

Universidad de las Ciencias Informáticas
Facultad 2



**Componente de visualización de estadísticas para los módulos
de Xilema GRHS en forma de gráficas.**

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

Autor: Handy Honderes Cruz
Tutores: Ing. Dianet Díaz Oduardo
Ing. Ramón Guzmán Alemañy
Cotutor: Ing. Lázaro Camacho Mesa

**“Año 60 de la Revolución”
La Habana, 11 Junio del 2018**



Si se puede soñar, se puede lograr

(Walt Disney)

Declaración de autoría

Declaro ser autor de la presente tesis y reconocemos 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 2018.

Handy Hondares Cruz

Firma del Autor

Ing. Dianet Díaz Oduardo

Firma del Tutor

Ing. Ramón Guzmán Alemañy

Firma del Tutor

Datos de Contacto

Tutora:

Ing. Dianet Díaz Oduardo. Profesor Instructor de la Facultad 2. Universidad de las Ciencias Informáticas.

Email: ddiaz@uci.cu.

Tutor:

Ing. Ramón Guzmán Alemañy. Recién graduado en adiestramiento del Departamento de Aplicaciones del Centro de Telemática de la Facultad 2. Universidad de las Ciencias Informáticas.

Email: ralemany@uci.cu.

Estudiante:

Handy Hondares Cruz de la Facultad 2. Universidad de las Ciencias Informáticas.

Email: hhondarez.estudiantes.@uci.cu.

Agradecimientos

A mis padres quiénes lo son todo para mí, por darme todo el amor y apoyo que un hijo puede recibir, por el sacrificio que han realizado, quiénes han hecho de mí la persona que soy actualmente, con mis valores, mis principios, mis ideales, mis defectos..., quiénes me han apoyado, perdonado, aconsejado..., quiénes su sacrificio y su dedicación es en sí, mi vida. Mami, gracias por brindarme tu cariño y el amor de madre que nunca me ha faltado, por insistir que siempre se puede, por ser mi estimulación, mi todo. Papá, por ser un ejemplo a seguir tanto en el plano personal como en el profesional, por darme tu cariño, tu apoyo, tu amistad y tus consejos. Papi quisiera ser como tú. Mi principal objetivo es lograr que se sientan orgullosos de mí.

A mi hermana, por tu apoyo y tu amor durante todos los años de mi carrera y de mi vida.

A mi novia, mi amiga, mi confidente, por tu apoyo en los momentos más difíciles, por su amor incondicional.

A mi familia en general, le agradezco su apoyo durante estos 5 años con el mismo amor que he recibido toda la vida de ustedes.

A mis tutores por todo el apoyo que me brindaron para poder llevar a cabo el desarrollo de este trabajo de curso.

A mi co-tutor por demostrarme que no necesitas conocer mucho a alguien para que te ayude.

A mis amigos más que mis amigos hermanos los cuales hemos estado durante 5 años de nuestras vida esforzándonos y dando lo mejor de nosotros para salir hacia adelante siempre juntos. Además por haber tenido el privilegio de conocerlos y estar siempre cuando los he necesitado.

A todas mis amistades no voy a mencionar nombre por temor a que alguien se me quede fuera. Pero gracias de verdad por haber compartido con ustedes.

A mis compañeros del grupo 2503, por todo lo compartido en estos cinco años y por ser tan buenos compañeros.

A todos los profesores que contribuyeron a mi desarrollo como profesional, que me apoyaron en los cinco años de la carrera les agradezco su dedicación y cariño, las horas que me brindaron de su tiempo para explicarme algún contenido o ayudarme en el desarrollo del presente trabajo.

A mi suegra por preocuparse por mí.

En sentido general, agradezco a todos aquellos que pusieron su granito de arena para que este día ocurriese, de corazón muchas gracias a todos.

Dedicatoria

Dedico esta tesis a mis padres.

RESUMEN

El surgimiento de la estadística sucedió a través de un proceso largo de desarrollo y de evolución, pasando, en un inicio, por la simple recolección de datos, hasta la compleja interpretación de los datos que se realiza hoy día. El análisis de los datos ha sido un tema de importancia para el hombre, ya que constantemente consultan la información para tomar decisiones, pero cuando se trata de grandes volúmenes de información se hace complejo este análisis. Para perfeccionar este proceso se utilizan técnicas que brinda la estadística permitiendo resumir, organizar y visualizar los datos en gráficos. La presente investigación permitió desarrollar el componente de visualización de estadísticas para los módulos del Gestor de Recursos de Hardware y Software (Xilema GRHS) en forma de gráfica. El mismo permite visualizar las estadísticas de los resultados que arrojan los inventarios de hardware y software en forma de gráfica. El desarrollo del componente contribuye a realizar resúmenes estadísticos, evaluar cómo se encuentra su infraestructura informáticamente y al proceso de la toma de decisiones. Se analizan las herramientas, tecnologías y metodologías de desarrollo con el fin de establecer la mejor selección para la construcción del componente, seleccionándose Proceso Unificado Ágil (AUP-UCI), como metodología, Python, PostgreSQL y Django para el desarrollo de la aplicación que permiten la creación de gráficos estadísticos. Se exponen los artefactos generados en las fases que establece la metodología de desarrollo de software seleccionada. Durante la elaboración del producto se sometió a pruebas para propiciar al cliente conformidad y seguridad en la aplicación desarrollada.

Palabras clave: gráficos, visualizar, estadísticas, toma de decisiones, Xilema GRHS.

SUMMARY

The emergence of statistics happened through a long process of development and evolution, passing, initially, by simple data collection, to the complex interpretation of the data that is done today. The analysis of the data has been a topic of importance for the man, since they constantly consult the information to make decisions, but when it comes to large volumes of information, this analysis becomes complex. To perfect this process, techniques are used that provide the statistics, allowing to summarize, organize and visualize the data in graphs. The present investigation allowed to develop the component of visualization of statistics for the modules of the Resource Manager of Hardware and Software (Xilema GRHS) in graphic form. It allows to visualize the statistics of the results that the inventories of hardware and software show in graphical form. The development of the component contributes to making statistical summaries, evaluating how your infrastructure is informally and the process of decision making. The tools, technologies and development methodologies are analyzed in order to establish the best selection for the construction of the component, selecting Agile Unified Process (AUP-UCI), as methodology, Python, PostgreSQL and Django for the development of the application that allow the creation of statistical graphs. The artifacts generated in the phases established by the selected software development methodology are exposed. During the elaboration of the product, it was subjected to tests to provide the client with compliance and security in the developed application.

Keywords: graphics, visualize, statistics, decision making, Xylema GRHS.

ÍNDICE

Introducción	12
Capítulo1: Fundamentación Teórica.....	16
Introducción	16
1.1 Conceptos asociados a la investigación.....	16
1.2 Ejemplos de gráficos estadísticos	16
1.2.1 Gráfico o diagrama de barras	16
1.2.2 Gráficos circulares y de anillo	16
1.2.3 Gráficos de líneas	17
1.2.4 Gráficos de columnas.....	17
1.3 Herramientas para graficar a nivel mundial	17
1.3.1 Herramientas privativas	17
1.3.2 Herramientas de código abierto	18
1.3.3 Herramientas para graficar en el ámbito nacional	20
1.4 Metodología de desarrollo	21
1.5 Librerías de apoyo para graficar	23
1.6 Lenguaje de programación del lado del servidor	23
1.7 Marco de trabajo de desarrollo web	24
1.8 Entorno Integrado de Desarrollo (IDE)	25
1.9 Herramienta CASE. Visual Paradigm	25
1.10 Sistema Gestor de Base de Datos	25
1.11 Conclusiones parciales	27
Capítulo 2: Análisis y Diseño del componente de visualización de estadísticas para los módulos de Xilema GRHS en forma de gráfica.....	28
Introducción	28
2.1 Modelo conceptual.....	28
2.1.1 Descripción del modelo conceptual	28
2.2 Propuesta de solución	29
2.3 Especificación de requisitos de software	30
2.3.1 Requisitos funcionales	31
2.3.2 Requisitos no funcionales.....	32
2.4 Diagrama de clases del diseño.....	37
2.4.1 Descripción de clases	38
2.5 Diagrama de despliegue	40
2.6 Conclusiones del capítulo	41
Capítulo 3: Implementación y Prueba	42

Introducción	42
3.1 Arquitectura de software	42
3.2 Patrones de arquitectura del componente	42
3.2.1 Patrón arquitectónico cliente servidor	43
3.2.2 Patrón arquitectónico Modelo Plantilla Vista	43
3.3 Patrones de diseño	46
3.3.1 Patrones General Responsibility Assignment Software Patterns (GRASP).....	46
3.3.2 Patrones de Diseño GOF	46
3.4 Pruebas del software	47
3.4.1 Pruebas unitarias	47
3.4.2 Pruebas de aceptación	48
3.5 Resumen de las pruebas de aceptación	49
3.6 Conclusiones	49
Conclusiones Generales.....	50
Referencias.....	52
Bibliografía	54
Anexo.....	57
Glosario de términos	69

ÍNDICE DE TABLAS Y FIGURAS

Tabla 1. Requisito no funcional de Confiabilidad.....	32
Tabla 2. Requisito no funcional de Confiabilidad.....	33
Tabla 3. Requisito no funcional de Usabilidad.....	34
Tabla 4. Requisito no funcional de Usabilidad.....	34
Tabla 5. Requisito no funcional de Eficiencia.	35
Tabla 6. Requisito no funcional de Hardware.....	35
Tabla 7. Requisito no funcional de Hardware.....	36
Tabla 8. Requisito no funcional de Portabilidad.	36
Tabla 9. Requisito no funcional de Portabilidad.	37
Tabla 10. Descripción de clases del diseño.	38
Tabla 11. Descripción de clases del diseño.	38
Tabla 12. Descripción de clases del diseño.	39
Tabla 13. Descripción de clases del diseño.	39
Tabla 14. Diseño del caso de prueba seleccionar tipo de gráfico.....	48
Tabla 15. Diseño del caso de prueba guardar imagen de la gráfica realizada.....	64
Tabla 16. Diseño del caso de prueba actualizar los datos de la gráfica.	64
Tabla 17. Diseño del caso de prueba graficar de acuerdo a la gráfica seleccionada.....	65
Tabla 18. Diseño del caso de prueba exportar a documento.	66
Tabla 19. Diseño del caso de prueba cambiar para tabla.....	66
Tabla 20. Graficar por un criterio de búsqueda.	67
Figura 1. Modelo conceptual.....	28
Figura 2.Propuesta de solución.....	29
Figura 3. Visualización de las estadísticas.....	30
Figura 4. Prototipo de interfaz seleccionar tipo de gráfico.....	32
Figura 5.Diagrama de clases del diseño.	37
Figura 6.Diagrama de despliegue.	40
Figura 8.Modelo cliente-servidor.....	43
Figura 9.Modelo vista plantilla de Django.....	44
Figura 10.Resultado de la primera iteración de las pruebas unitarias.	47
Figura 11.Resultado de la segunda iteración de las pruebas unitarias.....	48
Figura 12.Resultado de las pruebas de aceptación en cada iteración.....	49
Figura 13. Prototipo de interfaz guardar imagen.....	58
Figura 14. Prototipo de interfaz actualizar los datos de la gráfica.....	59
Figura 15. Prototipo de interfaz graficar de acuerdo a la gráfica seleccionada.....	60

Figura 16. Prototipo de interfaz exportar a documento..... 61

Figura 17. Prototipo de interfaz cambiar para tabla..... 62

Figura 18. Prototipo de interfaz graficar por un criterio de búsqueda 63

Introducción

El surgimiento de la estadística sucedió a través de un proceso largo de desarrollo y de evolución, pasando, en un inicio, por la simple recolección de datos, hasta la compleja interpretación que se realiza hoy día. Se hace muy difícil definir cuáles son sus orígenes, pues desde el nacimiento mismo de la civilización ya se empleaban formas simples de estadística, los hombres hacían representaciones gráficas y cuentas, tanto en paredes como en pieles de animales.

Esta ciencia mediante la recolección de datos cuantitativos brinda información que es empleada en las empresas para apoyar el proceso de toma de decisiones, permite conocer a una sociedad, información que es utilizada por los gobiernos para mejorar la situación de la población. También es utilizada en el área de la investigación científica ya que los datos estadísticos son el resultado de varios casos de entre los cuales se toma un promedio, por lo que sirve para demostrar que determinado caso posee un resultado particular; se emplea con propósitos descriptivos (tabulación de datos, representaciones gráficas, entre otros). Otras aplicaciones de las técnicas estadísticas son en mercadotecnia, contabilidad, control de calidad, análisis de resultados en deporte, la medicina, la educación y en la toma de decisiones.

Esta última en la mayoría de las empresas se hace un proceso engorroso sin el apoyo de una aplicación informática. Mediante el uso de computadoras y sistemas informáticos se obtienen resultados más precisos en numerosos cálculos que resultan complejos para el cerebro humano. También los investigadores pueden mejorar la calidad de los datos originales y la interpretación de los resultados. Con el avance en el desarrollo de herramientas informáticas, existen librerías y complementos que permitan realizar representaciones gráficas. Sin embargo, el problema se centra en elegir la más adecuada para cada situación, en cuanto a su licencia, costo, rendimiento, dependencias, entre otras, que permitan la interpretación de los datos para un análisis más efectivo. Estas herramientas contienen un conjunto de configuraciones y estilos visuales para los usuarios, lo que las hacen útiles en el aporte de recursos gráficos para la representación de valores numéricos y contribuyendo al análisis y la obtención de conclusiones.

Una de las principales formas de comprender el material disponible es la utilización de representaciones gráficas. Un gráfico estadístico es una representación visual de una serie de datos estadísticos. Es una herramienta muy eficaz, ya que un gráfico: capta la atención del lector, presenta la información de forma sencilla, clara y precisa, no induce a error, facilita la comparación de datos y destaca las tendencias y las diferencias, ilustra el mensaje. [1]

La Universidad de las Ciencias Informáticas (UCI) es una de las instituciones cubanas dedicadas al desarrollo de software. La misma cuenta con una red de centros productivos que apoyan la informatización de la sociedad cubana. Uno de estos centros es el de Telemática (TLM) donde se desarrolla el sistema Gestor de Recursos de Hardware y Software (Xilema GRHS), el cual es un sistema que realiza el inventario de la información de los componentes de hardware y software en una red de computadoras. El mismo realiza el inventario del hardware con que cuenta el equipo de cómputo:

memoria RAM, motherboard, procesador, dispositivos de almacenamiento, CD-ROM, teclado, ratón, impresora, escáner, multifuncional y monitor; así como elementos del software como son: nombre de la computadora, dominio, localización, sistema operativo instalado, fabricante y versión del BIOS. Además detecta la existencia de antivirus Kaspersky y Segurmática, usuarios del sistema y controladores instalados.

El sistema es capaz de detectar cambios no autorizados, configurables según el tipo (hardware o software), componentes, estado (agregado o eliminado) y el nivel (alto, medio o bajo según la clasificación definida por la institución) según las necesidades de la institución y lanzar alertas de correo a los interesados. Además Xilema GRHS permite conocer la localización del dispositivo inventariado según el lugar y la subred a la que pertenece. Xilema GRHS hasta el momento ha sido desplegado en la Universidad de las Ciencias Informáticas con más de 3 500 computadoras reportándose. En esta institución el sistema fue incluido entre las políticas de seguridad. En la sede central de la Fiscalía General de la República y la Empresa Comercializadora del Cemento fue desplegado también, para 2 000 y 200 computadoras respectivamente.[2]

Como parte de la utilización de los clientes del sistema (Xilema GRHS) se han detectado algunas oportunidades de mejoras en el software. A la hora de analizar los datos que devuelve sobre los inventarios, las tablas son muy engorrosas con varios campos y a la hora de hacer resúmenes estadísticos de este tipo, es más útil a los directivos observarlo en forma de gráficas lo cual ahora el sistema no cuenta con esa posibilidad. El mismo maneja una considerable cantidad de información, debido a esta situación el sistema muestra estadísticas cuantitativas y la información actualmente se visualiza por medio de tablas.

Existe actualmente una solución generada que grafica un conjunto de estadísticas cuantitativas utilizando gráficos de pastel, lineal y barra, de manera estática y general. El mismo utiliza un cuadro de mando para el sistema Xilema GRHS donde para observar los gráficos de las estadísticas que arroja el sistema hay que ir al cuadro de mando y de ahí visualizar las estadísticas de cada módulo, donde se define la solución que abarque los requisitos solicitados por el usuario y a sDashboard como herramienta para graficar. Dicha solución no se ha integrado a Xilema GRHS porque en la nueva versión del sistema que se va a comercializar se ha modificado el modelo de base de datos, la comunicación cliente – servidor y además se migró todo el sistema a una nueva versión del framework Django y se han agregado nuevos menús. Además para visualizar las estadísticas hay que seleccionar el cuadro de mando para de ahí observar los gráficos de las estadísticas de cada módulo ya que en el componente de visualización de estadísticas para los módulos de Xilema GRHS en forma de gráfica ya se podrán observar los gráficos de las estadísticas desde el módulo seleccionado. Actualmente no es posible graficar estadísticas teniendo en cuenta criterios de búsquedas seleccionados por el usuario que contribuyan a la toma de decisiones de los usuarios.

Como resultado del análisis de la situación planteada anteriormente surge el siguiente **problema de investigación**: ¿Cómo facilitar a los usuarios de Xilema GRHS el entendimiento de los datos que arroja el inventario de hardware y software?

Se define como **objeto de estudio** de la presente investigación, el proceso de visualización de estadísticas.

Para dar solución al problema de la investigación definido anteriormente se propone como **objetivo general**, desarrollar el componente de visualización de estadísticas para los módulos de Xilema GRHS en forma de gráficas.

El **campo de acción** a partir del objeto de estudio comprende, la gestión de la visualización de estadísticas para los módulos de Xilema GRHS en forma de gráficas.

