

Universidad de las Ciencias Informáticas

Facultad 4



**Título:** *Componentes gráficos personalizables para la representación de medidores analógicos y digitales en el sistema AREX*

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

**Autor:**

Arlette Lorenzo Tovar

**Tutores:**

Ing. Adolfo Yasser Santana Rojas

Ing. Claudia María González Fernández

Lic. Luis Gabriel Viciado Carabaloso

La Habana, Cuba 2019.

“Año del 61 de la Revolución”

*La vida es una preparación para el futuro; y la mejor preparación es vivir como si no hubiera ninguno.*

*A. Einstein*

## *Declaración de autoría*

### *Declaración de autoría*

Declaro ser autor del presente trabajo y autorizo a la Facultad 4 de la Universidad de las Ciencias Informáticas de hacer uso del mismo en su beneficio.

Para que así conste firmo la presente a los \_\_\_\_\_ días del mes de \_\_\_\_\_ del año \_\_\_\_\_.

Arlette Lorenzo Tovar

---

Firma del autor

Ing. Adolfo Yasser Santana Rojas

---

Firma del tutor

Ing. Claudia María González Fernández

---

Firma del tutor

Lic. Luis Gabriel Viciado Carabaloso

---

Firma del tutor

*Datos de contacto*

**Tutor:** Ing. Adolfo Yasser Santana Rojas

**Correo electrónico:** [aysantana@uci.cu](mailto:aysantana@uci.cu)

Graduado de Ingeniería en Ciencias Informáticas (UCI) en el año 2015 por la Universidad de las Ciencias Informáticas (UCI) ubicada en Carretera a San Antonio de los Baños, Km. 2 ½. Torrens, La Lisa, La Habana, Cuba. Ha trabajado en varios proyectos e investigaciones en el centro de Informática Industrial (CEDIN) de la Facultad 4 de la UCI. Actualmente es Especialista "A" en Ciencias Informáticas.

**Tutor:** Ing. Claudia M. González Fernández

**Correo electrónico:** [cmfernandez@uci.cu](mailto:cmfernandez@uci.cu)

Graduada de Ingeniería en Ciencias Informáticas (UCI) en el año 2016 por la Universidad de las Ciencias Informáticas (UCI) ubicada en Carretera a San Antonio de los Baños, Km. 2 ½. Torrens, La Lisa, La Habana, Cuba. Ha trabajado en varios proyectos e investigaciones en el centro de Informática Industrial (CEDIN) de la Facultad 4 de la UCI. Actualmente es Especialista "B" en Ciencias Informáticas.

**Tutor: Lic.** Luis Gabriel Viciado Carabaloso

**Correo electrónico:** [viciado@uci.cu](mailto:viciado@uci.cu)

Graduado de Licenciatura en Educación en la especialidad de Física y Matemáticas en el Instituto Superior Pedagógico José Martí de Camagüey en el año 1980. Fundador del proyecto Laboratorios Virtuales en la Facultad 5. Desarrolla proyectos de videojuegos y simulaciones para el aprendizaje y el entrenamiento profesional. Ha impartido clases de Programación, Informática Forense, Hacking Ético y Ética Informática. Tiene más de cuarenta años en la impartición de docencia en el área de la ingeniería.

*Dedicatoria*

*Mis logros durante los cinco años de la carrera, todo mi estudio, mi título de Ingeniera en Ciencias Informáticas van dedicado a mis abuelos y abuelas estén o no presentes físicamente en mi vida, a mis padres que han estado en todo momento conmigo apoyándome en las malas y en las buenas decisiones.*

*A mis hermanas y a mi hermanito el más pequeño de la familia.*

*A todas aquellas personas que han formado parte de mi vida, aunque algunas ya no estén a mi lado.*

### Agradecimientos

*Son muchas las personas a las que le tengo que agradecer y me siento muy feliz de que así sea, de que existan tantas personas importantes en mi vida.*

*Quiero empezar por las personas más importantes para mí, mi abuela Rosa y mi abuelo Jesús que, aunque no esté ya a mi lado lo adoro con todo mi ser, y les agradezco aún más por haber traído al mundo al mejor papito del planeta, a mi abuela Julita que ya no está conmigo y mi abuelo José que no me dejaron conocerlo, pero me regalaron a mi mamita bella, la súper mamá.*

*Ellos dos son mis mayores tesoros no sé qué voy a hacer cuando ya no los tenga a mi lado, pues quién me aguantará mis malcriadeces y quién celebrará mis triunfos.*

*A mi tío Jesusito que yo sé que no sabe demostrar cuanto me quiere y me molesta mucho, pero lo adoro.*

*A mis hermanas, Ailén la muchacha que nunca se queda callada y siempre está con la mala forma y la mala forma, pero yo sé que ella me quiere, aunque no me lo diga muy seguido.*

*A Lore la hermanita callada, la que parece que no rompe un plato, la que se pone insistente cuando le dices que vas a verla y pasó la hora que le habías dicho.*

*A Boris Alan el más pequeño de la familia pero que no se le olvida absolutamente nada, menos mal que nació este pequeñín porque el apellido Lorenzo tiene que seguir adelante.*

*A Eliannys por saber sobrellevar a mi segundo tesoro mi papito y por ayudarme en todo lo que está en sus manos.*

*A mi compañero de vida, de carrera, de éxitos estudiantiles y personales que ha estado conmigo en las malas y en las peores mi novio Ariel (mi titiki) que supo mantenerse despierto hasta altas horas de la madrugada dándome consultas personales de mis peores pesadillas: las Matemáticas Discretas y las programaciones.*

*A mi perrita consentida Leira, mi pequeñita que me despierta en la madrugada para dormir conmigo, ella siempre hace sus monerías para hacerme reír por muy triste que yo esté.*

## *Agradecimientos*

*A mi grupo de amigos preferidos bien pequeño, pero no importa: Maday mi loquita compañera de cuarto con la que siempre hay que pelearse para recoja su desorden.*

*A Lídice la gruñona del equipo que no hay manera de que ella deje de decir las cosas que duelen en la cara.*

*A Carlos el mensajero que hay que seguirle el ejemplo hay que dormir las mañanas, Tania la niña de los favores junto con Eliuvis el más delgado del grupo, Claudia y Tito los abuelos de la familia que casi nunca.*

*A todos los profesores que he tenido durante la carrera en especial a la profe Yanelis Benítez de Cálculos en la FICI, ella siempre supo obtener una sonrisa de sus estudiantes y regalarnos la de ella.*

*A mis tutores Adolfo que siempre me mantuvo correteando, pero yo sabía que era para bien y a Claudia por aplacar siempre a la fierecilla. Ellos supieron sobrellevar a su tesista y me apoyaron en todo momento.*

*A los miembros del tribunal y a mi oponente por sus incontables comentarios y recomendaciones para lograr hoy estar aquí.*

*A mis compañeros de aula, aunque ya no estamos todos los que empezamos.*

*A mis compañeras de apartamento, a toda la gente del 136 el edificio de muchos de los que nos graduaremos este año.*

*Sé que se quedan muchas personas, pero les agradezco infinitamente por de una forma u otra ser parte de mi vida.*

*Resumen*

El vertiginoso desarrollo de las Tecnologías de la Información y la Comunicación ha proporcionado que la informática industrial se haya convertido en los últimos tiempos en compañera inseparable de la automatización industrial. Por su parte, el Centro de Informática Industrial perteneciente a la Facultad 4, desarrolla el Sistema de Medición y Adquisición de Datos AREX teniendo como misión ser desplegado en las industrias.

La Interfaz Hombre Máquina que provee el sistema AREX posee un conjunto de paletas de componentes gráficos dentro las cuales se encuentra la de medidores, actualmente ésta paleta no posee componentes que permitan la representación visual de variables como la humedad relativa o el flujo de un gas o un líquido. Además, emplea imágenes en los componentes gráficos afectando la calidad de visualización al aplicar el redimensionamiento, no se permite la agrupación y desagrupación de los componentes gráficos que componen a un medidor provocando una pobre personalización y carencia de la atonicidad.

Para dar solución a estas limitantes se ha desarrollado una paleta especializada en medidores que permitan una mayor personalización, calidad de visualización y atonicidad de los medidores. La propuesta de solución fue desarrollada mediante la metodología AUP-UCI en su escenario cuatro: historia de usuario. Se utilizó el framework de desarrollo Qt y como lenguaje de modelado de Qt QML, para el diseño gráfico se empleó la tecnología SVG, como herramienta para la creación de los SVG se utilizó el editor Inkscape. Se realizaron los diagramas necesarios para el modelado, implementación y prueba, obteniéndose una paleta de componentes que cumple con los requerimientos funcionales y no funcionales que permita la reducción de tiempo para ser desplegados en la industria.

**Palabras clave:** medidores, industria, componente gráfico, paleta de componentes.

## Índice

Introducción .....	1
Capítulo I: Fundamentación teórica .....	6
Introducción.....	6
1.1    Sistemas de medición.....	6
1.2    Norma Sociedad de Instrumentos de América (ISA).....	7
1.3    Estudio de sistemas similares.....	7
1.4    Instrumentos de medición.....	9
1.4.1    Características principales de un instrumento de medida .....	10
1.4.2    Clasificación de los instrumentos de medición.....	10
1.5    Características de los medidores eléctricos digitales .....	10
1.5.1    Señal digital.....	10
1.5.2    Variable digital.....	11
1.5.3    Reloj digital.....	11
1.5.4    Flujómetro digital .....	11
1.6    Características de los medidores eléctricos analógicos .....	12
1.6.1    Señal analógica .....	12
1.6.2    Variable analógica .....	12
1.6.3    Higrómetro analógico.....	12
1.6.4    Reloj analógico .....	13
1.7    Componente gráfico .....	14
1.8    Módulo AREX-Editor.....	14
1.9    Arquitectura modular .....	15
1.10    Herramientas, tecnologías y metodología.....	15
1.10.1    Gráficos vectoriales escalables ( <i>Scalable Vector Graphics SVG</i> ).....	15
1.10.2    Editor gráfico de vectores .....	16
1.10.3    Marco de trabajo.....	16
1.10.4    Entorno de desarrollo .....	16
1.10.6    Lenguaje de modelado .....	17
1.10.7    Herramienta CASE .....	17

Capítulo II: Características y diseño del sistema.....	20
Introducción.....	20
2.1    Modelo de dominio .....	20
2.2    Requisitos.....	21
2.2.1    Requisitos funcionales.....	21
2.2.2    Requisitos no funcionales.....	24
2.3    Historias de usuarios .....	24
2.4    Fase de planificación .....	26
2.6    Plan de iteraciones .....	28
2.8    Diagrama de paquetes .....	29
2.10    Patrones de diseño.....	30
Capítulo III: Implementación y pruebas del sistema .....	32
3.1    Modelo de implementación .....	32
3.1.1    Diagrama de componentes.....	32
3.1.2    Diagrama de despliegue.....	33
3.2    Estándar de codificación.....	34
3.3    Solución del problema .....	36
3.4    Pruebas de <i>software</i> .....	41
3.4.1    Tipos de pruebas de <i>software</i> .....	41
3.5    Pruebas internas .....	43
3.5.1    Pruebas funcionales .....	43
Conclusiones generales.....	48
Glosario de términos.....	49
Recomendaciones .....	50
Referencias bibliográficas.....	51
Anexo 1 Historias de usuarios .....	53
Anexos 2 Pruebas de aceptación .....	59
Anexo 3 Diseños de casos de pruebas.....	68

*Índice de figuras*

Figura 1 Reloj digital ( <i>The Watches Times</i> 2018).....	11
Figura 2 Flujómetro digital (Suministros hidráulicos del sureste S.A. DE C.V. 2017).....	12
Figura 3 Higrómetro analógico (Second Episode S.L 2017).....	13
Figura 4 Partes de un reloj analógico de pulsera (ZonaCasio.com 2012). ....	13
Figura 5 Flujómetro analógico (Suministros hidráulicos del sureste s.a. de C.V. 2017).....	14
Figura 6 Modelo de dominio (Elaboración propia).....	20
Figura 7 Diagrama de paquetes (Elaboración propia).....	29
Figura 8 Diagrama de componentes (Elaboración propia). ....	33
Figura 9 Diagrama de despliegue de AREX (Centro de Informática Industrial, CEDIN. 2019). ....	34
Figura 10 Ejemplo de una función desarrollada (Elaboración propia). ....	35
Figura 11 Interfaz del módulo AREX-Edición. ....	37
Figura 12 Paleta de componentes Meters 2.....	37
Figura 13 Propiedades de los componentes gráficos.....	37
Figura 14 Higrómetro analógico definir valores mínimos y máximos de la escala (Elaboración propia). ....	38
Figura 15 Higrómetro analógico modificar valores mínimos y máximos de la escala (Elaboración propia)..	38
Figura 16 Modificación dinámica de los rangos (Elaboración propia).....	38
Figura 17 Adicionar rango de la escala en el medidor (Elaboración propia).....	38
Figura 18 Flujómetro digital, configurar opacidad (Elaboración propia).....	39
Figura 19 Flujómetro digital, propiedad opacidad modificada (Elaboración propia).....	39
Figura 20 Definición de la amplitud de la escala radial (Elaboración propia).....	39
Figura 22 Modificación del color de la escala del medidor (Elaboración propia).....	39
Figura 21 Definición de ángulo inicial y final de la escala radia (Elaboración propia). ....	39
Figura 23 Flujómetro analógico configuración de fondo (Elaboración propia). ....	40
Figura 24 Flujómetro analógico color de fondo modificado (Elaboración propia).....	40
Figura 25 Medidor radial configuración del color de la aguja (Elaboración propia).....	40
Figura 26 Medidor radial color de la aguja (Elaboración propia). ....	40
Figura 27 Higrómetro analógico configuración del valor y la unidad de medida (Elaboración propia). ....	41
Figura 28 Higrómetro analógico valor y la unidad de medida modificados (Elaboración propia). ....	41

*Índice de tablas*

Tabla 1 Representación en la norma ISA de los componentes a desarrollar. ....	7
Tabla 2 Requisitos funcionales. ....	21
Tabla 3 Historia de usuario#1 ....	25
Tabla 4 Historia de usuario#2 ....	26
Tabla 5 Historia de usuario#3 ....	26
Tabla 6 Estimación de esfuerzos e iteraciones por historia de usuario. ....	27
Tabla 7 Plan de entrega.....	29
Tabla 9 Diseño de casos de pruebas.....	43
Tabla 10 Prueba de aceptación #1 ....	44
Tabla 11 Prueba de aceptación #2 ....	44
Tabla 12 Prueba de aceptación #3 ....	45
Tabla 13 Historia de usuario#4 ....	53
Tabla 14 Historia de usuario#5 ....	53
Tabla 15 Historia de usuario#6 ....	53
Tabla 16 Historia de usuario#7 ....	54
Tabla 17 Historia de usuario#8 ....	54
Tabla 18 Historia de usuario#9 ....	55
Tabla 19 Historia de usuario#10 ....	55
Tabla 20 Historia de usuario#11 ....	55
Tabla 21 Historia de usuario#12 ....	56
Tabla 22 Historia de usuario#13 ....	56
Tabla 23 Historia de usuario#14 ....	56
Tabla 24 Historia de usuario#15 ....	57
Tabla 25 Historia de usuario#16 ....	57
Tabla 26 Historia de usuario#17 ....	57
Tabla 27 Historia de usuario#18 ....	58
Tabla 28 Historia de usuario#19 ....	58
Tabla 29 Prueba de aceptación #4 ....	59
Tabla 30 Prueba de aceptación #5 ....	59

Tabla 31 Prueba de aceptación #6 .....	60
Tabla 32 Prueba de aceptación #7 .....	60
Tabla 33 Prueba de aceptación #8 .....	61
Tabla 34 Prueba de aceptación #9 .....	61
Tabla 35 Prueba de aceptación #10 .....	62
Tabla 36 Prueba de aceptación #11 .....	62
Tabla 37 Prueba de aceptación #12 .....	63
Tabla 38 Prueba de aceptación #13 .....	63
Tabla 39 Prueba de aceptación #14 .....	64
Tabla 40 Prueba de aceptación #15 .....	64
Tabla 41 Prueba de aceptación #16 .....	65
Tabla 42 Prueba de aceptación #17 .....	65
Tabla 43 Prueba de aceptación #18 .....	66
Tabla 44 Prueba de aceptación #19 .....	66

### *Introducción*

Las Ciencias de la Computación tienen un gran impacto a nivel mundial tanto en las actividades que desarrollan los sectores productivos y de servicios, como en la vida cotidiana. Con el transcurso del tiempo se ha percibido que el ser humano posee cada vez mayor dependencia de las Tecnologías de la Información y la Comunicación (TIC).

En Cuba se le ha dado alta prioridad al uso de las TIC, para apoyar la gestión y el desarrollo en diferentes sectores como la salud, la educación y el deporte. Cuando se hace referencia a la informatización de la sociedad cubana, se está hablando de poner en uso las tecnologías de la informática en los diferentes sectores de la misma y con ello lograr una mayor generación de conocimiento y riquezas.

La informática industrial como rama de la ingeniería que integra y unifica el campo industrial a la informática es la encargada del tratamiento automático de la información proveniente de los procesos industriales; se ha convertido en los últimos tiempos en compañera inseparable de la automatización industrial (Miranda 2017).

En la actualidad, la cantidad de organizaciones y entidades, que usan sistemas informáticos encaminados a brindar ayuda han aumentado. Esto es debido a que con frecuencia la tecnología requiere de mayor conocimiento experto y toma de decisiones, como herramienta primordial en empresas de gran nivel hoy en día.

Los instrumentos de medida son un eslabón fundamental en cualquier sector de la sociedad. Día a día se desarrollan y evidencian diferentes actividades que requieren de algún tipo de medición, por ejemplo, el registrar el gasto de luz y agua mensual o verificar el voltaje de algún electrodoméstico. Las mediciones permiten verificar la veracidad y falsedad de un evento, de tal manera que es una parte importante del desarrollo de la ciencia, pues permite desechar ideas falsas e ir modificando teorías (Instrumentos de Medición 2019).

La Universidad de las Ciencias Informáticas (UCI) cuenta con diferentes facultades docentes. En la facultad 4 se encuentra el Centro de Informática Industrial (CEDIN), tiene como misión brindar productos informáticos de automatización industrial a partir de la integración de los procesos productivos, investigativos y docentes, con un alto valor agregado capaces de satisfacer las necesidades y expectativas de los clientes y la sociedad (Centro de Informática Industrial, CEDIN. 2019). Entre los proyectos que desarrolla el centro se encuentra el Sistema de Medición y Adquisición de Datos AREX.

El sistema AREX 3.0, está pensado para la medición y adquisición de datos en procesos de pequeña y mediana complejidad. Su instalación puede ser monolítica en un solo ordenador, o distribuida en varios e involucra diferentes componentes: aplicaciones no gráficas para adquisición de datos, publicación y almacenaje de los mismos; aplicaciones gráficas en dispositivos móviles o de escritorio, para la visualización de datos; y dispositivos físicos que conectan sensores y actuadores al sistema (Centro de Informática Industrial, CEDIN. 2019).

Además del control manual en las interfaces de usuario, se pueden configurar rutinas de control automático con apreciable grado de complejidad. El sistema es capaz de generar seis tipos de alarmas para el monitoreo de hasta 300 variables en los procesos. Su extensibilidad permite la adición de nuevos tipos de componentes gráficos a mostrar en las interfaces de usuario, así como nuevos protocolos de comunicación con dispositivos físicos, esta se realiza mediante el protocolo de comunicación MODBUS utilizando la conexión Serie y/o *Ethernet* (Centro de Informática Industrial, CEDIN. 2019).

El sistema incorpora la visualización de gráficas de tendencia o tiempo real para las variables adquiridas. Fue desarrollado en el *framework* QT, por lo que puede ser compilado para múltiples plataformas, consume pocos recursos de *hardware*, hace uso de una variedad de protocolos estándares y permite monitoreo y control de alta velocidad sobre procesos. Es aplicable en varios entornos como son: implementación de sistema de alarmas, control remoto de dispositivos y control de procesos industriales (Centro de Informática Industrial, CEDIN. 2019).

El sistema AREX 3.0 se podría definir como la integración de la tecnología en el diseño inteligente de un recinto cerrado.

El sistema está compuesto por cinco módulos funcionales (Centro de Informática Industrial, CEDIN. 2019):

**AREX-Recolección:** mediante la implementación de protocolos y haciendo uso de las interfaces de transporte, permite la comunicación directa con los dispositivos de campo, el procesamiento y adquisición de los datos.

**AREX-Históricos:** tiene como función principal almacenar todos los datos recolectados y generados en el sistema, para que puedan ser consultados con posterioridad. Así, será posible estudiar, a través de La Interfaz Hombre Máquina (*HMI*, por sus siglas en inglés), el comportamiento del proceso monitoreado, de los operadores que interactúan con él y del sistema en sí mismo.

**AREX-Seguridad:** permite la autenticación de los usuarios. Cada vez que un usuario es autenticado le envía la configuración al módulo desde el cual se autenticó.

**AREX-Edición:** encargado de las actividades de configuración y edición de los mímicos de los despliegues que intervienen en la realización de un proceso.

