Recibido: 21/11/2019
Aceptado: 12/01/2020
Publicado: 13/05/2020

Universidad&Ciencia
Vol. 9, No. 2, mayo-agosto (2020)
ISSN: 2227-2690 RNPS: 2450
http://revistas.unica.cu/uciencia

LENGUAJE DE MARCAS GENERALIZADO EN LA APLICACIÓN INFORMÁTICA MONOLINEALES VERSIÓN 7.0.0

EXTENSIBLE MARKUP LANGUAGE IN THE MONOLINEAL COMPUTER APPLICATION VERSION 7.0.0

Autores: Michael Rodríguez Álvarez
Lian Jesú Plasencia González
Institución: Empresa de Tecnología de la Información y Automática (ATI), Cuba
Correo electrónico: mralvarez@nauta.cu

RESUMEN

El Lenguaje de Marcas Generalizado es muy utilizado por los sistemas informáticos ya que permite estructurar información en un documento. El objetivo de la presente investigación es transformar los símbolos de los esquemas eléctricos en elementos dentro de un documento con Lenguaje de Marcas Generalizado que puedan ser enlazados con los elementos eléctricos registrados dentro de la base de datos del Sistema Integral de Gestión de Redes. Para ello la presente investigación se basa en el modelo ágil para el desarrollo de software y es SCRUM es el ciclo de vida de desarrollo de software a utilizar. Durante el desarrollo de la aplicación se utiliza el sistema de control de versiones de código abierto Git, también era necesario utilizar un lenguaje de programación simple, moderno, de propósito general y orientado a objetos, por tal motivo se selecciona al lenguaje de programación C#. Se utiliza el entorno de desarrollo integrado de Visual Studio, la biblioteca GoDiagram .NET, la herramienta Enterprise Architecture para el trabajo con el Lenguaje Unificado de Modelado. SQL Server es el sistema de bases de datos profesional de Microsoft y es utilizado para la persistencia de la información. La presente investigación se basa únicamente en un enfoque cualitativo ya que no pretende asociar las mediciones con números. La Aplicación Informática Monolineales representa un aporte significativo para la economía cubana pues sustituye importaciones al ser un producto informático nacional que está presente en todas las provincias del territorio cubano dentro de los Despachos Provinciales de Carga.

Palabras clave: Aplicación informática, Monolineales, XML, SIGERE.

ABSTRACT

Recibido: 21/11/2019
Aceptado: 12/01/2020
Publicado: 13/05/2020

ISSN: 2227-2690 RNPS: 2450
http://revistas.unica.cu/uciencia

The Extensible Markup Language is widely used by computer systems because it allows structuring information in a document. The objective of the present investigation is to transform the symbols of the electrical diagrams into elements within a document with Extensible Markup Language that can be linked to the electrical elements registered within the database of the Integrated Network Management System. For this, this research is based on the agile model for software development and is SCRUM the life cycle of software development was used. During the development of the application the Git open source version control system was used, it was also necessary to use a simple, modern, general purpose and object-oriented programming language, for this reason the C # programming language is selected. The integrated Visual Studio development environment, the GoDiagram .NET library, the Enterprise Architecture tool for working with the Unified Modeling Language is used. SQL Server is Microsoft's professional database system and it is used for information persistence. This research is based solely on a qualitative approach since it is not intended to associate the measurements with numbers. The Monolineales Computer Application represents a significant contribution to the Cuban economy because it substitutes imports due to it is a national computer product that is present in all the provinces of the Cuban territory within the Provincial Load Offices.

Keywords: Computer application, Monolineales, XML, SIGERE.

INTRODUCCIÓN

La utilización de las tecnologías de la información se hace cada día más extensa, lo cual viene dado por sus facilidades y bondades en cuanto al manejo de datos y prontitud de adquirir dichos datos que en la mayoría de los casos se encuentran remoto. Esto genera un avance y desarrollo para el país donde se utilice este tipo de tecnologías, lo cual permite la toma de decisiones de manera más breve comparado con la vía tradicional de manejar la información. En muchos casos se automatizan procesos que evitan la introducción de errores en los datos brindados por los sistemas.

La nación cubana no está ajena a la utilización de ese tipo de tecnologías, destinando recursos para la compra de equipos de cómputo y servidores para el

Recibido: 21/11/2019
Aceptado: 12/01/2020
Publicado: 13/05/2020

ISSN: 2227-2690 RNPS: 2450
http://revistas.unica.cu/uciencia

desarrollo y fomento de sistemas informáticos que contribuyan a la informatización de la sociedad. Además, en Cuba se han creado empresas para el desarrollo de aplicaciones informáticas que permiten la sustitución de importaciones, debido a que dichas aplicaciones informáticas son de producción nacional y realizadas con el esfuerzo de profesionales cubanos.

Entre estas empresas nacionales se encuentra la Empresa de Tecnología de la Información y Automática (ATI), la cual su producto líder es el Sistema Integral de Gestión de Redes (SIGERE).

