

**Universidad de las Ciencias Informáticas
Facultad 6**

**Sistema de Información Geográfica
para la gestión de recursos hidráulicos**



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

Autor (es): José Raúl de Armas de la Nuez
José Alejandro Varela Ortega

Tutor (es): **MsC.** Yuniel Eliades Proenza Arias
Ing. Pedro Manuel Salas Leyva

“Año 58 de la Revolución”
La Habana, Cuba
Julio, 2016

**“Lo que sabemos es una gota de agua;
lo que ignoramos es el océano”**



Declaración de autoría

Declaro ser el único autor de la presente tesis y reconozco a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

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

José Raúl de Armas de la Nuez

Autor

José Alejandro Varela Ortega

Autor

MsC. Yuniel Eliades Proenza Arias

Tutor

Ing. Pedro Manuel Salas Leyva

Tutor

Datos de contacto

Tutor: MsC. Yuniel Eliades Proenza Arias.

Edad: 33.

Ciudadanía: cubana.

Institución: Universidad de las Ciencias Informáticas (UCI).

Títulos: Ingeniero Informático, Máster en Ingeniería de Software e Inteligencia Artificial.

Síntesis del Tutor: Graduado de Ingeniería Informática en el año 2006 de la Universidad de Holguín y CUJAE. Máster en Ingeniería de Software e Inteligencia Artificial por la Universidad de Málaga en el 2011.

Se ha desempeñado como analista y desarrollador de aplicaciones Web y Desktop. Tiene experiencia en el trabajo con Sistemas de Información Geográfica y el desarrollo de Ontologías.

E-mail: yproenza@uci.cu.

Tutor: Ing. Pedro Manuel Salas Leyva.

Edad: 26.

Ciudadanía: cubana.

Institución: Universidad de las Ciencias Informáticas (UCI).

Títulos: Ingeniero en Ciencias Informáticas.

Síntesis de la Tutor: Graduado de Ingeniero en Ciencias Informáticas en el 2015 en la UCI.

E-mail: pmsalas@uci.cu.

Dedicatorias

José Alejandro:

Dedico esta tesis a mi familia por todo su amor, sacrificio y dedicación desde mis primeros pasos en esta vida.

A mi novia y a toda su familia por su apoyo incondicional.

José Raúl:

Deseo dedicar esta tesis principalmente a mis abuelos que aunque ya no estén a mi lado, siempre eran los que más elogiaban mis logros y se preocupaban porque siempre me superara.

A mis padres que siempre lo han dado todo para que yo sea una mejor persona y un buen profesional.

A mi tío Pepe y mi tía Marisol que siempre me han ayudado en todo lo que haga falta y apoyado en todos los momentos de mi vida, aunque estén lejos.

A mi ángel Rosmery por siempre estar a mi lado desde que empecé la universidad y a mis suegros Rosita y Robayna.

Agradecimientos

José Alejandro:

Deseo agradecer a mi compañero de tesis que más que un compañero es un hermano para mí, ya que estamos juntos desde casi nuestros primeros pasos de nuestra vida, hemos estado en todo momento juntos en los buenos y malos y seguiremos estando y podrás contar conmigo siempre para lo que sea, ha sido un privilegio tenerlo también como compañero de tesis y haber estado juntos todos estos años, si nos hubiéramos separado la universidad no hubiese sido igual.

A mis abuelos Fernando y Silvia por su ejemplo y apoyo incondicional, por ser mis segundos padres.

A mis padres que me han acompañado desde mis primeros pasos y me han guiado durante toda mi vida al éxito y a la felicidad. Por ser lo más grande que tengo, por todos los valores que me han inculcado que han hecho de mí una mejor persona, por enseñarme con su ejemplo y protegerme con su amor.

A mis tías Folexi y Gary, y mis abuelos Pepe y Caruca por su apoyo.

A mi princesa, a mi sol eterno que siempre me ha apoyado en todo y me ha dado las fuerzas para salir siempre adelante en los buenos y malos momentos sin ella no hubiese sido igual.

A mis suegros que han sido mi segunda familia y me han brindado todo su apoyo y dado fuerzas para vencer todos los obstáculos.

A los tutores Proenza y Pedro porque siempre estuvieron pendiente y preocupado por la evolución de este trabajo.

A mi hermano Wandy que ha estado desde el preuniversitario apoyándome en los buenos y malos momentos, por su ejemplo de sacrificio y podrá siempre contar conmigo

A mis hermanos César, Edel, Rafa, Carlos y el Cuso que ha sido un verdadero honor haberlos conocido

A todo mis compañeros del grupo 6501.

Quisiera agradecer a todas las personas que en gran medida contribuyeron a que me convirtiera en ingeniero.

José Raúl.

Primero que nada deseo agradecer a mi compañero de tesis, que es mi hermano mayor desde antes de dar nuestros primeros pasos y aprender a hablar, hemos compartido todas las etapas de la vida desde la primaria, hasta la universidad y espero que todo lo que nos quede por delante. Además siempre hemos sido uno en todo momento, ya sea en las buenas y en las malas, para mí también es un privilegio haber compartido este momento contigo porque ha sido de gran satisfacción, por todas las cosas que hemos vivido juntos.

A mis abuelos por inculcar e incentivar en mí los mejores valores y principios, para moldear en mí una mejor persona.

A mis padres que han sido el pilar fundamental de todo lo que soy hoy en día, por estar en todo momento presentes, por quererme, apoyarme, amarme y saber guiarme por el mejor camino ya que son mi luz y significan todo para mí.

A mi tío Pepe y mi tía Marisol por ser mis segundos padres y guiarme a ser un buen profesional, con sus consejos y sus enseñanzas.

A mi hermana Susana por seguir mis pasos.

A mi hermana Penli por estar siempre al tanto de todo

A mi novia Rosmery por apoyarme en todo y darme ganas y ánimo para lograr las cosas y ser lo mejor que me ha pasado en la vida, porque a su lado he vivido mucho de mis mejores y peores momentos, por quererme tanto y ser alguien especial.

A mis suegros Rosita y Robayna por siempre apoyarme con la universidad, en la vida y por quererme tanto.

A mi hermano Wandy por también compartir mis mejores y malos momentos y aunque hemos estado en universidades separadas ha sido para mí un ejemplo de dedicación y perseverancia.

A mis tutores Pedro y Proenza por guiarme en el desarrollo de este trabajo y ayudarme en todo lo que estaba a su alcance.

A mis hermanos de la universidad especialmente a: Edel, César, Carlos, Rafa y el Suso que tanto hemos vivido en estos 5 maravillosos años.

A todos mis amigos de la universidad y en especial a los de mi año.

Y finalmente a todas las personas que de una forma u otra tuvieron que ver con este trabajo y ayudaron a su desarrollo.

Resumen

Actualmente el país presenta problemas para resolver la gestión de los recursos hidráulicos específicamente del agua embalsada, la cual es la fuente principal de acueductos y suministros de agua para la población, debido a que se hace de forma manual y la información está guardada en formato duro. El trabajo se centra en el desarrollo de un sistema que permita la gestión de los recursos hidráulicos de los embalses que se encuentran dentro del territorio cubano. El mismo, forma parte de un Sistema de Información Geográfica que sigue como objetivo fundamental registrar, organizar, actualizar, conservar y mostrar información de una forma dinámica, eficiente y segura. Su desarrollo está basado en tecnologías libres o de código abierto. Se utiliza php como lenguaje de programación y PostgreSQL como servidor de aplicaciones. El uso de esta aplicación traerá como beneficios que las instituciones necesitadas que deseen gestionar información de embalses puedan hacerlo de una forma más rápida, además de aportarles información visual de donde se encuentran los mismos y también busca incorporar funcionalidades que en el acontecer internacional hayan tenido gran aceptación por personalidades relacionadas con el tema en cuestión.

Palabras clave: embalses, gestión, recursos hidráulicos.

Abstract

Nowadays our country has problems to solve the management of hydraulic resources specifically of embalmed water, which is the main fountain of aqueduct and water supply for the population, due to it is done in a manual way, and the information is kept in hard form .This work is based on the development of a system that lets the management of the hydraulic resources of the reservoirs which are in Cuban territories. It is included in a system of Geographic Information, and its main objective is to register, organize, actualize, conserve, and to show information in a more dynamic, efficient and safe way. Its development is based on technologies of books or open code. Programme language PHP and PostgreSQL are used as servers of applications. The use of these applications will bring as benefits that the needed institutions which desire to get information about reservoirs can do it in a faster way. They will also give a visual information, they will also show us where they try to incorporate functionalities which have happened internationally, and they have had a great acceptance by personalities related with this topic.

Key Words: reservoirs, management, hydraulic, resources.

Índice de contenidos

Introducción	- 15 -
Métodos de Investigación	- 4 -
Capítulo #1. Fundamentación teórica	- 1 -
1.1 Introducción	- 1 -
1.2 Sistemas de Información Geográfica	- 1 -
1.2.1 Componentes de un SIG	- 2 -
1.2.2 Características principales de un SIG	- 3 -
1.2.3 Funciones básicas de un SIG	- 4 -
1.2.4 Limitaciones de un SIG	- 4 -
1.3 Contexto relacionado con “Recursos Hidráulicos”	- 5 -
1.3.1 Concepto de embalse	- 6 -
1.3.2 Concepto de cuenca hidrográfica	- 6 -
1.4 Análisis de soluciones existentes	- 6 -
1.5 Herramientas y Tecnologías a utilizar para dar solución al problema	- 7 -
1.5.1 Plataforma GeneSIG 2.0	- 7 -
1.5.2 Metodología de desarrollo	- 8 -
1.5.3 Lenguaje de Modelado	- 9 -
1.5.4 Sistema gestor de Base de Datos. PostgreSQL9.3	- 9 -
1.5.5 Servidor de aplicaciones	- 10 -
1.5.6 Servidor de mapas	- 10 -
1.5.7 Lenguajes de programación del lado del servidor	- 10 -
1.5.8 Lenguajes de programación del lado del cliente	- 10 -
1.5.9 Entorno de Desarrollo Integrado (IDE). PhpStorm	- 11 -
1.6 Conclusiones parciales	- 11 -
Capítulo #2. Características y diseño del sistema	- 13 -
2.1 Introducción	- 13 -
2.2 Propuesta de solución	- 13 -
2.3 Modelo de dominio	- 13 -
2.4 Descripción de los conceptos relacionados en el modelo de dominio	- 14 -
2.5 Requisitos del sistema	- 15 -
2.5.1 Requisitos Funcionales	- 15 -
2.6 Requisitos No Funcionales	- 17 -

2.7	Casos de uso del sistema	- 26 -
2.7.1	Diagrama de casos de uso del sistema	- 26 -
2.7.2	Descripción del actor del sistema	- 26 -
2.7.3	Patrones de casos de uso del sistema	- 27 -
2.7.4	Especificación de casos de uso del sistema	- 28 -
2.8	Arquitectura del sistema	- 35 -
2.8.1	Estilo Arquitectónico	- 35 -
2.8.2	Patrones de arquitectura	- 36 -
2.9	Diseño del Sistema	- 36 -
2.9.1	Patrones de diseño	- 37 -
2.9.2	Diagrama de Clases del diseño	- 38 -
2.9.3	Descripción de las clases del diseño	- 38 -
2.10	Diseño de la Base de Datos	- 40 -
2.11	Conclusiones parciales	- 41 -
Capítulo #3. Implementación y prueba del sistema		- 42 -
3.1	Introducción	- 42 -
3.2	Estilo de programación	- 42 -
3.2.1	Definición de clases	- 42 -
3.2.2	Estilo de métodos	- 43 -
3.2.3	Declaración de variables	- 43 -
3.2.4	Estructuras de control	- 44 -
3.3	Diagrama de componentes	- 44 -
3.4	Diagrama de despliegue	- 45 -
3.5	Pruebas	- 45 -
3.5.1	Diseño de Casos de Prueba	- 46 -
3.5.2	Resultados de las pruebas realizadas	- 51 -
3.6	Conclusiones parciales	- 52 -
Conclusiones generales		- 53 -
Recomendaciones		- 54 -
Bibliografía referenciada		- 55 -

Índice de figuras

Figura 1. Modelo conceptual del SIG RH.	- 14 -
Figura 2. Diagrama de Casos de Uso del Sistema del SIG RH.	- 26 -
Figura 3. Patrón de CU CRUD Completo presente en el diagrama de CUS del SIG RH.	- 27 -
Figura 4. Patrón de CU Extensión concreta presente en el diagrama de CUS del SIG RH.	- 27 -
Figura 3. Diagrama de Clases del Diseño del CU Gestionar embalse del SIG RH.	- 38 -
Figura 4. Diagrama de Clases Persistentes.	- 40 -
Figura 5. Modelo físico de la base de datos.	- 41 -
Figura 6. Estructura de las clases.	- 42 -
Figura 7. Estructura de los métodos.	- 43 -
Figura 8. Estructura de una variable.	- 43 -
Figura 9. Estructura de un if.	- 44 -
Figura 10. Diagrama de componentes de CU Gestionar embalse.	- 45 -
Figura 11. Diagrama de despliegue del SIG RH.	- 45 -
Figura 12. Pruebas de Caja Negra. Cantidad de Casos Prueba y No Conformidades por cada iteración.	- 52 -

Índice de tablas

Tabla 1. Descripción de los conceptos del dominio del negocio.	- 14 -
Tabla 2. Requisitos funcionales del SIG RH.	- 15 -
Tabla 3. Descripción del RnF de Usabilidad-Operabilidad del SIG RH.	- 18 -
Tabla 4. Descripción del RnF de Usabilidad-Agradabilidad del SIG RH.....	- 19 -
Tabla 5. Descripción del RnF de Fiabilidad-Disponibilidad del SIG RH.....	- 20 -
Tabla 6. Descripción del RnF de Mantenibilidad- Analizabilidad del SIG RH.	- 21 -
Tabla 7. Descripción del RnF de Mantenibilidad- Modificabilidad del SIG RH.....	- 22 -
Tabla 8. Descripción del RnF de Adecuación funcional- Integridad funcional del SIG RH.	- 23 -
Tabla 9. Descripción del RnF de Eficiencia en el rendimiento- Utilización de recursos del SIG RH.-	24 -
Tabla 10. Descripción del actor del sistema del SIG RH.....	- 26 -
Tabla 11. Descripción textual del CU Gestionar embalse.	- 28 -
Tabla 12. DCP del CU Gestionar embalse, SC1 Adicionar embalse del SIG RH.	- 46 -
Tabla 13. DCP del CU Gestionar embalse, SC2 Modificar embalse del SIG RH.	- 47 -
Tabla 14. DCP del CU Gestionar embalse, SC3 Eliminar embalse del SIG RH.....	- 49 -
Tabla 15. Variables del DCP del CU Gestionar embalse del SIG RH.	- 49 -

Introducción

El agua es un recurso natural renovable, limitado y frágil, está presente en los tres pilares del desarrollo sostenible (economía, sociedad, y medio ambiente). La única fuente de formación lo constituyen las precipitaciones y esta se distribuye de manera no uniforme en el espacio geográfico y en el tiempo, existiendo dos periodos bien diferenciados el lluvioso o húmedo y el menos lluvioso o seco. El 97,47% del agua en el planeta es agua de mar y alrededor del 2,53% agua dulce. Cerca del 0,5% del total del agua dulce registrada, soporta las necesidades del desarrollo sostenible. A pesar de eso 1 400 millones de personas no tienen acceso al agua potable y aunque esta cifra ya releva una situación preocupante, la realidad es mucho peor aún, porque millones de personas pobres, que viven en asentamientos precarios simplemente no están contabilizadas en las estadísticas. Las causas básicas de la actual crisis del agua radican en la pobreza, las desigualdades, en la disparidad en las relaciones de poder, un mal manejo y uso irracional de la misma, y se ven agravadas por los retos sociales y ambientales, como la urbanización cada vez más rápida, el cambio climático, y la creciente contaminación y merma de los recursos hídricos. Para el 2025 existe un pronóstico de que 18 países enfrenten déficit crónico de agua dulce lo que abarcará el 35% de la población que se estima que exista para ese año. Para el año 2050 se espera que la población mundial aumente un 33%, pasando de 7 000 millones a 9 300 millones de habitantes, por lo cual la demanda de alimento aumentará un 60%. Además se prevé que la población que vive en las áreas urbanas casi se duplicará. Se considera fundamental mejorar la eficiencia en el uso del agua para hacer frente a la brecha del 40% entre la oferta y la demanda y mitigar la escasez de agua en 2050 (Gil, 2016).

