

Tipo de artículo: Artículo original  
Temática: Desarrollo de aplicaciones informáticas  
Recibido: 13/03/2014 | Aceptado: 1/07/2014

## GeReport: Sistema de Gestión de Reportes Dinámicos

### *GeReport: Management System of Dynamic Reports*

Rafael Felipe Bomate Gavio<sup>1</sup>, Yenia Román Bu<sup>1</sup>, Cinthya Rodríguez Hernández<sup>1\*</sup>, Carlos Manuel Delgado Rivero<sup>1</sup>, Manuel Cortés Cortés<sup>1</sup>

<sup>1</sup> Universidad de Cienfuegos, carretera a Rodas, km 4, Cienfuegos. Cuba.  
[rfbomate](mailto:rfbomate@ucf.edu.cu), [yroman](mailto:yroman@ucf.edu.cu), [cdelgado](mailto:cdelgado@ucf.edu.cu), [mcortes](mailto:mcortes@ucf.edu.cu)}@ucf.edu.cu

\* Autor para correspondencia: [crhdez@ucf.edu.cu](mailto:crhdez@ucf.edu.cu)

---

#### Resumen

GEReport es una herramienta destinada al diseño, generación y configuración de los reportes relacionados con los datos históricos almacenados en una fuente de datos. Además, luego de contar con toda la información del reporte es posible exportarlo como imagen y en los formatos HTML, PDF y Excel. Para la interacción con las aplicaciones externas, el sistema implementa un servicio que expone los metadatos de los reportes para poder utilizarlos sin restricciones de lenguajes y plataformas. El sistema se creó sobre un entorno web y se desarrolló siguiendo lo establecido por el Proceso Unificado de Desarrollo (RUP), utilizando UML como lenguaje de modelado. Para la gestión de la base de datos se seleccionó PostgreSQL, en la implementación se utilizaron los lenguajes de programación PHP, con CodeIgniter 2.0 como marco de trabajo del lado del servidor y JavaScript con Dojo Toolkit 1.8 para el trabajo del lado del cliente. Se utiliza en la Universidad de Cienfuegos por analistas y programadores del Grupo de Estudios y Desarrollo de Ingeniería y Sistemas, perteneciente a la Facultad de Ingeniería.

**Palabras clave:** consulta, diseño, generación, reporte.

#### Abstract

*GEReport is a tool for the design, generation and configuration of reports related to historical records stored in a data source. Also, after you have all the information in the report it can be exported as image also HTML, PDF and Excel. For the interaction with external applications, the system implements a service that exposes metadata reports for unrestricted use of languages and platforms. The system was built on a web environment and developed following the provisions of the Rational Unified Process (RUP), using UML as modeling language. To manage the database*

*PostgreSQL was selected, for implementation it was used PHP programming languages, with CodeIgniter framework 2.0 as server side and JavaScript with Dojo Toolkit 1.8 for client-side work. It is used at the University of Cienfuegos by analysts and programmers of the Study Group of Engineering and Systems Development, belonging to Faculty of Engineering.*

**Keywords:** consulting, design, generation, report.

---

## Introducción

Con el desarrollo de la ciencia y la técnica y el auge exponencial de la automatización en los procesos industriales, las producciones de las grandes industrias han alcanzado altos niveles. Para el control de los procesos de los distintos negocios, las empresas han automatizado en gran medida la gestión de la información, reduciendo los gastos de producción y la ejecución manual de estas operaciones.

Entre las principales funcionalidades de los sistemas informáticos, se encuentra la de proveer toda la información que se genera durante los distintos procesos de gestión en las empresas, relacionados con el control y supervisión de los mismos, así como los pertenecientes a niveles superiores dentro o fuera de la empresa; siendo el control de calidad, la supervisión y el mantenimiento algunos de los aspectos principales. Para lograr un control eficiente en la gestión de estas tareas, se hace necesario buscar mecanismos para la generación de reportes. Los mismos deben ser capaces de consolidar la información adquirida y mostrarla en formatos entendibles por el personal que espera recibir la información.