**AREX-Ejecución:** a partir de la configuración recibida en forma de XML, permite a los operadores, tener acceso a la información de los procesos que se miden.

Tanto para el Arex-Edición como para el Arex-Ejecución, el HMI provee una biblioteca de componentes gráficos, los cuales son agrupados en paletas. Entre las paletas existentes se encuentra la paleta especializada en la representación de medidores, la cual está conformada por componentes gráficos que tienen como objetivo la representación de la información obtenida de las variables de origen analógico y digital capturadas por los dispositivos que se encuentran situados en el área de supervisión y control.

Dicha paleta posee las siguientes deficiencias:

- Algunos de los componentes existentes en la paleta emplean imágenes para su representación; por lo que sufren de pérdida de calidad de visualización al realizarse el redimensionamiento.
- Carencia de componentes gráficos que representen medidores que permitan la representación de variables de medición como: humedad relativa, flujo de un gas o un líquido.
- Los componentes gráficos que componen un medidor no pueden ser desacoplados, lo cual dificulta la modificación de éstos individualmente, afectando la atomicidad.
- Poseen restricciones en la gestión dinámica de escalas y rangos, limitando la personalización mediante las propiedades definidas, lo que conlleva a la pérdida de la capacidad del sistema de ajustarse a las necesidades del área a representar en el despliegue.

A partir de la **situación problemática** descrita anteriormente se plantea como **problema a resolver** ¿Cómo proveer una mejor calidad de representación, personalización y atomicidad de los medidores pertenecientes al sistema AREX?

Se define como **objeto de estudio:** El proceso de representación y configuración de componentes gráficos para procesos industriales; como **campo de acción** la medición y adquisición de datos de medidores analógicos y digitales en el sistema AREX.

El **objetivo general** de este trabajo de diploma es: Desarrollar componentes gráficos personalizables para representar medidores analógicos y digitales en el sistema AREX.

Una vez concluida la investigación se espera como **posibles resultados:**

1. Paleta de componentes gráficos especializada en medidores.
2. Componentes gráficos personalizables especializados en medidores digitales y analógicos.

**Tareas de investigación:**

- Elaboración del marco teórico de la investigación a través del estudio del estado del arte que existe actualmente sobre el tema.
- Identificación de los principales elementos que representan medidores en los procesos industriales y su representación en el sistema AREX.
- Caracterización de los principales elementos que representan los medidores en los procesos correspondientes a sistemas de medición.
- Realización del levantamiento de requisitos funcionales y no funcionales.
- Implementación de componentes gráficos que brinden solución al problema planteado.
- Realización de pruebas para validar el cumplimiento de los requerimientos.

Se emplearon los siguientes **métodos investigativos** para el cumplimiento de las tareas de investigación, clasificados en teóricos y empíricos.

Los **métodos teóricos** permiten estudiar las características del objeto de investigación que no son observables directamente, facilitan la construcción de modelos e hipótesis de investigación; de ellos se emplearon los siguientes:

- **Histórico-lógico:** el empleo de este método facilitó realizar el estudio del estado del arte, investigar sobre los sistemas similares y las tendencias actuales referidas a la evolución de los sistemas de medición en el mundo.
- **Modelación:** este método se utilizó como herramienta para comprender el problema y crear modelos abstractos del mismo que permitan el diseño de la solución. Además, posibilita la elaboración de diagramas para el desarrollo de los componentes gráficos.
- **Análisis-síntesis:** la utilización de este método permitió realizar el análisis teórico e identificar los principales conceptos a incluir en la fundamentación teórica y el análisis de la información, permitiendo extraer los elementos importantes relacionados con los medidores analógicos y digitales.

Los **métodos empíricos** representan un nivel de la investigación cuyo contenido procede de la experiencia y es sometido a determinada elaboración racional; de ellos se emplearon los siguientes:

**Observación:** su empleo permite observar y estudiar el funcionamiento existente en los despliegues del sistema AREX mediante el comportamiento de los dispositivos de campo en las propiedades de los objetos gráficos y sumarios empleados para la toma de decisiones de los operadores.

La siguiente investigación está conformada por tres capítulos:

### **Capítulo I. Fundamentación teórica**

Se exponen los diferentes conceptos y términos necesarios para una mejor comprensión de la investigación. Se realiza una investigación sobre los sistemas similares. Además, se efectúa un estudio de las metodologías y herramientas de desarrollo que se pueden utilizar y se justifica la elección de cada una de ellas.

### **Capítulo II. Características y diseño del sistema**

Consiste en la descripción de la propuesta solución, explica el proceso de desarrollo de *software* que conlleva a la solución. Se exponen algunos de los elementos básicos a tener en cuenta para una solución exitosa: requisitos funcionales y no funcionales, arquitectura del *software* y los distintos artefactos creados en el proceso, así como el modelo de dominio.

### **Capítulo III. Implementación y pruebas del sistema**

Se abordan los aspectos relacionados con la implementación de los componentes gráficos. Además, se realiza la validación de la investigación desarrollada mediante la realización de pruebas de *software*, que garantizan que las funcionalidades cumplan las especificaciones requeridas.

*Capítulo I: Fundamentación teórica*

## **Introducción**

En este capítulo se abordará el estado del arte y los principales conceptos asociados al sistema AREX, así como de los demás elementos involucrados en la investigación. Se analizarán las principales tecnologías y herramientas para dar solución al problema y objetivo planteado.

### **1.1 Sistemas de medición**

Son sistemas que contienen un conjunto de instrumentos y medidores, normas, operaciones, métodos, instalaciones, *software*, personal, medio ambiente y los recursos utilizados para cuantificar una unidad de medida, constituyen una herramienta muy útil para mejorar los procesos y servicios en las empresas.

Estos sistemas tienen tres atributos principales:

- Tener ciclos cortos de recopilación, procesamiento y acción (Israel Flores 2017).

Es importante mantener breves los ciclos de medición para que la información no quede obsoleta, es decir; que no transcurra demasiado tiempo entre cuando se toma la medición y cuando se implementan las acciones de mejora.

- Los indicadores deben estar definidos para medir los objetos (Israel Flores 2017).

Si los datos responden justo a lo que se necesita saber, cuanto menos información se tenga, más enfocadas y eficientes serán las acciones (Israel Flores 2017).

- Diseñar el sistema de medición para recibir la información y después transformarla en mejoras (Israel Flores 2017).

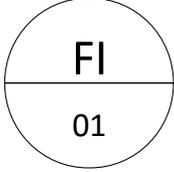
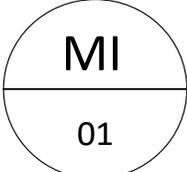
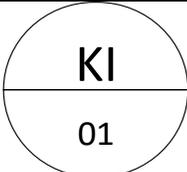
Con el estudio de los sistemas de medición se pudieron identificar las variables y componentes que intervienen en los procesos de medición. Esto resulta de vital importancia para identificar posibles requerimientos o funcionalidades que brindará la paleta de componentes en aras de desarrollar la propuesta de solución.

La siguiente tabla muestra los componentes que fueron identificados, su representación en la norma ISA<sup>1</sup> y su descripción técnica, ver tabla 1 Representación en la norma ISA de los componentes a desarrollar.

---

<sup>1</sup> ISA: Norma internacional para la representación de instrumentación industrial.

Tabla 1 Representación en la norma ISA de los componentes a desarrollar.

Nombre	Figura	Descripción
Flujómetro (análogo y digital)		Es un instrumento que se usa para medir el caudal lineal, no lineal, de masa o volumétrico de un líquido o gas.
Higrómetro (análogo)		Dispositivo utilizado para medir la humedad relativa del aire y el nivel de humedad relativa que se registra en la atmósfera.
Reloj (análogo y digital)		Dispositivo utilizado para indicar la hora ya sea digital o analógicamente.

Además, serán desarrollados los componentes gráficos escala radial, aguja, medidor radial y medidor lineal para así lograr una mayor atomicidad de los componentes a representar.

### 1.2 Norma Sociedad de Instrumentos de América (ISA)

En instrumentación y control, se emplea un sistema especial de símbolos con el objeto de transmitir de una forma más fácil y específica la información. Esto es indispensable en el diseño, selección, operación y mantenimiento de los sistemas de control.

Cada instrumento estará representado por una etiqueta o identificación y por un símbolo. La etiqueta es un conjunto de letras y números que indican cuál es la variable medida o contralada y cuáles son las funciones del instrumento. La simbología empleada en el diagrama permite ubicar el instrumento, determinar el tipo de señales empleadas y otras características de los mismos (Oscar Páez Rivera 2015).

### 1.3 Estudio de sistemas similares

Con el objetivo de detectar características asociadas a la representación de medidores analógicos y digitales en HMI pertenecientes a sistemas SCADAS se estudiaron tres soluciones: Eros, *EcoStructureWare* *Schneider Electric*, *Siemens*, la selección de la primera solución a estudiar fue a partir de que es un *software* de nuestro país mientras que las restantes son de carácter extranjeros y son líderes en el mercado mundial. A continuación, se realiza una caracterización de las mismas:

### **Sistema de Supervisión y Control de Procesos EROS**

Es un sistema para la supervisión y control de procesos industriales completamente cubano, diseñado y desarrollado en la división de automatización, SERCONI<sup>2</sup>.

Este sistema realiza variadas funciones dentro del entorno de la dirección de los procesos. Facilita a los operadores, ingenieros, supervisores y directivos operar y dirigir cualquier proceso con eficiencia y productividad. EROS puede trabajar acoplado con diversos sistemas de colección de datos y control, como elemento único o formando parte de una red industrial.

Tiene en cuenta todas las características de las variables medidas y realiza un potente tratamiento estadístico y determinístico de las mismas con sólo configurarlo. Facilita el mando a distancia y el control desde la aplicación, que son herramientas que potencian el automatismo del proceso tecnológico.

Es un sistema distribuido en el cual sus diferentes componentes se interconectan a través de la intranet empresarial. Los componentes que cooperan entre sí son: estaciones de medición, estaciones de visualización, servidores de reportes, páginas WEBS y otras aplicaciones que se desarrollen usando *ErosNet* y *ActiveX*. Estos componentes pueden estar en ordenadores separados vinculados a través de una red *ethernet* o en un mismo ordenador (Grupo Eros Serconi. 2014).

### ***EcoStruxureWare Schneider Electric***

Es el sistema de plataforma y arquitectura compatible con el IoT, de uso inmediato, abierto e interoperativo de *Schneider Electric*. Proporciona soluciones que le permiten concentrarse en aumentar la eficiencia del capital y la efectividad del operador, maximizar el rendimiento y la confiabilidad de los activos, administrar la producción y la seguridad.

Está disponible para hogares, edificios, centros de datos, infraestructuras e industrias. En particular *Electrical safety* asegura que los sistemas eléctricos estén actualizados, es de vital importancia para mantener el hogar seguro.

---

<sup>2</sup> **SERCONI**. La empresa de servicios técnicos de computación, comunicaciones y electrónica para la industria del níquel "Rafael Fausto Orejón Forment", fue creada por la Resolución 125 del 4 de marzo de 1996 del Ministro de Economía y Planificación. Está integrada por actividades afines: *Comunicaciones, Computación, Instrumentación y Automática*.

Para disfrutar de un entorno más seguro y una fuente de alimentación sin interrupciones, es esencial realizar actualizaciones de instalación periódicas. Con la tecnología líder en la industria de *Schneider Electric*, este sistema solo funciona con tecnología de *Schneider* (*Schneider Electric* 2018).

### **Sistema SIMATIC WinCC**

Es un SCADA de *Siemens* el cual puede ser utilizado en combinación con los controladores *Siemens*; es un sistema integrado exclusivo diseñado para el despliegue con todas las aplicaciones de fabricación y en todos los sectores. Tanto si necesita soluciones de fabricación como de automatización de procesos o para tareas de infraestructura, *SIMATIC* puede convertirse en un elemento decisivo para ayudar en el aumento de la productividad.

Se encuentra en el *Totally Integrated Automation Portal (TIA Portal)*; éste es un sistema de ingeniería que permite configurar de forma eficiente todos los procesos de planificación y producción (*Siemens AG* 2009). Una vez realizado el proceso de análisis del estado del arte sobre los sistemas similares al sistema AREX se pudo arribar a las siguientes conclusiones:

- Tanto los sistemas *WinCC* y *EcoStructureWare* se caracterizan por poseer licencias privativas además de ser desarrollados con tecnologías distintas a las empleadas por el sistema AREX.
- El estudio de estos sistemas marco la guía para el diseño e implementación de los medidores debido a que el mecanismo de visualización de los componentes de estos sistemas es utilizando SVG y poseen una mayor atomicidad de estos.
- El sistema EROS a pesar de ser un *software* desarrollado en Cuba no es un sistema multiplataforma y presenta deficiencia que impiden el correcto funcionamiento de las tareas.
- En estos sistemas los usuarios pueden interactuar con un mayor número de propiedades logrando la personalización y visualización de los componentes gráficos de forma más amena.

### **1.4 Instrumentos de medición**

Son aquellas herramientas que se utilizan para comparar proporciones o magnitudes con un patrón (unidad de medida) o estándar previamente establecido. Los instrumentos de medición, son los encargados de hacer posible esta comparativa entre el objeto a medir y el patrón, arrojando un número que indica de forma lógica esta relación. La acción que se lleva a cabo es medir y el proceso es la medición (*Instrumentos de Medición* 2019).

#### **1.4.1 Características principales de un instrumento de medida**

- Precisión: es la capacidad de un instrumento de dar el mismo resultado en mediciones diferentes realizadas en las mismas condiciones.
- Exactitud: es la capacidad de un instrumento de medir un valor exacto sin margen de error.
- Resolución: es la mínima variación de la magnitud que es posible medir con el instrumento de medida indicado.
- Apreciación: es la medida más pequeña perceptible en un instrumento de medida.
- Sensibilidad: es la relación de desplazamiento entre el indicador de la medida del instrumento utilizado y la medida real.

#### **1.4.2 Clasificación de los instrumentos de medición**

- Según la función que estos cumplan dentro del sistema:

Se basa en la función que cumple cada uno de los instrumentos dentro del proceso donde se encuentre instalado. Se tienen instrumentos que permiten: tomar, indicar, registrar, transmitir, transformar, recibir y controlar a partir de una señal recogida por un elemento primitivo; y luego accionar mediante un elemento final (Instrumentos de Medición 2019).

- Según el mecanismo de operación del instrumento pueden ser:

Eléctricos, hidráulicos, mecánicos, neumáticos y en algunos casos funcionan combinando dos tipos de energías como los electromecánicos (Instrumentos de Medición 2019).

- Según la variable que se mide con los dispositivos:

Pueden medir temperatura, humedad relativa, tiempo, presión, densidad, ángulos, masa, peso, longitud, capacidad, volumen y velocidad.

#### **1.5 Características de los medidores eléctricos digitales**

Un medidor eléctrico digital involucra, a partir de alguna etapa, un proceso digital, es decir, ante una señal de entrada cuya variación sea continua, proporciona una salida un número finito o discreto de valores. El número que representa el valor de la medida aparece representado por unas cifras visibles directamente en la pantalla (Instrumentos de Medición 2019).

##### **1.5.1 Señal digital**

Utiliza un sistema de códigos binarios (los números 0 y 1) con los que se lleva a cabo la transmisión bajo una pareja de amplitudes que proporciona grandes posibilidades. El proceso del que se habla con las ondas senoidales en las señales analógicas cambia de forma completa para dar paso a ondas cuadradas, lo que

permite hacer uso de la modulación digital y de un tipo de señal que no es continua (VIU Universidad Internacional de Valencia 2018).

Las señales digitales proporcionan una mayor capacidad para transmitir información de una manera fiel. Estas señales no producen deterioro en la información ni en la calidad de los datos, lo que ayuda a que el resultado sea más adecuado.

### 1.5.2 Variable digital

Sólo toma una serie de valores finitos, sus estados están limitados (VIU Universidad Internacional de Valencia 2018).

### 1.5.3 Reloj digital

Indica la hora mediante números, por oposición al reloj analógico que lo hace mediante manecillas. El funcionamiento de los relojes digitales es electrónico (*The Watches Times* 2018).

Las partes por las que está compuesto un reloj ya sea digital o analógico son iguales; la diferencia radica en la forma de mostrar la hora. El reloj electrónico tiene una pantalla de cristal líquido donde se indican las horas mediante dígitos, ver figura 1 Reloj digital (u/soribel 2017).

- Componentes principales de un reloj digital, ver figura 1 Reloj digital:



Figura 1 Reloj digital (*The Watches Times* 2018).

### 1.5.4 Flujoímetro digital

Estos instrumentos de medida suelen colocarse en línea con la tubería que transporta el fluido. También suelen llamarse medidores de caudal o medidores de flujo. El mismo está conformado por una pantalla de cristal líquido donde se indican las medidas tomadas en dígitos; dos cámaras divididas por un diafragma, válvula de alivio y sistema de palancas, ver figura 2 Flujoímetro digital (Suministros hidráulicos del sureste s.a. de C.V. 2017)

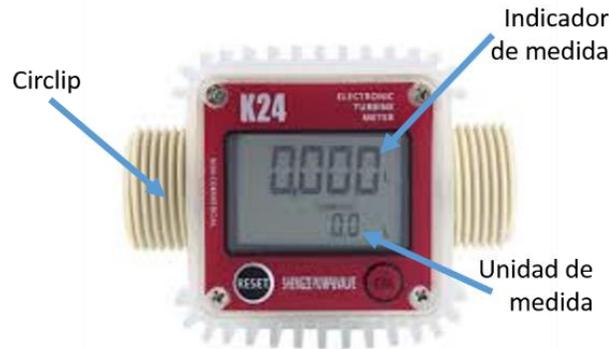


Figura 2 Flujómetro digital (Suministros hidráulicos del sureste S.A. DE C.V. 2017).

### 1.6 Características de los medidores eléctricos analógicos

Es aquel que indica el valor de la variable a medir en forma continua con ayuda de una aguja y se transfiere directamente a una escala de medición haciendo uso solo de las propiedades físicas de los materiales. La magnitud medida se traduce en una señal analógica. Tienen un sistema que traduce la medida (traductor) y un sistema que nos indica la medida (indicador) (Instrumentos de Medición 2019).

#### 1.6.1 Señal analógica

Son señales de tipo continuo, su expansión se produce por la entrada en escena de las ondas de tipo senoidal. Para que las distintas señales analógicas que se transmitan puedan ser interpretadas de una manera adecuada habrá que tener un decodificador que permita cumplir con el proceso de trabajo. Una de las ventajas del uso de la señal analógica es que hay poco consumo de ancho de banda, mientras que por otro lado es un tipo de acción que se procesa en tiempo real (VIU Universidad Internacional de Valencia 2018).

#### 1.6.2 Variable analógica

Una variable analógica es aquella que tiene un margen de variación infinito, esto es que puede tomar infinitos valores (VIU Universidad Internacional de Valencia 2018).

#### 1.6.3 Higrómetro analógico

Es una herramienta que nos permite conocer la humedad relativa que hay en el aire atmosférico. El resultado se expresa como un porcentaje que puede oscilar entre 0-100%. La humedad relativa es la proporción de vapor de agua presente en el aire en comparación con la cantidad de agua que haría falta para saturarlo a una determinada temperatura.

Los higrómetros ayudan a saber si la humedad del hogar, oficina, invernadero o, en general, cualquier entorno es o no adecuada, ver figura 3 Higrómetro analógico (Second Episode S.L 2017).



Figura 3 Higrómetro analógico (Second Episode S.L 2017).

#### 1.6.4 Reloj analógico

Indica la hora en una esfera numerada, mediante manecillas o agujas que indican la hora, los minutos y en algunos relojes los segundos, ver figura 4 Partes de un reloj analógico de pulsera (u/soribel 2017).



Figura 4 Partes de un reloj analógico de pulsera (ZonaCasio.com 2012).

#### 1.6.7 Flujómetro analógico

Cuenta con las mismas características que el flujómetro digital, pero a este instrumento se le incluye el mecanismo de manecillas (agujas) para mostrar las medidas, ver figura 5 Flujómetro analógico (Suministros hidráulicos del sureste s.a. de C.V. 2017).



Figura 5 Flujómetro analógico (Suministros hidráulicos del sureste s.a. de C.V. 2017).

### 1.7 Componente gráfico

Para una mejor comprensión del estudio realizado se define que un **Componente informático**: elemento de un sistema que ofrece un conjunto de servicios, o funcionalidades, a través de interfaces definidas (Graus 2013).

**Gráfico en informática**: es una representación visual de una serie de datos en un ordenador, que tiene como finalidad dar una rápida noción de volúmenes y cantidades estableciendo comparaciones (Lilian Molinero 2012).

Por lo antes expuesto se puede afirmar que un **componente gráfico**: son los elementos básicos con los que se construye la interfaz gráfica de un programa.

### 1.8 Módulo AREX-Edición

La interfaz de este módulo está compuesta por: barra de título, barra de menú, barra de herramientas, explorador de proyecto, explorador de componentes, inspector de propiedades y el área de trabajo para diseñar los despliegues.

