

**Universidad de las Ciencias Informáticas**

**Facultad 9**



**Título: Diseño y aplicación de pruebas al  
producto Simulador Termo Azúcar  
versión 4.0.**

**TRABAJO DE DIPLOMA PARA OPTAR POR EL TÍTULO DE  
INGENIERO EN CIENCIAS INFORMÁTICAS.**

**Autor(a):** Maylen Pérez Márquez.

**Tutor (es):** Ing. Abdel Alonso Martínez

**Cotutor:** Ing. Yurisnel Corrales Valdés

**La Habana, Junio del 2009.**

**DECLARACIÓN DE AUTORÍA**

Declaro que soy la única autora del trabajo titulado: Diseño y aplicación de pruebas al producto Simulador Termo Azúcar versión 4.0 y autorizo a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmo la presente a los \_\_ días del mes de \_\_\_\_\_ del año 2009.

\_\_\_\_\_  
Maylén Pérez Márquez

\_\_\_\_\_  
Ing.Abdel Alonso Martinez

### DATOS DE CONTACTO

**Tutor:**


**Ing. Abdel Alonso Martínez:** Ingeniero en Ciencias Informáticas recién graduado. Posee 2 años de experiencia en el desarrollo de software, donde ha desempeñado los roles de programador y diseñador. Actualmente se desempeña como profesor de Máquinas Computadoras en la Facultad 7. Forma parte del Proyecto Atención Primaria de Salud.

e-mail: [aamartinez@uci.cu](mailto:aamartinez@uci.cu).

**Cotutor:**

**Ing. Yurisnel Corrales Valdés:** Ingeniero en Ciencias Informáticas Instructor recién graduado. Posee 2 años de experiencia en el desarrollo de software, donde a desempeñado los roles de programador y diseñador. Actualmente se desempeña como profesor de Física en la Facultad 9. Forma parte del Proyecto de Simulación.

e-mail: [yvaldes@uci.cu](mailto:yvaldes@uci.cu).



*El hombre debe transformarse al mismo tiempo que la producción progresa; no realizaríamos una tarea adecuada si fuéramos tan sólo productores de artículos, de materias primas y no fuéramos al mismo tiempo productores de hombres.*

*Pero la juventud tiene que crear. Una juventud que no crea es una anomalía realmente.”*

*...aquí está una de las tareas de la juventud: empujar, dirigir con el ejemplo la producción del hombre de mañana. Y en esta producción, en esta dirección, está comprendida la producción de sí mismos...*

*Che*

### AGRADECIMIENTOS

*La culminación de este trabajo a pesar del esfuerzo realizado no hubiese sido posible sin la colaboración de un grupo de personas que emplearon parte de su tiempo en lograr que esta investigación tuviese un resultado exitoso y que el esfuerzo realizado no fuera en vano. Es por eso que quisiera aprovechar esta sección para demostrarles a todos ellos mi agradecimiento infinito por la ayuda prestada.*

*Llegue este profundo agradecimiento*

*A toda mi familia, especialmente a mi mamá por guiarme por el buen camino, brindarme su apoyo incondicional y por hacer que el camino a transitar fuera un poco más fácil.*

*A Abde por ser paciente y estar ahí cuando lo necesitaba.*

*A Alicia y Rodolfo por preocuparse como iba el desarrollo del trabajo de Diploma.*

*A todos mis maestros y profesores que aportaron su pequeño granito de arena para formar una profesional para el bien de la sociedad.*

*En fin, para no olvidar a nadie...*

*A todos muchas gracias!!!*

DEDICATORIA

*Dedico este trabajo*

*A mi mamá Marcia que siempre me ha apoyado en los momentos más difíciles,  
alentándome a seguir adelante.*

*A mi hermanito Amaurito que lo quiero mucho.*

*A Amaury que ha sido como un padre para mí.*

*A mi abuela Mirtha, a mis tías, tíos y primos que siempre han estado ahí  
cuando los he necesitado.*

*A Abde que siempre a estado cuando lo he necesitado y que me ha apoyado  
siempre.*

*A todos los que confiaron en mí.*

## **RESUMEN**

El presente trabajo tiene como objetivo, el diseño y aplicación de pruebas para el Simulador Termo Azúcar (STA) con la intención de detectar los posibles defectos que pueda tener el producto, y así lograr que este cuente con la calidad requerida para la industria química cubana.

Para obtener buenos resultados en la realización de las pruebas, se analizó los diferentes modelos de calidad, como es el Modelo de Madurez de la Capacidad (CMMI), las normas vigentes según la Organización Internacional para la Estandarización (ISO), así como el flujo de trabajo de prueba definido por el Proceso de Desarrollo Unificado (RUP). Además se analizó el estado de la calidad del producto para lograr la definición y aplicación de un conjunto de pruebas, que cumplan con las normas establecidas para obtener un resultado concreto en cuanto al estado de la calidad del software mencionado.

El desarrollo de los casos de prueba permitirá definir el estado actual de la calidad del producto STA, teniendo en cuenta que se encuentra en su versión de desarrollo 4.0 y no se le han aplicado pruebas de calidad en momentos precedentes. Además de brindarle al equipo de desarrollo un grupo de casos de prueba que puedan ser aplicadas a otros productos.

**Palabras Claves:** Simulador Termo Azúcar (STA), Pruebas de Calidad, Modelo de Madurez de la Capacidad (CMMI), Organización Internacional para la Estandarización (I

## TABLA DE CONTENIDOS

<b>AGRADECIMIENTOS .....</b>	<b>5</b>
<b>DEDICATORIA.....</b>	<b>6</b>
<b>INTRODUCCIÓN.....</b>	<b>11</b>
<b>CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA.....</b>	<b>16</b>
1.1. Introducción .....	16
1.2. Calidad del Software .....	16
1.3. Modelos de Calidad .....	20
1.4. Definición de Prueba de Software .....	20
1.5. Flujo de trabajo de prueba desde el punto de vista dinámico .....	22
1.6. Roles, Actividades y Artefactos.....	24
1.6.1. Roles .....	24
1.6.2. Actividades .....	25
1.6.3. Artefactos.....	26
1.7. Principios de la prueba.....	27
1.8. Tipos de pruebas .....	28
1.8.1. Prueba de Caja Negra .....	29
1.8.2. Prueba de Caja Blanca .....	30
1.9. Niveles de Prueba.....	34
1.9.1. Prueba de Desarrollador .....	34
1.9.2. Prueba Independiente.....	34
1.9.3. Prueba de Unidad .....	35
1.9.4. Prueba de unidad en el contexto OO (Orientado a Objetos) .....	35
1.9.5. Prueba de Integración.....	35
1.9.6. Prueba de Sistema.....	35
1.9.7. Prueba de Aceptación del Usuario .....	36
1.10. Tipos de Prueba basados en dimensiones de la calidad.....	36
1.10.1. Funcionalidad.....	37



1.10.2. Usabilidad .....	37
1.10.3. Fiabilidad .....	37
1.10.4. Performance (Rendimiento) .....	37
1.10.5. Soportabilidad .....	38
1.11. Conclusiones .....	38
<b>CAPÍTULO 2. PLAN DE PRUEBA. ....</b>	<b>39</b>
2.1. Introducción .....	39
2.2. Plan de Prueba para el STA.....	39
2.2.1. Introducción .....	39
2.2.2. Propósito.....	39
2.2.3. Alcance .....	40
2.2.4. Estrategia de evolución del Plan .....	40
2.2.5. Organización del Equipo de Prueba .....	40
2.2.6. Especificaciones de Software y Hardware .....	41
2.2.7. Descripción del Plan de Pruebas .....	42
2.2. 8. Estrategia de Prueba .....	44
2.2.9. Tipos de Pruebas a realizar .....	46
2.3. Entorno de Prueba .....	48
2.4. Recursos Requeridos.....	49
2.5. Conclusiones .....	51
<b>CAPÍTULO 3. DISEÑO, EJECUCIÓN Y EVALUACIÓN DE LAS PRUEBAS.....</b>	<b>52</b>
3.1. Introducción .....	52
3.2. Definición de los casos de pruebas de Caja Negra .....	52
3.2.1. Caso de prueba Identificar Corriente.....	52
3.2.2. Caso de prueba Identificar Módulo.....	54
3.2.3. Caso de prueba Conversor de unidades .....	57
3.2.4. Caso de prueba Reportar datos .....	58
3.2.5. Caso de prueba Reportar resultados .....	60
3.3. Prueba de Documentación.....	62
3.3.1. Registro No Conformidades para la Ayuda del STA.....	63
3.3.2. Registro No Conformidades para el Manual de Usuario del STA .....	73

---

3.3.3. Carpeta del Proyecto .....	73
3.4. Definición de los casos de pruebas de Caja Blanca .....	74
3.4.1. Grafos de Flujo para MostrarVentana .....	74
3.4.2. Grafos de Flujo para btnAceptarClick.....	76
3.5. Resumen de Evaluación .....	77
3.6. Sugerencias.....	78
3.7. Evaluación de las actividades implementadas anteriormente.....	78
3.8. Conclusiones .....	78
<b>CONCLUSIONES.....</b>	<b>80</b>
<b>RECOMENDACIONES.....</b>	<b>81</b>
<b>BIBLIOGRAFÍA.....</b>	<b>82</b>
<b>ANEXOS .....</b>	<b>86</b>
<b>GLOSARIO DE TÉRMINOS.....</b>	<b>104</b>

### INTRODUCCIÓN

La informatización para la sociedad cubana tiene una vital importancia, podríamos mencionar el derecho que tienen los pueblos a desarrollarse, comunicarse y mostrar a la comunidad internacional sus avances, logros y características propias de su cultura, capacidad que nos brinda la red de redes (Internet) y que solo es posible utilizar a totalidad con un desarrollo fuerte y sostenido de la tecnologías de la información y las comunicaciones (TIC) de manera que estas estén disponibles para todos los sectores sociales existentes en una región determinada, en este caso Cuba.

El gobierno revolucionario cubano a partir del año 2002 le dio inicio a un proyecto que intenta preparar las bases para masificar el conocimiento de la informática en la sociedad actual, con el objetivo de crear software que reporten beneficios económicos y sociales entre otras tareas. Conocido como Universidad de las Ciencias Informáticas (UCI) este proyecto gradúa profesionales preparados en esta materia desde el año 2007.

La formación de estos profesionales abarca desde una correcta preparación política ideológica hasta un profundo dominio de las diferentes etapas de la creación de software. Vinculados desde etapas tempranas de su formación como ingenieros a proyectos productivos deben transitar por los diferentes roles definidos en cada uno de ellos.

Como parte del proceso de informatización del país se están desarrollando software que vienen a dar solución a algunos problemas en el campo de la industria química cubana. Específicamente nos ocupa el Simulador Termo Azúcar (STA) desarrollado en su versión 4.0.3 por la facultad 9 de la UCI; que simula el proceso de producción de azúcar crudo y refino, mostrando los resultados de la misma, brindando análisis económicos, de sensibilidad, de dependencia y reportes de condensado e indicadores, este último por área, equipo o global.

Cualquier producto de software desarrollado debe ser probado, evaluado por sus desarrolla

dores y un equipo dedicado a garantizar su calidad dentro de la institución que lo desarrolla antes de desplegarse en manos del cliente. Este tema es de vital importancia, pues crea una especie de filtros por los que pasa el producto antes de declararse como terminado y que tributa a mejorar el prestigio de la institución en cuestión, pues sus productos una vez puestos en el mercado tienen pocas posibilidades de mostrar errores en su funcionamiento; independientemente que se le asegure o no su posterior mantenimiento.

Teniendo en cuenta estas ideas la **Situación Problemática** viene dada a que el STA ha transitado por sus diferentes fases de desarrollo, sin tener en cuenta en su total magnitud los beneficios que pueden aportar llevar a la par un proceso de calidad que chequee y garantice la fiabilidad de sus funcionalidades, la limpieza y reutilización de su código.

Tomando como premisa la infraestructura existente en el país y lo anteriormente planteado, el **Problema a Resolver** es: ¿Cómo realizar un proceso de pruebas para el Simulador Termo Azúcar que permita medir la calidad del producto y detectar los posibles errores en la construcción del mismo?

Para lograr este propósito se identificó como **Objeto de Estudio** el Proceso de pruebas de software para garantizar la calidad del STA.

El **Campo de Acción** de esta investigación está centrado en las pruebas de calidad del producto STA.

Como **Objetivo General** se ha propuesto, diseñar y aplicar pruebas de software al producto Simulador Termo Azúcar que permita la validación de la calidad del sistema.

Como **Idea a Defender** se planteó la siguiente: Si se lleva a cabo un adecuado proceso de pruebas en el producto STA entonces se podrá detectar la mayor cantidad de defectos posibles.

Para dar cumplimiento a todo lo anteriormente planteado se definieron las siguientes

### **Tareas de la Investigación:**

- Analizar las metodologías y tendencias existentes que permitan establecer el estado del arte en el desarrollo de pruebas de software.
- Analizar las plantillas definidas por la UCI y RUP para las pruebas.
- Seleccionar la estrategia más idónea de acuerdo a las características del proceso productivo de la UCI y a las características del producto.
- Diseñar pruebas de caja blanca y caja negra para el STA.
- Aplicar pruebas de caja blanca y caja negra para el STA.
- Comparar los resultados esperados y reales que se obtienen de la ejecución de las pruebas.

Para darle cumplimiento a las tareas propuestas se utilizaron diferentes métodos de investigación que posibilitaron conseguir resultados favorables, estos fueron:

### **Empíricos:**

- **La entrevista:** La entrevista es una conversación planificada entre el investigador y el entrevistado para obtener información. Su uso constituye un medio para el conocimiento cualitativo de los fenómenos.<sup>1</sup>
- **La observación:** La observación científica es la percepción planificada dirigida a un fin y relativamente prolongada de un hecho o fenómeno. Es el instrumento universal del científico, se realiza de forma consciente y orientada a un objetivo determinado.<sup>2</sup>

---

<sup>1</sup> Hernandez, León, Rolando Alfredo y Coello, González, Sayda. *EL PARADIGMA CUANTITATIVO DE LA INVESTIGACIÓN CIENTÍFICA [digital]*. ISBN: 959-16-0343-6 Ciudad de la Habana, Noviembre del 2002.

<sup>2</sup> Ídem referencia 1

## Teóricos:

- **Análisis y Síntesis:** Análisis, permite la división mental del fenómeno en sus múltiples relaciones y componentes para facilitar su estudio. Síntesis, establece mentalmente la unión entre las partes previamente analizadas, posibilita descubrir sus características generales y las relaciones esenciales entre ellas.<sup>3</sup>
- **Análisis histórico y el lógico:** Histórico (tendencial), está vinculado al conocimiento de las distintas etapas de los objetos en su sucesión cronológica; para conocer la evolución y desarrollo del objeto o fenómeno de investigación se hace necesario revelar su historia, las etapas principales de su desenvolvimiento y las conexiones históricas fundamentales. Mediante este método se analiza la trayectoria concreta de la teoría, su condicionamiento a los diferentes períodos de la historia. Lógico investigan las leyes generales y esenciales del funcionamiento y desarrollo de los fenómenos. Lo lógico reproduce en el plano teórico, lo más importante del fenómeno histórico lo que constituye su esencia.<sup>4</sup>

Después de aplicar los métodos científicos analizados y dándole cumplimiento a las tareas de la investigación se esperan obtener los siguientes **Resultados:**

1. Definir una serie de actividades del flujo de trabajo de prueba para el STA.
2. Obtener resultados de las pruebas realizadas al STA.
3. Evaluación de los resultados.
4. Evaluación de las actividades implementadas anteriormente.

El trabajo está estructurado en 3 capítulos:

## Capítulo

---

<sup>3</sup> Ídem referencia 1

<sup>4</sup> Hernández, Meléndrez, Dra. Edelsys. *Cómo Escribir una Tesis.* [digital]. Escuela Nacional de Salud Pública, Cuba. Mayo, 2007

Se refiere a conceptos fundamentales de acuerdo a las problemáticas planteadas, haciendo un estudio de los principales aspectos relacionados al aseguramiento de la calidad. Además del estudio de los niveles y métodos más importantes de pruebas.

## **Capítulo II**

Muestra todas las actividades del Plan de Prueba incluyendo la Estrategia de Prueba, expone los tipos de pruebas que se le realizaron al STA, sus objetivos, las técnicas aplicadas para la realización de las mismas así como los recursos necesarios para llevar a cabo un buen proceso de detección de defectos.

## **Capítulo III**

Se realiza el diseño y ejecución de las pruebas definidas en el Plan de pruebas, para hacer una evaluación de todos los defectos y dificultades encontrados en las funcionalidades escogidas; de esta manera se logra una mayor calidad en el producto y que este cuente con los menores defectos posibles.

## CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

### 1.1. Introducción

Este Capítulo menciona algunos conceptos importantes relacionados con la calidad de software, describe la ejecución del flujo de trabajo de prueba, roles, actividades y artefactos generados por las mismas. Además de los diferentes niveles de prueba que existen así como los métodos más utilizados.

### 1.2. Calidad del Software

#### Concepto de calidad del Software

La calidad de software es un “Conjunto de características de una entidad que le confieren su aptitud para satisfacer las necesidades expresadas y las implícitas<sup>5</sup>.”

Según otra bibliografía consultada “la calidad de software es la concordancia con los requisitos funcionales y de rendimiento explícitamente establecidos, con los estándares de desarrollo explícitamente documentados, y con las características implícitas que se espera de todo software desarrollado profesionalmente<sup>6</sup>.”

En general la calidad de software es el conjunto de cualidades que lo caracterizan y que determinan su utilidad y existencia. La calidad es sinónimo de eficiencia, flexibilidad, corrección, confiabilidad, mantenibilidad, portabilidad, usabilidad, seguridad e integridad.

#### Aseguramiento de la Calidad

Son todas aquellas acciones planificadas y sistemáticas necesarias para proporcionar la confianza adecuada de que un producto o servicio satisface los requerimientos de calidad establecidos.<sup>7</sup>

---

<sup>5</sup> (ISO 8402: 1984)

<sup>6</sup> Pressman, Roger. 2002. Ingeniería de Software. Un enfoque Práctico. 5ta Edición. s.l.: McGraw-Hill, 2002.

<sup>7</sup> Peña, Carlos Martínez. 2002. Departamento de Ciencias Matemáticas e Informáticas. [En línea]



El aseguramiento de la calidad se puede observar como una herramienta de la gestión de la calidad. Su objetivo es informar los datos necesarios sobre la calidad del producto, adquiriéndose una visión más profunda y segura de que el proceso de calidad del mismo se está realizando correctamente.

### **Plan de Calidad**

Es un documento que establece las prácticas específicas de calidad, recursos y secuencia de actividades relativas a un producto, servicio, contrato o proyecto en particular.<sup>8</sup>

EN RUP se define un Plan de Aseguramiento de la Calidad que contiene un plan de revisión, auditoría y otros artefactos, desarrollados en la fase de inicio que evidencian como la calidad del producto, artefactos y procesos serán aseguradas.

### **Gestión de la Calidad**

Es aquel aspecto de función general de la gestión de una organización que define y aplica la política de calidad. La gestión de la calidad incluye la planificación, asignaciones de recursos y otras actividades sistemáticas, tales como los planes de calidad<sup>9</sup>.

### **Control de la Calidad**

Técnicas y actividades de carácter operativo utilizadas para satisfacer los requisitos relativos a la calidad, centradas en dos objetivos fundamentales: mantener bajo control un proceso y eliminar las causas de defectos en las diferentes fases del ciclo de vida<sup>10</sup>.

El control de la calidad es una serie de inspecciones, revisiones y pruebas utilizadas a lo largo del proceso de desarrollo del software para asegurar que cada producto cumpla con los requisitos que le han sido asignados.

### **Política de la Calidad**

---

<sup>8</sup> Autores, Colectivo de. 2004. Departamento de Educación y Cultura. SCIENCE

<sup>9</sup> Ídem a la referencia 6.

<sup>10</sup> Autores, Colectivo de. 1999. eumet.net. [En línea] 1999

Se define como política de calidad a las Intenciones globales y orientación de una organización relativas a la calidad tal como se expresan formalmente por la alta dirección<sup>11</sup>

### **Garantía de la Calidad**

La garantía de la calidad consiste en la auditoría y las funciones de información de la gestión; su propósito es proporcionar la gestión para informar los datos necesarios sobre la calidad del producto, por lo que se va adquiriendo una visión más profunda y segura de que el proceso de calidad del producto está cumpliendo sus objetivos. Si los datos proporcionados mediante la garantía de calidad identifican problemas, es responsabilidad de la gestión afrontar los problemas y aplicar los recursos necesarios para resolver aspectos de calidad.

### **Auditoría de la Calidad**

Es un examen sistemático e independiente para determinar si las actividades y los resultados relativos a la calidad cumplen con las disposiciones previamente establecidas, si estas se han aplicado efectivamente y son adecuadas para lograr los objetivos. Existe auditoría de calidad del proceso, del producto y de los servicios. Uno de los propósitos de una auditoría de calidad es evaluar la necesidad de acciones correctivas o mejoramiento<sup>12</sup>

### **Factores que determinan la calidad del software**

#### **✓ Operaciones del producto: características operativas**

- **Corrección:** ¿Hace lo que se le pide? : Grado en que una aplicación satisface sus especificaciones y consigue los objetivos encomendados por el cliente.
- **Fiabilidad:** ¿Lo hace de forma fiable todo el tiempo?: Grado que se puede esperar de que una aplicación lleve a cabo las operaciones especificadas y con la precisión requerida.

---

<sup>11</sup> Autores, Colectivo de. 2006. UNAD. [En línea] <http://calidad.unad.org>, 2006

<sup>12</sup> Ídem referencia 6.

- Eficiencia: ¿Qué recursos, hardware y software necesito?: Cantidad de recursos, hardware y software que necesita una aplicación para realizar las operaciones con los tiempos de respuesta adecuados.
- Integridad: ¿Se puede controlar su uso?: Grado con que puede controlarse el acceso al software o a los datos a personal no autorizado.
- Facilidad de uso: ¿Es fácil y cómodo de manejar?: Esfuerzo requerido para aprender el manejo de una aplicación, trabajar con ella, introducir datos y conseguir resultados.