En la actualidad, existen herramientas que viabilizan el proceso de generación de reportes, dentro de las más importantes se encuentran: Active Reports (ActiveReports Server, 2014), Jasper Reports (Jaspersoft Corporation, 2013) y Crystal Reports (Crystal Solutions, 2013), pero estas solo cubren parte del ciclo de vida de los reportes, sin embargo, existen otras como Microsoft SQL Server 2012 Reporting Services (Krishnaswamy, 2013), que además de facilitar el diseño y generación de reportes, soporta los procesos de administración y entrega de los mismos, pero es una herramienta privativa. Las deficiencias de algunos sistemas para la elaboración de reportes, junto a las restricciones de licencias que presentan otros, dificultan el diseño, generación y configuración de los reportes de una manera rápida y eficiente. Además los clientes de base de datos no poseen mecanismos de acoplamiento para otras aplicaciones externas.

Existen otras tendencias que se han utilizado para mejorar el tiempo de desarrollo, como son los Sistemas Gestores de Contenidos (CMS). En este tipo de software para efectuar cambios en las particularidades de un módulo, ya sea un reporte u otra información relacionada, muchas veces es necesario interactuar directamente con el código fuente del mismo y conocer el lenguaje de programación implicado.

El objetivo del trabajo es construir un sistema informático que permita a los usuarios diseñar, generar y configurar sus propios reportes en un ambiente amigable (Galitz, 2007), sin necesidad de conocer lenguajes de programación, lo que permitirá una reducción en el tiempo de desarrollo del software y los costos de producción.

GEReport presenta opciones adicionales de exportación a los formatos de imágenes PNG y JPG, además de otros como HTML, PDF y Excel. Ofrece un servicio mediante el cual aplicaciones externas pueden utilizar los reportes creados por el sistema, permitiendo a los usuarios gestionar sus propios reportes y acceder a reportes creados por otros, con una gestión de políticas de seguridad en niveles de permisos basada en roles.

## **Materiales y métodos**

GEReport v1.0 es una herramienta capaz de construir reportes dinámicos que cumplan con los estándares mundiales de estos tipos de aplicaciones, y cuya utilización no se restringiese debido a las licencias de software, se ejecutase sobre un entorno web, se minimizaran las dependencias tecnológicas y fuese multiplataforma.

## **Metodología y lenguaje de modelado**

Luego de realizarse un estudio de las metodologías más usadas para el desarrollo de aplicaciones web (Villarreal, 2011) se decide utilizar el Proceso Unificado de Desarrollo (RUP). El mismo ofrece un enfoque disciplinado para la asignación de tareas y responsabilidades dentro de las organizaciones. Su meta principal es asegurar la producción de software de alta calidad que satisfagan las necesidades de los usuarios finales, enmarcándose en un calendario y presupuesto previsto. (Delgado, 2011)

Para el lenguaje de modelado se utilizó el Lenguaje Unificado de Modelado (UML) (Hernández, 2009), el cual presenta una notación gráfica muy expresiva que permite representar en mayor o menor medida todas las fases de un

proyecto informático: desde el análisis con los casos de uso, el diseño con los diagramas de clases, objetos y demás elementos, hasta la implementación y configuración con los diagramas de despliegue. (Jacobson, 2000)

Entre los artefactos generados por la metodología RUP se encuentra el diagrama de clases del dominio (ver Figura 1) y el diagrama de despliegue (ver Figura 2) que se muestran a continuación:

