

**Universidad de las Ciencias Informáticas**

**Facultad # 6**



**Título: Procedimiento para pruebas de software  
Benchmark para Base de Datos.**

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

**Autores:** Elieny Rodríguez González  
Yoel Sánchez Mesa

**Tutores:** Ing. Delmys Pozo Zulueta  
Ing. Linet Lores Sánchez

**Co-Tutor:** Ing. Delvis Echeverría Pérez

**Ciudad de La Habana, junio 2010**

**“Año 52 de la Revolución”**



*“EN LA TIERRA HACEN FALTA HOMBRES QUE TRABAJEN  
MÁS Y CRITIQUEN MENOS, QUE CONSTRUYAN MÁS Y  
DESTRUYAN MENOS, QUE PROMETAN MENOS Y  
RESUELVAN MÁS, QUE ESPEREN RECIBIR MENOS Y DAR  
MÁS, QUE DIGAN MEJOR AHORA QUE MAÑANA”.*

A handwritten signature in black ink, which appears to be 'Che'.

## **DECLARACIÓN DE AUTORÍA**

Declaramos ser autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

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

**Elieny Rodriguez González**

\_\_\_\_\_  
Firma del Autor

**Yoel Sánchez Mesa**

\_\_\_\_\_  
Firma del Autor

**Ing. Delmys Pozo Zulueta**

\_\_\_\_\_  
Firma del Tutor

**Ing. Linet Lores Sánchez**

\_\_\_\_\_  
Firma del Tutor

**Ing. Delvis Echeverría Perez**

\_\_\_\_\_  
Firma del Tutor

## DATOS DE CONTACTO

### **Autores:**

Elieny Rodriguez González

Universidad de las Ciencias Informáticas, Habana, Cuba

Email: [erodriguezg@estudiantes.uci.cu](mailto:erodriguezg@estudiantes.uci.cu)

Yoel Sánchez Mesa

Universidad de las Ciencias Informáticas, Habana, Cuba

Email: [ysmesa@estudiantes.uci.cu](mailto:ysmesa@estudiantes.uci.cu)

### **Tutores:**

Ing. Delmys Pozo Zulueta

Universidad de las Ciencias Informáticas, Habana, Cuba

Email: [dpozo@uci.cu](mailto:dpozo@uci.cu)

Ing. Linet Lores Sánchez

Universidad de las Ciencias Informáticas, Habana, Cuba

Email: [lloress@uci.cu](mailto:lloress@uci.cu)

### **Co-Tutor:**

Ing. Delvis Echeverria Perez

Universidad de las Ciencias Informáticas, Habana, Cuba

Email: [decheverria@uci.cu](mailto:decheverria@uci.cu)

## **RESUMEN**

En la Universidad de las Ciencias Informáticas, las revisiones de software se basan principalmente en el proceso de pruebas de software, que tiene gran importancia para la verificación y control de la calidad de los productos.

En el presente trabajo se definió y aplicó un procedimiento para pruebas Benchmark para Bases de Datos, que aportará grandes beneficios a los proyectos productivos de la Facultad 6. Estas pruebas estarán enmarcadas en la comparación del rendimiento de las Bases de Datos que sean diseñadas para poder tomar decisiones a la hora de elegir cuál Gestor de Base de Datos es el más indicado a utilizar. Para ello se centró la investigación en los diferentes tipos de pruebas de software, haciendo énfasis en las pruebas de rendimiento, específicamente en las pruebas Benchmark, que son el objetivo del presente trabajo investigativo; llegando a realizar un detallado argumento de estos aspectos. Se realizó una investigación en busca de herramientas para la automatización de este tipo prueba, quedando seleccionada Benchmark Factory.

### **Palabras clave**

Bases de Datos, Benchmark, Calidad de software, Pruebas de software.

## Tabla de contenido

RESUMEN .....	I
INTRODUCCIÓN.....	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA .....	5
INTRODUCCIÓN.....	5
1.1 CALIDAD DEL SOFTWARE.....	5
1.1.1 <i>Control de la calidad</i> .....	6
1.1.2 <i>Garantía de la calidad</i> .....	6
1.2 PRUEBAS DE SOFTWARE.....	7
1.2.1 <i>Objetivos de las pruebas de software</i> .....	7
1.2.2 <i>Principios de las pruebas de software</i> .....	8
1.2.3 <i>Características de las pruebas de software</i> .....	9
1.3 TIPOS DE PRUEBAS.....	9
1.4 PRUEBAS DE RENDIMIENTO .....	11
1.5 PRUEBAS BENCHMARK.....	12
1.5.1 <i>Pruebas Benchmark de software</i> .....	13
1.5.2 <i>Pruebas Benchmark de hardware</i> .....	13
1.5.3 <i>¿Por qué debemos evaluar comparativamente las aplicaciones?</i> .....	13
1.5.4 <i>Estándares industriales para Benchmark</i> .....	14
1.5.5 <i>Errores en Benchmark</i> .....	14
1.6 PRUEBAS BENCHMARK PARA BASE DE DATOS .....	15
1.6.1 <i>Tipos de pruebas Benchmark</i> .....	15
1.7 HERRAMIENTAS PARA REALIZAR PRUEBAS DE BENCHMARK A BASES DE DATOS.....	20
1.7.1 <i>Tabla comparativa de las herramientas de pruebas Benchmark de software</i> .....	21
CONCLUSIONES .....	23
CAPÍTULO 2: PROPUESTA DEL PROCEDIMIENTO .....	24
INTRODUCCIÓN.....	24
2.1 PROCEDIMIENTO DE PRUEBAS BENCHMARK.....	24
2.2.1 <i>Objetivo</i> .....	24
2.2.2 <i>Alcance</i> .....	24
2.2 DEFINICIÓN DEL PROCEDIMIENTO.....	24
2.3 DESCRIPCIÓN GRÁFICA DEL PROCEDIMIENTO.....	27
2.4 DESCRIPCIÓN DE LAS ACTIVIDADES DEL PROCEDIMIENTO .....	28
2.4.1 <i>Enviar solicitud</i> .....	28
2.4.2 <i>Analizar solicitud</i> .....	28
2.4.3 <i>Planificación de las pruebas</i> .....	28
2.4.4 <i>Diseño de las pruebas</i> .....	30
2.4.5 <i>Ejecución de las pruebas</i> .....	31
2.4.6 <i>Documentación e informe de los resultados</i> .....	31
2.4.7 <i>Evaluación de las pruebas</i> .....	32
2.4.8 <i>Depuración de los errores</i> .....	33
CONCLUSIONES .....	34
CAPÍTULO 3: VALIDACIÓN DEL PROCEDIMIENTO.....	35

INTRODUCCIÓN.....	35
3.1 APLICACIÓN DEL PROCEDIMIENTO .....	35
3.2 FASES DEL PROCEDIMIENTO.....	36
3.2.1 <i>Enviar solicitud</i> .....	36
3.2.2 <i>Aceptar solicitud</i> .....	36
3.2.3 <i>Planifiresponsable del laboratorio de calidad cación de las pruebas Benchmark</i> .....	36
3.2.4 <i>Diseño de las pruebas</i> .....	36
3.2.5 <i>Ejecución de las pruebas</i> .....	36
3.2.6 <i>Documentación e Informe de los resultados</i> .....	44
3.2.7 <i>Evaluación de las Pruebas</i> .....	44
3.2.8 <i>Depurar errores</i> .....	45
3.3 MÉTODO DELPHI.....	45
3.3.1 <i>Selección del grupo de expertos</i> .....	46
3.3.2 <i>Cálculo del coeficiente por competencia</i> .....	47
3.3.3 <i>Elaboración del cuestionario para la validación propuesta</i> .....	49
3.3.4 <i>Desarrollo práctico y explotación de resultados</i> .....	49
CONCLUSIONES .....	53
CONCLUSIONES GENERALES.....	54
RECOMENDACIONES.....	54
REFERENCIAS BIBLIOGRÁFICAS .....	56
BIBLIOGRAFÍA .....	58
ANEXOS .....	60
ANEXO 1: PLANILLA DE SOLICITUD DE EVALUACIÓN .....	60
ANEXO 2: PLANILLA PLAN DE PRUEBA .....	61
ANEXO 3: PLANILLA CASO DE PRUEBA .....	66
ANEXO 4: ENCUESTA PARA DETERMINAR EL COEFICIENTE DE COMPETENCIA DE LOS EXPERTOS. ....	67
ANEXO 5: ENCUESTA PARA LOS EXPERTOS.....	68
GLOSARIO .....	70

## **Introducción**

El avance precipitado de la ciencia y las tecnologías de la información ha traído como consecuencia que las empresas informáticas existentes desafíen cada día el reto de brindar una respuesta rápida, eficaz y con calidad a los clientes que cada vez se vuelven más exigentes, no sólo en cuanto al precio, sino también en la confiabilidad y seguridad que deben poseer los productos de software.

La Informática, como campo de conocimiento alrededor del cual gira todo el proceso de producción de software, ha alcanzado en la actualidad un notable auge y consecuente importancia a nivel mundial. La calidad de los sistemas informáticos se ha convertido en uno de los principales objetivos estratégicos de las organizaciones tanto productoras como consumidoras de software, las cuales priorizando el aseguramiento de la calidad, asegurarán un mejor funcionamiento de estos sistemas informáticos.

Cuba se ha ido desarrollando en este sentido y actualmente la vinculación de todas sus ramas (económicas, políticas y sociales) con el mundo informático, son de suma importancia para el Estado Cubano. La producción de software no sólo trae beneficios desde el punto de vista del desarrollo de sistemas para el uso interno, sino también es una manera de introducirse en el mercado a escala mundial aprovechando su perspectiva económica. Este proceso requiere de una mejora constante del producto que garantice la calidad del mismo contando con varias entidades productoras de software dentro de ellas la Universidad de las Ciencias Informáticas (UCI).

La UCI, institución surgida al calor de la Batalla de Ideas, actualmente cuenta con disímiles proyectos productivos, fuentes de ingresos a la economía del país, Por esta razón la UCI, como centro que se dedica al desarrollo de productos informáticos se ve obligada a velar celosamente por la calidad de los productos que se desarrollan, para lograr que sean competitivos a escala internacional y que garanticen el buen funcionamiento de las entidades donde estos sean incorporados.

A partir del surgimiento de los grandes sistemas informáticos en el mundo, se fue incluyendo al mismo tiempo y por necesidad el proceso de pruebas, usado sistemáticamente junto al desarrollo de cada aplicación. Para verificar la calidad de los productos de software y contribuir con la estabilidad, escalabilidad, eficiencia y seguridad del producto, es necesario llevar a cabo un conjunto de pruebas de software que se definen como:



“Una actividad en la cual un sistema o componente es ejecutado bajo condiciones específicas, se observan o almacenan los resultados y se realiza una evaluación de algún aspecto del sistema o componente” [1].

Para verificar que los proyectos entreguen productos de alta calidad, como resultado del proceso de desarrollo de software, la universidad se trazó como estrategia en cada facultad llevar a cabo pruebas de calidad en cada uno de los proyectos. En la Facultad 6, internamente, se gestiona y controla la calidad de los productos que se liberan aplicando pruebas para validar diferentes criterios, como son: Funcionalidad, Confiabilidad y Rendimiento. Dentro de este último, existen distintos tipos de pruebas: la de Contención, que son las que se encargan de validar que el elemento que se esté probando funcione adecuadamente cuando muchos usuarios soliciten de él; la de Carga, la cual valora la aceptabilidad de los límites operacionales de un sistema bajo carga de trabajo variable y la de Benchmark que son las pruebas comparativas que miden el rendimiento, ya sea al software o al hardware. Teniéndose experiencia en la facultad en las pruebas de carga y stress mediante la herramienta JMeter. Basado este trabajo investigativo en las pruebas Benchmark que no se realizan, debido a que no se cuenta con materiales que guíen cómo debe ser realizada su ejecución, tampoco con personal capacitado, ni con datos históricos de pruebas anteriores utilizando este tipo de prueba, que es de gran interés para la facultad que se apliquen a las Bases de Datos de los proyectos.

Con la aplicación de las pruebas Benchmark, se logra un mejor rendimiento de las Bases de Datos a las cuales se les apliquen dichas pruebas. Las mismas garantizan resultados factibles a través de la comparación que establecen, obteniendo una mejor elección en cuanto a determinados aspectos que se pueden considerar fundamentales, entre los cuales podríamos mencionar: Las potencialidades y características que pueden ser aprovechadas al máximo en las Bases de Datos, para en este sentido contribuir al perfeccionamiento de las mismas. Otro de los aspectos lo constituye identificar las debilidades o elementos que no se tuvieron en cuenta a la hora de diseñar dichas Bases de Datos, elementos que obstaculizan el buen funcionamiento y desempeño de las mismas. Por último, apoyándose en los aspectos mencionados con anterioridad, las pruebas de Benchmark brindan la posibilidad de elegir el sistema de gestión que cumpla y posea las mejores características, teniendo en cuenta para ello los resultados obtenidos a través de la comparación establecida entre los diferentes sistemas gestores de Base de Datos.