### ✓ **Revisión del producto: capacidad para soportar cambios**

- Facilidad de mantenimiento: ¿Se puede localizar los fallos?: Esfuerzo requerido para localizar y reparar errores.
- Flexibilidad: ¿Se puede añadir nuevas opciones?: Esfuerzo requerido para modificar una aplicación en funcionamiento.
- Facilidad de prueba: ¿Se puede probar todas las opciones?: Esfuerzo requerido para probar una aplicación de forma que cumpla con lo especificado en los requisitos.

### ✓ **Transición del producto: adaptabilidad a nuevos entornos**

- Portabilidad: ¿Se puede usar en otra máquina?: Esfuerzo requerido para transferir la aplicación a otro hardware o sistema operativo.
- Reusabilidad: ¿Se puede utilizar alguna parte del software en otra aplicación?: Grado en que partes de una aplicación pueden utilizarse en otras aplicaciones.
- Interoperabilidad: ¿Puede comunicarse con otras aplicaciones o sistemas informáticos?: Esfuerzo necesario para comunicar la aplicación con otras aplicaciones o sistemas informáticos.<sup>13</sup>

---

<sup>13</sup> Dorado, Carlos. 2003. Ingeniería de Software Educativo. *www.fi.net.ar*. [En línea] 2003.

### 1.3. Modelos de Calidad

#### ➤ **Modelo de Madurez y Capacidad Integrado (CMMI)**

El CMMI es un modelo de calidad utilizado mundialmente en la industria del software. Los modelos de calidad son herramientas que guían a las organizaciones a la mejora continua y la competitividad, dándoles especificaciones de qué tipo de requisitos debe implementar para poder brindar productos y servicios de alto nivel.

El CMMI del Instituto de Ingeniería de Software (SEI), tiene el propósito de proveer una única guía unificada para la mejora de múltiples disciplinas tales como ingeniería de sistemas (SE), ingeniería de software (SW) y el desarrollo integrado del producto y el proceso (IPPD). Es un marco de trabajo que describe cómo evolucionar de un proceso de desarrollo de software empírico e inmaduro a un proceso efectivo, maduro y disciplinado.<sup>14</sup>

#### ➤ **Normas ISO (Organización Internacional de Normalización)**

ISO es una federación mundial de organismos nacionales de normalización, donde se prepara el trabajo de las normas internacionales mediante sus comités técnicos. Las series de ISO 9000 son un grupo de 5 individuales, pero relacionadas, estándares internacionales de administración de la calidad y aseguramiento de calidad. Estos estándares fueron desarrollados para documentar efectivamente los elementos de calidad que son instrumentados para mantener un sistema eficiente de calidad en un proyecto.<sup>15</sup>

### 1.4. Definición de Prueba de Software

Las pruebas es una actividad en la cual un sistema o componente es ejecutado bajo unas condiciones o requerimientos especificados, los resultados son observados y registrados, y una evaluación es hecha de algún aspecto del sistema o componente.<sup>16</sup>

---

<sup>14</sup> Rico, Ariel Edgar Serrano. 2003. [En línea] 2003.

<sup>15</sup> Autores, Colectivo de. 2001. [En línea] 2001.

<sup>16</sup> Ídem referencia 6

Según personas que se han dedicado a estudiar el flujo de trabajo de prueba como Kaner, Falk y Nguyen plantean que una buena prueba es aquella que cumple determinadas características:

1. Tiene una alta probabilidad de encontrar un error.
2. No debe ser redundante.
3. Debería ser “la mejor de la cosecha”.
4. No debería ser ni demasiado sencilla ni demasiado compleja.

La prueba de software es un elemento crítico para la garantía de la calidad del software y representa una revisión final de las especificaciones del diseño y codificación. Además la prueba no puede asegurar la ausencia de defectos; solo pueden demostrar que existen defectos en el software.

### **Objetivos de las pruebas**

Pressman para definir los objetivos de las pruebas hace referencia a Glem Myers quien establece algunos atributos que pueden servir para definir los objetivos de las mismas:

1. La prueba es un proceso de ejecución de un programa con la intención de descubrir un error.
2. Un buen caso de prueba es aquel que tiene una alta probabilidad de mostrar un error no descubierto hasta entonces.
3. Una prueba tiene éxito si descubre un error no detectado hasta entonces.

### **Relación de las pruebas con otras disciplinas**

- La Disciplina de Requerimientos, captura los requerimientos para el producto del software, el cual es una de las entradas principales para identificar las pruebas a ejecutar.
- La Disciplina de Análisis y Diseño, determina el diseño apropiado para el software, el cual es otra de las entradas principales para identificar que pruebas ejecutar.
- La Disciplina de Implementación produce versiones operacionales del sistema (builds) que son validados por la prueba. Dentro de una iteración, múltiples builds serán probados (1 por ciclo de prueba).

- La Disciplina de Despliegue entrega el producto de software completo al usuario final. Antes el software es validado por el flujo de prueba, aunque pruebas de aceptación y pruebas betas son ejecutadas como parte de esta disciplina.
- La Disciplina de Gestión de proyecto planifica el proyecto y las iteraciones, el de Plan de Iteración descrito, artefacto que es una importante entrada usada cuando se va a definir la misión de evaluación para la prueba.
- La configuración y gestión del cambio disciplina que controla los cambios dentro del proyecto. La prueba verifica que cada cambio ha sido completado apropiadamente.

### **1.5. Flujo de trabajo de prueba desde el punto de vista dinámico**

El flujo de trabajo de prueba se realiza durante todas las fases del proceso de desarrollo de software para ir revisando todos los elementos del mismo. En unas fases este trabajo es más amplio que en otra, pero es importante llevar a cabo un proceso de prueba durante todo el desarrollo del producto para garantizar su calidad.

#### **En la Fase de Inicio**

El grupo de prueba se pone al corriente de las características del sistema propuesto, considera que pruebas requerirá y va desarrollando algunos planes provisionales de prueba. En esta etapa no se realiza un trabajo significativo de pruebas ya que es una fase exploratoria. Se puede generar un modelo de pruebas aunque no es necesario.

#### **En la Fase de Elaboración**

El objetivo es asegurarse de que los subsistemas de todos los niveles y de todas las capas (desde la capa del sistema hasta las capas específicas de la aplicación) funcionen. Sólo se pueden probar los componentes ejecutables.

Al empezar por las capas más bajas de la arquitectura se prueban los mecanismos de distribución, almacenamiento, recuperación y concurrencias de objetos, así como otros mecanismos de las capas inferiores del sistema. Con esto no solo se prueba la funcionalidad sino también el rendimiento. Todas las capas no son necesarias de probar sino es necesario probar cómo las capas superiores hacen uso de las inferiores.

Al planificar las pruebas se seleccionan los objetivos que evaluarán la línea base de la arquitectura. Al diseñar las pruebas se toman como base estos objetivos para identificar los casos de pruebas necesarios y preparar procedimientos de pruebas para comprobar la sucesiva integración de subsistemas hasta completar la línea base. Al comprobar los componentes se quedará listo para realizar las pruebas de integración. Al ser integrado el sistema, tal y como queda definido por los casos de uso arquitectónicamente significativos se realizan las pruebas de sistema.

### **En la Fase de Construcción**

Al planificar las pruebas se seleccionan los objetivos que comprueben las sucesivas construcciones, y por último el propio sistema.

Al diseñar las pruebas se determina cómo probar los requisitos en el conjunto de construcciones. Se preparan casos y procedimientos de pruebas con este fin. Se realizan las pruebas de integración informando los resultados para tomar las medidas necesarias en casos de errores. Se realizan también pruebas del sistema al alcanzarse el status de versión parcial del sistema, informando los resultados para tomar las medidas necesarias en casos de errores. En esta fase se deben evaluar las pruebas a medida que transcurren las pruebas de integración y de sistema comprobando que éstas alcancen los objetivos del plan de pruebas. Si una prueba no alcanza sus objetivos, los casos y procedimientos de pruebas deberán ser modificados para lograrlos.

### **En la Fase de Transición**

Se pueden realizar pruebas Alfa (se realizan en la empresa que desarrolla el software con la participación del cliente en un ambiente controlado) y/o pruebas Beta (prueba realizadas en organizaciones representativas “clientes beta”) y/o validaciones por terceros (una empresa especializada en pruebas realiza pruebas de aceptación por encargo del cliente). Se recopilan y analizan los resultados de estas pruebas con el objetivo de llevar a cabo acciones.

En esta fase se buscan pequeñas deficiencias que pasaron desapercibidas durante la fase de construcción y que pueden ser corregidas en el marco de la línea base de la arquitectura existente.

### 1.6. Roles, Actividades y Artefactos

#### 1.6.1. Roles

Cada proyecto en desarrollo debe crear un grupo de prueba con roles bien definidos, independientemente de que las pruebas pueden ser realizadas por usuarios finales y otros trabajadores del equipo de desarrollo.

#### **Administrador de Prueba**

Es el responsable de dirigir técnicamente al grupo de prueba y administrar el proceso. Participa en el proceso de planificación de las pruebas siendo el encargado de la elaboración del plan, participa en el proceso de evaluación de las pruebas, siendo el responsable de documentar este proceso. Se encarga de adquirir los recursos apropiados para ejecutar el proceso. Identifica los elementos que motiven o faciliten las pruebas. Evalúa y audita la calidad del proceso proponiendo mejoras para próximas iteraciones, mantiene informado al jefe de proyecto del estado del proceso de prueba.

#### **Analista de Prueba**

Es responsable de identificar los objetivos de las pruebas así como las ideas de las pruebas. Identifica y define las pruebas requeridas en la iteración. Participa en las actividades de planificación, diseño y evaluación de las pruebas.

#### **Diseñador de prueba**

Es responsable de la integridad del modelo de pruebas asegurando que el modelo cumple con su propósito. Del modelo de pruebas define y describe los casos de prueba y los procedimientos de prueba. Realiza la gestión y mantenimiento del entorno de los datos (base de datos) de prueba. Administra la base de datos de prueba. Responsable de la elaboración de los componentes de prueba para los procedimientos que puedan ser automatizados. Participa en las actividades de planificación, diseño, implementación y evaluación de las pruebas.

#### **Probador**

Es el responsable de ejecutar los casos de prueba guiados por los procedimientos de pruebas de forma manual o automática haciendo uso de los componentes de pruebas obtenidos como resultado de la implementación de las pruebas o utilizando herramientas



de pruebas identificadas. Participa en la actividad de ejecución de las pruebas siendo responsable de identificar y listar los errores encontrados.

### 1.6.2. Actividades

De forma general se proponen cinco grandes pasos para realizar las pruebas:

✓ **Planificar las pruebas de cada iteración obteniendo un plan de prueba por iteración**

La función fundamental de esta actividad es describir la estrategia de prueba de la iteración definiendo claramente el método apropiado para ejecutarlas, los objetivos, las posibles pruebas a realizar, las pautas a seguir y el enfoque de las mismas. El objetivo de esta actividad es realizar el Plan de prueba que describe los recursos y la planificación de las mismas.

✓ **Diseñar las pruebas con la elaboración de casos de prueba, especificar cómo realizar las pruebas**

Esta actividad verifica los enfoques de las pruebas. Su objetivo es lograr una solución de implementación apropiada de cada técnica de pruebas o encontrar técnicas alternativas que puedan usarse.

Los casos de pruebas de la iteración, también se coleccionan y se administran los datos de prueba generándose los scripts de datos de prueba, así como se realizan las descripciones de los casos de pruebas, los cuales describen la forma de probar un sistema incluyendo las entradas, salidas y resultados esperados.

Los casos de prueba describen bajo qué condiciones debe probarse el sistema y ellos pueden derivarse de las descripciones de los casos de uso del diseño. Los casos de pruebas deben quedar definidos:

- a. Descripción General.
- b. Nombre de la Prueba.
- c. Descripción.
- d. Condiciones de ejecución.
- e. Entrada.
- f. Resultado esperado.
- g. Evaluación de la Prueba.

### ✓ **Implementar pruebas**

El propósito de la implementación de pruebas es automatizar uno o varios procedimientos de prueba, creando componentes de pruebas cuando sea posible. Lo que ahorraría tiempo y esfuerzo a la hora de ejecutar las pruebas. Los componentes de pruebas usan por lo general grandes cantidades de datos de entrada y generan grandes cantidades de datos de salida para visualizar estos datos en ocasiones es útil utilizar hojas de cálculo y bases de datos

### ✓ **Realizar las pruebas y documentar los resultados**

Esta actividad tiene como objetivo llevar a cabo correctamente las pruebas planificadas y diseñadas. Se van a realizar las pruebas de unidad y de sistema según los objetivos de la iteración.

Se centra en la ejecución del artefacto casos de pruebas, a partir de las descripciones de los casos de prueba, apoyándose de los componentes de pruebas que automatizan uno o varios procedimientos de prueba proporcionando entradas de pruebas, controlando y monitoreando los componentes a probar.

### ✓ **Evaluar los resultados de las pruebas realizadas**

Esta actividad se centra en evaluar los esfuerzos de las pruebas en una iteración. Para evaluar los resultados de pruebas estos deben ser comparados con los objetivos descritos en el Plan de prueba. Esta actividad plantea fundamentalmente mantener y mejorar la calidad de las pruebas, pues permite re-usar las anteriores actividades del actual ciclo de prueba, en ciclos de pruebas posteriores.

### **1.6.3. Artefactos**

Como resultado de la realización de las actividades incluidas en este flujo de trabajo, los trabajadores construyen los siguientes artefactos:

#### **Modelo de prueba**

Este modelo describe principalmente cómo se prueban los componentes ejecutables (como las construcciones) en el modelo de implementación con pruebas de integración y de sistema. Describe además cómo deben probarse aspectos específicos del sistema (ejemplo interfaz de usuario, manual del usuario, etc). El modelo de pruebas es un conjunto de casos de prueba, procedimientos de prueba y componentes de prueba.

### **Casos de prueba**

Los casos de prueba especifican la forma de probar el sistema, incluyendo la entrada o resultado con la que se ha de probar y las condiciones bajo las que ha de probarse. Es un conjunto de entradas y resultados esperados que ejercitan a un componente con el propósito de causar fallas y detectar defectos. Entre los casos de prueba pueden existir todos los tipos de relaciones, pero las más importantes son las de precedencia. Para obtener los casos de prueba hay que definir las condiciones de prueba. Las condiciones de prueba describen todas las situaciones que reflejen qué se quiere probar del sistema. Hay que priorizar las condiciones de prueba porque es posible que no se puedan probar todas o que algunas sean más críticas que otras.

### **Procedimientos de prueba**

Los procedimientos de prueba especifican cómo realizar uno o varios casos de prueba o partes de estos, por lo general define los pasos a seguir por los probadores cuando tienen que verificar las condiciones de prueba incluidas en los casos de prueba. Tienen que corresponderse con el flujo de eventos del caso de uso, pero dando valores.

### **Componentes de prueba**

Un componente de prueba automatiza uno o varios procedimientos de prueba o parte de ellos. Pueden ser programados o utilizar una herramienta de automatización de pruebas.

### **Plan de prueba**

Este describe la estrategia, recursos y cronograma de trabajo. La estrategia de prueba incluye la definición de tipos de prueba a realizar en la iteración, sus objetivos, el enfoque de las pruebas, entre otros.

### **Evaluación de prueba**

Es una evaluación de los resultados de los esfuerzos de prueba.

### **1.7. Principios de la prueba**

Antes de la aplicación de métodos para el diseño de casos de prueba efectivos, se deberá entender ciertos principios básicos que guían las pruebas del software. Se ha realizado una selección de los principios más importantes planteados por Davis [DAV95], Edward [EDW96] y Myers [MYE79]:

1. La prueba puede ser usada para mostrar la presencia de errores, pero nunca de su ausencia.
2. La principal dificultad del proceso de prueba es decidir cuando parar.
3. Evitar casos de pruebas no planificados, no reusables y triviales a menos que el programa sea verdaderamente sencillo.
4. Una parte necesaria de un caso de prueba es la definición del resultado esperado.
5. Los casos de pruebas tienen que ser escritos no solo para condiciones de entrada válidas y esperadas sino también para condiciones no válidas e inesperadas.
6. Los casos de pruebas tienen que ser escritos para generar las condiciones de salida deseadas.
7. El número de errores sin descubrir es directamente proporcional al número de errores descubiertos.
8. Los pruebas deberían empezar por " lo pequeño " y progresar hacia " lo grande".
9. Con la excepción de las pruebas de unidad e integración, un programa deberá ser probado por la persona u organización que lo desarrolló.
10. Asigna el programador más creativo a la prueba.

### **1.8. Tipos de pruebas**

Cualquier producto de ingeniería puede ser probado de una de estas formas:

1. Conociendo la funcionalidad específica para la cual fue diseñado el producto, se pueden llevar a cabo pruebas que demuestren que cada función es completamente operativa.
2. Conociendo el funcionamiento del producto se pueden desarrollar pruebas que aseguren que "todas las piezas encajen", o sea, que la operación interna se ajusta a las especificaciones y que todos los componentes internos se han comprobado de forma adecuada.

El 1er método se denomina Prueba de Caja Negra y el 2do Prueba de Caja Blanca.

### 1.8.1. Prueba de Caja Negra

Se refiere a las pruebas que se llevan a cabo sobre la interfaz del software. O sea, los casos de prueba pretenden demostrar que las funciones del software son operativas, que la entrada se acepta de forma adecuada y que se produce un resultado correcto, así como que la integridad de la información externa se mantiene.

- Verifican las especificaciones funcionales y no consideran la estructura interna del programa.
- Es hecha sin el conocimiento interno del producto.
- No validan funciones ocultas (por ejemplo funciones implementadas pero no descritas en las especificaciones funcionales del diseño) por tanto los errores asociados a ellas no serán encontrados.

La prueba de Caja Negra no es una alternativa a las técnicas de prueba de la Caja Blanca, sino un enfoque complementario que intenta descubrir diferentes tipos de errores a los encontrados en los métodos de la Caja Blanca.

#### Técnicas de Prueba de Caja Negra

1. Técnica de la Partición de Equivalencia: esta técnica divide el campo de entrada en clases de datos que tienden a ejercitar determinadas funciones del software.
2. Técnica del Análisis de valores Límites: esta técnica prueba la habilidad del programa para manejar datos que se encuentran en los límites aceptables.
3. Técnica de Grafos de Causa-Efecto: esta técnica permite al encargado de la prueba validar complejos conjuntos de acciones y condiciones.

Dentro del método de Caja Negra la técnica de la Partición de Equivalencia es una de las más efectivas, pues permite examinar los valores válidos e inválidos de las entradas existentes en el software, descubre de forma inmediata una clase de errores que, de otro modo, requerirían la ejecución de muchos casos antes de detectar el error genérico. La partición equivalente se dirige a la definición de casos de pruebas que descubran clases de errores, reduciendo así en número de clases de prueba que hay que desarrollar.

### 1.8.2. Prueba de Caja Blanca

Comprueba los caminos lógicos del software proponiendo casos de prueba que ejerciten conjuntos específicos de condiciones y/o bucles. Se puede examinar el estado del programa en varios puntos para determinar si el estado real coincide con el esperado o mencionado.

Requieren del conocimiento de la estructura interna del programa y son derivadas a partir de las especificaciones internas del diseño o el código.

Mediante los métodos de prueba de la caja blanca, el ingeniero de software puede obtener casos de prueba que garanticen que:

- 1) Se ejerciten por lo menos una vez todos los caminos independientes para cada módulo.
- 2) Se ejerciten todas las decisiones lógicas en sus vertientes verdaderas y falsa.
- 3) Ejecuten todos los bucles en sus límites y con sus límites operacionales.
- 4) Se ejerciten las estructuras internas de datos para asegurar su validez.

### Técnicas de Prueba de Caja Blanca

#### 1. Técnica del camino básico

Esta técnica, permite al diseñador de casos de prueba obtener una medida de la complejidad lógica de un diseño procedimental y usar esa medida como guía para la definición de un conjunto básico de caminos de ejecución. Los casos de prueba obtenidos del conjunto básico garantizan que durante la prueba se ejecuta por lo menos una vez cada sentencia del programa.

La idea de utilizar esta técnica consiste en derivar casos de prueba a partir de un conjunto dado de caminos independientes por los cuales puede circular el flujo de control. Para obtener dicho conjunto de caminos independientes se construye el Grafo de Flujo asociado y se calcula su complejidad ciclomática. Los pasos que se siguen para aplicar esta técnica son:

- a. A partir del diseño o del código fuente, se dibuja el grafo de flujo asociado.
- b. Se calcula la complejidad ciclomática del grafo.
- c. Se determina un conjunto básico de caminos independientes.

- d. Se preparan los casos de prueba que obliguen a la ejecución de cada camino del conjunto básico.

Los casos de prueba derivados del conjunto básico garantizan que durante la prueba se ejecuta por lo menos una vez cada sentencia del programa.<sup>17</sup>

**Notación de Grafo de Flujo**

Para aplicar la técnica del camino básico se debe introducir una sencilla notación para la representación del flujo de control, el cual puede representarse por un Grafo de Flujo.

Cada nodo del grafo corresponde a una o más sentencias de código fuente. Todo segmento de código de cualquier programa se puede traducir a un Grafo de Flujo.

Para construir el grafo se debe tener en cuenta la notación para las instrucciones: (Ver fig. 1.1.)