La ocurrencia de los fenómenos naturales hidrometeorológicos extremos como la sequía y los huracanes, así como los impactos del cambio climático, han agravado el escenario de la escasez del agua en Cuba. Los recursos hídricos disponibles anuales per cápita en el país son limitados y están heterogéneamente distribuidos, siendo las provincias habaneras y las orientales, con excepción de Granma, las menos favorecidas. La media nacional de 1220 m³ por persona al año sitúa al país en un nivel de estrés hídrico moderado. Entre los factores de origen natural que intervienen en la carencia del agua en Cuba se distingue el carácter insular del país, conformado por una isla grande alargada y estrecha, en la que predominan los ríos de pequeña longitud (menor de 40 km) y de poco caudal. Por estas razones se hace necesario la creación de infraestructuras como presas y embalses para un mejor aprovechamiento de este recurso.

A finales de los años 80 la máxima dirección del país comenzó a señalar la urgencia de impulsar las construcciones hidráulicas, atendiendo a que los mecanismos administrativos implantados no habían sido capaces de resolver las necesidades para cuya demanda habían sido concebidos. En tal sentido, el

Presidente de los Consejos de Estado y de Ministros, Fidel Castro, en sus palabras pronunciadas en el XI Período de Sesiones de la Asamblea Nacional del Poder Popular, el 26 de diciembre de 1986 expresó: *“en determinado momento hubo una gran voluntad hidráulica en el país, que se fue desarrollando desde el ciclón Flora y las devastadoras inundaciones que produjeron pérdidas de más de mil vidas y grandes daños económicos (...) esa voluntad adquirió gran fuerza en la década que va de 1965 a 1975 y posteriormente decayó progresivamente. Faltaron asignaciones, recursos en los planes y decayó en general la intensidad del trabajo en la construcción de presas y sistemas de riego”* (Castro, 1986)

Con el objetivo de lograr un avance, se decidió crear un nuevo organismo de la Administración Central del Estado, encargado de dirigir, ejecutar y controlar la aplicación de la política estatal y gubernamental en cuanto a las actividades de planificación y control de los recursos hídricos del país, lo cual fue refrendado mediante el Decreto-Ley Número 114 del 6 de junio de 1989, y adoptaría igual denominación que a uno ya creado en 1962: Instituto Nacional de Recursos Hidráulicos. Con la misión de organizar y dirigir en coordinación con los organismos competentes la protección de las aguas terrestres, las cuencas, los cauces naturales, las obras e instalaciones hidráulicas contra los peligros de contaminación, azolvamiento y otras formas de degradación y deterioro, así como el control sistemático de la calidad de las aguas. Con respecto a la infraestructura hidráulica, el INRH administra en la actualidad 242 embalses y 700 micropresas con una capacidad total de 9 128 millones de m³.

El Instituto Nacional de Recursos Hidráulicos maneja mucha información de los embalses, como es la capacidad máxima, el volumen actual, provincia y municipio al que pertenecen, municipios a los que abastecen, entre otros de importancia. Todos estos datos los especialistas de cada embalse lo guardan en un formato físico diariamente, para su posterior consulta, por lo que genera un exceso de documentación y la pérdida de la misma por su deterioro. La consulta de los datos de los embalses por el INRH se realiza por vía telefónica o por correo electrónico, pudiendo existir errores en los datos y que, si surgiera la necesidad de consultarlos, los mismos no estén actualizados, debido a que no existe una política de actualización de los datos por parte de los especialistas de las estaciones de monitoreo.

Esto influye negativamente en la toma de decisiones por si ocurre alguna situación de emergencia como sequías o inundaciones ya que es necesario tener en cuenta las características y datos de los embalses. La toma de decisiones se realiza mediante la consulta de mapas físicos y la información recopilada hasta el momento, lo que puede ocasionar demoras en el tiempo de análisis, ya que tienen que conocer la ubicación de los embalses, para analizar la situación de cada provincia o municipio del país. También tienen que calcular de forma manual el porcentaje de llenado del embalse y el estado de la provincia para prever ante

fuerzas de lluvias o huracanes el vertimiento de los embalses a tiempo y que no ocurran inundaciones. La no realización de estos análisis a tiempo o la realización de los mismos incorrectamente implica en ocasiones lamentar pérdidas humanas y/o materiales. Por estas razones se hace necesario mejorar la gestión de la información que se tiene de estos sistemas hídricos.

En consecuencia a lo expuesto anteriormente se puede plantear como **problema de la investigación**. ¿Cómo mejorar la disponibilidad y gestión de la información de los Recursos Hidráulicos en Cuba para disminuir el tiempo empleado en la toma de decisiones ante determinadas situaciones? El **objeto de estudio** lo constituye el proceso de gestión y análisis de información de recursos hidráulicos. El **campo de acción** proceso de gestión y análisis de información de presas y embalses en Cuba.

Para solucionar el problema planteado es necesario definir como **objetivo general** desarrollar un Sistema de Información Geográfica para la gestión de Recursos los Hidráulicos en Cuba.

El objetivo propuesto conllevó a plantear, como base para el desarrollo de la investigación, las **preguntas científicas** siguientes:

¿Cómo se realiza en nuestro país la gestión y manejo de la información referente a embalses y presas?

¿Qué factores influyen en la demora para la toma de decisiones ante situaciones de emergencia?

¿Cómo se podría disminuir el tiempo empleado en el análisis de estado de los embalses y análisis de riesgo de inundación?

¿Cómo impactaría la automatización de los procesos de captura y análisis de la información de Recursos hidráulicos en Cuba?

Para el cumplimiento de la investigación serán desarrolladas las siguientes **tareas**:

- ✓ Elaboración del marco teórico de la investigación a partir del estado del arte existente sobre el tema actual.
- ✓ Análisis detallado sobre los Sistemas de Información Geográfica.
- ✓ Selección de las metodologías, herramientas de software a utilizar y lenguajes de programación para el desarrollo del sistema informático.
- ✓ Generación de los artefactos que corresponden a los flujos de trabajo definidos por la metodología seleccionada.

- ✓ Implementación y pruebas al sistema informático desarrollado.
- ✓ Validación de la propuesta realizada.

Métodos de Investigación

Los métodos teóricos posibilitan el conocimiento del estado del arte del fenómeno, su evolución en una etapa determinada y su relación con otros fenómenos. Además, permiten estudiar las características del objeto de investigación que no son observables directamente (Zayas, 1997). A continuación se explican los métodos teóricos empleados para el desarrollo de la investigación:

Métodos teóricos:

Análisis-histórico lógico: Este método se utiliza para realizar un análisis del estado del arte relacionado con el desarrollo de los SIG, sus principales conceptos, ventajas y desventajas, elementos que los componen y estructuración de la información de estos sistemas.

Analítico-sintético: Este método se emplea para analizar la información encontrada sobre los rasgos distintivos de los SIG, para poder extraer elementos generales y específicos tratando de establecer una relación entre los diferentes criterios que puedan existir y tributen al desarrollo de los mismos.

Capítulo #1. Fundamentación teórica

1.1 Introducción

En el presente capítulo se abordan los elementos principales que justifican y sirven de base a la solución del problema planteado, analizando los principales conceptos relacionados con el objeto de estudio y el campo de acción. Se realiza el análisis de algunas de las soluciones existentes a nivel mundial para el diseño de consultas y se describen las herramientas y metodología a utilizar durante el desarrollo de la solución propuesta.

1.2 Sistemas de Información Geográfica

Un Sistema de Información Geográfica es el conjunto de métodos, herramientas y actividades que actúan coordinada y sistemáticamente para recolectar, almacenar, validar, manipular, integrar, analizar, actualizar, extraer y desplegar información, tanto gráfica como descriptiva de los elementos considerados, con el fin de satisfacer múltiples propósitos (Mateus Pulido, 2009).

“Un Sistema de Información Geográfica (SIG, para sus siglas en español o GIS, para sus siglas en inglés) consiste en información de naturaleza diversa sobre un determinado territorio, almacenada en un conjunto de bases de datos tanto gráficas como alfanuméricas, cuya relación con el territorio se realiza a través de un sistema de referencia geográfico y se gestiona a través de uno o varios programas informáticos específicos; el conjunto es soportado por un sistema de computadores y por un personal especializado.” (Fallas, 2011).

“Un SIG puede ser concebido como una especialización de un sistema de bases de datos, caracterizado por su capacidad de manejar datos geográficos, que están geo-referenciados y los cuales pueden ser visualizados como mapas.” (Braken, 1992).

“Un SIG es un intento más o menos logrado según los casos de constituir una visión esquemática de una realidad compleja.” (Bosque-Sendra, 1992).

Se puede definir SIG como una base de datos especializada que contiene objetos geométricos (Cebrian, 1994)

“Un conjunto de hardware, software y datos geográficos para capturar, manipular, analizar y mostrar información geográficamente referenciada.” (ESRI, 1995).

Debido a la gran diversidad de definiciones propuestas para los SIG y luego de estudiar varias de ellas, los autores de la presente investigación asume la propuesta por el National Center for Geographic Information and Analysis (NCGIA): “Un Sistema de Información Geográfica es una integración organizada de hardware,

software, datos geográficos, métodos y relaciones organizativos, diseñado para capturar, almacenar, manipular, analizar y desplegar en todas sus formas la información geográficamente referenciada con el fin de resolver problemas complejos de planificación y gestión (NCGIA, 1990).

Se identifican tres principales clasificaciones de SIG:

SIG de Escritorios: Son aquellos que se utilizan para crear, editar, administrar, analizar y visualizar los datos geográficos sobre entornos de escritorio. Estos según la finalidad a la que vayan destinados se clasifican en Visores SIG, Editores SIG y SIG de análisis.

- ✓ Un Visor SIG permite visualizar información geográfica a través de un formulario al cual se le agregan varias capas de información.
- ✓ Un Editor SIG realiza tratamiento previo de información para su análisis. Antes de adicionar los datos estos se preparan para su uso transformándolos a un formato entendible por el software.
- ✓ Los SIG de Análisis presentan funcionalidades de análisis espacial y modelización cartográfica de procesos.

SIG Web: Son utilizados para mostrar, editar y analizar cartografía en entornos web. Permiten el acceso a grandes volúmenes de datos geográficos. Ofrecen por lo general imágenes aéreas o de satélite. Este tipo de SIG son los más utilizados debido al rápido acceso a través de un navegador web.

SIG Móviles: Son utilizados para crear, manipular, analizar y representar información geográfica en los dispositivos móviles. Estos SIG se pueden instalar en el mismo dispositivo o se pueden acceder a ellos mediante la utilización del Protocolo de Aplicaciones Inalámbricas a través de un navegador.

1.2.1 Componentes de un SIG

A continuación, se detallan los elementos que conforman esta clase de sistemas:

Equipos (Hardware): Equipo tecnológico en donde opera el SIG. Actualmente, un SIG puede ejecutarse en varios tipos de hardware, desde servidores de computadoras centralizados hasta computadoras de escritorio utilizadas en configuraciones individuales o de red. Una organización requiere de hardware suficientemente específico para cumplir las necesidades de la aplicación. Algunos aspectos a considerar incluyen: velocidad, costo, soporte, administración, disponibilidad, escalabilidad y seguridad (Olaya, 2011).

Programas (Software): Brindan las herramientas que posibilitan la entrada y manipulación de los datos, el análisis, la visualización, las búsquedas y la administración de las bases de datos. Con este fin un SIG está compuesto por:

- Sistema gestor de base de datos.
- Una interfaz gráfica de usuarios para el fácil acceso a las herramientas.
- Herramientas para captura y manejo de información geográfica.
- Herramientas para soporte de consultas, análisis y visualización de datos geográficos (Olaya, 2011).

Datos: Son elementos clave de un SIG, pues sin ellos el resto de los componentes no tienen utilidad alguna. Esto se debe a que es posible obtener la información deseada, pues a partir de su interpretación (Olaya, 2011).

Recurso humano: Para el correcto funcionamiento de los SIG se necesita personal capacitado para operar, desarrollar y administrar el sistema. Por ello, es imprescindible un perfil técnico con amplios conocimientos en el rol que desempeña, en los datos que se manejan y su naturaleza (Olaya, 2011).

Procedimientos: Los SIG deben operar acorde con un plan bien diseñado, con reglas claras del negocio, siguiendo modelos y prácticas operativas (Olaya, 2011).

1.2.2 Características principales de un SIG

Un grupo de características sobre los SIG son importantes destacar y a continuación se mencionan:

- La capacidad de visualización de información geográfica compleja a través de mapas.
- La funcionalidad de los SIG como una BD sofisticada, en la que se mantiene y relaciona información espacial y temática.
- La diferencia con las BD convencionales radica en que toda la información contenida en un SIG está unida a entidades geográficamente localizadas. Por ello, en un SIG la posición de las entidades constituye el eje del almacenamiento, recuperación y análisis de los datos.
- Representan una tecnología de integración de información.
- Se han desarrollado a partir de innovaciones tecnológicas sobre campos especializados de la geografía y otras ciencias (tratamiento de imágenes, análisis fotogramétricos, cartografía automática, entre otras), para constituir un sistema único más potente que la suma de las partes.
- Permiten unificar la información en estructuras coherentes.
- Este carácter integrador y abierto, hace de los SIG un área de contacto entre variados tipos de aplicaciones informáticas, destinadas al manejo de información con propósitos y formas diversas;

por ejemplo: programas estadísticos, gestores de bases de datos, programas gráficos, hojas de cálculo, procesadores de texto, entre otros.

- Los límites y diferencias entre los SIG se enfocan en, los programas de diseño asistido por computadoras, los de cartografía temática y los de tratamiento de imágenes que son especialmente difusos. Aunque sus diferencias estriban sobre todo en el modelo de datos y en las capacidades de análisis de información espacial.

1.2.3 Funciones básicas de un SIG

Un SIG funciona como una base de datos con información geográfica (datos alfanuméricos) que se encuentra asociada por un identificador común a los objetos gráficos de un mapa digital. De esta forma, señalando un objeto se conocen sus atributos e inversamente, así como preguntando por un registro de la base de datos se puede saber su localización en la cartografía.

La razón fundamental para utilizar un SIG es la gestión de información espacial. El sistema permite separar la información en diferentes capas temáticas y las almacena independientemente, permitiendo trabajar con ellas de manera rápida y sencilla, facilitando al profesional la posibilidad de relacionar la información existente a través de la topología de los objetos, con el fin de generar otra nueva que no podríamos obtener de otra forma.

Las principales cuestiones que puede resolver un SIG, ordenadas de menor a mayor complejidad, son:

- Localización: preguntar por las características de un lugar concreto.
- Condición: el cumplimiento o no de unas condiciones impuestas al sistema.
- Tendencia: comparación entre situaciones temporales o espaciales distintas de alguna característica.
- Rutas: cálculo de rutas óptimas entre dos o más puntos.
- Pautas: detección de pautas espaciales.
- Modelos: generación de modelos a partir de fenómenos o actuaciones simuladas.

