

GENERADOR DE RUTINAS PARA EL SISTEMA AUTOMATIZADO PARA LA GESTIÓN DE LA INFORMACIÓN

Lijandy Jiménez Armas – Universidad de las Ciencias Informáticas
Miguel Ángel Cárdenas Hernández - Universidad de las Ciencias Informáticas

RESUMEN

La estadística adquiere relevancia en tanto hace posible transformar grandes cantidades de datos en información útil a los procesos administrativos como soporte a la toma de decisiones. En Cuba la Oficina Nacional de Estadística e Información (ONEI) realiza la gestión de los datos estadísticos a través del Sistema Automatizado para la Gestión de la Información (SAGI), desarrollado en convenio con la Universidad de las Ciencias Informáticas (UCI). Este sistema permite el diseño de formularios y encuestas para la captura de información y la generación de reportes mediante el Generador Dinámico de Reportes (GDR), lo que contribuye al proceso de toma de decisiones. Uno de los procesos fundamentales para la concepción de reportes es la generación de rutinas, algo que no es posible realizar hoy en SAGI. Este proceso se encuentra implementado en una herramienta externa, SIGETools, lo que obliga a los especialistas a utilizar dos o más aplicaciones para la obtención de reportes. El objetivo de la presente investigación es desarrollar un módulo para SAGI que permita la creación de rutinas. Para guiar dicha investigación se seleccionó como metodología de desarrollo de software OpenUP, UML 2.1 como lenguaje de modelado y Visual Paradigm 8.0 como herramienta CASE., los lenguajes de programación PHP 5 y JavaScript, Symfony 1.1.7 y ExtJS 3.4 como framework de desarrollo y para el tratamiento de las bases de datos PostgreSQL 9.3 y PgAdmin III. Además, se diseñaron e implementaron las clases del módulo aplicando buenas prácticas de los patrones de diseño. Se le realizaron pruebas al mismo, comprobando su correcto funcionamiento.

PALABRAS CLAVES: Datos estadísticos. generación de rutinas. Reportes. Sistema Automatizado para la Gestión de la Información.

INTRODUCCIÓN

El análisis estadístico resulta fundamental para la recolección y agrupamiento de datos de diversos tipos para construir con ellos informes estadísticos que nos permitan conocer y analizar el comportamiento sobre diferentes temas, siempre desde un punto de vista cuantitativo. En Cuba dado el carácter esencialmente central de la administración del Estado, toda empresa, organismo, organización o institución que posea información de interés para la toma de decisiones de los órganos de la administración del Estado, está en la obligación de reportar información estadística bajo convenio con la Oficina Nacional de Estadística e Información (ONEI).

La ONEI ha priorizado la introducción de técnicas de almacenamiento y procesamiento de datos. En este contexto se inició de conjunto con la Universidad de las Ciencias Informáticas (UCI) el desarrollo de una aplicación que de soporte al marco metodológico del trabajo estadístico. Surgiendo el proyecto de informatización Sistema Automatizado para la Gestión de la Información (SAGI), este tiene como finalidad la gestión estadística en Cuba. Como parte de esta gestión comprende los procesos de diseño de plantillas de encuestas y formularios, para la captura de datos estadísticos, así como también permite organizar y exhibir los datos mediante reportes. Estos permiten dar un formato determinado a la información captada para mostrarla por medio de un diseño atractivo y que sea fácil de interpretar por los usuarios.

SAGI genera los reportes mediante el Generador Dinámico de Reportes (GDR) en su versión 1.8, para la obtención de los mismos el especialista debe crear un modelo (abstracción de la base de datos que contiene tablas, vistas y funciones). También debe implementar las consultas necesarias para obtener los datos que se desean visualizar a través del reporte. Posteriormente se selecciona un modelo y se procede a actualizar los metadatos de este, de manera que se puedan incluir las consultas implementadas o modificadas. En ocasiones los reportes difieren en algún parámetro, dígame provincia, mes, año, indicador y nombre de la empresa, aun cuando la diferencia entre los reportes sea mínima es necesario la repetición de todo el proceso para cada uno de estos, ya que la ONEI procesa anualmente un gran cúmulo de datos lo que trae como consecuencia que se emplee mucho tiempo y personal dedicado a ello.