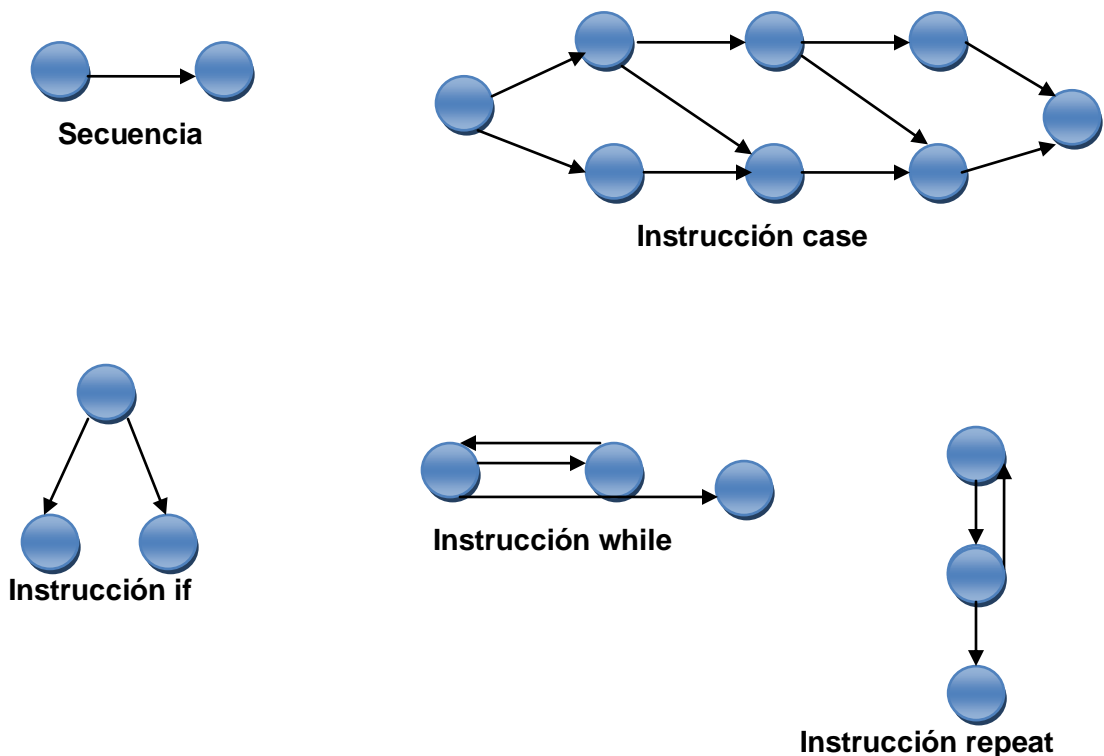


Figura 1.1. Notación del grafo de flujo para instrucciones case, if, while y repeat.

<sup>17</sup> Ídem referencia 6.

Un Grafo de Flujo está formado por 3 componentes fundamentales que ayudan a su elaboración, comprensión y brinda información para confirmar que el trabajo se está haciendo adecuadamente.

Los componentes son:

- ❖ **Nodo.**

Cada círculo representado se denomina nodo del Grafo de Flujo, el cual representa una o más secuencias procedimentales. Un solo nodo puede corresponder a una secuencia de procesos o a una sentencia de decisión. Puede ser también que hayan nodos que no se asocian, se utilizan principalmente al inicio y final del grafo.

- ❖ **Aristas.**

Las flechas del grafo se denominan aristas y representan el flujo de control, son análogas a las representadas en un diagrama de flujo. Una arista debe terminar en un nodo, incluso aunque el nodo no represente ninguna sentencia procedimental.

- ❖ **Regiones.**

Las regiones son las áreas delimitadas por las aristas y nodos. También se incluye el área exterior del grafo, contando como una región más. Las regiones se enumeran y la cantidad de regiones es equivalente a la cantidad de caminos independientes del conjunto básico de un programa. (Ver fig.1.2.)

- ❖ **Nodos Predicados:**

Nodo a partir del cual parten más de una arista, generalmente describen sentencias if.

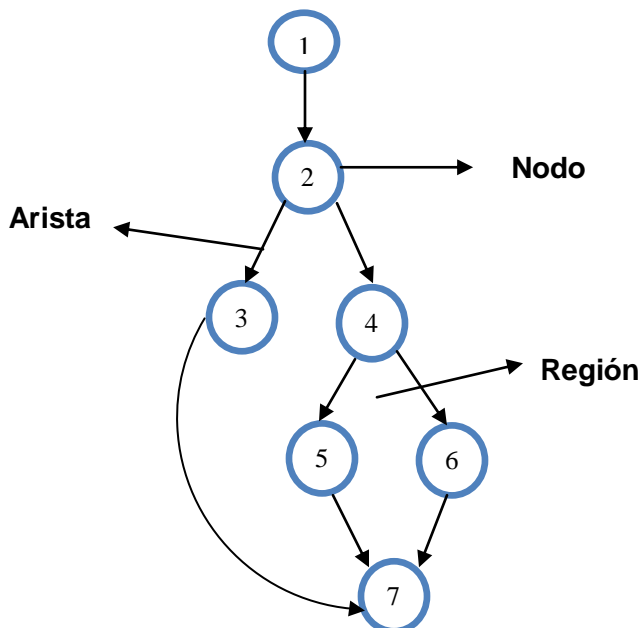




Figura 1.2. Componentes de un Grafo de Flujo.

### 🚦 Complejidad Ciclomática

La complejidad ciclomática es una métrica de software extremadamente útil pues proporciona una medición cuantitativa de la complejidad lógica de un programa. El valor calculado, define el número de caminos independientes del conjunto básico de un programa y nos da un límite superior para el número de pruebas que se deben realizar para asegurar que se ejecute cada sentencia al menos una vez.

Un camino independiente es cualquier camino del programa que introduce por lo menos un nuevo conjunto de sentencias de procesamiento o una nueva condición. El camino independiente se debe mover por lo menos por una arista que no haya sido recorrida anteriormente. La complejidad ciclomática de un grafo de flujo se puede calcular a través de 3 vías. Para el ejemplo mostrado en la figura 1. 2 sería:

I.  $V(G) = \text{Número de Regiones}$

$$V(G) = 2$$

II.  $V(G) = \text{Número de Aristas} - \text{Número de Nodos} + 2$

$$V(G) = 8 - 7 + 2$$

$$V(G) = 3$$

III.  $V(G) = \text{Número de Nodos Predicados} + 1$

$$V(G) = 2 + 1$$

$$V(G) = 3$$

### 🚦 Derivación de casos de prueba

Luego de tener elaborados los Grafos de Flujos y los caminos a recorrer, se preparan los casos de prueba que forzarán la ejecución de cada uno de esos caminos. Se escogen los datos de forma que las condiciones de los nodos predicados estén adecuadamente establecidas, con el fin de comprobar cada camino.

### **2. Técnica de condición**

Es un método de diseño de casos de prueba que ejercita las condiciones lógicas contenidas en el módulo de un programa.

### **3. Técnica de flujo de datos**

Se seleccionan caminos de prueba en un programa de acuerdo con la ubicación de las definiciones y los usos de las variables del programa.

### **4. Técnica de bucles**

Es una técnica de prueba de caja blanca que se centra exclusivamente en la validez de las construcciones de bucles.

#### **1.9. Niveles de Prueba**

La Prueba es aplicada para diferentes tipos de objetivos, en diferentes escenarios o niveles de trabajo. Se distinguen los siguientes niveles de pruebas:

- Prueba de Desarrollador
- Prueba Independiente
- Prueba de Unidad
- Prueba de Integración
- Prueba de Sistema
- Prueba de Aceptación

##### **1.9.1. Prueba de Desarrollador**

Es la prueba diseñada e implementada por el equipo de desarrollo. Tradicionalmente estas pruebas han sido consideradas solo para la prueba de unidad, aunque en la actualidad en algunos casos pueden ejecutar pruebas de integración. Se recomienda que estas pruebas cubran más que las pruebas de unidad.

##### **1.9.2. Prueba Independiente**

Es la prueba que es diseñada e implementada por alguien independiente del grupo de desarrolladores. El objetivo de estas pruebas es proporcionar una perspectiva diferente y en un

ambiente más rico que los desarrolladores. Una vista de la prueba independiente es la prueba de los clientes (stakeholder), que son pruebas basadas en las necesidades y preocupaciones de los stakeholders.

### **1.9.3. Prueba de Unidad**

Es la prueba enfocada a los elementos testeables más pequeños del software. Es aplicable a componentes representados en el modelo de implementación para verificar que los flujos de control y de datos están cubiertos, y que ellos funcionen como se espera. La prueba de unidad siempre está orientada a caja blanca. Antes de iniciar cualquier otra prueba es preciso probar el flujo de datos de la interfaz del componente. Si los datos no entran correctamente, todas las demás pruebas no tienen sentido. El diseño de casos de prueba de una unidad comienza una vez que se ha desarrollado, revisado y verificado en su sintaxis el código a nivel fuente.

### **1.9.4. Prueba de unidad en el contexto OO (Orientado a Objetos)**

En lugar de módulos individuales, una menor unidad a probar es la clase u objeto encapsulado. Una clase puede contener un cierto número de operaciones, y una operación particular puede existir como parte de un número de clases diferentes. Esta prueba de clases para el software OO es equivalente a la prueba de unidad para el software convencional. La prueba de clases para software OO está dirigida por las operaciones encapsuladas por la clase y el estado del comportamiento de la clase. No se puede probar una operación aisladamente, sino como parte de una clase.

### **1.9.5. Prueba de Integración**

Es ejecutada para asegurar que los componentes en el modelo de implementación operen correctamente cuando son combinados para ejecutar un caso de uso. Se prueba un paquete o un conjunto de paquetes del modelo de implementación. Estas pruebas descubren errores o incompletitud en las especificaciones de las interfaces de los paquetes. Esta prueba debe ser responsabilidad de desarrolladores y de independientes, sin solaparse las pruebas.

### **1.9.6. Prueba de Sistema**

Son las pruebas que se hacen cuando el software está funcionando como un todo. Es la actividad de prueba dirigida a verificar el programa final, después que todos los

componentes de software y hardware han sido integrados. En un ciclo iterativo, estas pruebas ocurren más temprano, tan pronto como subconjuntos bien formados de comportamiento de caso de uso son implementados.

### ➤ Tipos de Pruebas del Sistema

#### **Prueba de Recuperación**

Es una prueba del sistema que fuerza el fallo del software de muchas formas y verifica que la recuperación se lleva a cabo apropiadamente.

#### **Prueba de Seguridad**

Intenta verificar que los mecanismos de protección incorporados en el sistema lo protegerán de accesos impropios.

#### **Prueba de Resistencia**

Están diseñadas para enfrentar a los programas con situaciones anormales.

#### **Prueba de Rendimiento**

Está diseñada para probar el rendimiento del software en tiempo de ejecución dentro del contexto de un sistema integrado.

#### **1.9.7. Prueba de Aceptación del Usuario**

Es la prueba final antes del despliegue del sistema. Su objetivo es verificar que el software está listo y que puede ser usado por usuarios finales para ejecutar aquellas funciones y tareas para las cuales el software fue construido. Esta prueba también es conocida como “Prueba Piloto”.

Las pruebas de unidad son implementadas en la iteración más temprana como el primer nivel de prueba. Pero en un proceso iterativo ejecutar todas las pruebas de unidad antes de pasar a niveles siguientes de prueba como regla es inapropiado. Una mejor estrategia es identificar las pruebas de unidad, integración y sistema que ofrecen mayor potencial para encontrar errores y entonces implementarlas y ejecutarlas.

#### **1.10. Tipos de Prueba basados en dimensiones de la calidad**

Cada tipo de prueba tiene un objetivo específico y una técnica que lo soporte.

A continuación se muestra los tipos de pruebas basados en dimensiones de calidad:

### 1.10.1. Funcionalidad

- a. **Función.** Pruebas fijando su atención en la validación de las funciones, métodos, servicios y caso de uso.
- b. **Seguridad.** Asegurar que los datos o el sistema solamente es accedido por los actores deseados.
- c. **Volumen.** Enfocada en verificar las habilidades de los programas para manejar grandes cantidades de datos, tanto como entrada, salida o residente en la BD.

### 1.10.2. Usabilidad

Prueba enfocada a factores humanos, estéticos, consistencia en la interfaz de usuario, ayuda sensitiva al contexto y en línea, asistente documentación de usuarios y materiales de entrenamiento.

### 1.10.3. Fiabilidad

- a. **Integridad.** Enfocada a la valoración de la robustez (resistencia a fallos).
- b. **Estructura.** Enfocada a la valoración a la adherencia a su diseño y formación. Este tipo de prueba es hecho a las aplicaciones Web asegurando que todos los enlaces están conectados, el contenido deseado es mostrado y no hay contenido huérfano.
- c. **Stress.** Enfocada a evaluar cómo el sistema responde bajo condiciones anormales (Extrema sobrecarga, insuficiente memoria, servicios y hardware no disponible, recursos compartidos no disponible).

### 1.10.4. Performance (Rendimiento)

- a. **Benchmark (Criterio).** Es un tipo de prueba que compara el rendimiento de un elemento nuevo o desconocido a uno de carga de trabajo de referencia conocido.
- b. **Contención.** Enfocada a la validación de las habilidades del elemento a probar para manejar aceptablemente la demanda de múltiples actores sobre un mismo recurso (registro de recursos, memoria, etc.).
- c. **Carga.** Usada para validar y valorar la aceptabilidad de los límites operacionales de un sistema bajo carga de trabajo variable, mientras el sistema bajo prueba, permanece

constante. La variación en carga es simular la carga de trabajo promedio y con picos que ocurre dentro de tolerancias operacionales normales.

- d. **Performance profile (Perfil de desempeño).** Enfocadas a monitorear el tiempo en flujo de ejecución, acceso a datos, en llamada a funciones y sistema para identificar y direccionar los cuellos de botellas y los procesos ineficientes.

### 1.10.5. Soportabilidad

- a. **Configuración.** Enfocada a asegurar que funciona en diferentes configuraciones de hardware y software. Esta prueba es implementada también como prueba de rendimiento del sistema.
- b. **Instalación.** Enfocada a asegurar la instalación en diferentes configuraciones de hardware y software bajo diferentes condiciones (insuficiente espacio en disco, etc.).

### 1.11. Conclusiones

Con la realización de este Capítulo se observó los modelos y aspectos relacionados con la calidad del software; se estudió a profundidad el flujo de trabajo de prueba, el cual se puede realizar en todas las etapas de desarrollo del software pero principalmente en la fase de implementación es donde tiene el mayor peso, así como los diferentes métodos o enfoques de pruebas, en qué consiste cada uno y sus diferentes técnicas. Además se analizó los diferentes niveles de pruebas y las pruebas basadas en las dimensiones de calidad, con el objetivo de diseñar casos de prueba que permitan lograr que el producto se encuentre lo más libre de defectos posibles.

## **CAPÍTULO 2. PLAN DE PRUEBA.**

### **2.1. Introducción**

En este Capítulo se define el Plan de Prueba para el Simulador Termo Azúcar (STA) en su versión 4.0, producto que nunca ha sido sometido a un proceso de corrección de defectos; este plan guía el proceso de ejecución, evolución y seguimiento de las pruebas; y contiene la Estrategia de Prueba a seguir para garantizar el éxito del mismo.

Para la confección de este Plan se tienen en cuenta aspectos como:

- ✓ Propósito.
- ✓ Alcance.
- ✓ Estrategia de evolución del Plan.
- ✓ Especificaciones de Hardware y Software.
- ✓ Requerimientos a probar.
- ✓ Estrategia de prueba.
- ✓ Tipos de prueba
- ✓ Recursos requeridos.

### **2.2. Plan de Prueba para el STA**

#### **2.2.1. Introducción**

El Plan de Prueba para el STA en el primer ciclo de su desarrollo tiene como objetivo lograr una mayor y mejor organización para el proceso de pruebas, definir una estrategia de prueba adecuada, tipos de prueba y evolución del Plan para garantizar una mayor calidad del producto antes de que sea desplegado, eliminando los defectos que pueda contener el mismo.

#### **2.2.2. Propósito**

El Plan de Prueba para el STA tiene definidos los siguientes objetivos para su realización:

- Definir los tipos de pruebas a realizar.
- Definir la infraestructura de las pruebas a realizar.
- Definir la evolución y alcance de las pruebas.
- Definir los recursos requeridos.
- Definir la estrategia de prueba que se va a llevar a cabo en el primer ciclo de desarrollo del producto.

### 2.2.3. Alcance

Está dividido en diferentes etapas para lograr una mejor organización:

**Primera Etapa:** Se realizarán pruebas a nivel de unidad a la funcionalidad Insertar Información que requiere de la creación (precondición) de un Diagrama de Flujo de Información (DFI) y hace referencia a los Requisitos Funcionales (RF) Insertar Información Corriente e Insertar Información Módulo. Es una prueba enfocada al método de Caja Blanca.

**Segunda Etapa:** En esta etapa se realizará pruebas de sistema a las funcionalidades de Reportar resultados, Reportar datos, Identificar corriente, Identificar módulos y Conversor de Unidades; esta prueba está enfocada al método de Caja Negra.

### 2.2.4. Estrategia de evolución del Plan

El plan será chequeado diariamente y semanalmente de manera general a ver si es necesario algún cambio. Al finalizar la semana el equipo de prueba compuesto por una persona, es el encargado de analizar los posibles cambios e informar a los desarrolladores y Jefe de Proyecto para facilitar que el proceso de prueba que se está llevando a cabo no se detenga.

### 2.2.5. Organización del Equipo de Prueba

Para realizarle pruebas a un producto se debe contar con un equipo de prueba compuesto por:



1. Administrador de prueba
2. Analista de prueba
3. Diseñador de prueba
4. Probador.

En el caso del STA no se cuenta con este equipo, sino es una persona la que va a desarrollar los cuatro roles especializándose en los roles de Diseñador de casos de pruebas y Probador.

El diseñador de casos de pruebas es el encargado de planificar, diseñar y evaluar los casos de pruebas y sus resultados.

El probador es el encargado de ejecutar las pruebas de unidad y sistema guiado por los casos de pruebas, es el responsable de identificar y listar los errores encontrados y evaluar junto al diseñador de casos de pruebas los resultados de la evaluación de las mismas.

### **2.2.6. Especificaciones de Software y Hardware**

Para llevar a cabo un exitoso proceso de pruebas en el STA se deben utilizar como mínimo dos computadoras, una para la realización de las pruebas de caja blanca y otra para la realización de las pruebas de caja negra realizándose las mismas de manera simultánea.

Este producto es una aplicación de escritorio para simular los procesos que se realizan en la industria azucarera por lo que puede ser instalado en cualquier computadora personal (PC) que tenga más de 128 MB de memoria RAM, Procesador Pentium® III a 600 MHz y 500 MB de espacio en disco. Además Sistemas Operativos (SO) compatibles con Windows 2000; Windows Server 2003 Service Pack 1; Windows XP y Microsoft .NET Framework Versión 1.1. Sin embargo para que su funcionamiento sea más óptimo el computador debe de contar con una memoria de 512 Mega byte.

### 2.2.7. Descripción del Plan de Pruebas

#### Descripción de los requerimientos a probar

Se van a probar los requisitos funcionales y no funcionales que forman parte de la documentación del proyecto dentro del alcance definido según los recursos con los que se cuenta.

Dentro de los funcionales se probará:

1. Insertar Información corriente.
2. Insertar Información módulo.
3. Reportar todos los resultados.
4. Reportar resultados seleccionados.
5. Reportar datos.
6. Identificar corriente.
7. Identificar módulos.
8. Conversor de Unidades.

Dentro de los requisitos no funcionales se probará que:

1. El sistema debe ser una aplicación de escritorio.
2. Tener una paleta de figuras estándar.
3. Destacar con colores distintos las corrientes distintas por naturaleza.
4. Destacar con colores distintos los módulos cuya información ha sido suministrada.
5. Debe contar con una ayuda que trate tanto los elementos informáticos sobre el uso del simulador como los elementos generales de los modelos matemáticos que se simulan.

6. Debe contar con videos demos de la aplicación que muestren como usar el simulador.

### Requerimientos funcionales

Los requerimientos funcionales correspondientes al STA que se probarán son los siguientes:

El Insertar Información corriente e Insertar Información módulo, los cuales es la referencia del caso de uso Insertar Información; Reportar todos los resultados y Reportar resultados seleccionados los cuales es la referencia del caso de uso Reportar resultados; Reportar datos y Reportar resultados seleccionados los cuales es la referencia del caso de uso Reportar datos; Identificar corriente, Identificar módulos y Conversor de Unidades.

### Casos de Usos a probar

- **Insertar Información:** Para lograr la simulación no solo hace falta tener un DFI y vías para calcular determinados parámetros. También son necesarios datos de entrada para la realización de dichos cálculos y para lograr la completitud del DFI. El hecho de entrar datos implica que de alguna manera deben ser almacenados estos datos, es este el porqué se clasificó este caso uso como arquitectónicamente significativo, pues este implica la toma de decisiones importantes sobre la estructura del almacenamiento de los datos, así como cuestiones de accesibilidad a estos.
- **Reportar resultados:** La simulación de un proceso arroja gran cantidad de datos que deben ser organizados para una mejor comprensión y así convertirlos en información útil en el análisis del proceso por los usuarios. En la arquitectura base se implementara parte de este caso de uso, permitiendo solamente un tipo de reporte, en este caso el reporte condensado, que muestre los datos que resultan de la operación de simular. La inclusión de este en la arquitectura base permitirá evaluar la capacidad del sistema de calcular de manera correcta y completa, es decir, que se obtienen todos los parámetros esperados por los usuarios.
- **Reportar datos:** La simulación de un proceso necesita gran cantidad de datos relacionados con el proceso, la presión de los datos y la exactitud de estos tiene

gran importancia en el resultado final de la simulación. El sistema debe ser capaz de generar un reporte que incluya todos los datos introducidos por el usuario.

- **Identificar corriente:** Esta funcionalidad permitirá identificar los elementos del DFI que simulan las corrientes del proceso.
- **Identificar módulos:** Esta funcionalidad permitirá identificar los elementos del DFI que simulan equipos.
- **Conversor de unidades:** Las propiedades físicas y químicas de las corrientes y equipos tienen asociada una unidad de medida determinada que permite mejorar la calidad de la información que brinda el valor numérico asociado a esa propiedad. Existen distintos sistemas de unidades para designar las magnitudes, sistemas que son usados indistintamente por los especialistas, a veces mezclando los sistemas en que se dan los datos. El simulador debe dar la posibilidad de que el usuario suministre los datos y los reciba en la unidad de medida que prefiera, esto, en contraste con que los cálculos de las propiedades internamente se realicen con unidades específicas. La implementación de las conversiones entre unidades, entre sistemas de medida o incluso dentro del mismo puede convertirse en un problema serio si no es bien tratado.

### 2.2. 8. Estrategia de Prueba

A continuación se detalla el flujo de trabajo prueba que se llevará a cabo y las diferentes estrategias en cada una de las etapas definidas.

De manera general el flujo de prueba inicia cuando el grupo de prueba define el Plan de Prueba y comienza las actividades propias del proceso de prueba.