Los menús proveen el acceso a las funcionalidades que brinda este módulo agrupadas de la siguiente forma (Centro de Informática Industrial, CEDIN. 2019):

- Menú archivo (*File*) despliega las opciones para abrir, salvar y cerrar el proyecto.
- Menú edición (*Edit*) despliega las opciones para crear los elementos que componen un proyecto, así como editar sus propiedades, copiar, pegar y eliminar. Provee las opciones para editar la posición de

los componentes gráficos, rotarlos, vincularlos con las variables, así como aumentar y disminuir el tamaño.

- Menú vista (*View*) despliega las opciones para mostrar y ocultar las ventanas del sistema.
- Menú ayuda (*Help*) despliega las opciones que permiten visualizar información del sistema y el manual de usuario.

De la misma manera, estos elementos se encuentran distribuidos en la barra de herramientas proporcionando un acceso rápido a estas opciones.

### 1.9 Arquitectura modular

La modularidad es la manifestación más común de la división de problemas. El *software* se divide en componentes con nombres distintos y abordables por separado, en ocasiones llamados módulos, que se integran para satisfacer los requerimientos del problema. Lo novedoso de la arquitectura modular es que se pueden reemplazar o agregar cualquiera de sus componentes (o módulos) sin afectar al resto del sistema. La arquitectura modular es a la vez simple y compleja. Simple porque se puede agregar módulos donde sea necesario y cada módulo encaja en el edificio sin que haya que hacer ajustes sobre el diseño en general. Y es compleja porque cada módulo debe ser posible de cumplir con múltiples funciones, así como una función independiente, mientras que debe mantener los límites estrictos de forma y tamaño del módulo (*Somerville*, I 2011).

### 1.10 Herramientas, tecnologías y metodología.

El desarrollo de *software* no fuera posible sin las herramientas y tecnologías que le dan soporte, pues de no existir estas el trabajo sería engorroso y lento para la obtención de los resultados. El principal resultado obtenido del uso de herramientas tecnológicas es la agilización del trabajo básico, además del aumento de la eficiencia de los mismos.

Las tecnologías y herramientas a utilizar para llevar a cabo la propuesta de solución son las mismas que para el desarrollo del *software* al que el componente fue integrado para facilitar futuros mantenimientos perfectivos al sistema por parte de otros desarrolladores, éstas se describen a continuación:

#### 1.10.1 Gráficos vectoriales escalables (*Scalable Vector Graphics SVG*)

Es un formato de estándar abierto vectorial muy útil para su uso en línea por su flexibilidad y por la capacidad de ofrecer gráficos con calidad. SVG es vectorial, lo que supone tener todas las ventajas de cualquier formato vectorial. Es escalable, ocupa poco espacio en memoria y permite una definición mayor a tamaños reducidos (Paul, M 2007). Fue utilizado para la renderización de los componentes gráficos a desarrollar.

### 1.10.2 Editor gráfico de vectores

*Inkscape* es un *software* gratuito y de código libre, puede crear y editar gráficos vectoriales como ilustraciones, diagramas, líneas, gráficos, logotipos, e ilustraciones complejas. El formato principal que utiliza el programa es SVG. Se encuentra desarrollado principalmente para el sistema operativo GNU/Linux, pero es una herramienta multiplataforma y funciona en *Windows*, *Mac OS X*, y otros sistemas derivados de *Unix* (Tipigao 2013). Esta herramienta se utilizó para el diseño y edición de las plantillas SVG de los componentes gráficos.

### 1.10.3 Marco de trabajo

La herramienta Qt SDK en la versión Qt 5.11.0 es un marco de trabajo (*framework* en inglés) para el desarrollo de aplicaciones multiplataforma; el cual incluye clases, bibliotecas y herramientas para la producción de aplicaciones usando el lenguaje C++ de forma nativa y puede operar en varias plataformas incluyendo los sistemas *Unix* (*Linux*, *MacOS X*, *Solaris*) o incluso toda la familia de *Windows*.

Una de sus funciones más conocidas es la de la creación de interfaces gráficas de usuario, sin embargo, no se limita a esto, ya que también provee clases para facilitar otras tareas de programación como el manejo de *sockets*, soporte para programación multihilo, comunicación con bases de datos, así como el uso de XML, manejo de cadenas de caracteres (*The QT Company* 2019).

Este *framework* fue empleado en este trabajo para la utilización de bibliotecas y herramientas que el mismo facilita.

### 1.10.4 Entorno de desarrollo

Qt *Creator* en su versión 4.6.1 un IDE multiplataforma disponible para *Linux*, *Mac OSX*, *Windows*, *Symbian* y *Maemo*. Es una estrategia tecnológica que permite diseñar, desarrollar, implementar y mantener el *software* de forma rápida y rentable al tiempo que ofrece una experiencia de usuario perfecta en todos los dispositivos. Dicha herramienta se centra en proporcionar características que ayudan a los nuevos usuarios de Qt a aprender y con ello también aumentar la productividad. La herramienta soporta los lenguajes *C#.NET*, *Python*, *Ada*, *Pascal*, *Perl*, *PHP* y *Ruby*; además facilita la administración de proyectos (Programacion en Castellano 2019).

Esta herramienta fue empleada para proporcionar servicios integrales y facilitar al programador el desarrollo de *software* y porque está definida como IDE de desarrollo para el sistema AREX en el centro.

#### **1.10.5 Lenguaje de modelado del framework QT**

QML (*Qt Meta Language* o Lenguaje de Modelado de Qt) es un lenguaje basado en JavaScript, para el diseño de interfaz de usuario de aplicaciones. El mismo se utiliza principalmente para aplicaciones donde la entrada táctil, animaciones fluidas y la experiencia de usuario son cruciales (alfonsogu 2012).

Este lenguaje fue utilizado en este trabajo porque es el que está definido como lenguaje de modelado de QT para el sistema AREX.

#### **1.10.6 Lenguaje de modelado**

Lenguaje de Modelado Unificado (*UML* por sus siglas en inglés), es un estándar para la representación visual de objetos, estados y procesos dentro de un sistema. Puede servir de modelo para un proyecto y garantizar así una arquitectura de información estructurada; por el otro, ayuda a los desarrolladores a presentar la descripción del sistema de una manera que sea comprensible para quienes están fuera del campo. *UML* se utiliza principalmente en el desarrollo de *software* orientado a objetos (IONOS España S.L.U. 2019).

Este lenguaje fue empleado para el diseño de los diagramas necesarios para el desarrollo de *software*.

#### **1.10.7 Herramienta CASE**

*Visual Paradigm* es una herramienta para la Ingeniería de *Software* Asistida por Computadora que soporta el ciclo de vida completo de desarrollo de *software*: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. Facilita la representación de diagramas y establecer coherencia entre ellos, además de generar el código partiendo de los diagramas y viceversa, otra de las ventajas que presenta la herramienta es la integración con otras aplicaciones y el trabajo multiusuario (Patricia López 2017).

En este trabajo se utilizó la herramienta en su versión 8.0 para el diseño de los artefactos y establecer las relaciones entre ellos.

#### **1.10.8 Metodología del desarrollo de *software***

Las metodologías se basan en una combinación de los modelos de proceso genéricos (cascada, evolutivo, incremental, espiral). Una metodología de desarrollo de *software* tiene como principal objetivo aumentar la calidad del *software* que se produce en todas y cada una de sus fases de desarrollo. No existe una metodología de *software* universal, debido a que toda metodología debe ser adaptada a las características de cada proyecto (equipo de desarrollo, recursos) exigiéndose así que el proceso sea configurable.

Las metodologías de desarrollo se pueden dividir en dos grupos de acuerdo con sus características y los objetivos que persiguen: ágiles y robustas (*Pressman*, R. 2012).

En el desarrollo de la solución propuesta se empleó AUP (Proceso Ágil Unificado) en su versión UCI, se caracteriza por ser una metodología ágil y está definida por la universidad como el documento rector de la actividad productiva. La misma propone tres fases (inicio, ejecución y cierre) las cuales contienen las características de las cuatro fases de AUP:

**Inicio:** tiene como objetivo obtener una comprensión común cliente-equipo de desarrollo del alcance del nuevo sistema y definir una o varias arquitecturas candidatas para el mismo.

**Elaboración:** El objetivo es que el equipo de desarrollo profundice en la comprensión de los requisitos del sistema y en validar la arquitectura.

**Construcción:** Durante la fase de construcción el sistema es desarrollado y probado al completo en el ambiente de desarrollo.

**Transición:** El sistema se lleva a los entornos de preproducción donde se somete a pruebas de validación y aceptación y finalmente se despliega en los sistemas de producción.

AUP-UCI propone 7 disciplinas: Modelado de negocio, requisitos, análisis y diseño, pruebas internas, pruebas internas, pruebas de liberación.

La metodología AUP- UCI propone cuatro escenarios (*Somerville, I 2011*):

**Escenario No 1:** Aplica a los proyectos que hayan evaluado el negocio a informatizar y como resultado obtengan que puedan modelar una serie de interacciones entre los trabajadores del negocio/actores del sistema (usuario), similar a una llamada y respuesta respectivamente, donde la atención se centra en cómo el usuario va a utilizar el sistema. Es necesario que se tenga claro por el proyecto que los CUN muestran como los procesos son llevados a cabo por personas y los activos de la organización.

**Escenario No 2:** Aplica a los proyectos que hayan evaluado el negocio a informatizar y como resultado obtengan que no es necesario incluir las responsabilidades de las personas que ejecutan las actividades, de esta forma modelarían exclusivamente los conceptos fundamentales del negocio. Se recomienda este escenario para proyectos donde el objetivo primario es la gestión y presentación de información.

**Escenario No 3:** 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.

**Escenario No 4:** Aplica a los proyectos que hayan evaluado el negocio a informatizar y como resultado obtengan un negocio muy bien definido. El cliente estará siempre acompañando al equipo de desarrollo para convenir los detalles de los requisitos y así poder implementarlos, probarlos y validarlos. Se recomienda en proyectos no muy extensos, puesto que una historia de usuario (HU) no debe poseer demasiada información.

Luego del estudio de los cuatro escenarios anteriores y los artefactos que cada uno de ellos genera por las disciplinas de AUP se llegó a la conclusión que se utilizara el escenario 4 Historia de usuario.

### **Conclusiones parciales**

- Con el estudio del estado del arte sobre los sistemas de medición se logró definir las características y clasificaciones principales de los estos. Para lograr una mayor comprensión de los medidores fueron definidos los principales conceptos asociados a la investigación.
- La utilización de la norma ISA para la representación de los medidores permitió la uniformidad con todos los componentes existentes en el sistema AREX.
- Después de analizar la metodología de desarrollo, herramientas y arquitectura que utiliza AREX se determinó utilizarlas en el desarrollo de la propuesta de solución en aras de facilitar futuros mantenimientos perfectivos al sistema por parte de otros desarrolladores.

Capítulo II: Características y diseño del sistema

**Introducción**

En el presente capítulo se reflejan las actividades realizadas en los procesos de análisis y diseño de la solución propuesta. Se exponen los artefactos más importantes que describen el flujo normal de eventos que ocurren en la solución propuesta, realizándose la descripción de la misma, planteándose los requisitos tanto funcionales como no funcionales, las historias de usuarios, la arquitectura y los patrones de diseño empleados.

**2.1 Modelo de dominio**

En este diagrama se describen las entidades que intervienen en el diseño, con el objetivo de facilitar la comprensión de los principales conceptos que se utilizarán en el proceso identificado.

Se utiliza con frecuencia como fuente de inspiración para el diseño de los objetos de *software*. Este diagrama se elabora utilizando la notación *UML* y así representar un conjunto de diagramas de clases en los que no se define ninguna operación (Pressman, R. 2012), ver figura 6 Modelo de dominio.

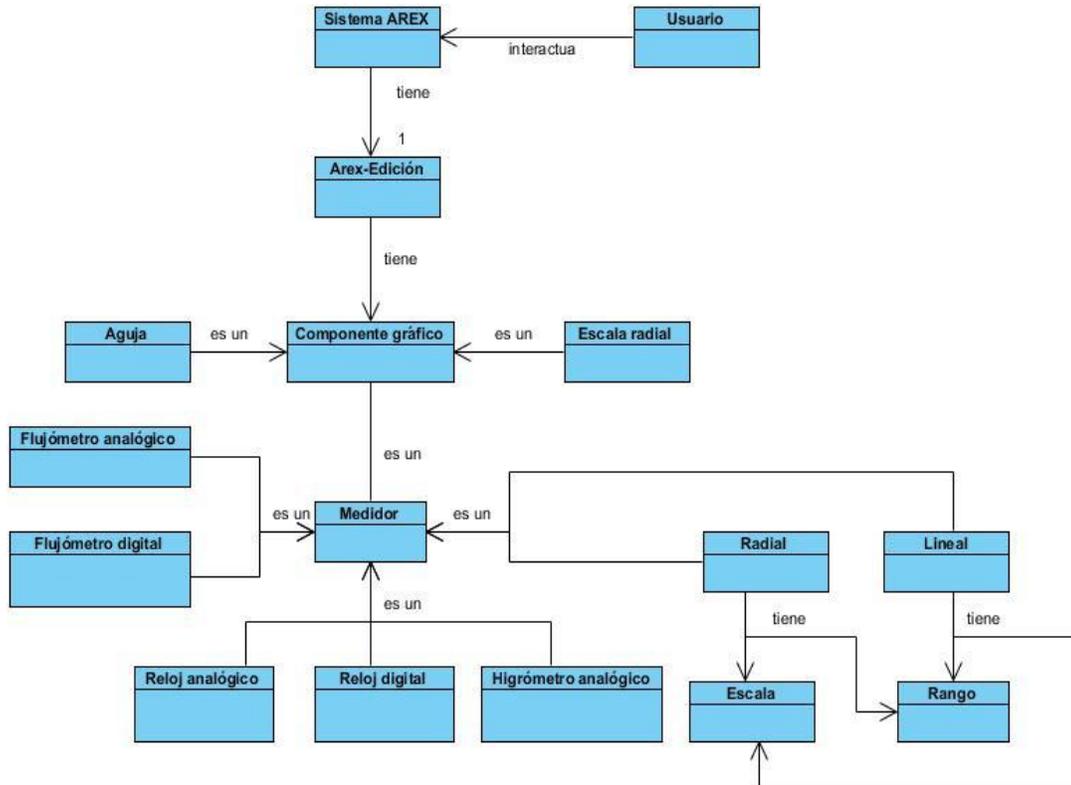


Figura 6 Modelo de dominio (Elaboración propia).

**Descripción del modelo de dominio**

Se describe el flujo del sistema a través de los diferentes módulos y principales entidades del dominio.

Sistema AREX: Sistema de medición y adquisición de datos.

AREX-Edición : módulo donde se visualizan las paletas de componentes gráficos disponibles.

Componente gráfico: representación abstracta de algún dispositivo del proceso que se está supervisando.

Medidor: especialización de un medidor.

**2.2 Requisitos**

Un requisito es una “condición o capacidad que necesita el usuario para resolver un problema o conseguir un objetivo determinado”. También se aplica a las condiciones que debe cumplir o poseer un sistema o uno de sus componentes para satisfacer un contrato, una norma o una especificación (Pressman, R. 2012).

**2.2.1 Requisitos funcionales**

Los requisitos del sistema son los elementos más importantes en el desarrollo de *software*, es así que un mal levantamiento de los mismos podría estropear meses de trabajo (Pressman, R. 2012). Los requisitos funcionales a desarrollar en la investigación se visualizan en la siguiente tabla, ver tabla 2 Requisitos funcionales.

Tabla 2 Requisitos funcionales.

No.	Nombre	Descripción	Prioridad	Complejidad
<b>Requisitos Funcionales de los Componentes Gráficos</b>				
RF1	Crear instancia del componente gráfico.	El sistema debe proporcionar que el usuario cree el componente gráfico seleccionado mediante el clic en la paleta del componente.	Alta	Baja
RF2	Definir valor mínimo de la escala del medidor.	El sistema debe proporcionar que el usuario defina el valor mínimo de la escala del medidor.	Alta	Baja
RF3	Definir valor máximo de la escala del medidor.	El sistema debe proporcionar que el usuario defina el valor máximo de la escala del medidor.	Alta	Baja

## *Capítulo II: Características y diseño del sistema*

RF4	Modificar valor mínimo de la escala del medidor.	El sistema debe proporcionar que el usuario modifique la propiedad “valor mínimo” del componente gráfico mediante el inspector de propiedades.	Baja	Baja
RF5	Modificar valor máximo de la escala del medidor.	El sistema debe proporcionar que el usuario modifique la propiedad “valor máximo” del componente gráfico mediante el inspector de propiedades.	Baja	Baja
RF6	Modificar valor mínimo del rango seleccionado en el medidor.	El sistema debe proporcionar que el usuario modifique la propiedad “rangos” del componente gráfico mediante el inspector de propiedades.	Alta	Baja
RF7	Modificar valor máximo del rango seleccionado en el medidor.	El sistema debe proporcionar que el usuario modifique la propiedad “rangos” del componente gráfico mediante el inspector de propiedades.	Baja	Baja
RF8	Modificar color de la escala del medidor.	El sistema debe proporcionar que el usuario modifique la propiedad “color de escala” del componente gráfico mediante el inspector de propiedades.	Baja	Baja
RF9	Modificar color de fondo del medidor.	El sistema debe proporcionar que el usuario modifique la propiedad “color de fondo” del componente gráfico mediante el inspector de propiedades.	Baja	Baja
RF10	Definir color de fondo de la aguja.	El sistema debe proporcionar que el usuario defina el color de fondo de la aguja.	Baja	Baja
RF11	Definir unidad de medida (texto) del medidor.	El sistema debe proporcionar que el usuario defina la unidad de medida del medidor.	Baja	Baja

## *Capítulo II: Características y diseño del sistema*

RF12	Modificar valor del medidor.	El sistema debe proporcionar que el usuario modifique la propiedad “valor” del componente gráfico mediante el inspector de propiedades.	Baja	Baja
RF13	Definir la amplitud de la escala radial.	El sistema debe proporcionar que el usuario defina la amplitud de la escala radial.	Alta	Alta
RF14	Definir ángulo inicial de la escala radial.	El sistema debe proporcionar que el usuario defina el ángulo inicial de la escala radial.	Alta	Alta
RF15	Modificar color del rango seleccionado en el medidor.	El sistema debe proporcionar que el usuario modifique la propiedad “rangos” del componente gráfico mediante el inspector de propiedades.	Baja	Baja
RF16	Configurar opacidad del componente gráfico.	El sistema debe proporcionar que el usuario modifique la propiedad “opacidad” del componente gráfico mediante el inspector de propiedades.	Baja	Baja
RF17	Adicionar rango de la escala en el medidor.	El sistema debe proporcionar que el usuario adicione un nuevo rango en la propiedad “rangos” del componente gráfico mediante el inspector de propiedades.	Alta	Media
RF18	Mostrar rango de la escala en el medidor.	El sistema debe proporcionar mostrar rango de la escala en el medidor.	Baja	Baja
RF19	Eliminar rango de la escala en el medidor.	El sistema debe proporcionar que el usuario elimine el rango deseado en la propiedad “rangos” del componente gráfico mediante el inspector de propiedades.	Media	Baja

### 2.2.2 Requisitos no funcionales

Los requisitos no funcionales no son más que restricciones que afectan las funciones del sistema, que pueden ser restricciones de estándares, interfaz, usabilidad, entre otros (Pressman, R. 2012).

#### **RNF1. Software:**

RNF1.1 Se deben poseer dispositivos de tipo interruptor automático de *SCHNAIDER Electric (MasterPac o Compac NSX)* o en su lugar una aplicación que permita realizar simulaciones (*Modsim32*).

#### **RNF2. Usabilidad:**

RNF2.1 El sistema debe contar con menús que agrupen las funcionalidades que están relacionadas.

#### **RNF3. Apariencia o interfaz:**

RNF3.1 Diseño sencillo, permitiendo la utilización del sistema sin mucho entrenamiento.

RNF3.2 El producto final debe tener una interfaz fácil de usar y amigable.

#### **RNF4. Hardware:**

RNF4.1 Microprocesador Intel Core 2 a 1.5 GHz, memoria RAM de 1 GB espacio en disco duro 80 GB, tarjeta de red cableada y/o puerto serie.

### 2.3 Historias de usuarios

El método empleado para describir el funcionamiento de la aplicación son las historias de usuario (HU). Estas son el equivalente a los casos de uso empleados en otras metodologías. Deben poder ser programadas en un tiempo entre una y tres semanas. Si la estimación es superior a tres semanas, debe ser dividida en dos o más historias. Si es menos de una semana, se debe combinar con otra historia (Rodríguez, T 2014).

Algunas HU no tienen un alto grado de complejidad para implementarlas, por lo cual se determinará que su período de desarrollo sea de 1, 3 o 4 días en un equivalente de 0.2, 0.6 y 0.8 en semana respectivamente, para así calcular el tiempo estimado del proyecto. El cálculo del tiempo se realizó teniendo en cuenta que una semana de trabajo consta de 5 días donde se trabaja 8 horas diarias y no se cuentan los días feriados. Están representadas mediante tablas con los siguientes elementos:

- **Número:** Número de la HU, incremental.
- **Nombre:** Nombre que identifica la HU.
- **Referencia:** Es el conjunto de HU de las cuales depende la actual.
- **Prioridad:** Esta característica es dada por el cliente con los valores: alta, media o baja en dependencia de la importancia y orden en que desean que sean implementadas.