El Sistema Integral de Gestión de Redes es un sistema informático de gestión con un componente técnico muy elevado. Tradicionalmente se ha considerado como un sistema tipo Sistema de Gestión de Distribución (Distribution Management System o DMS) en contraposición a los Sistemas de Gestión de Energía (Energy Management System o EMS) más enfocado en el despacho y control de la red fundamental y el manejo de la generación. (Fernández, 2008, p. 69)

La Aplicación Informática Monolineales es desarrollada por la Empresa de Tecnología de la Información y Automática (ATI) y utiliza la base de datos del Sistema Integral de Gestión de Redes con el fin de integrarse con las aplicaciones informáticas y sistemas informáticos que a él se vinculan. Esta aplicación informática permite crear los esquemas eléctricos (también llamados Monolineales) de la red eléctrica nacional, así como representar el estado operativo de la red mediante el coloreo dinámico de los objetos que la componen.

Los esquemas eléctricos requieren que sean guardados como un documento con el Lenguaje de Marcas Generalizado (XML) dentro de la base de datos del Sistema Integral de Gestión de Redes para su posterior utilización dentro del Despacho Provincial de Carga de cada provincia quienes son los encargados de operar de manera territorial el Sistema Electro-Energético Nacional (SEN) para mantener un servicio ininterrumpido y de calidad a todos los clientes, ya sean estatales o residenciales. También, los esquemas eléctricos son distribuidos hacia los niveles superiores de dirección de dicho Despacho para su revisión y análisis. Además, pueden ser utilizados por los especialistas afines al tema de manera aislada sin base

Recibido: 21/11/2019
Aceptado: 12/01/2020
Publicado: 13/05/2020

ISSN: 2227-2690 RNPS: 2450
http://revistas.unica.cu/uciencia

de datos donde estos esquemas eléctricos también son archivados como un documento con el Lenguaje de Marcas Generalizado.

El Dr. C. Jaime E. Villate Profesor Asistente del Departamento de Ingeniería Física de la Universidad de Oporto indica que el Lenguaje de Marcas Generalizado, es usado para estructurar información en un documento o en general en cualquier fichero que contenga texto, como por ejemplo ficheros de configuración de un programa o una tabla de datos. Apunta además de que es un estándar abierto y libre. (Villate, 2001, p. 1)

Afirma el Ingeniero de Software Arturo Granell Gorlanova de la Empresa Avaloq que el Lenguaje de Marcas Generalizado, fue desarrollado para definir lenguajes de marcas para almacenar datos de forma legible. (Granell, 2019, p. 22)

Los investigadores Paloma Galeote Bajo y David Díez Cebollero de la Universidad Carlos III de Madrid aclaran que, XML no define las etiquetas ni cómo se utilizan, sino que ofrece un escaso número de reglas sintácticas para poder crear documentos. Éste se propone como un lenguaje de bajo nivel para intercambio de información estructurada entre distintas plataformas. En su extenso análisis sobre el tema estos autores explican que un esquema XML define qué elementos puede contener un documento XML, cómo están organizados y qué atributos y qué tipo pueden tener sus elementos, pero la utilización de esquemas ofrece nuevas posibilidades en el tratamiento de los documentos ya que usan sintaxis de XML, permiten especificar tipos de datos y permiten crear nuevos elementos por lo que son extensibles. (Galeote, 2009, p. 21-22)

La razón por la cual es tan utilizado por los sistemas informáticos es debido a todas estas posibilidades que permite el Lenguaje de Marcas Generalizado.

Ejemplo de lo antes planteado se puede mencionar al Sistemas Web de Gestión de Emergencias el cual mediante la utilización del Lenguaje de Marcas Generalizado se puede proporcionar mecanismos de interoperabilidad que permite la gestión conjunta de situaciones de emergencia entre ARCE (Aplicación en Red para Casos de Emergencia) y SIGAME (Sistema de Gestión de Ayudas con Medios en Emergencias) de manera tal de que la información sobre la situación de emergencia

Recibido: 21/11/2019
Aceptado: 12/01/2020
Publicado: 13/05/2020

ISSN: 2227-2690 RNPS: 2450
http://revistas.unica.cu/uciencia

existente en uno de los sistemas se vea reflejada en el otro. (Galeote, 2009, p. 21-22)

Otro ejemplo bien difundido son los documentos SVG los cuales se pueden crear desde cero o a partir de un documento XML. Este tipo de documento SVG define tres tipos de objetos gráficos: vectoriales, (líneas, elipses, rectángulos, etc.) que pueden ser agrupados, formateados, transformados y compuestos para ser visualizados. También permite enlazar imágenes (GIF, JPEG, etc.) y añadir texto. SVG como tecnología XML es un código abierto y por tanto se puede editar y manipular desde cualquier editor de texto. (Estévez, 2019, p. 6)

Entre los sistemas informáticos vinculados con el tema de circuitos eléctricos está el Sistema de Definición de Pruebas de Circuitos Integrados el cual es un software que permite definir circuitos integrados, junto con la lógica de activación de cada pin de salida tanto en modo tabla de verdad como en modo cronograma, y almacenar el resultado en fichero. Utiliza el Lenguaje de Marcas Generalizado para almacenar la información sobre las características de los circuitos integrados. (Granell, 2019, p. vii)

Existe una gran cantidad y variedad de sistemas informáticos que utilizan el Lenguaje de Marcas Generalizado para el manejo de datos.