Para la etapa I se definirán y especificarán casos de prueba de unidad para la funcionalidad Insertar Información, mediante el tipo de prueba de caja blanca y utilizando la técnica del camino básico que comprueba la lógica interna de cada instancia.

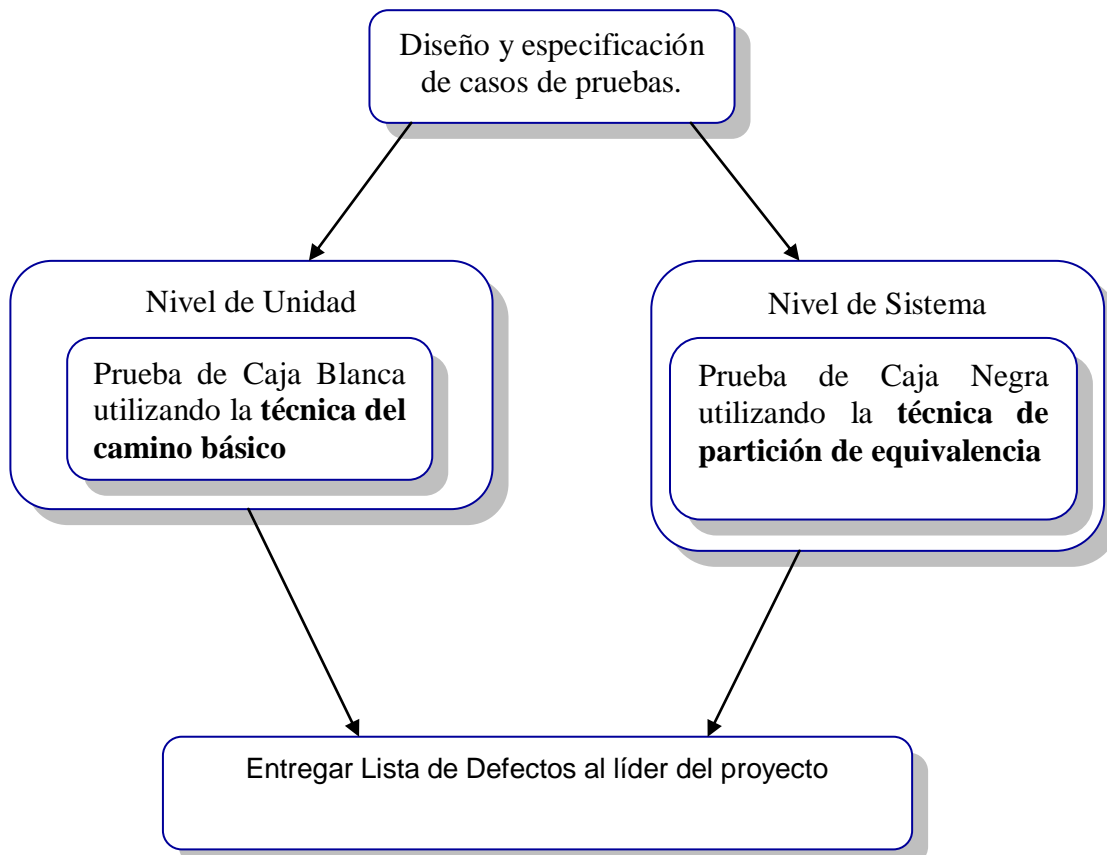
Estos resultados serán evaluados en dependencia a los resultados esperados, en la medida que empiecen a aparecer errores estos serán anotados para al terminar el día elaborar una lista de defectos, que se le entregará al líder de proyecto por parte del

equipo de calidad del proyecto, quien informará de estos errores al equipo de desarrollo motivando un proceso de gestión de cambios paralelo con el proceso de prueba.

Para la etapa II se definirán casos de prueba de caja negra utilizando la técnica de partición equivalente para probar los resultados según los datos introducidos en los campos de entrada, ya sean válidos o inválidos, previamente especificados en los casos de prueba. Estos casos de prueba se ejecutarán a través de la interfaz implementada para la aplicación del STA.

A partir de aquí se llevará cabo un proceso similar a la etapa anterior, en la medida que empiecen a aparecer errores estos serán anotados para al terminar el día elaborar una lista de defectos, para lo cual se recomienda utilizar el cuaderno del ingeniero para ir archivando los defectos resultantes de la ejecución de la pruebas de forma organizada, esta lista de defectos se le entregará al líder de proyecto por parte del equipo de calidad del proyecto, quien informará de estos errores al equipo de desarrollo motivando un proceso de gestión de cambios paralelo con el proceso de prueba.

A continuación se muestra mediante la figura 2 como queda definido el proceso de prueba que se desea realizar.



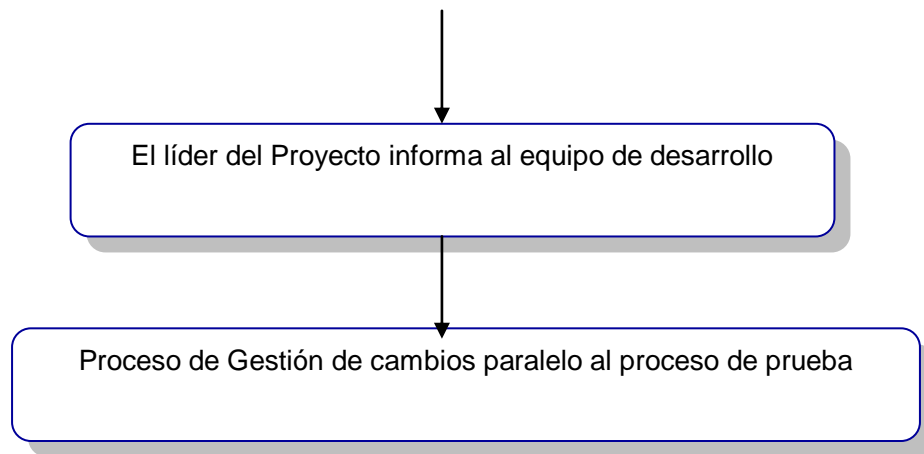


Figura 2. Proceso de pruebas a realizar.

### 2.2.9. Tipos de Pruebas a realizar

#### ➤ Prueba de Caja Blanca

Se le realizará a las funcionalidades:

1. Insertar Información corriente.
2. Insertar Información módulo.

#### ➤ Prueba de Caja Negra

Se le realizará a las funcionalidades:

1. Reportar todos los resultados.
2. Reportar resultados seleccionados.
3. Reportar datos.
4. Identificar corriente.
5. Identificar módulos.
6. Conversor de Unidades.

Además al probar las funcionalidades anteriormente mencionadas se debe de realizar:

- ✓ Prueba de Funcionalidad.
  
- ✓ Prueba de Documentación.

### ➤ **Prueba de Funcionalidad**

Esta prueba está compuesta por tres tipos diferentes de pruebas:

- La prueba de Función.
- La prueba de Seguridad.
- La prueba de Volumen.

Para el producto STA solamente se tendrá en cuenta a la hora de realizar prueba de Funcionalidad la prueba de Función, ya que las pruebas de Volumen y Seguridad son muy costosas en tiempo de máquina y el simulador no cuenta con un módulo de seguridad respectivamente.

### **Prueba de Función**

Está enfocada al trabajo apropiado de los requisitos funcionales, incluyendo la navegación, entrada de datos, procesamiento y obtención de resultados. Verifica el procesamiento, recuperación e implementación adecuada de las reglas del negocio y la apropiada aceptación de datos.

### **Objetivo de la Prueba**

Se ejecuta cada caso de uso a probar, flujo de caso de uso o función, usando datos válidos e inválidos, para verificar que se aplique apropiadamente cada regla de negocio, que los resultados esperados ocurran cuando se usen datos válidos y que sean desplegados los mensajes apropiados de error y precaución cuando se usan datos inválidos.

### **Técnica**

Se entrarán datos incorrectos a las funcionalidades de Conversor de Unidades, Identificar corriente e Identificar módulos para observar si son desplegados los mensajes de error cuando se utilizan datos inválidos, también se probará estas funcionalidades con datos que sean correctos para ver si se obtienen los resultados esperados.

### **Criterio de Término**

Luego que sea aplicada la prueba si se encuentra un determinado defecto se pondrá en práctica la evolución del Plan, donde se le informa al Jefe de Proyecto los defectos

encontrados para poder trabajar en paralelo y no detener el proceso de prueba. En caso contrario se informa también.

### Herramienta

Esta prueba se realizará de forma manual

#### ➤ Prueba de Documentación

Está enfocada a verificar que los proyectos cuentan con toda la documentación referida al software (expediente de proyecto) que se genera en las diferentes fases de desarrollo del mismo.

### Objetivo de la Prueba

Verificar que el expediente del proyecto cuente con todos los entregables y documentos definidos por la metodología utilizada para el desarrollo de la aplicación (RUP) y el equipo de desarrollo.

### Técnica

Se revisara el contenido del Manual de Usuario y la Ayuda del STA así como la existencia de los ficheros Config.ini, dotnetfx.exe, Settings.ini, STA 4.0.msi y Setup.exe. Además la presencia de los documentos Modelo del Negocio, Modelo del Sistema, Modelo Objeto del Negocio, Lista de Riesgos, Documento Visión, Modelo Diseño, Modelo Despliegue, Modelo Implementación, Modelo Prueba, Glosario de Términos, Especificación RF y RNF.

### Criterio de Término

Luego que sea aplicada la prueba si se encuentra un determinado defecto se pondrá en práctica la evolución del Plan, donde se le informa al Jefe de Proyecto los defectos encontrados para poder trabajar en paralelo y no detener el proceso de prueba. En caso contrario se informa también.

### Herramienta

Esta prueba se realizará de forma manual.

## 2.3. Entorno de Prueba

Entorno de Prueba	Características de la PC
Se cuenta con una	PC que va a ser Intel Pentium IV de 256 de RAM, con



utilizada por la persona que va a llevar a cabo todos los roles del flujo de trabajo de Prueba, especializándose en Diseñador de Pruebas y Probador. La PC en la que se van a desarrollar las pruebas está compartida con 2 estudiantes uno de tercero y uno de cuarto año de la carrera, lo que puede dificultar el proceso de prueba que se la va a realizar al STA.

Sistema Operativo Windows XP Professional, disco duro con una capacidad de 80 Gigabyte.

Tabla 1. Entorno de Pruebas

**2.4. Recursos Requeridos**

Para la realización de las pruebas al STA se cuenta solamente con una persona como capital humano que se va a desempeñar en todos los roles, especializándose en el de Diseñador de casos de prueba y en el del Probador. En la siguiente tabla se muestra una breve descripción de los roles y las responsabilidades que cada uno tiene para llevar a cabo un exitoso proceso de pruebas.

Roles	Responsabilidades
<b>Administrador de Pruebas</b>	Realizar el Plan de Prueba. Determinar las pruebas que se le van a realizar al producto. Verificar la evolución del Plan de Prueba. Mantener informado al Jefe de Proyecto y al grupo de desarrolladores del resultado de las pruebas.

## PLAN DE PRUEBA

<b>Analista de Pruebas</b>	<p>Participa en la confección del Plan de Prueba.</p> <p>Determina las pruebas a realizar.</p> <p>Determina los objetivos de las pruebas.</p> <p>Participa en la evaluación de los resultados.</p>
<b>Diseñador de Pruebas</b>	<p>Participa en la elaboración del Plan de Prueba.</p> <p>Diseña los casos de pruebas que se le van a realizar al STA.</p> <p>Mantiene la integridad del Plan de Prueba.</p> <p>Participa en la evaluación de los resultados derivados de las pruebas realizadas.</p>
<b>Probador</b>	<p>Ejecuta los casos de pruebas.</p> <p>Realiza una lista de los defectos encontrados.</p> <p>Participa en la evaluación de los resultados.</p>

Tabla 2. Recursos requeridos.

La planificación de los procesos de prueba es una etapa de vital importancia, ya que además de definir la estrategia y establecer los recursos necesarios, permite realizar estimaciones que establecen una base para el control y seguimiento durante la ejecución del proceso. Establece un referente para futuros proyectos y además ofrece información para realizar el análisis de los procesos para identificar aspectos de mejoramiento. La construcción de un buen plan de prueba es la piedra angular y en consecuencia uno de los principales factores críticos de un proceso de prueba que permite entregar un software

de mejor nivel. Es importante no hacer los planes de prueba suponiendo que prácticamente no hay defectos en los programas, y por lo tanto dedicando pocos recursos a las pruebas, pues siempre hay defectos.<sup>18</sup>

### **2.5. Conclusiones**

Con la realización de este Capítulo se dejó bien definido el Plan de Prueba para el STA, teniendo en cuenta las características del producto. Se escogieron las pruebas que se le van a realizar, se describió la estrategia a seguir para tener un proceso de prueba organizado y que arroje buenos resultados, se estudió y se dejó esclarecidas las condiciones en las que se van a realizar las pruebas y los inconvenientes que puede tener el proceso; así como los recursos con los que se cuenta.

---

<sup>18</sup> Gonzalez,C. Un Plan de Pruebas Exitoso.2002 [Disponible en <http://www.americaxxi.cl/modules.php?name=News&file=article&sid=20>].

## CAPÍTULO 3. DISEÑO, EJECUCIÓN Y EVALUACIÓN DE LAS PRUEBAS

### 3.1. Introducción

En este capítulo se definirá el diseño y ejecución de pruebas utilizando los métodos de caja blanca y caja negra para las diferentes funcionalidades que se mencionaron en el Capítulo 2. Luego de la realización de los casos de pruebas se hará una evaluación y un resumen de los resultados que se obtuvieron al finalizar el proceso de detección de dificultades.

Para las pruebas de caja negra se utilizó la plantilla de definición de casos de pruebas definidas por la Universidad de las Ciencias Informáticas (UCI) y para las pruebas de caja blanca se definió una plantilla de definición de casos de pruebas, ya que en la Universidad no se realiza este tipo de prueba por los equipos de calidad de los proyectos. Las técnicas de pruebas que se emplearon fueron la de partición de equivalencia y la del camino básico para los métodos de caja negra y caja blanca respectivamente.

### 3.2. Definición de los casos de pruebas de Caja Negra

#### 3.2.1. Caso de prueba Identificar Corriente

##### Descripción de la funcionalidad

Esta funcionalidad permite nombrar e identificar los elementos del DFI (Diagrama de Flujo de Información) que simulan las corrientes del proceso.

##### Flujo Central

1. Escritorio.
2. Dar clic en el programa Sistema Termo Azúcar.
3. Ir al DFI (área blanca a la derecha de la Paleta de módulos).
4. Dar clic derecho sobre la conexión que tiene cada uno de los módulos.
5. Escoger la opción **Nombrar corriente**.

### Iteraciones

Clases Válidas	Clases Inválidas	Resultado Esperado	Resultado de la Prueba
1. El campo Entre el nombre y entre el ID de la corriente están escritos correctamente		El sistema acepte el nombre y el ID de la corriente y que cuando se estacione el mouse sobre la conexión este muestre el nombre de la corriente.	El sistema acepta el nombre y el ID y la conexión muestra el nombre cuando se estaciona el mouse sobre él.
	1.1. Los campos Entre el nombre y/o entre el ID de la corriente están en blanco.	El sistema muestra un mensaje de error indicando que no puede dejar campos de entrada obligatoria en blanco.	El sistema acepta los dos o un campo en blanco.
	1.2 .El campo Entre el nombre tiene caracteres extraños, dígame @, #, \$, %, ^, &, *, {, }.	El sistema muestre un mensaje de error o no deje escribir caracteres extraños.	El sistema acepta los caracteres extraños.
	1.3. El campo Entre el ID tiene letras y caracteres extraños, dígame @, #, \$, %, ^, &, *, {, }.	El sistema muestre un mensaje de error o no deje escribir letras ni caracteres extraños.	El sistema no deja escribir letras ni caracteres extraños, solo números
	1.4. Se escribe un ID ya existente	El sistema muestre un mensaje de error.	El sistema muestra el mensaje de error: "Hay un elemento en

## DISEÑO, EJECUCIÓN Y EVALUACIÓN DE LAS PRUEBAS

			el diagrama con ese identificador.”
--	--	--	-------------------------------------

Tabla 3.1. Diseño del caso de prueba Identificar Corriente

### Registro de defectos y dificultades detectados

Elemento	No	No conformidad	Aspecto correspondiente	Etapas de detección Código del CP	Recomendación
Los campos Entre el nombre y/o entre el ID de la corriente están en blanco.	1	El sistema acepta los dos o un campo en blanco.	Anexo 1.	Prueba a nivel de Sistema.	Validar que estos campos sean llenados para tener un mejor control de las corrientes que se tienen.
El campo Entre el nombre tiene caracteres extraños, dígame @, #, \$, %, ^, &, *, {, }.	2	El sistema acepta caracteres extraños.	Anexo 1.	Prueba a nivel de Sistema.	Validar que en ese campo no se entren caracteres extraños para la simulación tenga los nombres más parecidos posible a la realidad.

Tabla 3.2. Defectos encontrados en la funcionalidad Identificar Corriente

### 3.2.2. Caso de prueba Identificar Módulo

#### Descripción de la funcionalidad

## DISEÑO, EJECUCIÓN Y EVALUACIÓN DE LAS PRUEBAS

Esta funcionalidad permite nombrar e identificar los elementos del DFI (Diagrama de Flujo de Información) que simulan equipos.

### Flujo Central

1. Escritorio.
2. Dar clic en el programa Sistema Termo Azúcar.
3. Ir al DFI (área blanca a la derecha de la Paleta de módulos).
4. Dar clic derecho sobre los módulos.
5. Escoger la opción **Renombrar**.

### Iteraciones

Clases Válidas	Clases Inválidas	Resultado Esperado	Resultado de la Prueba
1.El campo Teclee el nombre del módulo y Teclee el ID del módulo están escritos correctamente		El sistema acepte el nombre y el ID de módulo y que cuando se estacione el mouse en el módulo este muestre el nombre.	El sistema acepta el nombre y el ID y el módulo muestra el nombre cuando se estaciona el mouse sobre él.
	1.1. Los campos Teclee el nombre del módulo /y Teclee el ID de módulo están en blanco.	El sistema muestre un mensaje de error indicando que no puede dejar campos de entrada obligatoria en blanco.	El sistema acepta los dos o un campo en blanco.
	1.2 .El campo Teclee el nombre del módulo tiene caracteres extraños, dígame @, # \$, %, ^, &, *, {, }.	El sistema muestre un mensaje de error o no deje escribir caracteres extraños.	El sistema acepta los caracteres extraños
	1.3. El campo Teclee el ID del módulo tiene	El sistema muestre un mensaje de error o no	El sistema no deja escribir letras ni

## DISEÑO, EJECUCIÓN Y EVALUACIÓN DE LAS PRUEBAS

	letras y caracteres extraños, dígame @, # \$, %, ^, &, *, {, }.	deje escribir letras ni caracteres extraños.	caracteres extraños, solo números
	1.4. Se escribe un ID ya existente	El sistema muestre un mensaje de error.	El sistema muestra el mensaje de error: "Ya hay un elemento en el diagrama con ese identificador."

Tabla 3.3. Diseño del caso de prueba Identificar Módulo

### Registro de defectos y dificultades detectados

Elemento	No	No conformidad	Aspecto correspondiente	Etapas de detección Código del CP	Recomendación
Los campos Teclee el nombre del módulo o Teclee el ID del módulo están en blanco.	1	El sistema acepta los campos de Teclee el nombre del módulo o/y Teclee el ID del módulo en blanco.	Anexo 2	Pruebas a nivel de Sistema	Validar que estos campos sean llenados para tener un mejor control de los módulos que se tienen.
El campo Entre el nombre tiene caracteres extraños,	2	El sistema acepta en el campo de Teclee el nombre del módulo caracteres extraños.	Anexo 2	Pruebas a nivel de Sistema	Validar que en ese campo no se entren caracteres extraños para la simulación tenga los nombres más



## DISEÑO, EJECUCIÓN Y EVALUACIÓN DE LAS PRUEBAS

dígase @, #, \$, %, ^, &, *, {, }.					parecidos posible a la realidad.
--	--	--	--	--	-------------------------------------

Tabla 3.4. Defectos encontrados en la funcionalidad Identificar Módulo

### 3.2.3. Caso de prueba Conversor de unidades

#### Descripción de la funcionalidad

Esta funcionalidad permite que el usuario suministre los datos y los reciba en la unidad de medida que prefiera, esto, en contraste con que los cálculos de las propiedades se realicen internamente con unidades específicas.

#### Flujo Central

1. Escritorio.
2. Dar clic en el programa Sistema Termo Azúcar.
3. Ir a la opción Proyecto, dar clic.
4. Escoger la opción Herramienta y dentro de esta la opción **Conversor de unidades**.

Otra manera de llegar a esta funcionalidad es:

- 1) Escritorio.
- 2) Dar clic en el programa Sistema Termo Azúcar.
- 3) Apretar las teclas Ctrl +U

#### Iteraciones

Clases Válidas	Clases Inválidas	Resultado Esperado	Resultado de la Prueba
1.El campo "Magnitudes, De(en este campo se pone lo que se quiere convertir), Unidades de Medidas y cifras decimales"		El sistema acepte los datos introducidos y escogidos en cada uno de los campos.	El sistema acepta los datos introducidos y escogidos y realiza la conversión de unidades.

## DISEÑO, EJECUCIÓN Y EVALUACIÓN DE LAS PRUEBAS

están escritos correctamente.			
	1.1. Se trata de escribir en los campos Magnitudes y Unidades de medidas.	El sistema no permita escribir en los campos.	El sistema no permite que se escriba en los campos magnitudes y unidades de Medidas
	1.2. Se trata de escribir en el campo cifras decimales.	El sistema no permita escribir en este campo a no ser los elementos que se trae predefinido para escoger.	El sistema no acepta la introducción de ningún dato a no ser los que él trae por defecto.
	1.3. Se trata de introducir caracteres extraños a los campos De, dígase @, #, \$, %, ^, &, *, {, }.	El sistema no permita escribir los caracteres extraños @, #, \$, %, ^, &, *, {, }.	El sistema no acepta la introducción de ningún carácter extraño (@, #, \$, %, ^, &, *, {, }).
	1.4. Se trata de introducir letras a los campos De.	El sistema no permita escribir letras en los campos De.	El sistema no permite introducir letras en los campos De.