1.2.4 Limitaciones de un SIG

A las limitaciones que tiene un SIG se refiere el Manual de Caracterización de un GIS con aplicación en el registro de Establecimientos industriales, suelos industriales, y artesanía, 2005 planteando que un SIG es un gran sistema informático cuya implantación en una organización es siempre gradual y costosa. Se requiere siempre la adecuación del sistema al trabajo requerido, mediante programación y recopilación de

los datos necesarios. El costo y calidad de las cartografías digitales disponibles es una limitación ya que las empresas que ofertan cartografías digitales poseen permisos especiales para su comercialización que les permiten imponer los precios.

1.3 Contexto relacionado con “Recursos Hidráulicos”

Debido a los problemas actuales con el uso racional y la contaminación del agua potable, provoca que la humanidad tenga un mayor interés por gestionar fuentes de este preciado recurso, como son embalses, ríos y lagunas. Dentro de las fuentes mencionadas, las más importantes son los embalses ya que son grandes reservas de agua que pueden ser mejor controladas por el hombre que las otras debido a su estructura, que tienden a ser acumulaciones de aguas de forma natural o artificial con un aliviadero o compuertas de escape; de las mismas también se conoce su capacidad y estado actual de llenado.

En Cuba el INRH se creó con el objetivo de que una institución se encargara de la administración y control de todos los objetivos hidráulicos cubanos, para mantener centralizados todos los datos referentes a estos, mantener un control estricto de las zonas más vulnerables a las sequías y ubicar los puntos estratégicos para la construcción de los mismos.

El INRH es la organización que garantiza el agua de toda la isla con fines económicos y sociales. Tienen la responsabilidad de mantener el control estricto de estos objetivos y que se les dé el correcto uso. Para los trabajadores del INRH resulta muy complejo realizar análisis de diferentes fuentes a la vez, porque se encuentran ubicadas en diferentes archivos en los cuales resulta complejo realizar búsquedas y comparar criterios.

Los especialistas del instituto con frecuencia deben informar la situación actual de los diferentes embalses ubicados en el territorio cubano, diariamente se actualizan los datos de cantidad de m³ de agua almacenados en las lagunas, embalses y presas y con cierta frecuencia se hacen ciertos análisis sobre en qué lugar ubicar uno, se determinan las distancias de uno a otro, se hallan caminos así como otros estudios. Todas estas observaciones se guardan en un archivo físico para una posterior consulta. Como el INRH no cuenta con un sistema que les permita realizar consulta de todos los datos en tiempo real se hace complejo tener en un corto período de tiempo la información más actualizada posible de cualquier recurso hidráulico.

Después de indagar sobre la gestión de información de recursos hídricos, se encontraron algunas coincidencias pero solo gestionaban embalses que se utilizaban para la generación de electricidad en hidroeléctricas pero no gestionaban información de los embalses. La única información que se encontró

sobre un software que gestione información de embalses es la que a continuación se expone, pero no se ha logrado obtener más detalle sobre el mismo.

1.3.1 *Concepto de embalse*

Se denomina embalse a la acumulación de agua producida por una construcción en el lecho de un río o arroyo que cierra parcial o totalmente su cauce. En España se emplea ocasionalmente el término pantano con este significado. La construcción del embalse puede ocurrir por causas naturales como, por ejemplo, el derrumbe de una ladera en un tramo estrecho del río o arroyo, la acumulación de placas de hielo o las construcciones hechas por los castores, y por obras construidas por el hombre para tal fin, como son las presas (Gil, 2016).

1.3.2 *Concepto de cuenca hidrográfica*

Es el espacio de territorio delimitado por la línea divisoria de las aguas, conformado por un sistema hídrico que conducen sus aguas a un río principal, lago, mar o zona costera. En la cuenca hidrográfica se encuentran los recursos naturales y la infraestructura creada por las personas, en las cuales se desarrollan sus actividades económicas y sociales, generando a su vez consecuencias favorables y no favorables para el bienestar humano y el medio ambiente (Gil, 2016).

1.4 **Análisis de soluciones existentes**

- **SAIHEbro:** es el Sistema Automático de Información Hidrológica y de Comunicación Fónica de la Cuenca Hidrográfica del Ebro. La implantación de este sistema de información responde a la necesidad de relacionar y agilizar el proceso de toma de decisiones en dos objetivos fundamentales relacionados con la gestión hidráulica de la Cuenca. Estos son previsión y actuación en situaciones de avenida con objeto de reducir, en lo posible, los daños causados por las mismas y la gestión global de los recursos hidráulicos a fin de optimizar su asignación y explotación. Unas de sus principales funcionalidades es que permite conocer datos de los embalses que están dentro de la cuenca del Ebro, como su cota actual, volumen actual, entre otros de relevancia; representándolos en un mapa perteneciente a esa región y generar reportes de los mismos. Además monitorizan los caudales circulantes por los distintos ríos permitiendo la toma de decisiones en la gestión de las aguas reguladas en los embalses. En la gestión de recursos permiten conocer los grados de humedad del suelo y ajustar las demandas, y en la gestión de los riesgos permiten realizar previsiones de caudales circulantes en las próximas horas mediante modelos que convierten la precipitación en escurrimiento. Se monitorizan más de 700.000 ha desde la supervisión de 16 grandes sistemas de riego (Confederación Hidrográfica del Ebro, 2016).

En Cuba no existe un sistema que permita la gestión de recursos hidráulicos desde una visión territorial, es decir, que no permiten mostrar información visual de los embalses, ya que solo tienen en cuenta información estadística que no le es de ayuda a los especialistas a la hora de la toma de decisiones. El estudio de estas soluciones ayudó a identificar un conjunto de atributos y funcionalidades que no deben faltar, pero como no permiten que la gestión de los recursos hidráulicos pueda ser tramitada por los especialistas, se decidió implementar un sistema de información geográfica que permita la gestión de los recursos hidráulicos del INRH.

1.5 Herramientas y Tecnologías a utilizar para dar solución al problema

1.5.1 Plataforma GeneSIG 2.0

La plataforma GeneSIG es una herramienta que permite realizar representaciones y análisis de información referenciada geográficamente (Caballero, 2010). Posee una estructura basada en *plugins*, lo que la convierte en una plataforma con un alto grado de interoperabilidad debido a que permite agregar o quitar componentes de manera sencilla. De igual modo permite separar los servidores de bases de datos y web en dos estaciones de trabajo diferentes, balanceando la carga del sistema, aumentando su disponibilidad y disminuyendo la posibilidad de fallas.

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

Como parte de su organización interna, el sistema cuenta con 3 capas lógicas:

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

Es altamente modular y se emplea en el proyecto donde radica el equipo de desarrollo, donde se encuentra además abundante bibliografía, manuales de uso y personal con vasta experiencia en su uso.

1.5.2 Metodología de desarrollo

Una metodología de desarrollo de software en ingeniería de software es un marco de trabajo usado para estructurar, planificar y controlar el proceso de desarrollo en sistemas informáticos.

Existen metodologías ágiles y robustas. Las metodologías ágiles son apropiadas para guiar proyectos de poco volumen que requieran una rápida implementación. Las metodologías robustas pueden ser empleadas para guiar el proceso de desarrollo de proyectos grandes o pequeños, aunque son más apropiadas para proyectos grandes que por su importancia requieren una fuerte planificación (Ramírez Martín, y otros, 2009).

Teniendo en cuenta lo antes planteado y el sistema que se va a implementar es sencillo en cuanto a su volumen y su implementación se debe realizar en un tiempo corto y limitado, lo más indicado es utilizar una metodología ágil, y dentro de ellas la versión ágil de la metodología *Rational Unified Process* (RUP), denominada *Agile Unified Process* (AUP).

AUP:

- AUP abarca siete flujos de trabajo, cuatro ingenieriles y tres de apoyo: Modelado, Implementación, Prueba, Despliegue, Gestión de configuración, Gestión de proyectos y Ambiente. El Modelado agrupa los tres primeros flujos de RUP (Modelamiento del negocio, Requisitos y Análisis y Diseño).
- Dispone de cuatro fases igual que RUP: Creación, Elaboración, Construcción y Transición.
- La metodología AUP se basa en los principios de simplicidad y agilidad: todo se describe concisamente utilizando poca documentación, no miles de ellas. Además de centrarse en actividades de alto valor, no en todo el proyecto. Se puede usar cualquier conjunto de herramientas para trabajar con ella, siempre que sean las más adecuadas.

Descripción de los Flujos de Trabajo ingenieriles:

- Modelado: flujo de trabajo que tiene el objetivo de entender el negocio de la organización, el dominio del problema que se aborda en el proyecto y determinar una solución viable para resolver el problema.
- Implementación: tiene como objetivo transformar los modelos en código ejecutable y realizar un nivel básico de las pruebas, en particular, la unidad de pruebas.

- Prueba: tiene como objetivo realizar una evaluación objetiva para garantizar la calidad: incluye la búsqueda de defectos, validar que el sistema funciona tal como está establecido y verificar que se cumplan los requisitos.
- Despliegue: su objetivo es el plan para la prestación del sistema y la ejecución de dicho plan para que el sistema quede a disposición de los usuarios finales.

1.5.3 Lenguaje de Modelado

En todas las disciplinas de la Ingeniería se hace evidente la importancia de los modelos ya que describen el aspecto y la conducta de "algo". Ese "algo" puede existir, estar en un estado de desarrollo o estar todavía en un estado de planeación. Para realizar los modelos del sistema propuesto se hará uso del Lenguaje Unificado de Modelado 2.0 (UML, por sus siglas en inglés, *Unified Modeling Language*), porque permite “visualizar, especificar, construir y documentar los artefactos generados en el desarrollo de un sistema” (Jacobson, y otros, 2004). La visualización, especificación, construcción y documentación de los artefactos generados se realiza a través de diagramas, así como las relaciones entre los diferentes componentes y objetos.

Herramienta CASE. *Visual Paradigm for UML 8.0*

Las herramientas CASE (*Computer Aided Software Engineering*, Ingeniería de Software Asistida por Computadora) son diversas aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de software reduciendo el coste de las mismas en términos de tiempo y de dinero. Mejoran la forma en que ocurre el desarrollo y tienen influencia sobre la calidad del resultado final.

Para apoyar el modelado de los diagramas se hará uso de la herramienta CASE *Visual Paradigm for UML*, la cual soporta hasta la versión 2.1 de UML y permite modelar los procesos del negocio, la base de datos y las clases del sistema de manera visual. También soporta patrones de diseño para lograr mejores prácticas y permite la conexión a repositorios como: el CVS (*Concurrent Versions/Versioning System*) y el Subversion.

1.5.4 Sistema gestor de Base de Datos. *PostgreSQL9.3*

Un Sistema Gestor de Base de Datos es un tipo de software muy específico, dedicado a servir de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan, y controlar la entrada y salida de datos de la base de datos, manteniendo la integridad de la misma. La importancia de su uso es que proporcionan a los usuarios la capacidad de almacenar datos en la base de datos, acceder a ellos y actualizarlos.

1.5.5 Servidor de aplicaciones

Apache 2 es un servidor web multiplataforma de código abierto. Es modular (basado en módulos), donde cada módulo ofrece un grupo de funcionalidades específicas al servidor. Es uno de los servidores web más utilizado en Internet, lo que facilita el acceso a la documentación. Provee un alto nivel de seguridad y eficiencia, permitiendo además el uso de una versión local, la cual hace posible que el servidor actúe como servidor y cliente al mismo tiempo, creando así la posibilidad de pre visualizar y probar el código mientras este es desarrollado (Apache Software Foundation, 1997).

1.5.6 Servidor de mapas

MapServer 6 es un entorno de desarrollo o plataforma de código abierto para la publicación de datos espaciales y aplicaciones cartográficas en Internet/Intranet. Permite visualizar, consultar y analizar información geográfica a través de la red mediante la tecnología Internet MapServer. Ofrece la posibilidad de ser utilizado como servidor de mapas de terceros programas y admite múltiples formatos de datos vectoriales, características que hacen de MapServer una herramienta potente (Sphinx, 2014).

1.5.7 Lenguajes de programación del lado del servidor

“Acrónimo recursivo de *Hypertext Preprocessor*, PHP5 es un lenguaje de código abierto muy popular especialmente adecuado para el desarrollo web y que puede ser incrustado en HTML” (php, 2013).

Es un lenguaje del lado del servidor, por lo que su código se ejecuta en el servidor y el cliente solo recibe una página con el código HTML resultante de la ejecución PHP independientemente del código origen en el servidor. Las páginas que se ejecuten en el servidor tendrán acceso a bases de datos, conexiones de red y otras funcionalidades para crear la página resultante.

Es soportado por un grupo de programadores, lo que posibilita que los fallos de funcionamiento se encuentren y reparen rápidamente. Es simple comparado con otros lenguajes, facilitando el proceso de actualización y/o migración de código, además soporta conexiones a una gran cantidad de bases de datos (como: PostgreSQL, MySQL, Oracle, entre otras), y ofrece una solución simple y universal a la paginación dinámica (php, 2013).

1.5.8 Lenguajes de programación del lado del cliente

Usado principalmente en su forma del lado del cliente, JavaScript es un lenguaje interpretado, es decir que su código no se compila, sino se ejecuta a través de un intérprete. Este lenguaje es utilizado mayormente en páginas web. Todos los navegadores modernos interpretan el código JavaScript de las páginas web, el cual tiene integrado para tal propósito una implementación del Modelo de Objetos de Documentos (Mozilla Developer Network, 2005).

ExtJS 3.4

JS Ext o ExtJS es una librería JavaScript que incluye: componentes UI del alto rendimiento y personalizables, modelo de componentes extensibles, un API fácil de usar y licencias OpenSource y comerciales. Permite crear aplicaciones complejas utilizando componentes predefinidos, así como un manejador de layouts, que le posibilita proveer una experiencia consistente sobre cualquier navegador, evitando validar que el código escrito funcione bien en cada uno (Firefox, IE, Safari, etc.). La ventana flotante que provee JS Ext se considera excelente por la forma en la que funciona. Al moverla o redimensionarla solo se dibujan los bordes haciendo que el movimiento sea fluido. Además, brinda los siguientes beneficios:

- Balance entre Cliente – Servidor: La carga de procesamiento se distribuye, permitiendo que el servidor, al tener menor carga, pueda manejar más clientes al mismo tiempo.
- Comunicación asíncrona: En este tipo de aplicación el motor de render puede comunicarse con el servidor sin necesidad de estar sujeta a un clic o una acción del usuario, dándole la libertad de cargar información sin que el cliente lo note.
- Eficiencia de la red: El tráfico de red puede disminuir al permitir que la aplicación elija que información desea transmitir al servidor y viceversa, sin embargo, la aplicación que haga uso de la pre-carga de datos puede que revierta este beneficio por el incremento del tráfico. (Ventajas de ExtJS, 2011)

1.5.9 Entorno de Desarrollo Integrado (IDE). PhpStorm

PhpStorm es un IDE de programación desarrollado por JetBrains. Es uno de los entornos de programación más completos de la actualidad, permite editar código no sólo del lenguaje de programación php como lo indica su nombre. Actualmente es compatible con Sistemas Operativos Windows, Linux y Mac OS X.

- Permite la gestión de proyectos fácilmente.
- Proporciona un fácil autocompletado de código.
- Soporta el trabajo con PHP 5.5
- Sintaxis abreviada.