Por el análisis anterior se puede sintetizar que no existe una aplicación informática que permita el manejo de documentos con Lenguaje de Marcas Generalizado que tenga en cuenta los atributos de los símbolos de los esquemas eléctricos que se enlazan con los elementos eléctricos registrados dentro de la base de datos del Sistema Integral de Gestión de Redes.

Por tanto, se tiene la siguiente problemática: ¿Cómo transformar los símbolos de los esquemas eléctricos en elementos dentro de un documento con Lenguaje de Marcas Generalizado que puedan ser enlazados con los elementos eléctricos registrados dentro de la base de datos del Sistema Integral de Gestión de Redes?

El objetivo de la presente investigación es transformar los símbolos de los esquemas eléctricos en elementos dentro de un documento con Lenguaje de Marcas Generalizado que puedan ser enlazados con los elementos eléctricos registrados dentro de la base de datos del Sistema Integral de Gestión de Redes.

Recibido: 21/11/2019
Aceptado: 12/01/2020
Publicado: 13/05/2020

Universidad&Ciencia
Vol. 9, No. 2, mayo-agosto (2020)
ISSN: 2227-2690 RNPS: 2450
http://revistas.unica.cu/uciencia

MATERIALES Y MÉTODOS

Para dar solución al problema planteado, la presente investigación se basa en el modelo ágil para el desarrollo de software. Lo cual permite que se desarrolle en ciclos incrementales y rápidos. Esto da como resultado pequeñas versiones incrementales donde cada versión se basa en la funcionalidad anterior. Cada lanzamiento es probado a fondo para asegurar que la calidad del software se mantiene. Es específicamente SCRUM el ciclo de vida de desarrollo de software a utilizar (Ver Figura 1). Por lo cual se logra una satisfacción del cliente mediante entrega rápida y continua de software útil. Además, los clientes, desarrolladores y probadores interactúan constantemente entre sí. Se entrega software funcional con frecuencia, permite una atención continua a la excelencia técnica y buen diseño. Algo importante a destacar es la adaptación regular a las circunstancias cambiantes, incluso las modificaciones tardías en los requisitos son bienvenidas con la utilización de este tipo de modelo ágil. (Granell, 2019, p. 13)

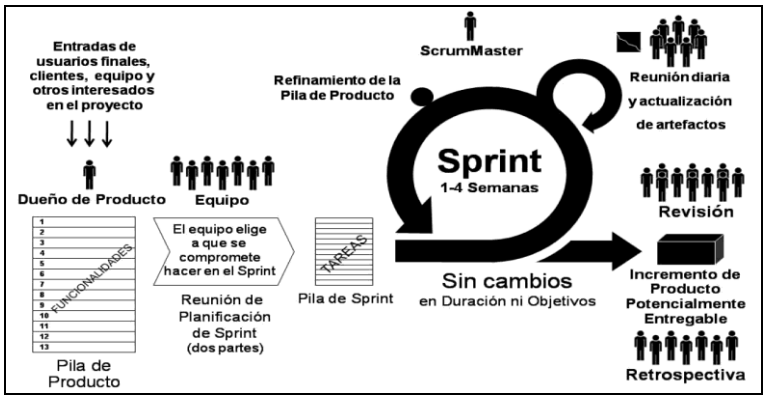


Figura 1. Ciclo de vida del Scrum. Fuente: (Deemer, 2009, p. 5)

Durante el desarrollo de la aplicación se requiere de un Sistema de Control de Versiones (SCV), herramienta que permite gestionar los cambios que se realizan sobre los elementos de un proyecto o repositorio, guardando así versiones del mismo en todas sus fases de desarrollo. En el caso de la presente investigación se utiliza el sistema de control de versiones de código abierto Git, quien a diferencia de otros SCV tiene una arquitectura distribuida, lo que significa que, en lugar de guardar todos los cambios de un proyecto en un único sitio, cada usuario contiene una copia

Recibido: 21/11/2019
Aceptado: 12/01/2020
Publicado: 13/05/2020

ISSN: 2227-2690 RNPS: 2450
http://revistas.unica.cu/uciencia

del repositorio con el historial de cambios completo del proyecto. Esto aumenta significativamente su rendimiento. (Sánchez, 2019, p. 2)