Tabla 3.5. Diseño del caso de Conversor de Unidades

### Registro de defectos y dificultades detectados

No se encontró ningún defecto ni dificultad.

### 3.2.4. Caso de prueba Reportar datos

#### Descripción de la funcionalidad

## DISEÑO, EJECUCIÓN Y EVALUACIÓN DE LAS PRUEBAS

La simulación de un proceso necesita gran cantidad de datos relacionados con el proceso, la presión de los datos y la exactitud de estos tiene gran importancia en el resultado final de la simulación. El sistema debe de ser capaz de generar un reporte que incluya todos los datos introducidos por el usuario.

### Flujo Central

1. Escritorio.
2. Dar clic en el programa Sistema Termo Azúcar.
3. Dar clic en el botón Reportar Resultado de la Barra de Herramienta.
4. Escoger el tipo de reporte Datos introducidos.

Otra manera de llegar a esta funcionalidad.

- 1) Escritorio.
- 2) Dar clic en el programa Sistema Termo Azúcar.
- 3) Dar clic en el Proyecto.
- 4) Escoger la opción Reporte.
- 5) Escoger el tipo de reporte Datos introducidos.

### Iteraciones

Clases Válidas	Clases Inválidas	Resultado Esperado	Resultado de la Prueba
1. Se selecciona en Reporte de condensado la opción Datos introducidos.		El sistema muestre un Reporte con los parámetros particulares de los módulos.	El sistema muestra un reporte con los parámetros particulares de los módulos.

## DISEÑO, EJECUCIÓN Y EVALUACIÓN DE LAS PRUEBAS

	1.1. No se selecciona en Reporte condensado la opción Datos introducidos).	El sistema muestra un mensaje avisándole al usuario que debe seleccionar esta opción para obtener este reporte.	El sistema no muestra ningún mensaje avisándole al usuario que debe escoger la opción Datos introducidos para hacer un reporte de este tipo.
1.2. Se selecciona el Reporte de indicadores.		El sistema muestra un reporte con los indicadores globales por áreas y equipos.	El sistema muestra un reporte con los indicadores globales por áreas y equipos.

Tabla 3.6. Diseño del caso de prueba Reportar Datos

### Registro de defectos y dificultades detectados.

Elemento	No	No conformidad	Aspecto correspondiente	Etapa de detección Código del CP
La opción Resultados de la simulación del tipo de reporte condensado no es escogida	1	El sistema acepta esta opción sin mostrarle al usuario un mensaje de que debe de escogerla para realizar un reporte de este tipo.	Anexo 3	Sistema

Tabla 3.7. Defectos detectados en la funcionalidad Reportar Datos

### 3.2.5. Caso de prueba Reportar resultados

#### Descripción de la funcionalidad

## DISEÑO, EJECUCIÓN Y EVALUACIÓN DE LAS PRUEBAS

Esta funcionalidad permite evaluar la capacidad del sistema de calcular de manera correcta y completa todos los datos introducidos, es decir, se obtienen todos los parámetros esperados por los usuarios.

### Flujo Central

1. Escritorio.
2. Dar clic en el programa Sistema Termo Azúcar.
3. Dar clic en el botón Reportar Resultado de la Barra de Herramienta.
4. Escoger el tipo de reporte Resultados de la simulación.

Otra manera de llegar a esta funcionalidad.

- 1) Escritorio.
- 2) Dar clic en el programa Sistema Termo Azúcar.
- 3) Dar clic en el Proyecto.
- 4) Escoger la opción Reporte.
- 5) Escoger el tipo de reporte Resultados de la simulación.

### Iteraciones

Clases Válidas	Clases Inválidas	Resultado Esperado	Resultado de la Prueba
1. Se selecciona en Reporte de condensado la opción Datos de la simulación.		El sistema muestre un Reporte con los parámetros adicionales de los módulos.	El sistema muestra un reporte con los parámetros adicionales de los módulos.
	1.1.No se selecciona en Reporte condensado la opción Datos de la simulación)	El sistema muestre un mensaje avisándole al usuario que debe de seleccionar esta opción para obtener este reporte.	El sistema no muestra ningún mensaje avisándole al usuario que debe de escoger la opción Datos de la simulación para hacer un reporte de este tipo.
1.2. Se selecciona e		El sistema muestra un	El sistema muestra un

## DISEÑO, EJECUCIÓN Y EVALUACIÓN DE LAS PRUEBAS

Reporte de indicadores.		reporte con los indicadores globales por áreas y equipos	reporte con los indicadores globales por áreas y equipos
-------------------------	--	--	--

Tabla 3.8. Diseño del caso de prueba Reportar Resultados

### Registro de defectos y dificultades detectados

Elemento	No	No conformidad	Aspecto correspondiente	Etapa de detección Código del CP
<b>La opción Resultados de la simulación del tipo de reporte condensado no es escogida.</b>	1	El sistema acepta esta opción sin mostrarle al usuario un mensaje de que debe de escogerla para realizar un reporte de este tipo.	Anexo 3	Sistema
<b>Falta de ortografía en la palabra Específica</b>	2	En Reporte de Indicadores Globales.	Anexo 3	Sistema

Tabla 3.9. Defectos detectados en la funcionalidad Reportar Resultados

**Nota:** En Reporte Condensado que incluye a las funcionalidades de Reportar datos y Reportar resultados se detectó de manera general que si el usuario no marca ninguna de las dos opciones que brinda el sistema se cierra esta ventana solapándose la función del botón Aceptar con el de Cancelar, por lo que se recomienda que el sistema muestre un mensaje avisándole al usuario que debe de escoger al menos una opción o que dicha ventana no se cierre.

### 3.3. Prueba de Documentación

La Prueba de Documentación al que fue sometido el STA se dividió en dos momentos de gran importancia; el primer momento fue revisar la Ayuda del STA y el Manual de Usuario,

## DISEÑO, EJECUCIÓN Y EVALUACIÓN DE LAS PRUEBAS

el segundo fue verificar todos los entregables con los que debe de contar la carpeta de proyecto.

### 3.3.1. Registro No Conformidades para la Ayuda del STA

La revisión de la Ayuda del STA en su versión 4.0 permitió detectar todos los defectos y dificultades que la misma tiene, logrando así que la Ayuda cuente con una mayor calidad.

A continuación se muestran todos los defectos y dificultades detectados:

Elemento	No Conformidad	Aspecto Correspondiente	Etapas de detección
Introducción	Mal escrita la palabra subrutina. (Ver Anexo 4)	Menú Ayuda/Ayuda STA4.0/Introducción o accionar F1.	Pruebas Documentación a nivel de Sistema.
¿Qué es simular?	En los incisos donde se describen las características del STA las oraciones terminan en (;) y hay un guión al empezar el primer inciso. (Ver Anexo 5)	Menú Ayuda/Ayuda STA4.0/Concepto de Simulación/ ¿Qué es simular?	Pruebas Documentación a nivel de Sistema.
DFP	Falta de ortografía en la palabra termo energético. (Ver Anexo 6)	Menú Ayuda/Ayuda STA4.0/Concepto de Simulación/DFP.	Pruebas Documentación a nivel de Sistema.
DFI	Mal empleo del punto y seguido. (Ver Anexo 7)	Menú Ayuda/Ayuda STA4.0/Concepto de Simulación/DFI.	Pruebas Documentación a nivel de Sistema.
Corrientes de Información.	Falta el punto final en algunas oraciones. (Ver Anexo 8)	Menú Ayuda/Ayuda STA4.0/Concepto de Simulación/Corrientes de Información.	Pruebas Documentación a nivel de Sistema.
Cálculo de los módulos para datos de corrientes de	Falta el punto final en algunas oraciones y mal empleo de los (:). (Ver Anexo	Menú Ayuda/Ayuda STA4.0/Concepto de	Pruebas Documentación a nivel de Sistema.

## DISEÑO, EJECUCIÓN Y EVALUACIÓN DE LAS PRUEBAS

entrada y reciclo.	9)	Simulación/Corrientes de Información /vínculo en esa misma pág. cálculo de corrientes de entrada y reciclo.	
Forma de expresar la composición de las corrientes en términos azucareros.	Faltan algunos puntos finales y en nomenclatura másico está escrito con mayúscula y en otros con minúscula.(Ver Anexo 10)	Menú Ayuda/Ayuda STA4.0/ Concepto de Simulación/Formas de expresar la composición de las corrientes en términos azucareros.	Pruebas Documentación a nivel de Sistema.
Nomenclatura de las corrientes en términos azucareros.	Cambio de letra en una palabra.(Ver Anexo 11)	Menú Ayuda/Ayuda STA4.0/ Concepto de Simulación/Nomenclatura de las corrientes en términos azucareros.	Pruebas Documentación a nivel de Sistema.
Tabla de vapor de agua.	Palabra en bloque, acento de la o entre palabras, mala utilización del punto final. (Ver Anexo 12)	Menú Ayuda/Ayuda STA4.0/ Concepto de simulación/ /Módulos/Módulos generales/turbogenerador1 contrapresión)/vínculo tablas de propiedades del agua.	Pruebas Documentación a nivel de Sistema.



## DISEÑO, EJECUCIÓN Y EVALUACIÓN DE LAS PRUEBAS

Generador de Vapor 1.	Se resaltan dos letras, falta el guión antes de poner el nombre de las figs. ,letras de diferentes estilos, ausencia de punto final, la palabra “cómo” no se acentúa, solo se hace cuando es empleada en preguntas(Ver Anexo 13)	Menú Ayuda/Ayuda STA4.0/ Concepto de Simulación/Módulos /Módulos Generales/Generador de Vapor 1.	Pruebas de Documentación a nivel de Sistema.
Antemperador1	Escritura incorrecta de las palabras desobrerrecalentado Termodinámicas, ausencia de punto final.(Ver Anexo 14)	Menú Ayuda/Ayuda STA4.0/ Concepto de Simulación/Módulos/ Módulos Generales Antemperador1.	Pruebas de Documentación a nivel de Sistema.
Tabla de propiedades termodinámicas del agua.	Algunas oraciones terminan con coma. (Ver Anexo 15)	Menú Ayuda/Ayuda STA4.0/ Concepto de Simulación/Módulos /Módulos Generales/ Antemperador1/vínculo tabla de propiedades termodinámicas del agua.	Pruebas de Documentación a nivel de Sistema.
Intercambiador de calor1 (vapor saturado-jugo).	Ausencia de puntos finales, la palabra subíndice tiene estilo cursivo, estilos de letras diferentes, paréntesis y coma en colores, falta de ortografía en la palabra limpiarlos, la oración termina en coma no en punto final. (Ver Anexo 16)	Menú Ayuda/Ayuda STA4.0/ Concepto de Simulación/Módulos /Módulos Generales/Intercambiador de calor1 (vapor saturado-	Pruebas de Documentación a nivel de Sistema.

## DISEÑO, EJECUCIÓN Y EVALUACIÓN DE LAS PRUEBAS

		jugo).	
Vaso evaporador 1 y 2.	Nombre de las figs. con letras diferentes, falta el guión delante del nombre de las figs., ausencia del punto final, punto antes de la coma, guión debajo de la coma, guión de otro color, acento de la o entre palabras, ausencia de los :,no hay uniformidad al empezar las oraciones, dentro de un subtítulo hay una letra que no está en negrita ,la última palabra está corrida hacia abajo, después del paréntesis hay una raya debajo del punto. (Ver Anexo 17).	Menú Ayuda/Ayuda STA4.0/ Concepto de Simulación/Módulos /Módulos Generales/Vaso evaporador 1 y 2.	Pruebas de Documentación a nivel de Sistema.
Converge2(CONV 2)	Cambio de letras, mal escritas las palabras “mas” y “esta” de acuerdo a como son utilizadas. (Ver Anexo 18)	Menú Ayuda/Ayuda STA4.0/ Concepto de Simulación/Módulos /Módulos Generales/Vaso evaporador 1 y 2/vínculo Converge2 (CONV2).	Pruebas de Documentación a nivel de Sistema.
Comparador	Falta un punto final.(Ver Anexo 19)	Menú Ayuda/Ayuda STA4.0/ Concepto de Simulación/Módulos	Pruebas de Documentación a nivel de Sistema.

## DISEÑO, EJECUCIÓN Y EVALUACIÓN DE LAS PRUEBAS

		/Módulos Generales/Comparador.	
Unión1	Redundancia de la “s”, ausencia de puntos finales, diferente tipo de letra. (Ver Anexo 20)	Menú Ayuda/Ayuda STA4.0/ Concepto de Simulación/Módulos /Módulos Generales/Unión1.	Pruebas de Documentación a nivel de Sistema.
Unión2	No coincide el nombre del elemento cuando se va al documento que corresponde, ausencia de puntos finales y textos resaltados con negrita, letra diferente. (Ver Anexo 21)	Menú Ayuda/Ayuda STA4.0/ Concepto de Simulación/Módulos /Módulos Generales/Unión2.	Pruebas de Documentación a nivel de Sistema.
Cálculo de los módulos para datos de corrientes de entrada y reciclo.	No se utiliza bien los (::), ausencia de puntos finales, estilos de letras diferentes, no existe concordancia entre el adjetivo y sustantivo y mala utilización de la contracción “del”.(Ver Anexos 22)	Menú Ayuda/Ayuda STA4.0/ Concepto de Simulación/Módulos /Módulos Generales/Cálculo para datos de corrientes de entrada y reciclo.	Pruebas de Documentación a nivel de Sistema.
Conectores de 1,2, y 3 salidas C1S, C2S, C3S.	El subrayado de la palabra pseudocorriente es diferente a las de las demás.(Ver Anexo 23)	Menú Ayuda/Ayuda STA4.0/ Concepto de Simulación/Módulos /Módulos lógicos/Conectores de 1,2 y 3 salidas	Pruebas de Documentación a nivel de Sistema.

## DISEÑO, EJECUCIÓN Y EVALUACIÓN DE LAS PRUEBAS

		C1S, C2S, C3S.	
CONVERGE1	Ausencia de puntos finales, falta el (-) delante del nombre de la fig., falta escribir los parámetros adicionales. (Ver Anexo 24)	Menú Ayuda/Ayuda STA4.0/ Concepto de Simulación/Módulos /Módulos lógicos/CONVERGE1.	Pruebas de Documentación a nivel de Sistema.
CONVERGE2(CO NV2)	Cambio de letra, las palabras estas y más se acentúan en el contexto en que se utiliza, ausencia del punto final. (Ver Anexo 25)	Menú Ayuda/Ayuda STA4.0/ Concepto de Simulación/Módulos /Módulos lógicos/CONVERGE2 (CONV2).	Pruebas de Documentación a nivel de Sistema.
Estación de Tachos2	Falta el (-) delante del nombre de la fig., estilo diferente en la palabra "Subíndice". (Ver Anexo 26)	Menú Ayuda/Ayuda STA4.0/ Concepto de Simulación/Módulos /Módulos específicos de crudo/Estación de Tachos2.	Pruebas de Documentación a nivel de Sistema.
Indicadores operacionales en producción de crudo	Ausencia de puntos finales. (Ver Anexo 27)	Menú Ayuda/Ayuda STA4.0/Otros conceptos/ Indicadores operacionales en producción de crudo.	Pruebas de Documentación a nivel de Sistema.
Análisis económico.	Mal escrito grados Celsius, dos palabras en negrita	Menú Ayuda/Ayuda STA4.0/Otros	Pruebas de Documentación a

## DISEÑO, EJECUCIÓN Y EVALUACIÓN DE LAS PRUEBAS

	dentro de un párrafo y ausencia del punto final, párrafos subrayados y con (-) al comenzar, signos de ¿? que no cumplen ninguna función, letra diferente un mismo subrayado de dos colores. (Ver Anexo 28)	conceptos/Búsqueda de problemas/Análisis económico.	nivel de Sistema.
Sistemas de vapor.	Falta de ortografía en generación, distribución, energía, mecánica, eléctrica alimentación, cogeneración, recuperación, tecnológico, líquido, líquida y en la palabra más, esta última de acuerdo al contexto en que es utilizada, dos puntos delante de la oración, espacio entre dos palabras muy grande, faltan los (:) después de la palabra son y mala utilización de los (:).(Ver Anexo 29)	Menú Ayuda/Ayuda STA4.0/Otros conceptos/Búsqueda de problemas/Sistemas de vapor.	Pruebas de Documentación a nivel de Sistema.
Diagrama del proceso.	Palabras escritas en bloque.(Ver Anexo 30)	Menú Ayuda/Ayuda STA4.0/Proceso de producción de azúcar crudo/Diagrama del proceso.	Pruebas de Documentación a nivel de Sistema.
Calores específicos.	Palabras escritas en bloque, falta la tilde en la palabra está de acuerdo al contexto en que se utilizó. ( Ver Anexo	Menú Ayuda/Ayuda STA4.0/Proceso de producción de azúcar	Pruebas de Documentación a nivel de Sistema.

## DISEÑO, EJECUCIÓN Y EVALUACIÓN DE LAS PRUEBAS

	31)	crudo/Propiedades físico – químicas/Calores específicos.	
Elevación del punto de ebullición.	Ausencia del punto final, hay una rayita entre dos palabras.( Ver Anexo 32)	Menú Ayuda/Ayuda STA4.0/Proceso de producción de azúcar crudo/ Propiedades físico – químicas/Elevación del punto de ebullición.	Pruebas de Documentación a nivel de Sistema.
Densidad de jugos, meladuras y mieles.	Palabras escritas en bloque.( Ver Anexo 33)	Menú Ayuda/Ayuda STA4.0/Proceso de producción de azúcar crudo/ Propiedades físico –químicas/ Densidad de jugos, meladuras y mieles.	Pruebas de Documentación a nivel de Sistema.
Reporte condensado.	Le falta el acento a la palabra número.( Ver Anexo 34)	Menú Ayuda/Ayuda STA4.0/Ejemplos de simulación/Ejemplo 1calentador- Evaporador- Control/Reporte condensado.	Pruebas de Documentación a nivel de Sistema.
Enunciado Ejemplo2.	Le falta el acento a la palabra está de acuerdo al contexto en que fue utilizada.( Ver	Menú Ayuda/Ayuda STA4.0/Ejemplos de simulación/	Pruebas de Documentación a nivel de Sistema.

## DISEÑO, EJECUCIÓN Y EVALUACIÓN DE LAS PRUEBAS

	Anexo 35)	Ejemplo2/Enunciado o Ejemplo2.	
Reporte condensado Ejemplo2.	Le falta el acento a la palabra número.( Ver Anexo 36)	Menú Ayuda/Ayuda STA4.0/Ejemplos de simulación/Ejemplo 2/Reporte condensado Ejemplo2.	Pruebas de Documentación a nivel de Sistema.
Enunciado Central A.	Ausencia de los puntos finales.( Ver Anexo 37)	Menú Ayuda/Ayuda STA4.0/Ejemplos de simulación /Ejemplo Central A.	Pruebas de Documentación a nivel de Sistema.
Reporte Condensado Central A.	Letras dentro de una palabra en negrita, falta de ortografía en las palabras números, presión, déficit y evaporación.( Ver Anexo 38)	Menú Ayuda/Ayuda STA4.0/Ejemplos de simulación/Ejemplo Central A/Reporte Condensado Central A.	Pruebas de Documentación a nivel de Sistema.
Reporte Indicadores Central A.	Diferente tipo de letra dentro del reporte, mucho espacio después de los (:).( Ver Anexo 39)	Menú Ayuda/Ayuda STA4.0/Ejemplos de simulación/Ejemplo Central A/Reporte Indicadores Central A.	Pruebas de Documentación a nivel de Sistema.
Nuevo	Falta un punto final.( Ver Anexo 40)	Menú Ayuda/Ayuda STA4.0/Barra de herramientas/archivo/ nuevo.	Pruebas de Documentación a nivel de Sistema.
Cargar	Falta los (:) y los puntos finales.( Ver Anexo 41)	Menú Ayuda/Ayuda STA4.0/Barra de herramientas/Archiv	Pruebas de Documentación a nivel de Sistema.

## DISEÑO, EJECUCIÓN Y EVALUACIÓN DE LAS PRUEBAS

		o/Cargar	
Salvar	Falta los (:) y los puntos finales.( Ver Anexo 42)	Menú Ayuda/Ayuda STA4.0/Barra de herramientas/archivo/salvar	Pruebas de Documentación a nivel de Sistema.
Copiar	No está bien aplicado el estilo negrito.(Ver Anexo 43)	Menú Ayuda/Ayuda STA4.0/Barra de herramientas/Edición/copiar.	Pruebas de Documentación a nivel de Sistema.
Pegar	Falta un punto final.(Ver Anexo 44)	Menú Ayuda/Ayuda STA4.0/Barra de herramientas/Edición/pegar.	Pruebas de Documentación a nivel de Sistema
Limpiar diagrama	Falta un punto final.(Ver Anexo 45)	Menú Ayuda/Ayuda STA4.0/Barra de herramientas/Edición/ Limpiar diagrama.	Pruebas de Documentación a nivel de Sistema
Paletas de módulos.	Falta un punto final.(Ver Anexo 46)	Menú Ayuda/Ayuda STA4.0/Barra de herramientas/Ver/Paletas de módulos.	Pruebas de Documentación a nivel de Sistema
Vista previa	Falta un punto final.(Ver Anexo 47)	Menú Ayuda/Ayuda STA4.0/Barra de herramientas/Ver/Vista previa.	Pruebas de Documentación a nivel de Sistema
Simular.	Falta un punto final.(Ver Anexo 48)	Menú Ayuda/Ayuda STA4.0/Barra de herramientas/Proyecto/Simular.	Pruebas de Documentación a nivel de Sistema
Botones de la barra de	Falta de ortografía en las palabras módulo y	Menú Ayuda/Ayuda STA4.0/Barra de	Pruebas de Documentación a



## DISEÑO, EJECUCIÓN Y EVALUACIÓN DE LAS PRUEBAS

herramientas.	adicionarán, esta última de acuerdo al contexto en que fue utilizado. .(Ver Anexo 49)	herramientas/Boton es de la barra de herramientas.	nivel de Sistema
Errores	Falta de ortografía en las palabras módulo, está y más estas últimas de acuerdo al contexto en que fueron utilizadas. Ver Anexo 50)	Menú Ayuda/Ayuda STA4.0/Errores	Pruebas de Documentación a nivel de Sistema

Tabla 3.10.Registro No Conformidades de la Ayuda

### 3.3.2. Registro No Conformidades para el Manual de Usuario del STA

En el Manual de Usuario se detectó la dificultad que es recogida en el Registro de No Conformidades correspondiente.