1.6 Conclusiones parciales

El estudio de las características, componentes y funcionalidades de los Sistemas de Información Geográficas, permitieron determinar un conjunto de herramientas y tecnologías necesarias para el desarrollo de la presente investigación, lo cual condujo a la selección de estos elementos de la manera más adecuada posible. De igual modo, se analizaron las principales ventajas y desventajas de estas tecnologías

para lograr una correcta integración entre estas y proponer una solución viable. El análisis del estado actual permitió determinar las funciones que contribuyen al desarrollo de sistemas de este tipo. Además, se estudiaron las características de la plataforma soberana GeneSIG, lo cual permite que la solución propuesta sea compatible con la misma.

Capítulo #2. Características y diseño del sistema

2.1 Introducción

Para guiar el proceso de desarrollo de la solución informática es necesario comprender su contexto e identificar las condiciones o capacidades que la misma debe tener. En este capítulo se definen las características del negocio a través de los principales conceptos o entidades destacadas. Esto posibilita la definición de los requisitos funcionales y no funcionales, así como modelar la solución propuesta a través de las clases del diseño. El diseño está guiado mediante la utilización de patrones de asignación de responsabilidades y grupo de los cuatros que fundamentan la representación originada. También la estructuración, aplicación y relación de estos elementos se establece mediante la concepción, orientada a objetos y basada en componentes, garantizando una arquitectura de software estable y de un alto grado de robustez. Además, se representa el modelo de datos que será manejado por la aplicación.

2.2 Propuesta de solución

La solución propuesta es un SIG personalizado sobre la plataforma soberana GeneSIG. Esto permite incorporar al sistema y por consecuencia a GeneSIG un grupo de ventajas y funcionalidades sobre la gestión de recursos hidráulicos, dotándolo de la capacidad de realizar análisis sobre los embalses y una mejora en el proceso de toma de decisiones por si ocurre alguna situación de emergencia, tales como: sequías o inundaciones. Como resultado se obtiene un SIG capaz de gestionar, representar, tematizar y brindar reportes de todos los embalses con su capacidad máxima, volumen actual, provincia, municipio al que pertenecen, cuenca de abastecimiento, uso y municipios que abastece cada embalse a nivel nacional. Esto permite obtener la información de una manera sencilla, sin necesidad de utilizar teléfonos o correos electrónicos. Si bien esta solución informática no cumple con la definición completa de un SIG, si constituye una parte usable de estos sistemas para el entorno al que fue desarrollado. Esto permite la integración con otros componentes que aporten los elementos que el sistema actual no presenta, para conformar de manera general un SIG completo.

2.3 Modelo de dominio

El modelado de dominio o conceptual (ver Figura 1) tiene como objetivo comprender y describir los conceptos principales dentro del contexto organizacional y operacional, lo cual permite definir los procedimientos, requisitos y roles más significativos. Esto ayuda a usuarios y desarrolladores a tener un vocabulario común, contribuyendo con el proceso de desarrollo de la aplicación.

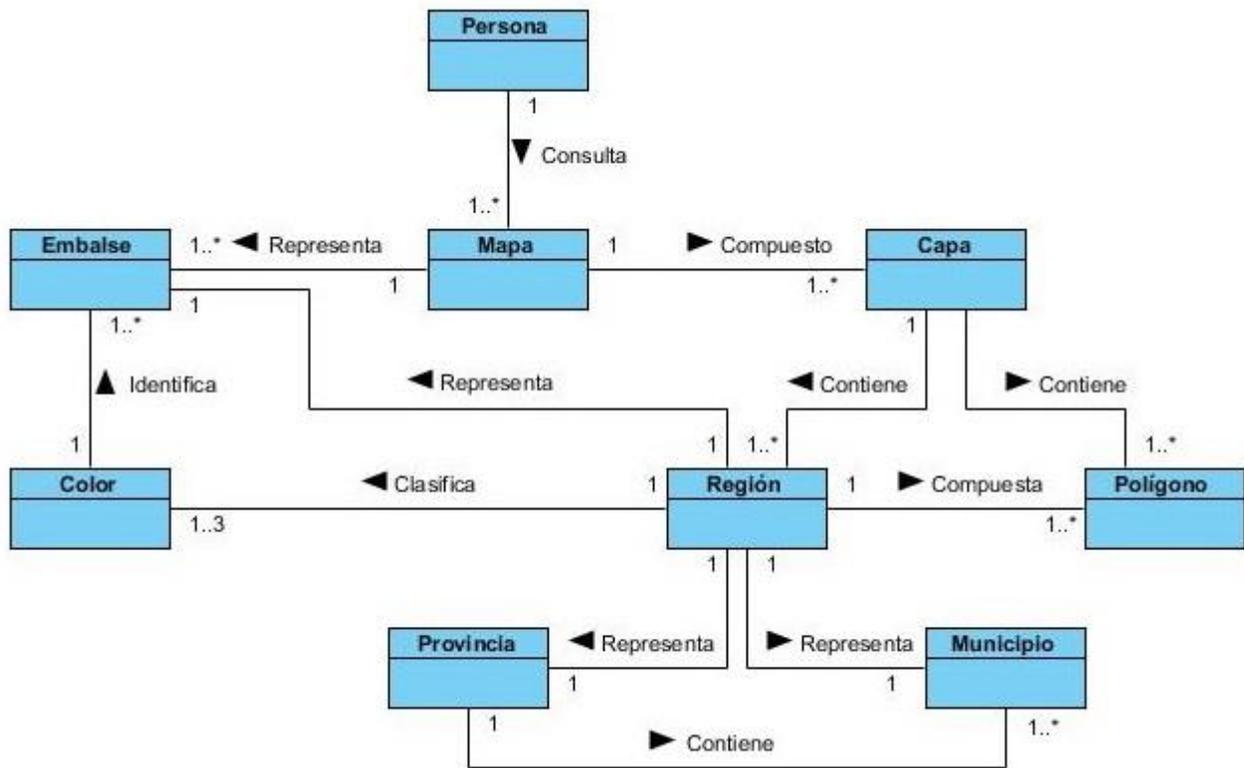


Figura 1. Modelo conceptual del SIG RH.

2.4 Descripción de los conceptos relacionados en el modelo de dominio

Tabla 1. Descripción de los conceptos del dominio del negocio.

Concepto	Descripción
Persona	Es quien consulta la información representada sobre los mapas.
Mapa	Es la entidad que es consultada por una persona, contiene las capas, se representan los embalses, provincias y municipios.
Embalse	Son la entidad fundamental, se representan sobre los mapas y tienen características asociadas como: nombre, capacidad máxima, volumen actual, entre otras.
Capa	Está contenida dentro de los mapas, sobre esta se pueden representar regiones y polígonos.

Polígono	Es una estructura que representa un determinado contorno en una capa.
Región	Es una estructura la cual se puede clasificar en colores y representa las presas, las provincias y los municipios.
Color	Es una característica del estado de los embalses, municipios y provincias.
Provincia	Es una forma de representar una determinada región en un país, que contiene municipios y embalses.
Municipio	Es una forma de representar una determinada región en un país, a menor escala que la provincia, conteniendo también embalses.

2.5 Requisitos del sistema

La Especificación de Requisitos de Software (ERS) es una descripción completa del comportamiento del sistema que se va a desarrollar. Un requisito funcional (RF) puede expresarse de dos formas: de alto nivel o de usuario y de bajo nivel o de sistema. Un requisito de alto y/o bajo nivel puede tener implícitos varios requisitos que responden al más general y por lo tanto, estos también tienen que ser especificados. Además, la ERS también contiene requisitos no funcionales o complementarios (RNF). Los no funcionales son requisitos que imponen restricciones en el diseño o la implementación (como por ejemplo restricciones en el diseño o estándares de calidad).

2.5.1 Requisitos Funcionales

Son capacidades o condiciones que el sistema debe cumplir. Los requisitos funcionales no alteran la funcionalidad del producto, esto quiere decir que los requisitos funcionales se mantienen invariables sin importarle con qué propiedades o cualidades se relacionen. (Jacobson, y otros, 2004).

Tabla 2. Requisitos funcionales del SIG RH.

No.	Nombre	Descripción
RF 1	Adicionar embalse.	El sistema permite insertar un embalse introduciendo la siguiente información: <ul style="list-style-type: none">• Nombre.• Capacidad Máxima.• Volumen Actual.

Capítulo 2. Características y diseño del sistema

		<ul style="list-style-type: none">• Uso.• Provincia a la que pertenece.• Municipio al que pertenece.• Contorno.
RF 2	Modificar embalse.	El sistema permite modificar los datos de los embalses por otros nuevos.
RF 3	Eliminar embalse.	El sistema permite eliminar un embalse con sus datos.
RF 4	Mostrar embalse.	El sistema permite mostrar los datos de un determinado embalse.
RF 5	Buscar embalse.	El sistema permite buscar un determinado embalse poniendo su nombre o parte de este.
RF 6	Limpiar datos de embalse.	El sistema permite limpiar para que no existan conflictos al escoger una nueva funcionalidad.
RF 7	Representar embalse.	El sistema permite representar el contorno de un embalse en el mapa pintando de azul su interior.
RF 8	Representar embalse según su completamiento.	El sistema permite representar el contorno de un embalse en el mapa de color azul, rojo o verde, dependiendo de su completamiento.
RF 9	Representar provincia según estado de los embalses.	El sistema permite representar el contorno de una provincia en el mapa de color azul, rojo o verde, dependiendo del completamiento de los embalses que pertenecen a la misma.

RF 10	Representar municipio según estado de los embalses.	El sistema permite representar el contorno de un municipio en el mapa de color azul, rojo o verde, dependiendo del completamiento de los embalses que pertenecen al mismo.
RF 11	Representar estado de los embalses de la provincia.	El sistema permite representar el contorno de los embalses de una provincia en el mapa con sus estados.
RF 12	Representar estado de los embalses del municipio.	El sistema permite representar el contorno de los embalses de un municipio en el mapa con sus estados.
RF 13	Generar reporte de los embalses.	El sistema permite generar un reporte con las características y los atributos de los embalses.
RF 14	Generar reporte de las provincias.	El sistema permite generar un reporte con el estado en porciento de la provincia.

2.6 Requisitos No Funcionales

Los requisitos no funcionales son propiedades o cualidades que el producto debe tener. Son las características que hacen al producto atractivo, usable, rápido y confiable. Dichos requisitos resultan fundamentales en la evaluación de las características no funcionales del software. Para esto, se tomó como referencia las buenas prácticas que propone AUP-vUCI para la especificación de requisitos no funcionales (RnF) basándose en la norma ISO 25010 (Excriba, 2013). Estas descripciones de los RnF son guiadas por los atributos y sub-atributos de calidad que propone la taxonomía en dicho estándar. Las partes que conforman la taxonomía pueden o no aplicar en dependencia del RnF que se esté evaluando.

A continuación, se presentan las descripciones de los RnF bajo los atributos y sub-atributos de calidad evaluados para la solución según la ISO 25010:

Tabla 3. Descripción del RnF de Usabilidad-Operabilidad del SIG RH.

Atributo de Calidad	Usabilidad.
Sub-atributos/Sub-característica	Operabilidad.
Objetivo	Capacidad del producto que permite al usuario operarlo y controlarlo con facilidad.
Origen	Proveedor de requisitos.
Artefacto	Pantallas pertenecientes al SIG de recursos hidráulicos.
Entorno	El sistema desplegado.
Estímulo	Respuesta: Flujo de eventos (Escenarios)
1. a. Número de pasos a ejecutar para llegar a interactuar con el SIG de recursos hidráulicos.	
Interactuar con las funcionalidades del SIG de recursos hidráulicos.	<ol style="list-style-type: none"> 1. Permite una navegación sencilla, tanto a los usuarios con conocimientos avanzados de informática como a los usuarios más inexpertos, esto se logrará a partir de una estructura de la información correcta. 2. Permite representar sobre el mapa el estado de las provincias con respecto a la intensidad y capacidad de los embalses que componen a estas. 3. Representa sobre el mapa las clasificaciones relacionadas con la capacidad que se le puede otorgar a un embalse (buena, media, crítica).
Medida de respuesta	
Navegar por el SIG de recursos hidráulicos.	

Tabla 4. Descripción del RnF de Usabilidad-Agradabilidad del SIG RH.

Atributo de Calidad	Usabilidad.
Sub-atributos/Sub-característica	Agradabilidad.
Objetivo	Capacidad del producto que permite al usuario interactuar con el sistema mediante interfaces agradables e intuitivas.
Origen	Proveedor de requisitos.
Artefacto	Pantallas pertenecientes al SIG de recursos hidráulicos.
Entorno	El sistema desplegado.
Estímulo	Respuesta: Flujo de eventos (Escenarios)
1. a. Ambiente gráfico de interacción con las funcionalidad del SIG de recursos hidráulicos.	
Interactuar con las funcionalidades del SIG de recursos hidráulicos mediante las interfaces de comunicación que las componen.	<ol style="list-style-type: none"> 1. Se activan las pantallas gráficas que permiten interactuar con las funcionalidades SIG de recursos hidráulicos desarrolladas. 2. Muestra los títulos de los componentes de las interfaces y los mensajes para interactuar con los usuarios, en idioma español y con una apariencia uniforme guiada por los colores gris, blanco, azul claro, mientras que los íconos funcionales se representan de color amarillo claro. 3. Muestra mensajes lo suficientemente informativos para dar a conocer la severidad de los mismos. 4. Dibuja sobre el mapa con los colores azul, verde y rojo el estado de los embalses de una provincia determinada. De igual forma con estos colores se representa también

	sobre el mapa los embalses específicos de acuerdo a su clasificación, azul oscuro (buena capacidad), verde (mediana capacidad) y rojo (capacidad crítica).
Medida de respuesta	
Navegar por el SIG de recursos hidráulicos.	

Tabla 5. Descripción del RnF de Fiabilidad-Disponibilidad del SIG RH.

Atributo de Calidad	Fiabilidad.
Sub-atributos/Sub-característica	Disponibilidad.
Objetivo	Capacidad del producto de estar operativo y accesible para su uso cuando se requiere.
Origen	Proveedor de requisitos.
Artefacto	El SIG de recursos hidráulicos.
Entorno	El sistema desplegado, funcionando en el tiempo deseado.
Estímulo	Respuesta: Flujo de eventos (Escenarios)
1. a. El sistema desplegado en los escenarios especificados por el usuario durante el tiempo deseado.	
El sistema no deberá tener limitante de tiempo y de inactividad. El sistema no guarda el estado antes una falla de energía eléctrica. Si el computador es apagado durante el juego la aplicación se verá interrumpida. El sistema al cambiar la hora del computador cliente no se afectará.	N/A
Medida de respuesta	

Navegar por el SIG de recursos hidráulicos.

Tabla 6. Descripción del RnF de Mantenibilidad- Analizabilidad del SIG RH.

Atributo de Calidad	Mantenibilidad.
Sub-atributos/Sub-característica	Analizabilidad.
Objetivo	Facilidad con la que se puede evaluar el impacto de un determinado cambio sobre el resto del software, diagnosticar las deficiencias o causas de fallos en el software, o identificar las partes a modificar.
Origen	Arquitecto de software.
Artefacto	El código fuente.
Entorno	El ambiente de desarrollo del SIG de recursos hidráulicos.
Estímulo	Respuesta: Flujo de eventos (Escenarios)
1. a. Impacto de un determinado cambio sobre el resto del SIG de recursos hidráulicos.	
Se evalúa el impacto de un cambio en el SIG de recursos hidráulicos.	<ol style="list-style-type: none"> 1. El arquitecto de software identifica los paquetes del diseño y clases implicadas en el cambio, así como las funciones que se reutilizarán o se crearán para introducir el cambio. 2. Se evalúa el impacto partiendo de los resultados que arrojó el análisis anterior con respecto a la arquitectura del SIG de recursos hidráulicos.
1. b. Diagnosticar las deficiencias o causas de fallos en el SIG de recursos hidráulicos.	Respuesta: Flujo de eventos (Escenarios)

Capítulo 2. Características y diseño del sistema

Se diagnostican las deficiencias o causas de fallos en el SIG de recursos hidráulicos.	<ol style="list-style-type: none"> 1. El arquitecto de software identifica las posibles deficiencias o causas de fallos que se pueden originar en del SIG de recursos hidráulicos. 2. Se diagnostican las deficiencias o causas de fallos partiendo de los resultados que arrojó el análisis anterior.
1. c. Identificar las partes a modificar.	Respuesta: Flujo de eventos (Escenarios)
Se identifican las partes a modificar en el SIG de recursos hidráulicos.	1. El arquitecto de software identifica los paquetes y clases implicados en el cambio, así como las funciones que se reutilizarán o se crearán para introducir el cambio.
Medida de respuesta	
Analizar el ambiente de desarrollo del SIG de recursos hidráulicos.	

