferencial para seleccionar la capa y el atributo, luego se selecciona la opción de Estimación y se selecciona el método Bayesiana, posteriormente se introducen los valores de media a priori y desviación a priori. Por último se le da clic al botón Calcular.	
		empresa	cant_industriales	Bayesiana	35.2	62.5			
	EC1.2: Campos Obligatorios vacíos.		I	V	V	V	V	El sistema no habilita el siguiente paso.	El usuario selecciona la opción estadística inferencial.
				N/A	N/A	N/A	N/A		
			V	I	V	V	V		El usuario selecciona la opción estadística inferencial e introduce la capa.
			empresa		N/A	N/A	N/A		
			V	V	I	V	V	El usuario selecciona la opción estadística inferencial e introduce la capa y el atributo.	
			empresa	cant_industriales		N/A	N/A		
			V	V	V	I	V	El sistema notifica que existen campos vacíos.	El usuario selecciona la opción estadística inferencial, selecciona capa, atributo, la opción de Estimación y se selecciona el método Bayesiana, luego introduce valores dejando la media a priori o desviación a priori vacíos. Luego le da clic al botón Calcular.
			empresa	cant_industriales	Bayesiana		61.5		
	V	V	V	V	I				

COMPONENTE DE ESTADÍSTICA INFERENCIAL PARA LA PLATAFORMA GENESISG

		empresa	cant_industriales	Bayesiana	40.5				
	EC1.3: El atributo no corresponde a la capa.	V	I	V	V	V	El sistema notifica que el atributo no corresponde a la capa.	El usuario le da al botón Calcular habiendo seleccionado un atributo que no corresponde a la capa.	
		empresa	cant_estanques	Bayesiana	32.3	46			
	EC1.4: Cerrar Ventana.	V	V	V	V	V	El sistema oculta la ventana.	El usuario selecciona la opción estadística inferencial y luego da clic en cerrar.	
		N/A	N/A	N/A	N/A	N/A			

Tabla 16: DCP Realizar verificación de hipótesis de media con varianza conocida.

Nombre de la sección	Escenarios de la sección	Capa	Atributo	Método	Caso	Media	Significación	Varz. Conoc	Respuesta del Sistema	Flujo Central
SC1: Realizar verificación de hipótesis de media con varianza conocida.	EC1.1: Realizar verificación de hipótesis de media con varianza conocida correctamente.	V	V	V	V	V	V	V	Muestra una tabla con los resultados de la verificación de hipótesis de media con varianza conocida y los guarda en la base de datos.	El usuario selecciona la opción estadística inferencial para seleccionar la capa y el atributo, luego se selecciona la opción de Verificación de Hip. de DP y se selecciona el método Media_Varz_Con, posteriormente se selecciona el caso, y se introducen los valores de media, significación y varianza conocida. Por último se le da clic al botón Calcular.
	EC1.2: Campos Obligatorios vacíos.	I	V	V	V	V	V	V	El sistema no habilita el siguiente paso.	El usuario selecciona la opción estadística inferencial.

COMPONENTE DE ESTADÍSTICA INFERENCIAL PARA LA PLATAFORMA GENESISG

		V	I	V	V	V	V	V		El usuario selecciona la opción estadística inferencial e introduce la capa.
		empresa		N/A	N/A	N/A	N/A	N/A		
		V	V	I	V	V	V	V		El usuario selecciona la opción estadística inferencial e introduce la capa y el atributo.
		empresa	cant_industriales		N/A	N/A	N/A	N/A		
		V	V	V	V	I	V	V	El sistema notifica que existen campos vacíos.	El usuario selecciona la opción estadística inferencial, selecciona capa, atributo, la opción de Verificación de Hip. de DP y se selecciona el método Media_Varz_Con, luego selecciona el caso e introduce valores dejando la media, la significación o la varianza poblacional vacíos. Luego le da clic al botón Calcular.
		empresa	cant_industriales	Media_Varz_Con	2		0.05	98.1		
		V	V	V	V	V	I	V		
		empresa	cant_industriales	Media_Varz_Con	1	22.4		426.25		
		V	V	V	V	V	V	I		
		empresa	cant_industriales	Media_Varz_Con	3	46	0.05			
EC1.3: Valor de		V	V	V	V	V	I	V	El sistema	El usuario realiza toda la

COMPONENTE DE ESTADÍSTICA INFERENCIAL PARA LA PLATAFORMA GENESISG

	la significación fuera de rango.		empresa	cant_industriales	Media_Varz_Con	3	54.14	1.02	355.1	notifica que el valor de la significación está entre 0 y 1.	secuencia de pasos para ejecutar la funcionalidad de forma correcta, pero introduce un valor de significación no válido.
EC1.4:	El atributo no corresponde a la capa.	V	empresa	cant_estanques	Media_Varz_Con	2	51.2	0.01	756.3	El sistema notifica que el atributo no corresponde a la capa.	El usuario le da al botón Calcular habiendo seleccionado un atributo que no corresponde a la capa.
EC1.5:	Cerrar Ventana.	V	N/A	N/A	N/A	N/A	N/A	N/A	N/A	El sistema oculta la ventana.	El usuario selecciona la opción estadística inferencial y luego da clic en cerrar.

Tabla 17: DCP Realizar test de K-S.

Nombre de la sección	Escenarios de la sección	Capa	Atributo	Test	Significación	Respuesta del Sistema	Flujo Central
SC1: Realizar test de K-S.	EC1.1: Realizar test de K-S correctamente.	V	V	V	V	Muestra una tabla con los resultados del test K-S y los guarda en la base de datos.	El usuario selecciona la opción estadística inferencial para seleccionar la capa y el atributo, luego selecciona la opción de Verificación de Hip. de DNP y se selecciona el test K-S, posteriormente se introduce el valor de la significación. Por último se le da clic al botón Calcular.
		empresa	cant_industriales	K-S	0.01		
	EC1.2: Campos Obligatorios vacíos.	I	V	V	V	El sistema no habilita el siguiente paso.	El usuario selecciona la opción estadística inferencial.
		V	I	V	V		El usuario selecciona la opción

COMPONENTE DE ESTADÍSTICA INFERENCIAL PARA LA PLATAFORMA GENESISG

		empresa					estadística inferencial e introduce la capa.
		V	V	I	V		El usuario selecciona la opción estadística inferencial e introduce la capa y el atributo.
		empresa	cant_industriales				
		V	V	V	I		
		empresa	cant_industriales				
				K-S		El sistema notifica que existen campos vacíos.	El usuario selecciona la opción estadística inferencial, selecciona capa y atributo, luego selecciona la opción de Verificación de Hip. de DNP y se selecciona el test K-S, posteriormente se deja el campo de la significación vacío. Luego le da clic al botón Calcular.
EC1.3: El atributo no corresponde a la capa.		V	I	V	V	El sistema notifica que el atributo no corresponde a la capa.	El usuario le da clic al botón Calcular habiendo seleccionado un atributo que no corresponde a la capa.
		empresa	cant_estanques				
				K-S	0.01		
EC1.4: Valor de la significación fuera de rango.		V	V	V	I	El sistema notifica que el valor de la significación está entre 0 y 1.	El usuario realiza toda la secuencia de pasos para ejecutar la funcionalidad de forma correcta, pero introduce un valor de significación no válido.
		empresa	cant_industriales				
				K-S	1.24		
EC1.5: Cerrar Ventana.		V	V	V	V	El sistema oculta la ventana.	El usuario selecciona la opción estadística inferencial y luego da clic en cerrar.
		N/A	N/A	N/A	N/A		