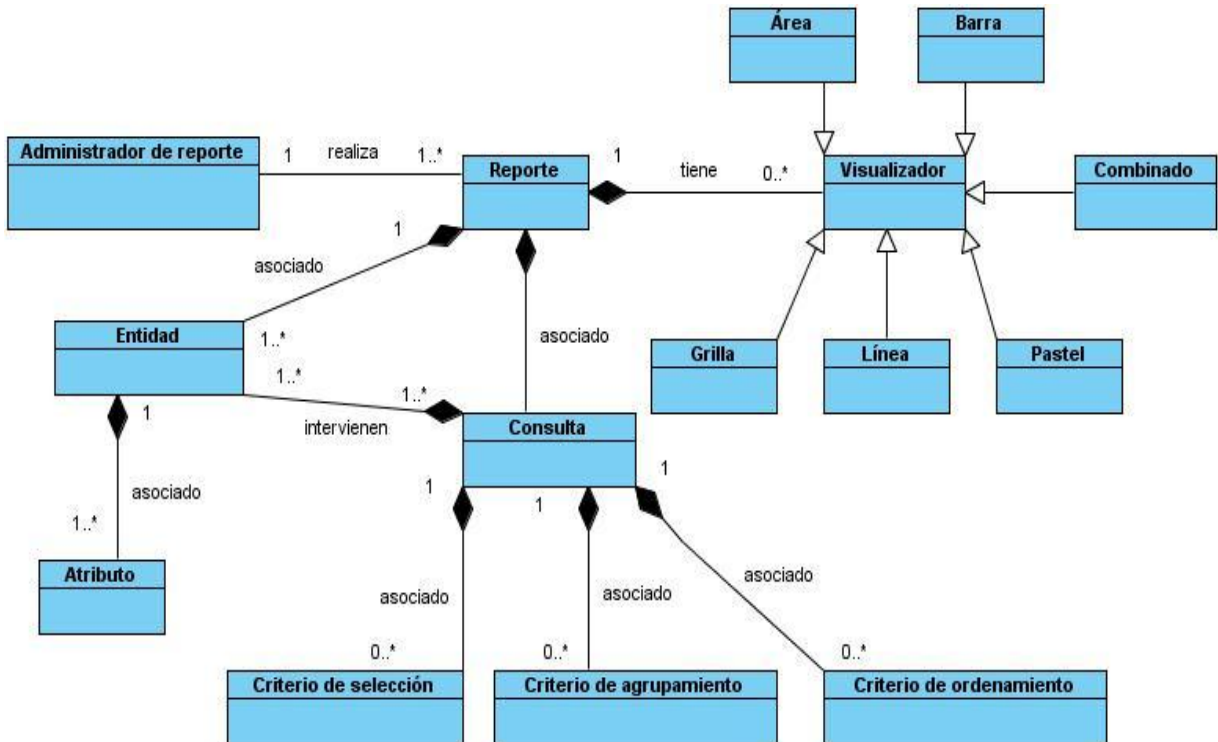


Figura 1: Diagrama de clases del dominio

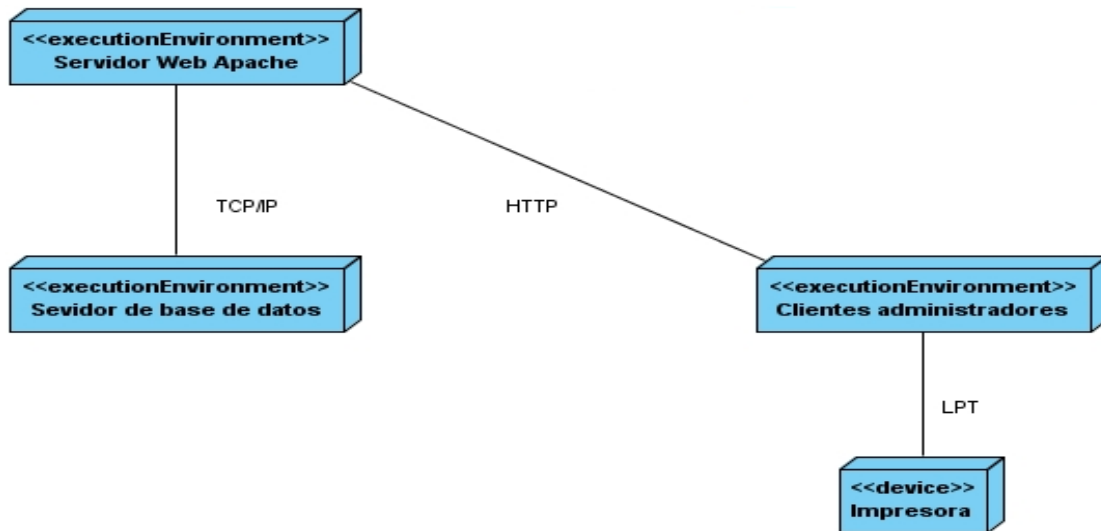


Figura 2: Diagrama de despliegue

## Lenguajes y herramientas de programación

La construcción de la aplicación se realizó sobre un entorno web, por lo que fue necesaria la utilización de tecnologías del lado del cliente y del servidor. El lado del cliente se escribió utilizando código JavaScript (Goodman, 2007), para lo cual, se usó Dojo 1.8 (Harmon, 2009) como marco de trabajo, además, se utilizaron lenguajes como XHTML y CSS (Álvarez, 2010) para el diseño de las vistas. El lado del servidor se desarrolló mediante código escrito en lenguaje PHP, para lo cual se utilizó CodeIgniter como marco de trabajo. (Velo, 2011)

## Base de Datos

Como gestor de base de datos se utilizó PostgreSQL, por libertad de su licencia y la creciente comunidad con que cuenta en Cuba y varios países de Latinoamérica. (Helmle, 2009)

## Arquitectura de la aplicación

La aplicación utiliza una arquitectura Cliente-Servidor, utilizando como estilo arquitectónico una variante del patrón Modelo-Vista-Controlador (MVC) (ver Figura 3). La variante basa su funcionamiento en las especificaciones descritas por el patrón MVC, pero se le agrega una capa adicional del lado del cliente llamada Caché (Teruel, 2011). Esta capa caché se utiliza para la persistencia de los datos del lado del cliente, lo cual garantiza que todas las