Actualmente la ONEI para dar solución a esta situación emplea SIGETools, una aplicación que facilita la generación de funciones parametrizadas o rutinas, las cuales son funciones SQL que permiten parámetros de diferentes tipos, posibilitando la creación de reportes y a su vez estas puedan ser reutilizadas por los especialistas de la entidad. Aun cuando SIGETools ha agilizado el trabajo con los reportes hasta el momento, esta presenta varios inconvenientes:

- Está desarrollada con un lenguaje de licencia privativa, lo cual no cumple con las políticas de soberanía tecnológica en Cuba.
- Es una herramienta externa al SAGI la cual hace el trabajo más engorroso ya que se utilizan dos herramientas totalmente diferentes con el mismo fin.
- Es una herramienta de escritorio, por lo que debe ser instalada en todos los ordenadores que se utiliza el sistema.
- No es compatible con todas las distribuciones de Microsoft Windows y ninguna de Linux.
- No existe documentación sobre la herramienta, lo que dificulta el trabajo con la misma.
- No cuenta con un ciclo de soporte.

El empleo de dos herramientas totalmente diferentes tecnológicamente trae consigo la necesidad de una cantidad más elevada de especialistas con mayor tiempo disponible y mayor capacitación.

Debido a la necesidad de dar solución a la situación planteada se identifica como problema de la investigación que el Sistema Automatizado para la Gestión de la Información(SAGI) requiere para la generación de funciones parametrizadas el uso de una o más herramientas externas.

Donde el objeto de estudio está dirigido a la generación de rutinas, enmarcado en el campo de acción generación de rutinas en el Sistema Automatizado para la Gestión de la Información.

Para dar solución al problema de investigación planteado se define como objetivo general desarrollar el módulo generador de rutinas en el Sistema Automatizado para la Gestión de la Información para el proceso de creación de reportes en la Oficina Nacional de Estadística e Información.

1

DESARROLLO

2

Metodología de desarrollo de software:

Una definición estándar de metodología puede ser el conjunto de métodos que se utilizan en una determinada actividad con el fin de formalizarla y optimizarla. Determina los pasos a seguir y cómo realizarlos para finalizar una tarea (INTECO, 2010), dentro de estas metodologías se encuentra OpenUP.

OpenUP: Es un proceso modelo y extensible, dirigido a gestión y desarrollo de proyectos de software basados en desarrollo iterativo, ágil e incremental apropiado para proyectos pequeños y de bajos
XIV EVIDOSOL y XI CILTEC-Online - junio / 2017 - <http://evidosol.textolivre.org>

recursos; y es aplicable a un conjunto amplio de plataformas y aplicaciones de desarrollo (eclipse, 2011).

Se escoge esta metodología ya que:

Es apropiado para proyectos pequeños y de bajos recursos permite disminuir las probabilidades de fracaso en los proyectos e incrementar las probabilidades de éxito.

Permite detectar errores tempranos a través de un ciclo iterativo.

Evita la elaboración de documentación, diagramas e iteraciones innecesarios requeridos en la metodología RUP.

Por ser una metodología ágil tiene un enfoque centrado al cliente y con iteraciones cortas.

Lenguaje de Modelado:

UML 2.1: UML 2 avanza la especificación UML éxito, y se está convirtiendo rápidamente en el estándar aceptado para especificar, documentación y visualización de sistemas de software. También se utiliza para el modelado de sistemas no de software, y se implementa ampliamente en la mayoría de los sectores de la industria, incluyendo las finanzas, militar y de ingeniería. (uml-diagrams, 2010)

Se utiliza este lenguaje de modelado por proporcionar orientación en cuanto al orden de las actividades de un equipo, dirigir las tareas de los desarrolladores individuales y del equipo en su conjunto, por ofrecer criterios para el seguimiento, la medición de los productos y las actividades de un proyecto.