Con los nuevos cambios realizados en la infraestructura productiva de la universidad, la facultad actualmente se encuentra inmersa en el desarrollo de varios proyectos productivos, los cuales utilizan una Base de Datos para almacenar la información generada en ellos. Por lo que se hace necesaria la aplicación de estas pruebas a los proyectos productivos asociados a este perfil y específicamente a DATEC, a fin de garantizar la calidad requerida y un buen diseño de las Bases de Datos que serán integradas a los productos informáticos de la universidad.

A raíz de la situación presentada en la universidad y en particular en la facultad que promueve el tema de esta investigación, cabe plantearse a modo de interrogante el siguiente **problema científico a resolver**: ¿Cómo aplicar pruebas de software Benchmark para evaluar el rendimiento de las Bases de Datos en la facultad 6?

El presente trabajo abarca un estudio de las pruebas de rendimiento y en particular las pruebas Benchmark a las Bases de Datos. De ahí que el **objeto de estudio de la investigación** sea las Pruebas de rendimiento Benchmark al software; siendo el **campo de acción** las pruebas Benchmark para Bases de Datos en la facultad 6.

Para dar solución al problema científico de la investigación se define como **objetivo general**: Desarrollar un procedimiento para aplicar las pruebas Benchmark a Bases de Datos en la Facultad 6.

Desglosado en los sucesivos **objetivos específicos**:

- Definir el marco teórico de la investigación.
- Definir un procedimiento para las pruebas Benchmark para Bases de Datos en la Facultad 6.
- Aplicar el procedimiento propuesto para la Facultad 6.
- Analizar los resultados obtenidos.

En pos de lograr el cumplimiento de los objetivos anteriormente señalados se proponen las siguientes **tareas**:

- Fundamentación de los principales conceptos, procedimientos y herramientas relacionados con el proceso de pruebas de software.

- Estudio de las pruebas de rendimiento de software.
- Estudio de las pruebas de rendimiento de software Benchmark.
- Estudio de las Bases de Datos.
- Investigación de las herramientas para la automatización de pruebas de rendimiento de software Benchmark.
- Selección de las herramientas para realizar las pruebas de software Benchmark.
- Definición de un procedimiento para pruebas de software Benchmark en la Facultad 6.
- Aplicación del procedimiento definido para las pruebas Benchmark en la Facultad 6.
- Análisis de los resultados obtenidos en la aplicación del procedimiento propuesto.
- Registro de las no conformidades detectadas en la aplicación del procedimiento para pruebas Benchmark.

El presente trabajo está estructurado en 3 capítulos.

- **Capítulo 1:** “Fundamentación teórica” comprende todo lo relacionado al objeto de estudio: conceptos importantes, relación directa entre las pruebas y la calidad de software, se hace un detallado argumento acerca de las pruebas Benchmark y un extracto de todo lo relacionado a las Bases de Datos.
- **Capítulo 2:** “Propuesta de procedimiento” es donde se obtienen los roles, artefactos y la(s) herramienta(s) seleccionadas para automatizar las pruebas, así como la definición del procedimiento.
- **Capítulo 3:** “Validación del procedimiento” es donde se aplica el procedimiento definido, aplicándolo a la Base de Datos de un proyecto, además de la validación realizada por los expertos mediante el método Delphi.

## **CAPÍTULO 1: Fundamentación teórica**

### **Introducción**

En el presente capítulo se exponen los temas fundamentales que sustentan la investigación. Se mencionan diferentes conceptos entre los que se destacan Calidad y Base de Datos. Se aborda sobre los tipos de pruebas de software, enfatizando en las pruebas Benchmark y en específico su aplicación en Base de Datos. Se muestra un estudio de las herramientas que existen para la ejecución de pruebas Benchmark para Base de Datos, así como comparaciones entre ellas.

### **1.1 Calidad del software**

Entre los problemas que afronta actualmente la esfera de la computación se encuentra la calidad del software. Desde la década del 70, este tema es motivo de preocupación para especialistas, ingenieros, investigadores y comercializadores de software, debido a que los clientes se vuelven más selectivos y comienzan a rechazar productos poco fiables o que realmente no den respuesta a sus necesidades.

Existen diversas definiciones de Calidad del Software enunciadas por varias compañías entre ellas la ISO y la IEEE que proponen normas y estándares para llevar a cabo una correcta práctica que garantice la buena ejecución de los procesos, dentro de las cuales pueden citarse:

“La calidad del software es el grado con el que un sistema componente o proceso cumple los requerimientos especificados y las necesidades o expectativas del cliente o usuario” [2].

“El conjunto de características de una entidad que le confieren su aptitud para satisfacer las necesidades expresadas y las implícitas” [3].

Por lo que a esta investigación respecta, la calidad del software se define como:

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.

Para obtener un software con calidad se requiere de la utilización de metodologías y procedimientos estándares para el desarrollo de los requerimientos, el análisis, el diseño, la implementación y, finalmente, las pruebas del software, que son el elemento fundamental para el logro de la calidad de cualquier sistema o parte integrante de éste. Las pruebas permiten nivelar la estrategia de trabajo en aras de lograr una mayor confiabilidad y facilidad de las soluciones.

## **1.1.1 Control de la calidad**

El control de la calidad son aquellas técnicas y actividades de carácter operativo, utilizadas para satisfacer los requisitos relativos a la calidad, centradas en dos objetivos fundamentales los cuales son:

- Mantener bajo control un proceso.
- Eliminar las causas de los defectos en las diferentes fases del ciclo de vida [3].

El control de la calidad está regido por una serie de técnicas dinámicas y estáticas donde éstas incluyen métodos para la evaluación del software. El control de la calidad encierra un bucle de realimentación del proceso que creó el producto. La combinación de medición y realimentación permite afinar el proceso cuando los productos de trabajo creados fallan al cumplir sus especificaciones. Este enfoque ve el control de calidad como parte del proceso de fabricación [4].

En general son las actividades para evaluar la calidad de los productos desarrollados. Las actividades de control de calidad pueden ser manuales, completamente automáticas o una combinación de herramientas automáticas e interacción humana. Un concepto clave del control de calidad es que se hayan definido todos los productos y las especificaciones mensurables en las que se puedan comparar los resultados de cada proceso.

## **1.1.2 Garantía de la calidad**

La Garantía de Calidad del Software, como algunos autores acostumbran llamarle, se define como una actividad que posibilita asegurar y proteger todo el proceso de ingeniería de software.

Es un conjunto de procedimientos, técnicas y herramientas, aplicados por profesionales, durante el ciclo de desarrollo de un producto, para asegurar que el producto satisface o excede los estándares o niveles de calidad preestablecidos [1].

Es la guía de los preceptos, de gestión y de las disciplinas de diseño para el espacio tecnológico y la aplicación de la ingeniería del software [2].

La garantía de calidad consiste en la auditoría y las funciones de información de la gestión. El objetivo de la garantía de calidad 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 la calidad del producto está cumpliendo sus objetivos.

## **1.2 Pruebas de software**

En todo proceso de desarrollo de aplicaciones es indispensable la presencia de un proceso de Pruebas de Software que coexista y se integre con este primero para garantizar así el buen funcionamiento y la calidad del producto final. Para lograr lo antes expuesto se debe partir del concepto de que las mismas desempeñan un papel fundamental en esta disciplina.

Toda prueba de software desempeña un papel fundamental en el desarrollo de cualquier tipo de aplicación, pero si se estudia la mejor forma de hacerlo, siguiendo los pasos de acuerdo con los especialistas en el tema, se incrementan las posibilidades de que ésta llegue a un feliz término y arroje resultados más cercanos a los esperados, permitiendo así, realizar a posteriori un mejor análisis de la situación. Para ilustrar mejor esta situación, se presentan a continuación algunos objetivos, características y principios con que deben contar las Pruebas de Software.

### **1.2.1 Objetivos de las pruebas de software**

Estos son algunos de los objetivos que debe perseguir todo diseño y ejecución de Pruebas de Software [5]:

- ✓ Probar si el software hace lo que debe.
- ✓ Probar si el software hace lo que no debe, es decir, si provoca efectos secundarios adversos.
- ✓ Descubrir un error que aún no ha sido descubierto.
- ✓ Encontrar el mayor número de errores con la menor cantidad de tiempo y esfuerzo posibles.

- ✓ Mostrar hasta qué punto las funciones del software operan de acuerdo con las especificaciones y requisitos del cliente.

## 1.2.2 Principios de las pruebas de software

A continuación se mencionan algunos de los principios básicos que guían las pruebas del software:

- ✓ A todas las pruebas se les debería poder hacer un seguimiento hasta los requisitos del cliente.
- ✓ Deberían planificarse mucho antes de que empiecen. La planificación de las pruebas pueden empezar tan pronto como esté completo el modelo de requisitos y se tenga consolidado el modelo de diseño [5].
- ✓ Deberían empezar por «lo pequeño» y progresar hacia «lo grande». Las primeras planeadas y ejecutadas se centran generalmente en módulos individuales del programa. A medida que se avanza en éstas, el enfoque de las pruebas cambia en un intento de encontrar nuevos errores relacionados con la integración de estos módulos y finalmente con la interacción del sistema completo [5].
- ✓ No son posibles las pruebas exhaustivas. El número de permutaciones de caminos para incluso un programa de tamaño moderado es demasiado grande. Por lo cual, es imposible ejecutar todas las combinaciones de caminos durante las mismas. Sin embargo, es posible elegir y ejecutar una serie de caminos lógicos importantes que permitan probar adecuadamente el software.
- ✓ Para ser más eficaces, deberían ser realizadas por un equipo independiente. El ingeniero de software que creó el sistema no es el más indicado para realizar las pruebas debido a que consciente o inconscientemente puede omitir Casos de prueba importantes que conlleven a descubrir nuevos errores. Por consiguiente, es recomendable organizar un grupo de trabajo independiente para las mismas que suministre una visión más objetiva del software [5].
- ✓ Se debe inspeccionar a fondo los resultados de cada prueba.
- ✓ Examinar un programa para ver que haga lo que se espera, es sólo la mitad de la batalla; la otra mitad es ver si hace lo que no se espera.

- ✓ No se debe hacer una planificación de los esfuerzos de prueba bajo la suposición de que no se encontrarán errores.
- ✓ La probabilidad de que existan más errores en una sección de programa es proporcional al número de estos que se hayan encontrado en esa sección.
- ✓ Son una tarea extremadamente creativa e intelectualmente desafiante [6].

### 1.2.3 Características de las pruebas de software

Algunas de las características que debe tener una «buena prueba» son:

- ✓ Tiene una alta probabilidad de encontrar un error. El ingeniero de software debe tener un alto nivel de entendimiento de la aplicación a construir para poder diseñar Casos de prueba que encuentren el mayor número de defectos.
- ✓ No debe ser redundante. Uno de los objetivos de las pruebas es encontrar el mayor número de errores con la menor cantidad de tiempo y esfuerzo posible, por lo cual no se deben diseñar Casos de prueba que tengan el mismo propósito que otros, sino que se debe tratar de diseñar el menor número de Casos de prueba que permitan probar adecuadamente el software y optimizar los recursos [2].
- ✓ Debería ser la mejor de la cosecha. La limitación en tiempo y recursos puede impedir que se ejecuten todos los Casos de prueba de un grupo de pruebas similares por lo cual en estas situaciones se debería seleccionar la prueba que tenga la mayor probabilidad de descubrir errores [6].
- ✓ No debería ser ni demasiado sencilla ni demasiado compleja.

### 1.3 Tipos de pruebas

Las pruebas en conjunto tienen como objetivo general verificar y validar un software, independientemente de las características y el entorno donde se desarrollen, además de los recursos y los factores vinculados al proceso de desarrollo. A continuación se exponen algunos ejemplos basados en atributos de calidad.



Siendo un atributo: Una entidad la cual puede ser verificada o medida en el producto de software. Los atributos no están definidos en el estándar, ya que varían entre diferentes productos de software [7].