interacciones de la aplicación se realizan sin tener que interactuar con el servidor, por lo que el tiempo de respuesta de la herramienta es mejor.

Toda la lógica de la aplicación se realiza del lado del cliente y solo se va al servidor para almacenar la configuración de un estado de la aplicación en la base de datos. Con esta variante de arquitectura se emulan características que siempre han sido parte de las tecnologías de escritorio sobre la web (Bahit, 2011).

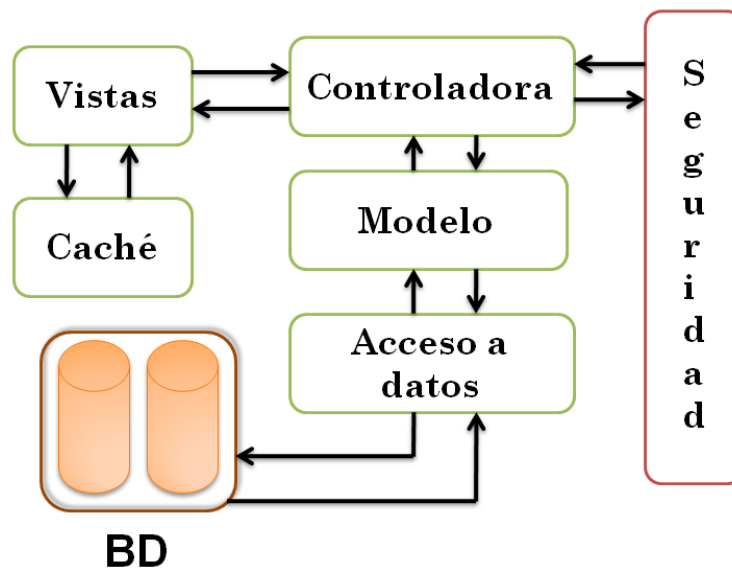


Figura 3: Arquitectura MVC

### Funcionamiento general de la aplicación

Para entender el funcionamiento del Reportador Dinámico es necesario conocer los componentes que se describen a continuación:

**Reporte:** estructura que contiene la información obtenida después de realizada la consulta a un conjunto de datos.

**Consulta:** conjunto de criterios y condiciones realizados a los campos de las entidades provenientes de una fuente de datos, permitiendo especificar diversos tipos de operaciones con los mismos. Se compone por la selección de los campos y las condicionales que se especifiquen en el diseño del reporte.