Herramientas Case:

Visual Paradigm 8.0

Visual Paradigm (anteriormente VP-UML) es una herramienta UML. La herramienta está diseñada para una amplia gama de usuarios, incluyendo ingenieros de software, analistas de sistemas, analistas de negocios y arquitectos de sistemas, o para cualquier persona que esté interesada en la construcción de forma fiable los sistemas de software a gran escala con un enfoque orientado a objetos (Paradigm, 2010).

Se escoge la herramienta Visual Paradigm por soportar el ciclo de vida completo del proceso de desarrollo del software a través de la representación de todo tipo de diagramas, sirviendo como guía y facilitando el proceso de desarrollo de todas las fases de creación del módulo (STEWART, 2016).

Lenguajes de Programación:

PHP5: Acrónimo de "PHP: Hypertext Preprocessor", es un lenguaje de 'scripting' de propósito general y de código abierto que está especialmente pensado para el desarrollo web y que puede ser embebido en páginas HTML. Su sintaxis recurre a C, Java y Perl, siendo así sencillo de aprender. El objetivo principal de este lenguaje es permitir a los desarrolladores web escribir dinámica y rápidamente páginas web generadas; aunque se puede hacer mucho más con PHP (KANEIWA, y otros, 2015).

JavaScript: JavaScript es un lenguaje de scripting multiplataforma y orientado a objetos. Es un lenguaje pequeño y liviano. Dentro de un ambiente de host, JavaScript puede conectarse a los objetos de su ambiente y proporcionar control programático sobre ellos.

JavaScript contiene una librería estándar de objetos, tales como Array, Date, y Math, y un conjunto central de elementos del lenguaje, tales como operadores, estructuras de control, y sentencias. El núcleo de JavaScript puede extenderse para varios propósitos, complementándolo con objetos adicionales (MDN, 2009).

Marco de trabajo:

Symfony 1.1.7: Symfony es un completo framework diseñado para optimizar, gracias a sus características, el desarrollo de las aplicaciones web, este separa la lógica de negocio, la lógica de servidor y la presentación de la aplicación web. Proporciona varias herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación web compleja. Además, automatiza las tareas más comunes, permitiendo al desarrollador dedicarse por completo a los aspectos específicos de cada aplicación (Potencier, 2009).

ExtJS 3.4: De acuerdo a la definición, ExtJS es una librería JavaScript que permite construir aplicaciones complejas en Internet. Esta librería incluye:

Componentes UI del alto desempeño y personalizables.

Modelo de componentes extensibles.

Un API fácil de usar.

Licencias open source y comerciales.

Una de las grandes ventajas de utilizar ExtJS es que permite crear aplicaciones complejas utilizando componentes predefinidos, así como un manejador de layouts similar al que provee Java Swing, gracias a esto provee una experiencia consistente sobre cualquier navegador, evitando el tedioso problema de validar que el código escrito funcione bien en cada uno (Firefox, IE, Safari, etc.) (CHEN, 2011).

Como marcos de trabajo se utilizarán Symfony 1.1.7 y ExtJS 3.4 teniendo en cuenta que la programación del lado del servidor en SAGI se llevó a cabo con Symfony 1.1.7 y que ExtJS 3.4, en este caso utilizado para la programación de la interfaz, están diseñados basados en el patrón arquitectónico Modelo-Vista-Controlador, que facilita la separación de estos elementos posibilitando un balance entre Cliente – Servidor, de esta forma la carga de procesamiento se distribuye, permitiendo que el servidor, al tener menor carga, pueda manejar más clientes al mismo tiempo. También contienen una base de componentes amplia que permite la reutilización y extensión de los mismos, así como una extensa documentación.