Para el desarrollo de la aplicación era necesario utilizar un lenguaje de programación simple, moderno, de propósito general y orientado a objetos. Por tal motivo se selecciona al lenguaje de programación C# por cumplir con estas características. Es importante destacar que sus implementaciones proporcionan soporte para los principios de ingeniería de software, como es la verificación de tipo fuerte, la verificación de límites de matriz, la detección de intentos de usar variables no inicializadas, y recolección automática de basura. La robustez del software, la durabilidad y la productividad del programador son importantes. (ECMA-334, 2006, p. xix)

El entorno de desarrollo integrado de Visual Studio es un panel creativo que se puede usar para editar, depurar y compilar código y, después, publicar una aplicación. Un entorno de desarrollo integrado (IDE) es un programa con numerosas características que se pueden usar para muchos aspectos del desarrollo de software. Más allá del editor estándar y el depurador que proporcionan la mayoría de IDE, Visual Studio incluye compiladores, herramientas de finalización de código, diseñadores gráficos y muchas más características para facilitar el proceso de desarrollo de software. Por tal motivo será utilizado este entorno de desarrollo integrado para la creación de la aplicación informática que dará solución a la problemática. (Microsoft, 2019, p. 1)

Northwoods Software es una empresa que tiene una vasta experiencia de más de 20 años desarrollando para los programadores bibliotecas para el trabajo con diagramas de forma rápida y sencilla, lo cual ahorra a los proyectos un valioso tiempo de desarrollo. Para crear diagramas potentes en formularios de tipo Windows Forms se utiliza la biblioteca GoDiagram.NET. La cual fue construida desde cero en C# con una gran variedad de características y diseños. Permite crear gráficos dirigidos y no dirigidos, diagramas de red, diagramas de flujo, diagramas de flujo de trabajo, diagramas de relación de entidades, aplicaciones BPMN y BPEL, organigramas, diagramas de circuitos, diagramas de árbol (por ejemplo, árboles de sintaxis, mapas de sitio), diagramas de flujo de datos, mapas mentales, diagramas

Recibido: 21/11/2019
Aceptado: 12/01/2020
Publicado: 13/05/2020

ISSN: 2227-2690 RNPS: 2450
http://revistas.unica.cu/uciencia

SCADA, aplicaciones de red inteligente, herramientas de plano-grama y mucho más. Esta biblioteca resulta adecuada para utilizarla en el proyecto de la presente investigación. (Northwoods Software, 2019, p. 1)

Para el trabajo con el Lenguaje Unificado de Modelado (UML, por sus siglas en inglés, Unified Modeling Language) el cual es muy necesario para un mejor entendimiento del proyecto de software que aborda la presente investigación, se utiliza la herramienta Enterprise Architecture de la compañía Sparx Systems. La razón fundamental de su uso viene dada a que soporta UML 2.0, genera de código, valida los modelos, soporta elementos fuera de los diagramas como por ejemplo los requisitos y además genera documentación. (Escalona, 2007, p. 4)

SQL Server es el sistema de bases de datos profesional de Microsoft y es utilizado para la persistencia de la información en el Sistema Integral de Gestión de Redes. Contiene una variedad de características y herramientas que se pueden utilizar para desarrollar y administrar bases de datos y soluciones de todo tipo basadas en ellas. El Motor de base de datos es el servicio principal para almacenar, procesar y proteger los datos. Asimismo, proporciona acceso controlado y procesamiento rápido de transacciones para cumplir los requisitos de las aplicaciones de base de datos más exigentes lo cual es una característica indispensable en la solución que se aborda en la presente investigación. La herramienta SQL Server Management Studio la cual es un entorno integrado para obtener acceso, configurar y administrar todos los componentes de SQL Server, será de mucha utilidad debido a que está diseñada para desarrollar y administrar objetos de base de datos. (Pérez, 2011, p. XIII-XV)

La presente investigación se basa únicamente en un enfoque cualitativo, debido a que no involucra la recolección de datos utilizando técnicas que no pretenden medir ni asociar las mediciones con números. Para ello se realiza una observación no estructurada de los procesos que se realizan dentro del Despacho Provincial de Carga de la provincia La Habana en Cuba, con el objetivo de entender e identificar aquellos procesos que son candidatos a informatizarse. Esto se complementa con entrevistas abiertas hacia los especialistas que en ese lugar laboran, para además tener en cuenta la evaluación de experiencias personales tanto de ellos y como de

Recibido: 21/11/2019
Aceptado: 12/01/2020
Publicado: 13/05/2020

ISSN: 2227-2690 RNPS: 2450
http://revistas.unica.cu/uciencia

los integrantes del equipo de desarrollo de software que se personan para realizar una discusión en grupo con el objetivo de definir la Pila del Producto que contendrá las funcionalidades de la aplicación informática lo cual genera un Acta de Reunión con las Tareas Técnicas a realizar en un periodo de tiempo determinado por los participantes. Se realiza además una revisión de documentos que contengan los procedimientos establecidos como normas dentro del despacho que se convertirán en las reglas del negocio para la aplicación informática. La utilización de estos métodos de la investigación científica será una práctica común y reiterativa durante el desarrollo de la aplicación informática. (Hernández, 2004, p. 15)

RESULTADO Y DISCUSIÓN

A partir de las funcionalidades contenidas dentro de la Pila del Producto, el equipo de desarrollo de software define de manera planificada y se compromete con la realización de las Tareas para el desarrollador (Issue para el trabajo con el Git) y el tiempo en horas estimado para cada una de ellas en la Pila de Sprint (Ver Figura 2).