**Entidad:** modelo lógico que representa las tablas de la base de datos.

Campo: modelo lógico que representa los atributos de las tablas que pertenecen a la base de datos.

Selección: proceso en el cual se seleccionan los campos de las entidades que formarán parte del reporte.

Diseñador de reporte: sección de la aplicación donde se confecciona el reporte deseado a través de la selección de los campos y las condiciones asociadas a los mismos.

Generador de reporte: sección de la aplicación en la cual una vez diseñado el reporte se pasa a generar el mismo, con el objetivo de obtener resultados.

Visualizador: componente gráfico que permite mostrar la información del reporte.

Configurador de reporte: componente de la aplicación donde se modela la forma en que será mostrado el contenido del reporte, una vez seleccionado el visualizador. En este paso, se especifican indicadores de acuerdo con un criterio de selección, con el fin de resaltar la información relevante del reporte.

## **Fases del sistema**

Diseño: a partir de las entidades involucradas en la base de datos es posible construir el reporte seleccionando aquellos campos a los cuales se les desea aplicar un criterio determinado y está dividido de la manera siguiente:

Criterio de selección: cuenta con un listado de los campos seleccionados, así como un conjunto de funciones matemáticas y condicionales que se deseen aplicar a los mismos.

Criterio de Agrupamiento: dado el criterio de selección, es posible definir la forma en que será agrupada la información resultante.

Criterio de Ordenamiento: se especifica el orden de la selección que será mostrada, la misma puede ser ascendente o descendente.

Generación: proceso en el cual se ejecuta el reporte diseñado.

Resultado de la consulta en lenguaje SQL: al generar el reporte se muestra en una vista la consulta realizada en lenguaje SQL.

Resultado de la consulta en lenguaje natural: al generar el reporte se muestra en una vista la consulta realizada en lenguaje natural, que puede comprenderse por cualquier persona sin conocimientos de programación.

Configuración: posee un conjunto de componentes en los cuales será mostrada la información ya sea por grillas, gráficos, plantillas, tipo formulario, dando la posibilidad de personalizar los mismos de forma rápida y fácil. Cada uno de estos elementos se añade al área de configuración y a través de sus propiedades se configuran. En un mismo reporte se pueden crear tantos modos de visualización como sean necesarios, por lo que la movilidad entre estos se realizará a través del arrastre del ratón. También cuenta con indicadores para resaltar los datos de interés.

Opciones Adicionales: luego de contar con toda la información del reporte, es posible exportar los reportes en imágenes y formatos como HTML, PDF, Excel. Además ofrece un servicio mediante el cual las aplicaciones externas pueden utilizar los reportes creados por el sistema.

Guardar reporte: se brinda la posibilidad de guardar los reportes realizados, o los nuevos cambios que se realicen sobre los mismos.

Eliminar reporte: el usuario encargado en configurar los reportes podrá eliminar los reportes.

Editar reporte: en caso de que se necesite efectuar algún cambio sobre un reporte, el usuario responsable en esta tarea podrá seleccionarlo y actualizar la información del mismo.

Compartir reporte: el usuario creador del reporte puede especificar si su reporte será compartido o propio. El reporte compartido será visualizado para un conjunto de usuarios definidos por el administrador de reporte, mientras que el reporte propio, solo será visualizado por el usuario que lo creó.

Imprimir reporte: el sistema permite imprimir el reporte deseado.



## **La plataforma basa sus políticas de seguridad en niveles de permisos basados en roles. Un usuario en GEReport v1.0 puede ser:**

Superadministrador de reporte: es el encargado de gestionar todos los reportes creados, así como las cuentas de aquellos usuarios que interactúen con la aplicación, para lo cual, otorga y elimina permisos a los mismos.