Tabla 7. Descripción del RnF de Mantenibilidad- Modificabilidad del SIG RH.

Atributo de Calidad	Mantenibilidad.
Sub-atributos/Sub-característica	Modificabilidad.
Objetivo	Capacidad del producto que permite que sea modificado de forma efectiva y eficiente sin introducir defectos o degradar el desempeño.
Origen	Arquitecto de software.
Artefacto	El código fuente.
Entorno	El ambiente de desarrollo del SIG de recursos hidráulicos.
Estímulo	Respuesta: Flujo de eventos (Escenarios)
1. a. Capacidad de modificación del SIG de	

recursos hidráulicos.	
La arquitectura del SIG de recursos hidráulicos está diseñada para brindar facilidades a la hora de introducir modificaciones en el sistema. Esto permite que se puedan introducir cambios o modificaciones, que no afecten el correcto desempeño del resto de las funcionalidades de la solución.	N/A
Medida de respuesta	
Introducir una modificación al SIG de recursos hidráulicos.	

Tabla 8. Descripción del RnF de Adecuación funcional- Integridad funcional del SIG RH.

Atributo de Calidad	Adecuación funcional.
Sub-atributos/Sub-característica	Integridad funcional.
Objetivo	Grado en el que el conjunto de funciones cubre todas las tareas y objetivos del usuario especificados.
Origen	Proveedor de requisitos.
Artefacto	El SIG de recursos hidráulicos.
Entorno	El sistema desplegado.
Estímulo	Respuesta: Flujo de eventos (Escenarios)
1. a. Grado en que el SIG de recursos hidráulicos cubre todas las tareas y objetivos especificados.	
El desarrollo del SIG de recursos hidráulicos está guiado por las necesidades expresadas por parte de los proveedores de requisitos, dándole total cumplimiento a sus especificaciones declaradas e implícitas.	N/A
Medida de respuesta	
Navegar por el SIG de recursos hidráulicos.	

Tabla 9. Descripción del RnF de Eficiencia en el rendimiento- Utilización de recursos del SIG RH.

Atributo de Calidad	Eficiencia en el rendimiento.
Sub-atributos/Sub-característica	Utilización de recursos.
Objetivo	Grado en el que las cantidades y tipos de recursos utilizados por un producto o sistema, al realizar sus funciones cumplen con los requisitos.
Origen	Arquitecto de software.
Artefacto	El SIG de recursos hidráulicos.
Entorno	El sistema desplegado.
Estímulo	Respuesta: Flujo de eventos (Escenarios)
1. a. Características de Hardware	
<p>Para las PCs Clientes:</p> <ul style="list-style-type: none"> • Se requiere tengan tarjeta de red. • Al menos 1 GB de memoria RAM. • Procesador 512 MHz como mínimo. <p>Para los Servidores:</p> <ul style="list-style-type: none"> • Se requiere tarjeta de red. • El Servidor de Mapas debe tener como mínimo 2 GB de RAM. • El Servidor de base de datos debe tener como mínimo 2 GB de RAM. • Procesador 3 GHz como mínimo. 	1. El SIG de recursos hidráulicos funcionando correctamente.
Estímulo	Respuesta: Flujo de eventos (Escenarios)
1. b. Características de Software	
Para las PCs Clientes:	1. El SIG de recursos hidráulicos funcionando correctamente.

<ul style="list-style-type: none"> • Un navegador web. • Sistemas operativos: GNU/Linux y Windows. <p>Para las PCs Servidores:</p> <ul style="list-style-type: none"> • Sistema operativo GNU/Linux Debian 6. • Servidor Web Apache 2.0 o superior, con módulo PHP 5 configurado con la extensión pgsql incluida. • PostgreSQL como Sistema Gestor de Base de Datos. • PostGIS como extensión de PostgreSQL como soporte de datos espaciales. • MapServer 5.2.2 o superior, con extensión PHP MapScript. 	
Estímulo	Respuesta: Flujo de eventos (Escenarios)
1. c. Características del diseño y la implementación.	
<ul style="list-style-type: none"> • Plataforma GeneSIG. • Lenguaje de programación: JAVA, JavaScript y PHP. • Lenguaje de marcado: HTML. • Como gestor de base de datos: PostgreSQL 8.4 y PostGIS. 	<p>1. El SIG de recursos hidráulicos en proceso de desarrollo.</p>
Medida de respuesta	
Despliegue del SIG de recursos hidráulicos.	

2.7 Casos de uso del sistema

Un caso de uso es una secuencia de transacciones que son desarrolladas por un sistema en respuesta a un evento que inicia un actor sobre el propio sistema. Cada caso de uso proporciona uno o más escenarios que indican cómo debería interactuar el sistema con el usuario o con otro sistema para conseguir un objetivo específico.

2.7.1 Diagrama de casos de uso del sistema

Los diagramas de casos de uso son el punto de partida en todo proyecto de desarrollo de software basado en UML y constituyen una representación gráfica de los procesos y su interacción con los actores. Sirven para especificar la funcionalidad y el comportamiento de un sistema mediante su interacción con los usuarios y/u otros sistemas. Constituye una entrada para el análisis, el diseño y las pruebas (Mastermagazine, 2012).

En la figura 2 se muestra el diagrama de casos de uso del sistema propuesto para el desarrollo del sistema. En el que se incluyen todos los casos de uso necesarios para satisfacer los requisitos funcionales especificados.



Figura 2. Diagrama de Casos de Uso del Sistema del SIG RH.

2.7.2 Descripción del actor del sistema

Tabla 10. Descripción del actor del sistema del SIG RH.

No.	Actor	Descripción
-----	-------	-------------

1	Especialista	Es la persona que tendrá acceso al sistema. También podrá gestionar y representar la información referente a los embalses existentes por provincias y municipios en el mapa de Cuba.
---	--------------	--

2.7.3 Patrones de casos de uso del sistema

Entre los patrones de caso de uso utilizados en la solución se encuentra el CRUD (Create, Reading, Update, Delete) Completo (ver Figura 3), que propone formar un caso de uso a partir de los requisitos funcionales relacionados con las acciones de crear, actualizar, guardar, ejecutar, renombrar, abrir, eliminar y modificar una consulta SQL. Haciendo uso de este se conforma el caso de uso “Gestionar embalse”.

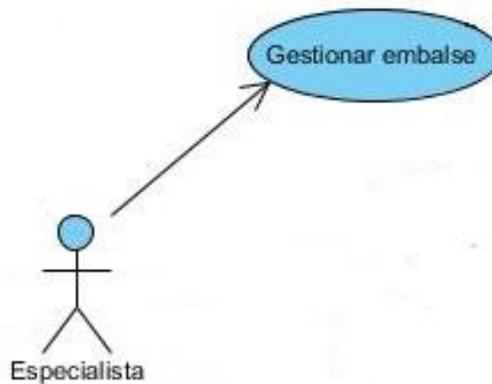


Figura 3. Patrón de CU CRUD Completo presente en el diagrama de CUS del SIG RH.

Otro de los patrones de CU utilizados es el de Extensión o inclusión concreta, en este caso Extensión concreta (ver Figura 4). Este patrón consiste en la representación de un único caso de uso extendido de otro caso de uso base. A continuación, se presenta la evidencia de la utilización de este patrón que involucra a los CU Gestionar embalse y Buscar embalse.

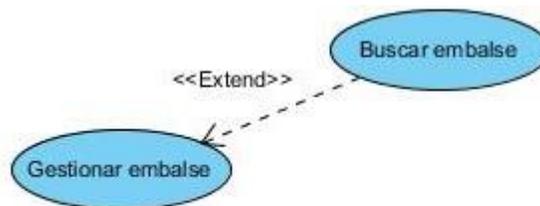


Figura 4. Patrón de CU Extensión concreta presente en el diagrama de CUS del SIG RH.

2.7.4 Especificación de casos de uso del sistema

Tabla 11. Descripción textual del CU Gestionar embalse.

Objetivo	Permitir adicionar, modificar y eliminar los datos de un embalse que corresponde a un municipio o provincia.	
Actores	Especialista.	
Resumen	El caso de uso se inicia por el especialista una vez que decide insertar, modificar o eliminar información relacionada con un embalse a partir de un formulario que contiene los campos correspondientes. Como valor agregado el usuario también tendrá la posibilidad de limpiar los datos introducidos con anterioridad en cualquiera de los escenarios.	
Complejidad	Alta.	
Prioridad	Crítico.	
Precondiciones	Se selecciona de la barra de herramientas el ícono “Gestionar embalse”.	
Postcondiciones	Se registró, actualizó o eliminó información relacionada con un embalse.	
Flujo de eventos		
Flujo básico “Gestionar embalse”		
	Actor	Sistema
1.	Selecciona de la barra de herramientas el ícono “Gestionar embalse”.	
2.		<p>Muestra una ventana para realizar las siguientes operaciones:</p> <ul style="list-style-type: none"> • Adicionar embalse (Ir a la Sección 1. Adicionar embalse). • Modificar embalse (Ir a la Sección 2. Modificar embalse). • Eliminar embalse (Ir a la Sección 3. Eliminar embalse).
Flujos alternos		

2a. Se selecciona la opción de botón “Limpiar”.	
Actor	Sistema
1.	Limpia los campos del formulario: TextField, NumberField y Combobox.
Sección 1: “Adicionar embalse”	
Flujo básico “Adicionar embalse”	
Actor	Sistema
1.	Muestra un formulario con los siguientes campos: <ul style="list-style-type: none"> • Nombre. • Capacidad Máxima. • Volumen actual. • Uso. • Provincia. • Municipio. • Cuenca. • Municipios que abastece. • Ubicación. • Coordenadas.
2.	Introduce los datos requeridos hasta completar el formulario de registro.
3.	Activa el botón “Adicionar”.
4.	Selecciona la opción del botón “Adicionar”.
5.	Valida que los datos introducidos estén correctos.
6.	Verifica que no existan campos vacíos.
7.	Registra el embalse.
8.	Muestra un mensaje indicando: “Se ha adicionado el embalse correctamente”.

9.		<p>Recarga el listado de embalses (grid).</p> <p><i>Terminando así el caso de uso.</i></p>
----	--	--

Prototipo elemental de interfaz gráfica

Flujos alternos

5a. Se introducen campos incorrectos.

	Actor	Sistema
1.		No permite introducir caracteres no admitidos por el campo correspondiente.

6a. Se dejan campos vacíos.

	Actor	Sistema
1.		No se activa la opción del botón "Adicionar".

Sección 2: “Modificar embalse”	
Flujo básico “Modificar embalse”	
Actor	Sistema
1.	Selecciona el embalse deseado con doble clic izquierdo (mediante el componente grid), contenido en el listado de registro de embalses.
2.	Activa la opción del botón “Modificar”.
3.	Carga los datos del embalse seleccionado.
	Muestra los datos del embalse seleccionado en los campos del formulario: <ul style="list-style-type: none"> • Nombre. • Capacidad Máxima. • Volumen actual. • Uso. • Provincia. • Municipio. • Cuenca. • Municipios que abastece. • Ubicación. Coordinadas.
4.	Modifica los datos deseados y selecciona la opción del botón “Modificar”.
5.	Valida que los datos introducidos estén correctos.
6.	Verifica que no existan campos vacíos.
7.	Modifica el embalse.

Capítulo 2. Características y diseño del sistema

8.		Muestra un mensaje indicando: “Se ha modificado el embalse correctamente”.
9.		Recarga el listado de embalses (grid). <i>Terminando así el caso de uso.</i>

Prototipo elemental de interfaz gráfica

GestionarEmbalses

Nombre:

Capacidad Máxima (m3):

Volumen Actual (m3):

Municipios que avastece:

Ubicación

Uso:

Provincia:

Municipio:

Cuenca de Abastecimiento:

Buscar:

<input type="checkbox"/>	Numero Embalse	Capacidad Máxima	Volumen Actual	Calidad	Embalse
<input checked="" type="checkbox"/>	1	65385	8548	water	Laguna de la Leche
<input type="checkbox"/>	2	4665		water	
<input type="checkbox"/>	4	864		water	Bahia Buena Vista
<input type="checkbox"/>	5	4665		water	Bahia de La Gloria
<input type="checkbox"/>	6	4458		water	Bahia de Perros
<input type="checkbox"/>	7	65666564	65656	water	Ensenada Bautista
<input type="checkbox"/>	8	99494949		water	
<input type="checkbox"/>	10	656565		water	

Page of NaN

Displaying 1 - 10 of 3105

Flujos alternos

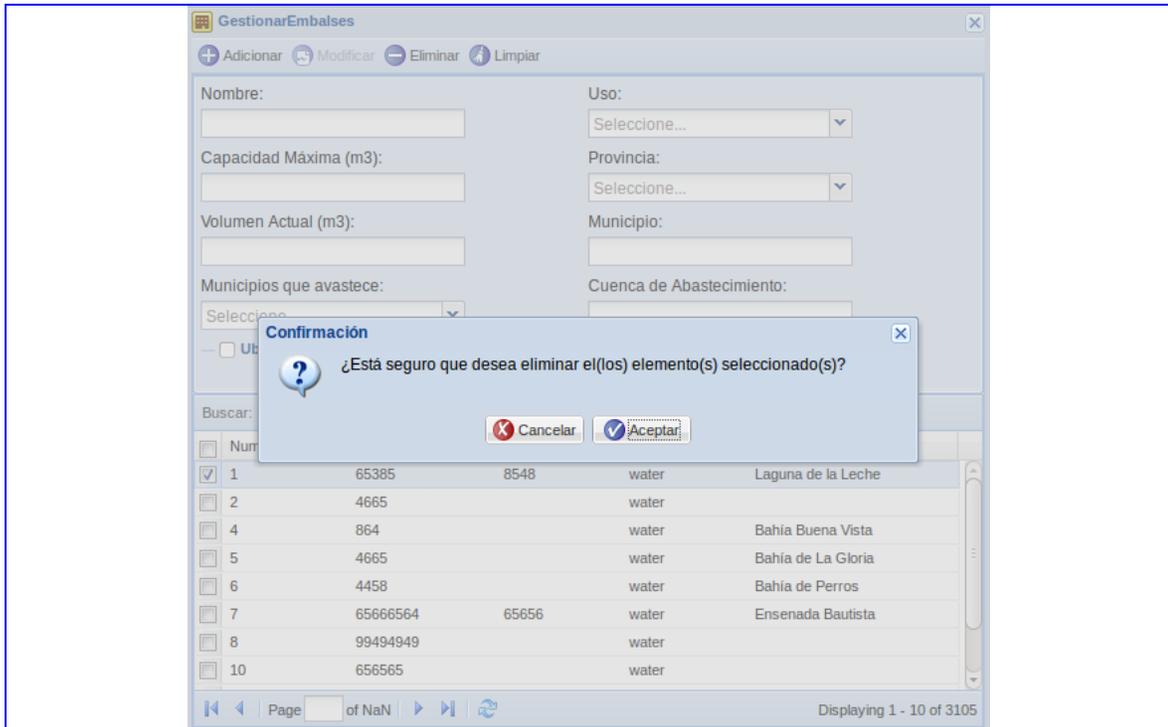
5a. Se introducen campos incorrectos.