Entorno de desarrollo Integrado:

NetBeans 8.0.2: Es un entorno de desarrollo gratuito y de código abierto que en el momento de escribir este artículo está en su versión 8.0.2. Permite el uso de un amplio rango de tecnologías de desarrollo tanto para escritorio, como aplicaciones Web, o para dispositivos móviles. Da soporte a las siguientes tecnologías, entre otras: Java, PHP, Groovy, C/C++, HTML5. Además, puede instalarse en varios sistemas operativos: Windows, Linux, Mac OS, etc. (genbetadev, 2014).

Este IDE de desarrollo se usa fundamentalmente por ser multilinguaje, pero además por ser de código abierto y gratuito. Por lo cual es la herramienta idónea para el desarrollo del módulo.

Sistema Gestor de Bases de Datos (SGBD)

PostgreSQL 9.3: PostgreSQL es un sistema de gestión de bases de datos objeto-relacional, distribuido bajo licencia BSD y con su código fuente disponible libremente. Es el sistema de gestión de bases de datos de código (OBE, y otros, 2015). Utiliza un modelo cliente/servidor y usa multiprocesos en vez de multihilos para garantizar la estabilidad del sistema. Un fallo en uno de los procesos no afectará el resto y el sistema continuará funcionando (OBE, y otros, 2015).

PgAdmin III: es una herramienta de código abierto para la administración de bases de datos PostgreSQL y derivados (EnterpriseDB, PostgresPlus, Advanced Server y GreenplumDatabase. Responde a las necesidades de la mayoría de los usuarios, desde escribir simples consultas SQL hasta desarrollar bases de datos complejas. La interfaz gráfica soporta todas las características de

PostgreSQL y hace simple la administración. Está disponible en más de una docena de lenguajes y para varios sistemas operativos, incluyendo Microsoft Windows, Linux, FreeBSD, Mac OSX y Solaris. Soporta versiones de servidores 7.3 y superiores. Versiones anteriores a 7.3 deben usar PgAdmin II (MICHAUD, 2015).

Servidor Web:

Apache 2.2: Un servidor web es, como su nombre indica, un software instalado en el equipo con todas las condiciones necesarias para servir o entregar páginas web que le sean solicitadas por un navegador, asegurando que se muestren y representen todos los elementos necesarios para su correcto funcionamiento y visualización. Existen varios tipos de servidores web, Apache es un software de código abierto, libre de uso y totalmente configurable, es en este momento el más utilizado en la red, ya sea en plataformas Linux o Windows (MORO, y otros, 2013).

El servidor está dirigido a servir a una gran cantidad de plataformas web que trabajan sobre sistemas operativos como Unix, Windows, Linux, Solaris, Novell NetWare, FreeBSD, Mac OS X, Microsoft Windows, OS / 2 (PAYER, y otros, 2015).

3

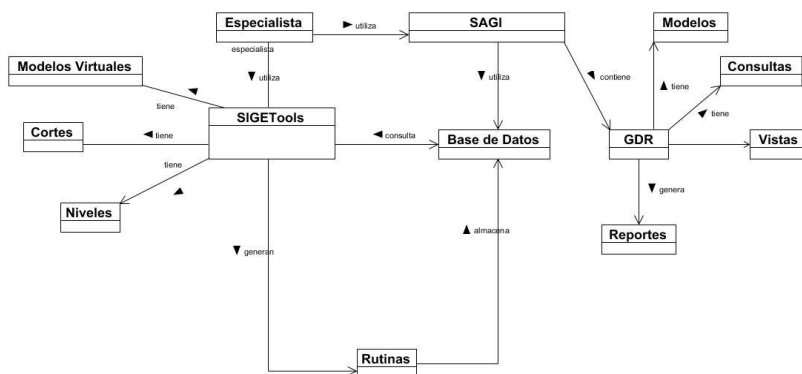


Figura 1: Modelo de Dominio.