Para dar cumplimiento al objetivo general de la presente investigación se definen las siguientes **tareas de investigación**:

1. Elaboración de la fundamentación teórica del trabajo para garantizar la base teórica necesaria para el desarrollo de la investigación.
2. Análisis de las herramientas y tecnologías que permitan representar la información en forma de gráficos con el objetivo de seleccionar la más adecuada para la presente investigación.
3. Selección de la metodología y herramientas a utilizar para la implementación.
4. Análisis y diseño del componente para su posterior implementación.
5. Implementación del componente de visualización de estadísticas para los módulos de Xilema GRHS en forma de gráficas.
6. Realización de pruebas para la validación del funcionamiento del componente de visualización de estadísticas para los módulos de Xilema GRHS en forma de gráficas.

Para el desarrollo de esta investigación y el logro de su objetivo, se utilizan los siguientes métodos de investigación científica:

Métodos Teóricos

Analítico sintético:

Permitirá analizar las teorías y los documentos referentes al componente de visualización de estadísticas para los módulos de Xilema GRHS en forma de gráficas, ayudando de esta forma la extracción de los elementos más importantes relacionados con la representación gráfica.

Histórico-lógico:

Posee importancia para la elaboración de la fundamentación teórica de la investigación, pues permite estudiar lo más notable en el plano teórico acerca de las herramientas informáticas que permitan graficar. Conocer los tipos de gráficos y sus funcionalidades.

Métodos Empíricos:

Entrevista:

Este método se utilizará para el desarrollo del componente, detectando las necesidades del cliente a través del intercambio con él, para identificar los principales problemas existentes y así poder satisfacer sus necesidades y obtener un software con la calidad requerida. Se realizan una serie de entrevistas específicamente a especialistas del proyecto Xilema GRHS, para poder entender mejor la arquitectura modelo-vista-controlador y los patrones de diseño a usar.

La presente investigación está compuesta por tres capítulos. A continuación, se resume el contenido de cada uno de ellos:

En el **Capítulo 1** “Fundamentación Teórica”. Se aborda los conceptos y definiciones investigadas durante el análisis de las tendencias actuales. Incluye además un estudio de las herramientas utilizadas para el desarrollo del componente, el estudio del estado del arte de los temas relacionados con el trabajo que se desarrolla, así como las tecnologías y lenguajes existentes útiles para el proceso de implementación.

En el **Capítulo 2** “Análisis y diseño del componente de visualización de estadísticas para los módulos de Xilema GRHS en forma de gráficas”, en este capítulo se realiza el análisis y diseño del componente de visualización de estadísticas para los módulos de Xilema GRHS en forma de gráficas. Para desarrollar estas etapas del proceso de desarrollo de software, se plantea una representación visual de las clases conceptuales del negocio que se deben analizar, a través del modelo conceptual. Posteriormente se hace un levantamiento de los requisitos funcionales y no funcionales necesarios para dar cumplimiento al objetivo general de la investigación. Además se muestran los diagramas de clases del diseño y el diagrama de despliegue.

En el **Capítulo 3** “Implementación y Pruebas”, en este capítulo se define la arquitectura y los patrones de diseño utilizados en el componente a desarrollar. Se muestra la organización de los componentes y sus relaciones lógicas mediante el diagrama de despliegue. Además, se evalúa la calidad de la solución final mediante la realización de pruebas, con el objetivo de solucionar errores y no conformidades presentes en la implementación.

Capítulo1: Fundamentación Teórica

Introducción

En el siguiente capítulo se expone la fundamentación teórica del componente a desarrollar y se brinda un análisis de los principales conceptos asociados al componente de visualización de estadísticas para los módulos de Xilema GRHS en forma de gráficas. Como aspectos esenciales se profundiza en el estudio de los sistemas similares, así como la selección de la metodología y el lenguaje a emplear para dar solución al objetivo general planteado.

1.1 Conceptos asociados a la investigación

Una de las acciones más importantes en las empresas es el control de las estadísticas para conocer cómo se comportan determinados parámetros en las entidades. La misma es una rama de las matemáticas que recopila, organiza, analiza e interpreta los datos obtenidos de un problema estadístico, para obtener conocimiento de los hechos pasados, para prever situaciones futuras y tomar decisiones en base a las experiencias.[3]

Una de las maneras de visualizar las estadísticas es utilizando gráficos. Los gráficos son representaciones visuales de un análisis de datos. Independientemente de su complejidad o abundancia, los datos pueden reestructurarse en medidas estadísticas para ofrecer un panorama más claro. Los gráficos estadísticos simples se usan como apoyos de mercadotecnia en anuncios o letreros. Los gráficos más complejos, que no sólo son productos de las clases de matemáticas, se utilizan como herramientas culturales y empresariales. Existen diversos tipos de gráficos que permiten presentar los datos de forma que resulten significativos para los usuarios. La ventaja de trabajar expresando información en diferentes tipos de gráficas estadísticas, es que todas ellas darán información clara y rápida del conjunto de datos obtenidos en el estudio o investigación en cuestión. Un detalle importante a señalar es que existen varios tipos de gráficas estadísticas.[4]

1.2 Ejemplos de gráficos estadísticos

1.2.1 Gráfico o diagrama de barras

Un gráfico de barras, suele expresar mediante la elevación de barras de diferente color (pueden ser horizontales) aquella información que intenta dilucidar un solo aspecto entre un grupo de personas encuestadas. Depende de cómo haya sido graduado el eje vertical "y", se expresará en distintas unidades o valores el impacto de los resultados en cuestión. Puede usarse para representar porcentajes, pero en esta circunstancia suele ser otro el tipo de gráfico elegido. [5]

1.2.2 Gráficos circulares y de anillo

Los gráficos circulares son útiles para resaltar proporciones. Usan segmentos circulares para mostrar la relación de las partes con todo el conjunto. Para resaltar los valores reales, use otro tipo de gráfico, como el gráfico apilado. Los gráficos circulares trazan una única serie de datos. Si tiene que trazar varias series

de datos, utilice un gráfico 100% apilado. Puede seleccionar los formatos siguientes: estándar, de anillo o tridimensional.[5]

1.2.3 Gráficos de líneas

Los gráficos de líneas son útiles para mostrar tendencias en el tiempo y para comparar muchas series de datos. Los gráficos de líneas trazan los datos en puntos regulares conectados con líneas. Puede seleccionar los formatos siguientes: estándar o tres dimensiones.[5]

1.2.4 Gráficos de columnas

Los gráficos de columnas son útiles para comparar datos discretos. Los gráficos de columnas usan marcadores de datos verticales para comparar valores individuales. Puede seleccionar los formatos siguientes: estándar, apilado, 100% apilado o tridimensional.[5]

Conclusiones de los diferentes tipos de gráficas

Los diferentes tipos de gráficos en lo general muestran la facilidad de manejar, supervisar el buen funcionamiento de las empresas y se puede detectar rápidamente cualquier anomalía que ocurra en la institución. Las gráficas son herramientas que se utilizan en función de representar resultados de una forma visible para apreciar, mejor términos estadísticos cualquiera que sea su aplicación, desde lo simple a lo abstracto de lo sencillo a lo complejo para hacer más comprensible un proceso. Permiten a los directivos hacer resúmenes estadísticos, evaluar cómo se encuentra su infraestructura informáticamente y ayuda al proceso de toma de decisiones en las empresas. Una de las principales formas de comprender el material disponible es la utilización de representaciones gráficas.

1.3 Herramientas para graficar a nivel mundial

Con el fin de seleccionar la herramienta informática más adecuada para graficar un conjunto de estadísticas cuantitativas y lograr una solución satisfactoria, se llevó a cabo un estudio de las principales herramientas que existen a nivel mundial. Cada uno tiene sus características y puntos de vistas para enfrentar al problema. Se puede afirmar que muchas de estas herramientas poseen elementos de valor que pudieran ser de interés para la propuesta de solución en la presente investigación. Por esta razón se hace necesario contar con un estudio de las mismas.

1.3.1 Herramientas privativas

PChart:

Es un marco de orientación de clase PHP¹ diseñado para crear gráficos con alias. Los datos pueden ser recuperados de consultas SQL². Este proyecto está todavía en desarrollo.

¹ *Hypertext Preprocessor*

² *Structured Query Language*

Desarrolla gráficas estáticas, generando un archivo de imagen con formato PNG³. Además, proporcionan la sintaxis de codificación orientada a objetos y está plenamente en consonancia con los nuevos estándares de la web que le permite mejorar sus aplicaciones web 2.0. Puede generar líneas, barras, áreas y circulares con sus clases.

Es liberado bajo dos modelos de licencia. La primera es la licencia GNU⁴ GPL versión 3 que permite integrar libremente esta biblioteca en las aplicaciones, modificar el código y redistribuirlo y la segunda licencia si la aplicación tiene fines comerciales, se debe de obtener una licencia comercial.[6]

Highcharts:

Es una biblioteca de gráficos escrita en JavaScript, que ofrece una manera de añadir gráficos interactivos a un sitio o aplicación web. Tiene tipos de gráficos como son de línea, área, columna, barras, circulares, de dispersión, medidores angulares, burbuja, diagrama de caja, barras de error, embudo, cascada y tipos de gráficos polares. Funciona en todos los navegadores móviles y de escritorio moderno, como Internet Explorer desde la versión 6. En iOS⁵ y Android. Navegadores estándar SVG (gráficos vectoriales escalables) utilizan para el procesamiento de gráficos.

Highcharts se basa únicamente en las tecnologías del navegador nativo y no requiere complementos secundarios como Flash o Java. El núcleo highcharts.js. Se puede utilizar en un proyecto personal o sin fines de lucro bajo la forma gratuita de la licencia 3.0 de Creative Commons Reconocimiento-No comercial.[7]

Google Chart:

La API de gráficos de Google es una herramienta que permite crear fácilmente un gráfico a partir de algunos datos e introducirlos en una página web. Estos datos y parámetros se introducen en forma de una petición HTTP⁶ y Google devuelve una imagen PNG de la tabla. Ofrece la posibilidad de visualizar datos en el sitio web. La galería gráfica provee diferentes tipos de gráficos listos para su uso: desde gráficos de barras o líneas simples a mapas de árboles. Los gráficos se representan mediante la tecnología HTML5⁷ / SVG (gráficos vectoriales escalables) para garantizar la compatibilidad entre navegadores.[8]

1.3.2 Herramientas de código abierto

Charts:

Es una librería la cual permite realizar gráficos y visualizar datos en 6 formas diferentes, cada uno de ellos animados, con variadas opciones de personalización y extensiones de interactividad. Además, utiliza el elemento canvas de HTML5. El cual está disponible en todos los navegadores modernos.

³ Portable Network Graphics

⁴ General Public License

⁵ iPhone OS

⁶ HyperText Transfer Protocol

⁷ *HyperText Markup Language*

Chart.js es modular, y cada uno de los tipos de gráficos se ha dividido, por lo que se puede cargar solo el tipo de gráfico que se necesita para el trabajo. Chart.js es de código abierto y disponible bajo la licencia MIT (Massachusetts Institute of Technology) lo que significa que impone muy pocas limitaciones en la reutilización. La licencia MIT permite reutilizar software dentro de software propietario lo que permite su modificación.[9]

Documentos Data-Driven (D3):

Es una librería de JavaScript para la manipulación de documentos basados en datos. D3 ayuda a la manipulación y envío de los datos usando HTML, SVG y CSS⁸. El énfasis de D3 en estándares web muestra todas las capacidades de los navegadores modernos sin obligarse a usar un marco exclusivo, que combina componentes de visualización de gran alcance y un enfoque impulsado por los datos a la manipulación DOM. D3 soporta navegadores como Firefox, Chrome, Safari, Opera, Android y iOS.

Permite enlazar datos arbitrarios a un Document Object Model (DOM), y aplicar transformaciones por datos al documento. Resuelve el problema de manipulación eficiente de los documentos basados en datos. Es flexible exponiendo todas las capacidades de los estándares web como CSS3, HTML5 y SVG. Es extremadamente rápido, tiene un apoyo a grandes conjuntos de datos y comportamientos dinámicos de interacción y animación. Contiene un estilo funcional que permite la reutilización de código a través de una variada colección de componentes, la cual no es una nueva representación gráfica a diferencia de procesamiento, el vocabulario de las marcas viene directamente de los estándares web: HTML, SVG y CSS. Puede usar efectos de filtro de material compuesto, trazos discontinuos y de recorte. [10]

Fusion Charts Free:

Fusion Charts Free es una librería para crear gráficos animados e interactivos en Flash para aplicaciones web y desktop. Anima sus aplicaciones convirtiendo datos monótonos para un visual agradable al usuario. Fusion Charts se puede integrar con una gran variedad de tecnologías web como ASP⁹, ASP.NET, PHP, JSP¹⁰, Cold Fusion, Ruby on Rails, Python o incluso hasta páginas HTML sencillas. Funciona con todos los bancos de datos, incluyendo Oracle, MySQL, PostgreSQL, entre otros.

Ofrece más de 45 tipos de gráficos 2D/3D, incluyendo líneas, áreas, barras, columnas, pastel, combinación, dispersión de burbujas, gráficos de desplazamiento. Los aspectos funcionales y estéticos de cada trabajo pueden ser extensivamente personalizados utilizando las herramientas profesionales de Fusion Charts. No se requieren conocimientos de Flash para usar Fusion Charts. Está bajo la licencia de la GPL y MIT por lo que puede utilizarse, modificarse y redistribuirse.[11]

⁸ Cascading Stylesheets

⁹ Active Server Pages

¹⁰ Java Server Pages

JQuery sparklines:

Es un complemento JQuery que genera mini gráficos (pequeños gráficos en línea) directamente en el navegador utilizando datos suministrados, o por medio de JavaScript. Este complemento es compatible con la mayoría de los navegadores modernos y ha sido probado con Firefox 2 +, Safari 3 +, Opera 9, Google Chrome e Internet Explorer 6, 7, 8, 9 y 10, así como iOS y Android.

Sparklines pretende ser lo suficientemente pequeño como para caber junto a una línea de texto, para dar una impresión rápida de una tendencia o patrón. No tiene la configuración de los gráficos de tamaño completo. Desde la versión 2.0 se puede obtener sobre los mini gráficos los datos subyacentes.

El complemento no tiene soporte para añadir texto a un mini gráfico, más allá de la proporcionada por la información de herramientas. Puede usar HTML y CSS para agregar texto apropiado al lado del mini gráfico. Además, se puede utilizar la librería Flot y el complemento jqPlot para generar gráficos más grandes. Está liberado y está bajo la licencia BSD¹¹ Nueva.[12]

SDashboard:

Es un marco de trabajo en JavaScript, contiene un complemento de JQuery que convierte una matriz de objetos en un cuadro de mando. Cada objeto de la matriz se representa como un widget del panel que puede intercalar sus posiciones. SDashboard se ha construido en apoyo a la prestación de los datos de tablas y gráficos flot2. También tiene soporte para detectar eventos tales como fila de la tabla y datos del gráfico.

Tiene como dependencias:

- JQuery.
- jQuery UI.
- DataTables (necesario para los widgets de tabla).
- Gráficos Flotr2 (necesario para los widgets de gráficos).

Está bajo la licencia de MIT de código abierto.[13]

1.3.3 Herramientas para graficar en el ámbito nacional

En el desarrollo de la investigación se encontró una solución desarrollada en la empresa de Tecnologías de la Información para la Defensa (XETID), la cual cuenta con una herramienta que se encarga del diseño de gráficas para el apoyo a la toma de decisiones. Esta herramienta es un servidor gráfico que trabaja con 3 librerías fundamentales para la representación de los datos, las mismas son: pChart, JqPlot y Fusion charts Free. Esta herramienta genera soluciones de gran calidad, pero su integración a la plataforma del sistema Xilema GRHS traería grandes cambios en el diseño y en la arquitectura del mismo, siendo no factible para su utilización.

Actualmente existe el módulo de presentación de resultado de inventario de hardware y software en el sistema de Xilema GRHS es una aplicación que utiliza un cuadro de mando para el sistema Xilema

¹¹ Berkeley Software Distribution

GRHS, donde se define la solución que abarque los requisitos solicitados por el usuario y a sDashboard como herramienta para graficar, esta solución generada grafica un conjunto de estadísticas utilizando gráficos de pastel, lineal y barra, de manera estática y general.[14] Dicha solución no se ha integrado ya que se han agregado nuevos menús. Además para visualizar las estadísticas hay que seleccionar el cuadro de mando para de ahí observar los gráficos de las estadísticas de cada módulo ya que en el componente de visualización de estadísticas para los módulos de Xilema GRHS en forma de gráfica ya se podrán observar los gráficos de las estadísticas desde el módulo seleccionado. Actualmente no es posible graficar estadísticas teniendo en cuenta criterios de búsquedas seleccionados por el usuario que contribuyan a la toma de decisiones de los usuarios.

Los cambios realizados al sistema han impedido la integración de la solución de forma rápida y sencilla a Xilema GRHS. De esta solución ya existente se va a utilizar a sDashboard como herramienta para graficar las estadísticas que muestra Xilema GRHS. Del presente módulo se reutilizó la herramienta de graficar estadísticas y se utilizó parte de su código por estar escrita en Python, utilizar a Django como marco de trabajo y estar bajo una licencia de código abierto.

Conclusiones del estudio de herramientas para graficar.

Luego de un estudio de las principales herramientas libres y privadas que permiten la representación de los datos en forma de gráfica, fueron descartadas varias herramientas por sus características como son JQuery sparklines, que solo permite crear gráficos pequeños y su estilo visual es bajo. Maani y Fusion Charts Free requiere de un navegador web con Flash Player 10 o posterior y utilizan XML¹² y SWF¹³, lo cual trae consigo una dependencia más para el sistema Xilema GRHS. Flot, Flotr2, gRaphaël, Protovis y Charts tienen gráficos de baja calidad y estilo visual. Google Chart, Highcharts, pChart están bajo licencias privativas que no permiten fines comerciales.