- **Iteración Asignada:** Número de la iteración en la cual se desarrollará la HU.
- **Puntos Estimados:** Tiempo estimado en semanas que se le asignará.
- **Descripción:** Breve descripción del proceso que define la historia.
- **Observaciones:** Alguna acotación importante a señalar sobre la historia.
- **Prototipo de interfaz:** Prototipo de interfaz, si aplica.

Para asignar los valores de prioridad se tiene en cuenta lo siguiente:

- **Alta:** Se le otorga a las HU que resultan funcionalidades fundamentales en el desarrollo del sistema, las que el cliente define como principales para el control del sistema.
- **Media:** Se le otorga a las HU que resultan para el cliente como funcionalidades a tener en cuenta, sin que estas tengan una afectación sobre el sistema que se esté desarrollando.
- **Baja:** Se le otorga a las HU que constituyen funcionalidades que sirven de ayuda al control de elementos asociados al equipo de desarrollo, a la estructura, y no tienen que ver directamente con el sistema en desarrollo.

Se redactaron 19 HU, de ellas 7 se clasificaron con prioridad alta, 1 con prioridad media y 11 con prioridad baja. A continuación, aparecen representadas 3 de las HU especificadas (las demás aparecen en el Anexo 1 Historias de usuarios), ver tabla 3 Historia de usuario#1 y tabla 4 Historia de usuario#2.

HU01

Tabla 3 Historia de usuario#1

Historia de usuario	
<b>Número:</b> 1	<b>Nombre de requisito:</b> Crear instancia del componente gráfico.
<b>Programador:</b> Arlette Lorenzo Tovar	<b>Iteración Asignada:</b> 1
<b>Prioridad:</b> Alta	<b>Tiempo Estimado:</b> 1
<b>Riesgo en Desarrollo:</b> Medio	<b>Tiempo Real:</b> 1
<b>Descripción:</b> El sistema debe proporcionar que el usuario cree el componente gráfico seleccionado mediante el clic en la paleta del componente.	

HU02

Tabla 4 Historia de usuario#2

Historia de usuario	
<b>Número:</b> 1	<b>Nombre del requisito:</b> Definir valor mínimo de la escala del medidor.
<b>Programador:</b> Arlette Lorenzo Tovar	<b>Iteración Asignada:</b> 1
<b>Prioridad:</b> Alta	<b>Tiempo Estimado:</b> 0.8
<b>Riesgo en Desarrollo:</b> Alto	<b>Tiempo Real:</b> 0.8
<b>Descripción:</b> El sistema debe proporcionar que el usuario defina el valor mínimo de la escala del medidor.	

HU03

Tabla 5 Historia de usuario#3

Historia de usuario	
<b>Número:</b> 2	<b>Nombre del requisito:</b> Definir valor máximo de la escala del medidor.
<b>Programador:</b> Arlette Lorenzo Tovar	<b>Iteración Asignada:</b> 1
<b>Prioridad:</b> Alta	<b>Tiempo Estimado:</b> 0.8
<b>Riesgo en Desarrollo:</b> Medio	<b>Tiempo Real:</b> 0.8
<b>Descripción:</b> El sistema debe proporcionar que el usuario defina el valor máximo de la escala del medidor.	

El desarrollo de las historias de usuarios por cada uno de los requisitos funcionales permite comprender mejor las respuestas que dará el sistema.

#### 2.4 Fase de planificación

En la fase de planificación se establece prioridad a las HU más importantes. Los desarrolladores estiman el esfuerzo necesario para cada HU y a partir de esto se define el plan de desarrollo de las iteraciones.

#### 2.5 Estimación de esfuerzos e iteraciones por historia de usuario.

La estimación del esfuerzo necesario para el desarrollo de la aplicación mediante las HU consiste en la suma del tiempo establecido en cada una de las iteraciones donde se distribuyeron esas HU. La siguiente tabla muestra las HU con los puntos de estimación en semanas, ver tabla 6 Estimación de esfuerzos e iteraciones por historia de usuario.

Tabla 6 Estimación de esfuerzos e iteraciones por historia de usuario.

No.	Historias de usuario	Tiempos de estimación	Iteración	Duración total de las iteraciones(semanas)
HU01	Crear instancia del componente gráfico.	1	1	6
HU02	Definir valor mínimo de la escala del medidor.	0.8		
HU03	Definir valor máximo de la escala del medidor.	0.8		
HU04	Modificar valor mínimo de la escala del medidor.	0.8		
HU05	Modificar valor máximo de la escala del medidor.	0.8		
HU06	Modificar valor mínimo del rango seleccionado en el medidor.	1		
HU07	Modificar valor máximo del rango seleccionado en el medidor.	0.8		
HU08	Modificar color de la escala del medidor.	0.8	2	6
HU09	Modificar color de fondo del medidor.	0.8		
HU10	Definir color de fondo de la aguja.	0.8		
HU11	Definir unidad de medida (texto) del medidor.	0.8		
HU12	Modificar valor del medidor.	0.8		
HU13	Definir la amplitud de la escala radial.	0.8		

<b>HU14</b>	Definir ángulo inicial y final de la escala radial.	0.8		
<b>HU15</b>	Modificar color del rango seleccionado en el medidor.	0.8	3	5
<b>HU16</b>	Configurar opacidad del componente gráfico.	0.8		
<b>HU17</b>	Adicionar rango de la escala en el medidor.	1		
<b>HU18</b>	Mostrar rango de la escala en el medidor.	0.8		
<b>HU19</b>	Eliminar rango de la escala en el medidor.	0.8		

Al realizar la estimación de esfuerzos e iteraciones por historia de usuario el tiempo de desarrollo obtenido fue de 17 semanas aproximadamente siendo equivalente a 4 meses y 1 semana.

### 2.6 Plan de iteraciones

Luego de estimar el tiempo de desarrollo de las HU se procede a realizar el plan de iteraciones. Se seleccionan las HU a desarrollar en cada una de las entregas pactadas. Por cada iteración se definen las pruebas que se realizarán para verificar si se cumplió con lo especificado en la HU correspondiente. A las pruebas fallidas se les realiza un análisis y se corrigen las dificultades para la próxima iteración.

**Iteración I:** tiene como objetivo realizar las HU 1,2,3,4,5,6,7 las cuales se encargan de configurar los componentes gráficos.

**Iteración II:** se encarga de realizar las HU 8,9,10,11,12,13,14 configurar valores de los medidores.

**Iteración III:** tiene como objetivo realizar las HU 15,16,17,18,19 las cuales se encargan de definir y modificar información de la escala, el rango y colores de los medidores.

### 2.7 Plan de entregas

El plan de entregas se realiza para definir las fechas en que se deberán desplegar las versiones de la aplicación que resultan de cada iteración durante el proceso de desarrollo. Su planificación se basa en la estimación realizada de los tiempos de cada una de las iteraciones. Según el cálculo de tiempo de las iteraciones se confeccionó el siguiente plan de entregas, en el cual se tiene como fecha definitiva el 15 de mayo de 2019, ver tabla 7 Plan de entrega.

Tabla 7 Plan de entrega

Iteración	Fecha de inicio	Fecha de fin
1era Iteración	14 de enero de 2018	15 de febrero de 2019
2da Iteración	16 de febrero de 2019	29 de marzo de 2019
3era Iteración	30 de marzo de 2019	10 de mayo de 2019

### 2.8 Diagrama de paquetes

Se usan para reflejar la organización de paquetes y sus elementos. Cuando se usan para representaciones, los diagramas de paquete de los elementos de clase se usan para proveer una visualización de los espacios de nombres. Los usos más comunes para los diagramas de paquete son para organizar diagramas de casos de uso y diagramas de clase, a pesar de que el uso de los diagramas de paquete no es limitado a estos elementos UML (Cardozzo, D. R. 2014).

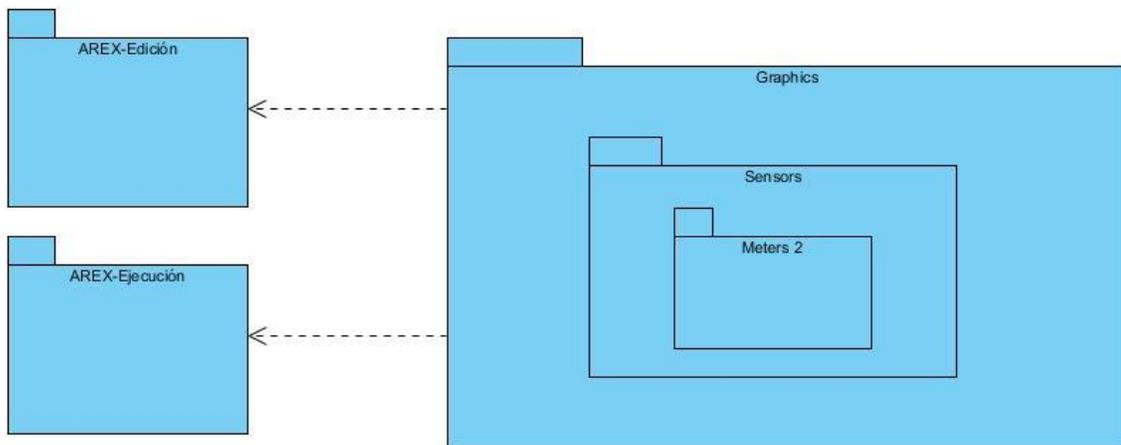


Figura 7 Diagrama de paquetes (Elaboración propia).

El diagrama representa los paquetes AREX-Edición y AREX-Ejecución que son los módulos que consumen del paquete *Graphics* el cual contiene los componentes gráficos entre los que se encuentra el paquete *Sensors* y dentro de este el paquete *Meters 2* el cual contiene medidores analógicos y digitales.

### 2.9 Patrones de arquitectura

Son patrones de diseño de *software* que ofrecen soluciones a problemas de arquitectura de *software* en ingeniería de *software*. Dan una descripción de los elementos y el tipo de relación que tienen junto con un conjunto de restricciones sobre cómo pueden ser usados. Un patrón arquitectónico expresa un esquema de organización estructural esencial para un sistema de *software*, que consta de subsistemas, sus

responsabilidades e interrelaciones. En comparación con los patrones de diseño, los patrones arquitectónicos tienen un nivel de abstracción mayor (arletparedes 2012).

El sistema AREX está desarrollado con una arquitectura modular, la misma se refiere a cualquier diseño compuesto por componentes separados que conectados o unidos forman una unidad habitable.

El módulo AREX-Edición presenta el patrón arquitectónico Modelo Vista Controlador (*MVC*, por sus siglas en inglés). La solución propuesta se enfoca en la Vista del patrón, el cual se evidencia en la representación de los componentes gráficos que representan medidores agrupados en la paleta de componentes nombrada "Meter 2" situada en el visualizador de paletas de componentes del AREX-Edición.

### 2.10 Patrones de diseño

Los patrones de diseño son el esqueleto de las soluciones a problemas comunes en el desarrollo de *software*. En otras palabras, brindan una solución ya probada y documentada a problemas de desarrollo de *software* que están sujetos a contextos similares. Se debe tener presente los siguientes elementos de un patrón: su nombre, el problema (cuando aplicar un patrón), la solución (descripción abstracta del problema) y las consecuencias (costos y beneficios) (Somerville, I 2011).

### 2.11 Los patrones GRASP utilizados fueron:

Los patrones GRASP (*General Responsibility Assignment Software Patterns*, por sus siglas en inglés) o Patrones Generales de *Software* para Asignación de Responsabilidades, indican cual es la manera de asignar responsabilidades a objetos *software* (Somerville, I 2011). Estos patrones describen los principios fundamentales de la asignación de responsabilidades a objetos, expresados en forma de patrones.

- **Creador:** El patrón Creador guía la asignación de responsabilidades relacionadas con la creación de objetos, tarea muy frecuente en los sistemas orientados a objetos. El propósito fundamental de este patrón es encontrar un creador que se debe conectar con el objeto producido en cualquier evento. Al escogerlo como creador, se da soporte al bajo acoplamiento. Este patrón está evidenciado en la entidad *GraphicPack* la cual se encarga de crear todos los componentes definidos como *SVGItem* dentro de la paleta.
- **Experto:** Consiste en la asignación de responsabilidades a la clase que contiene la información necesaria para el cumplimiento de las mismas. Se puede aprovechar la oportunidad de reutilizar componentes en futuras aplicaciones. Este patrón está evidenciado en la clase *SvgItem*.

- **Bajo Acoplamiento:** dentro de un sistema indica que los modulo no conocen o conocen muy poco del funcionamiento interno de otros módulos, evitando la fuerte dependencia entre ellos. Este patrón se evidencia en las clases que modelan los componentes (SVGItem), los cuales carecen de total relación.
- **Alta cohesión:** Mantiene la asignación de responsabilidades de manera que la información almacenada en un *Ítem* sea coherente. El uso de este patrón se ve reflejado en cada componente, ya que este almacena la información del mismo de manera coherente.

### Conclusiones parciales

- El diseño del modelo de dominio permitió obtener la representación de los principales conceptos utilizados en el desarrollo de la propuesta de solución con sus respectivas relaciones.
- El levantamiento de requisitos permitió identificar las funcionalidades que se deben implementar para dar solución al problema y objetivo planteado.
- El levantamiento de requisitos no funcionales facilitó definir los requerimientos de *hardware*, *software*, apariencia y usabilidad que debe cumplir la paleta de componentes.
- La realización de los artefactos ingenieriles generados dieron paso a la construcción de la propuesta de solución logrando un mejor entendimiento del sistema.

### Capítulo III: Implementación y pruebas del sistema

#### Introducción

En el presente capítulo se exponen los elementos relacionados con la construcción de los componentes y su estructura. Se describen también las diferentes pruebas aplicadas y el resultado de estas. Una vez corregidos los errores los componentes desarrollados podrán ser utilizados en el sistema AREX.

Para la metodología AUP-UCI el flujo de trabajo implantación tiene como objetivo transformar su modelo en código ejecutable y realizar un nivel básico de las pruebas, en particular, la unidad de pruebas; así como el flujo de trabajo de prueba tiene como objetivo realizar una evaluación objetiva para garantizar la calidad. Esto incluye la búsqueda de defectos, validar que el sistema funciona tal como está establecido, verificando que se cumplan los requerimientos (Somerville, I 2011).

#### 3.1 Modelo de implementación

Conjunto de componentes y subsistemas que constituyen la composición física de la implementación del sistema. Entre los componentes podemos encontrar datos, archivos, ejecutables, código fuente y los directorios. Fundamentalmente, se describe la relación que existe desde los paquetes y clases del modelo de diseño a subsistemas y componentes físicos. Un diagrama de implementación muestra (Somerville, I 2011):

- Las dependencias entre las partes de código del sistema (diagramas de componentes).
- La estructura del sistema en ejecución (diagrama de despliegue).

##### 3.1.1 Diagrama de componentes

Describe los elementos físicos de un sistema y sus relaciones. Muestran las opciones de realización incluyendo código fuente, binario y ejecutable (IONOS España S.L.U. 2019).

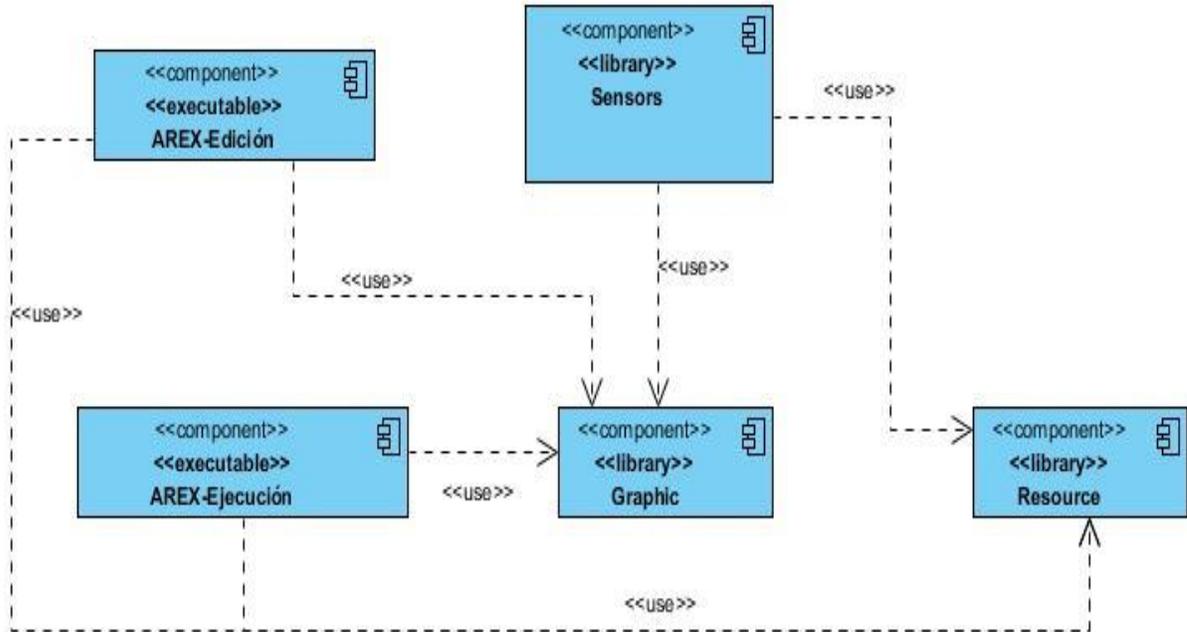


Figura 8 Diagrama de componentes (Elaboración propia).

#### Descripción y estructura del diagrama de componentes anterior:

- AREX-Edición: módulo donde se realizan las configuraciones de los procesos que se desean supervisar haciendo uso de los componentes gráficos para simular los procesos.
- AREX-Ejecución: módulo donde se realiza el control y la supervisión de los procesos configurados en el módulo AREX-Edición.
- Biblioteca *Graphics*: librería que almacena los objetos gráficos que son utilizados para representar los componentes gráficos, ya sea de forma simple o componentes complejos.
- Biblioteca *Sensors*: Se encuentran almacenados los objetos gráficos incluyendo los medidores analógicos y digitales.
- *Resource*: Biblioteca donde se encuentran agrupadas cada una de las entidades encargadas de brindar las imágenes, los iconos y los diseños de los componentes en SVG.

#### 3.1.2 Diagrama de despliegue

Se utiliza para modelar el *hardware* utilizado en las implementaciones de sistemas y las relaciones entre sus componentes. El mismo permite modelar la disposición física o topología de un sistema; muestra el *hardware* usado y los componentes instalados en el *hardware* y también las conexiones físicas entre el *hardware* y las relaciones entre componentes (IONOS España S.L.U. 2019).

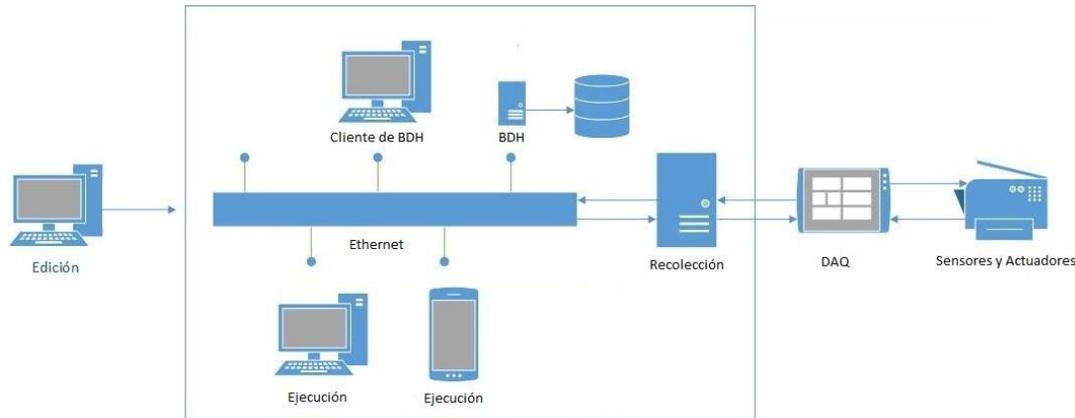


Figura 9 Diagrama de despliegue de AREX (Centro de Informática Industrial, CEDIN. 2019).

El diagrama de despliegue anterior es el correspondiente al sistema AREX; el cual relaciona los principales elementos que conforman dicho sistema:

El AREX-Edición es el encargado de la configuración de todos los componentes, generando un archivo global de configuración en formato XML, dentro de este se incluirá la nueva biblioteca de medidores analógicos y digitales desarrollados. El AREX-Ejecución (Ejecución), a partir de la configuración recibida y en comunicación permanente con el servidor de AREX-Recolección, permite a cualquier usuario tener acceso de forma gráfica a la información del proceso que se monitorea, y ejercer control manual sobre él. Este módulo, a partir de la configuración recibida, permite la comunicación directa con los dispositivos de adquisición de datos (dispositivos DAQ) y brinda dicha información a quien la solicite.

El AREX-Históricos (Servidor de BDH), a partir de la configuración recibida y en comunicación permanente con el AREX-Recolección, permite almacenar puntos de variables y alarmas, para inmediata o posterior consulta. El HMI Cliente de BDH, en comunicación directa con el AREX-Históricos, permite generar gráficas en tiempo real y la consulta de los datos históricos almacenados.