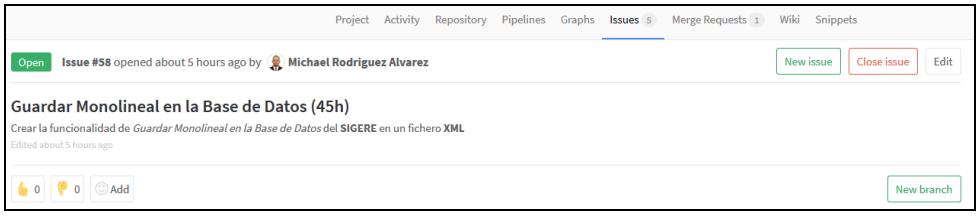


Figura 2. Tarea Guardar Monolineal en la Base de Datos asignada en GitLab.

Antes de programar la tarea se debe definir con el uso de la herramienta Enterprise Architecture y utilizando el Lenguaje Unificado de Modelado dentro del artefacto Modelo de Diseño el Diagrama de Casos de Uso del Sistema (Ver Figura 3) para lograr una visión más clara de la aplicación para su mantenimiento o desarrollo. Dentro de este diagrama se asocian los Casos de Uso a los Requisitos Funcionales o funcionalidades de la Pila del Producto para el caso de Scrum.

Recibido: 21/11/2019
Aceptado: 12/01/2020
Publicado: 13/05/2020

Universidad&Ciencia
Vol. 9, No. 2, mayo-agosto (2020)
ISSN: 2227-2690 RNPS: 2450
http://revistas.unica.cu/uciencia

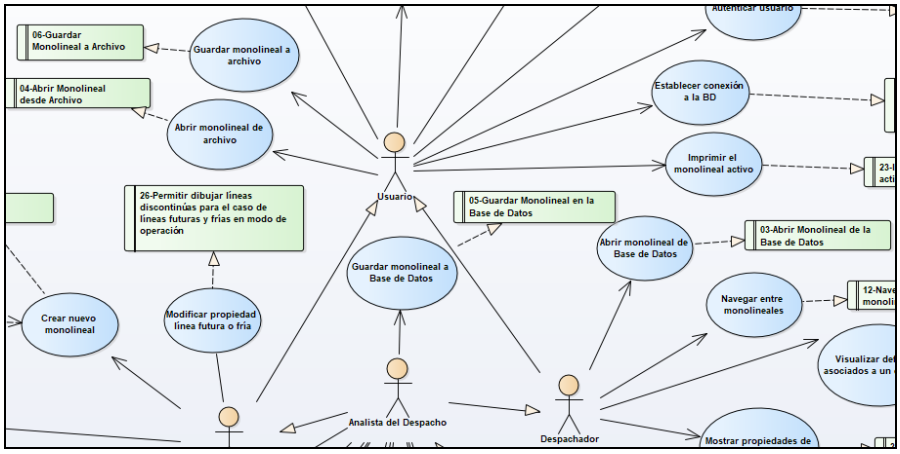


Figura 3. Fragmento del Diagrama de Casos de Uso del Sistema.

Se define la estructura de las tablas que contendrá la información relacionada con el Monolineal dentro de la Base de Datos del SIGERE. Para el caso de la Tarea "Guardar Monolineal en la Base de Datos" se define la tabla Monolineales (Ver Figura 4). Entre las columnas más importantes a destacar se encuentra la Versión, el Nombre, el Monolineal (que almacena el documento XML en la base de datos) y la Descripción. Las columnas NumAccion e Instalación se asocian con otras tablas de la base de datos del SIGERE, lo cual integra la Aplicación Informática Monolineales a dicho Sistema.

	Column Name	Data Type	Allow Nulls
🔑	Id_EAdministrativa	smallint	☐
🔑	NumAccion	int	☐
	Instalacion	varchar(7)	☐
	Version	smallint	☐
	Nombre	varchar(20)	☑
▶	Monolineal	image	☑
	Aprobado_por	smallint	☑
	Ultversion	bit	☑
	Ejecutadapor	smallint	☑
	Descripcion	varchar(250)	☑
	tipo	smallint	☐

Figura 4. Estructura de la tabla Monolineales dentro de la Base de Datos del SIGERE

Se procede a programar la funcionalidad desde el entorno de desarrollo integrado de Visual Studio utilizando el lenguaje de programación C# y la biblioteca GoDiagram .NET para el manejo de los gráficos y comportamientos de los monolineales. Dentro de esta biblioteca se encuentra el ensamblado Northwoods.Go.Xml, este permite al desarrollador leer y escribir documentos XML. El desarrollador controla

Recibido: 21/11/2019
Aceptado: 12/01/2020
Publicado: 13/05/2020

ISSN: 2227-2690 RNPS: 2450
http://revistas.unica.cu/uciencia

completamente el formato de estos documentos XML (la definición del tipo de documento o el esquema).