Seleccionando para el desarrollo del componente el marco de trabajo (sDashboard) y en el mismo integrar el complemento JQuery (Jqplot). Cumpliendo con las características de sus licencias que son de código abierto, además no trae ningún efecto negativo en la integración del sistema Xilema GRHS. Brinda numerosas opciones de estilo de gráficos como sombras, espaciados, orientación del texto y color a las áreas del gráfico. Provee de ejes con formato personalizable y permite manipular fácilmente la información de los gráficos.

1.4 Metodología de desarrollo

1.4.1 AUP-UCI

El Proceso Unificado Ágil de Scott Ambler Agile Unified Process (AUP) en inglés es una versión simplificada del Proceso Unificado de Rational (RUP). Este describe de una manera simple y fácil de

¹² Lenguaje de Etiquetado Extensible

¹³ Small Web Format -formato web pequeño

entender la forma de desarrollar aplicaciones de software de negocio usando técnicas ágiles y conceptos que aún se mantienen válidos en RUP.

Al no existir una metodología de software universal, ya que toda metodología debe ser adaptada a las características de cada proyecto (equipo de desarrollo, recursos, etc.) exigiéndose así que el proceso sea configurable, se decide hacer una variación de la metodología AUP, de forma tal que se adapte al ciclo de vida definido para la actividad productiva de la UCI. Una metodología de desarrollo de software tiene entre sus objetivos aumentar la calidad del software que se produce, de ahí la importancia de aplicar buenas prácticas, para ello se apoyan en el Modelo CMMI-DEV¹⁴ v1.3. El cual constituye una guía para aplicar las mejores prácticas en una entidad desarrolladora. Estas prácticas se centran en el desarrollo de productos y servicios de calidad.

De las 4 fases que propone AUP (Inicio, Elaboración, Construcción, Transición) se decide para el ciclo de vida de los proyectos de la UCI mantener la fase de Inicio, pero modificando el objetivo de la misma, se unifican las restantes 3 fases de AUP en una sola, a la que llamaremos Ejecución y se agrega una fase de Cierre.

Inicio

Durante el inicio del proyecto se llevan a cabo las actividades relacionadas con la planeación del proyecto. En esta fase se realiza un estudio inicial de la organización cliente que permite obtener información fundamental acerca del alcance del proyecto, realizar estimaciones de tiempo, esfuerzo y costo y decidir si se ejecuta o no el proyecto.

Ejecución

En esta fase se ejecutan las actividades requeridas para desarrollar el software, incluyendo el ajuste de los planes del proyecto considerando los requisitos y la arquitectura. Durante el desarrollo se modela el negocio, obtienen los requisitos, se elaboran la arquitectura y el diseño, se implementa y se libera el producto. Durante esta fase el producto es transferido al ambiente de los usuarios finales o entregado al cliente. Además, en la transición se capacita a los usuarios finales sobre la utilización del software.

Cierre

En esta fase se analizan tanto los resultados del proyecto como su ejecución y se realizan las actividades formales de cierre del proyecto.

AUP-UCI propone para el ciclo de vida de los proyectos en la UCI tener siete disciplinas, los flujos de trabajo son los siguientes: Modelado de negocio, Requisitos, Análisis y diseño, Implementación, Pruebas internas, Pruebas de liberación y Pruebas de aceptación. Además está compuesta por cuatro escenarios:

¹⁴ Capability Maturity Model Integration

El escenario número uno propone a los proyectos que modelan el negocio con casos de usos del negocio, que solo pueden modelar el sistema con casos de usos del sistema.

El escenario número dos propone modelar el negocio con un modelo conceptual y el sistema con casos de uso del sistema.

El escenario número tres propone modelar el negocio con descripción de proceso de negocio, junto al modelo conceptual y el sistema mediante la descripción de requisitos por proceso.

El escenario número cuatro propone no modelar el negocio y describir el sistema mediante historias de usuario.

Se selecciona para esta investigación el escenario número 3 pues este aplica a los proyectos que hayan evaluado el negocio a informatizar y como resultado obtengan un negocio con procesos muy complejos, independientes de las personas que los manejan y ejecutan, proporcionando objetividad, solidez, y su continuidad. Se debe tener presente que este escenario es muy conveniente si se desea representar una gran cantidad de niveles de detalles y la relaciones entre los procesos identificados.[15]

1.5 Librerías de apoyo para graficar

- **ReportLab:**

Es una librería de software que permite crear directamente documentos en formato (PDF¹⁵), utilizando el lenguaje de programación Python. PDF es el estándar mundial para documentos electrónicos. Es compatible con la impresión de alta calidad y es totalmente portátil en todas las plataformas.[16] Es utilizado para generar los reportes con extensión PDF de los gráficos del componente.

- **JqPlot:**

JqPlot es un complemento de gráficos para el marco de trabajo JQuery. Produce gráficos de líneas, barras y circulares con muchas características, como son numerosas opciones de estilo gráfico, ejes con formato personalizable. Además, tiene hasta 9 ejes Y. Permite el cálculo automático de la línea de tendencia y de la información sobre herramientas y puntos de datos resaltados. Contiene parámetros por defecto para facilitar su uso. JqPlot es un proyecto de código abierto con licencia dual bajo las MIT y GPL versión 2. Se elige la licencia según la adaptación al proyecto en desarrollo.

Este complemento tiene amplias funcionalidades sobre los ejes (X, Y), estilos de sombras, líneas en el plano y efectos, los cuales permiten controlar eventos para una mejor personalización de los componentes del gráfico, permitiendo una representación agradable al usuario.[17]

1.6 Lenguaje de programación del lado del servidor

- **Python 2.7:**

Python es un lenguaje de programación multipropósito de alto nivel, simple y sencillo de aprender además es libre. Se propone su utilización pues brinda muchas funcionalidades para aplicaciones web

¹⁵ Portable Document Format

que trabajen a un nivel bien cercano al sistema operativo, además posee mecanismos eficientes para su integración con bases de datos[18]. Se decidió utilizar Python porque Xilema GRHS está implementado sobre este lenguaje, además el componente a implementar se va a integrar a este sistema y debe ser desarrollado bajo sus mismas tecnologías. En el centro de TLM se definió este lenguaje de programación para sus productos, lo que beneficia que cualquier persona pueda darle mantenimiento al sistema o programar nuevas funcionalidades cuando sea necesario.

Lenguaje de programación del lado del cliente.

- **HTML5**

HTML5 es la última evolución de la norma que define el lenguaje de marcas de hipertexto (HTML por sus siglas en inglés). Se trata de una nueva versión del lenguaje HTML, con nuevos elementos, atributos y comportamientos. Diseñado para ser utilizado por todo tipo de desarrolladores. Describe la estructura y el contenido semántico de un documento web [19]. Se utilizará el lenguaje HTML5 para el diseño de las interfaces del componente, siendo este el utilizado en el sistema Xilema GRHS actualmente.

- **JavaScript**

JavaScript es un lenguaje de programación interpretado que se utiliza principalmente para crear páginas web dinámicas. JavaScript, es un lenguaje de script multiplataforma basado en objetos. Es un lenguaje ligero, está diseñado para una fácil incrustación en otros productos y aplicaciones, tales como los navegadores web[20]. Se utiliza para construir los gráficos de cada módulo del sistema Xilema GRHS.

- **JQuery**

Jquery es una biblioteca JavaScript rápida, pequeña y con varias funciones, facilita la creación de aplicaciones del lado del cliente. Hace las cosas como documento HTML recorrido y la manipulación, el control de eventos, animación y Ajax más simple con un API fácil de usar que funciona a través de una multitud de navegadores. Con una combinación de flexibilidad y extensibilidad, además permite manipulación de documentos, gestión de eventos y animaciones. Está bajo la licencia MIT y la licencia GNU versión 2 la cual permite el uso en cualquier tipo de plataforma, personal o comercial. Además elimina la recarga constante de páginas mediante la creación de un elemento intermedio entre el usuario y el servidor[19]. Se utiliza para manejar todas las operaciones de las acciones del cliente.

1.7 Marco de trabajo de desarrollo web

- **Django 1.8**

Django es un framework de desarrollo web sobre Python que permite desarrollar rápidamente aplicaciones web. Define una forma de desarrollar software que cuenta con 3 capas donde el Modelo es la capa de acceso a los datos, la plantilla es capa de presentación y la vista es la capa lógica del negocio. Se centra en automatizar todo lo posible[21]. Es el framework de desarrollo que se utilizará para la implementación del componente por ser actualmente el que se emplea en el sistema Xilema GRHS.

1.8 Entorno Integrado de Desarrollo (IDE)

Un Entorno Integrado de Desarrollo (IDE de su significado en inglés Integrated Development Environment), es un conjunto de herramientas que ha sido creado como un programa de aplicación, es decir, consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica. Un IDE provee un marco de trabajo amigable para la mayoría de los lenguajes de programación y en algunos casos puede funcionar como un sistema en tiempo de ejecución en donde se permite utilizar el lenguaje de programación en forma interactiva, sin necesidad de trabajo orientado a archivos de texto. [22]

- **PyCharm 2017**

PyCharm es un IDE o entorno de desarrollo integrado multiplataforma utilizado para desarrollar en el lenguaje de programación Python. Proporciona análisis de código, depuración gráfica, integración con VCS / DVCS y soporte para el desarrollo web con Django, entre otras bondades. PyCharm es desarrollado por la empresa JetBrains y debido a la naturaleza de sus licencias tiene dos versiones, la *Community* que es gratuita y orientada a la educación y al desarrollo puro en Python y la *Professional*, que incluye más características como el soporte a desarrollo web. Los desarrolladores utilizan al Pycharm para el desarrollo del componente Xilema GRHS.

1.9 Herramienta CASE. Visual Paradigm

Es una herramienta de diseño UML¹⁶ para facilitar el desarrollo de software. Soporta lenguajes de modelado clave de la industria y estándares. Visual Paradigm for UML es un potente, multiplataforma y, sin embargo el más fácil de usar y visual de modelado UML y herramienta CASE¹⁷. Visual Paradigm UML ofrece a los desarrolladores de software de la plataforma de desarrollo de borde de corte para construir aplicaciones de calidad más rápida y mejor [23]. Se utilizará para el diseño de los prototipos de interfaz de usuario del componente de visualización de estadísticas de los módulos de Xilema GRHS en forma de gráficas.

1.10 Sistema Gestor de Base de Datos

Un sistema gestor de base de datos se define como el conjunto de programas que administran y gestionan la información contenida en una base de datos. Ayuda a realizar las siguientes acciones:

- Definición de los datos
- Mantenimiento de la integridad de los datos dentro de la base de datos
- Control de la seguridad y privacidad de los datos
- Manipulación de los datos

¹⁶ lenguaje unificado de modelado

¹⁷ Computer Aided Software Engineering

Se trata de un conjunto de programas no visibles al usuario final que se encargan de la privacidad, la integridad, la seguridad de los datos y la interacción con el sistema operativo. Proporciona una interfaz entre los datos, los programas que los manejan y los usuarios finales. Cualquier operación que el usuario hace contra la base de datos está controlada por el gestor. El gestor almacena una descripción de datos en lo que llaman diccionario de datos, así como los usuarios permitidos y los permisos. Tiene que haber un usuario administrador encargado de centralizar todas estas tareas. Se compone de tres lenguajes: lenguaje de definición de datos, lenguaje de manipulación de datos y lenguaje de consulta.[24]

- **PostgreSQL9.8**

PostgreSQL es la base de datos relacional de código abierto más avanzada del mundo. Distribuida bajo licencia BSD (Berkeley Software Distribution por sus siglas en inglés) y con su código fuente disponible libremente. PostgreSQL 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. Sus características técnicas la hacen una de las bases de datos más potentes del mercado. Las funciones de formateo de PostgreSQL proporcionan un poderoso conjunto de herramientas para convertir varios tipos de datos (fecha / hora, entero, punto flotante, numérico) a cadenas formateadas y para convertir cadenas formateadas a tipos de datos específicos.[25] Además de todas las características antes mencionadas, es el que utiliza el sistema Xilema GRHS en la actualidad.

- **PgAdmin III**

PgAdmin III es una aplicación gráfica para gestionar el gestor de bases de datos PostgreSQL, siendo la más completa y popular con licencia Open Source. Está escrita en C++, lo que permite que se pueda usar en Linux, FreeBSD, Solaris, Mac OSX y Windows. Es capaz de gestionar versiones a partir de la PostgreSQL 7.3 ejecutándose en cualquier plataforma, así como versiones comerciales de PostgreSQL como Pervasive Postgres, EnterpriseDB, Mammoth Replicator. PgAdmin III está diseñado para responder a las necesidades de todos los usuarios, desde escribir consultas SQL simples hasta desarrollar bases de datos complejas. El interfaz gráfico soporta todas las características de PostgreSQL y facilita enormemente la administración. La aplicación también incluye un editor SQL con resaltado de sintaxis, un editor de código de la parte del servidor, un agente para lanzar scripts programados, soporte para el motor de replicación Slony-I. La conexión al servidor puede hacerse mediante conexión TCP/IP¹⁸

¹⁸ Protocolo de Control de Transmisión/Protocolo de Internet

o Unix Domain Sockets (en plataformas *nix), y puede encriptarse mediante SSL¹⁹ para mayor seguridad. La interfaz gráfica soporta todas las características de PostgreSQL y hace simple la administración.[26]

1.11 Conclusiones parciales

En el desarrollo del capítulo se analizaron los elementos teóricos que sirven de sustento a la investigación, del cual se puede concluir que:

De las metodologías de desarrollo de software se escogió AUP-UCI porque es la metodología utilizada en el centro de TLM por lo que se decide trabajar sobre ese escenario, con el fin de garantizar que la documentación que se genere como parte de la realización del componente pueda ser reutilizada por el proyecto. El estudio realizado permitió el análisis de las principales herramientas libres y privadas que permiten la representación de los datos en forma de gráfica, fueron descartadas varias herramientas por sus características, se seleccionó la más adecuada que no trae ningún efecto negativo en la integración del sistema Xilema GRHS. Luego del estudio de las herramientas propuestas se pudo evidenciar que las mismas son correctas para el futuro desarrollo del sistema por ser estas tecnologías multiplataforma, libres y que son compatibles entre ellas. Es necesario la creación de un componente que contribuya a mejorar la toma de decisiones en el proyecto Xilema GRHS.

¹⁹Secure Sockets Layer es un protocolo diseñado para permitir que las aplicaciones para transmitir información de ida y de manera segura hacia atrás.

Capítulo 2: Análisis y Diseño del componente de visualización de estadísticas para los módulos de Xilema GRHS en forma de gráfica

Introducción

En este capítulo se abordan los aspectos relacionados con el dominio de la investigación y la propuesta del trabajo del componente de visualización de estadísticas para los módulos de Xilema GRHS en forma de gráficas. Se crea el diagrama de modelo conceptual. Se plantean los requerimientos funcionales, sus prototipos de interfaz y los requisitos no funcionales para el correcto funcionamiento. Se modela el diagrama de clases del diseño y el diagrama de despliegue.

2.1 Modelo conceptual

El modelo conceptual es una representación que se realiza para entender el proyecto donde se está trabajando, cuando no se tiene una visión clara de los procesos que se realizan en el mismo. Se utiliza para capturar y expresar el entendimiento obtenido en un área bajo análisis como paso previo al diseño de un sistema. Contiene conceptos que estarán asociados tanto a su definición natural como al papel que juegan desde el punto de vista informático.

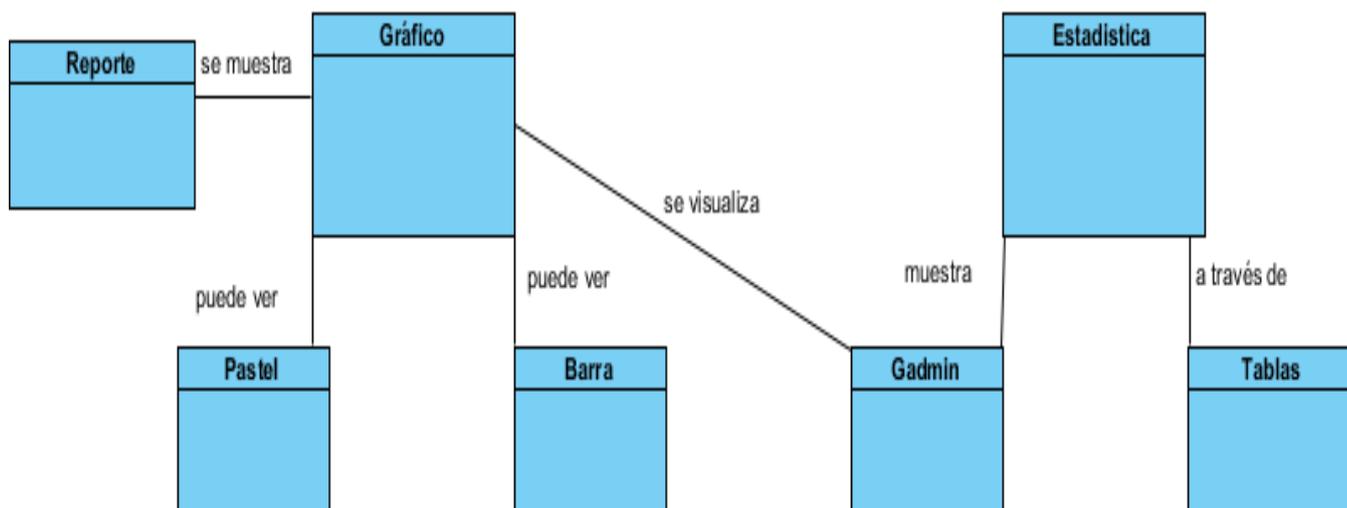


Figura 1. Modelo conceptual

2.1.1 Descripción del modelo conceptual

En el sistema Xilema GRHS maneja una considerable cantidad de estadísticas, actualmente estas estadísticas pueden consultarse mediante tablas paginadas a través del subsistema web denominado Gadmin (Interfaz visual del servidor web). Por lo que el componente de visualización de estadística de

los módulos del sistema Xilema GRHS va a mostrar todas las estadísticas a través de gráficos, pueden ser de barra y de pastel. Se podrán realizar reportes en PDF y PNG de los gráficos y cambiar la información de los gráficos a tablas. Los gráficos se visualizaran a través del Gadmin.