### 3.2 Estándar de codificación

Los estándares de codificación, también llamados estilos de programación o convenciones de código, son convenios para escribir código fuente en ciertos lenguajes de programación. Estos estándares facilitan el mantenimiento del código, sirven como punto de referencia para los programadores, mantienen un estilo de programación y ayudan a mejorar el proceso de codificación (Manuel Arias Calleja 2017).

Como la solución propuesta en este trabajo es parte del sistema AREX el estándar de codificación utilizado fue definido por el proyecto y se utilizará la codificación nomenclatura camello (*Lower Camel Case en inglés*):

- El código está escrito en inglés y la documentación en español.
- Las variables y funciones comienzan con letra minúscula. Cada palabra consecutiva en el nombre comienza con letra mayúscula.

Ejemplo: `drawScale()`;

- Los atributos de las clases deben empezar con `m_` seguido del nombre del atributo, en el caso de atributos compuestos, la inicial de la segunda palabra debe comenzar con mayúscula.

Ejemplo: `m_minimunScaleValue`;

- Los valores de los numerativos deben ser con letras mayúsculas.

Ejemplo: `polarToCartesian(xPos, yPos, radius, angleInDegrees)`;

Ejemplo de una función desarrollada durante la investigación `drawRanges()` encargada de la representación y cálculo de los rangos, ver figura 11 Ejemplo de una función desarrollada.

```
function drawRanges(){
  var tempRangesPath="";
  for(var i = 0 ; i < _ranges.length ; ++i) {
    var range = _ranges[i];
    if(range.min <= m_maximunScaleValue &&
       range.max >= m_minimunScaleValue) {
      var minValueCalculated = (range.min >= m_minimunScaleValue)?
        range.min : m_minimunScaleValue;
      var maxValueCalculated = (range.max <= m_maximunScaleValue)?
        range.max : m_maximunScaleValue;
      var minAngle = angleByValue(minValueCalculated,
        m_minimunScaleValue,
        m_maximunScaleValue,
        m_startAngleScale,
        m_amplitudeScale);
      var maxAngle = angleByValue(maxValueCalculated,
        m_minimunScaleValue,
        m_maximunScaleValue,
        m_startAngleScale,
        m_amplitudeScale);
      var rangeArc = drawArc(250, 250,m_radiusScale, minAngle,maxAngle );
      var path = '<path class="_range" d="d" stroke="_stroke" fill="none" stroke-width="_strokeWidth" />';
      path = path.replace('_d',rangeArc);
      path = path.replace('_stroke',range.color);
      path = path.replace('_strokeWidth',m_widthScale);
      tempRangesPath = tempRangesPath + path;
    }
  }
  _pathRanges = tempRangesPath;
}
```

Figura 10 Ejemplo de una función desarrollada (Elaboración propia).

### **3.3 Solución del problema**

En este epígrafe se muestra la interfaz del sistema AREX donde se encuentra la clasificación de medidores analógicos y digitales (*Meters 2*) dentro de la paleta de componentes en el módulo AREX-Edición y AREX-Ejecución. Durante el análisis al sistema AREX fueron detectadas algunas deficiencias a las cuales se les dio respuesta:

- Los nuevos componentes gráficos realizados en la investigación emplean el formato SVG para su representación; por lo que tienen una mejor calidad de visualización al realizarse el redimensionamiento.
- Fueron desarrollados componentes gráficos que representen medidores que permiten la representación de variables de medición como: humedad relativa, flujo de un gas o un líquido.
- Los componentes gráficos que componen un medidor pueden ser desacoplados, lo cual no constituye una afectación para la modificación de éstos individualmente, logrando la atomicidad de los componentes.
- La gestión dinámica de escalas y rangos, no limita la personalización mediante las propiedades definidas, lo que permite al sistema ajustarse a las necesidades del área a representar en el despliegue.

En la siguiente imagen se muestra los 9 componentes gráficos para la representación de medidores analógicos y digitales; cada uno de ellos con funcionalidades individuales. De esta manera se da cumplimiento al problema planteado sobre como proveer una mejor calidad de visualización, personalización y atomicidad de los medidores pertenecientes al sistema AREX, ver figura 11 Interfaz del AREX-Edición del sistema AREX.

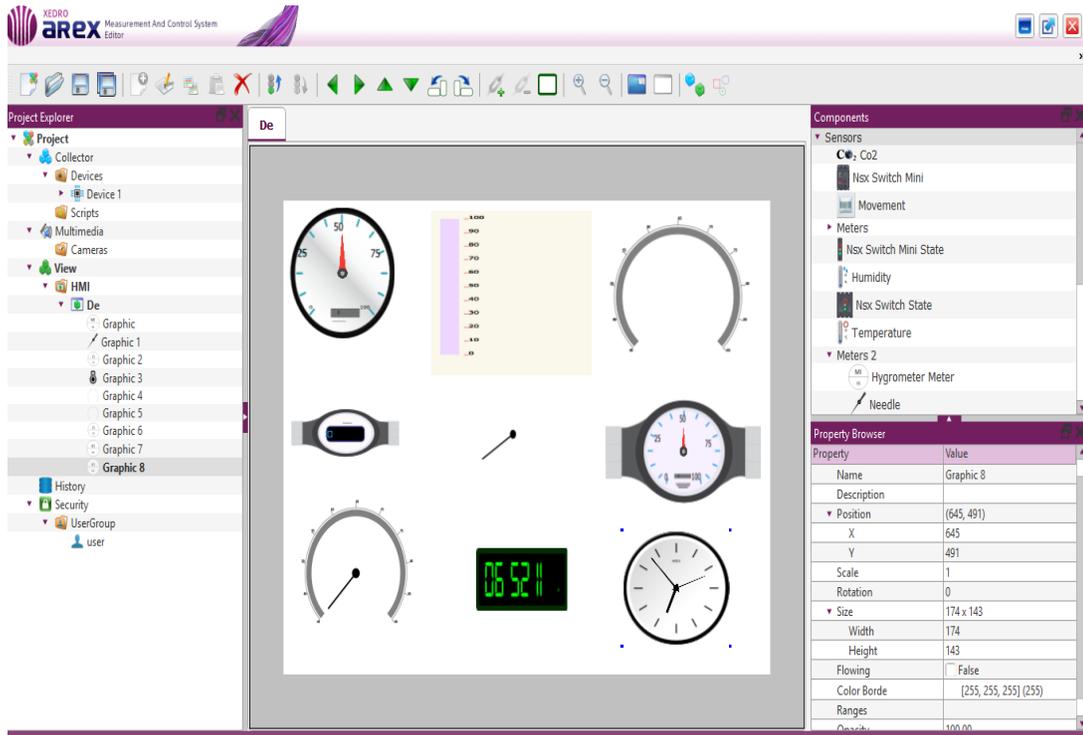


Figura 11 Interfaz del módulo AREX-Edición.

A continuación, se muestra una imagen con la paleta de componentes gráficos obtenida y las propiedades que pueden ser modificadas por cada uno de los componentes, ver figura 12 Paleta de componentes *Meters 2* y la figura 13 Propiedades de los componentes gráficos.

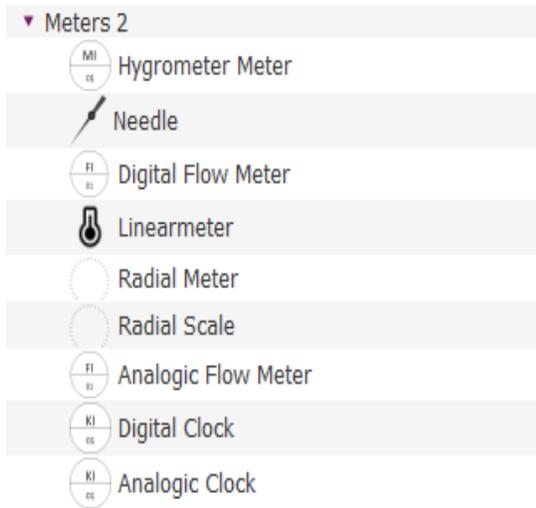


Figura 12 Paleta de componentes *Meters 2*.

Property	Value
Name	Graphic
Description	
Position	(403, 294)
X	403
Y	294
Scale	0.75
Rotation	0
Size	600 x 600
Width	600
Height	600
Min Value	0.00
Max Value	100.00
Valor	0.00
Tick Count	10
Ranges	
Font Size	10
Color Default Level	[236, 213, 255] (255)
Color Background	[250, 247, 235] (255)
Color Level Indicator	[255, 0, 0] (255)
Ranges Visibility	visible
Tick Visibility	visible
Opacity	100.00

Figura 13 Propiedades de los componentes gráficos.

Las siguientes imágenes representan el cumplimiento de los requisitos funcionales RF1, RF2, RF3, RF4, RF5, ver Figura 14 Higrómetro analógico definir valores mínimos y máximos de la escala y Figura 15 Higrómetro analógico modificar valores mínimos y máximos de la escala.



Figura 14 Higrómetro analógico definir valores mínimos y máximos de la escala (Elaboración propia).



Figura 15 Higrómetro analógico modificar valores mínimos y máximos de la escala (Elaboración propia)

Las siguientes imágenes corresponden al resultado de los requisitos funcionales RF6, RF 7, RF 17, RF 15, RF 18, RF 19 ver Figura 16 Adicionar rango de la escala en el medidor y Figura 17 Modificación dinámica de los rangos.

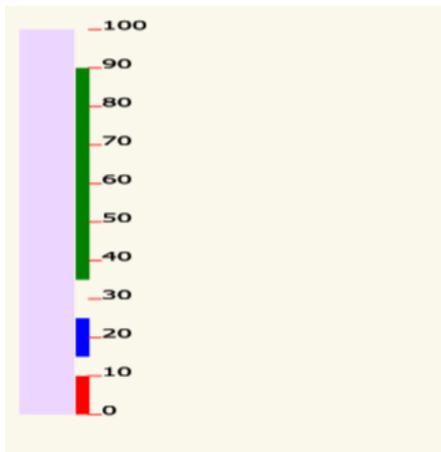


Figura 16 Modificación dinámica de los rangos (Elaboración propia).

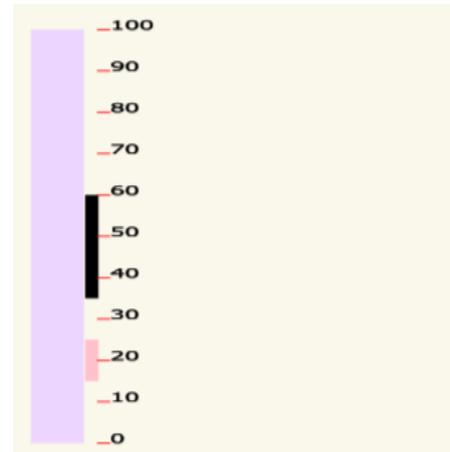


Figura 17 Adicionar rango de la escala en el medidor (Elaboración propia).

Las imágenes siguientes representan el cumplimiento del requisito funcional RF 16 Configurar opacidad del componente gráfico, ver figura 18 y 19.

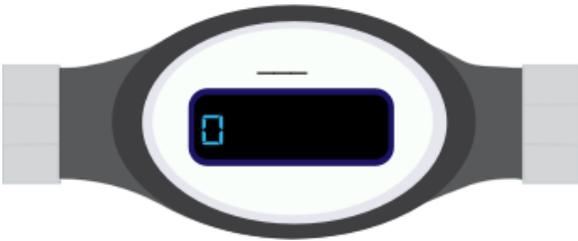


Figura 18 Flujómetro digital, configurar opacidad (Elaboración propia).

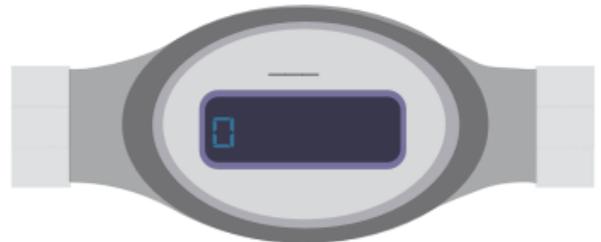


Figura 19 Flujómetro digital, propiedad opacidad modificada (Elaboración propia).

Las imágenes siguientes representan los requisitos funcionales RF 8, RF 13, RF 14, ver Figura 20 Modificación del color de la escala del medidor, Figura 21 Definición de la amplitud de la escala radial, Figura 22 Modificación del color de la escala del medidor.

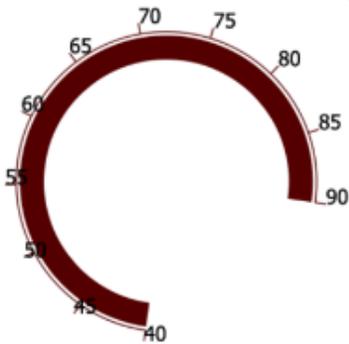


Figura 21 Definición de ángulo inicial y final de la escala radial (Elaboración propia).

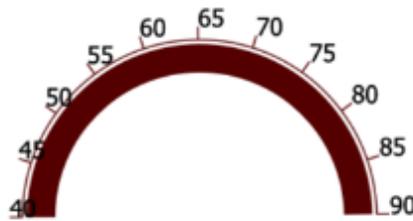


Figura 20 Definición de la amplitud de la escala radial (Elaboración propia).

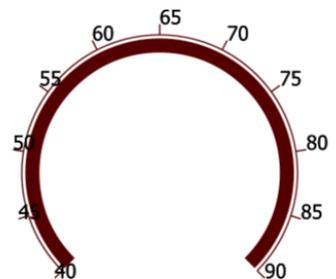


Figura 22 Modificación del color de la escala del medidor (Elaboración propia).

Las siguientes imágenes corresponden al resultado de los requisitos funcionales RF9, RF10, ver Figura 23 Flujómetro analógico configuración de fondo, Figura 24 Flujómetro analógico color de fondo modificado,

Figura 25 Medidor radial Configuración del color de la aguja, Figura 26 Medidor radial color de la aguja modificado.

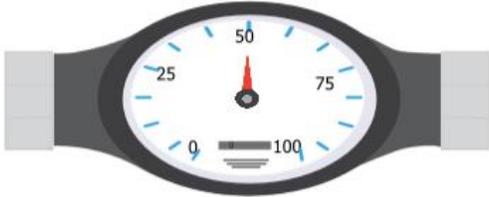


Figura 23 Flujómetro analógico configuración de fondo (Elaboración propia).

Figura 24 Flujómetro analógico color de fondo modificado (Elaboración propia).



Figura 25 Medidor radial configuración del color de la aguja (Elaboración propia).

Figura 26 Medidor radial color de la aguja (Elaboración propia).

Las imágenes siguientes representan el cumplimiento de los requisitos funcionales RF 11, RF 12, ver figura 27 Higrómetro analógico configuración del valor y la unidad de medida y Figura 28 Higrómetro analógico valor y la unidad de medida modificados.



Figura 27 Higrómetro analógico configuración del valor y la unidad de medida (Elaboración propia).



Figura 28 Higrómetro analógico valor y la unidad de medida modificados (Elaboración propia).

### 3.4 Pruebas de *software*

La realización de las pruebas permite evaluar el desempeño de los componentes gráficos desarrollados. En relación con el aspecto que se desee comprobar existen diferentes tipos de pruebas que pueden ser ejecutadas.

La metodología AUP-UCI describe varios tipos de pruebas de *software*; las cuales son un elemento crítico para la garantía de la calidad del *software* y representan una revisión final de las especificaciones, del diseño y de la codificación. El objetivo fundamental de las pruebas es descubrir diferentes clases de errores con la menor cantidad de tiempo y de esfuerzo. Aunque las pruebas no pueden asegurar la ausencia de defectos; sí pueden demostrar que existen defectos en el *software* (Pressman, R. 2012).

#### 3.4.1 Tipos de pruebas de *software*

A continuación, se exponen algunos tipos de pruebas de las cuales se hace referencia al objetivo general de cada una de ellas:

- **Pruebas unitarias:** se focaliza en ejecutar cada módulo (o unidad mínima a ser probada) lo que provee un mejor modo de manejar la integración de las unidades en componentes mayores. Busca asegurar que el código funciona de acuerdo con las especificaciones y que el módulo lógico es válido (Pressman, R. 2012).

#### Pruebas de integración

- **Pruebas de integración:** identificar errores introducidos por la combinación de programas probados unitariamente. determina cómo la base de datos de prueba será cargada. Verificar que las interfaces

entre las entidades externas (usuarios) y las aplicaciones funcionan correctamente. Verificar que las especificaciones de diseño sean alcanzadas. Determina el enfoque para avanzar desde un nivel de integración de las componentes al siguiente (*Pressman, R. 2012*).

- **Pruebas de regresión:** determinar si los cambios recientes en una parte de la aplicación tienen efecto adverso en otras partes (*Pressman, R. 2012*).

### Pruebas del sistema

- **Pruebas del sistema:** asegurar la apropiada navegación dentro del sistema, ingreso de datos, procesamiento y recuperación (*Pressman, R. 2012*).
- **Pruebas de desempeño:** miden tiempos de respuesta, índices de procesamiento de transacciones y otros requisitos sensibles al tiempo. El objetivo de las pruebas de desempeño es verificar y validar los requisitos de desempeño que se han especificado (en este caso, el desempeño ofrecido por el proponente (*Pressman, R. 2012*)).
- **Pruebas de seguridad y control de acceso:** tiene como objetivo verifica que un actor solo pueda acceder a las funciones y datos que su usuario tiene permitido (*Pressman, R. 2012*).

### Pruebas de validación a sistemas a la medida

- **Pruebas de aceptación:** son ejecutadas antes de que la aplicación sea instalada dentro de un ambiente de producción. Son generalmente desarrolladas y ejecutadas por el cliente o un especialista de la aplicación y son conducidas a determinar como el sistema satisface sus criterios de aceptación validando los requisitos que han sido levantados para el desarrollo, incluyendo a documentación y procesos de negocio (*Pressman, R. 2012*).

Las **Pruebas de Caja Negra**, es una técnica de pruebas de *software* en la cual la funcionalidad se verifica sin tomar en cuenta la estructura interna de código, detalles de implementación o escenarios de ejecución internos en el *software* (*Pressman, R. 2012*).

Esta técnica, se enfoca solamente en las entradas y salidas del sistema, sin preocuparnos en tener conocimiento de la estructura interna del programa de *software*. Para obtener el detalle de cuáles deben ser esas entradas y salidas, se basa en los requerimientos de *software* y especificaciones funcionales (*Pressman, R. 2012*).

La técnica de partición de equivalencias se basa en dividir en subconjuntos equivalentes respecto a una relación específica el dominio de las entradas. La prueba de un valor representativo de una clase permite

suponer que el resultado obtenido será el mismo que para otro valor de la clase. Se realiza un conjunto representativo de casos de prueba para cada clase de equivalencia (Blanco, B. C 2015).

### 3.5 Pruebas internas

#### 3.5.1 Pruebas funcionales

Para comprobar el correcto funcionamiento de la paleta de componentes se aplicó la técnica de pruebas de caja negra utilizando a su vez la técnica de partición de equivalencias; pues estas son utilizadas para realizar pruebas funcionales, basadas en las funciones o características del sistema y su interacción con otros sistemas o componentes.

Además, se realizaron las pruebas de aceptación, pues estas son las indicadas para evaluar el *software* mediante las pruebas planificadas y organizadas formalmente para determinar si se cumplen los requisitos de aceptación marcados por el cliente.

Los casos de prueba que se presentan a continuación pretenden demostrar que las funciones del sistema son operativas, que la entrada de los valores válidos y no válidos se acepta de forma adecuada y que se produce un resultado correcto, ver Tabla 10 Diseño de casos de pruebas.

#### Diseño de casos de prueba

En las siguientes tablas se muestran los diseños de casos de pruebas referentes al uso de la técnica de caja negra, ver tabla 10 Diseño de casos de pruebas.

Tabla 8 Diseño de casos de pruebas.

SC1 Crear instancia del componente gráfico.				
<b>Descripción general</b>				
El sistema debe proporcionar que el usuario cree el componente gráfico seleccionado mediante el clic en la paleta del componente.				
Escenario	Descripción	Variable	Respuesta del sistema	Flujo central

## *Capítulo III: Implementación y pruebas del sistema*

EC 1.1 Crear instancia del componente gráfico.	El sistema debe proporcionar que el usuario cree el componente gráfico seleccionado mediante el clic en la paleta del componente.	N/A	Muestra el componente en el despliegue tantas veces sea instanciado.	Siguiendo la siguiente ruta: 1-Se selecciona el componente de la paleta con un clic. 2-Se da clic en el despliegue. 3-Se repite este proceso cuantas instancias se desee crear.
--	---	-----	--	--

Los restantes diseños de casos de pruebas se encuentran en el Anexo 3 Diseños de casos de pruebas. Un caso de prueba es una serie de acciones que se realizan para determinar una función o funcionalidad particular de su aplicación (Somerville, I 2011), ver tabla 9 Prueba de aceptación#1.