Elemento	No conformidad	Aspecto correspondiente	Etapas de detección
Instalando el simulador.	Los mensajes aparecen en dos idiomas. (Ver Anexo 51 )	Menú Ayuda/manual de usuario	Pruebas de Documentación.
Instalando Microsoft .NET Framework 1.1.	Los mensajes aparecen en dos idiomas(Ver Anexo 52)	Menú Ayuda/manual de usuario	Pruebas de Documentación.

Tabla 3.11.Registro No Conformidades del Manual de Usuario

### 3.3.3. Carpeta del Proyecto

Se definió que la carpeta de proyecto debe de contar con los siguientes entregables: Modelo del Negocio, Modelo de Diseño, Modelo de Despliegue, Modelo de Implementación, Modelo de Prueba, Documento Visión, Modelo del Sistema, Modelo de Objeto del Negocio, Lista de Riesgo, Glosario de Términos, Prototipo Interfaz de Usuario, Especificación de Requisitos Funcionales y Requisitos No Funcionales, Ayuda, Manual de Usuario y los ficheros Config.ini, Dotnetfx.exe, Settings.ini, STA 4.0.msi,Setup.exe.

De todos estos entregables y ficheros se verificó que el STA cuenta solamente con los que se mencionan a continuación, por lo que se sugiere completar la Carpeta de Proyecto para que el simulador cuente con suficientes documentos que lo describan.

1. Especificación de Requisitos Funcionales y Requisitos No Funcionales.
2. Ayuda.
3. Manual de Usuario.
4. Config.ini.
5. Dotnetfx.exe.
6. Settings.ini.
7. STA 4.0.msi.
8. STA 4.0.msi.
9. Setup.exe.

### **3.4. Definición de los casos de pruebas de Caja Blanca**

Se le realizó el diseño de prueba utilizando el método de caja blanca a la funcionalidad Insertar Información, esta funcionalidad se escogió por pertenecer al caso de uso con ese mismo nombre, el cual tiene una prioridad crítica para el simulador. Para entender mejor como ocurre este proceso a continuación se describe el mismo.

Todos los datos que son introducidos por el usuario se recogen en el método SetGetValores, el cual cumple con las funciones de crear y actualizar estos datos, que finalmente se guardan en un objeto que fue creado por el método antes mencionado para luego ser enviado a los módulos para su manipulación a través de los métodos MostrarVentana y btnAceptarClick; siendo estos a los que se le realizó la prueba de caja blanca por ser los responsables de la inserción de la información.

#### **3.4.1. Grafos de Flujo para MostrarVentana**

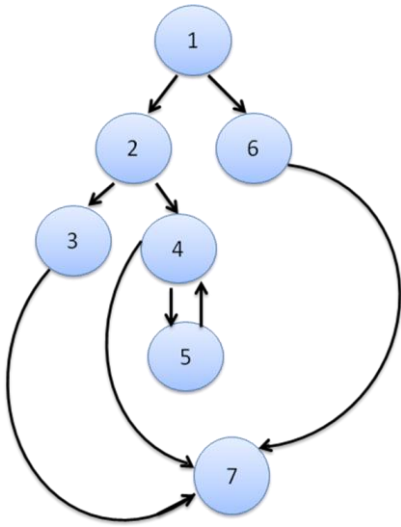


Fig.3.1.Grafo de Flujo para el método MostrarVentana

## Complejidad ciclomática de MostrarVentana

- I.  $V(G) = \text{Número de regiones}$   
 $V(G) = 4$
- II.  $V(G) = \text{Número de aristas} - \text{números de nodos} + 2$   
 $V(G) = 4$
- III.  $V(G) = \text{Número de nodos predicados} + 1$   
 $V(G) = 4$

## Caminos Básicos de Calcular Mudo.

### CB 1

1-2-3-7

### CB 2

1-2-4-7

### CB 3

1-2-4-5-4-7

### CB 4

1-6-7

## 3.4.2. Grafos de Flujo para btnAceptarClick

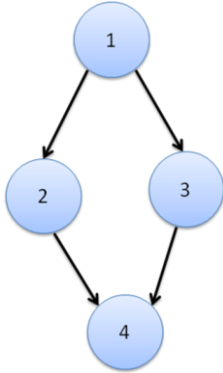


Fig.3.2.Grafo de Flujo para el método btnAceptarClick

### Complejidad ciclomática de btnAceptarClick.

- I.  $V(G) = \text{Número de regiones}$   
 $V(G) = 2$
- II.  $V(G) = \text{Número de aristas} - \text{números de nodos} + 2$   
 $V(G) = 2$
- III.  $V(G) = \text{Número de nodos predicados} + 1$   
 $V(G) = 2$

### Caminos Básicos de btnAceptarClick.

#### CB 1

1-2-3

#### CB 2

1-3-4

Para observar si se detectaba un defecto durante la realización de la prueba, se utilizó los datos pertenecientes al módulo Intercambiador de Calor, estos son:

- Caída de presión jugo: 0.1psia
- Caída de presión vapor: 0.1psi
- Temperatura del jugo a la salida: 115 °C
- Presión vapor de calentamiento jugo: 0.1psia
- Diámetro de los tubos: 0.15 pies
- Longitud de los tubos: 19.68 pies

- Caída de los tubos por pase: 43
- Cantidad de pases: 0
- Cantidad de intercambiadores: 2

Con los mismos se comprobó que luego de creado el objeto por el método SetGetValores, si este no es null (vacío) pasa por los caminos básicos 1-2-3-7;1-2-4-7 y 1-2-4-5-4-7 del Grafo de Flujo Mostrar Ventana para luego seguir por el camino básico 1-2-4 del Grafo de Flujo btnAceptarClick. En caso contrario, o sea, que el objeto creado se encuentre vacío los datos se irían por el camino básico 1-6-7 del Grafo de Flujo MostrarVentana para seguir por el camino básico 1-3-4 del grafo de Flujo btnAceptarClick.

Luego de concluida esta prueba se comprobó que no existe defectos ni errores en los diseños procedimentales escogidos.

### 3.5. Resumen de Evaluación

Lugo de evaluar cada funcionalidad y entregable por separado, se considera que se han detectado los defectos de cada una de ellas, con el objetivo de que el producto adquiriera una mayor calidad antes de salir a su destino final: ser utilizado por especialistas en fábricas de azúcar crudo.

Defectos en los Reportes:

- Solapamiento de la funcionalidad del botón aceptar con el de cancelar en reporte condensado cuando no se escogida ninguna en las dos opciones por el usuario.
- Error ortográfico en reporte de indicadores globales.

Defectos comunes en Identificar módulo e Identificar corriente:

- ✓ Entrada de caracteres extraños en el campo de introducir nombre(@,#,\$...etc.).
- ✓ Ausencia de mensajes de errores si el usuario deja uno o los dos campos en blanco (campo introducir nombre e introducir id).

Defectos más reiterados encontrados en la Ayuda:

- ❖ Falta de ortografía en varias palabras.
- ❖ Palabras escritas en bloque.
- ❖ Ausencia del punto final al finalizar una oración.
- ❖ Reemplazo del punto final por coma o punto y coma.

Defecto en el Manual de usuario:

- Mensajes de cómo instalar el software en dos idiomas.

A las funcionalidades Conversor de unidades e Insertar información a las que se les realizó pruebas utilizando métodos de caja negra y caja blanca respectivamente no se le detectó ningún defecto ni dificultad.

### 3.6. Sugerencias

- Reporte condensado: Cuando no se marque ninguna de las dos opciones presentada y se de aceptar no se solape la función del botón Aceptar con el de Cancelar, sino que el sistema muestre un mensaje avisándole al usuario que debe escoger al menos una opción.
- Identificar módulo e Identificar corriente: El sistema muestre mensajes de aviso cuando se dejen campos en blanco o se entren caracteres extraños(@,#,\$...etc.).
- Ayuda y Manual de usuario: Remitirse a las Plantillas de no Conformidades correspondientes para eliminar los defectos que se encuentran reflejados en las mismas.

### 3.7. Evaluación de las actividades implementadas anteriormente

Con la realización de todas las actividades previstas (diseño de casos de pruebas y aplicación de las mismas) se obtuvo un buen resultado a pesar de no contar con un equipo de pruebas completo. A medida que se fueron realizando las actividades se encontraron varias dificultades en el software de diversa índole, esto le permite al equipo de desarrollo eliminar los defectos detectados y así elevar la calidad del producto.

### 3.8. Conclusiones

En este Capítulo se realizaron las pruebas definidas para el software Simulador Termo Azúcar versión 4.0 mediante casos de pruebas, los que posibilitaron llevar a cabo un proceso de detección de defectos de forma exitosa, logrando que se obtuviera y garantizara una mayor calidad del mismo. Además se pudo arribar a algunos consejos de gran importancia a la hora de diseñar y ejecutar los casos de prueba; estos son:

El programador debe evitar probar sus propios programas ya que desea (consciente o inconscientemente) demostrar que funcionan sin problemas. Además que es normal que

## DISEÑO, EJECUCIÓN Y EVALUACIÓN DE LAS PRUEBAS

las situaciones que olvidó al crear el programa queden de nuevo olvidadas al crear los casos de prueba. Al generar casos de prueba se debe incluir tantos datos de entrada válidos y esperados como no válidos e inesperados.<sup>19</sup>

---

<sup>19</sup>Collado, M., pruebas de Software.2003 [Disponible en <http://lms.ls.fi.upm.es/ftp/ed2/0203/Apuntes/pruebas.ppt>].

### CONCLUSIONES

Luego de dar cumplimiento a los objetivos y tareas trazadas para lograr una investigación exitosa se arribó a las siguientes conclusiones:

1. Se realizó un estudio detallado de todas las Pruebas existentes para definir cuáles eran las que se adecuaban más a las características del STA.
2. Se demostró la importancia que tiene llevar a cabo un proceso de pruebas a un determinado software para lograr una mayor calidad del mismo.
3. Se definió un correcto Plan de Pruebas, definiendo de manera concreta las pruebas que se le iban a realizar al producto.
4. Se realizó diferentes diseños de pruebas a varias funcionalidades, encontrando diversos defectos en las mismas.
5. Se probó la existencia de defectos en el STA ya que nunca había sido sometido a un proceso de pruebas.
6. Se conoció como un producto que no contenga defectos permite elevar su aceptación en el mercado y elevar el prestigio de la organización y personas que intervinieron en la confección del mismo.



### RECOMENDACIONES

Al concluir la presente investigación y luego de haber cumplido con los objetivos previstos se recomienda al equipo de trabajo de simulación:

- Eliminar todos los defectos detectados durante el proceso de prueba realizado al simulador y que se encuentran recogidos en este documento y en las plantillas de diseño de casos de pruebas y registro de no conformidades.
- Desarrollar un módulo de seguridad para el producto, pues es un software concebido para realizar la evaluación en sistemas termo-energéticos en fábricas de azúcar crudo, lo que un mal uso del mismo puede tener un impacto desfavorable en la economía de la fábrica y por ende en la del país.
- Generar todos los entregables que faltan y con los que debe de contar la carpeta de proyecto.
- Establecer un equipo de prueba dentro del proyecto con el objetivo de que el mismo se encargue de detectar los defectos de manera simultanea al trabajo que va realizando el equipo de desarrollo.
- Realizar nuevamente un proceso de detección de defectos y dificultades al STA y que este proceso sea ejecutado por un equipo de prueba, donde estén bien definidos todos los roles (Administrador de pruebas, Analista de pruebas, Diseñador de Pruebas y Probador) para lograr un trabajo más efectivo y una mayor calidad en el producto.

## BIBLIOGRAFÍA

### Bibliografía citada

1. Hernandez, León, Rolando Alfredo y Coello, González, Sayda. EL PARADIGMA CUANTITATIVO DE LA INVESTIGACIÓN CIENTÍFICA [digital. ISBN: 959-16-0343-6 Ciudad de la Habana, Noviembre del 2002.
2. Ídem referencia 1.
3. Ídem referencia 1.
4. Hernández, Meléndrez, Dra. Edelsys. *Cómo Escribir una Tesis.* [digital. Escuela Nacional de Salud Pública, Cuba. Mayo, 2007.
5. (ISO 8402: 1984).
6. Pressman, Roger. 2002. Ingeniería de Software. Un enfoque Práctico. 5ta Edición. s.l. : McGraw-Hill, 2002.
7. Peña, Carlos Martínez. 2002. Departamento de Ciencias Matemáticas e Informáticas. [En línea].
8. Autores Colectivo de. 2004. Departamento de Educación y Cultura. SCIENCE.
9. Ídem a la referencia 6.
10. Autores, Colectivo de. 1999. eumet.net. [En línea] 1999.
11. Autores, Colectivo de. 2006. UNAD. [En línea] <http://calidad.unad.org>, 2006.
12. Ídem referencia 6.
13. Dorado, Carlos. 2003. Ingeniería de Software Educativo. *www.fi.net.ar*. [En línea] 2003.
14. Rico, Ariel Edgar Serrano. 2003. [En línea] 2003.
15. Autores, Colectivo de. 2001. [En línea] 2001.

16. Ídem referencia 6.

17. Ídem referencia 6.

18. Gonzalez, C. Un Plan de Pruebas Exitoso. 2002 [Disponible en <http://www.americaxxi.cl/modules.php?name=News&file=article&sid=20>].

19. Collado, M., pruebas de Software. 2003 [Disponible en: <http://lml.ls.fi.upm.es/ftp/ed2/0203/Apuntes/pruebas.ppt>].

### **Bibliografía consultada**

1. Hernandez, León, Rolando Alfredo y Coello, González, Sayda. EL PARADIGMA CUANTITATIVO DE LA INVESTIGACIÓN CIENTÍFICA [digital. ISBN: 959-16-0343-6 Ciudad de la Habana, Noviembre del 2002.

2. Ídem referencia 1.

3. Ídem referencia 1.

4. Hernández, Meléndrez, Dra. Edelsys. Cómo Escribir una Tesis. [digital. Escuela Nacional de Salud Pública, Cuba. Mayo, 2007.

5. (ISO 8402: 1984).

6. Pressman, Roger. 2002. Ingeniería de Software. Un enfoque Práctico. 5ta Edición. s.l. : McGraw-Hill, 2002.

7. Peña, Carlos Martínez. 2002. Departamento de Ciencias Matemáticas e Informáticas. [En línea].

8. Autores Colectivo de. 2004. Departamento de Educación y Cultura. SCIENCE.

9. Ídem a la referencia 6.

10. Autores, Colectivo de. 1999. eumet.net. [En línea] 1999.

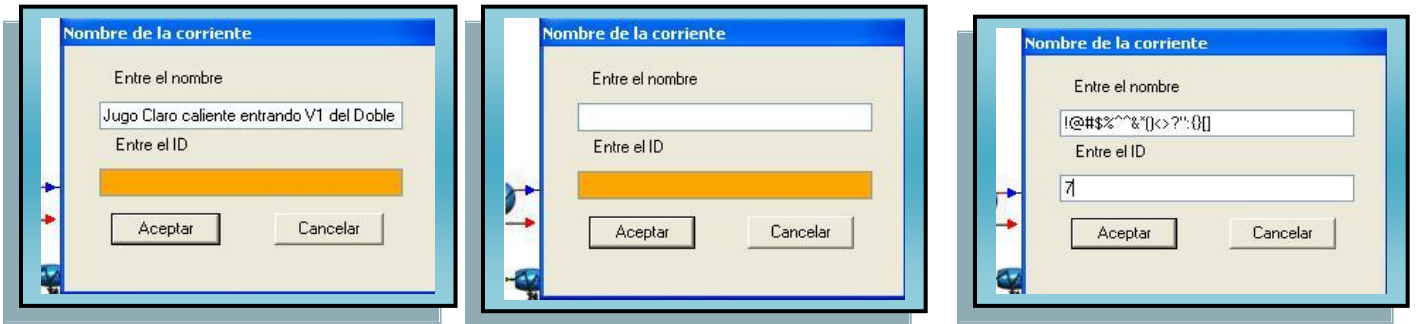
11. Autores, Colectivo de. 2006. UNAD. [En línea] <http://calidad.unad.org>, 2006.

12. Ídem referencia 6.
13. Dorado, Carlos. 2003. Ingeniería de Software Educativo. www.fi.net.ar. [En línea] 2003.
14. Rico, Ariel Edgar Serrano. 2003. [En línea] 2003.
15. Autores, Colectivo de. 2001. [En línea] 2001.
16. Ídem referencia 6.
17. Ídem referencia 6.
18. Gonzalez, C. Un Plan de Pruebas Exitoso. 2002 [Disponible en <http://www.americaxxi.cl/modules.php?name=News&file=article&sid=20>].
19. Collado, M., pruebas de Software. 2003 [Disponible en: <http://lml.ls.fi.upm.es/ftp/ed2/0203/Apuntes/pruebas.ppt>].
20. B., J. Goodenough. Team Software Process Reliability Results. 2000 [Disponible en: <http://www.softwaretechnews.com/stn3-4/teamspi.html>].
21. Beizer, B. Software Testing Techniques. 2a ed. 1990.
22. Booch, G., Rumbaugh, J., Jacobson, I. El Proceso Unificado de Desarrollo de Software. La Habana, Cuba 2004.
23. Colectivo de Profesores. Prueba. 2004 [Disponible en: [http://ucimedia.uci.cu/teleclases/2005-2006/2do-sem/mc/ingenieria\\_de\\_software\\_2\\_04-05/conf3](http://ucimedia.uci.cu/teleclases/2005-2006/2do-sem/mc/ingenieria_de_software_2_04-05/conf3)].
24. Colectivo de Profesores. Fase de Elaboración. Flujo de trabajo de prueba. 2005 [Disponible en: [http://ucimedia.uci.cu/teleclases/2005-2006/2do-sem/3er/ingenieria\\_de\\_software\\_2/conf3](http://ucimedia.uci.cu/teleclases/2005-2006/2do-sem/3er/ingenieria_de_software_2/conf3)].
25. Colectivo de Profesores. Fase de Construcción. Flujo de trabajo de prueba. 2005 [Disponible en: [http://ucimedia.uci.cu/teleclases/20052006/2dosem/3er/ingenieria\\_de\\_software\\_2/conf4](http://ucimedia.uci.cu/teleclases/20052006/2dosem/3er/ingenieria_de_software_2/conf4)].

26. Colectivo de Profesores. Culminación de la Fase de Elaboración. Flujo de trabajo de Prueba. 2005
28. Gonzalez,C.,Un Plan de Pruebas Exitoso.2002 [Disponible en <http://www.americaxxi.cl/modules.php?name=News&file=article&sid=20>].
29. Humphrey, W. S. Introducción al Proceso de Software en Equipo. 2001.
30. Humphrey, W. S. Introducción al Proceso Software Personal. Madrid 2001.
31. Jacobson, I., Booch, G. y Rumbaugh, J. El Proceso Unificado de Desarrollo de Software. 2000.
32. Mañas, J. Prueba de Programas. 1994 [Disponible en: <http://www.lab.dit.upm.es/~lprg/material/apuntes/pruebas/testing.htm#sA>].
33. Marin M. Consideraciones sobre el proyecto de informatización de la Atención Primaria de Salud.  
Editorial. Revista Habanera de Ciencias Médicas. Vol 3. No. 10. Año 2004. [Disponible en: [http://www.ucmh.sld.cu/rhab/editorial\\_rev10.htm](http://www.ucmh.sld.cu/rhab/editorial_rev10.htm)]
34. Marin M. Capacitación en el área de la Informática en Salud. Revista Cubana de Informática Médica (RCIM).
35. Montes de Oca, C. V. Team Software Process (TSP): Integración de Equipos de Desarrollo de Alto Rendimiento. 2007 [Disponible en: <http://www.emagister.com/el-team-software-process-tsp-cursos-1052582.htm>]
36. Padrón, L. A., Informática para la atención primaria de salud. VI Congreso Internacional de Informática en Salud. 2007
37. Pressman, R. S. Ingeniería del Software. Un enfoque práctico.5 ed. 2001, Madrid.
38. Pressman, R. S. Ingeniería del Softawre. Un enfoque práctico. La Habana, Cuba 2005.
39. Roca, M.V., Pruebas Funcionales. Aspectos relevantes.2006 [Disponible en: <http://www.greensqa.com/archivos/Art02-PlaneacionPruebasFuncionales.pdf>].

## ANEXOS

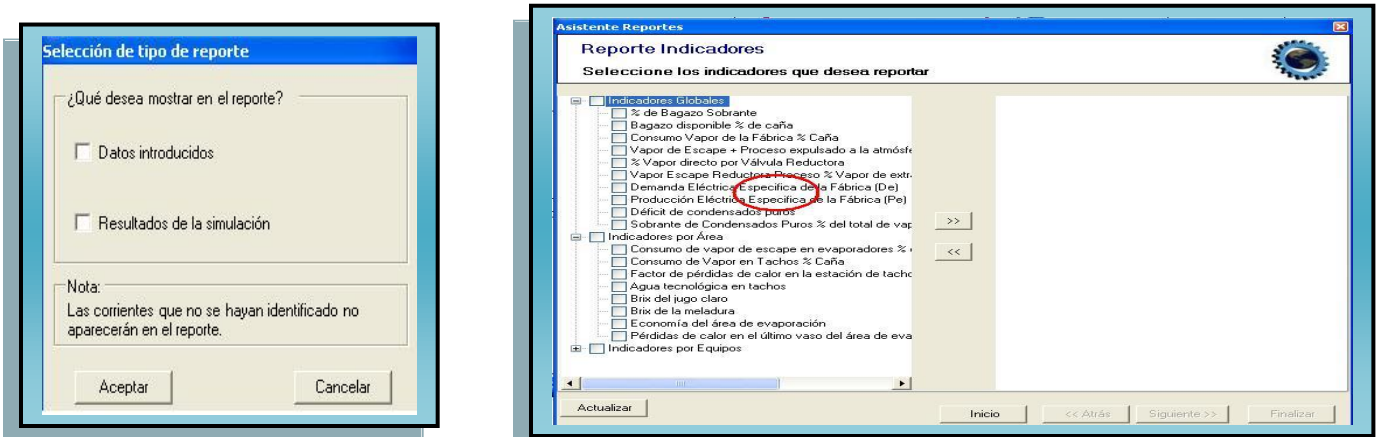
### Anexo 1. Defectos encontrados en la funcionalidad Identificar Corriente.