Administrador de reporte: usuario encargado de gestionar los reportes, realizando las opciones de inserción, eliminación y actualización, así como el diseño, generación y configuración. Además realiza tareas adicionales de exportación e importación de reportes especificando, si su uso será compartido o propio.

Usuario simple: cualquier usuario que interactúe con los Reportes Dinámicos construidos en la aplicación sin permisos administrativos. Estos tipos de usuarios son los que realizan análisis sobre la información que es mostrada en el reporte.

GEReport se implementa con filtros para prevenir ataques por inyecciones SQL o inyecciones de código JavaScript, así como el limpiado y verificación de tipos en las variables que se capturan a través de las URL.

### **Principales funcionalidades**

Las funcionalidades de GEReport v1.0 se dividen de acuerdo con los permisos que posean los usuarios que interactúen el sistema. A continuación se realizará un resumen de las principales funcionalidades por usuarios.

#### **Funcionalidades del administrador de cuentas**

1. Gestión de las cuentas de los usuarios que interactuarán con el Sistema de Gestión de Reportes Dinámicos.
2. Asignación de usuarios al Sistema de Gestión de Reportes Dinámicos.
3. Manejo de permisos.

#### **Funcionalidades del administrador de reportes**

1. Gestión de los reportes.
2. Diseño, generación y configuración del reporte.

3. Importación de reportes desde sitios externos.
4. Exportación de reportes a los formatos HTML, PDF, Excel y a otros como PNG o JPG.

### **Funcionalidades del usuario simple**

- Visualización y construcción de consultas dinámicas sobre los datos representados en los reportes, siempre y cuando tenga permisos de visualización.
- Exportación de reportes a los formatos HTML, PDF, Excel y a otros como PNG o JPG, si presenta permisos de visualización.
- Búsqueda de información a partir varios criterios.
- Vinculación de los reportes y sus características creados en GEReport v1.0 con aplicaciones externas a través de exportaciones como sitio web y de un servicio de reportes que brinda el sistema.

### **Resultados y discusión**

GEReport se utiliza en la Universidad de Cienfuegos por analistas y programadores del Grupo de Estudios y Desarrollo de Ingeniería y Sistemas (GEDIS) perteneciente a la Facultad de Ingeniería. Es consumido como un servicio por las aplicaciones que necesitan obtener reportes de diferentes tipos, minimizándose el tiempo de construcción de nuevos sistemas informáticos y facilitando el trabajo de los programadores. Los resultados obtenidos son positivos, con una alta aceptación y críticas positivas.

### **Validación del sistema**

Para la validación del software se aplicó una encuesta a los posibles clientes del sistema, que pueden ser desarrolladores de software que necesiten gestionar reportes, profesores y alumnos de las universidades del país que requieran utilizar reportes para consultar información de interés y profesionales de las empresas del territorio que en un momento dado precisasen de los servicios que brinda dicha aplicación. Se realizaron 6 preguntas, enfocadas en las características y utilidad del sistema.

Luego de aplicada la encuesta y recogida la base de datos se utilizó el paquete de programa SPSS v19 (Martínez, 2012) para la realización del análisis estadístico, obteniéndose los resultados que se muestran a continuación:

La pregunta 1 fue para evaluar la utilidad del software para la gestión de reportes, las respuestas posibles eran: Muy bueno, Bueno, Regular o Malo, oscilando las respuestas de los encuestados entre los valores Muy bueno y Bueno, con un porcentaje del 82.2% para el primer caso y un 17.8% para el segundo.

La pregunta 2 fue para evaluar el proceso de construcción de reportes, en cuanto a la forma de diseñarse, generarse y configurarse, las respuestas posibles eran: Muy bueno, Bueno, Regular o Malo, oscilando las respuestas de los encuestados entre los valores Muy bueno y Bueno, con un porcentaje del 93.3% para el primer caso y un 6.7% para el segundo.