Atributos de Calidad / Riesgos de Calidad	Tipos de prueba
<b>Funcionalidad</b>	<p><b>Función:</b> Pruebas fijando su atención en la validación de las funciones, métodos, servicios, Caso de uso.</p> <p><b>Seguridad:</b> Asegurar que los datos o el sistema solamente es accedido por los actores deseados.</p> <p><b>Volumen:</b> Enfocada en verificar las habilidades de los programas para manejar grandes cantidades de datos, tanto como entrada, salida o residente en la Base de Datos.</p>
<b>Usabilidad</b>	<p>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</p>
<b>Fiabilidad</b>	<p><b>Integridad:</b> Enfocada a la valoración exhaustiva de la robustez (resistencia a fallos).</p> <p><b>Estructura:</b> Enfocada a la valoración a la adherencia a su diseño y formación. Este tipo de prueba es hecha a las aplicaciones web asegurando que todos los enlaces están conectados, el contenido deseado es mostrado y no hay contenido huérfano.</p> <p><b>Stress:</b> Enfocada a evaluar cómo el sistema responde bajo condiciones anormales (extrema sobrecarga, insuficiente memoria, servicios y hardware no disponible, recursos compartidos no disponible).</p>

Atributos de Calidad / Riesgos de Calidad	Tipos de prueba
<p style="text-align: center;"><b>Rendimiento</b></p>	<p><b>Benchmark:</b> Es un tipo de prueba que compara el rendimiento de un elemento nuevo o desconocido a uno de carga de trabajo de referencia conocido.</p> <p><b>Contención:</b> 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).</p> <p><b>Carga:</b> 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.</p>
<p style="text-align: center;"><b>Soportabilidad</b></p>	<p><b>Configuración:</b> Enfocada a asegurar que funciona en diferentes configuraciones de hardware y software. Esta prueba es implementada también como prueba de rendimiento del sistema.</p> <p><b>Instalación:</b> Enfocada a asegurar la instalación en diferentes configuraciones de hardware y software bajo diferentes condiciones (insuficiente espacio en disco, etc.).</p>

**Tabla #1:** Tipos de pruebas de software [8]

#### 1.4 Pruebas de rendimiento

Las pruebas de rendimiento son las pruebas que se realizan, desde una perspectiva, para determinar lo rápido que realiza una tarea un sistema en condiciones particulares de trabajo [9]. Pueden demostrar que el sistema cumple los criterios de rendimiento; Pueden comparar dos sistemas para encontrar cuál de ellos funciona mejor o pueden medir que partes del sistema o de carga de trabajo provocan que el conjunto no funcione bien. Los ingenieros de software utilizan herramientas como pueden ser monitorizaciones que midan qué partes de un dispositivo o software contribuyen más al mal rendimiento o

para establecer niveles y umbrales del mismo que mantengan un tiempo de respuesta aceptable. Es fundamental para alcanzar un buen nivel de rendimiento de un nuevo sistema, que los esfuerzos en estas pruebas comiencen en el inicio del proyecto de desarrollo y se amplíen durante su construcción. Cuanto más se tarde en detectar un defecto de rendimiento, mayor es el coste de la solución.

Una vez el problema definido, se pueden proporcionar soluciones para erradicar estos problemas. Siendo capaces de probar su programa informático bajo condiciones múltiples, a fin de definir cómo éste reacciona bajo condiciones predeterminadas. Se pueden comparar las capacidades de su programa informático y compararlos a sus objetivos de resultados. Se puede definir los problemas y sus causas, y comprobar sus fiabilidades bajo condiciones de tensión. En función de estos resultados se puede proporcionar soluciones así como configuraciones que serán capaces de optimizar los resultados de su aplicación.

Existen varias pruebas que se utilizan para medir el rendimiento, entre ellas están:

**Benchmark:** Comparan el desempeño del sujeto de prueba con el de un sistema y carga de trabajo de referencia [10].

**Contención:** Valida que el elemento que se prueba maneja adecuadamente las demandas cuando muchos actores solicitan el mismo recurso [11].

**Carga:** Encargada de probar el funcionamiento del software bajo condiciones extremas. Estudia la especificación del software, las funciones que debe realizar, las entradas y las salidas analizando los valores límite [12].

El presente trabajo de diploma se enfoca en las pruebas comparativas (Benchmark) que logran medir el rendimiento del sistema en distintos escenarios, proporcionando la opción de conocer los puntos de afectación, factor clave que las demás pruebas no facilitan.

## 1.5 Pruebas Benchmark

Cuando se habla en el contexto de pruebas de rendimiento, la mayoría de las veces se utiliza la palabra Benchmark. El Benchmark es una prueba utilizada para medir el rendimiento de un sistema o componente del mismo, en diversas bibliografías frecuentemente, al hablar de este tipo de prueba, se refieren específicamente a la acción de ejecutar un Benchmark. Formalmente, un Benchmark puede ser definido

como “Un examen que se puede utilizar para comparar los componentes o sistemas entre sí o con una norma” [13], con el objetivo de estimar el rendimiento de un elemento concreto o la totalidad de la misma, y poder comparar los resultados con datos similares.

Cada prueba Benchmark realiza un trabajo diferente, algunas de éstas indican lo rápido que es un ordenador generando documentos, otras indican lo veloz que es en los gráficos y rellenos de pantalla, otras determinan la velocidad en operaciones matemáticas. Algunas hacen una mezcla de todas estas pruebas.

Estas pruebas son aplicadas tanto a elementos de software como de hardware, midiendo en ambos el rendimiento de las aplicaciones o de los elementos de un ordenador. A continuación se muestran ejemplos de estas pruebas.

## **1.5.1 Pruebas Benchmark de software**

Cuando se compara el rendimiento de un software contra otro o de parte del mismo se está realizando un Benchmark de software. Un ejemplo podría ser comparar distintas consultas a una Base de Datos para saber cuál es la más rápida o directamente partes de código

## **1.5.2 Pruebas Benchmark de hardware**

Para evaluar el requisito de hardware, puede ser que necesite identificar parámetros referentes a las computadoras, como por ejemplo, el micro que usa, o la marca, y los servidores con los que trabaja para alcanzar el mejor funcionamiento de una manera rentable.

En términos de ordenadores, un Benchmark podría ser realizado en cualquiera de sus componentes, ya sea UCP, RAM, Tarjeta gráfica, etc. También puede ser dirigido específicamente a una función dentro de un componente, por ejemplo, la unidad de coma flotante de la UCP, o incluso a otros programas.

## **1.5.3 ¿Por qué debemos evaluar comparativamente las aplicaciones?**

Cuando se compara uno con otro, contra una acción común (por ejemplo, diga la altura) que es una medida aplicable para ambas entidades, no podemos decir quién es la mejor. Llega a ser muy difícil si no se sabe cuál es la medida estándar que se puede utilizar para compararse contra el otro, igual es aplicable para los sistemas informáticos. Se necesita siempre saber dónde los competidores se colocan.

Para comparar diversos funcionamientos del software o del soporte físico, necesitamos tener las medidas estándares comunes.

## 1.5.4 Estándares industriales para Benchmark

Se hace difícil para los propietarios de aplicaciones probar el rendimiento de las mismas contra la variedad de máquinas y servidores disponibles en el mercado y mucho más elegir entre ellas. Hay organizaciones disponibles que hacen esta evaluación comparativa, entre ellas están: Standard Performance Evaluation Corporation (SPEC) y Transaction Procession Performance Council (TPC) que son los estándares industriales para Benchmark disponibles en el mercado. Estas organizaciones realizan las pruebas en diferentes servidores con hardware y configuraciones variadas para facilitar las cifras de rendimiento. Se tiene que tener alguna acción común para comparar con los competidores, estos estándares de la industria proporcionan la lista de medidas que es común en todas las plataformas del servidor [14].

**SPEC:** Es una organización sin fines de lucro formada para establecer, mantener y avalar un conjunto estandarizado de pruebas de rendimiento relevantes que se pueden aplicar a la más reciente generación de computadoras de alto rendimiento [14].

**TPC:** Es una organización sin fines de lucro fundada para definir las pruebas de rendimiento para el procesamiento de transacciones y Base de Datos, así como para difundir información sobre el rendimiento TPC, en forma objetiva y verificable para la industria [14].

## 1.5.5 Errores en Benchmark

Los Benchmark proporcionados por organizaciones como el TPC, SPEC se basan en el volumen de trabajo creado por una aplicación típica utilizada para las pruebas de rendimiento. Si los propietarios del software desarrollan aplicaciones con una carga de trabajo específico que es totalmente diferente del software típico de la carga de trabajo utilizado para la evaluación comparativa, entonces no es una comparación de elementos iguales. Es imposible probar el producto a través de diferentes configuraciones para escoger el mejor. Por tanto, no se puede hacer referencia a los Benchmark si la carga de trabajo de la aplicación sometida a las pruebas es completamente diferente de la usada por Benchmark.

## 1.6 Pruebas Benchmark para Base de Datos

Las Bases de Datos, como parte de las nuevas tecnologías, tienen un impacto decisivo con el creciente uso de los ordenadores; desempeñan un papel crucial en casi todas las áreas del conocimiento que se incorporan a los procesos de informatización y automatización, dentro de las cuales se pudieran citar: el comercio internacional, la ingeniería, la medicina, el derecho y la educación.

“Una Base de Datos es un conjunto de información que está almacenada en forma sistemática, de manera tal que los datos que la conforman puedan ser utilizados en forma fragmentada cuando sea necesario”[15].

Los datos almacenados pueden ser muy diversos: nombres, números telefónicos, direcciones y años. Todo depende de la finalidad para la que sea armada la Base de Datos. Actualmente, en todos los quehaceres cotidianos se utiliza una Base de Datos: cajeros automáticos, catálogos de bibliotecas o librerías, páginas amarillas, listado de medicamentos, e incluso los mismos buscadores de Internet. Todo cuenta con una Base de Datos a la cual recurrir para consultar su información y mantenerla actualizada.

Hoy día es crucial verificar y evaluar la calidad de las Bases de Datos de modo que se minimice el costo de su reparación. Mientras más temprano se detecte una falla, más barata es su corrección. Si a esto se añade que las Bases de Datos, hoy en día en un mundo globalizado, de alta incertidumbre y competitivo, ocupan un lugar determinante en cualquier área del quehacer humano, comercial, tecnológico y militar; se puede estar cerca de imaginar cuán importante resulta garantizar su calidad y lo catastrófico que podría resultar la pérdida de información o de la capacidad para gestionarla, que se convierte en una forma de marcar la diferencia y hacer ventaja competitiva. Con la ayuda de las pruebas Benchmark se puede supervisar el rendimiento factor clave para garantizar un buen funcionamiento del producto y así lograr resultados satisfactorios. A continuación se presentan varias pruebas Benchmark para Bases de Datos.

### 1.6.1 Tipos de pruebas Benchmark

#### **AS3AP**

Es un comparador de rendimiento escalable y portátil para Base de Datos relacionales ANSI (Instituto Nacional de Normalización Estadounidense) SQL. Este Benchmark proporciona un conjunto de pruebas

para medir el poder de procesamiento de las Bases de Datos; posee una escalabilidad y portabilidad incorporadas para probar una amplia gama de sistemas; minimiza el esfuerzo humano en la implementación y ejecución de pruebas de rendimiento; y proporciona una uniforme y directa interpretación métrica de los resultados comparativos.

## **Scalable Hardware**

Evalúa los sistemas de Base de Datos relacionales. Es un subconjunto del Benchmark AS3AP. Evalúa los siguientes elementos: UCP, disco, red y combinaciones de estos tres.

## **TPC**

El objetivo de TPC, fundado en 1988 y al que pertenecen más de cuarenta fabricantes de plataformas físicas y de lógica de Base de Datos, es definir las especificaciones de pruebas estándar para sistemas de proceso de transacciones en entornos comerciales.

## **TPC-A**

Evalúa todos los componentes del sistema necesarios para ejecutar las tareas asociadas con entornos de procesos de transacciones en línea, en donde es necesario efectuar intensas actualizaciones de Base de Datos sencillas. Con esta prueba se miden los catorce componentes más relevantes en aplicaciones comerciales:

- ✓ Arquitectura cliente-servidor.
- ✓ Tamaño de la longitud de palabra.
- ✓ Efectividad del compilador.
- ✓ Tamaño de memoria.
- ✓ Velocidad de conmutación de contexto.
- ✓ Eficiencia de red.

- ✓ Ante memoria de datos.
- ✓ Eficiencia del monitor OLTP.
- ✓ Eficiencia de la Base de Datos.
- ✓ Eficiencia del sistema operativo.
- ✓ Velocidad del procesador.
- ✓ Arquitectura de entrada/salida.
- ✓ Gestión del terminal.
- ✓ Ancho de banda de las interfaces de entrada-salida.

Los resultados son expresados en tpsA (transacciones por segundo) y \$/tpsA (relación expresada entre el precio en dólares al número de transacciones procesadas por segundo). El precio total incluye el coste del equipamiento físico y lógico del sistema probado y de su mantenimiento durante cinco años. Además, junto con los resultados de la prueba.

El TPC-A es una prueba adecuada para comparar el rendimiento de proceso de transacciones en línea para diferentes sistemas y se trata de la medida más ampliamente extendida en entornos OLTP.