Tabla 9 Prueba de aceptación #1

<b>Prueba de Aceptación</b>	
Número: 1	Historia de usuario: 1
Nombre: Crear instancia del componente gráfico.	
Descripción: El sistema debe proporcionar que el usuario cree el componente gráfico seleccionado mediante el clic en la paleta del componente.	
Condiciones de Ejecución: : El usuario debe comprobar que el usuario cree el componente gráfico seleccionado mediante el clic en la paleta del componente.	
Entradas/ Pasos de Ejecución:	
<ul style="list-style-type: none"> <li>• Seleccionar componente con el clic izquierdo.</li> <li>• Clic sobre el área de trabajo del despliegue donde se visualizará dicho componente.</li> </ul>	
Resultado esperado: El componente se inserta correctamente.	
Evaluación de la prueba: Una vez realizada la primera iteración no se detectaron no conformidades, por lo cual no fue necesario otra iteración.	

Tabla 10 Prueba de aceptación #2

<b>Prueba de Aceptación</b>	
Número: 2	Historia de usuario: 2
Nombre: Definir valor mínimo de la escala del medidor.	
Descripción: El sistema debe proporcionar que el usuario defina el valor mínimo de la escala del medidor.	

<b>Condiciones de Ejecución:</b> El usuario debe comprobar que el valor mínimo de la escala del medidor pueda ser definido.
<b>Entradas/ Pasos de Ejecución:</b> <ul style="list-style-type: none"> <li>• Seleccionar el componente.</li> <li>• Abrir el inspector de propiedades.</li> <li>• Seleccionar opción <i>minValue</i>.</li> <li>• Introducir el valor deseado como valor mínimo.</li> </ul>
<b>Resultado esperado:</b> El valor mínimo se inserta correctamente.
<b>Evaluación de la prueba:</b> Una vez realizada la primera iteración se detectaron varias no conformidades, a las cuales se les da solución y se realiza otra iteración que arrojó una evaluación satisfactoria.

Tabla 11 Prueba de aceptación #3

Prueba de Aceptación	
<b>Número:</b> 3	<b>Historia de usuario:</b> 3
<b>Nombre:</b> Definir valor máximo de la escala del medidor.	
<b>Descripción:</b> El sistema debe proporcionar que el usuario defina el valor máximo de la escala del medidor.	
<b>Condiciones de Ejecución:</b> El usuario debe comprobar que el valor máximo de la escala del medidor pueda ser definido.	
<b>Entradas/ Pasos de Ejecución:</b> <ul style="list-style-type: none"> <li>• Seleccionar el componente.</li> <li>• Abrir el inspector de propiedades.</li> <li>• Seleccionar opción <i>maxValue</i>.</li> <li>• Introducir el valor deseado como valor máximo.</li> </ul>	
<b>Resultado esperado:</b> El valor máximo se inserta correctamente.	
<b>Evaluación de la prueba:</b> Una vez realizada la primera iteración se detectaron varias no conformidades, a las cuales se les da solución y se realiza otra iteración que arrojó una evaluación satisfactoria.	

Las restantes pruebas de aceptación se encuentran en el Anexo 2 Pruebas de aceptación.

Después de realizar las pruebas funcionales mediante la técnica de caja negra y las pruebas de aceptación, se comprobó el correcto funcionamiento de la interfaz y la codificación del sistema. Cada problema detectado en el desarrollo del sistema fue resuelto a raíz del trabajo continuo del desarrollador, con un total

de seis no conformidades encontradas en la primera iteración, cuatro en la segunda y una en la tercera, las cuales se dividieron en significativas y no significativas.

Además, estas no conformidades fueron clasificadas según su tipo:

- Para una cantidad de seis no conformidad el error detectado fue de ejecución, durante el despliegue del sistema fueron encontradas diferentes fallas, siendo estas ocasionadas porque el sistema no realizaba todos los procedimientos definidos.
- Para las siguientes cuatro no conformidades se revelan errores de presentación porque no se visualiza correctamente toda la información que se desea mostrar.
- Para la restante no conformidad el error fue de redacción se localizó un error ortográfico siendo este de acentuación.

A continuación, se representa lo expuesto anteriormente a través de la siguiente gráfica, ver figura 29 Iteración de caja negra.

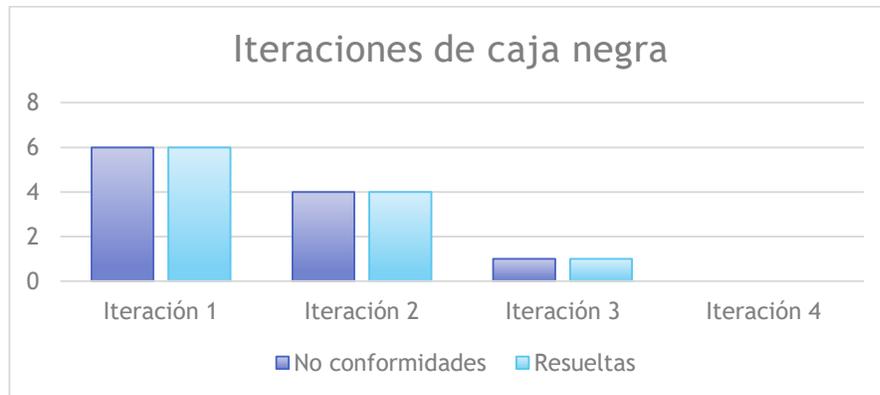


Figura 29 Iteraciones de caja negra.

El resultado de las pruebas realizadas al sistema comprobó que tanto los requisitos funcionales y no funcionales planteados durante la etapa de análisis y diseño funcionan satisfactoriamente.

### Conclusiones parciales

- La realización del diagrama de componente permitió comprender las restricciones impuestas por el lenguaje de programación.
- La definición del estándar de codificación heredado del propio sistema AREX permitió obtener una codificación lineal y clara para futuros mantenimientos al sistema por otros desarrolladores.
- La realización de las pruebas a los componentes gráficos permitió evaluar la calidad y desempeño de los medidores desarrollados.

### *Capítulo III: Implementación y pruebas del sistema*

- Al realizar las pruebas funcionales al sistema fueron detectadas a tiempo un grupo de no conformidades en cada iteración, las cuales fueron resueltas al final de cada una.

*Conclusiones generales*

- El estudio del sistema AREX evidencia la necesidad de implementar una paleta de componentes que represente a los medidores analógicos y digitales para poder ser aplicados en la industria.
- El estudio del funcionamiento de los sistemas de medición permitió identificar los componentes necesarios para su representación en el sistema AREX.
- Con la implementación de la paleta de componentes que representa los medidores se da cumplimiento a los requerimientos funcionales y no funcionales establecidos.
- El desarrollo de la investigación permitió lograr un mayor grado de atonicidad en los componentes permitiendo que estos puedan ser desacoplados y diseñar nuevos medidores a partir de los ya existentes.
- La realización de las pruebas a los componentes gráficos permitió evaluar la calidad y desempeño de los medidores desarrollados.

*Glosario de términos*

**TIC:** son aquellas cuya base se centra en los campos de la informática, la microelectrónica y las telecomunicaciones, para dar paso a la creación de nuevas formas de comunicación. Son un conjunto de herramientas o recursos de tipo tecnológico y comunicacional, que sirven para facilitar la emisión, acceso y tratamiento de la información mediante códigos variados que pueden corresponder a textos, imágenes, sonidos, entre otros.

**IoT:** la internet de las cosas (*IoT*, por sus siglas en inglés) es un sistema de dispositivos de computación interrelacionados, máquinas mecánicas y digitales, objetos, animales o personas que tienen identificadores únicos y la capacidad de transferir datos a través de una red, sin requerir de interacciones humano a humano o humano a computadora («¿Qué es Internet de las cosas (IoT)? - Definición en WhatIs.com» [sin fecha]).

**Framework:** es el esquema o estructura que se establece y que se aprovecha para desarrollar y organizar un software determinado.

**Atomicidad:** es la propiedad que asegura que una operación se ha realizado o no, y por lo tanto ante un fallo del sistema no puede quedar a medias.

**HMI:** es la interfaz entre el proceso y los operadores, básicamente un panel del operador. Es la herramienta principal con la cual los operadores y los supervisores de la línea coordinan y controlan los procesos industriales y de fabricación en la planta. Las HMI sirven para traducir las variables del proceso complejas en información útil y aprovechable.

**Electromecánicos:** estos dispositivos son los que combinan partes eléctricas y mecánicas para conformar su mecanismo. Ejemplos de estos dispositivos son los motores eléctricos y los dispositivos mecánicos movidos por estos, así como las ya obsoletas calculadoras mecánicas y máquinas de sumar; los relés; las válvulas a solenoide; y las diversas clases de interruptores y llaves de selección eléctricas.

**Electroneumáticos:** están constituidos básicamente para la manipulación y acondicionamiento de las señales de voltaje y corriente que deberán de ser transmitidas a dispositivos de conversión de energía eléctrica a energía neumática para lograr la activación de los actuadores neumáticos.

**Modsim (*Modular simulator for ore dressing plants*):** es un *software* de simulación de plantas de procesamiento de minerales. Este *software* ha sido utilizado principalmente como herramienta académica de simulación de procesos estacionarios, ya que no está diseñado para operaciones dinámicas.

### *Recomendaciones*

Teniendo en cuenta los resultados obtenidos durante la investigación y basados en la experiencia adquirida, se recomienda:

- Incorporar nuevos componentes gráficos que representen medidores en la paleta "Meters 2".
- Agregar funcionalidades en el Editor AREX que permitan una mayor usabilidad de los componentes gráficos tales: manipulación de grillado, alineación, agrupación y desagrupación, rotación dinámica.

### Referencias bibliográficas

- ALFONSOGU, 2012. QML. *Velneo* [en línea]. [Consulta: 20 marzo 2019]. Disponible en: <https://velneo.es/qml/>.
- ARLETHPAREDES, 2012. Patrones de arquitectura vs. Patrones de diseño. *Ingeniería del Software* [en línea]. [Consulta: 21 febrero 2019]. Disponible en: <https://arlethparedes.wordpress.com/2012/08/27/patrones-de-arquitectura-vs-patrones-de-diseno/>.
- BLANCO, B. C, 2015. *Ingeniería de Software - Construcción y pruebas*. [en línea]. 2015. S.l.: s.n. Disponible en: [http://ocw.unican.es/enseñanzas-tecnicas/ingenieria-del-software-ii/materiales/tema1 - %0ApruebasSistemasSoftware.pdf](http://ocw.unican.es/enseñanzas-tecnicas/ingenieria-del-software-ii/materiales/tema1-%0ApruebasSistemasSoftware.pdf).
- CARDOZZO, D. R., 2014. *Desarrollo de Software*. 2014. S.l.: Campus Academy. IT Campus Academy.
- CENTRO DE INFORMÁTICA INDUSTRIAL, CEDIN., C. de I.I., CEDIN., 2019. *Manual de Usuario Sistema de Medición y Adquisición AREX 3.0*. 29 marzo 2019. S.l.: s.n.
- GRUPO EROS SERCONI., G.E.S., 2014. *SISTEMA DE SUPERVISIÓN Y CONTROL DE PROCESOS*. S.l.: s.n.
- IONOS ESPAÑA S.L.U., 2019. UML: lenguaje unificado de modelado orientado a objetos. *1&1 Digitalguide* [en línea]. [Consulta: 30 mayo 2019]. Disponible en: <https://www.ionos.es/digitalguide/paginas-web/desarrollo-web/uml-lenguaje-unificado-de-modelado-orientado-a-objetos/>.
- ISRAEL FLORES, 2017. Sistemas de medición. [en línea]. Educación. S.l. [Consulta: 9 mayo 2019]. Disponible en: <https://es.slideshare.net/israelflores7712/sistemas-de-medicion-72488066>.
- LILIAN MOLINERO, 2012. Componentes informáticos. [en línea]. Educación. S.l. [Consulta: 30 abril 2019]. Disponible en: <https://es.slideshare.net/ProfeLilian/componentes-informticos>.
- MANUEL ARIAS CALLEJA, 2017. *Estándares de codificación*. 2017. S.l.: s.n.
- MIRANDA, C.V., 2017. *Informática industrial*. S.l.: Ediciones Paraninfo, S.A. ISBN 978-84-9732-614-8.
- OSCAR PÁEZ RIVERA, 2015. 304\_Norma\_ISA\_PID.pdf. *Scribd* [en línea]. [Consulta: 29 mayo 2019]. Disponible en: <https://es.scribd.com/document/180069317/304-Norma-ISA-PID-pdf>.
- PATRICIA LÓPEZ, 2017. *INGENIERÍA DEL SOFTWARE I Práctica 1 Herramienta CASE Visual Paradigm Univ. Cantabria – Fac.* 2017. S.l.: s.n.
- PAUL, M, 2007. *Importing Vector Graphics: The grim port Package for R*. Retrieved [en línea]. 2007. S.l.: s.n. Disponible en: <https://cran.r-project.org/web/packages/grlImport/vignettes/import.pdf>.
- PRESSMAN, R., 2012. *Ingeniería Del Software I Un Enfoque Práctico*. In *Ingeniería Del Software I (7th ed.)*. NUEVA YORK: s.n.

- PROGRAMACION EN CASTELLANO, S.L., 2019. Qt Creator, un completo entorno de desarrollo. *Programación en Castellano*. [en línea]. [Consulta: 20 febrero 2019]. Disponible en: [http://programacion.net/noticia/qt\\_creator-\\_un\\_completo\\_entorno\\_de\\_desarrollo\\_1723](http://programacion.net/noticia/qt_creator-_un_completo_entorno_de_desarrollo_1723).
- ¿Qué es Internet de las cosas (IoT)? - Definición en WhatIs.com. [en línea], [sin fecha]. [Consulta: 16 mayo 2019]. Disponible en: <https://searchdatacenter.techtarget.com/es/definicion/Internet-de-las-cosas-IoT>.
- RODRÍGUEZ, T, 2014. *Metodología de desarrollo para la Actividad productiva de la UCI*. 2014. S.l.: s.n.
- SCHNEIDER ELECTRIC, 2018. StruxureWare Building Operation | Schneider Electric. [en línea]. [Consulta: 29 mayo 2019]. Disponible en: <https://www.se.com/es/es/work/solutions/enterprise-solutions/solutions/enterprise-software-suites-building-operation/case-studies.jsp>.
- SECOND EPISODE S.L, 2017. Higrómetros: Descubre todos los tipos, analógicos, digitales... | GESA. [en línea]. [Consulta: 29 mayo 2019]. Disponible en: <https://www.termometros.com/higrometro>.
- SIEMENS AG, 2009. SIMATIC WinCC flexible.
- SOMERVILLE, I, 2011. *Software Engineering*. Addison-Wesley. S.l.: s.n.
- THE QT COMPANY, T.Q., 2019. Qt | Cross-platform software development for embedded & desktop. [en línea]. [Consulta: 20 febrero 2019]. Disponible en: <https://www.qt.io>.
- THE WATCHES TIMES, 2018. El Reloj digital. ¿Cómo funciona? | The Watches Times. *The Watches Times Fabricantes de Relojes* [en línea]. [Consulta: 29 mayo 2019]. Disponible en: <http://www.thewatchestimes.com/el-reloj-digital-como-funciona/>.
- TIPIGAO, M., 2013. Acerca de | Inkscape. [en línea]. [Consulta: 30 mayo 2019]. Disponible en: <https://inkscape.org/es/acerca-de/>.
- U/SORIBEL, 2017. El reloj analógico y el reloj Digital. *GeoGebra* [en línea]. [Consulta: 22 febrero 2019]. Disponible en: <https://www.geogebra.org/m/xW8mUVyP>.
- VIU UNIVERSIDAD INTERNACIONAL DE VALENCIA, 2018. Diferencias entre señal analógica y digital | VIU. [en línea]. [Consulta: 31 mayo 2019]. Disponible en: <https://www.universidadviu.com/diferencias-senal-analogica-digital/>.

Anexo 1 Historias de usuarios

HU04

Tabla 12 Historia de usuario#4

Historia de usuario	
<b>Número:</b> 4	<b>Nombre del requisito:</b> Modificar valor mínimo de la escala del medidor.
<b>Programador:</b> Arlette Lorenzo Tovar	<b>Iteración Asignada:</b> 1
<b>Prioridad:</b> Baja	<b>Tiempo Estimado:</b> 0.8
<b>Riesgo en Desarrollo:</b> Medio	<b>Tiempo Real:</b> 0.8
<b>Descripción:</b> El sistema debe proporcionar que el usuario modifique la propiedad "valor mínimo" del componente gráfico mediante el inspector de propiedades.	

HU05

Tabla 13 Historia de usuario#5

Historia de usuario	
<b>Número:</b> 5	<b>Nombre del requisito:</b> Modificar valor máximo de la escala del medidor.
<b>Programador:</b> Arlette Lorenzo Tovar	<b>Iteración Asignada:</b> 1
<b>Prioridad:</b> Baja	<b>Tiempo Estimado:</b> 0.8
<b>Riesgo en Desarrollo:</b> Medio	<b>Tiempo Real:</b> 0.8
<b>Descripción:</b> El sistema debe proporcionar que el usuario modifique la propiedad "valor máximo" del componente gráfico mediante el inspector de propiedades.	

HU06

Tabla 14 Historia de usuario#6

Historia de usuario	
<b>Número:</b> 6	<b>Nombre del requisito:</b> Modificar valor mínimo del rango seleccionado en el medidor.
<b>Programador:</b> Arlette Lorenzo Tovar	<b>Iteración Asignada:</b> 1
<b>Prioridad:</b> Alta	<b>Tiempo Estimado:</b> 1
<b>Riesgo en Desarrollo:</b> Alto	<b>Tiempo Real:</b> 1

**Descripción:** El sistema debe proporcionar que el usuario modifique la propiedad “rangos” del componente gráfico mediante el inspector de propiedades.

HU07

Tabla 15 Historia de usuario#7

Historia de usuario	
<b>Número:</b> 7	<b>Nombre del requisito:</b> Modificar valor máximo del rango seleccionado en el medidor.
<b>Programador:</b> Arlette Lorenzo Tovar	<b>Iteración Asignada:</b> 1
<b>Prioridad:</b> Baja	<b>Tiempo Estimado:</b> 0.8
<b>Riesgo en Desarrollo:</b> Medio	<b>Tiempo Real:</b> 0.8
<b>Descripción:</b> El sistema debe proporcionar que el usuario modifique la propiedad “rangos” del componente gráfico mediante el inspector de propiedades.	

HU08

Tabla 16 Historia de usuario#8

Historia de usuario	
<b>Número:</b> 8	<b>Nombre del requisito:</b> Modificar color de la escala del medidor.
<b>Programador:</b> Arlette Lorenzo Tovar	<b>Iteración Asignada:</b> 3
<b>Prioridad:</b> Baja	<b>Tiempo Estimado:</b> 0.8
<b>Riesgo en Desarrollo:</b> Medio	<b>Tiempo Real:</b> 0.8
<b>Descripción:</b> El sistema debe proporcionar que el usuario modifique la propiedad “color de escala” del componente gráfico mediante el inspector de propiedades.	

HU09

Tabla 17 Historia de usuario#9

Historia de usuario	
<b>Número:</b> 9	<b>Nombre del requisito:</b> Modificar color de fondo del medidor.

<b>Programador:</b> Arlette Lorenzo Tovar	<b>Iteración Asignada:</b> 2
<b>Prioridad:</b> Baja	<b>Tiempo Estimado:</b> 0.8
<b>Riesgo en Desarrollo:</b> Medio	<b>Tiempo Real:</b> 0.8
<b>Descripción:</b> El sistema debe proporcionar que el usuario modifique la propiedad "color de fondo" del componente gráfico mediante el inspector de propiedades.	

HU10

Tabla 18 Historia de usuario#10

Historia de usuario	
<b>Número:</b> 10	<b>Nombre del requisito:</b> Definir color de fondo de la aguja.
<b>Programador:</b> Arlette Lorenzo Tovar	<b>Iteración Asignada:</b> 2
<b>Prioridad:</b> Baja	<b>Tiempo Estimado:</b> 0.8
<b>Riesgo en Desarrollo:</b> Medio	<b>Tiempo Real:</b> 0.8
<b>Descripción:</b> El sistema debe proporcionar que el usuario defina el color de fondo de la aguja.	

HU11

Tabla 19 Historia de usuario#11

Historia de usuario	
<b>Número:</b> 11	<b>Nombre del requisito:</b> Definir unidad de medida (texto) del medidor.
<b>Programador:</b> Arlette Lorenzo Tovar	<b>Iteración Asignada:</b> 2
<b>Prioridad:</b> Baja	<b>Tiempo Estimado:</b> 0.8
<b>Riesgo en Desarrollo:</b> Medio	<b>Tiempo Real:</b> 0.8
<b>Descripción:</b> El sistema debe proporcionar que el usuario defina la unidad de medida del medidor.	

HU12

Tabla 20 Historia de usuario#12

Historia de usuario	
<b>Número:</b> 12	<b>Nombre del requisito:</b> Modificar valor del medidor.
<b>Programador:</b> Arlette Lorenzo Tovar	<b>Iteración Asignada:</b> 2

<b>Prioridad:</b> Baja	<b>Tiempo Estimado:</b> 0.8
<b>Riesgo en Desarrollo:</b> Medio	<b>Tiempo Real:</b> 0.8
<b>Descripción:</b> El sistema debe proporcionar que el usuario modifique la propiedad "valor" del componente gráfico mediante el inspector de propiedades.	