2.2 Propuesta de solución

Para un mejor entendimiento se explicará brevemente cómo funciona el Gestor de Recursos de Hardware y software.

Xilema GRHS está compuesto de tres aplicaciones: Gclient, Gserver y Gadmin. La aplicación Gclient es la encargada de obtener las propiedades de las piezas y programas instalados en una computadora. La aplicación Gserver es el centralizador de inventarios que recibe la información enviada por los recolectores de Gclient, notifica las alertas a los interesados cuando ocurren incidencias. Por su parte, la aplicación Gadmin, es la interfaz de administración encargada de visualizar y gestionar toda la información recolectada por cada uno de los clientes.

La propuesta de solución consiste en un componente que grafica las estadísticas que devuelve los inventarios de hardware y software. El sistema maneja una considerable cantidad de información, debido a esta situación el sistema muestra estadísticas y la información actualmente se visualiza por medio de tablas. Las tablas son muy engorrosas con varios campos y a la hora de hacer resúmenes estadísticos de este tipo a los directivos le es más útil observarlo en forma de gráficas.

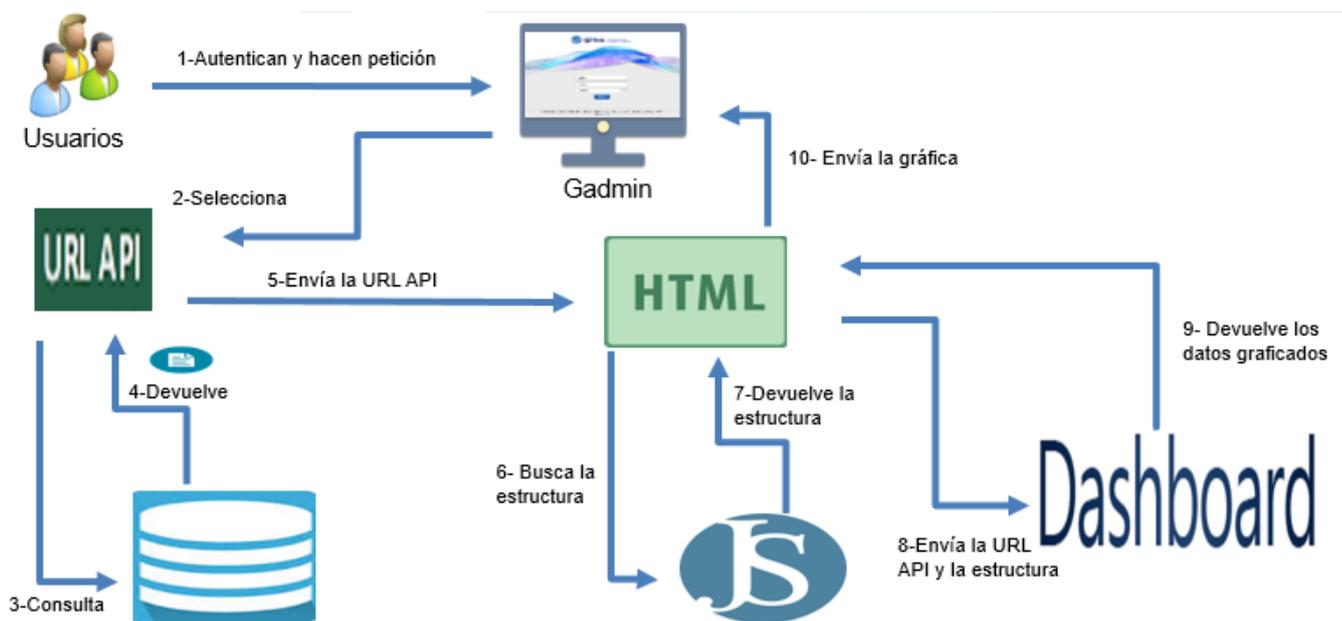


Figura 2.Propuesta de solución.

Funcionamiento del componente de visualización de estadísticas para los módulos de Xilema GRHS en forma de gráfico:

1. Los usuarios se autentican y realizan peticiones a través del Gadmin.
2. El Gadmin selecciona la url API adecuada.
3. La URL API consulta con la base de datos para obtener los datos.

4. La URL API agencia los datos.
5. A través de la URL API se envían los datos hacia el HTML.
6. El HTML busca su estructura con el JS.
7. El JS le envía la estructura que debe tener el HTML.
8. El HTML mediante la URL API envía los datos y la estructura a la herramienta sDashboard.
9. sDashboard devuelve al HTML los datos graficados.
10. El HTML envía en forma de gráfica los datos al Gadmin.

A continuación se muestra cómo el usuario visualizará esas estadísticas y qué acciones podrá realizar a través del Gadmin.

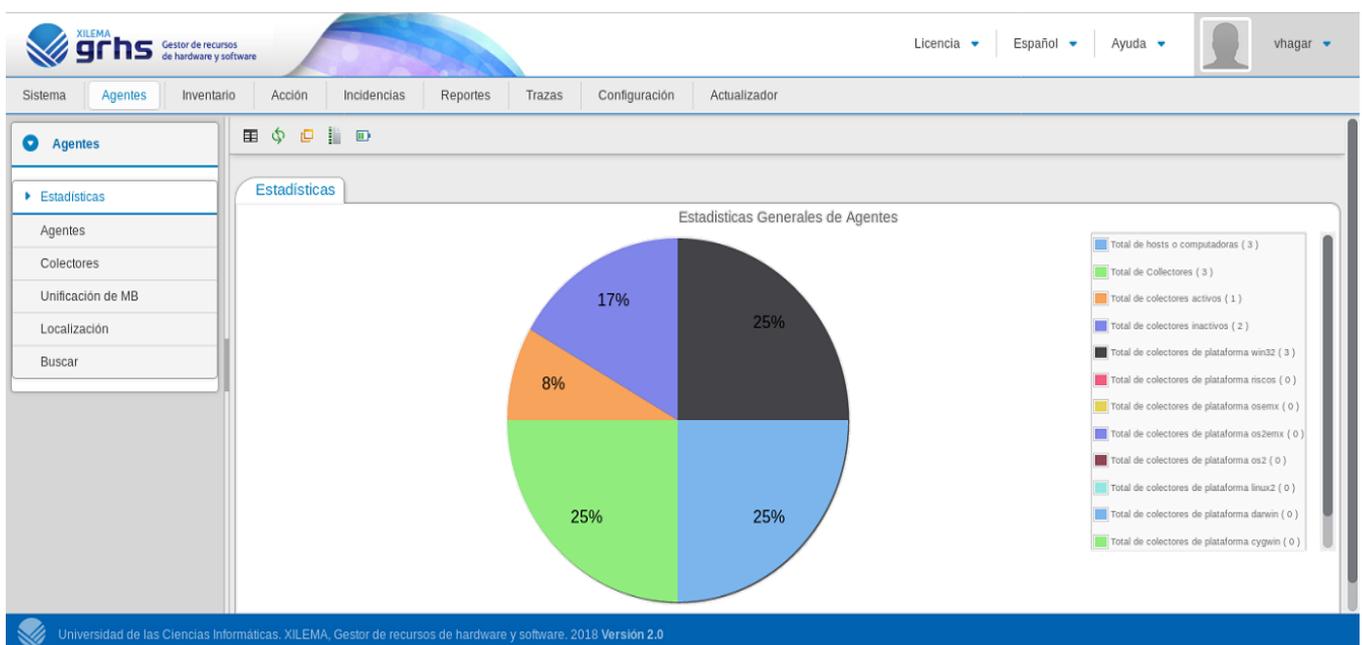


Figura 3. Visualización de las estadísticas.

Cuando los usuarios se autentiquen en Xilema GRHS mediante el Gadmin podrán:

1. Visualizar y gestionar las estadísticas en forma de gráfico, que pueden ser de pastel o de barra.
2. Actualizar los datos de las gráficas.
3. Generar reportes en formato PDF con una descripción.
4. Podrán guardar la imagen de la gráfica realizada en forma PNG.
5. Podrán visualizar además las estadísticas en forma de gráfico a partir de criterios de búsquedas.
6. Además pueden visualizar las estadísticas en forma de tabla.

2.3 Especificación de requisitos de software

Los requisitos para un sistema son la descripción de los servicios proporcionados por el sistema y sus restricciones operativas. Estos requisitos reflejan la necesidad de los clientes de un sistema que ayude a resolver algún determinado problema como el control de un dispositivo, hacer un pedido o encontrar

información. El proceso de descubrir, analizar, documentar y verificar esos servicios y restricciones es denominado ingeniería de requisitos.[27]

2.3.1 Requisitos funcionales

Los requisitos funcionales de un sistema son los que describen las capacidades o funciones que el componente debe cumplir y los servicios que de él se esperan, a continuación las funcionalidades definidas por el cliente y la descripción del requisito seleccionar tipo de gráfica el resto de las descripciones se encuentran en Anexo.

- **RF-1:** Seleccionar tipo de gráfica.
- **RF-2:** Guardar imagen de la gráfica realizada.
- **RF-3:** Actualizar los datos de la gráfica.
- **RF-4:** Graficar de acuerdo a la gráfica seleccionada.
- **RF-5:** Exportar a documento.
- **RF-6:** Cambiar para tabla.
- **RF-7:** Graficar por un criterio de búsqueda.

Especificación de requisito: seleccionar tipo de gráfica.

Descripción textual del requisito

Precondiciones	El usuario que desee seleccionar un tipo de gráfica debe estar autenticado en el sistema Xilema GRHS 2.0.
Flujo de eventos	
Flujo básico Seleccionar tipo de gráfica	
1.	Selecciona el módulo del cuál desea visualizar las estadísticas en gráficos.
2.	Se elige el modo en que se deseen ver los datos. Los datos se pueden visualizar a través de gráficas de pastel o barra. Si no se muestran los datos graficados ver flujo alternativo 1.a
Pos-condiciones	
1	Se selecciona el tipo de gráfica que desee.
Flujos alternativos	
Flujo alternativo 1.a No hay conexión con la base de datos(BD) de GRHS	
1.	El sistema muestra un mensaje. "No existen datos para mostrar".
Pos-condiciones	

1.	Se selecciona el tipo de gráfica.	
Validaciones		
1	NA	
Conceptos	Gráfico	
Requisitos especiales	Usabilidad, Eficiencia	
Asuntos pendientes	NA	

Prototipo elemental de interfaz gráfica de usuario

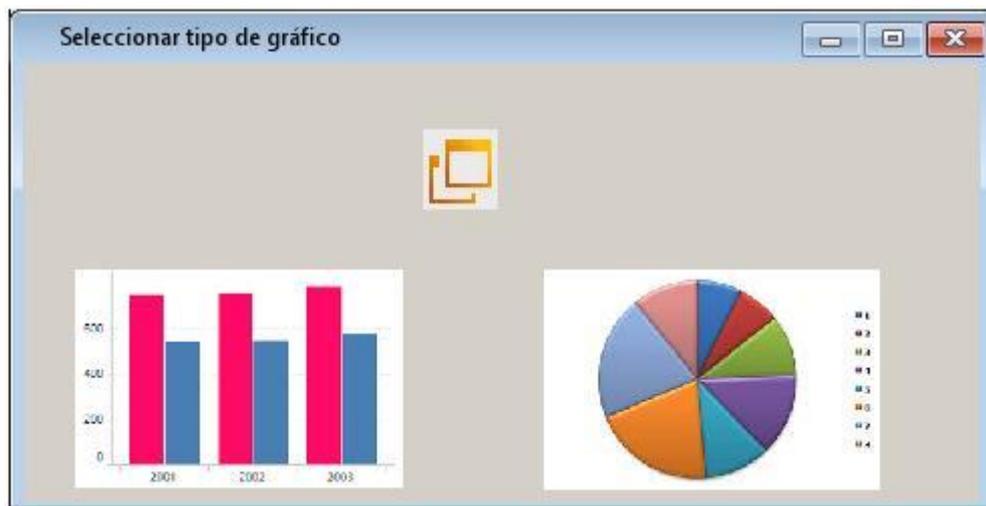


Figura 4. Prototipo de interfaz seleccionar tipo de gráfico

2.3.2 Requisitos no funcionales

Los requisitos no funcionales son aquellos que definen las restricciones del sistema, son propiedades, cualidades que el sistema debe poseer. Representan las características del producto.[27]

Tabla 1. Requisito no funcional de Confiabilidad.

Atributo de Calidad	Confiabilidad
Sub-atributos/Sub-características	Tolerancia a fallos
Objetivo	Lograr que el sistema sea capaz de recuperarse ante fallos.
Origen	Interno al sistema
Artefacto	Servicios del sistema (cliente y servidor) / Canales de comunicación
Entorno	Operación normal

Estímulo	Respuesta: Flujo de eventos (Escenarios)
1.a Interrupción de comunicaciones de red en la PC	
	Tratar de conectarse cada cierto tiempo para terminar el proceso de comunicación / Continuar funcionando en modo normal
Medida de respuesta	
NA	
2.a Interrupción de comunicaciones de red en la PC servidor	
	Tratar de conectarse cada cierto tiempo para terminar el proceso de comunicación / Continuar funcionando en modo normal
Medida de respuesta	
NA	
3.a Inactividad del cliente	
	El servidor se encargará de notificar los clientes inactivos / Continuar funcionando en modo normal
Medida de respuesta	
NA	
4.a Ocurrencia de una excepción	
	Se notifica al usuario / Continuar funcionando en modo normal
Medida de respuesta	
NA	

Tabla 2. Requisito no funcional de Confiabilidad.

Atributo de Calidad	Confiabilidad
Sub-atributos/Sub-características	Cumplimiento de fiabilidad
Objetivo	Prevenir el acceso no autorizado, a los datos.
Origen	El Usuario
Artefacto	Canal de Comunicación, Sistema
Entorno	El usuario desea obtener información sobre los

	datos procesados por el sistema.
Estímulo	Respuesta: Flujo de eventos (Escenarios)
1.a Persona no autorizada intenta acceder a los datos del sistema.	
	Bloquear el acceso al sistema
Medida de respuesta	
NA	
2.a Persona con permisos restringidos intenta acceder a información a la que no tiene acceso.	
	Bloquear el acceso a estos datos.
Medida de respuesta	
NA	

Tabla 3. Requisito no funcional de Usabilidad.

Atributo de Calidad	Usabilidad
Sub-atributos/Sub-características	Comprensibilidad
Objetivo	Lograr que el sistema sea entendible fácilmente para los usuarios.
Origen	El usuario
Artefacto	El sistema
Entorno	El sistema está funcionando correctamente.
Estímulo	Respuesta: Flujo de eventos (Escenarios)
NA	NA
Medida de respuesta	
NA	

Tabla 4. Requisito no funcional de Usabilidad.

Atributo de Calidad	Usabilidad
Sub-atributos/Sub-características	Operabilidad
Objetivo	Lograr que el sistema sea capaz de realizar todas las operaciones solicitadas por el cliente.
Origen	El usuario

Artefacto	El sistema
Entorno	El sistema está funcionando correctamente.
Estímulo	Respuesta: Flujo de eventos (Escenarios)
NA	NA
Medida de respuesta	
NA	

Tabla 5. Requisito no funcional de Eficiencia.

Atributo de Calidad	Eficiencia
Sub-atributos/Sub-características	Comportamiento en el tiempo
Objetivo	Lograr que el sistema ejecute correctamente las peticiones solicitadas por el usuario en el menor tiempo posible.
Origen	El usuario
Artefacto	El sistema
Entorno	Funcionando correctamente
Estímulo	Respuesta: Flujo de eventos (Escenarios)
1.a El usuario realiza una petición al sistema	
	El sistema responde correctamente la petición.
Medida de respuesta	
NA	

Tabla 6. Requisito no funcional de Hardware.

Atributo de Calidad	Hardware
Sub-atributos/Sub-características	Utilización de recursos
Objetivo	CPU (4 x 2.33 GHz (Intel Xeon 5140 Core2 2.33 GHz)) RAM (8GB) HDD (30 Gb RAID 5) LAN (1 x NIC (1 Gbit))
Origen	Externo al sistema

Artefacto	Servidor de aplicación
Entorno	Operación normal
Estímulo	Respuesta: Flujo de eventos (Escenarios)
NA	NA

Tabla 7. Requisito no funcional de Hardware.

Atributo de Calidad	Hardware
Sub-atributos/Sub-características	Utilización de recursos
Objetivo	CPU (4 x 2.33 GHz (Intel Xeon 5140 Core2 2.33 GHz)) RAM (8GB) HDD (50 Gb RAID 5) LAN (2 x NIC (1 Gbit))
Origen	Externo al sistema
Artefacto	Servidor de almacenamiento de información
Entorno	Operación normal
Estímulo	Respuesta: Flujo de eventos (Escenarios)
NA	NA

Tabla 8. Requisito no funcional de Portabilidad.

Atributo de Calidad	Portabilidad
Sub-atributos/Sub-características	Adaptabilidad
Objetivo	Lograr que el sistema sea adaptable a diferentes entornos especificados.
Origen	El usuario
Artefacto	El sistema
Entorno	Funcionando correctamente
Estímulo	Respuesta: Flujo de eventos (Escenarios)
1.a Se prueba el sistema en diferentes entornos.	
	El sistema funciona correctamente.
Medida de respuesta	
NA	

Tabla 9. Requisito no funcional de Portabilidad.