## **TPC-B**

La principal diferencia entre el TPC-A y el TPC-B es que el primero considera el tiempo de respuesta del usuario, y el segundo el tiempo que el servidor de Base de Datos tarda en acabar la transacción.

Esta prueba mide los mismos catorce componentes que el TPC-A, excepto la gestión de terminales, la eficiencia de red, la arquitectura cliente-servidor y la eficiencia del monitor OLTP, pero añade la capacidad del uso de procesadores por lotes que lancen transacciones sin parar.



El TPC-B sobrecarga las Bases de Datos y se caracteriza por una significativa entrada/salida de disco; controla el sistema, el tiempo de ejecución de la aplicación y la integridad de la transacción. Los resultados de esta prueba se miden en tpsB y \$/tpsB con idéntico cálculo que para el TPC-A.

TPC-B es un Benchmark adecuado para evaluar el rendimiento de Bases de Datos sencillas de diferentes fabricantes en un entorno de proceso por lotes (batch) multitarea. Por este motivo, esta medida no es la adecuada para definir el comportamiento de un sistema OLTP, puesto que no se demuestra si el equipo físico y el equipo lógico del sistema pueden soportar el número de usuarios que una aplicación real requeriría.

## **TPC-C**

El inconveniente del TPC-A y del TPC-B es que las transacciones son siempre iguales, por lo que el fabricante de plataformas físicas y de equipos lógicos puede optimizar su sistema para obtener una mayor puntuación en la realización de una prueba TPC.

Con el TPC-C se miden quince componentes del sistema (los catorce que evalúa el TPC-A y la eficiencia de proceso de formularios) con una carga de trabajo real para evaluar el rendimiento de proceso de entradas de órdenes de transacciones en línea.

Los resultados del TPC-C son medidos en transacciones por minuto (tpmC) y por la relación entre el precio expresado en dólares al número de transacciones procesadas por minuto (\$/tpmC). El precio total incluye los mismos costes que los TPC-A y TPC-B.

El TPC-C es más realista que el TPC-A, por ello en la actualidad es el más aceptado entre los fabricantes y el más adecuado que tiene el comprador para comparar rendimientos de estaciones de trabajo. Es un Benchmark de procesamiento de transacciones en línea. Consiste en una mezcla de cinco transacciones simultáneas de diferentes tipos y complejidad, de ejecución ya sea en línea o consultas para ejecución diferida. La Base de Datos se compone de nueve tipos de tablas con una amplia gama de registros y variaciones de llenado.

TPC-C simula un entorno de cómputo completo donde una población de usuarios ejecuta transacciones contra una Base de Datos. El valor de referencia se centra en las principales actividades (transacciones)

de un entorno de entrada de pedidos. Estas transacciones incluyen la entrada y la entrega de pedidos, registro de los pagos, verificar el estado de sus pedidos y controlar el nivel de existencias en los almacenes. Mientras que el valor de referencia representa la actividad de un proveedor al por mayor, TPC-C no se limita a la actividad de cualquier segmento de negocio en particular, sino, más bien representa cualquier industria que debe gestionar, vender o distribuir un producto o servicio.

## **TPC-D**

En la actualidad se está desarrollando el TPC-D, cuyo propósito es medir el rendimiento de sistemas de soporte de decisiones. En estos sistemas, además de los accesos a las Bases de Datos para realizar consultas, se realizan operaciones con los datos contenidos en ellas, que implican consultar varias Bases de Datos interrelacionadas y realizar cálculos matemáticos y operaciones lógicas con los datos recabados. El TPC-D incluye 17 consultas sobre grandes Bases de Datos, su resultado se expresa en transacciones por hora y su cálculo otorga un valor similar a todas las consultas, independientemente de la duración de éstas, con lo que se hace muy difícil que el fabricante pueda ajustar sus sistemas para obtener una puntuación más alta.

## **TPC-H**

El TPC-H es un Benchmark de soporte para decisiones. Se compone de un conjunto de consultas ad-hoc orientadas a los negocios y modificaciones concurrentes de datos. Las consultas y los datos de poblar la Base de Datos han sido elegidos para tener una amplia relevancia de toda la industria. Este banco de pruebas muestra los sistemas de apoyo a las decisiones que examinan grandes volúmenes de datos, ejecutar consultas con un alto grado de complejidad, y dan respuestas a preguntas críticas del negocio. La métrica de rendimiento reportado por TPC-H es llamado el TPC-H compuesto de consultas por hora Métrica de los resultados (QphH @ Size), y refleja múltiples aspectos de la capacidad del sistema para procesar las consultas. Estos aspectos incluyen el tamaño de la Base de Datos seleccionada en contra de las consultas que se ejecutan, el poder de procesamiento de consultas cuando las consultas se envían a una sola corriente, y el rendimiento de las consultas cuando las consultas se envían a múltiples usuarios concurrentes. El precio de TPC-H/rendimiento métrico se expresa en \$/QphH@Size.

## **TPC-E**

TPC-E es un nuevo OLTP de volumen de trabajo desarrollado por el TPC. El punto de referencia TPC-E utiliza una Base de Datos para modelar una firma de corretaje con los clientes que generan las transacciones relacionadas con los oficios, las investigaciones en cuenta, y estudios de mercado. La firma de corretaje, a su vez interactúa con los mercados financieros para ejecutar órdenes por cuenta de los clientes y actualizaciones de información de la cuenta correspondiente. El valor de referencia es escalable, lo que significa que el número de clientes definidos para la firma de corretaje se puede variar para representar las cargas de trabajo de las empresas de diferente tamaño. La métrica TPC-E se da en las transacciones por segundo (TPS). Específicamente, se refiere al número de transacciones de Comercio-Resultados del servidor, que se puede mantener durante un período de tiempo. Aunque el modelo de negocio subyacente de TPC-E es una firma de corretaje, el esquema de Base de Datos, la población de datos, transacciones y normas de aplicación han sido diseñadas para ser ampliamente representativo de los modernos sistemas OLTP.

Para la realización de estas pruebas Benchmark es necesario utilizar herramientas automatizadas, que ayudan a la ejecución de las pruebas específicamente en la ejecución de TPC-B.

## 1.7 Herramientas para realizar pruebas de Benchmark a Bases de Datos.

Para comenzar a hablar del uso de herramientas para la realización de pruebas automatizadas, es necesario el conocimiento de dos conceptos básicos.

**Prueba automática:** “Es aquella realizada por un programa o herramienta que prueba el sistema sin necesidad de la interacción de una persona. La herramienta suministra una serie de valores de prueba, o acciones de prueba al sistema y verifica los resultados devueltos por éste con los resultados esperados” [16].

**Prueba manual:** “Es aquella prueba realizada por una o más personas que interactúan directamente con el sistema. Estas personas verifican si los resultados obtenidos son válidos o no” [16].

Las pruebas de software se pueden ejecutar de cualquiera de estas dos formas. Toda forma automática trae grandes beneficios, pues simplifica el trabajo y el esfuerzo humano necesario para realizar cualquier tipo de tarea, además de aportar velocidad, eficiencia y precisión. Las pruebas Benchmark por las características que tienen no se pueden realizar manuales.

**Benchmark Factory for Databases** es una herramienta de prueba de rendimiento y escalabilidad para Base de Datos Oracle, SQL Server, DB2 UDB, Sybase ASE, MySQL y otras que usen conectividad ODBC que: Genera carga para la Base de Datos repitiendo cargas de trabajo de producción, ejecutando scripts SQL definidos por el usuario o usando pruebas estándares industriales (AS3AP, TPC-C, TPC-B, TPC-D, Hardware Escalable). Simula usuarios concurrentes ejecutando transacciones en la Base de Datos para descubrir cuellos de botella de rendimiento antes de sacarla a producción. Evalúa cómo el código de la Base de Datos en SQL, PL/SQL, T-SQL escala bajo diferentes cargas de usuarios. Hace pruebas de estrés para determinar el punto de ruptura midiendo escalabilidad y rendimiento. Compara los resultados de rendimiento para permitir al usuario comparar la escalabilidad y el rendimiento de la Base de Datos entre plataformas de Bases de Datos, lo cual es necesario durante la migración de bases de datos (SQL Server 2000 a SQL Server 2005, Oracle 9i a Oracle 10g, etc.), evaluar plataforma del Sistema Operativo (Linux vs. Solaris vs. Windows) y cuando se despliega y pone a punto un Oracle RAC (Real Application Cluster).

**TPC Benchmark™ C (TPC-C)**, Benchmark propietario utilizado para el procesamiento en línea de transacciones. Es considerado como el estándar para OLTP en cuanto a Benchmark para gestores de bases de datos, principalmente porque intenta simular el procesamiento de transacciones en línea que se dan en el mundo real.

**TPC Benchmark™ B (TPC-B)**, Benchmark adecuado para evaluar el rendimiento de bases de datos sencillas de diferentes fabricantes en un entorno de procesos por lotes.

### 1.7.1 Tabla comparativa de las herramientas de pruebas Benchmark de software

A continuación se aprecia una tabla que resume una parte del estudio realizado, con el afán de mostrar claramente las bondades de algunas de estas herramientas y facilitar la comparación entre las mismas.

Los aspectos a tener en cuenta para realizar dicha comparación son:

**Tipos de pruebas:** en este punto se analizan los tipos de pruebas que se pueden automatizar con la herramienta. Es un parámetro importante pues permite conocer si la herramienta se adapta al enfoque de automatización que se espera obtener, o sea, para ver si es útil según el plan de automatización elaborado.

**Reportes:** en este punto se analiza la forma en que la herramienta presenta las salidas de las pruebas realizadas, sobre todo, hacia qué formatos exporta esta información.

**Última versión:** analizar este aspecto permite saber el seguimiento de la herramienta por parte de los productores. Esto da la medida de cuán periódicamente es mejorada dicha herramienta.

**Gratuita:** en este punto se analiza si la herramienta se adquiere de forma gratuita o no. Analizar este aspecto permite decidir entre herramientas de software libre o software propietario y dentro de las herramientas propietarias la que mejor se adapte al presupuesto del proyecto.

**Tipo de Base de Datos:** analizar este aspecto permite saber si tu herramienta es compatible con el tipo de Base de Datos que se vaya a probar.

Aspectos / Herramientas	TPC Benchmark™ C (TPC-C)	Benchmark Factory	TPC Benchmark™ B (TPC-B)
Gratuita	Propietario	Propietario	Propietario
Última versión	5.11	5.8	2.0
Tipos de Bases de Datos	OLTP	Oracle, SQL Server, DB2 UDB, Sybase ASE, MySQL	Oracle, MySQL, SQL Server
Reportes	SI	SI	SI
Tipos de Pruebas	P4	P1	P3
		P2	
		P3	
		P4	
		P5	
		P6	

**Tabla #2:** Herramientas para la realización de pruebas Benchmark

**Leyenda**

**P1:** Prueba AS3AP

**P4:** Prueba TPC-B

**P2:** Prueba Scalable Hardware

**P5:** Prueba TPC-C

**P3:** Prueba TPC-D

**P6:** Prueba TPC-H

En el momento de desarrollar un software, se utilizan una serie de herramientas y sería sumamente útil, acorde con las limitaciones tecnológicas o aquellas que generan los programas debido a su propio

funcionamiento interno, conocer las más productivas, rápidas y que estén más acordes a la configuración tecnológica. Del análisis realizado a las herramientas de prueba Benchmark mediante la tabla comparativa se llegó a la conclusión que la mejor y más eficiente por todo lo analizado anteriormente es Benchmark Factory superando a las demás, en cantidad de pruebas que realiza abarcando una gran cantidad de parámetros que logran comprobar el rendimiento que un software ha de tener. Como única desventaja de la herramienta se encuentra que es propietaria.

## **Conclusiones**

En este capítulo se abordó acerca de la calidad de software y su relación con las pruebas de software además definir los conceptos generales de pruebas de software, y los tipos que existen representados mediante una tabla. Se destaca un estudio de las pruebas Benchmark para Base de Datos quedando seleccionada para aplicar TPC –B. Se realizó un análisis sobre las principales herramientas investigadas para la automatización de estas pruebas y mediante una tabla comparativa. Todas las herramientas evaluadas tienen la desventaja de que son propietarias, entre ellas se decidió seleccionar Benchmark Factory debido a que es la que más tipos de pruebas Benchmark ejecuta y además a pesar de ser propietaria cuenta con un trial que puede ser utilizado para su evaluación. De la investigación realizada se puede concluir que no existe actualmente en la UCI una guía para llevar a cabo las pruebas de software Benchmark.

## CAPÍTULO 2: Propuesta del procedimiento

### Introducción

Las pruebas de Benchmark que se le realizan al software tienen un impacto importante en la calidad del producto final, por esta razón en el presente capítulo se ofrece una propuesta de solución para garantizar la realización de pruebas Benchmark para Base de Datos. Esta idea surge por la necesidad que existe en los proyectos productivos de la facultad 6 de medir el rendimiento de las Bases de Datos, para lograr un mejor resultado en tiempo y costo. Además de poder elegir cuál Gestor de Base de Datos será el más indicado a escoger por el equipo de desarrolladores. Este procedimiento estará enmarcado en una serie de actividades, tipos de pruebas y herramientas que persiguen lograr la calidad de las Bases de Datos.