HU13

Tabla 21 Historia de usuario#13

Historia de usuario	
<b>Número:</b> 13	<b>Nombre del requisito:</b> Definir la amplitud de la escala radial.
<b>Programador:</b> Arlette Lorenzo Tovar	<b>Iteración Asignada:</b> 2
<b>Prioridad:</b> Alta	<b>Tiempo Estimado:</b> 0.8
<b>Riesgo en Desarrollo:</b> Medio	<b>Tiempo Real:</b> 0.8
<b>Descripción:</b> El sistema debe proporcionar que el usuario defina la amplitud de la escala radial.	

HU14

Tabla 22 Historia de usuario#14

Historia de usuario	
<b>Número:</b> 14	<b>Nombre del requisito:</b> Definir ángulo inicial de la escala radial.
<b>Programador:</b> Arlette Lorenzo Tovar	<b>Iteración Asignada:</b> 2
<b>Prioridad:</b> Alta	<b>Tiempo Estimado:</b> 0.8
<b>Riesgo en Desarrollo:</b> Medio	<b>Tiempo Real:</b> 0.8
<b>Descripción:</b> El sistema debe proporcionar que el usuario defina el ángulo inicial de la escala radial.	

HU15

Tabla 23 Historia de usuario#15

Historia de usuario	
<b>Número:</b> 15	<b>Nombre del requisito:</b> Modificar color del rango seleccionado en el medidor.
<b>Programador:</b> Arlette Lorenzo Tovar	<b>Iteración Asignada:</b> 2
<b>Prioridad:</b> Baja	<b>Tiempo Estimado:</b> 0.8

<b>Riesgo en Desarrollo:</b> Medio	<b>Tiempo Real:</b> 0.8
<b>Descripción:</b> El sistema debe proporcionar que el usuario modifique la propiedad "rangos" del componente gráfico mediante el inspector de propiedades.	

HU16

Tabla 24 Historia de usuario#16

Historia de usuario	
<b>Número:</b> 16	<b>Nombre de requisito:</b> Configurar opacidad del componente gráfico.
<b>Programador:</b> Arlette Lorenzo Tovar	<b>Iteración Asignada:</b> 3
<b>Prioridad:</b> Baja	<b>Tiempo Estimado:</b> 0.8
<b>Riesgo en Desarrollo:</b> Medio	<b>Tiempo Real:</b> 0.8
<b>Descripción:</b> El sistema debe proporcionar que el usuario modifique la propiedad "opacidad" del componente gráfico mediante el inspector de propiedades.	

HU17

Tabla 25 Historia de usuario#17

Historia de usuario	
<b>Número:</b> 17	<b>Nombre de requisito:</b> Adicionar rango de la escala en el medidor
<b>Programador:</b> Arlette Lorenzo Tovar	<b>Iteración Asignada:</b> 3
<b>Prioridad:</b> Alta	<b>Tiempo Estimado:</b> 1
<b>Riesgo en Desarrollo:</b> Medio	<b>Tiempo Real:</b> 1
<b>Descripción:</b> El sistema debe proporcionar que el usuario adicione un nuevo rango en la propiedad "rangos" del componente gráfico mediante el inspector de propiedades.	

HU18

Tabla 26 Historia de usuario#18

Historia de usuario	
<b>Número:</b> 18	<b>Nombre del requisito:</b> Mostrar rango de la escala en el medidor.
<b>Programador:</b> Arlette Lorenzo Tovar	<b>Iteración Asignada:</b> 3

<b>Prioridad:</b> Baja	<b>Tiempo Estimado:</b> 0.8
<b>Riesgo en Desarrollo:</b> Medio	<b>Tiempo Real:</b> 0.8
<b>Descripción:</b> El sistema debe proporcionar mostrar rango de la escala en el medidor.	

HU19

Tabla 27 Historia de usuario#19

Historia de usuario	
<b>Número:</b> 19	<b>Nombre del requisito:</b> Eliminar rango de la escala en el medidor.
<b>Programador:</b> Arlette Lorenzo Tovar	<b>Iteración Asignada:</b> 3
<b>Prioridad:</b> Media	<b>Tiempo Estimado:</b> 0.8
<b>Riesgo en Desarrollo:</b> Alto	<b>Tiempo Real:</b> 0.8
<b>Descripción:</b> El sistema debe proporcionar que el usuario elimine el rango deseado en la propiedad "rangos" del componente gráfico mediante el inspector de propiedades.	

Anexos 2 Pruebas de aceptación

Tabla 28 Prueba de aceptación #4

Prueba de Aceptación	
Número: 4	Historia de usuario: 4
<b>Nombre:</b> Modificar valor mínimo de la escala del medidor.	
<b>Descripción:</b> El sistema debe proporcionar que el usuario modifique la propiedad "valor mínimo" del componente gráfico mediante el inspector de propiedades.	
<b>Condiciones de Ejecución:</b> El usuario debe comprobar que se modifique la propiedad "valor mínimo" del componente gráfico mediante el inspector de propiedades.	
<b>Entradas/ Pasos de Ejecución:</b> <ul style="list-style-type: none"> <li>• Seleccionar el componente.</li> <li>• Abrir el inspector de propiedades.</li> <li>• Seleccionar opción <i>minValue</i>.</li> <li>• Introducir el valor deseado como valor mínimo.</li> </ul>	
<b>Resultado esperado:</b> El valor mínimo puede ser cambiado por otro cualquiera menor que el valor máximo definido.	
<b>Evaluación de la prueba:</b> Una vez realizada la primera iteración se detectaron varias no conformidades, a las cuales se les da solución y se realiza otra iteración que arrojó una evaluación satisfactoria.	

Tabla 29 Prueba de aceptación #5

Prueba de Aceptación	
Número: 5	Historia de usuario: 5
<b>Nombre:</b> Modificar valor máximo de la escala del Medidor.	
<b>Descripción:</b> El sistema debe proporcionar que el usuario modifique la propiedad "valor máximo" del componente gráfico mediante el inspector de propiedades.	
<b>Condiciones de Ejecución:</b> : El usuario debe comprobar que se modifique la propiedad "valor máximo" del componente gráfico mediante el inspector de propiedades definiendo un nuevo valor máximo en el mismo.	
<b>Entradas/ Pasos de Ejecución:</b> <ul style="list-style-type: none"> <li>• Seleccionar el componente.</li> <li>• Abrir el inspector de propiedades.</li> <li>• Seleccionar opción <i>maxValue</i>.</li> <li>• Introducir el nuevo valor deseado como valor máximo.</li> </ul>	

<b>Resultado esperado:</b> El valor máximo puede ser cambiado por otro cualquiera mayor que el valor mínimo definido.
<b>Evaluación de la prueba:</b> Una vez realizada la primera iteración se detectaron varias no conformidades, a las cuales se les da solución y se realiza otra iteración que arrojó una evaluación satisfactoria.

Tabla 30 Prueba de aceptación #6

Prueba de Aceptación	
<b>Número:</b> 6	<b>Historia de usuario:</b> 6
<b>Nombre:</b> Modificar valor mínimo del rango seleccionado en el medidor.	
<b>Descripción:</b> El sistema debe proporcionar que el usuario modifique la propiedad "Ranges" del componente gráfico mediante el inspector de propiedades.	
<b>Condiciones de Ejecución:</b> El usuario debe comprobar que se modifique la propiedad "Ranges" del componente gráfico mediante el inspector de propiedades definiendo un nuevo valor mínimo en el mismo.	
<b>Entradas/ Pasos de Ejecución:</b>	
<ul style="list-style-type: none"> <li>• Seleccionar el componente.</li> <li>• Abrir el inspector de propiedades.</li> <li>• Seleccionar opción <i>Ranges</i>.</li> <li>• Introducir el nuevo valor deseado como valor mínimo.</li> </ul>	
<b>Resultado esperado:</b> El valor mínimo puede ser cambiado por otro cualquiera menor que el valor máximo definido.	
<b>Evaluación de la prueba:</b> Una vez realizada la primera iteración se detectaron varias no conformidades, a las cuales se les da solución y se realiza otra iteración que arrojó una evaluación satisfactoria.	

Tabla 31 Prueba de aceptación #7

Prueba de Aceptación	
<b>Número:</b> 7	<b>Historia de usuario:</b> 7
<b>Nombre:</b> Modificar valor máximo del rango seleccionado en el medidor.	
<b>Descripción:</b> El sistema debe proporcionar que el usuario modifique la propiedad "Ranges" del componente gráfico mediante el inspector de propiedades.	
<b>Condiciones de Ejecución:</b> : El usuario debe comprobar que se modifique la propiedad "Ranges" del componente gráfico mediante el inspector de propiedades definiendo un nuevo valor máximo en el mismo.	
<b>Entradas/ Pasos de Ejecución:</b>	
<ul style="list-style-type: none"> <li>• Seleccionar el componente.</li> <li>• Abrir el inspector de propiedades.</li> </ul>	

<ul style="list-style-type: none"> <li>• Seleccionar opción <i>Ranges</i>.</li> <li>• Introducir el nuevo valor deseado como valor máximo.</li> </ul>
<b>Resultado esperado:</b> : El valor máximo puede ser cambiado por otro cualquiera mayor que el valor mínimo definido.
<b>Evaluación de la prueba:</b> Una vez realizada la primera iteración se detectaron varias no conformidades, a las cuales se les da solución y se realiza otra iteración que arrojó una evaluación satisfactoria.

Tabla 32 Prueba de aceptación #8

Prueba de Aceptación	
<b>Número:</b> 8	<b>Historia de usuario:</b> 8
<b>Nombre:</b> Modificar color de la escala del medidor.	
<b>Descripción:</b> El sistema debe proporcionar que el usuario modifique la propiedad "color de escala" del componente gráfico mediante el inspector de propiedades.	
<b>Condiciones de Ejecución:</b> El usuario debe comprobar que se modifique la propiedad " <i>colorDefaultScale</i> " del componente gráfico mediante el inspector de propiedades.	
<b>Entradas/ Pasos de Ejecución:</b> <ul style="list-style-type: none"> <li>• Seleccionar el componente.</li> <li>• Abrir el inspector de propiedades.</li> <li>• Seleccionar opción <i>Color Default Scale</i>.</li> <li>• Seleccionar color del panel de colores.</li> </ul>	
<b>Resultado esperado:</b> El color de la escala del medidor puede ser cambiado por otro cualquiera.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria.	

Tabla 33 Prueba de aceptación #9

Prueba de Aceptación	
<b>Número:</b> 9	<b>Historia de usuario:</b> 9
<b>Nombre:</b> Modificar color de fondo del medidor.	
<b>Descripción:</b> El sistema debe proporcionar que el usuario modifique la propiedad "color de fondo" del componente gráfico mediante el inspector de propiedades.	
<b>Condiciones de Ejecución:</b> El usuario debe comprobar que se modifique la propiedad " <i>backgroundColor</i> " del componente gráfico mediante el inspector de propiedades.	

<b>Entradas/ Pasos de Ejecución:</b> <ul style="list-style-type: none"> <li>• Seleccionar el componente.</li> <li>• Abrir el inspector de propiedades.</li> <li>• Seleccionar opción <i>Background Color</i>.</li> <li>• Seleccionar color del panel de colores.</li> </ul>
<b>Resultado esperado:</b> El color de fondo del medidor puede ser cambiado por otro cualquiera.
<b>Evaluación de la prueba:</b> Prueba satisfactoria.

Tabla 34 Prueba de aceptación #10

Prueba de Aceptación	
<b>Número:</b> 10	<b>Historia de usuario:</b> 10
<b>Nombre:</b> Definir color de fondo de la aguja.	
<b>Descripción:</b> El sistema debe proporcionar que el usuario defina el color de fondo de la aguja.	
<b>Condiciones de Ejecución:</b> El usuario debe comprobar que se modifique la propiedad " <i>colorIndicatorNeedle</i> " del componente gráfico mediante el inspector de propiedades.	
<b>Entradas/ Pasos de Ejecución:</b> <ul style="list-style-type: none"> <li>• Seleccionar el componente.</li> <li>• Abrir el inspector de propiedades.</li> <li>• Seleccionar opción <i>Color Indicator Needle</i>.</li> <li>• Seleccionar color del panel de colores.</li> </ul>	
<b>Resultado esperado:</b> El color de fondo de la aguja puede ser modificado por cualquier otro.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria.	

Tabla 35 Prueba de aceptación #11

Prueba de Aceptación	
<b>Número:</b> 11	<b>Historia de usuario:</b> 11
<b>Nombre:</b> Definir unidad de medida (texto) del medidor.	
<b>Descripción:</b> El sistema debe proporcionar que el usuario defina la unidad de medida del medidor.	
<b>Condiciones de Ejecución:</b> El usuario debe comprobar que pueda ser definida la unidad de medida del medidor a través de la propiedad " <i>Unit</i> ".	
<b>Entradas/ Pasos de Ejecución:</b> <ul style="list-style-type: none"> <li>• Seleccionar el componente.</li> </ul>	

<ul style="list-style-type: none"> <li>• Abrir el inspector de propiedades.</li> <li>• Seleccionar opción <i>Unit</i>.</li> <li>• Seleccionar la unidad de medida.</li> </ul>
<b>Resultado esperado:</b> El valor de la variable <i>unit</i> puede ser modificado por cualquier otra unidad de medida del medidor seleccionado.
<b>Evaluación de la prueba:</b> Una vez realizada la primera iteración se detectaron varias no conformidades, a las cuales se les da solución y se realiza otra iteración que arrojó una evaluación satisfactoria.

Tabla 36 Prueba de aceptación #12

Prueba de Aceptación	
<b>Número:</b> 12	<b>Historia de usuario:</b> 12
<b>Nombre:</b> Modificar valor del medidor.	
<b>Descripción:</b> El sistema debe proporcionar que el usuario modifique la propiedad “valor” del componente gráfico mediante el inspector de propiedades.	
<b>Condiciones de Ejecución:</b> El usuario debe comprobar que pueda modificada la propiedad “ <i>Value</i> ” del componente gráfico mediante el inspector de propiedades .	
<b>Entradas/ Pasos de Ejecución:</b> <ul style="list-style-type: none"> <li>• Seleccionar el componente.</li> <li>• Abrir el inspector de propiedades.</li> <li>• Seleccionar opción <i>Value</i>.</li> <li>• Insertar el valor deseado.</li> </ul>	
<b>Resultado esperado:</b> La variable valor puede ser modificado por cualquier otro valor numérico.	
<b>Evaluación de la prueba:</b> Una vez realizada la primera iteración se detectaron varias no conformidades, a las cuales se les da solución y se realiza otra iteración que arrojó una evaluación satisfactoria.	

Tabla 37 Prueba de aceptación #13

Prueba de Aceptación	
<b>Número:</b> 13	<b>Historia de usuario:</b> 13
<b>Nombre:</b> Definir la amplitud de la escala radial.	
<b>Descripción:</b> El sistema debe proporcionar que el usuario defina la amplitud de la escala radial.	

<b>Condiciones de Ejecución:</b> El usuario debe comprobar que pueda ser definida la propiedad “ <i>amplitudeScale</i> ” de la escala radial.
<b>Entradas/ Pasos de Ejecución:</b> <ul style="list-style-type: none"> <li>• Seleccionar el componente.</li> <li>• Abrir el inspector de propiedades.</li> <li>• Seleccionar opción <i>Amplitude Scale</i>.</li> <li>• Insertar el valor deseado.</li> </ul>
<b>Resultado esperado:</b> La amplitud de la escala radial puede ser definida correctamente.
<b>Evaluación de la prueba:</b> Una vez realizada la primera iteración se detectaron varias no conformidades, a las cuales se les da solución y se realiza otra iteración que arrojó una evaluación satisfactoria.

Tabla 38 Prueba de aceptación #14

Prueba de Aceptación	
Número: 14	Historia de usuario: 14
<b>Nombre:</b> Definir ángulo inicial y final de la escala radial.	
<b>Descripción:</b> El sistema debe proporcionar que el usuario defina el ángulo inicial de la escala radial.	
<b>Condiciones de Ejecución:</b> El usuario debe comprobar que pueda ser definido la propiedad “ <i>startAngleScale</i> ”.	
<b>Entradas/ Pasos de Ejecución:</b> <ul style="list-style-type: none"> <li>• Seleccionar el componente.</li> <li>• Abrir el inspector de propiedades.</li> <li>• Seleccionar opción <i>Start Angle Scale</i>.</li> <li>• Insertar el valor deseado.</li> </ul>	
<b>Resultado esperado:</b> : El valor del ángulo inicial de la escala radial puede ser definido correctamente.	
<b>Evaluación de la prueba:</b> Una vez realizada la primera iteración se detectaron varias no conformidades, a las cuales se les da solución y se realiza otra iteración que arrojó una evaluación satisfactoria.	

Tabla 39 Prueba de aceptación #15

Prueba de Aceptación	
Número: 15	Historia de usuario: 15
<b>Nombre:</b> Modificar color del rango seleccionado en el medidor.	

<b>Descripción:</b> El sistema debe proporcionar que el usuario modifique la propiedad “Ranges” del componente gráfico mediante el inspector de propiedades.
<b>Condiciones de Ejecución:</b> El usuario debe comprobar que se modifique la propiedad “Ranges” del componente gráfico mediante el inspector de propiedades definiendo un nuevo color para el mismo.
<b>Entradas/ Pasos de Ejecución:</b> <ul style="list-style-type: none"> <li>• Seleccionar el componente.</li> <li>• Abrir el inspector de propiedades.</li> <li>• Seleccionar opción Ranges</li> <li>• Escribir el nuevo color deseado.</li> </ul>
<b>Resultado esperado:</b> El color del rango puede ser cambiado por otro cualquiera.
<b>Evaluación de la prueba:</b> Prueba satisfactoria.

Tabla 40 Prueba de aceptación #16

Prueba de Aceptación	
<b>Número:</b> 16	<b>Historia de usuario:</b> 16
<b>Nombre:</b> Configurar opacidad del componente gráfico.	
<b>Descripción:</b> El sistema debe proporcionar que el usuario modifique la propiedad “opacidad” del componente gráfico mediante el inspector de propiedades.	
<b>Condiciones de Ejecución:</b> El usuario debe comprobar que se modifique la propiedad de opacidad para el componente.	
<b>Entradas/ Pasos de Ejecución:</b> <ul style="list-style-type: none"> <li>• Seleccionar el componente.</li> <li>• Abrir el inspector de propiedades.</li> <li>• Seleccionar opción de opacidad.</li> </ul>	
<b>Resultado esperado:</b> La opción de opacidad para el componente es configurable.	
<b>Evaluación de la prueba:</b> Una vez realizada la primera iteración no se detectaron no conformidades, por lo cual no fue necesario otra iteración.	

Tabla 41 Prueba de aceptación #17

Prueba de Aceptación	
<b>Número:</b> 17	<b>Historia de usuario:</b> 17
<b>Nombre:</b> Adicionar rango de la escala en el medidor.	

<b>Descripción:</b> El sistema debe proporcionar que el usuario adicione un nuevo rango en la propiedad "rangos" del componente gráfico mediante el inspector de propiedades.
<b>Condiciones de Ejecución:</b>
<b>Entradas/ Pasos de Ejecución:</b> <ul style="list-style-type: none"> <li>• Seleccionar el componente.</li> <li>• Abrir el inspector de propiedades.</li> <li>• Seleccionar opción <i>Ranges</i> y añadir el mínimo, máximo y color asociados al rango que desea añadir.</li> </ul>
<b>Resultado esperado:</b> Los <i>Ranges</i> para el componente son insertados correctamente.
<b>Evaluación de la prueba:</b> Una vez realizada la primera iteración se detectaron no conformidades, a las cuales se les da solución y se realiza otra iteración que arrojó una evaluación satisfactoria.

Tabla 42 Prueba de aceptación #18

Prueba de Aceptación	
<b>Número:</b> 18	<b>Historia de usuario:</b> 18
<b>Nombre:</b> Mostrar rango de la escala en el medidor.	
<b>Descripción:</b> El sistema debe proporcionar mostrar rango de la escala en el medidor.	
<b>Condiciones de Ejecución:</b> El usuario debe comprobar que se muestre el rango de la escala en el medidor.	
<b>Entradas/ Pasos de Ejecución:</b> <ul style="list-style-type: none"> <li>• Seleccionar el componente.</li> <li>• Abrir el inspector de propiedades.</li> <li>• Seleccionar opción <i>Ranges</i> y añadir el mínimo, máximo y color asociados al rango que desea visualizar.</li> <li>• Dar clic en el inspector de propiedades en cualquier otra propiedad.</li> </ul>	
<b>Resultado esperado:</b> La opción <i>Ranges</i> funciona correctamente, por lo que visualiza los mismos sin problemas.	
<b>Evaluación de la prueba:</b> Una vez realizada la primera iteración no se detectaron no conformidades, por lo cual no fue necesario otra iteración.	