Atributo de Calidad	Portabilidad
Sub-atributos/Sub-características	Instabilidad
Objetivo	Lograr que el sistema se pueda instalar en un entorno especificado.
Origen	El usuario
Artefacto	El sistema
Entorno	Funcionando correctamente
Estímulo	Respuesta: Flujo de eventos (Escenarios)
1.a Se instala el sistema en diferentes entornos.	
	El sistema funciona correctamente.
Medida de respuesta	
	NA

2.4 Diagrama de clases del diseño

Los diagramas de clases contienen normalmente los siguientes elementos, clases, interfaces, colaboraciones, relaciones de dependencia, generalización y asociación. Los diagramas pueden contener también notas, restricciones, paquetes o subsistemas, los cuales se usan para agrupar los elementos de un modelo en partes más grandes.[28]

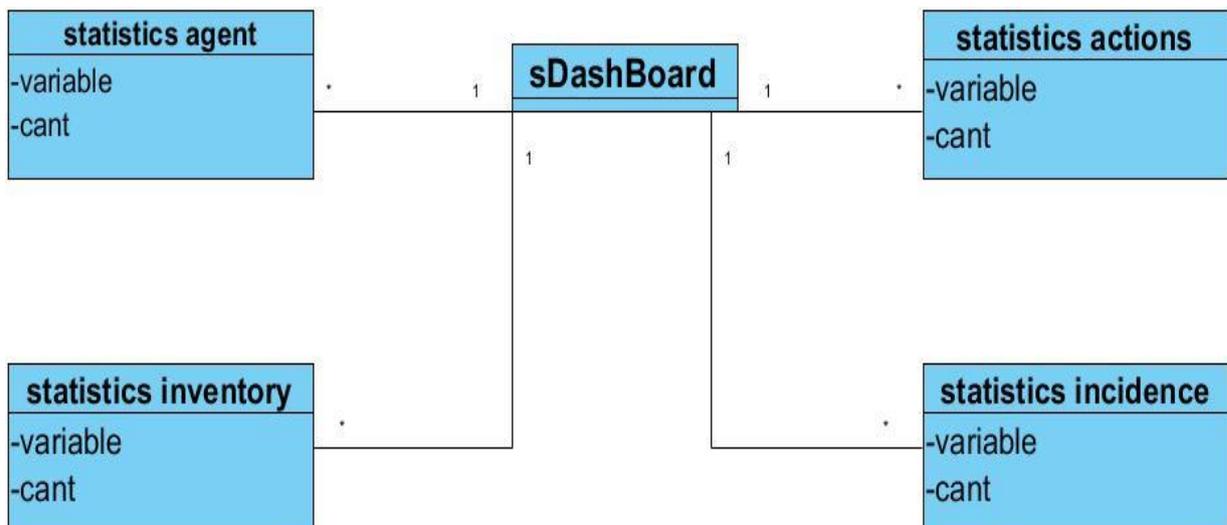


Figura 5. Diagrama de clases del diseño.

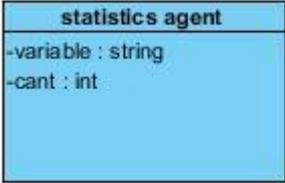
El diagrama de clases está compuesto por cinco clases teniendo a sDashboard como clase principal y a statistics agent, statistics actions, statistics inventory, statistics incidence que tienen cada uno dos

atributos que son variable y cantidad, son los elementos necesarios para graficar. Las clases statistics tienen una relación de mucho a uno con la clase sDashboard.

2.4.1 Descripción de clases

Clase statistics agent

Tabla 10. Descripción de clases del diseño.

Número del módulo	1
Número de la clase	1
Clase	statistics agent
Propósito	Contiene las estadísticas de agentes
Descripción	 <pre> classDiagram class statistics_agent { -variable : string -cant : int } </pre>
Observaciones	

Clase statistics action

Tabla 11. Descripción de clases del diseño.

Número del módulo	1
Número de la clase	2
Clase	statistics action
Propósito	Contiene las estadísticas de acción

Descripción	<pre> statistics action -variable : string -cant : int </pre>
Observaciones	

Clase statistics inventory

Tabla 12. Descripción de clases del diseño.

Número del módulo	1
Número de la clase	3
Clase	statistics inventory
Propósito	Contiene las estadísticas de inventario
Descripción	<pre> statistics inventory -variable : string -cant : int </pre>
Observaciones	

Clase statistics incidence

Tabla 13. Descripción de clases del diseño.

Número del módulo	1
Número de la clase	4
Clase	statistics incidence

Propósito	Contiene las estadísticas de las incidencias
Descripción	<pre> statistics incidence -variable : string -cant : int </pre>
Observaciones	

2.5 Diagrama de despliegue

Un diagrama de despliegue modela la arquitectura en tiempo de ejecución de un sistema. Esto muestra la configuración de los elementos de hardware (nodos) y muestra cómo los elementos y artefactos del software se trazan en esos nodos. Un nodo es un elemento de hardware o software.[29]

A continuación se muestra el modelo de despliegue propuesto para el componente en el cual los clientes se conectan por el protocolo HTTPS al servidor de la aplicación el cual utilizará una conexión TCP-IP para la comunicación con el servidor de base de datos.

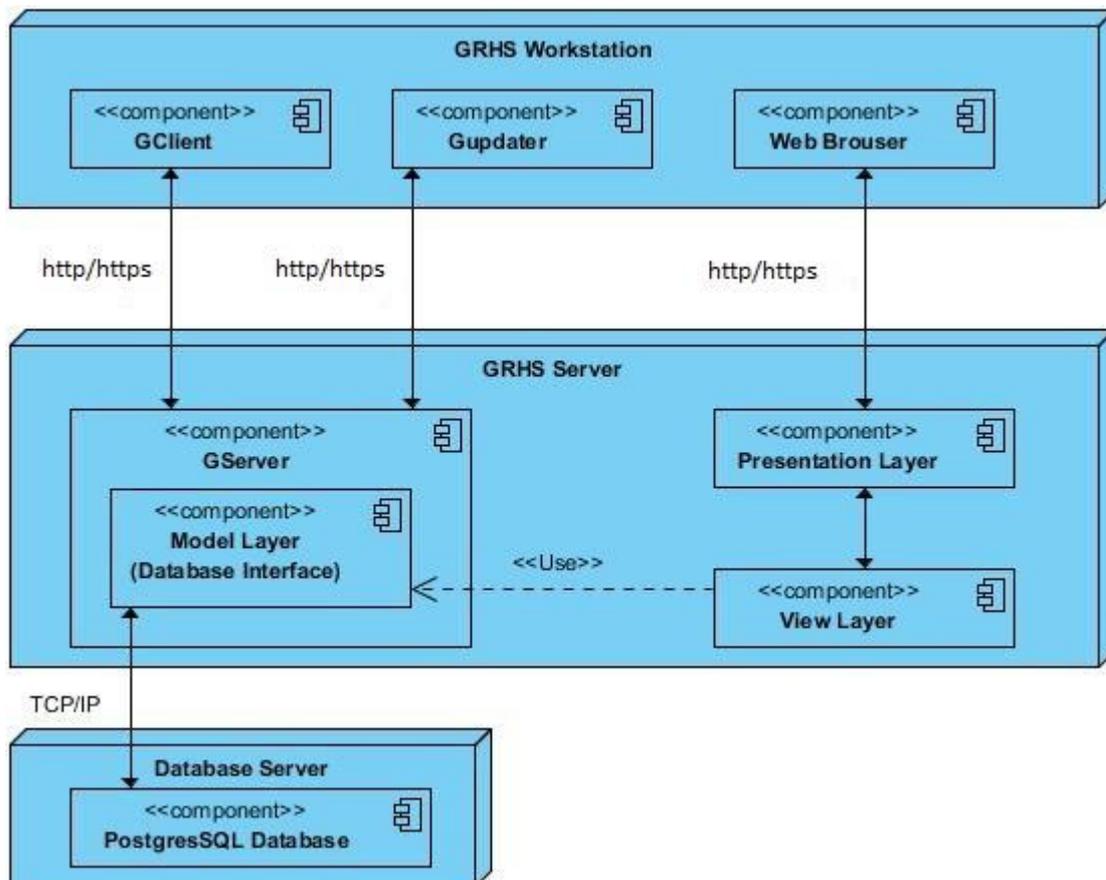


Figura 6. Diagrama de despliegue.

En la aplicación Gserver se encuentra el componente de visualización de estadísticas para los módulos de Xilema GRHS en forma de gráfica, en cual se fueron distribuyendo por cada módulo cómo se iba a graficar, de esta manera se puede visualizar las estadísticas en los módulos sin entrar a un cuadro de mando.

2.6 Conclusiones del capítulo

Los artefactos obtenidos durante esta etapa del desarrollo de la solución propuesta dan cumplimiento a los requisitos funcionales describiendo lo que el sistema debe hacer en cada momento y los requisitos no funcionales los que definieron las restricciones del mismo. Mediante la representación del modelo de dominio, el autor identificó las entidades relacionados con el sistema y las relaciones entre ellas. Los diagramas de clases y el diagrama de despliegue lograron que se obtuvieran de forma concreta y detallada las relaciones existentes entre las clases. Todos estos artefactos propuestos por la metodología AUP-UCI, permitió estructurar todo el proceso de desarrollo para las siguientes actividades de implementación y prueba.

Capítulo 3: Implementación y Prueba

Introducción

Luego de haber definido las herramientas informáticas a utilizar y las funcionalidades del componente, se debe proseguir a la implementación y a la realización de las pruebas al componente. En este capítulo se definen importantes parámetros para la construcción de la implementación de la solución propuesta. Se explican, a continuación, la arquitectura empleada y el conjunto de patrones de diseños a utilizar para garantizar la calidad del producto. De igual forma se analiza la fase de implementación a partir de los resultados del diseño, describiendo el estado actual del componente. Para la fase de prueba se describen los tipos de pruebas a realizar y los resultados obtenidos de las mismas.

3.1 Arquitectura de software

La arquitectura del software aporta una visión abstracta de alto nivel, posponiendo el detalle de cada uno de los módulos definidos a pasos posteriores. Es definida según la IEEE²⁰ Estándar 1471-2000 como: la organización fundamental de un sistema formada por sus componentes, las relaciones entre ellos, el contexto en el que se implantarán, y los principios que orientan su diseño y evolución.[30]

La arquitectura del software alude a la estructura global del software y a las formas en que esta proporciona la integridad conceptual de un sistema. En su forma más simple, la arquitectura es la estructura jerárquica de los componentes del programa (módulos), la manera en que los componentes interactúan y la estructura de datos que van a utilizar los componentes. [31]

3.2 Patrones de arquitectura del componente

La arquitectura de software es una descripción de los subsistemas y componentes de un sistema software, y de las relaciones entre ellos. Los subsistemas y componentes son especificados en diferentes vistas para mostrar las propiedades funcionales y no funcionales relevantes del sistema. La arquitectura de software es un artefacto; es el resultado de la actividad de diseño de software. [32]

Un patrón describe un problema que ocurre una y otra vez en el entorno y describe también el núcleo de la solución al problema, de forma que puede reutilizarse continuamente.

Los patrones de arquitectura expresan los esquemas de organización estructural fundamental para sistemas software. Además, proveen de un conjunto de subsistemas predefinidos, especifican sus responsabilidades e incluyen reglas y guías para la organización de las relaciones entre ellos. [32]

²⁰ Instituto de Ingeniería Eléctrica y Electrónica

3.2.1 Patrón arquitectónico cliente servidor

El sistema Xilema GRHS cuenta con una arquitectura cliente-servidor, como se muestra en la figura 8, el cual permite a los usuarios finales obtener acceso a la información en forma transparente aún en entornos multiplataforma. En el arquitectura cliente servidor, el cliente (Gclient) envía un mensaje solicitando un determinado servicio al servidor (Gserver), o lo que es lo mismo le hace una petición, y este envía uno o varios mensajes con la respuesta (provee el servicio).

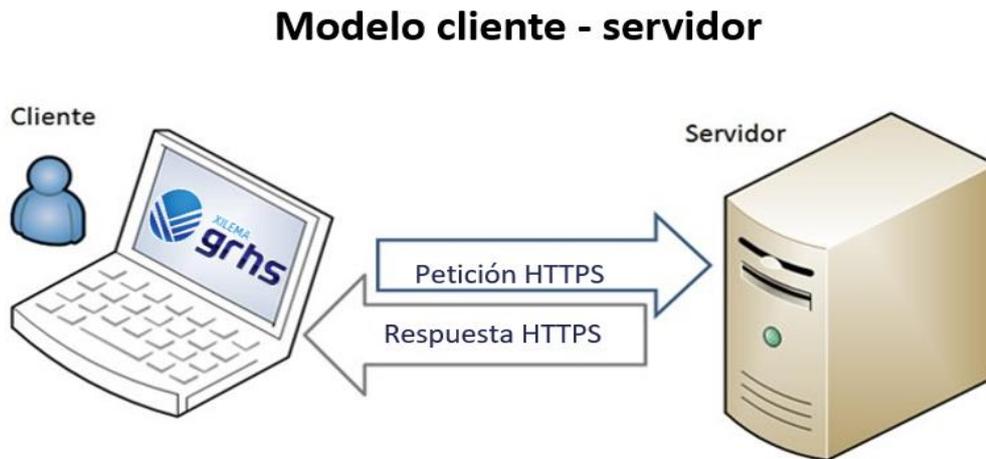


Figura 7. Modelo cliente-servidor.

Para el desarrollo de Xilema GRHS se contará con un servidor desarrollado con las tecnologías de Python 2.7 y Django 1.8, el cual brindará los datos necesarios para las estadísticas que se encuentran en cada módulo, para con el lenguaje de JavaScript y la herramienta sDashboard empleado graficar las estadísticas y mostrarla a los clientes a través de Gadmin.

3.2.2 Patrón arquitectónico Modelo Plantilla Vista

Django se basa en el Modelo Plantilla Vista (MPV), que es una modificación del Modelo Vista Controlador (MVC). El controlador pasa a ser la vista y la vista pasa a denominarse plantilla. En Django, una vista describe los datos que se ofrecen al usuario, pero no necesariamente su aspecto. Una vista habitualmente delega los datos a una plantilla que describe la forma de presentarlos. El modelo es el encargado de las consultas a la base de datos.[33]

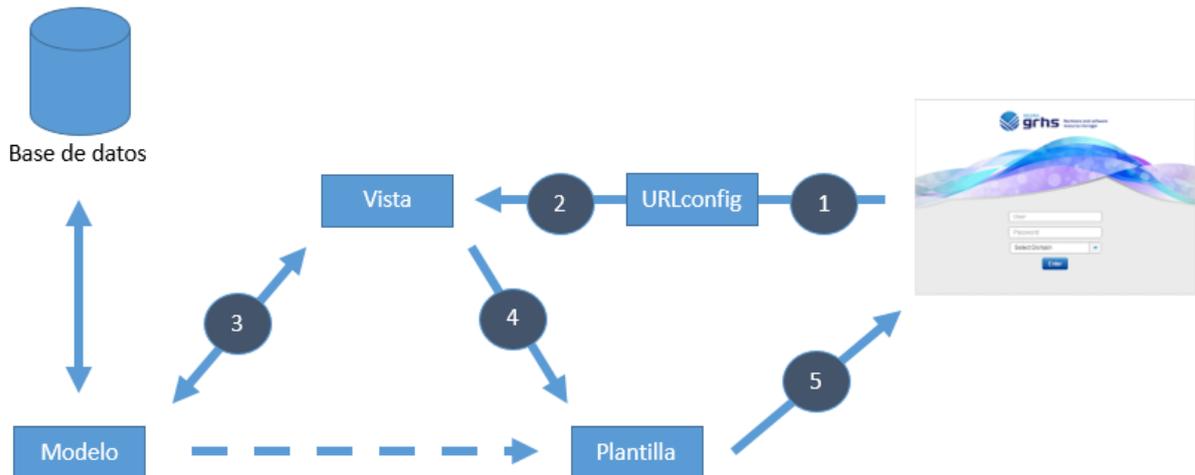


Figura 8. Modelo vista plantilla de Django.

Leyenda:

1. El navegador manda una solicitud.
2. El URLConf interpreta la solicitud y ubica la vista apropiada.
3. La vista interactúa con el modelo para obtener los datos.
4. La vista selecciona la plantilla apropiada.
5. La plantilla renderiza la respuesta a la solicitud del navegador.

Se presentan los diagramas de clases que generan las capas de la arquitectura MTV. Para un mejor entendimiento del marco de trabajo Django.

Capa Modelo

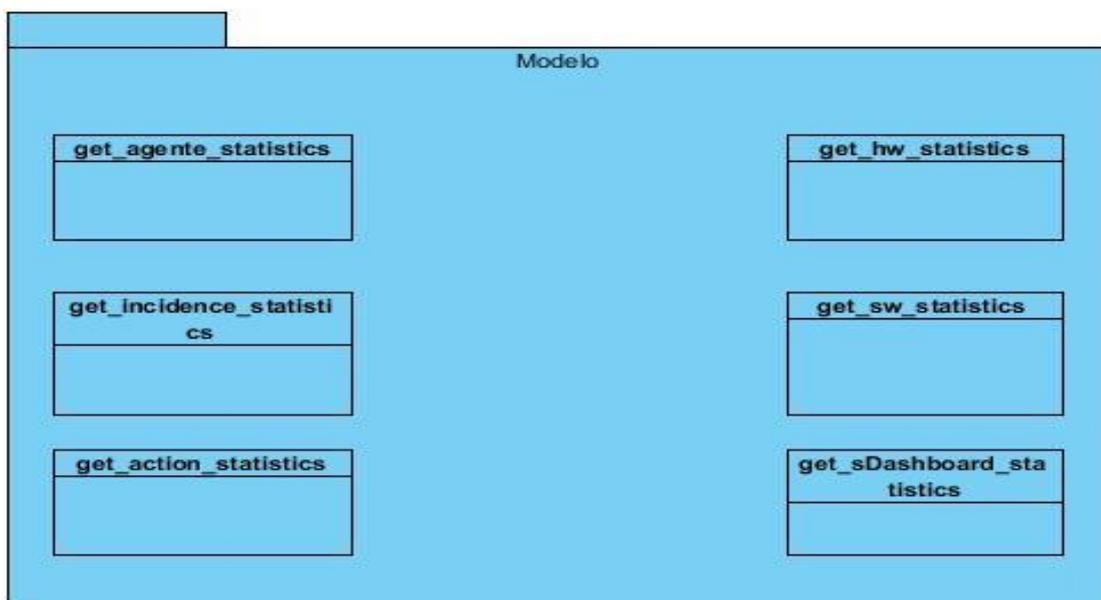


Figura 7. Capa modelo

Estas clases se encuentran dentro del modelo que brinda la arquitectura MTV. Encargadas de las consultas a la base de datos.