	Actor	Sistema
1.		No permite introducir caracteres no admitidos por el campo correspondiente.

Flujos alternos

6a. Se dejan campos vacíos.

Actor		Sistema
1.		No se activa la opción del botón "Modificar".
Sección 3: "Eliminar embalse"		
Flujo básico "Eliminar embalse"		
Actor		Sistema
1.	Selecciona el o los embalse(s) deseado(s) presionando el clic izquierdo (mediante el componente grid), contenido en el listado de registro de embalses.	
2.		Activa la opción del botón "Eliminar".
3.	Selecciona la opción del botón "Eliminar".	
4.		Obtiene el embalse seleccionado.
5.		Muestra un mensaje de confirmación: "Está seguro que desea eliminar el o los embalse(s) seleccionado(s)".
6.	Selecciona del mensaje de confirmación la opción de botón "Aceptar".	
7.		Elimina el embalse.
8.		Muestra un mensaje indicando: "Se ha eliminado el embalse correctamente".
9.		Recarga el listado de embalses (grid). <i>Terminando así el caso de uso.</i>
Prototipo elemental de interfaz gráfica		



Flujos alternos

6a. Se selecciona del mensaje de confirmación la opción del botón “Cancelar”.

Actor		Sistema
1.		Recarga el listado de embalses (grid).
Relaciones	CU incluidos	No Aplica.
	CU extendidos	Buscar embalse.
Requisitos no funcionales	No Aplica.	
Asuntos pendientes	No Aplica.	

2.8 Arquitectura del sistema

“La arquitectura es una vista estructural de alto nivel, que define estilo o combinación de estilos para una solución. Se puede decir que la arquitectura es esencial para éxito o fracaso de un proyecto. Además, la arquitectura de un software es necesaria para comprender el sistema, organizar el desarrollo del mismo, fomentar la reutilización y controlar la evolución del proyecto” (Pressman, 2005).

La Plataforma GeneSIG emplea la arquitectura orientada a objeto y la arquitectura basada en componentes. A continuación, se realiza una breve descripción de estas:

Arquitectura Orientada a Objetos: “nombres alternativos para este estilo han sido Arquitecturas Basadas en Objetos, Abstracción de Datos y Organización Orientada a Objetos. Los componentes del estilo se basan en principios orientados a objetos: encapsulamiento, herencia y polimorfismo. Las interfaces están separadas de las implementaciones. Las representaciones de los datos y las operaciones están encapsuladas en un tipo abstracto de datos u objeto. La comunicación entre los componentes es a través de mensajes” (Reynoso, y otros, 2004).

Arquitectura 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 (de composición)”, abogaba por la utilización de componentes prefabricados sin tener que desarrollarlos de nuevo” (Robaina, 2008).

2.8.1 Estilo Arquitectónico

Un estilo arquitectónico: “Expresa componentes y las relaciones entre estos, con las restricciones de su aplicación y la composición asociada, así como también las reglas para su construcción. Se consideran como un tipo particular de estructura fundamental para un sistema de software, conjuntamente con un método asociado que especifica cómo construirlo incluyendo información acerca de cuándo usar la arquitectura que describe, sus invariantes y especializaciones, así como las consecuencias de su aplicación” (Camacho, y otros, 2004).

Los estilos arquitectónicos son un tipo particular de estructura que definen los posibles patrones a usar en la implementación de la aplicación. Los estilos más conocidos son: Flujo de Datos, Centrado en datos y, Llamada y Retorno que es el que se emplea en este trabajo. Los componentes y las relaciones entre estos se realizan a través de este estilo que: “Permite construir una estructura de programa relativamente fácil de modificar y ajustar. Esta familia de estilos enfatiza la modificabilidad y la escalabilidad. Es el estilo más generalizado en sistemas de 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 objeto y los sistemas jerárquicos en capas” (Reynoso, y otros, 2004).

2.8.2 Patrones de arquitectura

Un patrón se define como una solución probada con éxito que aparece una y otra vez ante determinado tipo de problema en un contexto dado. Los patrones se definen por un nombre, un problema, una solución y las consecuencias de su aplicación. Este define una posible solución correcta para un problema de diseño dentro de un contexto dado, describiendo las cualidades invariantes de todas las soluciones (Pressman, 2006).

A continuación, se realiza una breve descripción de los patrones arquitectónicos utilizados en la solución:

Patrón de arquitectura orientada a objetos: El patrón de arquitectura orientada a objetos define el sistema como un conjunto de objetos que cooperan entre sí en lugar de un conjunto de procedimientos. Los componentes del estilo se basan en principios orientados a objetos: encapsulamiento, herencia y polimorfismo. Son 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 y en cuanto a las restricciones, puede admitirse o no que una interfaz pueda ser implementada por múltiples clases.

Patrón de arquitectura basado en componentes: Las características principales de este patrón son el modularidad, la reusabilidad y la compatibilidad. El mismo describe un acercamiento al diseño de sistemas como un conjunto de componentes que exponen interfaces bien definidas y que colaboran entre sí para resolver problemas. Los componentes son diseñados de forma que puedan ser reutilizados en distintos escenarios en disímiles aplicaciones, aunque algunos componentes son realizados para una tarea específica.

Luego de haber analizado la arquitectura de la plataforma GeneSIG y haber decidido mantener la misma para el sistema a desarrollar, se hace necesario entonces definir los patrones de diseño que se pueden emplear en la implementación de la solución.

2.9 Diseño del Sistema

Siguiendo las buenas prácticas de la metodología AUP-vUCI, la disciplina de Diseño y Arquitectura se especifica detalladamente. Por tanto, a continuación, se presentan algunos aspectos importantes que se tuvieron en cuenta asociados a esta disciplina:

2.9.1 Patrones de diseño

Los patrones de diseño representan un conjunto de estrategias y buenas prácticas que facilitan la creación de un software al describir un problema y su solución. En este trabajo se emplean los siguientes Patrones Generales de Asignación de Responsabilidades (GRASP, por sus siglas en inglés):

Experto: se encarga de asignar una responsabilidad al experto en información, es decir, la clase que cuenta con la información necesaria para cumplir la responsabilidad. En la solución de la aplicación informática se puede evidenciar en la clase **embalseResult**, ya que es la entidad que contiene toda la información (parámetros inmediatos) concerniente a un embalse.

Creador: tiene la responsabilidad de identificar quién debe ser el responsable de la creación de nuevos objetos o clases. La nueva instancia deberá ser creada por la clase que tiene la información necesaria para realizar la creación del objeto, usa directamente las instancias creadas del objeto, o maneja varias instancias de la clase. El uso de este patrón se puede apreciar en el diseño de la solución cuando se le solicita información a la clase **ClientEmbalse** y esta se encarga de crear instancias de la clase **embalseRequest**. De igual forma la clase **ServerEmbalse** crea instancias de la clase **embalseResult**.

Alta Cohesión: se encarga de asignar a las clases responsables que trabajen sobre una misma área de la aplicación sin mucha complejidad. Una alta cohesión caracteriza a las clases con responsabilidades estrechamente relacionadas que no realicen un trabajo enorme. En el diseño de la solución existen las clases clientes y las clases servidoras, las cuales se encargan de realizar cada una su función, esto permite que las clases no estén saturadas y que se garantice a través de la asignación de responsabilidades, que el sistema presente alta cohesión en sus clases.

Bajo Acoplamiento: es el encargado de asignar las responsabilidades de forma que las clases se comuniquen con el menor número de clases que sea posible. En el modelo de diseño se muestra que no existen dependencias fuertes entre las clases, permitiendo así que al producirse una modificación en alguna de estas, se tenga la mínima repercusión posible en el resto de las clases y pueda ser reutilizado el código.

Los patrones de Grupo de los Cuatro (GoF, por sus siglas en inglés) resultan otra clasificación de los patrones de diseño que pueden ser utilizados en una solución informática. En la solución propuesta se pusieron en práctica los siguientes patrones GoF:

Singleton: en el diseño de clases es necesario aplicar la solución de este patrón, que no es más que garantizar el acceso único a una clase mediante una única instancia. Por este medio se puede controlar el acceso a las clases. El patrón Singleton se evidencia al modificar el framework CartoWeb, donde el objetivo del mismo es crear el objeto "mapa" para que no se cree cada vez que se hace un envío en la aplicación.

Command: este patrón permite encapsular las peticiones a través de un objeto, lo que permite realizar operaciones como gestionar las acciones de dicho objeto. Se utiliza para la comunicación a través de las interfaces de usuario, específicamente a través de la clase **AJAXHelper** que es la encargada de comunicar las interfaces con el servidor. Uno de los aspectos más importantes en el sistema son las interfaces gráficas de usuario, ya que el usuario interactúa constantemente con ellas y por eso principalmente se aplica este patrón GoF a la solución.

2.9.2 Diagrama de Clases del diseño

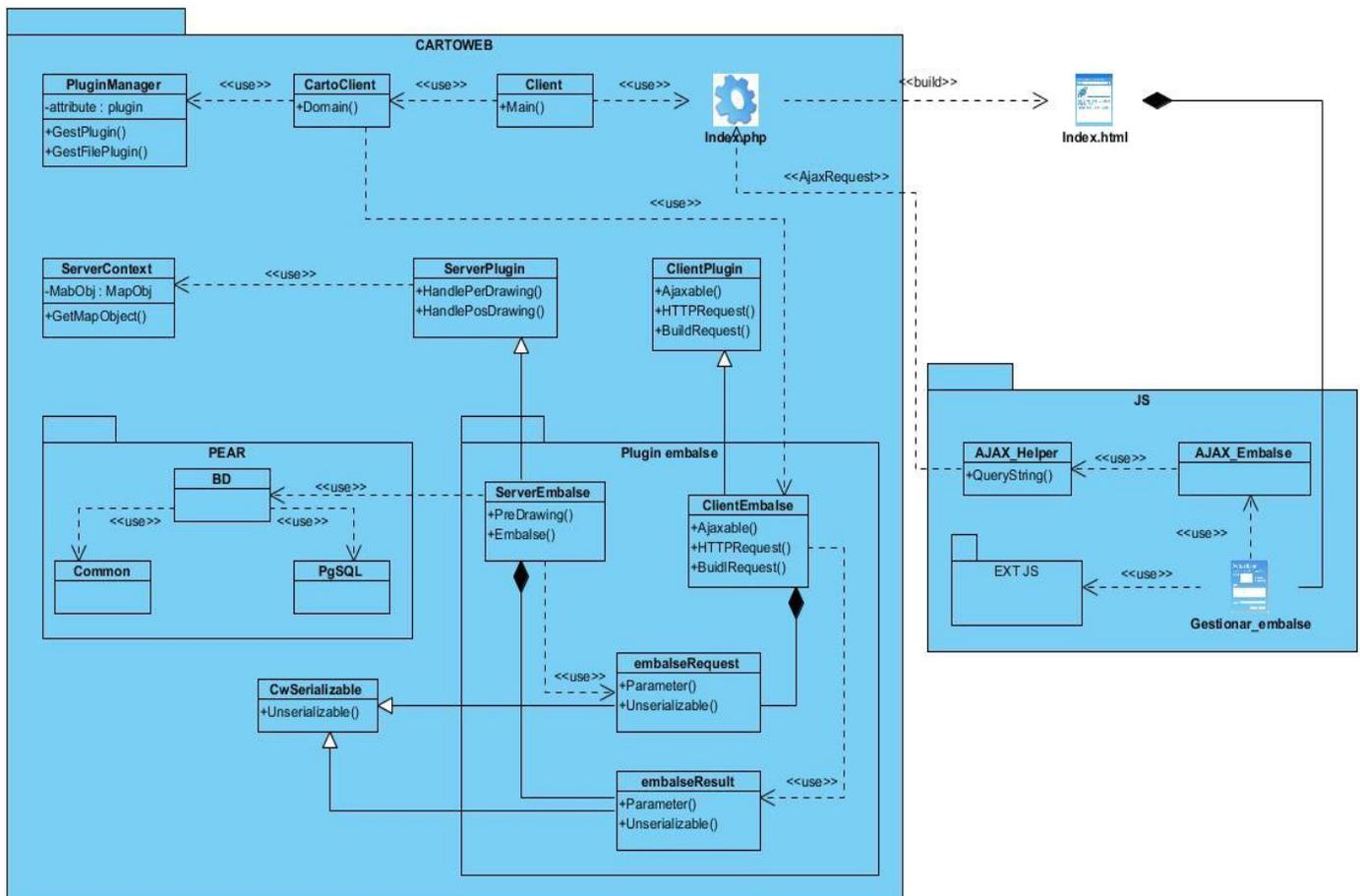


Figura 3. Diagrama de Clases del Diseño del CU Gestionar embalse del SIG RH.

2.9.3 Descripción de las clases del diseño

Cada uno de los diagramas de Clases de diseño que emplea el paquete CartoWeb posee clases que son comunes, así como también las clases que son JavaScript y PHP. Para una mejor comprensión del propósito de cada clase del diseño de la solución, se describen las mismas del Proceso 1. Realizar análisis sobre huracán:

Capítulo 2. Características y diseño del sistema

- **index.php:** tiene como propósito controlar la realización del RF en sí, recibe las peticiones realizadas por el cliente, gestiona las mismas y manda a construir la ClientPage.
- **Client:** contiene todos los archivos específicos de PHP del lado de CartoClient y permite la interacción entre la index.php y la CartoClient.
- **CartoClient:** integra y recoge todos los datos y funciones realizadas por cada una de las .js que intervienen en el RF y se definen una serie de variables globales que van a ser utilizadas por la aplicación.
- **PluginManager:** clase que se utiliza para gestionar la base de plugins.
- **ClientPlugin:** contiene las interfaces necesarias para los plugins del lado del cliente.
- **ServerPlugin:** esta clase proporciona la base de herramientas para el desarrollo de plugins.
- **ServerContex:** es la contenedora de la información común que ha de ser utilizada por la parte cliente y la servidora, empleando la información seleccionada como un objeto para un fácil manejo de los datos.
- **BD:** es la clase encargada de establecer la conexión con el servidor de base de datos para procesar los objetos a editar.
- **Common:** es la encargada de administrar las conexiones a la base de datos para ejecutar las consultas a la misma satisfactoriamente, esto incluye tratamiento de los datos.
- **PgsqI:** gestiona desde PHP las funciones de PostgreSQL.
- **CwSerializable:** se encarga de serializar todas aquellas clases que pueden ser serializadas, permitiendo la comunicación entre el Client y el Server del plugin.
- **AJAX_Helper:** tiene como propósito enviar las respuestas de los plugins “AJAX”, para alimentar a los plugins que responden a las peticiones del usuario.
- **Index.html:** es la encargada de mostrar en el mapa la región localizada.
- **AJAX_Embalse:** es la encargada de gestionar el pedido y respuesta a las peticiones del usuario por Ajax.
- **ServerEmbalse:** es la clase servidora que tiene como principal función la conexión con la base de datos y enviar las respuestas necesarias al **ClientEmbalse**.