4

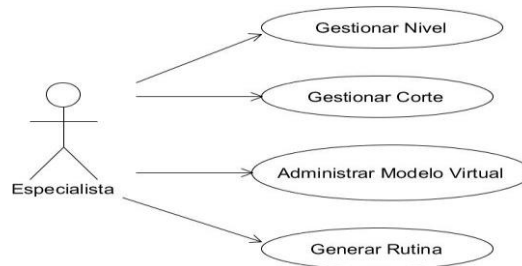


Figura 2: Diagrama de caso de uso del sistema.

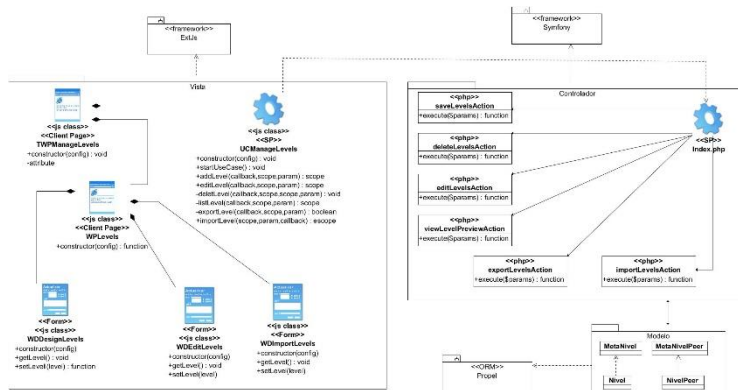


Figura 3: Diagrama de clase de diseño del CU Gestionar Nivel.



Figura 4: Diagrama entidad relación.

Resultado de la investigación:

Luego de aplicadas las técnicas y utilizadas las herramientas anteriormente descritas se obtuvo como resultado el módulo generador de rutinas para el Sistema Automatizado para la Gestión de la Información.



Figura 5: Sistema Automatizado para la Gestión de la Información.

CONCLUSIONES

- El análisis de SAGI y SIGETools, facilitó la selección de las herramientas y las tecnologías a utilizar para el correcto desarrollo del módulo.
- La implementación de las funcionalidades del módulo permitió obtener una aplicación informática que responde a las necesidades del cliente definidas durante la etapa de análisis.
- Las pruebas diseñadas y aplicadas al módulo garantizaron el correcto funcionamiento del mismo, detectando no conformidades que fueron solucionadas posteriormente.
- El desarrollo del Módulo para la Gestión de Rutinas para el SAGI permitió la creación de las mismas en el SAGI posibilitando así un mayor control del cumplimiento de los reportes y que los mismos puedan ser entregados y analizados en el tiempo estimado.

REFERENCIAS

Agualló, P. 2010. Desarrollo Cliente / servidor: ubicación de las reglas de negocio. Desarrollo Cliente / servidor: ubicación de las reglas de negocio. [En línea] 2010. [Citado el: 15 de febrero de 2015.] <http://www.ctv.es /USERS/pagullo /arti/esbr/esbr.htm..>

Anglada, Alain Abel Garófalo Hernández. 2013. Revista Cubana de Ciencias Informáticas. Revista Cubana de Ciencias Informáticas. [En línea] abr-jun. de 2013. [Citado el: 26 de octubre de 2015.] http://scielo.sld.cu/scielo.php?pid=S2227-18992013000200006&script=sci_arttext. ISSN 2227-1899.

Boehm. SW Cost Estimation with COCOMO II.

BOOCH, Grady, RUMBAUGH, James, JACOBSON y Ivar. 2009. El lenguaje unificado de modelado. Manual de. 2009.

Bruegge, B. Y Dutoit, A. 2002. Ingeniería de Software Orientado a Objetos. 2002.