Capa Vista

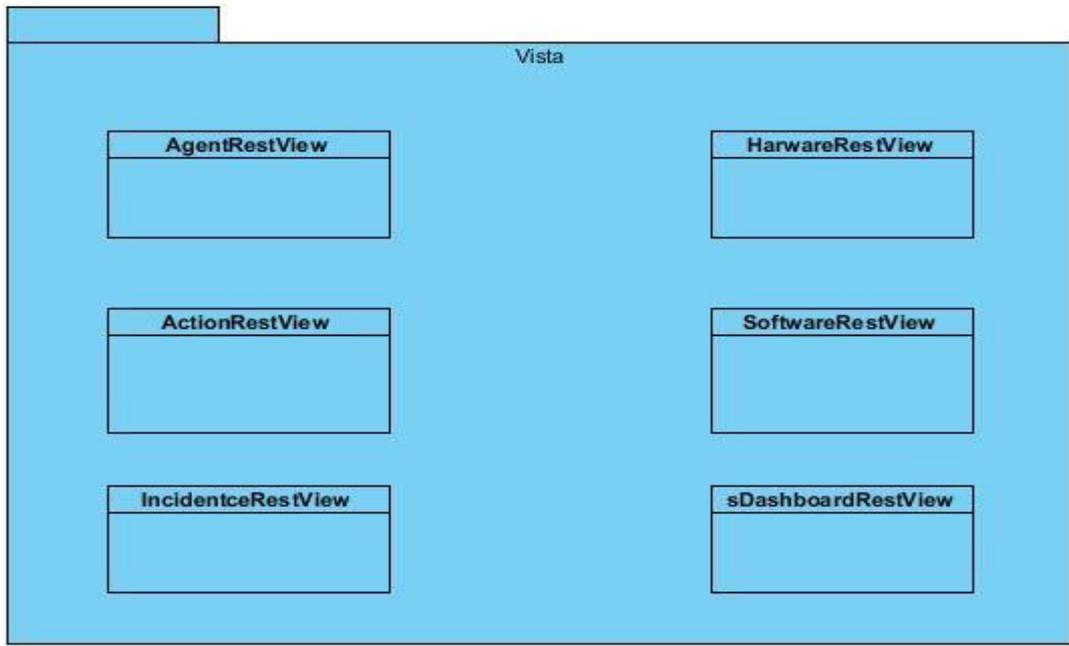


Figura 7.Capa vista

Estas clases se encuentran dentro de la vista que brinda la arquitectura MTV. Encargadas de generar el contenido de la plantilla a través de la interacción con la capa modelo.

Capa Plantilla

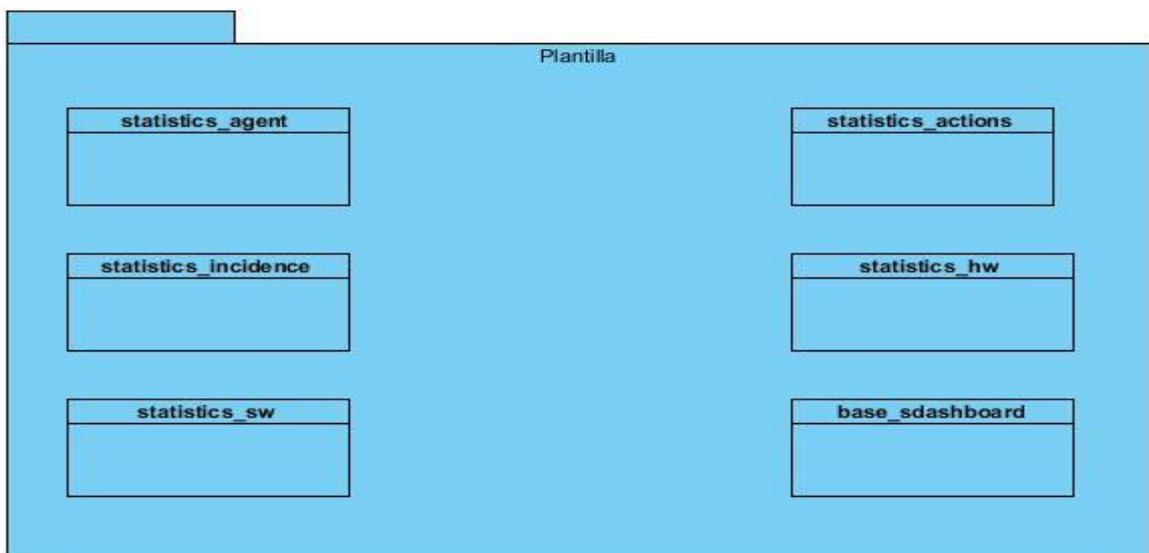


Figura 7.Capa plantilla

Los archivos action, incidence, agent, hardware (hw) y software (sf) contienen las plantillas html asociadas a su estructura dentro del módulo heredando todas estas plantillas de base_dashboard.

3.3 Patrones de diseño

Los patrones de diseño son la base para la búsqueda de soluciones a problemas comunes en el desarrollo de software y otros ámbitos referentes al diseño de interacción o interfaces. Representan una descripción de las clases y objetos comunicándose entre sí adaptada para resolver un problema de diseño general en un contexto particular. El uso de patrones posibilita estandarizar el modo en que se realiza el diseño y proporciona reusabilidad, extensibilidad y mantenimiento del código. [31]

3.3.1 Patrones General Responsibility Assignment Software Patterns (GRASP)

Los patrones GRASP, describen la asignación de responsabilidades a objetos. Para el desarrollo del componente fueron utilizados los siguientes patrones de esta clasificación que en su mayoría son implementados de forma nativa por el marco de trabajo Django: [31]

- **Experto:** Es un patrón encargado de asignar responsabilidades; es un principio básico que utiliza un diseño orientado a objetos [31]. Las Clases statistics agent, statistics actions, statistics inventory, statistics incidence contienen la información para cumplir con sus responsabilidades.
- **Alta cohesión:** propone asignar la responsabilidad de manera que la complejidad se mantenga dentro de límites manejables asumiendo solamente las responsabilidades que deben manejar, evadiendo un trabajo excesivo. Este patrón se pone de manifiesto durante todo el desarrollo del componente, ya que cada clase se encarga solamente de hacer lo que le corresponde, evitando la dependencia entre las clases. [31]
- **Bajo Acoplamiento:** debe haber pocas dependencias entre las clases. Si todas las clases dependen de todas no se puede extraer software de forma independiente y reutilizable. Este patrón está presente ya que cada clase tiene dependencia de hacer lo que le corresponde. [31]
- **Creador:** el patrón Creador guía la asignación de responsabilidades relacionadas con la creación de objetos (una tarea muy común). La intención básica del patrón es encontrar un creador que necesite conectarse al objeto creado en alguna situación. La ventaja de utilizar este patrón es eliminar la dependencia entre las clases logrando una mayor mantenibilidad y reutilización [31]. Dentro del componente se evidencia por sus acciones en la clase controladora "view.py", las cuales crean objetos para sus entidades.

3.3.2 Patrones de Diseño GOF

Dentro de los patrones de diseño se encuentran los patrones Gang of Four o Grupo de los Cuatro. Estos patrones definen el comportamiento entre las clases y los objetos. Los patrones de diseño GoF "se clasifican en 3 grandes categorías basadas en su propósito Creacionales, Estructurales y de Comportamiento" [31]. En el desarrollo del módulo se utilizaron los siguientes patrones GoF.

- **Decorador:** es un patrón de tipo estructura, ya que permite que clases y objetos sean utilizados para componer estructuras de mayor tamaño. Este patrón añade dinámicamente nuevas responsabilidades a un objeto. Django contiene un decorador que permite agregar funcionalidades dinámicamente a las aplicaciones desarrolladas bajo sus principios. Cada una de las vistas generadas hereda su diseño de la plantilla "base.html", siendo esta la plantilla contenedora de la estructura y el diseño básico de las plantillas.

3.4 Pruebas del software

La prueba de software es un elemento crítico para la garantía del correcto funcionamiento del software. Entre sus objetivos están: detectar defectos en el software, verificar la integración adecuada de los componentes y verificar que todos los requisitos se han implementado correctamente. Además de identificar y asegurar que los defectos encontrados se han corregido antes de entregar el software al cliente y diseñar casos de prueba que sistemáticamente saquen a la luz diferentes clases de errores, haciéndolo con la menor cantidad de tiempo y esfuerzo. [31]

3.4.1 Pruebas unitarias

Las pruebas unitarias están dirigidas a las funciones internas del sistema. La prueba es una verificación técnica del software que los desarrolladores pueden usar para examinar si el código trabaja como se esperaba. Se realizan probando la lógica de la aplicación y comprobando el estado del software en varios puntos, para verificar que los resultados de dicho estado coincidan con los esperados.[31]

En la ejecución de las pruebas unitarias se utilizó la herramienta llamada PyUnit, la cual está contenida en el lenguaje de programación Python. Al aplicarle la prueba al componente, en la primera iteración, arrojó como resultado 4 no conformidades y el tiempo de la ejecución fue de 1.557 segundos, como se observa en la imagen.



```
-----  
Ran 11 tests in 1.557s  
FAILED (errors=4)  
Destroying test database for alias 'default'...
```

Figura 9. Resultado de la primera iteración de las pruebas unitarias.

Las 4 no conformidades, eran producto de cuando se realizaba el exportar documento que el sistema no agregaba la descripción al PDF; luego de la primera iteración, se corrigieron estos errores y se realizó otra iteración de pruebas obteniendo como resultado ninguna no conformidad y el tiempo de ejecución aumentó a 1.363 segundos.

```

Ran 11 tests in 1.363s
OK
Destroying test database for alias 'default'...

```

Figura 10. Resultado de la segunda iteración de las pruebas unitarias.

3.4.2 Pruebas de aceptación

Las pruebas de caja negra se centran en los requisitos funcionales de la aplicación, sin internarse en el funcionamiento interno de la misma. Se desarrollan sobre la interfaz visual del software y se encarga de verificar que las funciones que debe desempeñar el sistema son operativas. Mediante estas pruebas se pueden encontrar errores de interfaz, funciones incorrectas, errores de salida de información y problemas con el acceso a datos.[31]

Para garantizar el buen funcionamiento del componente de visualización de estadísticas para los módulos de Xilema GRHS en forma de gráficas, se realizaron las pruebas funcionales correspondientes. A continuación se muestra el diseño de caso de prueba para el requisito seleccionar el tipo de gráfica el resto de las descripciones referentes a los casos de prueba se encuentra en Anexo.

Descripción general

Seleccionar el tipo de gráfica a visualizar.

Condiciones de ejecución

Estar autenticado.

SC Seleccionar

Tabla 14. Diseño del caso de prueba seleccionar tipo de gráfico.

Escenario	Descripción	Respuesta del sistema	Flujo central
EC 1.1 Seleccionar gráfica de pastel	El usuario autenticado selecciona la gráfica de pastel.	Se muestra la gráfica de pastel.	Seleccionar el módulo del cuál se desee visualizar las estadísticas y después la gráfica de pastel. 
EC 1.2 Seleccionar gráfica de pastel y no hay datos guardados en la base de datos.	El usuario autenticado selecciona la gráfica de pastel.	El sistema muestra la gráfica de pastel pero sin información por encontrarse la BD vacía.	Seleccionar el módulo del cuál se desee visualizar las estadísticas y después la gráfica de pastel. 

EC 1.3 Seleccionar gráfica de barra.	El usuario autenticado selecciona la gráfica de barra.	Se muestra la gráfica de barra.	Seleccionar el módulo del cuál se desee visualizar las estadísticas y después la gráfica de barra. 🗑️
EC 1.4 Seleccionar gráfica de barra y no hay datos guardados en la base de datos.	El usuario autenticado selecciona la gráfica de barra.	El sistema muestra la gráfica de barra pero sin información por encontrarse la BD vacía.	Seleccionar el módulo del cuál se desee visualizar las estadísticas y después la gráfica de barra. 🗑️

3.5 Resumen de las pruebas de aceptación

Se realizaron un total de 7 diseños de casos de prueba de aceptación divididos en 2 iteraciones. Para una primera iteración se detectaron 2 no conformidades eran 2 iconos que no cumplían ninguna función en el componente. En la segunda iteración realizada, no se detectaron no conformidades y fueron corregidas las no conformidades detectadas con anterioridad.

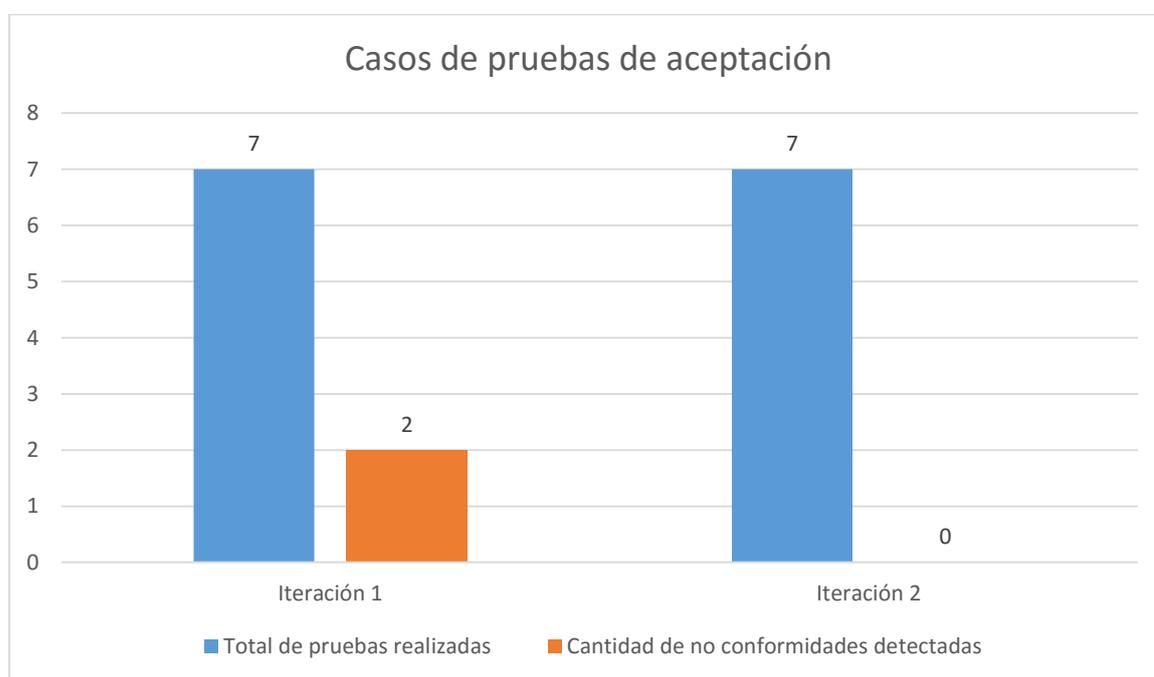


Figura 11. Resultado de las pruebas de aceptación en cada iteración.

3.6 Conclusiones

En el desarrollo del capítulo se ha llevado a cabo el trabajo en las fases de implementación y pruebas, partiendo de lo realizado se puede concluir que: partiendo del estudio del framework Django, se definieron los patrones de diseño a utilizar para el desarrollo de la aplicación. La identificación de las

librerías a utilizarse y la integración entre ellas, permitió la implementación de los componentes. En la fase de pruebas además de las métricas de evaluación para los diseños de casos de pruebas, se realizaron las pruebas unitarias y de aceptación, arrojando resultados satisfactorios. Garantizando el cumplimiento de los requisitos planteados.

Conclusiones Generales

Con la realización de la presente investigación se adquirieron y pusieron en práctica los conocimientos necesarios para el desarrollo del componente de visualización de estadísticas para los módulos de Xilema GRHS en forma de gráfico, arribando a las siguientes conclusiones:

1. La realización de un estudio a las soluciones similares existentes permitió seleccionar a la herramienta sDashboard como la adecuada para graficar las estadísticas que producen los inventarios de hardware y software del sistema Xilema GRHS.
2. La metodología encargada de guiar el desarrollo de la solución fue AUP-UCI.
3. Como lenguaje de programación fue usado Python para la implementación del producto final.
4. La correcta definición de los requisitos del componente de visualización permitió el desarrollo de una solución que cumple con los objetivos planteados al inicio de la investigación.
5. Las pruebas unitarias y de aceptación realizadas, permitieron comprobar el buen funcionamiento del componente de visualización de estadísticas para los módulos de Xilema GRHS en forma de gráfica, al aportar las no conformidades detectadas en cada iteración, logrando un producto final con la calidad requerida.

Por todo lo anteriormente expuesto, se concluye que los objetivos presentados al inicio de la investigación han sido cumplidos en su totalidad.