### Anexo 2. Defectos encontrados en la funcionalidad Identificar Módulo.



### Anexo 3. Defectos encontrados en la funcionalidad Reportar Datos y Resultados.



#### Anexo 4. Defecto encontrado en la Introducción de la Ayuda del STA.

La información inicial requerida consiste en el flujo másico, composición, temperatura y presión de las corrientes de entrada y reciclo, datos específicos de los equipos simulados y el orden de cálculo de los módulos en reciclos. Se cuenta con un grupo de subrutinas específicas relacionadas con las evaluaciones de las propiedades físicas y termodinámicas de las corrientes de jugo, meladura, mieles, azúcar (entalpía, densidad, capacidad calorífica y elevación del punto de ebullición), agua y vapor de agua (tablas de vapor).

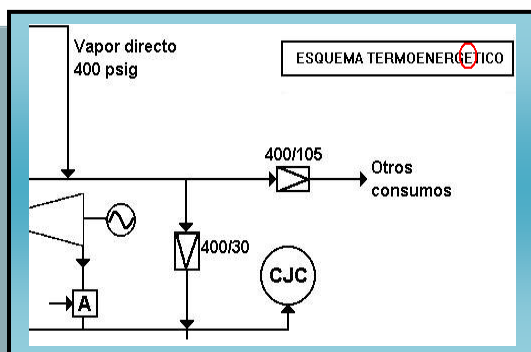
#### Anexo 5. Defecto encontrado en la sección ¿Qué es simular? de la Ayuda del STA.

**Un Simulador del proceso de producción de azúcar crudo es el TermoAzúcar (STA 4.0) cuyas principales características son:**

- a) orientado básicamente al proceso de producción de azúcar crudo.
- b) se asume que tanto en la parte del proceso continuo como en la discreta se alcanza un comportamiento equivalente al estado estacionario.
- c) se emplea fundamentalmente un enfoque modular donde las ecuaciones modelan un Equipo o Subproceso. En algunos Módulos, que representan subprocesos con muchos equipos y una tecnología compleja como sucede en las Estaciones de Tachos, Cristalizadores y Centrifugas, se puede considerar que la modelación es del tipo "orientada a ecuaciones".
- d) a cada Módulo le corresponde un Modelo Matemático con su respectiva y única selección de variables independientes.
- e) la Biblioteca de Módulos contiene múltiples Módulos que permite representar, no solo todos los equipos y subprocesos, sino también emplear diferentes Módulos para calcular un mismo tipo de equipo o subproceso.
- f) los Modelos son fundamentalmente Determinísticos, formados por expresiones analíticas que representan leyes químicas y físicas, aunque hay algunos Probabilísticos como las correlaciones estadísticas para la determinación de propiedades físicas y químicas.
- g) se emplea un enfoque sistémico, pues se puede manejar con relativa facilidad gran cantidad de información.
- h) existe la necesaria flexibilidad para representar integralmente todo tipo de Esquemas Termoenergéticos de Fábricas de azúcar crudo.
- i) el enfoque de cálculo más empleado es el de Evaluación (Vaso Evaporador, Turbogenerador); aunque también hay Módulos con enfoque de Diseño (Estación de Tachos).

El STA no está orientado a la Simulación de los Sistemas de Control Automático, caracterizados por modelos básicos generalmente del tipo "Señal de Entrada (Input) - Caja Negra - Señal de Salida (Output)".

#### Anexo 6. Defecto encontrado en la sección DFP de la Ayuda del STA.



## Anexo 7. Defecto encontrado en la sección DFI de la Ayuda del STA.

### Diagrama de Flujo de Información (DFI)

El DFI es una representación gráfica del sentido en que fluye la Información inicial y la que se origina durante los cálculos. El DFI está formado por Módulos de Cálculo que representan matemáticamente lo que ocurre en los Equipos y/o Subprocesos y las Corrientes o Flujos de Informaciones que entran y salen de los Módulos.

Como es de esperar para la confección de DFI se requiere del conocimiento del DFP. Debe tenerse en cuenta que no siempre el sentido de flujo de la información tiene que coincidir con el de los flujos de materiales y energía en el proceso. Por ejemplo, En un Módulo, que representa un Intercambiador de Calor, donde lo que interesa es calcular la cantidad de Vapor Consumido se tiene:

## Anexo 8. Defecto encontrado en la sección Corriente de información de la Ayuda del STA.

### Módulos que simulan las entradas y salidas del sistema:

Son de dos tipos:

Entrada o salidas de Vapor o Agua

Entrada o salidas de Materiales Azucarados

En el caso de los datos de los módulos de las entradas ver [cálculo de corrientes de entrada y reciclo](#).

Y en el caso de los módulos de las corrientes de salida los valores son calculados en la simulación.

### Pseudocorrientes:

Es un caso especial de corrientes de entrada, salida o reciclo y no es más que un artificio computacional que permite, en determinados módulos, hacer que el DFI se parezca lo más posible al DFP.

Las pseudocorrientes se crean para que el sentido de una corriente en el DFI sea el mismo de cómo aparece en el proceso y por ende de como se dibuja en el DFP.

O sea, en algunos módulos (Intercambiador de calor 4, Atemporador 1 y Turbogenerador 1) se dibujan como si entraran al módulo cuando en realidad salen calculadas. Sin embargo en el módulo Generador de vapor 3 la pseudocorriente "sale" pues el vapor producido "sale" del equipo (DFP). Pero "entra" al módulo que representa el equipo para poder hacer el cálculo (DFI).

La conexión de la pseudocorriente se hace mediante los módulos Conectores de 1, 2 y 3 salidas (C1S, 2S y 3S).

## Anexo 9. Defectos encontrados en la sección Cálculo de los módulos para datos de corrientes de entrada y reciclo de la Ayuda del STA.

Los valores correspondientes a las corrientes de entrada se suministran como datos en los módulos que representan a las mismas. Hay varios tipos de módulos siendo estos:

Para un Simple Efecto  $ECO = 0,90 \pm 0,05$ ; pues generalmente la temperatura del jugo claro es menor que la de la cámara de evaporación

### Nomenclatura para ambas alternativas:

FMJE = Flujo másico de jugo entrando al vaso ó múltiple  
 FMJS = Flujo másico de jugo saliendo al vaso ó múltiple  
 BXJE = Brix del jugo entrando  
 BXJS = Brix del jugo saliendo  
 ECO = Economía del Simple Efecto o del Múltiple.



-Cuando se conocen los valores absolutos de temperatura(°C), presión (psia) y los flujos másicos de los componentes (libras/hora), aunque estas unidades pueden ser cambiadas usando el conversor de unidades

-A partir de datos azucareros tales como: peso de la corriente por ciento del peso de la caña, Brix, Pol y Pureza. Ver [cálculo de las corrientes de entrada y reciclo](#).

**Existen dos variantes para entrar los datos por composición:**

1- Se conoce el flujo total de la corriente de caña, entonces se pide entrar el flujo total de la corriente como fracción de la corriente de caña.

$$\text{FMTTotal} = \text{FMTTotalcaña} * \text{fracción}$$

2- No se conoce el flujo de la corriente de caña y se entra directamente el flujo de la corriente (FMTTotal)

**Teniendo el flujo total de la corriente (FMTTotal), se tienen dos alternativas:**

1- Corriente de bagazo:

Se dan como datos: %Fibras, Pol y Humedad

$$\text{FMAgua} = \text{Humedad} * \text{FMTTotal} / 100.$$

$$\text{FMSSA} = \text{Pol} * \text{FMTTotal} / 100.$$

$$\text{Brix} = 100 - \% \text{Fibra} - \text{Humedad}.$$

$$\text{FMSSNA} = (\text{Brix} - \text{Pol}) * \text{FMTTotal} / 100.$$

$$\text{FMFibra} = \% \text{Fibra} * \text{FMTTotal} / 100.$$

$$\text{FMNFibra} = \text{FMTTotal} - (\text{FMAgua} + \text{FMSSA} + \text{FMSSNA} + \text{FMFibra}).$$

2- Otras corrientes azucaradas, sin fibras ni insolubles no fibra:  
Se dan como datos Brix y Pureza

## Anexo 10. Defectos encontrados en la sección Forma de expresar la composición de las corrientes en términos azucareros de la Ayuda del STA.

### Formas de expresar la composición de las corrientes en términos azucareros

FMSST = Flujos Másicos de SSSac + SSNS  
 FMLSI = Flujo Másico de Líquido (Jugo) sin Insolubles = Flujos Másicos de (Agua + Sólidos Solubles)  
 Totales = FMAgua + FMSST  
 $\text{Brix} = (\text{FMSST} / \text{FMLSI}) * 100 = [\text{Componentes (7) + (8)}] * 100 / [\text{Componentes (6 + 7 + 8)}]$   
 $\text{Pol} = (\text{FMSSSac} / \text{FMLSI}) * 100 = [\text{Componentes (7)}] * 100 / [\text{Componentes (6 + 7 + 8)}]$   
 $\text{Pureza} = (\text{FMSSSac} / \text{FMSST}) * 100 = [ (7) / [ (7) + (8) ] ] * 100 = (\text{Pol} / \text{Brix}) * 100$   
 $\% \text{Fibra} = (\text{FMFibra} / \text{FMTTotal}) * 100 = [\text{Componente (9)}] * 100 / [\text{Componente (3)}]$   
 $\% \text{Insolubles No Fibra} = [\text{Componente (10)}] * 100 / [\text{Componente (3)}]$   
 $\text{FMTTotal} = \text{FMLSI} + \text{FMFibra} + \text{FMINF}$   
 $\text{FMAgua} = \text{FMTTotal} - \text{Flujos Másicos de (SSSac + SSNS + Fibra + Insolubles no Fibra)}$   
 $\text{Humedad} = (\text{FMAgua} / \text{FMTTotal}) * 100 = (\text{Componente (6)} * 100) / (\text{Componente (3)})$

**NOMENCLATURA:**

FMSSSac = Flujo Másico de Sólidos Solubles Sacarosa  
 FMSSNS = Flujo Másico de Sólidos Solubles No Sacarosa  
 FMSSA -> Flujo Másico de sólidos solubles azúcares.  
 FMSSNA -> Flujo másico de sólidos solubles no azúcares.  
 FMFibras -> Flujo másico de fibras.  
 FMNFibras -> Flujo másico de insolubles no fibras  
 FMAgua = Flujo másico de agua.

## Anexo 11. Defectos encontrados en la sección Nomenclatura de las corrientes en términos azucareros de la Ayuda del STA.

Flujo Total de Jugo = FMJugo  
 Flujo de Agua en Jugo = FMAguaJugo  
 Flujo de Sacarosa en Jugo = FMSacJugo  
 Flujo Sólidos Soluble no Sacarosa en Jugo = FMSSNSJugo  
 Flujo Insoluble no Fibra en Jugo = FMINFJugo  
 Flujo Fibra en Jugo = FMFibraJugo  
 Peso Jugo % Peso Bagazo = Jugo % Caña  
 Brix del Jugo = BrixJugo  
 Pureza Jugo = PUJugo  
 Pol Jugo = PolJugo  
 % Insolubles no Fibra en jugo = FMINFJugo  
 % Fibra en Jugo = % FibraJugo  
 Tasa Horario Caña Molina = THCaña

## Anexo 12. Defectos encontrados en la sección tabla de vapor de agua de la Ayuda del STA.

Además de la mencionada Tabla de Propiedades del Agua está disponible la siguiente **correlación** que permite, para condiciones de saturación, calcular la Presión (PRESS) si se conoce la Temperatura (TEMP):  

$$PRESS = T * (((0.29033 * 10^{-8} * T + 58.39654 * 10^{-6} * T - 104.198 * 10^{-4}) * T + 80.4261 * 10^{-2}) - 21.11649)$$

Donde la Humedad debe expresarse como fracción. Es una variable independiente..

En el caso de Vapor Húmedo se da un solo valor, la Humedad **o** la Calidad.  
 Como las fracciones de ambos son igual a 1 cuando se da la Humedad se calcula la Calidad o viceversa.

## Anexo 13. Defectos encontrados en la sección Generador de vapor1 de la Ayuda del STA.

El Valor Calórico Bruto Bagazo (VCB) depende de la Humedad que tenga el Bagazo, la Entalpia del Vapor Directo de la Presión y Temperatura de trabajo del GV y la Temperatura del Agua de Alimentación de si hay suficientes condensados puro o si se tiene que usar condensados contaminados u otros tipo de agua con temperaturas generalmente más bajas.

Los valores del  $IG_{BRUTO}$  en la Tabla 2, como los de la Tabla de Rendimientos, corresponden a la operación a 100 % de la Carga Nominal y de que las condiciones mecánicas del equipo sean las adecuadas.

Por tanto el valor del Rendimiento Bruto, que se da **como dato** en el módulo GV1, debe concordar con un valor que se corresponda con los posibles del Generador de Vapor específico que se simule. Valor que puede ser seleccionado a partir de la Tabla 1, donde aparecen los Rendimientos (Eficiencias) Brutos Máximos para Generadores de Vapor, hechos o montados en Cuba, que estén en buenas condiciones mecánicas y que se operan a carga nominal.

$$5) \quad VCIBag = [4250 - (4850 * FHumedadBag)] * 1,8, \quad BTU/lb$$

En el DFI del módulo GV1, que es la representación del flujo de las corrientes de información vinculadas a los cálculos que se realizan en el módulo, "entra" una corriente (Pseudocorriente) de Vapor que equivale numéricamente a la Demanda de Vapor de la Fábrica, que debe ser calculada con anterioridad. Y "sale" la corriente de información de Bagazo Sobrante, equivalente a la diferencia calculada entre los flujos máxicos de la corriente de Bagazo Disponible y el Consumido para producir la cantidad de vapor demandado.



Figura 1. Generador de Vapor que usa Bagazo Tipico.

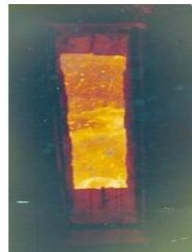
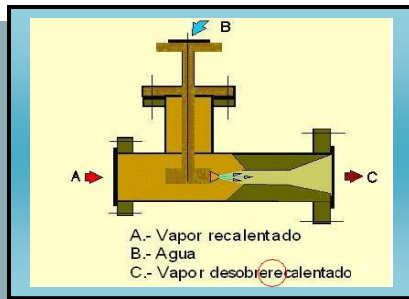


Figura 2. Combustión del Bagazo

### Anexo 14. Defectos encontrados en la sección Antemperador 1 de la Ayuda del STA.



#### ECUACIONES:

1.  $H_{vsc}$  = función (  $P_{vsc}$ ,  $T_{vsc}$  ) Ver [Tabla de Propiedades Termodinámicas del Agua](#)
2.  $P_{vs} = P_{vsc}$
3.  $H_{vs}$  = función (  $P_{vs}$  ) Ver [Tabla Propiedades Termodinámicas del Agua](#)
4.  $H_{agua} = T_{agua}$
5.  $F_{Maguaa} = \{ [F_{Mvsc} * (H_{vsc} - H_{vs})] - [F_{Mvs} * H_{vs}] \} / (T_{agua})$
6.  $F_{Mvs} = F_{Mvsc} + F_{Maguaa}$
7.  $T_{vs} =$  función (  $P_{vs}$  ) Ver [Correlación PRESS en Tabla de Propiedades Termodinámicas del Agua](#)

**MÉTODO DE CÁLCULO:** Se realiza un balance de materiales y otro de energía, suponiendo en este último que la entalpía del agua es igual a su temperatura en grados Celsius ( $^{\circ}C$ ). Para el cálculo la información disponible es todo lo relativo a la corriente de entrada de vapor sobrecalentado, la temperatura del agua de atemperar y la presión del vapor saturado que sale; esta última se asume que es igual a la presión del vapor sobrecalentado

### Anexo 15. Defectos encontrados en la sección Tabla de propiedades termodinámicas del agua de la Ayuda del STA.

Temp - Temperatura del vapor en  $^{\circ}C$ .

$H_v$  - Entalpía del vapor saturado o sobrecalentado en  $P_{cu}/lb = kcal/kg$

$S_v$  - Entropía del vapor saturado o sobrecalentado en  $P_{cu}/(lb \cdot ^{\circ}C)$

$H_f$  - Entalpía del líquido saturado en  $P_{cu}/lb = kcal/kg$

$S_f$  - Entropía del líquido saturado en  $P_{cu}/(lb \cdot ^{\circ}C) = kcal/(kg \cdot K)$

### Anexo 16. Defectos encontrados en la sección Intercambiador de calor1 (vapor saturado-jugo) de la Ayuda del STA.

#### Condiciones y Suposiciones

- 1.- Es un proceso continuo y se asume que se alcanza el Estado Estacionario
- 2.- El Jugo es el Fluido Limitante para la transferencia del calor
- 3.- Se asume que el  $CP_{jugo}$  solo varía con el Brix; o sea la diferencia entre las temperaturas ( $T_{jusosale} / T_{jugoentra}$ ) no tiene efecto sobre el valor de  $CP_{jugo}$ . Para determinar el valor se emplea una correlación propuesta en [HUG78]
- 4.- Que el vapor de calentamiento está saturado (VCS) al igual que el condensado (CS)
- 5.- Como se condensa el Vapor el factor de corrección  $FT$  de la DMLT, es igual a 1; que equivale a asumir flujos a contracorriente pura.

matemáticamente por el modelo INTER1. Este Indicador se denomina "Efectividad de la Temperatura" ( $E_{fectTemp}$ ) del fluido caliente para calentar el frío y es parte del Método "Efectividad - CUT".

Es conveniente aclarar que el criterio o Indicador, de mayor interés y uso general, es el de Efectividad Termodinámica de la Transferencia de Calor (EfectTransf). Se define como la relación entre el Calor "Real" (QREAL) transferido y el Calor "Máximo Posible de Transferir" (QMAX)". O sea:

La Velocidad Lineal del Jugo (Vt) tiene una influencia significativa en el Coeficiente Individual de Calor. El intervalo deseado de valores está entre 1.5 - 2 m/s. Con valores mayores la tendencia es a un crecimiento más lento de las incrustaciones. Lo anterior tiene relativa poca importancia en Cuba, pues los calentadores se sacan de línea para desincrustarlos ("limpiarlos") cada 24 horas. Algunos autores proponen emplear una velocidad tal en los tubos que no se formen incrustaciones; aunque esto trae aparejado un aumento de los costos de la energía adicional necesaria para el bombeo del jugo y del desgaste de la superficie de los tubos. Pero como es de esperar aumenta significativamente el valor del Coeficiente Individual de Transferencia de Calor del jugo

#### Anexo 17. Defectos encontrados en la sección Vaso evaporador 1 y 2 de la Ayuda del STA.

##### Figura 1. Esquema de un triple efectos

En una fábrica donde los Tachos no estén automatizados el manejo de meladuras con valores de Brix iguales o mayores a 70 se hace muy difícil. Por eso en Cuba se recomienda que el Brix de la Meladura no exceda el valor de 67; aunque teniendo siempre presente que los bajos Brix en meladura aumentan significativamente el consumo de vapor en los tachos y por ende en el proceso

El aumento de la temperatura del jugo claro entrando al evaporador (Tjugoentraevap) induce un aumento en la Economía del Vaso y por tanto a un menor consumo de vapor. En determinadas condiciones, de operación y política de generación de la máxima cantidad posible de electricidad en un central con turbinas de contrapresión, lo anterior puede no ser conveniente,

Se recomienda asumir que las tasas mayores de 5,50-6,00 lbs/ (h- pie<sup>2</sup>) dan lugar a "arrastres de sólidos solubles". Este es un criterio heurístico que debe ser comprobado industrialmente pero que permite tener un criterio inicial que es importante para definir si los condensados puedan considerarse "puros" o al menos adecuados para emplearlos en los Generadores de Vapor.

la de la Cámara de Evaporación del Primer Efecto del Área de Evaporación en una combinación Simple Efecto + Múltiple Efecto donde el Primer Vaso del Múltiple se calienta con vapor de escape saturado de la misma presión que el Simple Efecto:

B.-  $ECO_{Vaso} > 1$  en los vasos intermedios del múltiple.

C.- Los valores de la Economía en Múltiples Efectos varían en dependencia del esquema de evaporación – calentamiento del área de evaporación, o sea

el CGTCE del Primer Vaso del Área [Simple Efecto (Celda de Vapor ) o Vaso 1 del Múltiple] puede alcanzar valores máximos entre 600 – 700 Btu/hr.pie<sup>2</sup>.F<sup>0</sup>, sobretodo si se le hacen extracciones para calentadores y tachos. Lo que no quiere decir que valores menores no sean convenientes en determinadas condiciones de operación inadecuados

- Si un Simple Efecto precede a un Múltiple, el CGTCE del Vaso 1 del Múltiple tiende a ser menor que el del Simple Efecto; lo que puede deberse a una política es de hacer el máximo de extracciones en los Simples Efectos. Política discutible, pues aunque facilita la operación del Área de Evaporación crea condiciones para que el uso de la energía en esta Área sea menos eficiente. También porque aumentan las pérdidas de sacarosa pues el jugo está más concentrado y la temperatura en el cuerpo es la mayor o una de las mayores de toda el Área.

#### **Ecuación Experimental de Jiménez, Traoré y Pérez de Alejo (CGTCE<sub>JTPA</sub>).**

El Coeficiente Global de Transferencia de Calor (CGTCE<sub>JTPA</sub>) según Jiménez, Traoré y Pérez de Alejo [JIM87] puede ser calculado con el Nomograma de la Figura 1

### **Anexo 18. Defectos encontrados en la sección Converge2 (CONV2) de la Ayuda del STA.**

**DIFERENCIAS ENTRE DFP Y DFI. No existen**

El proceso se repite hasta que se cumple el criterio.

$$|X^{(R+1)} - X^{(R)}| < p^{-k} (tol 1)$$

Se traza una línea entre los 2 puntos anteriores, la que intercepta el eje de las x en un punto  $X^{(2)}$ ; que como se puede ver en la Figura 1 este valor está más cerca del valor que se desea determinar por  $\bar{X}$ , que la suposición inicial  $X^{(0)}$ .