La pregunta 3 fue sobre las características del uso del sistema presentado, las respuestas posibles eran: es más fácil de usar que otros, es igual a los otros, es más difícil, es único; siendo el 57.8% de las respuestas que es más fácil de usar, el 2,2% que es igual a los otros y el 40% que es único.

La pregunta 3.a fue en cuanto a la novedad del sistema, las respuestas posibles eran: es novedoso, tiene mejoras, es igual o es más malo; siendo el 93.3% de las respuestas que es novedoso.

La pregunta 4 fue en cuanto a la forma de presentación, las respuestas posibles eran: Muy bueno, Bueno, Regular o Malo, oscilando las respuestas de los encuestados entre los valores Muy bueno y Bueno, con un porcentaje del 86.7% para el primer caso y un 13.3% para el segundo.

La pregunta 5 fue en cuanto a las ventajas, las respuestas posibles eran: en la entrada de datos, en la facilidad de la búsqueda, en el empleo de los visualizadores, en la impresión, en la generación de informes, en la calidad de la aplicación, en la generación de consultas, en todas las anteriores, no tiene ventajas, siendo las respuestas más significativas: el empleo de los visualizadores con un porcentaje del 20%, en la calidad de la aplicación con el 11.1% y en todas las ventajas señaladas con el 42.2%.

La pregunta 6 fue para otorgarle una evaluación al sistema, que puede oscilar entre los valores desde 1 hasta 5, siendo 5 la máxima calificación. Se obtuvo como resultado que el 4.4% de los encuestados le otorgaron 4 puntos y el 95.6% de 5 puntos. La media fue de 4.96, lo que implica que la tendencia de la evaluación de los clientes es a la máxima puntuación.

Los beneficios obtenidos con el desarrollo del software permiten agilizar el proceso de creación de reportes dinámicos para las aplicaciones que lo requieran, además de disminuye el tiempo de desarrollo de software. Unido esto a las ventajas de la digitalización y mejora de la calidad de la información por su integridad y confiabilidad.

## **Conclusiones**

GEReport v1.0 es una plataforma que puede ser usada sin pagos de licencia, se ejecuta bajo un entorno web, lo que facilita la interacción de los usuarios con la misma y es multi-plataforma, ampliando su utilización para todos los sistemas operativos.

La utilización de RUP como metodología para guiar el proceso de construcción del sistema garantiza un desarrollo organizado y una completa documentación del producto terminado, que el sistema sea fácil de mantener y los cambios que se puedan incorporar a este no requieran un alto grado de complejidad.

El uso de PHP como lenguaje del lado del servidor, PostgreSQL como gestor de base de datos y marcos de trabajo del lado del cliente y del servidor como Dojo 1.8 y CodeIgniter 2.0 respectivamente, unido a un serio trabajo en la implementación de los requisitos funcionales, hacen de la herramienta una solución robusta.

La seguridad basada en roles garantiza que cada usuario pueda tener acceso solo a las funcionalidades para el cual su rol tiene permiso. La prevención de ataques por inyecciones SQL y JavaScript asegura que no se violen las políticas de usuarios y la integridad de los mismos permanezca intacta.

Las características tecnológicas de la aplicación, unidas al conjunto de funcionalidades que esta ofrece, hacen de GEReport v1.0 una excelente solución para construir reportes dinámicos que cumplan con las necesidades del mercado mundial en este tipo de aplicaciones.