### 2.1 Procedimiento de pruebas Benchmark

#### 2.2.1 Objetivo

Este procedimiento tiene como objetivo la medición y evaluación de las transacciones que puede realizar el servidor con determinada cantidad de usuarios conectados concurrentemente realizando operaciones en una Base de Datos. En dicho procedimiento se especifican los roles que intervendrán, las actividades que se realizarán y artefactos que se generarán.

#### 2.2.2 Alcance

Las transacciones se suelen aplicar a una amplia variedad de negocios y funciones del ordenador, a través de un terminal o un ordenador de sobremesa conectado a una Base de Datos. Este procedimiento abarca el software de Base de Datos de los proyectos de la facultad 6. El mismo va dirigido a aquellas personas que pertenecen a la facultad 6 y que se encargan de probar el software.

### 2.2 Definición del procedimiento

#### Recursos

Se denomina recursos a aquellos elementos que aportan algún tipo de beneficio a la sociedad, dentro de ellos se encuentran los recursos humanos y materiales. Se denomina recursos humanos al trabajo que

aporta el conjunto de empleados o colaboradores de una organización y los recursos materiales que no son más que los medios físicos y concretos que ayudan a conseguir algún objetivo.

- **Recursos materiales:** Se necesita 1 máquina por cada probador que cumpla con los requerimientos mínimos de la herramienta que se utilice.
- **Recursos humanos:** Se encuentran los roles y las responsabilidades de las personas incluidas en el proceso de prueba.

Rol	Responsabilidades	Cantidad
<b>Jefe de proyecto</b>	<ul style="list-style-type: none"> <li>✓ Es la persona encargada de planear, dirigir y destinar los recursos necesarios para el proyecto, además de ser el responsable y asegurar que el equipo de desarrollo depure los errores.</li> </ul>	1
<b>Probador</b>	<ul style="list-style-type: none"> <li>✓ Se encarga de la ejecución de los Casos de pruebas que le son asignados por el responsable de pruebas.</li> <li>✓ Tiene como tarea registrar y documentar los defectos que detecte durante la ejecución de las mismas.</li> </ul>	2
<b>Especialista</b>	<ul style="list-style-type: none"> <li>✓ Elaborar lista de chequeo para realizar las pruebas.</li> <li>✓ Elaborar el plan de pruebas.</li> <li>✓ Supervisar el trabajo de pruebas, recogiendo las no conformidades para elaborar el informe de no conformidades.</li> </ul>	1



Rol	Responsabilidades	Cantidad
	<ul style="list-style-type: none"> <li>✓ Controlar, monitorear y ejecutar el plan de pruebas.</li> <li>✓ Evaluación del proceso de pruebas y los resultados de las mismas.</li> </ul>	
<b>Responsable del laboratorio de Calidad</b>	<ul style="list-style-type: none"> <li>✓ Es la persona que está al frente del proyecto de calidad y el encargado de asignar los especialistas para las pruebas.</li> </ul>	1

**Tabla #3:** Roles y Responsabilidades

**Artefacto**

Un Artefacto es un tipo de elemento o conjunto de información, consumido o producido por algún(os) Proceso de Negocio, identificable en la organización, y con características definidas.

Un Artefacto puede utilizarse en alguna(s) Entrada o Salida de algún(os) Proceso de Negocio.

- **Artefactos de Entrada:** La Base de Datos a quien se le aplicará las pruebas.
- **Artefactos de Salida:** Los resultados arrojados por las pruebas en conjunto de la planilla de No Conformidades, los Casos de Prueba y la plantilla de Evaluación de las Pruebas.

2.3 Descripción gráfica del procedimiento

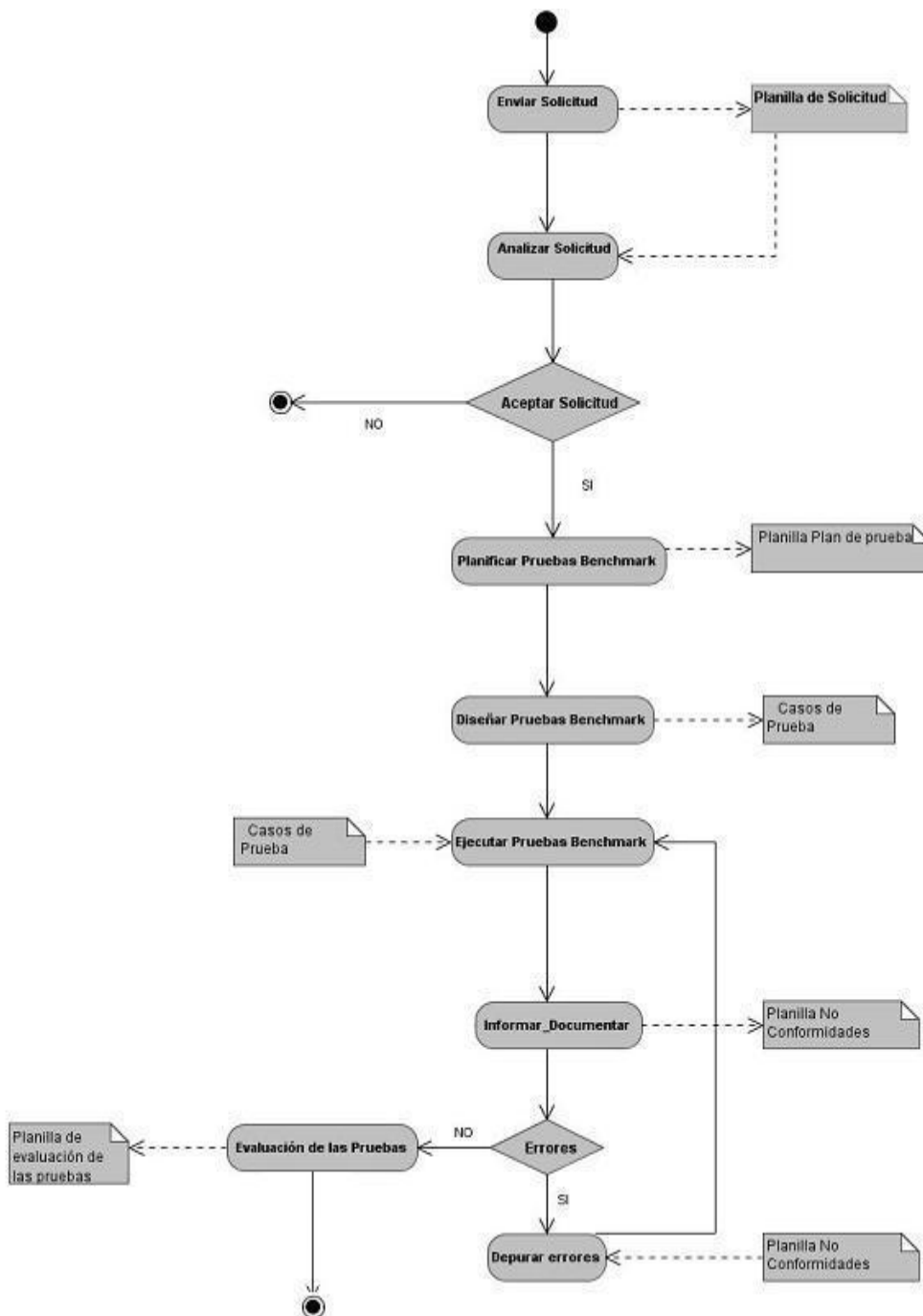


Figura 1: Flujo gráfico del procedimiento.

### 2.4 Descripción de las actividades del procedimiento

#### 2.4.1 Enviar solicitud

Este es el primer paso que se ejecuta en el procedimiento y lo inicia el jefe del proyecto cuando hace la solicitud para la evaluación de la aplicación al responsable del laboratorio de calidad mediante la planilla de solicitud (Anexo 1) propuesta para ello, utilizando las vías digitales o tangibles. Las vías digitales serían correo electrónico o la subida en línea de la planilla a un sistema web y la vía tangible sería, haciendo la solicitud en la oficina determinada.

#### 2.4.2 Analizar solicitud

En este punto el responsable del laboratorio de calidad es el que se encarga de analizar la solicitud que le ha sido enviada y tiene la responsabilidad de decidir su aprobación.

Para este paso se consultó a la ingeniera que dirigió el proyecto de calidad de la facultad 6 por unos años y actualmente pertenece al Departamento de Pruebas del Centro de Calidad, para que diera sus criterios acerca de cuáles son los elementos que se tienen en cuenta a la hora de aceptar una solicitud, a lo cual respondió:

Lo primero que se tiene en cuenta es la cantidad de proyectos que se tengan para liberar, es decir, la demanda de proyectos que existan en ese momento. Luego observar si tienen la versión estable de la aplicación, que no tengan dificultades con los artefactos que se le van a entregar al cliente que son los que se van a revisar, además de analizar que esté preparado el entorno de prueba del proyecto percibiendo si se cuenta con los probadores necesarios para asumir la prueba.

#### 2.4.3 Planificación de las pruebas

En este paso se planificarán las pruebas que se le realizarán a la aplicación, así como los roles y los recursos que intervendrán en la ejecución de las pruebas. Tiene como resultado final el artefacto Plan de prueba, pues es este quien define los objetivos de las pruebas en el ámbito de la iteración (o el proyecto), los elementos de destino, el enfoque que se adopta, los recursos necesarios y los entregables que se deben generar. Se realiza con el objetivo de esquematizar y comunicar la intención del esfuerzo de prueba de una planificación determinada. Y para obtener la aceptación y la aprobación de los interesados en el esfuerzo de prueba.

El Plan de prueba forma la infraestructura dentro de la cual el equipo que realiza las pruebas trabajará durante la planificación determinada. Dirige, orienta y restringe el esfuerzo de prueba, centrándose el trabajo en los entregables útiles y necesarios. Como tal, el Plan de prueba debe evitar detalles que no se entenderán, o que los interesados considerarían irrelevantes en el esfuerzo de prueba. Para el procedimiento se adapta el Plan de prueba del expediente de proyecto 2.0 el cual se basa en la plantilla Plan de prueba de RUP.

La planificación quedará plasmada en la **plantilla Plan de pruebas** que contiene los siguientes epígrafes más significativos:

- ✓ **Introducción:** Breve descripción del módulo que se va a probar. Incluye el alcance, las definiciones, acrónimos y abreviaturas, además de las referencias.
- ✓ **Organización del Equipo de Pruebas:** Descripción del equipo de probadores, por quienes está compuesto, responsabilidad de cada miembro.
- ✓ **Especificaciones del Software y Hardware:** Describe las características que debe tener el hardware y el software utilizado para ejecutar la aplicación.
- ✓ **Descripción del Plan de Pruebas:** Describe cómo serán planificadas las pruebas así como, las funcionalidades a probar.
- ✓ **Casos de Prueba:** Describe dónde estarán ubicadas las plantillas que contienen los diseños de Casos de prueba .
- ✓ **Estrategia de Prueba:** Describe cómo los objetivos de la prueba serán alcanzados para cada uno de los tipos de pruebas que hacen parte del plan. Además del objetivo, la técnica, el entorno de prueba, el proceso, los Casos de prueba y las herramientas.
- ✓ **Recursos Requeridos:** Identifica los roles y las responsabilidades que serán requeridas para la ejecución del Plan de pruebas.

- ✓ **Calendario y Plazos:** Documenta el plazo en el cual la aplicación a probar estará disponible para pruebas y el tiempo estimado para ejecutar los Casos de prueba.
- ✓ **Definición de los Entregables:** Lista de entregables asociado a las pruebas.
- ✓ **Seguimiento y Reporte de Defectos:** Documenta el instrumento y el proceso usado para registrar y rastrear los defectos.
- ✓ **Aprobación del Plan:** Obtiene las firmas de aprobación en todas las páginas del mismo.
- ✓ **Documentación de los Resultados:** Cuando el esfuerzo de prueba esté terminado, se deben documentar los resultados y mediciones.

#### 2.4.4 Diseño de las pruebas

El diseño de prueba incorpora todas las decisiones referentes a los elementos estructurales de diseño y sus colaboraciones, necesarios para habilitar las pruebas que ya se han identificado. En esta actividad se diseñarán los Casos de pruebas correspondientes a las pruebas Benchmark. A continuación se muestra la estructura del Caso de prueba que se propone para dichas pruebas. Este artefacto de prueba fue elaborado por los autores del presente trabajo investigativo modificando la tabla Secciones a probar de la planilla Caso de prueba del expediente de proyecto 2.0 que se utiliza para evaluar las pruebas.