Se crea la clase TransformBase la cual hereda de GoXmlTransformer (que se encuentra en el ensamblado Northwoods.Go.Xml) la cual contiene los métodos llamados GoXmlWriter y GoXmlReader, utilizados para generar o analizar XML para objetos de una clase en particular. De la clase TransformBase heredan todos los elementos eléctricos que pueden ser guardados en el documento XML.

Luego se crea la clase llamada TransformElementoElectrico la cual representa un elemento eléctrico genérico y hereda de la clase TransformBase. Para cargar la información del Monolineal desde el documento XML, en esta nueva clase se sobrescribe el método ConsumeAttributes (Ver Figura 5) que está definido en la clase GoXmlTransformer. Se le pasa como parámetro un objeto el cual es convertido a tipo TElementoElectrico (clase que representa un elemento eléctrico como un nodo dentro del gráfico ya que hereda de la clase GoSimpleNode de la biblioteca GoDiagram .NET) y este posee atributos que son cargados a partir de la información que posee el documento XML. Para la lectura estos atributos y correspondiente conversión al tipo de dato desde el documento XML, GoXmlTransformer proporciona los métodos PointAttr (para puntos de coordenadas {x,y}), Int32Attr (para enteros), StringAttr (para cadenas de texto), ColorAttr (para colores) y BooleanAttr (para booleanos).

```
public override void ConsumeAttributes(object obj)
{
    base.ConsumeAttributes(obj);
    TElementoElectrico n = (TElementoElectrico)obj;

    n.Icon.Location = this.PointFAttr("xy", new PointF(100, 100));
    //n.Posicion = this.Int32Attr("position", 0);
    n.InitializeLabel(this.StringAttr("label", n.Text));
    n.LabelSpot = this.Int32Attr("labelspot", GoObjectSpot.BottomMiddle);
    n.Pen = this.PenAttr("pen", this.DefafultPen);
    n.BindCode = this.StringAttr("bindcode", "");
    n.ToolTipText = this.StringAttr("info", null);
    n.TextColor = this.ColorAttr("textcolor", Color.Black);
    n.LabelBackColor = this.ColorAttr("labelbackcolor", Color.Transparent);
    n.PanelInfo_Text = this.StringAttr("texto_panel", "");
    n.ShowLabel = this.BooleanAttr("show_label", true);
    if (n.Panel_Info != null)
      n.Panel_Info.BgColor = this.ColorAttr("colorfondo_panel", Color.Empty);
```

Figura 5. Fragmento del método ConsumeAttributes de la clase TransformElementoElectrico

De igual manera para el caso de escribir los atributos y valores dentro del documento XML se utiliza la clase TransformElementoElectrico en la cual se sobrescribe el método GenerateAttributes (Ver Figura 6) de la clase

Recibido: 21/11/2019
Aceptado: 12/01/2020
Publicado: 13/05/2020

Universidad&Ciencia
Vol. 9, No. 2, mayo-agosto (2020)
ISSN: 2227-2690 RNPS: 2450
http://revistas.unica.cu/uciencia

GoXmlTransformer. En este caso también funciona igual que el caso de ConsumeAttributes pero de manera inversa, o sea, se lee los atributos del objeto instanciado de la clase TElementoElectrico y se escribe en el documento XML. Para este objetivo la clase GoXmlTransformer utiliza el método WriteAttrVal la cual recibe como parámetro el nombre del atributo y el valor.

```
public override void GenerateAttributes(object obj)
{
    base.GenerateAttributes(obj);

    TElementoElectrico n = (TElementoElectrico)obj;
    this.WriteAttrVal("xy", n.Icon.Location);
    this.WriteAttrVal("position", n.Posicion);
    this.WriteAttrVal("textcolor", n.TextColor);
    this.WriteAttrVal("labelbackcolor", n.LabelBackColor);
    this.WriteAttrVal("pen", n.Pen);
    this.WriteAttrVal("escala", n.Escala);
    this.WriteAttrVal("anguloLabel", n.LabelAngle);
    this.WriteAttrVal("show_label",n.ShowLabel);
```

Figura 6. Fragmento del método GenerateAttributes de la clase TransformElementoElectrico

La clase TransformElementoElectrico contiene los atributos básicos de un elemento eléctrico, se pueden crear otras clases que hereden de esta para reutilizar estos atributos y adicionar otros más específicos. Dentro de las nuevas clases se puede volver a sobrescribir los dos métodos ConsumeAttributes y GenerateAttributes para la lectura y escritura de los nuevos atributos dentro del documento XML.