## Referencias

- ACTIVEREPORTS SERVER. *ActiveReports 8 launched with Server and Ad-hoc Report Designer*. [En línea] Component one, 2014 [Consultado el: 21 de mayo de 2014]. Disponible en: <http://www.activereportserver.com>
- ÁLVAREZ, M. A. *Características y ventajas de las CSS*. [En línea] Características y ventajas de las CSS, 2010. [Consultado el: 21 de marzo de 2012]. Disponible en: <http://www.desarrolloweb.com/articulos/182.php>
- ÁLVAREZ, M. A. *Qué es un CMS*. [En línea] Qué es un CMS, 2008. [Consultado el: 6 de junio de 2013]. Disponible en: <http://www.desarrolloweb.com/articulos/que-es-un-cms.html>
- GARRETT, JESSE JAMES. *Ajax: un nuevo acercamiento a aplicaciones web*. [En línea]. AJAX un nuevo acercamiento a Aplicaciones Web, 2011. [Consultado el: 30 de noviembre de 2012] Disponible en: <http://www.uberbin.net/archivos/internet/ajax-un-nuevo-acercamiento-a-aplicaciones-web.php>
- BAHIT, EUGENIA. *El paradigma de la Programación Orientada a Objetos en PHP con el patrón arquitectónico MVC*. [En línea]. Bubok Publishing S.L., 2011. [Consultado el: 26 de octubre de 2011] Disponible en: <http://www.bubok.es/libros/205199/POO-y-MVC-en-PHP>
- CRYSTAL SOLUTIONS. *SAP Crystal Solutions: Essential BI for small business*. [En línea]. SAP Crystal Solutions, 2013. [Consultado el: 20 de mayo de 2014] Disponible en: <http://www.crystalreports.com>
- DELGADO RIVERO, CARLOS MANUEL. *Sistema informático para la gestión de la información de los cursos de postgrado de la Universidad “Carlos Rafael Rodríguez”*. Tesis para obtener el título de Ingeniero Informático, Universidad de Cienfuegos “Carlos R. Rodríguez”, Cienfuegos, 2011.
- GALITZ, WILBERT O. *The Essential Guide to User Interface Design*. Indianapolis, Wiley Publishing, Inc., 2007. 888 p.
- GOODMAN, D. & MORRISON, M. *JavaScript Bible, 6th ed*. Indianapolis, Wiley Publishing, Inc., 2007. 1200 p.
- HARMON, JAMES E. *Using the Dojo JavaScript Library to Build Ajax Application*. Boston: Addison-Weasley, 2009. 336 p.

- HELMLE, BERND. *Administración en PostgreSQL*. Washington, O'Reilly Vlg. GmbH & Co, 2009. 393 p.
- HERNÁNDEZ ORALLO, ENRIQUE. *El Lenguaje Unificado de Modelado (UML)*. [En línea] 2009. [Consultado el: 21 de marzo de 2012]. Disponible en: <http://www.disca.upv.es/enheror/pdf/ActaUML.PDF>
- JACOBSON, I; BOOCH, G. & RUMBAUGH, J. *El Proceso Unificado de Desarrollo de Software*. Massachusetts, Addison – Wesley, 2000. 464 p.
- JASPERSOFT CORPORATION. *JasperReports Server*. [En línea] Jaspersoft Community, 2013. [Consultado el: 20 de mayo de 2014] Disponible en: <http://community.jaspersoft.com>
- MARTÍNEZ, ANDRÉS G. *Spss Para Todos*. [En línea]. SPSS free, 2012. [Consultado el: 26 de septiembre de 2012] Disponible en: <http://www.spssfree.com>
- KRISHNASWAMY, J. *Learning SQL Server Reporting Services*. Birmingham, Packt Publishing, 2013. 566p.
- TERUEL, A. *Introducción a la arquitectura de capas*. [En línea]. Introducción a la arquitectura de capas, 2011. [Consultado el: 23 de marzo de 2012] Disponible en: <http://www ldc.usb.ve/~teruel/ci3715/clases/arqCapas.html>.
- VELO, FERNANDO. *CodeIgniter 2.1.0. Guía de Usuario en español*. [En línea]. Diseño y programación Web, 2011. [Consultado el: 7 de octubre de 2011] Disponible en: <http://www.etnassoft.com>
- VILLARROEL ACEVEDO, R. & RIOSECO REINOSO, C. *Una comparación de metodologías para el modelado de aplicaciones web*, 2011. [En línea]. Revista Cubana de Ciencias Informáticas. [Consultado el: 5 de febrero de 2012] Disponible en: <http://rcci.uci.cu>