Referencias

- [1] M. E. Rendón-Macías, M. Á. Villasis-Keever, y M. G. Miranda-Navales, «Estadística descriptiva», *Rev. Alerg. México*, vol. 63, n.º 4, pp. 397-407, nov. 2016.
- [2] Adrian E. Mena Rodríguez *et al.*, «GESTOR DE RECURSOS D E HARDWARE Y SOFTWARE». 14-mar-2018.
- [3] «Estadística descriptiva: Conceptos Generales», *Biblioteca de Investigaciones*, 02-feb-2012. .
- [4] «Tipos de gráficas estadísticas | Matemáticas modernas», *Matematicas Modernas*, 28-ene-2014. .
- [5] «IBM Knowledge Center - Tipos de gráficos». [En línea]. Disponible en: https://www.ibm.com/support/knowledgecenter/es/SSEP7J_10.2.0/com.ibm.swg.ba.cognos.ug_buxc.10.2.0.doc/c_ug_buxc_charts_cnfgs.html. [Accedido: 23-feb-2018].
- [6] «pChart 2.0 - a PHP charting library». [En línea]. Disponible en: <http://www.pchart.net/>. [Accedido: 22-feb-2018].
- [7] «Highcharts, Highstock and Highmaps documentation | Highcharts». [En línea]. Disponible en: <https://www.highcharts.com/docs>. [Accedido: 22-feb-2018].
- [8] «Charts», *Google Developers*. [En línea]. Disponible en: <https://developers.google.com/chart/>. [Accedido: 23-feb-2018].
- [9] «Chart.js | Open source HTML5 Charts for your website». [En línea]. Disponible en: <http://www.chartjs.org/>. [Accedido: 23-feb-2018].
- [10] M. Bostock, «D3.js - Data-Driven Documents». [En línea]. Disponible en: <https://d3js.org/>. [Accedido: 23-feb-2018].
- [11] «FusionCharts Free - Open Source Charting Component for Web & Mobile Apps». [En línea]. Disponible en: <https://www.fusioncharts.com/goodies/fusioncharts-free/>. [Accedido: 23-feb-2018].
- [12] «jQuery Sparklines». [En línea]. Disponible en: <https://omnipotent.net/jquery.sparkline/#s-about>. [Accedido: 01-mar-2018].
- [13] «ModelN/sDashboard». [En línea]. Disponible en: <http://modeln.github.io/sDashboard/>. [Accedido: 23-feb-2018].
- [14] Duarny Meneses Hernández y Ciro Alejandro Carbonell Figueredo, «Módulo de presentación de resultados de inventarios de hardware y software en el sistema GRHS.», Universidad de las Ciencias Informáticas, Facultad 2, 2015.
- [15] «Metodología de desarrollo para la Actividad productiva de la UCI.» .
- [16] «ReportLab - Content to PDF Solutions». [En línea]. Disponible en: <https://www.reportlab.com/>. [Accedido: 31-may-2018].
- [17] «jqPlot Charts and Graphs for jQuery». [En línea]. Disponible en: <http://www.jqplot.com/>. [Accedido: 23-feb-2018].
- [18] «Welcome to Python.org», *Python.org*. [En línea]. Disponible en: <https://www.python.org/>. [Accedido: 01-mar-2018].

- [19] C. J. González y R. V. Rodríguez, «Editor web visual para HTML, CSS y JavaScript de apoyo a la docencia», *Virtualidad Educ. Cienc.*, vol. 8, n.º 14, pp. 136-152, 2017.
- [20] «Introducción a JavaScript». [En línea]. Disponible en: <http://librosweb.es/libro/javascript/>. [Accedido: 01-mar-2018].
- [21] «DevDocs — Django 1.8 documentation». [En línea]. Disponible en: <http://devdocs.io/django~1.8/>. [Accedido: 01-mar-2018].
- [22] «CyTA». [En línea]. Disponible en: <http://www.cyta.com.ar/ta1401/v14n1a2.htm>. [Accedido: 10-may-2018].
- [23] Universidad interamericana para el desarrollo, «HERRAMIENTAS DE SOFTWARE». .
- [24] DesarrolloWeb.com, «Sistemas gestores de bases de datos», *DesarrolloWeb.com*. [En línea]. Disponible en: <http://www.desarrolloweb.com/articulos/sistemas-gestores-bases-datos.html>. [Accedido: 02-mar-2018].
- [25] «PostgreSQL: Documentation: 10: 9.8. Data Type Formatting Functions». [En línea]. Disponible en: <https://www.postgresql.org/docs/10/static/functions-formatting.html>. [Accedido: 02-mar-2018].
- [26] «PgAdmin III - Guía Ubuntu». [En línea]. Disponible en: https://www.guia-ubuntu.com/index.php?title=PgAdmin_III. [Accedido: 02-mar-2018].
- [27] *Sommerville*, 2005.^a ed. .
- [28] «Diagrama de Clases de Diseño | LENGUAJE DE MODELADO UNIFICADO - UML». [En línea]. Disponible en: http://stadium.unad.edu.co/ovas/10596_9836/diagrama_de_clases_de_diseo.html. [Accedido: 27-feb-2018].
- [29] «Sparx Systems - Tutorial UML 2 - Diagrama de Despliegue». [En línea]. Disponible en: http://www.sparxsystems.com.ar/resources/tutorial/uml2_deploymentdiagram.html. [Accedido: 23-feb-2018].
- [30] DesarrolloWeb.com, «Usabilidad y arquitectura del software», *DesarrolloWeb.com*. [En línea]. Disponible en: <http://www.desarrolloweb.com/articulos/1622.php>. [Accedido: 28-feb-2018].
- [31] Roger S. Pressman, *Ingeniería de software : UN enfoque práctico séptima edición*, vol. 7. María Teresa Zapata Terrazas.
- [32] «Pattern-Oriented Software Architecture, Volume 1 - A System Of Patterns.pdf - Google Drive». [En línea]. Disponible en: <https://docs.google.com/file/d/0B1MogsyNASj9a1BVbi12VzhPX2M/edit>. [Accedido: 20-feb-2017].

Bibliografía

- [1] «Chart.js | Open source HTML5 Charts for your website». [En línea]. Disponible en: <http://www.chartjs.org/>. [Accedido: 23-feb-2018].
- [2] «Charts», *Google Developers*. [En línea]. Disponible en: <https://developers.google.com/chart/>. [Accedido: 23-feb-2018].
- [3] «CyTA». [En línea]. Disponible en: <http://www.cyta.com.ar/ta1401/v14n1a2.htm>. [Accedido: 10-may-2018].
- [4] M. Bostock, «D3.js - Data-Driven Documents». [En línea]. Disponible en: <https://d3js.org/>. [Accedido: 23-feb-2018].
- [5] «DevDocs — Django 1.8 documentation». [En línea]. Disponible en: <http://devdocs.io/django~1.8/>. [Accedido: 01-mar-2018].
- [6] «Diagrama de Clases de Diseño | LENGUAJE DE MODELADO UNIFICADO - UML». [En línea]. Disponible en: http://stadium.unad.edu.co/ovas/10596_9836/diagrama_de_clases_de_diseo.html. [Accedido: 27-feb-2018].
- [7] C. J. González y R. V. Rodríguez, «Editor web visual para HTML, CSS y JavaScript de apoyo a la docencia», *Virtualidad, Educación y Ciencia*, vol. 8, n.º 14, pp. 136-152, 2017.
- [8] «Ejemplos de tipos de gráficos». [En línea]. Disponible en: <https://support.office.com/es-es/article/ejemplos-de-tipos-de-gr%C3%A1ficos-96ac1fbf-ac20-40e1-a3c0-54ec215031fe>. [Accedido: 17-may-2018].
- [9] «Entorno de Desarrollo Integrado (IDE). | Fergarcia». [En línea]. Disponible en: <https://fergarcia.wordpress.com/2013/01/25/entorno-de-desarrollo-integrado-ide/>. [Accedido: 01-mar-2018].
- [10] U. Formulas, «Estadística descriptiva», *Universo Formulas*, 13-abr-2014. .
- [11] M. E. Rendón-Macías, M. Á. Villasis-Keever, y M. G. Miranda-Navales, «Estadística descriptiva», *Revista Alergia México*, vol. 63, n.º 4, pp. 397-407, nov. 2016.
- [12] «Estadística descriptiva». [En línea]. Disponible en: <http://www.universoformulas.com/estadistica/descriptiva/>. [Accedido: 22-feb-2018].
- [13] «Estadística descriptiva». [En línea]. Disponible en: <http://www.medigraphic.com/cgi-bin/new/resumen.cgi?IDARTICULO=68683>. [Accedido: 10-may-2018].
- [14] «Estadística descriptiva: Conceptos Generales», *Biblioteca de Investigaciones*, 02-feb-2012. .
- [15] «Estadística descriptiva: Conceptos Generales | Biblioteca de Investigaciones». [En línea]. Disponible en: <https://bibliotecadeinvestigaciones.wordpress.com/matematicas/estadistica-descriptiva-conceptos-generales/>. [Accedido: 22-feb-2018].
- [16] «Estadística para todos». [En línea]. Disponible en: http://www.estadisticaparatodos.es/historia/histo_esta.html. [Accedido: 22-feb-2018].
- [17] «FusionCharts Free - Open Source Charting Component for Web & Mobile Apps». [En línea].

- Disponible en: <https://www.fusioncharts.com/goodies/fusioncharts-free/>. [Accedido: 23-feb-2018].
- [18] Adrian E. Mena Rodríguez *et al.*, «GESTOR DE RECURSOS D E HARDWARE Y SOFTWARE». 14-mar-2018.
- [19] Universidad interamericana para el desarrollo, «HERRAMIENTAS DE SOFTWARE». .
- [20] «Highcharts, Highstock and Highmaps documentation | Highcharts». [En línea]. Disponible en: <https://www.highcharts.com/docs>. [Accedido: 22-feb-2018].
- [21] «IBM Knowledge Center». [En línea]. Disponible en: https://www.ibm.com/support/knowledgecenter/es/SSEP7J_10.2.1/com.ibm.swg.ba.cognos.ug_bu_xc.10.2.1.doc/c_ug_bu_xc_charts_cnfgs.html. [Accedido: 17-may-2018].
- [22] «IBM Knowledge Center - Tipos de gráficos». [En línea]. Disponible en: https://www.ibm.com/support/knowledgecenter/es/SSEP7J_10.2.0/com.ibm.swg.ba.cognos.ug_bu_xc.10.2.0.doc/c_ug_bu_xc_charts_cnfgs.html. [Accedido: 23-feb-2018].
- [23] Roger S. Pressman, *Ingeniería de software : UN enfoque práctico séptima edición*, vol. 7. María Teresa Zapata Terrazas.
- [24] «Interactive JavaScript charts for your webpage | Highcharts». [En línea]. Disponible en: <https://www.highcharts.com/>. [Accedido: 23-feb-2018].
- [25] «Introducción a JavaScript». [En línea]. Disponible en: <http://librosweb.es/libro/javascript/>. [Accedido: 01-mar-2018].
- [26] «JavaScript a fondo». [En línea]. Disponible en: <https://desarrolloweb.com/javascript/#librerias>. [Accedido: 01-mar-2018].
- [27] «JavaScript a fondo». [En línea]. Disponible en: <https://desarrolloweb.com/javascript/#librerias>. [Accedido: 01-mar-2018].
- [28] «jqPlot Charts and Graphs for jQuery». [En línea]. Disponible en: <http://www.jqplot.com/>. [Accedido: 23-feb-2018].
- [29] «jQuery Sparklines». [En línea]. Disponible en: <https://omnipotent.net/jquery.sparkline/#s-about>. [Accedido: 01-mar-2018].
- [30] «Manual de HTML5 en español « HTML 5, información útil en español». .
- [31] «Metodología de desarrollo para la Actividad productiva de la UCI.» .
- [32] «ModelN/sDashboard». [En línea]. Disponible en: <http://modeln.github.io/sDashboard/>. [Accedido: 23-feb-2018].
- [33] Duarny Meneses Hernández y Ciro Alejandro Carbonell Figueredo, «Módulo de presentación de resultados de inventarios de hardware y software en el sistema GRHS.», Universidad de las Ciencias Informáticas, Facultad 2, 2015.
- [34] «pChart 2.0 - a PHP charting library». [En línea]. Disponible en: <http://www.pchart.net/>. [Accedido: 22-feb-2018].
- [35] «PgAdmin III - Guía Ubuntu». [En línea]. Disponible en: https://www.guia-ubuntu.com/index.php?title=PgAdmin_III. [Accedido: 02-mar-2018].

- [36] «PgAdmin III - Guía Ubuntu». [En línea]. Disponible en: https://www.guia-ubuntu.com/index.php?title=PgAdmin_III. [Accedido: 02-mar-2018].
- [37] «PostgreSQL: Documentation: 10: 9.8. Data Type Formatting Functions». [En línea]. Disponible en: <https://www.postgresql.org/docs/10/static/functions-formatting.html>. [Accedido: 02-mar-2018].
- [38] «PostgreSQL: The world's most advanced open source database». [En línea]. Disponible en: <https://www.postgresql.org/>. [Accedido: 10-may-2018].
- [39] «psutil 5.4.3: Python Package Index». [En línea]. Disponible en: <https://pypi.python.org/pypi/psutil>. [Accedido: 01-mar-2018].
- [40] «py-cpuinfo 0.1.4: Python Package Index». [En línea]. Disponible en: <https://pypi.python.org/pypi/py-cpuinfo/0.1.4>. [Accedido: 01-mar-2018].
- [41] «PyPI - the Python Package Index: Python Package Index». [En línea]. Disponible en: <https://pypi.python.org/pypi>. [Accedido: 01-mar-2018].
- [42] «ReportLab PDF Library». .
- [43] DesarrolloWeb.com, «Sistemas gestores de bases de datos», *DesarrolloWeb.com*. [En línea]. Disponible en: <http://www.desarrolloweb.com/articulos/sistemas-gestores-bases-datos.html>. [Accedido: 02-mar-2018].
- [44] *Sommerville*, 2005.^a ed. .
- [45] «Sparx Systems - Tutorial UML 2 - Diagrama de Despliegue». [En línea]. Disponible en: http://www.sparxsystems.com.ar/resources/tutorial/uml2_deploymentdiagram.html. [Accedido: 23-feb-2018].
- [46] «Tipos de gráficas estadísticas | Matemáticas modernas», *Matemáticas Modernas*, 28-ene-2014. .
- [47] «UML y Patrones - 2da Edición - Craig Larman.pdf», *Google Docs*. [En línea]. Disponible en: https://docs.google.com/file/d/0B3zChaLaPHJbS2hTQU5VaExBOGs/preview?usp=embed_facebook. [Accedido: 02-mar-2018].
- [48] DesarrolloWeb.com, «Usabilidad y arquitectura del software», *DesarrolloWeb.com*. [En línea]. Disponible en: <http://www.desarrolloweb.com/articulos/1622.php>. [Accedido: 28-feb-2018].
- [49] «Welcome to Python.org», *Python.org*. [En línea]. Disponible en: <https://www.python.org/>.

Anexo

Especificación de requisito: guardar imagen de la gráfica realizada.

Descripción textual del requisito

Precondiciones	El usuario que desee guardar imagen de la gráfica realizada debe estar autenticado en el sistema Xilema GRHS 2.0.	
Flujo de eventos		
Flujo básico Guardar imagen de la gráfica realizada		
1.	Selecciona el módulo del cuál desea visualizar las estadísticas en forma de gráficos. Si no se muestran los datos graficados ver flujo alternativo 1.a.	
2.	Seleccionar el ícono de imagen.	
3.	Selecciona el botón Ok en el formulario.	
4.	La gráfica se descarga de forma automática.	
Pos-condiciones		
1.	Se guarda la imagen de la gráfica realizada	
Flujos alternativos		
Flujo alternativo 1.a No hay conexión con la BD de GRHS		
1.	El sistema muestra un mensaje. "No existen datos para mostrar".	
Pos-condiciones		
1.	Se guarda la imagen de la gráfica realizada.	
Validaciones		
1.	NA	
Conceptos	Gráfico	
Requisitos especiales	Usabilidad, Eficiencia	
Asuntos pendientes	NA	

Prototipo elemental de interfaz gráfica de usuario



Figura 12. Prototipo de interfaz guardar imagen

Especificación de requisito: actualizar los datos de la gráfica.

Descripción textual del requisito

Precondiciones	El usuario que desee actualizar los datos de la gráfica debe estar autenticado en el sistema Xilema GRHS 2.0.
Flujo de eventos	
Flujo básico Actualizar los datos de la gráfica.	
1.	Selecciona el módulo del cuál desea actualizar las estadísticas en gráficos. Si no se muestran los datos graficados ver flujo alternativo 1.a.
2.	Seleccionar el ícono de actualizar.
3.	Se actualizan los datos de la gráfica.
Pos-condiciones	
1.	Se actualiza los datos de la gráfica de forma automática.
Flujos alternativos	
Flujo alternativo 1.a No hay conexión con la BD de GRHS	
1.	El sistema muestra un mensaje. "No existen datos para mostrar".
Pos-condiciones	
1.	Se actualiza los datos de la gráfica.
Validaciones	

1.	NA
Conceptos	Gráfico
Requisitos especiales	Usabilidad, Eficiencia
Asuntos pendientes	NA

Prototipo elemental de interfaz gráfica de usuario



Figura 13. Prototipo de interfaz actualizar los datos de la gráfica

Especificación de requisito: graficar de acuerdo a la gráfica seleccionada.

Descripción textual del requisito

Precondiciones	El usuario que desee graficar de acuerdo a la gráfica seleccionada debe estar autenticado en el sistema Xilema GRHS v2.0.
Flujo de eventos	
Flujo básico Graficar de acuerdo a la gráfica seleccionada	
1.	Selecciona el módulo del cuál desea visualizar las estadísticas en gráficos.
2.	Se elige el modo en que se deseen ver los datos. Los datos pueden ser gráfica de pastel o de barra. Si no se muestran los datos graficados ver flujo alternativo 1.a.
Pos-condiciones	
4.	Se ven las estadísticas en la gráfica seleccionada.
Flujos alternativos	
Flujo alternativo 1.a No hay conexión con la BD de GRHS	
1.	El sistema muestra un mensaje. "No existen datos para mostrar".

Pos-condiciones	
1.	Se ven los datos de la gráfica.
Validaciones	
1.	NA
Conceptos	Gráfico
Requisitos especiales	Usabilidad, Eficiencia
Asuntos pendientes	NA

Prototipo elemental de interfaz gráfica de usuario



Figura 14. Prototipo de interfaz graficar de acuerdo a la gráfica seleccionada

Especificación de requisito: exportar a documento.