Para el almacenamiento del documento XML que representa el Monolineal dentro de la base de datos del SIGERE se dispone de un método llamado StoreBD (Ver Figura 7) que es ejecutado al presionar en la interfaz visual un botón para ese propósito. Este método está contenido dentro de la clase TDocument la cual se puede tratar como un monolineal con sus atributos, esta hereda de la clase GoDocument de la biblioteca GoDiagram.NET. En resumen, el código inicializa un objeto de tipo GoXmlWriter quien se encarga de generar el documento XML con los atributos de la clase TDocument o del Monolineal a guardar. Haciendo una llamada al método GuardarVersionMonolineal de la clase Consultas quien es la encargada del manejo con la base de datos o sea del Modelo (si se mira desde el punto de vista de arquitectura de software), a dicho método se le pasa los parámetros que serán registrados en la base de datos y los más importantes a destacar se encuentra newVersion (la nueva Versión), fIdMonolineal (el Identificador del Monolineal), el

Recibido: 21/11/2019
Aceptado: 12/01/2020
Publicado: 13/05/2020

Universidad&Ciencia
Vol. 9, No. 2, mayo-agosto (2020)
ISSN: 2227-2690 RNPS: 2450
http://revistas.unica.cu/uciencia

Nombre, la Descripción y por último data (el Monolineal transformado a documento XML).

```
public void StoreBD(string nombre, string descripcion, DateTime fecha, bool newVersion, bool vigente)
{
    StartTransaction();

    GoXmlWriter myWriter = new TXmlWriter();
    myWriter.RootElementName = "monolineal";
    this.RegisterTransformers(myWriter);
    myWriter.Objects = this;
    XmlDocument xml = myWriter.Generate();

    string data = xml.InnerXml;
    try
    {
        if ((fIdMonolineal == Constants.NullInt) || (!inBD))
        {
            fIdMonolineal = Consultas.GuardarMonolineal(fIdMonolineal, this.NombreMonolineal, this.CodigoInstalacion);
        }

        MON_Version version = Consultas.GuardarVersionMonolineal(newVersion, fIdMonolineal, nombre, descripcion, data,
```

Figura 7. Fragmento del método StoreBD de la clase TDocument.

De esta manera concluye el proceso de programación de la Tarea y se suben los cambios del código mediante el Sistema de Control de Versiones Git para su revisión y posterior integración mezclándose con la rama principal del proyecto. Se le realizan las Pruebas Alfa (Ver Anexo 1) por parte del desarrollador y por parte del implementador y las Pruebas Beta que son aceptadas por el Director de la Empresa encargada del desarrollo de la aplicación informática y el Coordinador del grupo de expertos que representa al cliente. Luego se entrega el incremento del producto al cliente para que este lo ponga en producción y explote las funcionalidades. En caso de existir errores o cambios en la funcionalidad se refina la Pila de Producto, logrando tener una aplicación informática lo más cercano a las peticiones y exigencias del cliente.

CONCLUSIONES

La implantación y uso de la Aplicación Informática Monolineales permite transformar los símbolos de los esquemas eléctricos en elementos dentro de un documento con Lenguaje de Marcas Generalizado que puedan ser enlazados con los elementos eléctricos registrados dentro de la base de datos del Sistema Integral de Gestión de Redes, lo cual establece una integración entre ambos. Ello contribuye en gran medida a llevar un registro de las personas que establecen nuevas versiones de los Monolineales y cuáles son las personas que lo aprueban para su utilización y sobre todo permite gestionar los permisos para dichas acciones mediante la Aplicación Informática Administración que pertenece al Sistema Integral de Gestión de Redes.

Recibido: 21/11/2019
Aceptado: 12/01/2020
Publicado: 13/05/2020

Se puede además vincular los Monolineales o esquemas eléctricos a las Instalaciones eléctricas registradas dentro del Sistema Integral de Gestión de Redes. Por último, es importante destacar que la Aplicación Informática Monolineales representa para la economía cubana un aporte significativo debido a que sustituye importaciones al ser un producto informático nacional que está presente en todas las provincias del territorio cubano dentro de los Despachos Provinciales de Carga.

REFERENCIAS BIBLIOGRÁFICAS

DEEMER, P. (2009). Información básica de SCRUM. En *Isentics*, p. 5, Scrum Training Institute, California.

ECMA-334 (2006). *C# Language Specification*, ECMA International, 4ta ed. Ginebra.

ESCALONA CUARESMA, M. (2007). Introducción a Enterprise Architecture. En *Departamento de Lenguajes y Sistemas Informáticos de la Universidad de Sevilla*, p. 4, Universidad de Sevilla, Sevilla.