Tabla 43 Prueba de aceptación #19

Prueba de Aceptación	
<b>Número:</b> 19	<b>Historia de usuario:</b> 19
<b>Nombre:</b> Eliminar rango de la escala en el medidor.	
<b>Descripción:</b> El sistema debe proporcionar que el usuario elimine el rango deseado en la propiedad "rangos" del componente gráfico mediante el inspector de propiedades.	
<b>Condiciones de Ejecución:</b> : El usuario debe comprobar que se elimine el rango de la escala en el medidor.	

**Entradas/ Pasos de Ejecución:**

- Seleccionar el componente.
- Abrir el inspector de propiedades.
- Seleccionar opción *Ranges* y presionar *delete* o *backspace* en el teclado sobre el rango definido.

**Resultado esperado:** La opción *Ranges* funciona correctamente, por lo que visualiza los mismos sin problemas.

**Evaluación de la prueba:** Una vez realizada la primera iteración no se detectaron no conformidades, por lo cual no fue necesario otra iteración.

Anexo 3 Diseños de casos de pruebas

SC2 Definir valor mínimo de la escala del medidor.				
<b>Descripción general</b>				
El sistema debe proporcionar que el usuario defina el valor mínimo de la escala del medidor.				
Escenario	Descripción	Variable <i>Min Value</i>	Respuesta del sistema	Flujo central
EC 2.1 Definir valor mínimo de la escala del medidor correctamente.	Debe proporcionar que el usuarios defina el valor mínimo de la escala del medidor.	(m_minValue<m_maxValue && m_minValue> =0)  0	Introduce el valor mínimo deseado.	Siguiendo la siguiente ruta: 1-Se selecciona el componente con un clic. 2- Se abre el inspector de propiedades. 3- Se selecciona la opción <i>minValue</i> . 4-Introducir el valor deseado como valor mínimo.
EC 2.2 Definir valor mínimo de la escala del medidor/ valor incorrecto.	Se introduce un valor mínimo incorrecto.	-0.1	Se mantiene el valor mínimo definido inicialmente.	Siguiendo la siguiente ruta: 1-Se selecciona el componente con un clic. 2- Se abre el inspector de propiedades. 3- Se selecciona la opción <i>minValue</i> .

## Anexo 3 Diseños de casos de pruebas

				4-Introducir el valor incorrecto como valor mínimo.
SC3 Definir valor máximo de la escala del medidor.				
<b>Descripción general</b>				
El sistema debe proporcionar que el usuario defina el valor máximo de la escala del medidor.				
<b>Escenario</b>	<b>Descripción</b>	<b>Variable Max Value</b>	<b>Respuesta del sistema</b>	<b>Flujo central</b>
EC 3.1 Definir valor máximo de la escala del medidor.	Debe proporcionar que el usuarios defina el valor máximo de la escala del medidor.	$(m\_maxValue > m\_minValue)$ $100 > 0$	Introduce el valor máximo deseado.	<p>Siguiendo la siguiente ruta:</p> <ol style="list-style-type: none"> <li>1- Se selecciona el componente con un clic.</li> <li>2- Se abre el inspector de propiedades.</li> <li>3- Se selecciona la opción <i>maxValue</i>.</li> <li>4- Introducir el valor deseado como valor máximo.</li> </ol>
EC 3.2 Definir valor máximo de la escala del medidor/ valor incorrecto.	Se introduce un valor máximo incorrecto.	$(m\_maxValue > m\_minValue)$ $-1 > 0$	Se mantiene el valor máximo definido inicialmente.	<p>Siguiendo la siguiente ruta:</p> <ol style="list-style-type: none"> <li>1- Se selecciona el componente con un clic.</li> <li>2- Se abre el inspector de propiedades.</li> </ol>

## Anexo 3 Diseños de casos de pruebas

				<p>3- Se selecciona la opción <i>maxValue</i>.</p> <p>4-Introducir el valor incorrecto como valor máximo.</p>
SC4 Modificar valor mínimo de la escala del medidor.				
<b>Descripción general</b>				
El sistema debe proporcionar que el usuario modifique la propiedad “valor mínimo” del componente gráfico mediante el inspector de propiedades.				
Escenario	Descripción	Variable <i>Min Value</i>	Respuesta del sistema	Flujo central
EC 4.1 Modificar valor mínimo de la escala del medidor.	Debe proporcionar que el usuarios defina el valor mínimo de la escala del medidor.	$(m\_minValue < m\_maxValue \ \&\& \ m\_minValue > = 0)$ $0 < 100 \ \&\& \ 0 = 0$ 0	Introduce el nuevo valor mínimo deseado.	Siguiendo la siguiente ruta: 1-Se selecciona el componente con un clic. 2- Se abre el inspector de propiedades. 3- Se selecciona la opción <i>minValue</i> . 4-Introducir el valor deseado como valor mínimo.
EC 4.2 Modificar valor mínimo de la escala del medidor / valor incorrecto.	Se introduce un valor mínimo incorrecto.	-0.1	Se mantiene el valor mínimo definido inicialmente.	Siguiendo la siguiente ruta: 1-Se selecciona el componente con un clic.

## Anexo 3 Diseños de casos de pruebas

				<p>2- Se abre el inspector de propiedades.</p> <p>3- Se selecciona la opción <i>minValue</i>.</p> <p>4-Introducir el valor incorrecto como valor mínimo.</p>
SC5 Modificar valor máximo de la escala del medidor.				
<b>Descripción general</b>				
El sistema debe proporcionar que el usuario modifique la propiedad “valor máximo” del componente gráfico mediante el inspector de propiedades.				
Escenario	Descripción	Variable <i>Max Value</i>	Respuesta del sistema	Flujo central
EC 5.1 Modificar valor máximo de la escala del medidor.	Debe proporcionar que el usuarios defina el valor máximo de la escala del medidor.	(m_maxValue>m_minValue) 100>0 100	Introduce el nuevo valor máximo deseado.	<p>Siguiendo la siguiente ruta:</p> <p>1-Se selecciona el componente con un clic.</p> <p>2- Se abre el inspector de propiedades.</p> <p>3- Se selecciona la opción <i>maxValue</i>.</p> <p>4-Introducir el valor deseado como valor máximo.</p>
EC 5.2 Modificar valor	Se introduce un valor máximo incorrecto.	-0.1	Se mantiene el valor máximo	<p>Siguiendo la siguiente ruta:</p>

## Anexo 3 Diseños de casos de pruebas

máximo de la escala del medidor / valor incorrecto.			definido inicialmente.	<p>1-Se selecciona el componente con un clic.</p> <p>2- Se abre el inspector de propiedades.</p> <p>3- Se selecciona la opción maxValue.</p> <p>4-Introducir el valor incorrecto como valor máximo.</p>
<b>Descripción general</b>				
El sistema debe proporcionar que el usuario modifique la propiedad "rangos" del componente gráfico mediante el inspector de propiedades.				
SC 6 Modificar valor mínimo del rango seleccionado en el medidor.				
<b>Escenario</b>	<b>Descripción</b>	<b>Variable Ranges</b>	<b>Respuesta del sistema</b>	<b>Flujo central</b>
EC 6.1 Modificar valor mínimo del rango seleccionado en el medidor.	Debe proporcionar que el usuarios defina el valor mínimo del rango del medidor e introducir una coma (,) para separar el próximo valor.	0	Introduce el nuevo valor mínimo del rango deseado.	<p>Siguiendo la siguiente ruta:</p> <p>1-Se selecciona el componente con un clic.</p> <p>2- Se abre el inspector de propiedades.</p> <p>3- Se selecciona la opción Ranges.</p> <p>4-Introducir el valor deseado como valor mínimo.</p>

## Anexo 3 Diseños de casos de pruebas

				5- Introducir una coma (,) para separar el próximo valor.
EC 6.2 Modificar valor mínimo del rango seleccionado en el medidor/ valor mínimo incorrecto.	Se introduce un valor mínimo incorrecto.	-0.1	Toma el ultimo valor definido como mínimo.	Siguiendo la siguiente ruta: 1-Se selecciona el componente con un clic. 2- Se abre el inspector de propiedades. 3- Se selecciona la opción <i>Ranges</i> . 4-Introducir el valor deseado como valor mínimo. 5- Introducir una coma (,) para separar el próximo valor.
<b>Descripción general</b>				
El sistema debe proporcionar que el usuario modifique la propiedad "rangos" del componente gráfico mediante el inspector de propiedades.				
SC 7 Modificar valor máximo del rango seleccionado en el medidor.				
<b>Escenario</b>	<b>Descripción</b>	<b>Variable <i>Ranges</i></b>	<b>Respuesta del sistema</b>	<b>Flujo central</b>
EC 7.1 Modificar valor máximo del	Introduce el valor deseado como valor máximo y se separa	(m_maxValue>m_minValue) 100>0 100	Introduce el valor máximo del rango deseado.	Siguiendo la siguiente ruta:

## Anexo 3 Diseños de casos de pruebas

<p>rango seleccionado en el medidor.</p>	<p>por una barra ( ) para separar el próximo valor (el color).</p>			<p>1-Se selecciona el componente con un clic.                  2- Se abre el inspector de propiedades.                  3- Se selecciona la opción <i>Ranges</i>.                  4-Introducir el valor deseado como valor máximo.                  5- Introducir una barra ( ) para separar el próximo valor (el color).</p>
<p>EC 7.2                  Modificar valor mínimo del rango seleccionado en el medidor/                  valor mínimo incorrecto.</p>	<p>Se introduce un valor mínimo incorrecto.</p>	<p>-0.1</p>	<p>Toma el ultimo valor definido como mínimo.</p>	<p>Siguiendo la siguiente ruta:                  1-Se selecciona el componente con un clic.                  2- Se abre el inspector de propiedades.                  3- Se selecciona la opción <i>Ranges</i>.                  4-Introducir el valor deseado como valor mínimo.                  5- Introducir una coma (,) para separar el próximo valor.</p>
<p><b>Descripción general</b></p>				

## Anexo 3 Diseños de casos de pruebas

El sistema debe proporcionar que el usuario modifique la propiedad "color de escala" del componente gráfico mediante el inspector de propiedades.				
SC 8 Modificar color de la escala del medidor.				
Escenario	Descripción	Variable Color Scale	Respuesta del sistema	Flujo central
EC 8.1 Modificar color de la escala del medidor.	Se intenta seleccionar color deseado para la escala.	<i>blue</i>	Introduce el color deseado para la escala.	Siguiendo la siguiente ruta: 1-Se selecciona el componente con un clic. 2- Se abre el inspector de propiedades. 3- Se selecciona la opción <i>Color Scale</i> 4- Introducir el color deseado para la escala.
<b>Descripción general</b>				
El sistema debe proporcionar que el usuario modifique la propiedad " <i>Color Background</i> " del componente gráfico mediante el inspector de propiedades.				
SC 9 Modificar color de fondo del medidor.				
Escenario	Descripción	Variable Color Background	Respuesta del sistema	Flujo central
EC 9.1 Modificar color de fondo del medidor.	Se intenta seleccionar el color deseado para fondo del medidor.	<i>pink</i>	Toma el nuevo color deseado para el fondo del medidor.	Siguiendo la siguiente ruta:

## Anexo 3 Diseños de casos de pruebas

				<p>1-Se selecciona el componente con un clic.</p> <p>2- Se abre el inspector de propiedades.</p> <p>3- Se selecciona la opción <i>Color Background</i>.</p> <p>4- Introducir el color deseado para el fondo del medidor.</p>
<b>Descripción general</b>				
El sistema debe proporcionar que el usuario modifique la propiedad " <i>Color Needle</i> " del componente gráfico mediante el inspector de propiedades.				
SC 10 Definir color de fondo de la aguja.				
<b>Escenario</b>	<b>Descripción</b>	<b>Variable Color Needle</b>	<b>Respuesta del sistema</b>	<b>Flujo central</b>
EC 10.1 Definir color de fondo de la aguja.	Se intenta seleccionar el color deseado para la aguja.	#FAFEFA	Toma el nuevo color deseado para el fondo de la aguja.	<p>Siguiendo la siguiente ruta:</p> <p>1-Se selecciona el componente con un clic.</p> <p>2- Se abre el inspector de propiedades.</p> <p>3- Se selecciona la opción <i>Color Needle</i>.</p>

Anexo 3 Diseños de casos de pruebas

				4- Introducir el color deseado para la aguja.
<b>Descripción general</b>				
El sistema debe proporcionar que el usuario defina la unidad de medida del medidor.				
SC 11 Definir unidad de medida "Unit" del medidor.				
Escenario	Descripción	Variable Unit	Respuesta del sistema	Flujo central
EC 11.1 Definir unidad de medida "Unit" del medidor.	Se intenta introducir la unidad de medida.	%	Toma el valor de la unidad de medida.	<p>Siguiendo la siguiente ruta:</p> <p>1- Se selecciona el componente con un clic.</p> <p>2- Se abre el inspector de propiedades.</p> <p>3- Se selecciona la opción <i>Unit</i>.</p> <p>4- Introducir la unidad de medida.</p>
<b>Descripción general</b>				
El sistema debe proporcionar que el usuario modifique la propiedad "valor" del componente gráfico mediante el inspector de propiedades.				
SC 12 Modificar valor del medidor.				
Escenario	Descripción	Variable Value	Respuesta del sistema	Flujo central

Anexo 3 Diseños de casos de pruebas

EC 12.1 Modificar valor del medidor.	Se intenta introducir el valor deseado.	(m_maxValue > m_value > m_minValue) 100 > 15 > 0	Toma el nuevo valor del medidor.	Siguiendo la siguiente ruta: 1- Se selecciona el componente con un clic. 2- Se abre el inspector de propiedades. 3- Se selecciona la opción <i>Unit</i> . 4- Introducir la unidad de medida.
EC 12.1 Modificar valor del medidor/ valor incorrecto.	Se intenta introducir un valor incorrecto.	(m_maxValue > m_value > m_minValue) 100 > 120 > 0	Toma el valor máximo como nuevo valor del medidor.	Siguiendo la siguiente ruta: 1- Se selecciona el componente con un clic. 2- Se abre el inspector de propiedades. 3- Se selecciona la opción <i>Unit</i> . 4- Introducir la unidad de medida.
<b>Descripción general</b>				
El sistema debe proporcionar que el usuario defina la amplitud de la escala radial.				
SC 13 Definir la amplitud de la escala radial.				
<b>Escenario</b>	<b>Descripción</b>	<b>Variable <i>Start Angle Scale</i></b>	<b>Respuesta del sistema</b>	<b>Flujo central</b>

*Anexo 3 Diseños de casos de pruebas*

<p>EC 13.1 Definir la amplitud de la escala radial.</p>	<p>Se introduce el nuevo valor de la amplitud de la escala.</p>	<p>0&lt;m_startAngleScale&lt; 359.9 320</p>	<p>Toma el nuevo valor de la amplitud de la escala.</p>	<p>Siguiendo la siguiente ruta: 1-Se selecciona el componente con un clic. 2- Se abre el inspector de propiedades. 3- Se selecciona la opción <i>Amplitude Scale</i>. 4- Introducir el valor de la amplitud de la escala.</p>
<p>EC 13.2 Definir la amplitud de la escala radial/ valor incorrecto.</p>	<p>Se introduce un valor incorrecto para la amplitud de la escala.</p>	<p>0&lt;m_startAngleScale&lt; 359.9 370</p>	<p>Toma el último valor definido de la amplitud de la escala.</p>	<p>Siguiendo la siguiente ruta: 1-Se selecciona el componente con un clic. 2- Se abre el inspector de propiedades. 3- Se selecciona la opción <i>Amplitude Scale</i>. 4- Introducir el valor de la amplitud de la escala.</p>
<p><b>Descripción general</b></p>				
<p>El sistema debe proporcionar que el usuario defina el ángulo inicial de la escala radial.</p>				
<p>SC 14 Definir ángulo inicial de la escala radial.</p>				

*Anexo 3 Diseños de casos de pruebas*

<b>Escenario</b>	<b>Descripción</b>	<b>Variable <i>Start Angle Scale</i></b>	<b>Respuesta del sistema</b>	<b>Flujo central</b>
EC 14.1 Definir ángulo inicial de la escala radial.	Se intenta introducir el valor del ángulo inicial.	0<m_startAngleScale< 359.9 120	Toma el valor del ángulo inicial de la escala radial.	<p>Siguiendo la siguiente ruta:</p> <p>1-Se selecciona el componente con un clic.</p> <p>2- Se abre el inspector de propiedades.</p> <p>3- Se selecciona la opción "<i>Start Angle Scale</i>".</p> <p>4- Introducir el valor del ángulo inicial.</p>
EC 14.21 Definir ángulo inicial de la escala radial / valor incorrecto.	Se introduce un valor incorrecto para el ángulo inicial.	0<m_startAngleScale< 359.9 370	Toma el último valor definido del ángulo inicial.	<p>Siguiendo la siguiente ruta:</p> <p>1-Se selecciona el componente con un clic.</p> <p>2- Se abre el inspector de propiedades.</p> <p>3- Se selecciona la opción "<i>Start Angle Scale</i>".</p> <p>4- Introducir el valor del ángulo inicial.</p>
<b>Descripción general</b>				

El sistema debe proporcionar que el usuario modifique la propiedad "rangos" del componente gráfico mediante el inspector de propiedades.				
SC 15 Modificar color del rango seleccionado en el medidor.				
Escenario	Descripción	Variable <i>Ranges</i>	Respuesta del sistema	Flujo central
EC 15.1 Modificar color del rango seleccionado en el medidor.	Se intenta introducir el color deseado para el rango del medidor.	#34A5E5	Toma el color deseado para el rango del medidor.	Siguiendo la siguiente ruta: 1-Se selecciona el componente con un clic. 2- Se abre el inspector de propiedades. 3- Se selecciona la opción " <i>Ranges</i> ". 4- Introducir el color deseado para el rango del medidor.
<b>Descripción general</b>				
El sistema debe proporcionar que el usuario modifique la propiedad "opacidad" del componente gráfico mediante el inspector de propiedades.				
SC 16 Configurar opacidad del componente gráfico.				
Escenario	Descripción	Variable <i>opacity</i>	Respuesta del sistema	Flujo central
EC 16.1 Configurar opacidad del	Se intenta introducir el valor de la opacidad del medidor.	m_opacity >= 0 && m_opacity <= 100 15	Toma el valor de la opacidad del medidor.	Siguiendo la siguiente ruta:

## Anexo 3 Diseños de casos de pruebas

componente gráfico.				<p>1-Se selecciona el componente con un clic.</p> <p>2- Se abre el inspector de propiedades.</p> <p>3- Se selecciona la opción “<i>Start Angle Scale</i>”.</p> <p>4- Introducir el valor de la opacidad del medidor.</p>
<b>Descripción general</b>				
El sistema debe proporcionar que el usuario adicione un nuevo rango en la propiedad “rangos” del componente gráfico mediante el inspector de propiedades.				
SC 17 Adicionar rango de la escala en el medidor.				
<b>Escenario</b>	<b>Descripción</b>	<b>Variable Ranges</b>	<b>Respuesta del sistema</b>	<b>Flujo central</b>
EC 17.1 Adicionar rango de la escala en el medidor.	Se introducen los valores máximos, mínimos el color separados por una coma (,).	15,25,blue	Toma los valores mínimos, máximos y color definidos.	<p>Siguiendo la siguiente ruta:</p> <p>1-Se selecciona el componente con un clic.</p> <p>2- Se abre el inspector de propiedades.</p> <p>3- Se selecciona la opción “<i>Start Angle Scale</i>”.</p>

### Anexo 3 Diseños de casos de pruebas

				4- Introducir los valores máximos, mínimos y color para el nuevo rango separando los mismos por la barra ().
<b>Descripción general</b>				
El sistema debe proporcionar mostrar rango de la escala en el medidor.				
SC 18 Mostrar rango de la escala en el medidor.				
Escenario	Descripción	Variable	Respuesta del sistema	Flujo central
EC 18.1 Mostrar rango de la escala en el medidor.	Se da clic en el despliegue.	N/A	Se actualiza el rango definido.	<p>Siguiendo la siguiente ruta:</p> <p>1- Se selecciona el componente con un clic.</p> <p>2- Se abre el inspector de propiedades.</p> <p>3- Se selecciona la opción "Start Angle Scale".</p> <p>4- Introducir los valores máximos, mínimos y color para el nuevo rango separando los mismos por la barra ().</p>

### Anexo 3 Diseños de casos de pruebas

				5- Dar clic en el despliegue.
<b>Descripción general</b>				
El sistema debe proporcionar que el usuario elimine el rango deseado en la propiedad "rangos" del componente gráfico mediante el inspector de propiedades.				
SC 19 Eliminar rango de la escala en el medidor.				
<b>Escenario</b>	<b>Descripción</b>	<b>Variable</b>	<b>Respuesta del sistema</b>	<b>Flujo central</b>
EC 19.1 Eliminar rango de la escala en el medidor.	Se intenta borrar ( <i>Backspace</i> ) el rango deseado y se da clic en el despliegue.	N/A	Se borra ( <i>Backspace</i> ) el rango deseado.	<p>Siguiendo la siguiente ruta:</p> <p>1- Se selecciona el componente con un clic.</p> <p>2- Se abre el inspector de propiedades.</p> <p>3- Se selecciona la opción "<i>Start Angle Scale</i>".</p> <p>4- Borrar (<i>Backspace</i>) el rango deseado.</p> <p>5- Dar clic en el despliegue.</p>