Este Caso de prueba contiene:

- ✓ **Descripción General:** Describe las características de la Base de Datos a probar.
- ✓ **Secciones a probar:** Contiene el nombre de las pruebas, las características y los resultados.

ID del escenario	Escenarios de la sección	Cantidad de Usuarios	Descripción	Resultado esperado	Resultado de la prueba
<Identificador.>	<EC 1.1: Nombre del Escenario, el cual coincide con el tipo de benchmark.>		<Descripción del tipo de prueba seleccionada>	<Descripción del resultado esperado para cada intervalo de usuarios>	<Descripción del resultado real>
	<EC 1.1: Nombre del Escenario, el cual coincide con el tipo de benchmark.>		<Descripción del tipo de prueba seleccionada>	<Descripción del resultado esperado>	<Descripción del resultado real>
	<EC 1.1: Nombre del Escenario, el cual coincide con el tipo de Benchmark.>		<Descripción del tipo de prueba seleccionada>	<Descripción del resultado esperado>	<Descripción del resultado real>

**Tabla #4:** Caso de prueba

#### 2.4.5 Ejecución de las pruebas

En esta actividad se ejecuta la herramienta manual diseñada en el paso anterior, también apoyados en la herramienta automatizada que el probador decida escoger para evaluar los indicadores de Benchmark. La ejecución de las pruebas debe estar estrictamente coordinada con lo establecido en el Plan de prueba, en el cronograma de actividades, para evitar cualquier tipo de atraso. Durante este período se van a evaluar la ejecución del proceso de prueba, registrando los defectos o las no conformidades (NC) encontradas durante la ejecución. Se registrarán las NC en la plantilla que pertenece al expediente de proyecto 2.0.

#### 2.4.6 Documentación e informe de los resultados

En esta actividad se van a examinar todos los incidentes y anomalías de la prueba. Es aquí donde se va a evaluar la realización de las pruebas, las condiciones de ejecución, si se presentó algún imprevisto, si el entorno de prueba presentó problemas, cómo se desempeñaron los probadores, etc. Se van a recopilar

todas las NC encontradas y se van a entregar al equipo de desarrollo. El proceso de pruebas es evaluado de acuerdo a la asistencia de los probadores, calidad del trabajo y cumplimiento del cronograma.

La **Plantilla NC** contiene los siguientes epígrafes más significativos:

- ✓ **Aspectos Generales:** Aspectos a tener en cuenta a la hora de analizar el resultado de las pruebas, incidencias en el momento de su desarrollo y otros aspectos relevantes.
- ✓ **Elementos Probados:** Descripción general o lista de los elementos probados, y otros aspectos importantes a tener en cuenta a la hora de analizar las NC detectadas.
- ✓ **Elementos no Probados y causas:** Descripción general o lista de los elementos no probados, la causa de que no se hayan podido realizar las pruebas y cualquier otro elemento importante que aporte la información necesaria para que sean analizadas estas causas y resueltas para la siguiente iteración.

Elemento	No	No conformidad	Aspecto correspondiente	Etapas de detección	Importancia	Recomendación
<Nombre del Elemento>	<1>	<Descripción de la No Conformidad>	<Descripción del Aspecto correspondiente>	<Etapas de detección del error>	<X>	<X>

**Tabla #5:** Planilla NC

### 2.4.7 Evaluación de las pruebas

Este paso se realiza luego de haber culminado el proceso de prueba y consiste en evaluar las pruebas mediante un conjunto de indicadores por parte de los especialistas. Para esta fase los autores del presente trabajo investigativo elaboraron la planilla evaluación de las pruebas que se muestra a continuación.

Indicadores definidos	
Indicadores a Evaluar	Evaluación
1. ¿Se planificaron correctamente las pruebas?	
2. ¿Se identificaron los roles y recursos que intervendrán en la ejecución de las pruebas?	
3. ¿Se diseñaron correctamente los Casos de prueba correspondiente a las pruebas Benchmark?	
4. ¿Al ejecutar las pruebas diseñadas se obtuvieron los resultados esperados?	
5. ¿La ejecución de las pruebas está estrictamente establecido con el Plan de prueba?	
6. ¿Se informaron y documentaron las NC?	
<b>Evaluación final</b>	

**Tabla #6:** Indicadores para Evaluar las pruebas

**Evaluación:** es la forma de evaluar el indicador en cuestión. El mismo se evalúa mediante un rango de 1 a 5

1 a 2.....Mal

4.....Bien

3.....Regular

5.....Muy Bien

La evaluación final es el promedio de todas las evaluaciones dadas en cada indicador.

#### 2.4.8 Depuración de los errores

Este paso corresponde al equipo de desarrollo, el cual al momento en que se le entregue el informe de errores y defectos, va a encaminar sus esfuerzos a erradicarlos. Luego informa al equipo de prueba, el cual va a proceder nuevamente a realizar una nueva iteración. Además, actualiza la columna de Respuesta del equipo de desarrollo de la plantilla de No Conformidades poniendo el estado de la no conformidad: resuelto, no resuelto, procede.



### **Conclusiones**

En este capítulo se definió el procedimiento para pruebas de software Benchmark a Bases de Datos, mostrando la composición del mismo para llevar a cabo las pruebas, revelando las 8 fases por las cuales está constituido, además de reflejar los 4 artefactos generados por las actividades del procedimiento mostrándose a través de una representación gráfica. Para la eficiente ejecución de estas pruebas se diseñó un Caso de prueba que puede ser utilizado como apoyo para realizar las pruebas. Además, se elaboró una plantilla de evaluación de las pruebas que sirve como guía al especialista para evaluar el desarrollo del proceso de prueba.

## CAPÍTULO 3: VALIDACIÓN DEL PROCEDIMIENTO

### Introducción

En el presente capítulo se aplicará el procedimiento de pruebas Benchmark para Base de Datos a un proyecto desarrollado en la universidad, luego de aplicado se realiza un análisis detallado de los resultados. Se detallan todos los pasos del proceso de aplicación del procedimiento propuesto permitiendo un mejor entendimiento de las actividades y artefactos que forman al procedimiento, demostrando su utilidad para ser aplicado en los proyectos productivos de la facultad. Además, se evaluará el procedimiento a través del método Delphi donde los expertos seleccionados han estado vinculados al estudio realizado durante el transcurso de la investigación sobre las pruebas Benchmark. A los expertos se les realizará una encuesta para comprobar la efectividad y eficiencia del procedimiento. Para la validación del procedimiento por el método Delphi se empleó la entrevista como técnica para obtener la información referente al tema, el criterio de los expertos para la validación y aceptación del procedimiento mediante el uso de técnicas propuestas por el método Delphi siendo este uno de los métodos subjetivos de pronosticación más confiable.

### 3.1 Aplicación del procedimiento

El procedimiento propuesto en esta investigación se aplicó al proyecto Ambiente Integrado para la Modelación y Ejecución de Procesos (AIMEP), desarrollado en la facultad 6 de la universidad de las Ciencias Informáticas. Este proyecto tiene como propósito lograr un producto software para la gestión eficiente en cuanto al control, acceso y manejo de la información en los centros biotecnológicos del país, de manera que aumente la producción y comercialización de sus productos biológicos. La aplicación desarrollada necesita ser validada para luego ser comercializada, uno de los requisitos más importantes establece que los tiempos de respuestas deben ser los más cortos posibles, aproximadamente de 3 a 5 segundos, al igual que la velocidad de procesamiento de la información.

## 3.2 Fases del procedimiento

### 3.2.1 Enviar solicitud

En este paso es donde el jefe del proyecto realiza una solicitud para que le evalúen una aplicación de su proyecto, en este caso la Base de Datos. Esta solicitud es enviada por el jefe del proyecto al responsable del laboratorio de Calidad. Ver (anexo 1).

### 3.2.2 Aceptar solicitud

Luego de haberse enviado la solicitud prosigue la aceptación de la misma donde el responsable del laboratorio de Calidad la evalúa basándose en los criterios de aceptación, expresados en el presente trabajo de diploma en el Capítulo 2, para aceptar la solicitud.

### 3.2.3 Planificación de las pruebas Benchmark

En este paso el Especialista de Prueba planifica la prueba que se realizará definiendo el personal involucrado, los recursos y actividades necesarias, arrojando como resultado final el Plan de Prueba que contiene todos los datos necesarios para llevar a cabo el proceso de prueba (Ver anexo 2).

### 3.2.4 Diseño de las pruebas

En esta fase se diseñó el Caso de Prueba correspondiente a la prueba TPC-B, reflejado en la planilla Caso de Prueba elaborada por los autores del presente trabajo de diploma (Ver Anexo 3).

### 3.2.5 Ejecución de las pruebas

Se ejecutó la prueba diseñada mediante herramientas automatizadas y manuales. Para la ejecución con la herramienta automatizada se utilizó Benchmark Factory y para las manuales el Caso de prueba.

La aplicación del procedimiento se enfocó en la prueba TPC-B que puede ser considerada como una prueba de tensión de Bases de Datos. Proporciona una manera conveniente de poner carga a la Base de Datos de forma repetible, consistente y estresante en un sistema, lográndose así una visión centrada en la capacidad de procesamiento de transacciones, que es extremadamente útil para la planificación de capacidad del sistema.

El punto de referencia TPC-B se compone de una sola transacción que actualiza tres tablas y se inserta un registro en una tabla de historial (Ver figura #2).

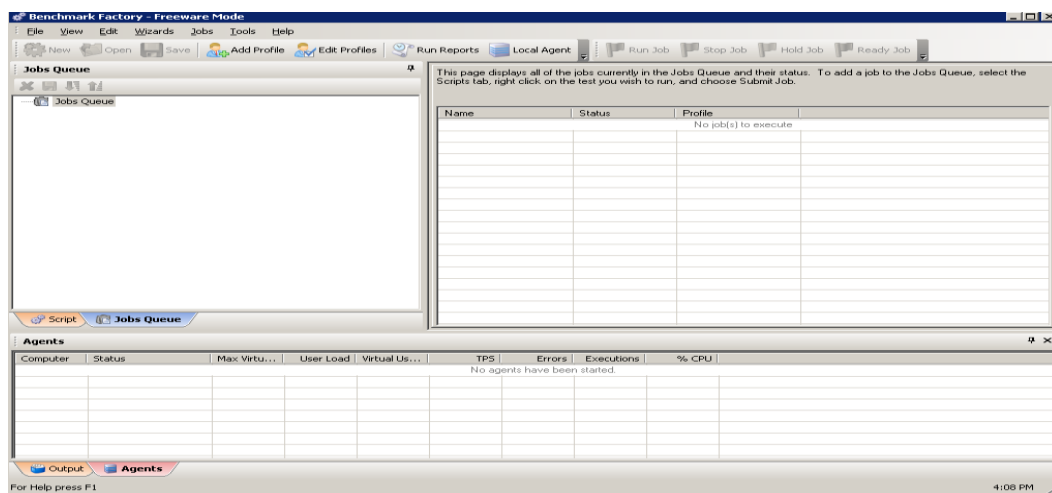
branches			history		
<b>+bid</b>	<b>int4</b>	<b>Nullable = false</b>	tid	int4	Nullable = true
bbalance	int4	Nullable = true	bid	int4	Nullable = true
filler	char(88)	Nullable = true	aid	int4	Nullable = true
			delta	int4	Nullable = true
			mtime	timestamp	Nullable = true
			filler	char(22)	Nullable = true

tellers			accounts		
<b>+tid</b>	<b>int4</b>	<b>Nullable = false</b>	<b>+aid</b>	<b>int4</b>	<b>Nullable = false</b>
bid	int4	Nullable = true	bid	int4	Nullable = true
tbalance	int4	Nullable = true	abalance	int4	Nullable = true
filler	char(84)	Nullable = true	filler	char(84)	Nullable = true

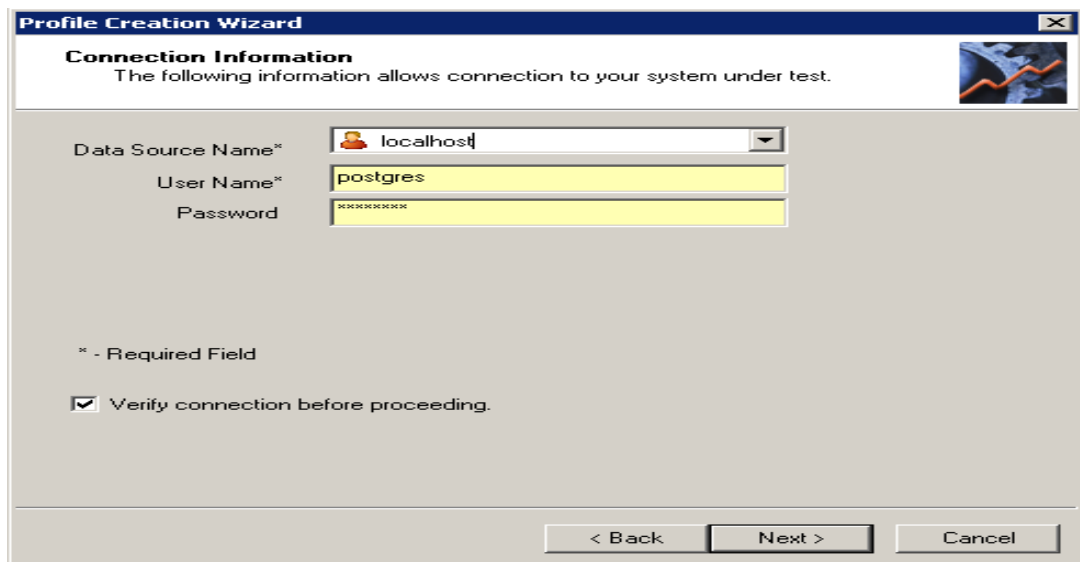
**Figura #2:** Tablas de la Base de Datos de TPC-B

Como se mencionó anteriormente ésta prueba se realizará con la herramienta Benchmark Factory, que permite ejecutar pruebas de carga y de rendimiento altamente escalable, capaz de simular el acceso de miles de usuarios a sus servidores de Bases de Datos (Ver figura #3).