ESTÉVEZ, E. (2019). Definición y visualización de los lenguajes gráficos de IEC 61131-3, basada en tecnologías XML. En *Comité Español de Automática*, p. 6. Disponible en: https://intranet.ceautomatica.es/old/actividades/jornadas/XXVIII/documentos/1742.pdf. Visitado el 10 de diciembre de 2019.

FERNÁNDEZ ÁLVAREZ, R. (2008). Propuesta de nueva visión del sistema de gestión de redes. En *Revista Científica Ingeniería Energética*, pp. 68-74, Centro de Investigaciones y Pruebas Electro-energéticas (CIPEL), La Habana.

GALEOTE BAJO, P. (2009). Definición de un modelo de intercambio de datos entre SIGEs. En *Biblioteca de la Universidad Carlos III de Madrid*, pp. 21-22. Departamento de Informática Universidad Carlos III, Madrid.

GRANELL GORLANOVA, A. (2019). Sistema de definición de pruebas de circuitos integrados. En *Archivo Digital UPM: Biblioteca de la Universidad Politécnica de Madrid*, p. 22. Disponible en: http://oa.upm.es/49031/1/TFC_ARTURO_GRANELL_GORLANOVA.pdf. Visitado el 10 de diciembre de 2019.

HERNÁNDEZ SAMPIERI, R. (2004). *Metodología de la investigación.* 1 ed. México : McGraw-Hill Interamericana.

Recibido: 21/11/2019
Aceptado: 12/01/2020
Publicado: 13/05/2020

Universidad&Ciencia
Vol. 9, No. 2, mayo-agosto (2020)
ISSN: 2227-2690 RNPS: 2450
http://revistas.unica.cu/uciencia

MICROSOFT (2019). Información general sobre Visual Studio. En *Documentos de Visual Studio*, p. 1. Disponible en: https://docs.microsoft.com/es-es/visualstudio/get-started/visual-studio-ide?view=vs-2019. Visitado el 10 de diciembre de 2019.

Northwoods Software (2019). GoDiagram for WinForms. En GoDiagram Advanced Controls for .NET Diagrams, p. 1. Disponible en: https://www.nwoods.com/products/godiagram/index.html. Visitado el 10 de diciembre de 2019.

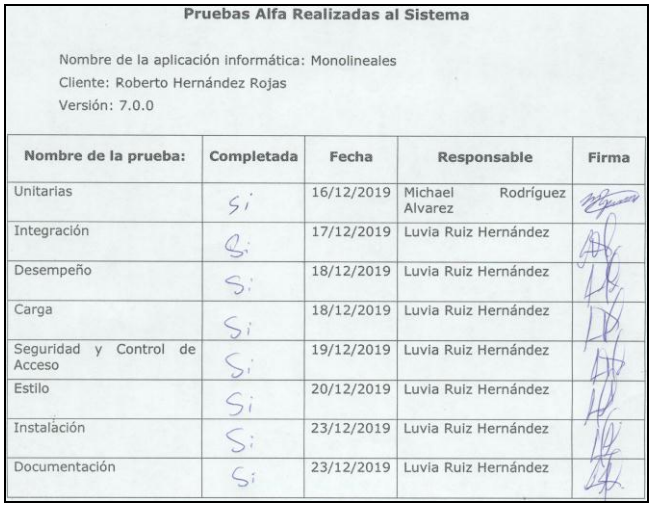
PÉREZ MARQUÉS, M. (2011). *SQL Server 2008 R2. Motor de base de datos y administración.* 1ra ed. Madrid : RC Libros.

SÁNCHEZ ALBERCA, A. (2019). Introducción a GIT. Sistema de control de versiones. En *Blog Personal Aprende con Alf*, p. 2. Disponible en: http://aprendeconalf.es/git/manual/manual-git.pdf. Visitado el 10 de diciembre de 2019.

VILLATE, J. (2001). Introducción al XML. En *Seminario sobre Programación en entorno GNU/Linux*, p. 1, Universidad de Oporto, Oporto.

ANEXO

Anexo 1. Pruebas Alfa para la Aplicación Informática Monolineales.



Pruebas Alfa Realizadas al Sistema

Nombre de la aplicación informática: Monolineales
Cliente: Roberto Hernández Rojas
Versión: 7.0.0

Nombre de la prueba:	Completada	Fecha	Responsable	Firma
Unitarias	Si	16/12/2019	Michael Rodríguez Alvarez	
Integración	Si	17/12/2019	Luvia Ruiz Hernández	
Desempeño	Si	18/12/2019	Luvia Ruiz Hernández	
Carga	Si	18/12/2019	Luvia Ruiz Hernández	
Seguridad y Control de Acceso	Si	19/12/2019	Luvia Ruiz Hernández	
Estilo	Si	20/12/2019	Luvia Ruiz Hernández	
Instalación	Si	23/12/2019	Luvia Ruiz Hernández	
Documentación	Si	23/12/2019	Luvia Ruiz Hernández	