- **ClientEmbalse:** se encarga de recoger y seleccionar de las .js contenidas en el paquete JS, toda la información correspondiente a los datos a precisar, entrados a través de los formularios, y los envía al **ServerEmbalse**.
- **EmbalseRequest:** es una clase común encargada de transportar los datos recogidos en **ClientEmbalse** desde la interfaz y transportarlos a la clase **ServerEmbalse**.
- **EmbalseResult:** es una clase común encargada de transportar los datos generados en **ServerEmbalse** a la clase **ClientEmbalse**.

2.10 Diseño de la Base de Datos

El diseño de la BD tiene como principal propósito asegurarse de que los datos persistentes son almacenados y consistentes. Además, se encarga de definir el comportamiento que debe ser implementado en la base de datos.

Para definir las clases persistentes se tuvieron en cuenta los conceptos identificados en el dominio del negocio que iban a persistir en el tiempo, pero esto no quiere decir que todos los definidos sean transformados en clases de este tipo. Demostrando esto que la persistencia es la capacidad de un objeto de mantener su valor en el espacio y en el tiempo.

A continuación, se presenta el diagrama de clases persistentes modelado para la solución:

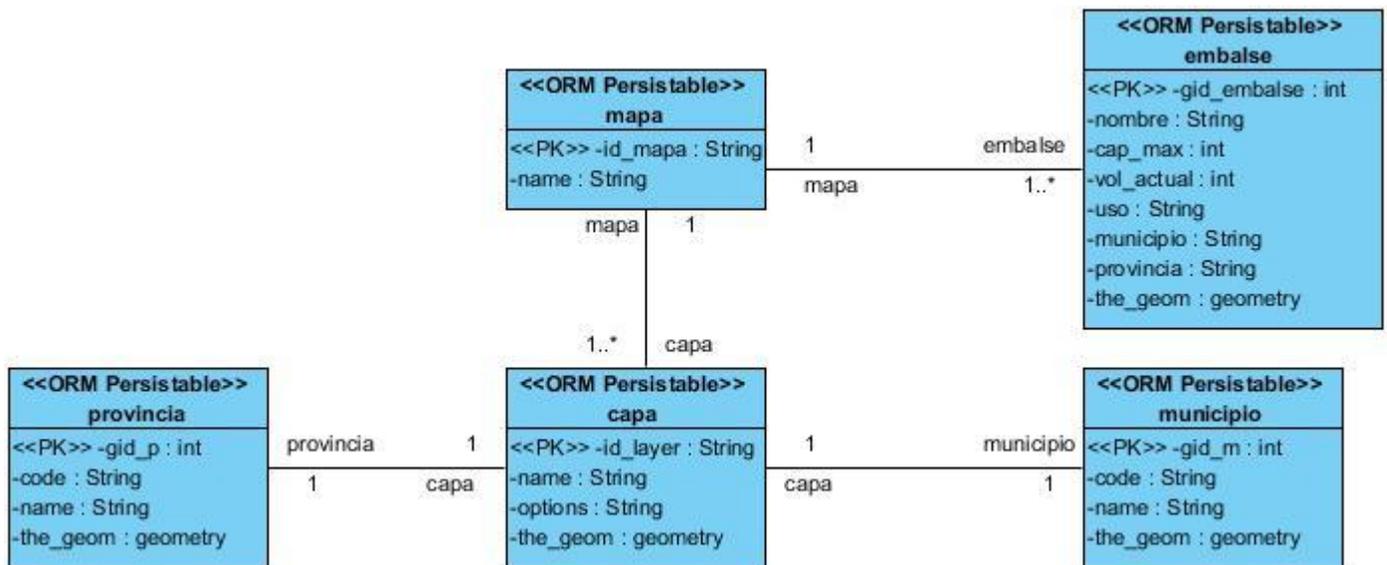


Figura 4. Diagrama de Clases Persistentes.

A partir del diagrama de clases persistentes anteriormente obtenido se generó el siguiente modelo físico de la BD:

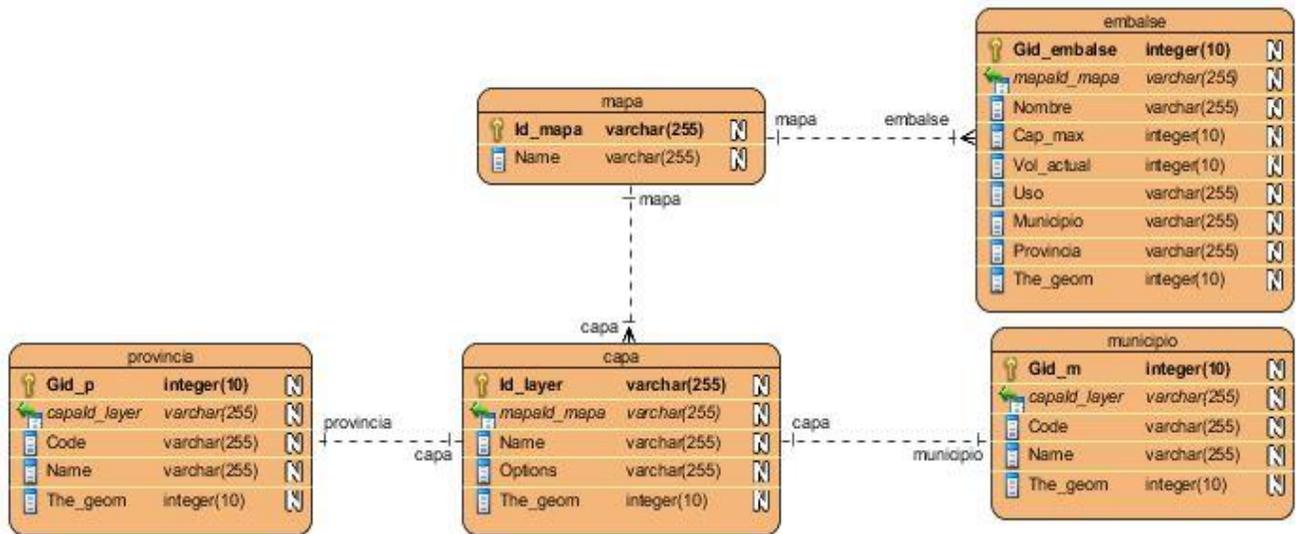


Figura 5. Modelo físico de la base de datos.

2.11 Conclusiones parciales

En este capítulo se realiza el análisis del negocio mediante conceptos utilizando el modelo de dominio como técnica de esta disciplina. Se especifican los requisitos funcionales y no funcionales para tener un mayor grado de detalle sobre las características del sistema. Se detallan las categorías de los requisitos del software posibilitando un mejor entendimiento sobre las características a implementar y una línea base arquitectónica robusta. Se realizó el diagrama de clases del diseño exponiendo los diferentes patrones de asignación de responsabilidades utilizados en la solución. También se diseñó el modelo de datos respondiendo a las exigencias y a las clases persistentes que posee la solución propuesta.

Capítulo #3. Implementación y prueba del sistema

3.1 Introducción

Una vez presentados los elementos referentes al diseño del sistema, en esta sección se tratará lo relacionado a la implementación y pruebas del mismo. Se hará énfasis, como en el capítulo anterior, a la representación de los componentes que forman la parte programable y reemplazable de la solución informática. Se presentará la distribución física de la solución propuesta para lograr una adecuada, estructurada y organizada ejecución de la misma. Además, se estarán generando las pruebas aplicadas a la solución informática mediante el método de prueba, caja negra, utilizando la técnica de partición equivalente, así como los resultados que demuestran el grado de satisfacción de la aplicación informática desarrollada.

3.2 Estilo de programación

Los estilos de programación definen la estructura y apariencia física del código, lo que facilita su comprensión, mantenimiento y lectura. En la implementación se utilizaron diferentes estilos que se describen a continuación:

3.2.1 Definición de clases

El nombre de las clases está escrito en inglés-español y su llave de apertura se encuentra en la próxima línea del nombre (ver Figura 6).

```
<?php
class ClientGestionarEmbalses extends ClientPlugin
implements ServerCaller, Ajaxable, GuiProvider
{
    protected $datos;
    protected $pluginAction;
    protected $response;

    public function __construct(){
    }

    public function handleHttpRequest($request)
    {
        if(isset ($request['action']))
        {
            switch ($request['action'])
            {
                default:
                    $this->pluginAction= $request['action'];
                    $this->datos = isset($request) ? $request : '';
                    break;
            }
        }
    }
}
```

Figura 6. Estructura de las clases.

3.2.2 Estilo de métodos

Los métodos tendrán letra inicial minúscula, en caso de ser dos palabras se mantendrán unidas y la segunda comenzará con mayúscula; seguidamente se colocan dos puntos entre espacios y la palabra reservada **function**. En caso de pasarle elementos por parámetros, estos se escribirán con minúscula (ver Figura 7).

```
public function handleHttpRequest($request){
    $this->handleHttpRequest($request);
}

public function buildRequest(){
    $queryRequest = new GestionarEmbalsesRequest();
    $queryRequest->pluginAction = $this->pluginAction;
    $queryRequest->datos = $this->datos;
    return $queryRequest;
}

public function initializeResult($result){
    if ($result == null) {
        return null;
    }
    $this->response = $result;
}

public function handleResult($result){
    $this->response = json_encode($this->result);
}
```

Figura 7. Estructura de los métodos.

3.2.3 Declaración de variables

Las variables se escribirán con minúscula y estarán anteceditas de la palabra reservada **var**. El signo igual (=) estará ubicado entre espacios (ver Figura 8).

```
gestionarEmbalsesUI = Ext.extend(Ext.CRUDWindow, {

    constructor: function (config) {
        var _this = this;
        var cmps = Genesis.Componentes;
        _this.toolbar = [];
        _this.isCoord = false;
        _this.arreglo = [];
    }
});
```

Figura 8. Estructura de una variable.

3.2.4 Estructuras de control

Dentro de las estructuras de control se pueden encontrar if, for, while, do while, entre otras. Se utilizaron, fundamentalmente, las condicionales, *if*.

```
var _this = AjaxPlugins.TemaEmbalses;
var data = Ext.decode(pluginOutput.variables.response);

if (data) {
  if(data.response[0]=="El embalse esta Bien"){
    _this.drawEmbalsesBien(data.response[1]);
  }
  else if(data.response[0]=="El embalse esta Mediano"){
    _this.drawEmbalsesMediano(data.response[1]);
  }
  else if(data.response[0]=="El embalse esta Critico"){
    _this.drawEmbalsesCritico(data.response[1]);
  }
  // _this.ventana.collapse();
}
},
}
```

Figura 9. Estructura de un if.

3.3 Diagrama de componentes

Este tipo de diagrama representa la parte modular, desplegable y reemplazable de un sistema que encapsula implementación. También expone un conjunto de interfaces que funcionan como elementos intermediarios de comunicación entre componentes que posibilitan la independencia funcional.

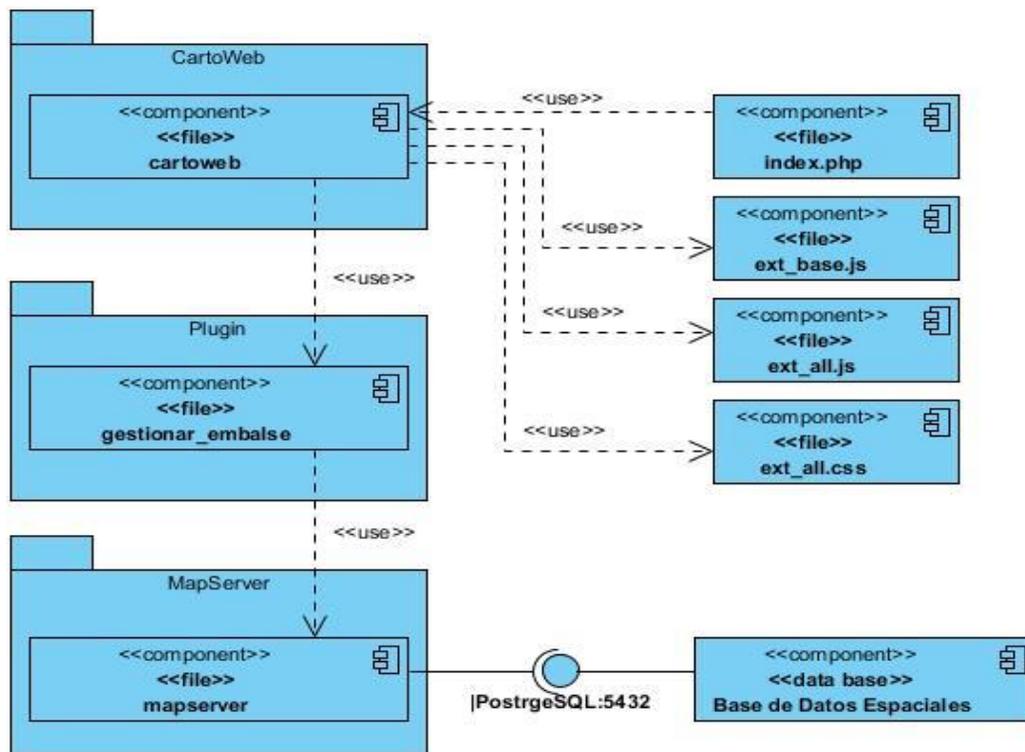


Figura 10. Diagrama de componentes de CU Gestionar embalse.

3.4 Diagrama de despliegue

Un diseño del despliegue del sistema permite establecer una correspondencia entre la arquitectura de software y hardware a utilizar, modelando la disposición física de los componentes que integran al sistema, también mostrando las conexiones físicas entre el hardware y los respectivos componentes. A continuación se presenta el diagrama de despliegue de la solución:

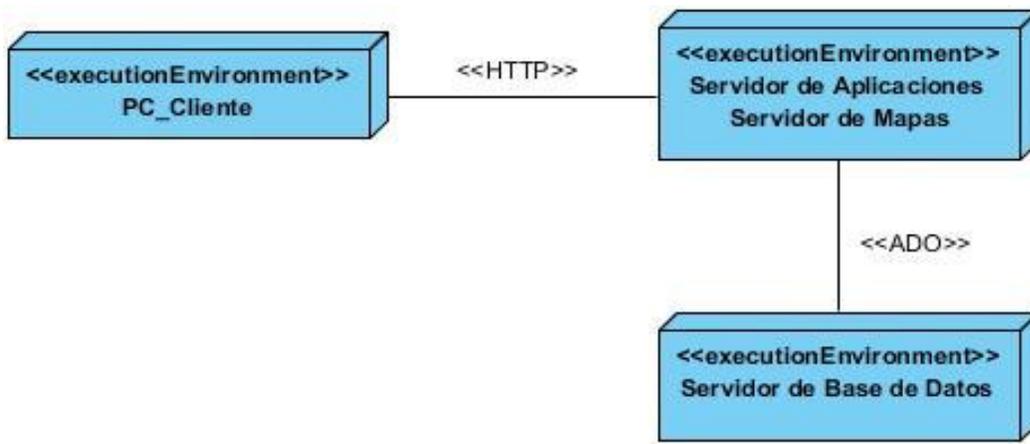


Figura 11. Diagrama de despliegue del SIG RH.

3.5 Pruebas

Las pruebas representan actividades claves que ayudan a entregar el producto con la calidad requerida para satisfacer las necesidades del cliente, cumplir con sus expectativas y con la certeza de que el producto cumple las especificaciones definidas.

Las pruebas de caja negra también denominadas pruebas de comportamiento, se centran en verificar el cumplimiento de los requisitos funcionales del software, se emplean cuando se conoce la función específica para la que se diseñó la solución y se aplican a la interfaz del software, permitiendo obtener conjuntos de condiciones de entrada que ejerciten completamente todos los requisitos funcionales del programa (Pressman, 2005), con el fin de encontrar la mayor cantidad de no conformidades existentes en el producto.