**Figura #3:** Herramienta Benchmark Factory

El asistente para la creación del perfil permite incluir la información que Benchmark Factory utiliza para conectarse con el sistema bajo prueba. Esta información puede incluir, nombre del servidor, la dirección IP y nombre de usuario si es una conexión remota. Si es local sólo hay que poner local host, el usuario y contraseña del Gestor de Base de Datos (Ver figura #4).



**Figura #4:** Información de la conexión

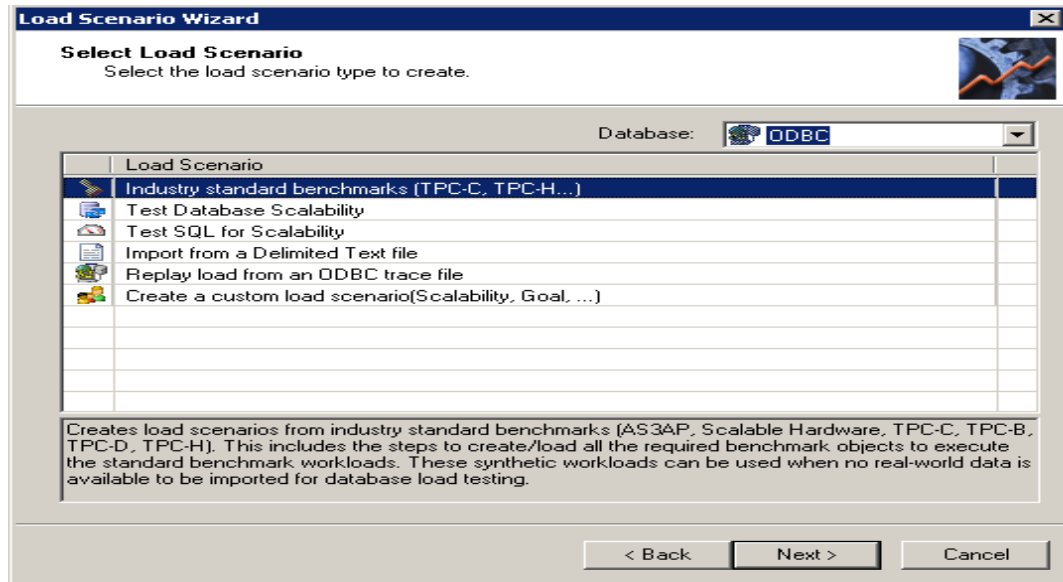
El asistente para la carga de un escenario es un punto de partida para la creación de scripts de Benchmark Factory que guía los pasos para construir una secuencia de comandos para completar las pruebas de carga de trabajo.

Este asistente permite configurar las siguientes entradas que se describen a continuación:

- ✓ Seleccionar el escenario a cargar.
- ✓ Seleccionar el tipo Benchmark.
- ✓ La escala de Benchmark.
- ✓ Los intervalos de Medición.

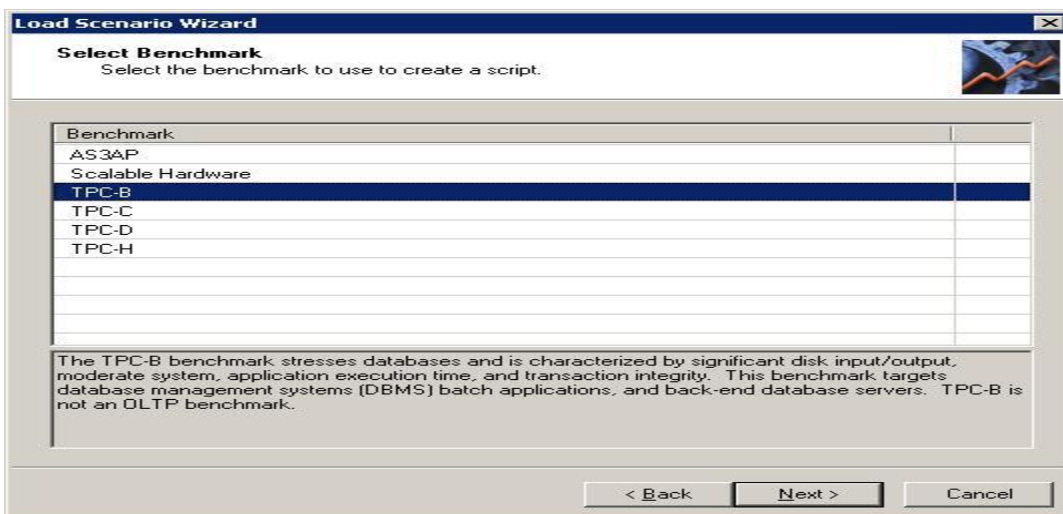
**Seleccionar el escenario a cargar:** Esta opción crea escenarios de carga de la industria de referencia, como se muestran (Ver figura #5). Esto incluye los pasos para crear / cargar todos los objetos necesarios

de referencia para ejecutar las cargas de trabajo de referencia estándar. Estas cargas de trabajo de síntesis se pueden utilizar cuando no hay datos del mundo real (Ver figura 5).



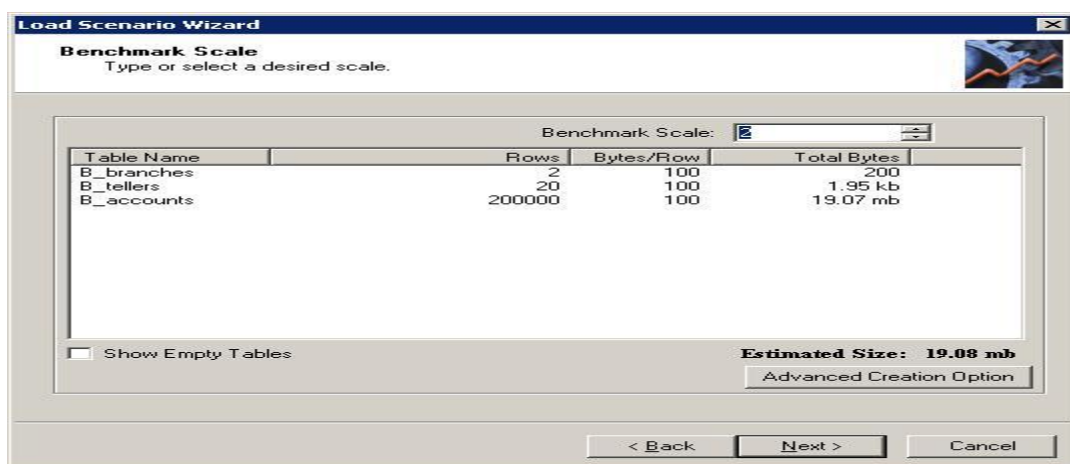
**Figura #5:** Selección del asistente a cargar

**Seleccionar el tipo Benchmark:** En este paso se procede a seleccionar la prueba Benchmark que se desee ejecutar, siendo seleccionada TPC-B (Ver figura 6).



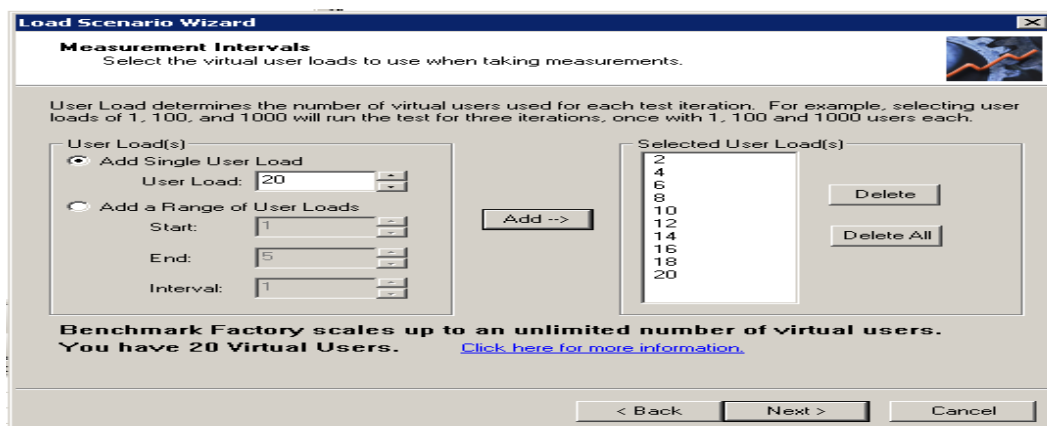
**Figura #6:** Selección del tipo de Benchmark

**Factores de escala de Benchmark:** Consiste en aumentar el tamaño de una Base de Datos durante el proceso de prueba. Al cambiar los factores de escala, el número de filas agregadas incrementa en las tablas, lo que permite cargar mayor cantidad de usuarios, poniendo un énfasis mayor en el sistema bajo prueba. Este valor de escala según la prueba que se realice inserta una cierta cantidad de tablas a la Base de Datos. Siendo el valor de la escala directamente proporcional a los valores que contienen estas tablas, debido a que al aumentar el valor de la escala aumentan los datos, las filas, los bytes por fila y el total de bytes siendo estos los valores con que se llenan las tablas (Ver figura #7).



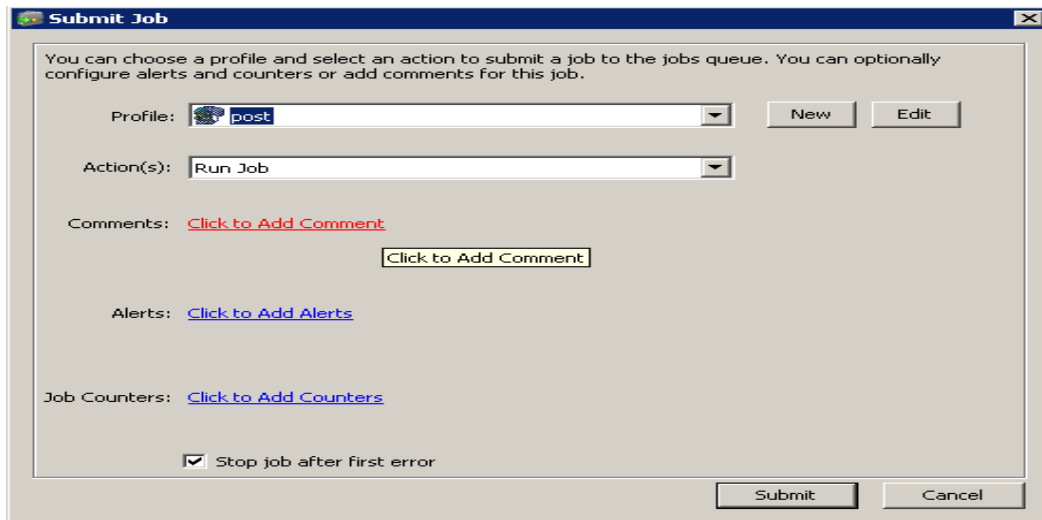
**Figura #7:** Factor de escala

**Intervalos de Medición:** Especifica el número de usuarios virtuales para cada iteración de una prueba, pero como es un trial sólo permite seleccionar hasta 20 usuarios (Ver figura #8).