Se localiza, sobre la curva de la función, el punto  $\{f[X^{(2)}], X^{(2)}\}$  y se conecta con una línea recta al punto inicial  $\{f[X^{(0)}], X^{(0)}\}$  dando lugar al punto  $X^{(3)}$  y así sucesivamente se obtienen valores cada vez más cercanos a  $\bar{X}$ .

### **Anexo 19. Defectos encontrados en la sección Comparador de la Ayuda del STA.**

**PARÁMETROS ESPECIALES: No existen**

### **Anexo 20. Defectos encontrados en la sección Unión 1 de la Ayuda del STA.**

**PARAMETROS ESPECIALES: No tiene**

**SIMULA:** Unión de corrientes, distribución de corrientes(s) y cambios de temperatura y presión de una corriente en Colectores de Vapor y Tanques con líquidos.

**MÉTODOS DE CÁLCULO:** Balances de materiales totales y por componentes, sin reacción química. Pueden entrar de 1 a 4 corrientes de información, que se suman. Esta "Suma de los componentes de las Corrientes de Entrada" se puede convertir en 1, 2, 3 ó 4 corrientes de salida, con la misma composición que la suma de corrientes. Los respectivos flujos másicos de las corrientes de salida se establecen mediante Fracciones de Separación

## Anexo 21. Defectos encontrados en la sección Unión 2 de la Ayuda del STA.

### Unión 1



DFP Y DFI

**SIMULA:** El UNIÓN 2 permite unir hasta 4 corrientes de entrada y dividir las hasta en 4 corrientes de salida, de la misma forma que se realiza en el módulo UNIÓN 1. Pero además calcula las temperaturas de las corrientes de salida.

TEMPFMSUMA = [(FME(1,6) \* TEMPCE (1,4)) + (FME(1,6) \* TEMPCE (1,4))] / [FMSUMA]  
 TEMPCS (1,4) = TEMPFMSUMA  
 TEMPCS (2,4) = TEMPFMSUMA

Para el Caso de 2 Corrientes de Entrada y 2 de Salida  
**ECUACIONES:** 3 (adicionales a las del módulo UNIÓN 1)

**VARIABLES:** 5 (adicionales a las del módulo UNIÓN 1)

**GRADOS DE LIBERTAD (GL):** 5 - 3 = 2

**VARIABLES INDEPENDIENTES:** 2 (adicionales a las del módulo UNIÓN 1): TEMPCE (1,4) y TEMPCE (2,4)

**PARAMETROS ESPECIALES:** No tiene

#### ECUACIONES:

$$TEMPFMSUMA = \frac{\sum_{i=1}^{TCE} (FME(i,6) * TEMPCE(i,4))}{FMSUMA}$$

$$\sum_{i=1}^{TCS} [TEMPCS(i,3)] = TEMPFMSUMA$$

Para dos corrientes de entrada

TEMPFMSUMA = [(FME(1,6) \* TEMPCE (1,4)) + (FME(1,6) \* TEMPCE (1,4))] / [FMSUMA]  
 TEMPCS (1,4) = TEMPFMSUMA  
 TEMPCS (2,4) = TEMPFMSUMA

## Anexo 22. Defectos encontrados en la sección Cálculo de los módulos para datos de corriente de entrada y reciclo de la Ayuda del STA.

Los valores correspondientes a las corrientes de entrada se suministran como datos en los módulos que representan a las mismas. Hay varios tipos de módulos siendo estos::

Son módulos de cálculo con el objetivo de facilitar la entrada de los valores correspondientes a las variables independientes (datos)

Se puede localizar en la Paleta en los Módulos Lógicos. En la opción *Propiedades* se entran los datos correspondientes.

Cálculo del Valor Inicial (suposición) del Flujo Másico de Vapor de Calentamiento necesario para que mediante el módulo CONVERGE 2 se calcule el valor adecuado consumido en un simple ó *múltiple efectos*

## Anexo 23. Defectos encontrados en la sección Conectores de 1, 2 y 3 salidas C1S, C2S, C3S de la Ayuda del STA.

pseudocorriente

## Anexo 24. Defectos encontrados en la sección CONVERGE 1 de la Ayuda del STA.

DIFERENCIAS ENTRE DFP Y DFI. No existen

Figura 2 Propiedades del módulo CONVERGE1, en el caso del DFI de la Figura 1.

Parámetros Adicionales:

## Anexo 25. Defectos encontrados en la sección CONVERGE2 (CONV2) de la Ayuda del STA.

DIFERENCIAS ENTRE DFP Y DFI. No existen

**Parámetros Adicionales:** No existen

Considere la Figura 1 donde está graficado  $f(x)$  en función de  $x$ ; donde la curva continua representa la función  $f(x)$ .

Se localiza, sobre la curva de la función, el punto  $\{f[X^{(2)}], X^{(2)}\}$  y se conecta con una línea recta al punto inicial  $\{f[X^{(0)}], X^{(0)}\}$  dando lugar al punto  $X^{(3)}$  y así sucesivamente se obtienen valores cada vez más cercanos a  $\bar{X}$ .

#### Anexo 26. Defectos encontrados en la sección Estación de Tachos 2 de la Ayuda del STA.

Figura 1. Estación de Tachos y Centrífugas

*Subíndices:*

#### Anexo 27. Defectos encontrados en la sección Indicadores operacionales en producción de crudo de la Ayuda del STA.

5.1.- Consumo Especifico Real de Vapor en Turbogenerador o Turbina [lb/ (kW-h)]

(Depende tipo de Turbogenerador o Turbina y de la carga y otras condiciones con que se trabaje o calcule)

#### Anexo 28. Defectos encontrados en la sección Análisis económico de la Ayuda del STA.

temperatura requerida (95 a 110 0C)

DFP y DFI: Este Módulo **No** tiene Corrientes de Entrada y Salida. Su DFI no es del tipo usado en otros Módulos, excepto algunos de los de Reportes. La Información se obtiene en la Opción "Propiedades". Como estas Corrientes pueden ser **diferentes** de un caso a otro deben ser seleccionadas por el Usuario

Matches' Process Equipment Cost Estimates. <http://matche.com/EquipCost/>.



En una Fábrica de Azúcar Crudo con Turbinas de Contrapresión (Turbogenerador y Generadores de Vapor que trabajen a presiones menores de 2 MPa (500 psia) y

$$IB (\$/\text{año}) = BS (\text{lb/h}) \times \frac{1\text{kg}}{2.21\text{lb}} \times \frac{1\text{t}}{10^3\text{kg}} \times \text{HO} \left( \frac{\text{h}}{1\text{día}} \right) \times \text{PB} \left( \frac{\$/\text{ton}}{\text{ton}} \right) \times \text{DZ} \left( \frac{\text{día}}{1\text{zafra}} \right) \times 1 \frac{\text{zafra}}{\text{año}} \quad (1)$$

BS - Bagazo sobrante.  
PB - Precio del bagazo.  
HO - Horas de operación en el día.  
DZ - Días de zafra.

$$IB (\$/\text{año}) = BS (\text{lb/h}) \times \text{PB} \left( \frac{\$/\text{ton}}{\text{ton}} \right) \times \text{HO} \left( \frac{\text{h}}{1\text{día}} \right) \times \text{DZ} \left( \frac{\text{día}}{1\text{zafra}} \right) \times 0,000455 \frac{(\text{ton})(\text{zafra})}{(\text{lb})(\text{año})} \quad (2)$$

### Anexo 29. Defectos encontrados en la sección Sistemas de Vapor de la Ayuda del STA.

En las líneas de distribución de vapor la acumulación de agua puede provocar **golpe de ariete** con sus efectos destructivos: Cuando en una tubería se acumula una cantidad apreciable de líquido y la velocidad del vapor sobrepasa determinados valores, se produce un fenómeno

Sus valores de capacidad calorífica y calor latente son altos en comparación con otras sustancias, lo que tiene dos consecuencias deseables:

Vapor	Función	Razón
Sobrecalentado	Generar potencia	Mayor entalpía
	Calentamiento directo	Mayor temperatura
Saturado	Calentamiento por superficie	Mayor coeficiente individual de transferencia de calor Temperatura fácilmente controlable

- Entregar el vapor con la calidad requerida.
  - Generar el vapor con el mayor rendimiento térmico posible.
- Los posibles factores que influyen en el rendimiento térmico, así como incidir en los mismos para maximizar el rendimiento térmico, son
- Entregar el vapor con la calidad requerida.
- La **calidad** del vapor viene dada por los siguientes aspectos:

**Vapor seco**

Para garantizar que el vapor en las tuberías se mantenga seco:

Independientemente de los separadores de arrastre que debe tener el domo superior de la caldera, a la salida de la misma se debe instalar en la tubería un **separador de gotas** para prevenir posibles anomalías en la operación de la caldera (como una sobrecarga o un incorrecto tratamiento del agua de alimentación) que provoquen arrastres de agua. Su principio de funcionamiento es análogo al de los separadores ciclónicos del domo y se basa en provocar un cambio de dirección del vapor.

### Anexo 30. Defectos encontrados en la sección Diagrama del proceso de la Ayuda del STA.

PARA PRECISAR ASPECTOS DEL PROCESO DE AZÚCAR CRUDO SE RECOMIENDA LEER LA MONOGRAFÍA: "Notas sobre la Producción de Azúcar Crudo" de Héctor E. Pérez de Alejo Victoria; Osvaldo Goza León; Osney Pérez Ones y Rolando Santana Machado, ISPJAE, Noviembre 2005, ISBN: 959-16-0397-5, DISPONIBLE EN LA RED DE UNIVERSIDADES ADSCRIPTAS AL MES (REDUNIV).

### Anexo 31. Defectos encontrados en la sección Calores específicos de la Ayuda del STA.

El calor específico a presión constante, que presenta la cantidad de calor es necesario suministrarle a la unidad de masa para elevar su temperatura, puede determinarse mediante un **sin número** de expresiones.

Entre las ecuaciones posibles de usar **estas** de Janowsky y Archangelsky que aunque obtenida para productos de fábricas remolacheras, es posible usarlas en la industria cubana. Esta ecuación en su forma más general viene dada por la expresión:

### Anexo 32. Defectos encontrados en la sección Elevación del punto de ebullición de la Ayuda del STA.

Para Bx menores o iguales a 70

Estos resultados concuerdan con los parámetros industriales y como esta correlación **depende** solamente del brix es relativamente fácil incluir en los balances de energía.

### Anexo 33. Defectos encontrados en la sección Densidad de jugos, meladuras y mieles de la Ayuda del STA.

**mientras que** para la meladura y las mieles se tiene:

### Anexo 34. Defectos encontrados en la sección Reporte condensado de la Ayuda del STA.

m4(Converge)	
Controla	Brix
Converge	Si
Valor del parámetro modificado	24.99
Valor esperado del parámetro modificado	25
Diferencia obtenida	0.01
Diferencia permisible	0.1
Valor deseado del parámetro	25
Numero máximo de iteraciones	10

### Anexo 35. Defectos encontrados en la sección Enunciado ejemplo 2 de la Ayuda del STA.

Este es un ejemplo para entrenarse en el uso del STA 4.0. No es exactamente el sistema termo-energético de un ingenio (STE) por lo que no se incluye un diagrama de flujo de proceso (DFP). Esta formado por intercambiador de calor tipo Webre, un evaporador tipo Robert, un turbogenerador genera 2, 5 MW, un atemperador del vapor que sale del turbogenerador y el generador de vapor.

### Anexo 36. Defectos encontrados en la sección Reporte condensado ejemplo 2 de la Ayuda del STA.

m2(Control de convergencia)	
Valor deseado del parámetro	20.00
Numero máximo de iteraciones	10.00
Diferencia permisible	0.10
A controlar 1. Brix 2. Temp 3. Pres 4. Flujo	0.00
Controla 1. Brix 2. Temp 3. Pres 4. Flujo	1.00
Converge?(1-Si,0-No)	1.00
Valor del parámetro modificado	19.98
Valor esperado del parámetro modificado	20.00
Diferencia obtenida	0.02
Diferencia permisible	0.10

### Anexo 37. Defectos encontrados en la sección Enunciado central A de la Ayuda del STA.

El GV No. 1 tiene las características siguientes:

- capacidad nominal = 35 t/h
- temperatura del vapor directo = 330 °C
- rendimiento térmico bruto con bagazo = 79 %.
- temperatura del agua de alimentación = 130 °C.
- la combustión es en parrilla. Tiene sobrecalentador y economizador.
- no se realizan mediciones de la temperatura y composición de los gases de combustión.

### Anexo 38. Defectos encontrados en la sección Reporte condensado central A de la Ayuda del STA.

m0(0)
Temperatura de las corrientes de salida
Presión de las corrientes de salida
Cantidad de corrientes que salen
Fracción en la primera corriente de salida
Fracción en la segunda corriente de salida
Fracción en la tercera corriente de salida
Fracción en la cuarta corriente de salida

m19(CONVERGE 1 Control Convergencia c)
Converge
Diferencia obtenida en flujo total
Diferencia permisible en flujo
Diferencia obtenida en temperatura
Diferencia permisible en temperatura
Diferencia obtenida en presión
Diferencia permisible en presión
Numero máximo de iteraciones
m20(Unión Condensados Puros)
Temperatura de las corrientes de salida
Presión de las corrientes de salida
Cantidad de corrientes que salen
Fracción en la primera corriente de salida
Fracción en la segunda corriente de salida
Fracción en la tercera corriente de salida
Fracción en la cuarta corriente de salida

### Anexo 39. Defectos encontrados en la sección Reporte de indicadores central A de la Ayuda del STA.

Producción Eléctrica Especifica de la Fábrica (Pe)
Sobrante de Condensados Puros % caña molida
Déficit de condensados puros
Indicadores por equipos
Relación entre el CGTC Calculado y Hugot en Intercambiador de Calor
Relación entre el CGTC Calculado y Hugot en Intercambiador de Calor


Ultima modificación: 04/12/2008 13:47:05

### Anexo 40. Defectos encontrados en la sección Nuevo de la Ayuda del STA.

- Para crear un Nuevo Proyecto tiene tres opciones:
1. Seleccionar en el Menú Archivo: Nuevo.
  2. Usar la combinación de teclas Ctrl.+ n.
  3. O a través del botón de acceso directo

### Anexo 41. Defectos encontrados en la sección Cargar de la Ayuda del STA.

Para cargar o abrir un proyecto creado tiene tres opciones

- 1- Seleccionar en el menú archivo: Cargar
- 2- Usar la combinación de teclas: Ctrl + a
- 3- Ó a través del botón de acceso directo 


**Anexo 42. Defectos encontrados en la sección Salvar de la Ayuda del STA.**

Para guardar o salvar tiene tres opciones

1- Seleccione el en el menú archivo: Salvar

Le pedirá el nombre con el que usted reconocerá el diagrama y la extensión aquí es recomendable .sta ó .xml

2- Usar combinación de teclas Ctrl + s, y ponerle nombre y dirección

3- Ó a través del boton de acceso directo 

**Anexo 43. Defectos encontrados en la sección Copiar de la Ayuda del STA.**


Con esta operación se copia lo que se tenga seleccionado para pegarlo en otra parte.

Operaciones:

1. Necesita tener seleccionado lo que se desea copiar.

2. Ejecute la operación de Copiar en el Menú Edición la cual puede realizarse también:

2.1 Usando la combinación de teclas Ctrl.+ c.

2.2 Ó a través del botón de acceso directo: 

3. Busque el lugar donde lo desea poner.


4. Seleccione en el Menú Edición la operación: Pegar.

**Anexo 44. Defectos encontrados en la sección Pegar de la Ayuda del STA.**

Antes de seleccionar esta operación debe haber ejecutado las de Cortar o Copiar. De lo contrario donde esté el cursor en ese momento se pegará lo último que haya sido mandado a copiar.

1. Ejecute la operación de Pegar en el Menú Edición la cual puede realizarse también:

2. Usando la combinación de teclas Ctrl.+ v


3. Ó a través del botón de acceso directo: 

**Anexo 45. Defectos encontrados en la sección Limpiar diagrama de la Ayuda del STA.**

Esta operación permite tener la aplicación en blanco sin necesidad de crear un proyecto nuevo. Puede ejecutarse de tres formas diferentes:


1. Ejecutando la operación de Limpiar Diagrama en el Menú Edición.

2. Usando la combinación de teclas Ctrl.+ Del

3. Ó a través del botón de acceso directo: 


#### Anexo 46. Defectos encontrados en la sección Paletas de módulos de la Ayuda del STA.

Para darle visibilidad a la paleta de donde seleccionará los módulos con los que va a trabajar puede:

1. Seleccionar en el Menú Ver: Paleta de Módulos.
2. Presionar F4.
3. O a través del botón de acceso directo:  que al activarlo permite que aparezca la Paleta con solo pasar el cursor por encima de ella


#### Anexo 47. Defectos encontrados en la sección Vista previa de la Ayuda del STA.

Si desea que el diagrama se le muestre como una imagen puede:

1. Seleccionar en el Menú Ver: Como Imagen.
2. Presionando F9
3. O a través del botón de acceso directo: 


#### Anexo 48. Defectos encontrados en la sección Simular de la Ayuda del STA.

Para obtener resultados de todo el proceso de simulación debe ir a:



1. Menú Proyecto: Simular
2. Pulsando F10
3. O usando el botón: 

#### Anexo 49. Defectos encontrados en la sección Botones de la barra de herramientas de la Ayuda del STA.


##### **Puntero**

Al seleccionar un modulo de la paleta de módulos el puntero queda activado de modo que se adicionaran tantos módulos según los clics que se den, para evitar esto se activa el botón de acceso directo: 

##### **Girar 90°**

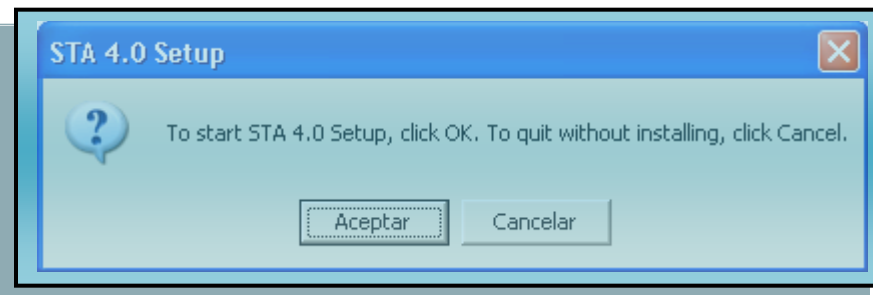
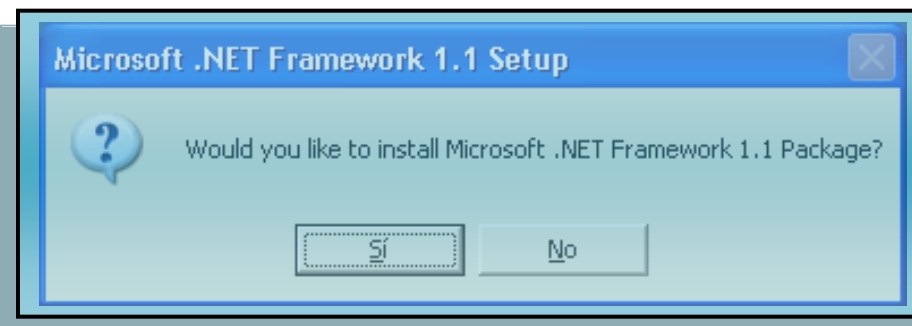
Con esta opción se puede girar el modulo seleccionado 90° a la izquierda o a la derecha en dependencia del botón que se elija, los botones para estas opciones se encuentran en la barra de herramientas:  

##### **Espejo horizontal**

Al seleccionar este botón  logramos que aparezca un modulo igual al seleccionado pero como una imagen respecto a la vertical.

**Anexo 50. Defectos encontrados en la sección Errores de la Ayuda del STA.**

1. Cuando entran un número fuera de los rangos del sistema operativo. En el evaporador también sucede lo mismo.
2. Cuando abandona el Textbox numérico y no es un número lo cambia por uno válido.
3. Cuando le doy doble click sobre el Textbox que enseña el error que tiene.
4. Cuando el módulo no se llama igual que el proyecto a la hora de insertar el módulo. Librería
5. Cuando no se encuentra la referencia al proyecto. (La misma que la 4)
6. Cuando intenta exportar a imagen un diagrama o sea sin módulos.
7. Error cuando se intento generar el reporte.
8. Cuando intenta redondear algún número que está fuera de los alcances numéricos del sistema.
9. Cuando no se puede copiar algún módulo.
10. Cuando ocurre un error de simulación.
11. Cuando intenta insertar más de un módulo de caña en un diagrama determinado.
12. Cuando le da un nombre a un módulo que ya existe en el diagrama.
13. Cuando algún campo en el asistente de reportes está vacío.
14. En el análisis económico cuando hay algún dato erróneo.

**Anexo 51. Defectos encontrados en la sección Instalando el simulador del Manual de Usuario del STA.****Anexo 51. Defectos encontrados en la sección Instalando Microsoft .NET Framework 1.1 del Manual de Usuario del STA.**

### GLOSARIO DE TÉRMINOS

**Simulador.** Aparato que reproduce el comportamiento de un sistema en determinadas condiciones, aplicado generalmente para el entrenamiento de quienes deben manejar dicho sistema.

**Informatización.** Aplicar los métodos de la informática en un negocio, un proyecto, etc.

**Internet.** Red informática mundial, descentralizada, formada por la conexión directa entre computadoras u ordenadores mediante un protocolo especial de comunicación.

**TIC.** Tecnología de la Informática y las comunicaciones.

**UCI.** Universidad de las Ciencias Informáticas.

**Proyecto productivo.** En la UCI está definido como un equipo de trabajo compuesto por estudiantes y profesores con el objetivo de desarrollar software para elevar la economía del país.

**Roles.** Papel que juega cada integrante de un proyecto productivo.

**STA.** Simulador Termo-Azúcar.

**Software.** Conjunto de programas, instrucciones y reglas informáticas para ejecutar ciertas tareas en una computadora.

**Prueba.** Proceso al que es sometido un producto para verificar su calidad.

**Caja Blanca.** Es un método de prueba para evaluar la calidad de un producto.

**Caja Negra.** Es un método de prueba para evaluar la calidad de un producto.