BRUEGGE, Bernd, DUTOIT, Allen. 2008. Ingeniería de software orientado a objetos. 2008.

CHEN, Dao-xin. 2011. EXTJS Framework in the Development of Application [J]. Computer Knowledge and Technology. 2011.

Cornejo, José Enrique González. 2008. DocIRS. DocIRS. [En línea] DocIRS, enero de 2008. [Citado el: 24 de octubre de 2015.] <http://www.docirs.com/uml.htm>.

disca. 2012. disca. disca.upv. [En línea] upv, enero de 2012. <http://www.disca.upv.es/enheror/pdf/ActaUML.PDF>.

eclipse. 2011. eclipse. eclipse. [En línea] 2011. [Citado el: 10 de 7 de 2016.] <http://epf.eclipse.org/wikis/openup/>.

GAMMA, E., HELM, R. y JOHNSON, R. y VLISSIDES, J. 2009. Patrones de diseño. Patrones de diseño. [En línea] vico, 2009. <http://www.vico.org/pages/PatronsDisseny.html..>

genbetadev. 2014. genbetadev: Desarrollo de Software. genbetadev: Desarrollo de Software. [En línea] 9 de enero de 2014. [Citado el: 15 de diciembre de 2015.] <http://www.genbetadev.com/herramientas/netbeans-1>.

ibiblio. 2014. ibiblio. ibiblio.org. [En línea] Postgresql, 2014. <https://www.ibiblio.org/pub/linux/docs/LuCaS/Postgresql->

es/web/navegable/todopostgresql/xfunc.htm.

IBM. 2009. IBM Knowledge Center. IBM Knowledge Center. [En línea] IBM, diciembre de 2009. [Citado el: 24 de octubre de 2015.] [/www-01.ibm.com/support/knowledgecenter/SSEPGG_8.2.0/com.ibm.db2.udb.doc/ad/c0009415.htm?lang=es](http://www-01.ibm.com/support/knowledgecenter/SSEPGG_8.2.0/com.ibm.db2.udb.doc/ad/c0009415.htm?lang=es).

JACOBSON, Ivar y BOOCH, Grady, RUMBAUGH, James. 2004. El Proceso Unificado. 2004.

Larman, C. 2008. UML y patrones. 2008.

Paradigm. 2010. Paradigm, 2010. Paradigm, 2010. [En línea] www.visual-paradigm.com, 2010. [Citado el: 15 de noviembre de 2015.] <http://www.visual-paradigm.com/aboutus/newsreleases/vpuml80.jsp>.

PAYER, Mathias, BARRESI, Antonio y GROSS, Thomas R. 2015. Fine-grained control-flow integrity through binary hardening. En International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment. s.l. : Springer International Publishing, 2015.

PHP. 2011. php. php. [En línea] php.net, 2011. [Citado el: 26 de octubre de 2015.] <http://php.net/manual/es/preface.php>.

Potencier, Fabien. 2015. librosweb. [En línea] 2015. [Citado el: 6 de Febrero de 2016.] http://librosweb.es/libro/symfony_1_1/capitulo_2/el_patron_mvc.html.

Potencier, François Zaninotto. 2009. Symfony, la guía definitiva. Symfony, la guía definitiva. [En línea] 2009. [Citado el: 20 de noviembre de 2015.] http://librosweb.es/libro/symfony_1_4/.

PRESSMAN, Roger. 2009. Ingeniería del Software. Un enfoque práctico. s.l. : McGraw-Hill/Interamericana, 2009.

Sparx. 2000-2007. Sparx Systems. Sparx Systems. [En línea] Sparx Systems, 2000-2007. [Citado el: 21 de enero de 2016.] http://www.sparxsystems.com.ar/resources/tutorial/use_case_model.html. ISBN.

uml-diagrams. 2010. uml-diagrams. uml-diagrams. [En línea] 2010. [Citado el: 14 de octubre de 2015.] <http://www.uml-diagrams.org/>.