Dentro de las técnicas empleadas por las pruebas de caja negra se utilizó, partición de equivalencia, que según define Pressman: “se dirige a la definición de casos de prueba que descubran clases de errores, reduciendo así el número total de casos de prueba que hay que desarrollar” (Pressman, 2005). Con la aplicación de estas técnicas se demuestra el empleo de las pruebas de tipo Funcionalidad, las cuales fijaron su atención en la validación de las funciones, métodos y casos de uso presentes en la solución.

3.5.1 Diseño de Casos de Prueba

Un diseño de caso de prueba (DCP) está compuesto por un conjunto de entradas, respuesta que emite el sistema de acuerdo a esas entradas y el flujo central que indica el camino del escenario descrito. Estos son desarrollados para verificar el cumplimiento total o parcial de un requisito. Las entradas representan las variables que se pueden especificar y las mismas contienen: V, I, o N/A. V indica válido, I indica inválido, y N/A que no es necesario proporcionar un valor del dato en este caso, ya que es irrelevante. A continuación, se presenta en las Tablas 11, 12 y 13, un ejemplo de los DCP realizados para comprobar el funcionamiento de la solución, específicamente para el caso de uso del sistema Gestionar embalse:

 **DCP 1.** Gestionar embalse:

Tabla 12. DCP del CU Gestionar embalse, SC1 Adicionar embalse del SIG RH.

Nombre de la sección (SC)	Escenarios (EC) de la sección	Nombre	Cap_Max	Vol_Actual	Uso	Provincia	Municipio	Ubicación	Muncp. abast	Respuesta del Sistema	Flujo Central
SC1:Adicionar embalse	EC1.1: Adicionar embalse correctamente.	V	V	V	V	V	V	V	V	Muestra un mensaje de confirmación: "Embalse almacenado satisfactoriamente" y se adiciona a la lista de embalse el nuevo embalse.	Selecciona de la barra de herramientas el ícono "Gestionar embalse", posteriormente se muestra un formulario para que se introduzcan los datos y se presiona el botón 'Adicionar'.
	EC1.2: Campos vacíos.	I	V	V	V	V	V	V	V	Notifica mediante un mensaje que hay campos vacíos: "Existen	Se comienza a introducir los campos dejando alguno o algunos vacíos.
	Nulo	5425	3655	consumo	Artemisa	Caimito	5 5,4 5,5 5	Caimito			
		V	I	V	V	V	V	V	V		

Capítulo 3. Implementación y prueba del sistema

		Coronela	nulo	3655	consumo	Artemisa	Caimito	5,5,4,5,5,5	Caimito	campos vacíos”.	
		Coronela	5425	nulo	Consum	Artemisa	nulo	nulo	Caimito		
EC1.3: Cerrar Ventana		N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	Oculto la ventana.	Selecciona de la barra de herramientas el ícono “Gestionar embalse”, y luego de aparecer la ventana la cierra.
EC1.4: Limpiar datos.		N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	Limpiar los datos introducidos en los campos del formulario.	Selecciona del formulario Gestionar embalse la opción del botón “Limpiar”.

Tabla 13. DCP del CU Gestionar embalse, SC2 Modificar embalse del SIG RH.

Nombre de la sección	Escenarios de la sección	Nombre	Cap_Max	Vol_Actual	Uso	Provincia	Municipio	Ubicación	Muncp. abast	Respuesta del Sistema	Flujo Central
SC2:Modific ar embalse	EC2.1:	V	V	V	V	V	V	V	V	Muestra un mensaje “Embalse	Selecciona de la barra de herramientas el ícono “Gestionar

Capítulo 3. Implementación y prueba del sistema

Modificar embalse correctamente.										se ha modificado correctamente” y se actualizan los datos del embalse señalado.	embalse”, posteriormente se muestra un formulario con todos los embalses, se selecciona el que se va a modificar, se modifican los valores deseados y se presiona el botón ‘Modificar’.	
EC2.2: Campos vacíos.	I	V	V	V	V	V	V	V	V	Notifica mediante un mensaje que hay campos vacíos: “Existen campos vacíos”.	El usuario comienza a introducir los campos a modificar dejando alguno o algunos vacíos.	
		5425	3655	consumo	Artemisa	Caimito	5	5,4	5,5			Caimito
	V	I	V	V	V	V	V	V	V			
	Coronela		3655	consumo	Artemisa	Caimito	5	5,4	5,5			Caimito
	V	V	I	V	V	V	I	I	I			
Coronela	5425		Consum	Artemisa					Caimito			
EC2.3: Cerrar Ventana.	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	Oculto la ventana.	Selecciona de la barra de herramientas el	

Capítulo 3. Implementación y prueba del sistema

											ícono “Gestionar embalse”, y luego de aparecer la ventana selecciona el icono en forma de cruz que se encuentra en la parte superior derecha para cerrarla.	
	EC2.4 Limpiar datos.	N/A	Limpia los datos introducidos en los campos del formulario.	Seleccionar del formulario Gestionar embalse la opción del botón “Limpiar”.								
		-	-	-	-	-	-	-	-	-		

Tabla 14. DCP del CU Gestionar embalse, SC3 Eliminar embalse del SIG RH.

Tabla 15. Variables del DCP del CU Gestionar embalse del SIG RH.

No.	Variable	Valor Nulo	Descripción
1	Nombre	No	Es un TextField que permite introducir el nombre del embalse.
2	Capacidad Máxima	No	Es un NumberField que permite introducir la capacidad máxima del embalse.
3	Volumen Actual	No	Es un NumberField que permite introducir el volumen actual del embalse.
4	Uso	No	Es un ComboBox que permite escoger el uso del embalse.
5	Provincia	No	Es un ComboBox que permite escoger la provincia en que se encuentra embalse.

Capítulo 3. Implementación y prueba del sistema

Nombre de la sección	Escenarios de la sección	Nombre	Cap_Max	Vol_Actual	Uso	Provincia	Municipio	Ubicación	Embalse	Muncp. abast	Respuesta del Sistema	Flujo Central
SC3: Eliminar Embalse	EC3.1: Eliminar embalse correctamente	N/A	N/A	N/A	N/A	N/A	N/A	N/A	V	N/A	Muestra un mensaje "Embalse se ha eliminado correctamente" y se actualizan la tabla de embalse.	Selecciona de la barra de herramientas el ícono "Gestionar embalse", posteriormente se muestra un formulario para que se seleccione el embalse a eliminar y posteriormente se presiona el botón 'Eliminar'.
		-	-	-	-	-	-	-	-	-		
	EC3.2: Cerrar Ventana.	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	Oculto la ventana.	Selecciona de la barra de herramientas el ícono "Gestionar embalse", y luego de aparecer la ventana la cierra.
6	Municipio	No					Es un ComboBox que permite escoger el municipio en que se encuentra embalse.					
7	Cuenca de Abastecimiento	No					Es un TextField que permite introducir el nombre de la cuenca que abastece al embalse.					

8	Embalse	No	Selección única del listado de todos los embalse en el caso del modificar y múltiple en el caso del eliminar.
---	---------	----	---

3.5.2 Resultados de las pruebas realizadas

Una vez diseñados los casos de prueba para cada caso de uso, se procedió a la ejecución de las pruebas. Durante la disciplina de Pruebas para comprobar y evaluar las funcionalidades descritas se ejecutaron 3 iteraciones en total. En una primera iteración se ejecutó un total de 9 casos de prueba en correspondencia con la cantidad de casos de uso de la solución, lo que conllevó a la detección de 6 no conformidades que quedaron pendientes para una próxima iteración. En la segunda iteración se aplican las pruebas de regresión sobre los casos de pruebas afectados en la iteración anterior, lo que arrojó 3 no conformidades relacionadas con los 6 casos de pruebas ejecutados. Sin embargo, en una tercera iteración se ejecutaron 3 casos de prueba sobre los cuales incidían los errores detectados en la segunda iteración, quedando en esta última la solución, las no conformidades corregidas.

Los resultados se pueden observar en la Figura 12, donde se muestran la cantidad de casos de prueba usados y las no conformidades encontradas, las cuales fueron corregidas. En total fueron realizadas tres iteraciones, luego de las cuales no se encontraron no conformidades, cumpliéndose con el funcionamiento correcto de los casos de uso del sistema.

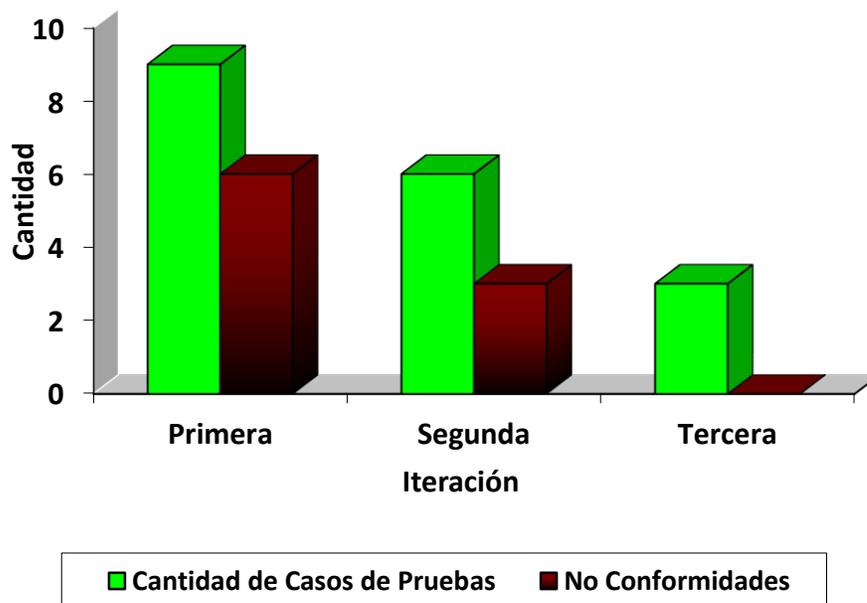


Figura 12. Pruebas de Caja Negra. Cantidad de Casos Prueba y No Conformidades por cada iteración.

3.6 Conclusiones parciales

Se explicó el estilo de programación utilizado, declaración de clases, variables y métodos, para lograr un mejor entendimiento del código resultante. Se realizaron los diagramas de despliegue y componentes para lograr una representación abstracta de la distribución física del sistema, así como de los elementos que encapsulan la implementación. Se diseñaron los casos de prueba y se validó la solución propuesta analizando los resultados generados en los DCP, lo que permitió obtener una solución informática viable para su utilización.

Conclusiones generales

Una vez culminada la investigación es posible afirmar que se dio cumplimiento al objetivo general trazado, por lo que se concluye que:

- A través del estudio del estado del arte se pudieron determinar los métodos, herramientas y tecnologías factibles para desarrollar un SIG personalizado sobre la plataforma GeneSIG, al igual que la necesidad y factibilidad de desarrollar la investigación relacionada con recursos hidráulicos.
- La elección de AUP en su versión UCI como metodología de desarrollo de software permitió guiar y documentar el ciclo de vida de la aplicación, permitiendo que esta cumpliera con los estándares y estilos definidos en las líneas bases de GeneSIG.
- La utilización de las tecnologías y herramientas definidas en las líneas bases de GeneSIG permitió que la solución se integrara a esta plataforma sin provocar incompatibilidades.
- La realización de las pruebas mediante la técnica de partición de equivalencia permitió detectar los posibles errores, para así dar cumplimiento a los requisitos funcionales y no funcionales establecidos y obtener un producto con la calidad adecuada.

Recomendaciones

Una vez culminada la investigación y teniendo en cuenta las experiencias obtenidas en el desarrollo de la misma, se proponen las siguientes recomendaciones:

- Realizar un estudio de factibilidad para la inclusión de módulos que permitan otros análisis sobre los embalses para determinar el nivel de degradación de sus suelos.
- Inclusión en la aplicación del análisis que permitan realizar el estado material de las cortinas que presenta cada embalse.
- Utilizar la propuesta presentada como base para el desarrollo de una personalización del aplicativo GeneSIG para el instituto nacional de recursos Hidráulicos, por parte del equipo del aplicativo SIG.

Bibliografía Referenciada

Apache Software Foundation. 1997. Apache. [En línea] 1997. [Citado el: 22 de 11 de 2015.] <http://httpd.apache.org>.

Bosque-Sendra, J. 1992. *Sistemas de Información Geográfica*. Madrid : Rialp, 1992.

Braken, I. 1992. *Information technology in geography and planning*. Londres & New York : Routledge, 1992.

Caballero, Varen. 2010. *Estrategia para la implementación de Sistemas de Información Geográfica del petróleo sobre la base de la Plataforma GeneSIG. TRABAJO DE DIPLOMA PARA OPTAR POR EL TÍTULO DE INGENIERO EN CIENCIAS INFORMÁTICAS*. La Habana : s.n., 2010.

Camacho, E, Nuñez, F y Cardeso, G. 2004. *Arquitecturas de Software, Guías de Estudio*. 2004.

Castro, Fidel. 1986. *Palabras en el XI Período de Sesiones de la Asamblea Nacional del Poder Popular*. La Habana : s.n., 1986.

Confederación Hidrográfica del Ebro. 2016. SAIHEbro. [En línea] 8 de 4 de 2016. <http://www.saihebro.com>.

ESRI. 1995. ESRI (Environmental Systems Research Institute). What is GIS? [En línea] 1995. <http://www.gis.com/content/what-gis..>

Fallas, Jorge. 2011. *SISTEMAS DE INFORMACIÓN GEOGRÁFICA. SISTEMAS DE INFORMACIÓN GEOGRÁFICA. SISTEMAS DE INFORMACIÓN GEOGRÁFICA. SISTEMAS DE SISTEMAS DE INFORMACIÓN GEOGRÁFICA*. 2011.

Gil, Leal Yosmary. 2016. *Principales indicadores y datos de Infraestructura Hidráulica. Instituto Nacional de Recursos Hidráulicos*. 2016.

Jacobson, Booch y Rumbaugh. 2004. *El Proceso Unificado de Desarrollo. Vol I, Vol II*. La Habana: Félix Varela : s.n., 2004.

Mateus Pulido, Pablo Andrés. 2009. *Registro y gestión patrimonial de yacimientos arqueológicos bajo*. 2009.

Mozilla Developer Network. 2005. MDN. [En línea] 2005. [Citado el: 20 de 1 de 2016.] <https://developer.mozilla.org/es/docs/JavaScript>.

NCGIA. 1990. *National Center for Geographical Information and Analysis*. Santa Bárbara : Universidad de California, 1990.

Olaya, Victor. 2011. *Sistemas de Información Geográfica*. 2011.

php. 2013. php. [En línea] 2013. [Citado el: 3 de 2 de 2016.] php. (2013). php. Recuperado el 8 de noviembre de 2015, de <http://php.net/manual/es/intro-what-is.php>.

Pressman, R. 2005. *Ingeniería de Software*. 2005.

—. 2006. *Ingeniería de Software. Un enfoque práctico. 5ta Edición.* 2006.

Ramírez Martín y Rodríguez Donatien. 2009. *Sistema para la Identificación de Aguas en Pozos Petroleros (SIAPP).* La Habana: Universidad de las Ciencias Informáticas : s.n., 2009.

Reynoso, C y Kiccillof, N. 2004. *Estilos y Patrones en la Estrategia de Arquitectura.* Buenos Aires : Universidad de Buenos Aires, 2004.

Robaina, I. 2008. *Propuesta del Diseño Arquitectónico del Simulador de Sistemas Biológicos.* La Habana : s.n., 2008.

Sphinx. 2014. MapServer. [En línea] 2014. [Citado el: 17 de 1 de 2016.]
<http://www.mapserver.org/es/about.html#about>.

Zayas. 1997. *Metodología de la Investigación Científica.* . 1997.