Descripción textual del requisito

Precondiciones	El usuario que desee exportar la gráfica a documento debe estar autenticado en el sistema Xilema GRHS v2.0
Flujo de eventos	
Flujo básico Exportar a documento	
1.	Selecciona el módulo del cuál desea exportar a documento las estadísticas en forma de gráfico.
2.	Se elige el modo en que se deseen ver los datos. Los datos pueden ser gráfica de pastel o de barra. Si no se muestran los datos graficados ver flujo alternativo 1.a.
3.	Se selecciona el ícono de reporte.
4.	Se selecciona el botón de crear reporte en el formulario.
Pos-condiciones	

1.	Se exporta la gráfica realizada a documento.	
Flujos alternativos		
Flujo alternativo 1.a No hay conexión con la BD de GRHS		
1.	El sistema muestra un mensaje. "No existen datos para mostrar".	
Pos-condiciones		
1.	Se exporta los datos de la gráfica.	
Validaciones		
1.	NA	
Conceptos	Gráfico	
Requisitos especiales	Usabilidad, Eficiencia	
Asuntos pendientes	NA	

Prototipo elemental de interfaz gráfica de usuario

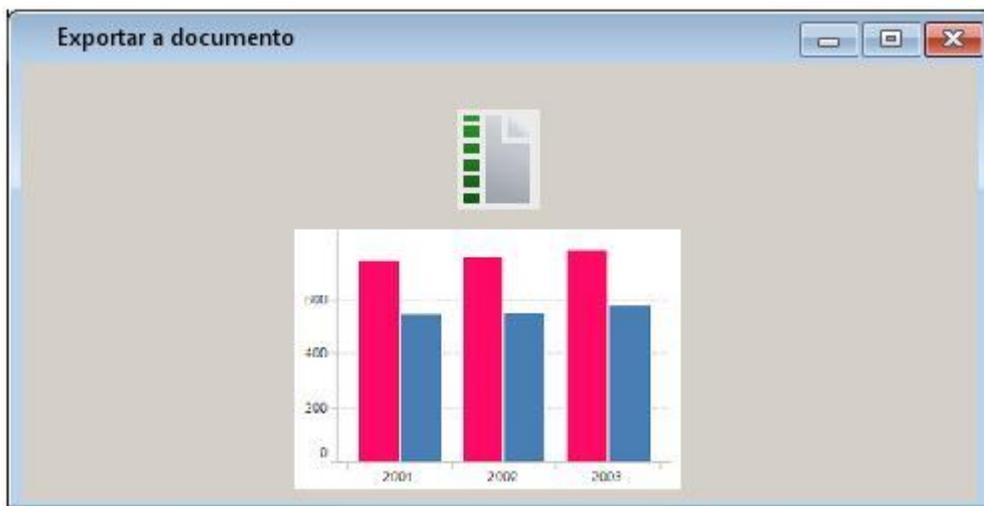


Figura 15. Prototipo de interfaz exportar a documento

Especificación de requisito: cambiar para tabla.

Descripción textual del requisito

Precondiciones	El usuario que desee cambiar para tabla debe estar autenticado en el sistema GRHS v2.0.
Flujo de eventos	
Flujo básico Cambiar para tabla	
1.	Selecciona el módulo del cuál desea visualizar las estadísticas en forma de tabla. Si no se muestran los datos graficados ver flujo alternativo 1.a.

2.	Se selecciona el icono de cambiar para tabla.
Pos-condiciones	
1.	Se cambia la visualización de los datos de gráfico para tabla.
Flujos alternativos	
Flujo alternativo 1.a No hay conexión con la BD de GRHS	
1.	El sistema muestra un mensaje. "No existen datos para mostrar".
Pos-condiciones	
1.	Se cambia de gráfico para tabla..
Validaciones	
1.	NA
Conceptos	Gráfico
Requisitos especiales	Usabilidad, Eficiencia
Asuntos pendientes	NA

Prototipo elemental de interfaz gráfica de usuario



Figura 16. Prototipo de interfaz cambiar para tabla

Especificación de requisito: graficar por un criterio de búsqueda.

Descripción textual del requisito

Precondiciones	El usuario que desee graficar por un criterio de búsqueda debe estar autenticado en el sistema GRHS v2.0.
Flujo de eventos	
Flujo básico Graficar por un criterio de búsqueda	

1.	Selecciona el módulo del cuál desea visualizar las estadísticas en gráficos.
2.	Seleccionar en el menú izquierdo el elemento del cual se desea visualizar los datos.
3.	Se selecciona los criterios de búsqueda. Si no se muestran los datos graficados ver flujo alternativo 1.a.
4.	Se elige el criterio por el cual se desee graficar. Si no se muestran los datos graficados ver flujo alternativo 2.a.
Pos-condiciones	
1.	Se ven las estadísticas del criterio seleccionado en gráfica.
Flujos alternativos 1.a Graficar sin haber seleccionado el criterio por el cual graficar	
1.	El sistema muestra un mensaje. "Seleccione un criterio del cuál desea graficar".
Flujos alternativos 2.a No hay conexión con la BD de GRHS	
1.	El sistema muestra un mensaje. "No existen datos para mostrar".
Pos-condiciones	
1.	Se grafica de acuerdo al criterio seleccionado.
Validaciones	
1.	NA
Conceptos	Gráfico
Requisitos especiales	Usabilidad, Eficiencia
Asuntos pendientes	NA

Prototipo elemental de interfaz gráfica de usuario

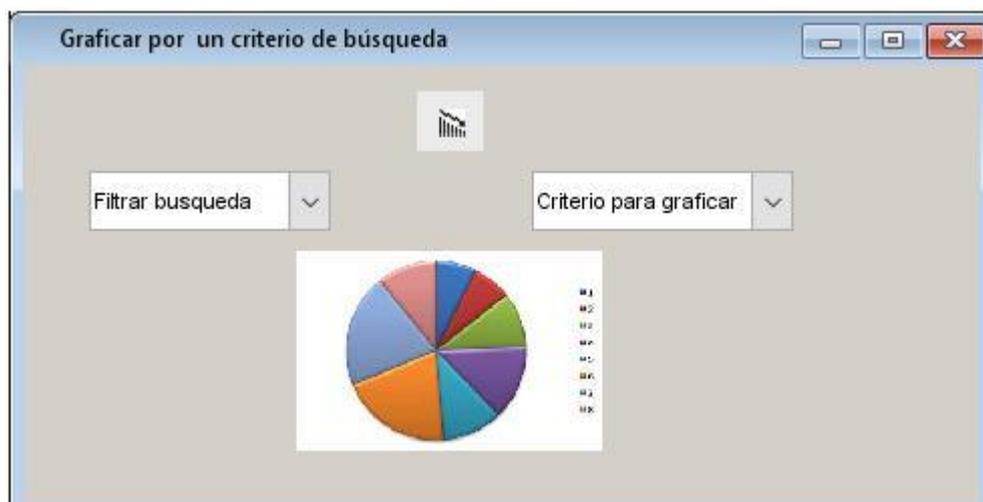


Figura 17. Prototipo de interfaz graficar por un criterio de búsqueda

Descripción general

Guardar imagen de la gráfica deseada.

Condiciones de ejecución

Estar autenticado.

SC Guardar

Tabla 15. Diseño del caso de prueba guardar imagen de la gráfica realizada.

Escenario	Descripción	Respuesta del sistema	Flujo central
EC 1.1 Seleccionar el icono de imagen.	El usuario autenticado selecciona el icono de imagen.	Se guarda la imagen de la gráfica realizada.	Se selecciona el módulo del cuál se desea visualizar las estadísticas en forma de gráficos. Se elige el modo en que se deseen ver los datos. Puede ser gráfica de pastel o de barra. Se selecciona el icono imagen. 
EC 1.2 Seleccionar el icono de imagen y no hay datos guardados en la base de datos.	El usuario autenticado selecciona el icono de imagen.	Se guarda una imagen en blanco por encontrarse vacía la BD.	Se selecciona el módulo del cuál se desea visualizar las estadísticas en forma de gráficos. Se elige el modo en que se deseen ver los datos. Puede ser gráfica de pastel o de barra. Se selecciona el icono imagen. 

Descripción general

Permite ver los datos actualizados del módulo seleccionado.

Condiciones de ejecución

Estar autenticado.

SC Actualizar

Tabla 16. Diseño del caso de prueba actualizar los datos de la gráfica.

Escenario	Descripción	Respuesta del sistema	Flujo central
-----------	-------------	-----------------------	---------------

EC 1.1 Actualizar los datos de la gráfica.	El usuario autenticado selecciona el icono de actualizar para actualizar los datos que muestra la gráfica.	Se actualiza los datos de la gráfica.	Seleccionar el módulo del cuál se desea ver los datos actualizados, luego marcar el icono de actualizar. 
EC 1.2 Actualizar los datos de la gráfica y no hay datos guardados en la base de datos.	El usuario autenticado selecciona el icono de actualizar para actualizar los datos que muestra la gráfica.	No se actualiza los datos de la gráfica.	Seleccionar el módulo del cuál se desea ver los datos actualizados, luego marcar el icono de actualizar. 

Descripción general

Graficar de acuerdo a la gráfica seleccionada

Condiciones de ejecución

Estar autenticado.

SC Graficar

Tabla 17. Diseño del caso de prueba graficar de acuerdo a la gráfica seleccionada.

Escenario	Descripción	Respuesta del sistema	Flujo central
EC 1.1 Graficar de acuerdo a la gráfica seleccionada	El usuario autenticado selecciona el tipo de gráfica que desee.	El sistema muestra las estadísticas en el tipo de gráfica seleccionada.	Selecciona el módulo del cuál se desea visualizar las estadísticas en gráficos. Se elige el modo en que se deseen ver los datos. Puede ser gráfica de pastel o de barra. 
EC 1.2 Graficar de acuerdo a la gráfica seleccionada y no hay datos guardados en la base de datos.	El usuario autenticado selecciona el tipo de gráfica que desee.	El sistema muestra la gráfica seleccionada pero sin estadísticas por encontrarse la BD vacía.	Selecciona el módulo del cuál se desea visualizar las estadísticas en gráficos. Se elige el modo en que se deseen ver los datos. Puede ser gráfica de pastel o de barra. 

Descripción general

Exportar a documento la gráfica realizada.

Condiciones de ejecución

Estar autenticado.

SC Exportar a documento

Tabla 18. Diseño del caso de prueba exportar a documento.

Escenario	Descripción	Respuesta del sistema	Flujo central
EC 1.1 Seleccionar el icono de exportar a documento.	El usuario autenticado selecciona el icono de exportar a documento.	Se exporta a documento la gráfica realizada, con una descripción si desea el usuario.	Se selecciona el módulo del cuál se desea visualizar las estadísticas en gráficos. Se elige el modo en que se desea ver los datos. Puede ser gráfica de pastel o de barra. Se selecciona el icono exportar a documento. 
EC 1.2 Seleccionar el icono de exportar a documento y no hay datos guardados en la base de datos.	El usuario autenticado selecciona el icono de exportar a documento.	Se genera el reporte pero sin la gráfica por encontrarse la BD vacía.	Se selecciona el módulo del cuál se desea visualizar las estadísticas en gráficos. Se elige el modo en que se desea ver los datos. Puede ser gráfica de pastel o de barra. Se selecciona el icono exportar a documento. 

Descripción general

Cambiar para tabla

Condiciones de ejecución

Estar autenticado.

SC Cambiar

Tabla 19. Diseño del caso de prueba cambiar para tabla.

Escenario	Descripción	Respuesta del sistema	Flujo central
EC 1.1 Seleccionar el icono de	El usuario autenticado selecciona el icono	Los datos se muestran en tabla.	Se selecciona el módulo del cuál se desea visualizar las

cambiar para tabla.	de cambiar para tabla.		estadísticas. Se selecciona el icono de cambiar para tabla. 
EC 1.2 Seleccionar el icono de cambiar para tabla y no hay datos guardados en la base de datos.	El usuario autenticado selecciona el icono de cambiar para tabla.	Las tablas se muestran vacías.	Se selecciona el módulo del cuál se desea visualizar las estadísticas. Se selecciona el icono de cambiar para tabla. 

Descripción general

Graficar de acuerdo a un criterio de búsqueda.

Condiciones de ejecución

Estar autenticado.

SC Graficar

Tabla 20. Graficar por un criterio de búsqueda.

Escenario	Escenario	Respuesta del sistema	Flujo central
EC 1.1 Filtrar todas las búsquedas por los criterios deseados	El usuario autenticado filtra las búsquedas por los criterios deseados.	El sistema muestra el resultado de las búsquedas.	Selecciona el módulo del cuál se desea visualizar las estadísticas en gráficos, seleccionar en el menú izquierdo el elemento del cual se desea visualizar los datos, se selecciona los criterios de búsqueda.
EC 1.2 Filtrar todas las búsquedas por los criterios deseados y no hay datos guardados en la base de datos.	El usuario autenticado filtra las búsquedas por los criterios deseados.	El sistema no muestra ningún resultado de las búsquedas por encontrarse vacía la BD.	Selecciona el módulo del cuál se desea visualizar las estadísticas en gráficos, seleccionar en el menú izquierdo el elemento del cual se desea visualizar los datos, se selecciona los criterios de búsqueda.

<p>EC 1.3 Graficar de acuerdo a un criterio de búsqueda</p>	<p>El usuario autenticado selecciona el criterio por el cual graficar</p>	<p>El sistema muestra la gráfica con los datos del criterio seleccionado.</p>	<p>Selecciona el criterio por del cuál se desea graficar los datos.</p>
<p>EC 1.4 Graficar de acuerdo a un criterio de búsqueda y no hay datos guardados en la base de datos.</p>	<p>El usuario autenticado selecciona el criterio por el cual graficar</p>	<p>El sistema no muestra la gráfica por encontrarse la BD vacía.</p>	<p>Selecciona el criterio por del cuál se desea graficar los datos.</p>

Glosario de términos

- **PHP:** acrónimo recursivo en inglés de PHP Hypertext Preprocessor, es un lenguaje de programación de propósito general de código del lado del servidor originalmente diseñado para el desarrollo web de contenido dinámico.
- **SQL:** es un lenguaje específico del dominio que da acceso a un sistema de gestión de bases de datos relacionales que permite especificar diversos tipos de operaciones en ellos.
- **GNU/GPL:** licencia de documentación libre. GPL especificando que se trata de la versión 3 o cualquier versión posterior.
- **JavaScript:** es un lenguaje ligero e interpretado, orientado a objetos, más conocido como el lenguaje de script para páginas web, pero también usado en numerosos entornos sin navegador.
- **iOS:** es un sistema operativo móvil de la multinacional Apple Inc. Originalmente desarrollado para el iPhone, después se ha usado en dispositivos como el iPod touch y el iPad. No permite la instalación de iOS en hardware de terceros.
- **Android:** es un sistema operativo basado en el núcleo Linux. Fue diseñado principalmente para dispositivos móviles con pantalla táctil, como teléfonos inteligentes, tabletas y también para relojes inteligentes, televisores y automóviles.
- **SVG:** Gráficos vectoriales escalables, o Gráficos vectoriales redimensionales o SVG es un formato de gráficos vectoriales bidimensionales, tanto estáticos como animados, en formato XML.
- **CSS:** (*Cascading Stylesheets*), un lenguaje de estilo que define la presentación de los documentos HTML.
- **D3:** es una librería de JavaScript para producir, a partir de datos, infogramas dinámicos e interactivos en navegadores web. Hace uso de tecnologías bien sustentadas como SVG, HTML5, y CSS.
- **DOM:** es esencialmente una interfaz de plataforma que proporciona un conjunto estándar de objetos para representar documentos HTML y XML, un modelo estándar sobre cómo pueden combinarse dichos objetos, y una interfaz estándar para acceder a ellos y manipularlos.

- **jQuery:** es una biblioteca multiplataforma de JavaScript, creada inicialmente por John Resig, que permite simplificar la manera de interactuar con los documentos HTML.
- **Licencia MIT** es una de tantas licencias de software que se origina en el Instituto Tecnológico de Massachusetts (MIT, *Massachusetts Institute of Technology*). Quizás debería llamarse más correctamente **licencia X11**, ya que es la licencia que llevaba este software de muestra de la información de manera gráfica.
- **Licencia BSD:** es la licencia de software otorgada principalmente para los sistemas BSD (Berkeley Software Distribution).
- **HTML:** (*HyperText Markup Language*), hace referencia al lenguaje de marcado para la elaboración de páginas web.
- **API:** Una herramienta que permite a desarrolladores de aplicaciones web crear gráficos a partir de los datos escogidos e incrustarlos en las páginas web. La API además ofrece una gran variedad de diseños de gráficos a escoger.
- **ASP.NET:** es un entorno para aplicaciones web desarrollado y comercializado por Microsoft. Es usado por programadores y diseñadores para construir sitios web dinámicos, aplicaciones web y servicios web XML.
- **Java Server Pages (JSP):** es una tecnología que ayuda a los desarrolladores de software a crear páginas web dinámicas basadas en HTML y XML, entre otros tipos de documentos. JSP es similar a PHP, pero usa el lenguaje de programación Java.
- **MySQL:** es un sistema de gestión de bases de datos relacional desarrollado bajo licencia dual: Licencia pública general/Licencia comercial por Oracle Corporation y está considerada como la base datos de código abierto más popular del mundo.
- **XETID:** empresa de Tecnologías de la Información para la Defensa, es una empresa de origen cubano.
- **XML:** son las siglas del Lenguaje de Etiquetado Extensible. Se trata también de un lenguaje estándar que posee una Recomendación del World Wide Web.
- **SWF:** Small Web Format -formato web pequeño- para evitar confusiones con Shockwave del que deriva es un formato de archivo de gráficos vectoriales creado por la empresa Macromedia (actualmente Adobe Systems).
- **CMMI:** es un modelo para la mejora y evaluación de procesos para el desarrollo, mantenimiento y operación de sistemas de software.
- **UML:** es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad; está respaldado por el Object Management Group (OMG). Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema.