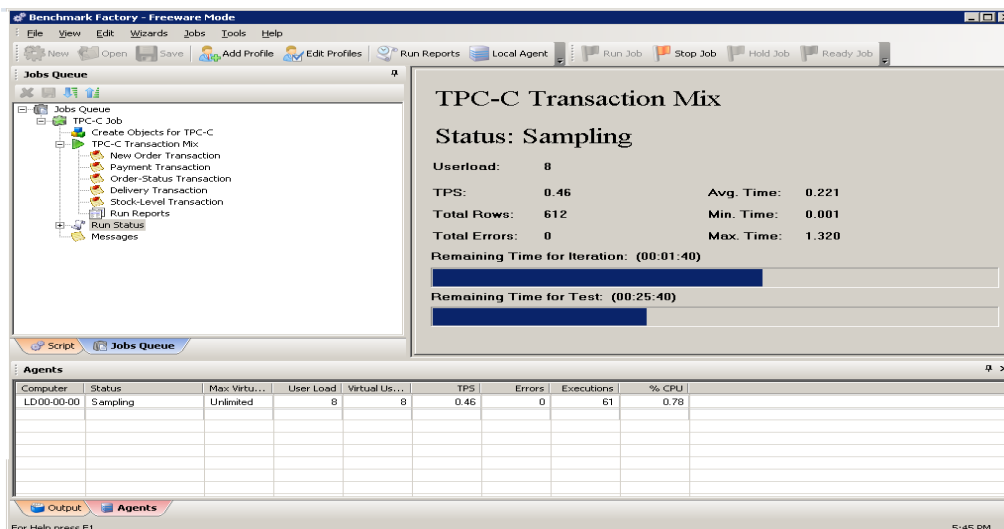
**Figura #8:** Intervalos de Medición

Al haber realizado todos los pasos anteriores, lo siguiente es enviar el trabajo, donde se puede elegir un perfil, configurar alertas y contadores o añadir comentarios para este trabajo. Al enviar el trabajo este asistente agrega el trabajo a la cola de puestos de trabajo (Ver figura #9).



**Figura #9:** Enviar trabajo

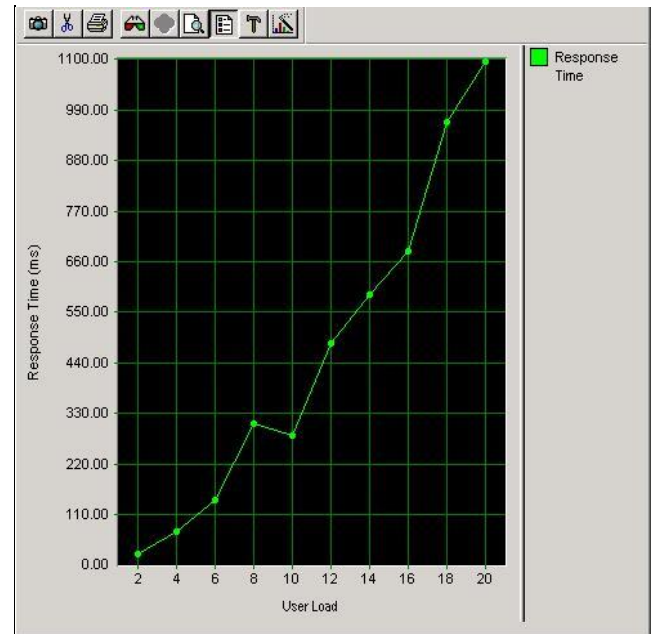
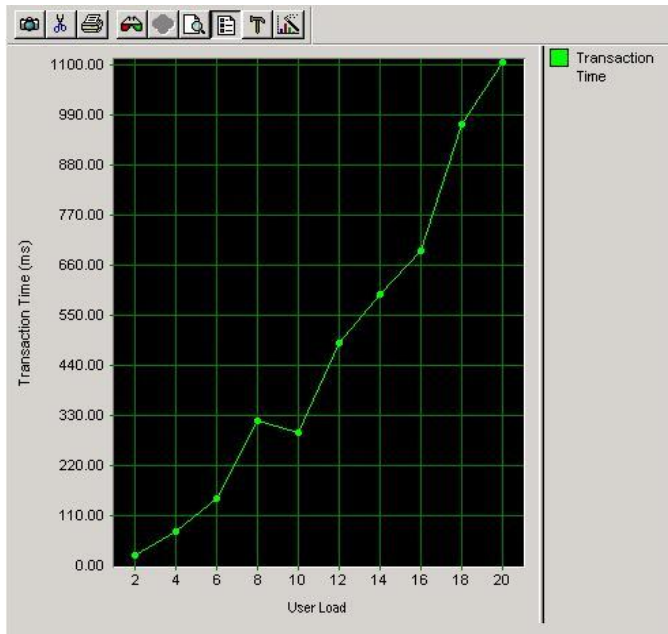
Después de haber enviado el trabajo prosigue la ejecución de la prueba (Ver figura #10).



**Figura #10:** Ejecución de la prueba



Luego de la ejecución de la prueba ésta arrojó los siguientes resultados, que se muestran en una nueva ventana después de ejecutar los reportes. Estos resultados se exponen mediante gráficas de comparaciones, donde se ven las distintas comparaciones que se hacen, como son el tiempo de transacción vs usuarios cargados (Ver gráfica #1), el tiempo de respuesta vs usuarios cargados (Ver gráfica #2) entre otras.



**Gráfica #1:** Tiempo transacción vs Usuarios.

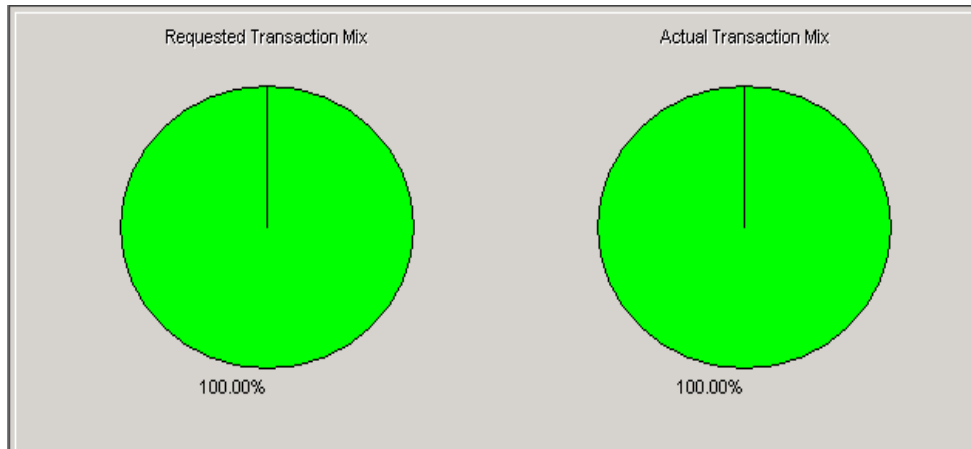
**Gráfica #2:** Tiempo respuesta vs Usuarios.

En la gráfica #1 se aprecia cómo el tiempo de la transacción para cada usuario cargado, en la mayoría de los resultados a medida que va aumentando la cantidad de usuarios realizando operaciones aumenta el tiempo en que la transacción es ejecutada. Debido a que todos estos usuarios acceden al mismo tiempo a la Base de Datos.

En la gráfica #2 los resultados se comportan similares a los de la gráfica # 1, mostrándose en ésta el tiempo de respuesta que demora el sistema en terminar la transacción ejecutada por cada usuario.

En la figura que se muestra a continuación, se representa cómo se comportaron los resultados de las transacciones ejecutadas, tanto las esperadas por el sistema como las actuales que realizó (Ver figura

#11). Teniendo como resultado que todas las transacciones esperadas fueron realizadas en su totalidad de acuerdo al por ciento establecido.



**Figura #11:** Transacciones Esperadas y Transacciones Actuales

Para un mejor análisis de los resultados, en esta figura se muestran todos los parámetros que se miden en la trayectoria de la prueba, mostrándose para cada usuario, la ejecución realizada, el tiempo de transacciones por segundo, la cantidad de bytes, los bytes por segundo y la cantidad de filas.

	Userload	Executions	TPS	BPS	Rows	Bytes	Errors
↑	2	11020	61.22	489.77	11020	88160	0
↑	4	7291	40.51	324.05	7291	58328	0
↑	6	5725	31.81	254.47	5725	45800	0
↑	8	3836	21.41	171.29	3836	30688	0
↑	10	4794	26.66	213.30	4794	38352	0
↑	12	3953	21.97	175.75	3953	31624	0
↑	14	3742	20.81	166.48	3742	29936	0
↑	16	3421	19.10	152.79	3421	27368	0
↑	18	3094	17.21	137.65	3094	24752	0
↑	20	3042	16.89	135.14	3042	24336	0

**Figura #12:** Transacciones Actuales

El Benchmark TPC-B expresa el resultado en TpsB (transacciones por segundo), calculándose de la siguiente forma:

$$\text{TpsB} = \text{Total de transacciones comenzadas y terminadas} / \text{tiempo transcurrido del intervalo.}$$

Realizando un total de transacciones de 49918, con un tiempo transcurrido de 277.59.

Dando el tpsB 179,83 transacciones por segundo, cumpliendo lo establecido con el requisito de rendimiento del proyecto.

Sirviendo estos resultados para un posterior análisis de estos valores contra los resultados de otra prueba realizada por TPC-B a la misma Base de Datos del proyecto AIMEP pero con un mayor número de usuarios y una mayor cantidad de información almacenada.

### 3.2.6 Documentación e Informe de los resultados

La prueba no arrojó ningún error en el transcurso de su ejecución y los resultados de la misma fueron satisfactorios a pesar de las limitaciones con que se ejecutaron éstas, debido a que la herramienta con la que se cuenta es un trial, no permite adicionar más de 20 usuarios los y resultados arrojados son positivos porque no es una cantidad significativa comparada con las capacidades de la BD.

### 3.2.7 Evaluación de las Pruebas

Una vez finalizadas las pruebas, el especialista encargado de la evaluación del proceso de las pruebas basándose en la planilla de evaluación de las pruebas desarrollada por los autores del presente trabajo de diploma evalúa el desarrollo de las mismas (Ver Tabla #7).

Para evaluar las pruebas el especialista se basó en los siguientes indicadores:

Indicadores definidos	
Indicadores a Evaluar	Evaluación
1. ¿Se planificaron correctamente las pruebas?	5
2. ¿Se identificaron los roles y recursos que intervendrán en la ejecución de las pruebas?	5
3. ¿Se diseñó correctamente los Casos de prueba	4

correspondiente a las pruebas Benchmark?	
Indicadores definidos	
4. ¿Al ejecutar las pruebas diseñadas se obtuvieron los resultados esperados?	4
5. ¿La ejecución de las pruebas está estrictamente establecido con el Plan de prueba?	4
6. ¿Se informaron y documentaron las NC?	5
<b>Evaluación final</b>	27

**Tabla #7:** Indicadores

Promedio es  $3+5+4+4+4+5/6 = 4.5$  la evaluación final del proceso de pruebas es Muy Bien

### 3.2.8 Depurar errores

En esta fase el equipo de desarrollo se reúne, ven las no conformidades encontradas en la fase anterior y las resuelve, pero como no se encontraron NC en la prueba aplicada, no hay errores que depurar.

### 3.3 Método Delphi

El método de validación Delphi es un método para obtener las opiniones de un panel de expertos. En esta técnica se interroga de forma individual a los expertos y se hace circular entre los integrantes del panel un resumen de sus opiniones. Este proceso se repite las veces que sean necesarias para lograr un determinado consenso. Se inicia enviando a los expertos una serie de cuestiones. La encuesta se remite de forma anónima a todos los integrantes del panel de modo que se evita el encuentro entre ellos. Esta precaución permite que las respuestas de unos no influyan en las de otros. Después de esta primera ronda, se agrupan las respuestas y se vuelve a enviar la información al panel de expertos. El número de rondas varía según el nivel de consenso deseado por el investigador, el grado de concreción de los ítems y del número de ítems surgidos.

Para aplicar el método se siguieron tres etapas fundamentales:

- ✓ Elección de experto.

- ✓ Elaboración del cuestionario para la validación de la propuesta.
- ✓ Desarrollo práctico y explotación de resultados.

### 3.3.1 Selección del grupo de expertos

En el proceso de selección de expertos primeramente se debe mencionar: ¿qué se entiende por experto? No es más que aquel individuo en sí como a un grupo de personas u organizaciones capaces de ofrecer valoraciones conclusivas de un problema en cuestión y hacer recomendaciones respecto a sus momentos fundamentales con un máximo de competencia.

Se tuvo en cuenta a la hora de seleccionar a los expertos (Ver Anexo 4) entre sus cualidades la disposición de participar en la encuesta como una característica importante a tener presente para que influya de manera positiva en la solución exitosa del problema tratado, por cuanto esta posibilita conocer si se incluye o no en la realización de la misma. La capacidad de análisis y de pensamiento; la propiedad de colectivismo y su espíritu autocrítico, se observó en la valoración de su grado de competencia, así como en la toma de decisión en el análisis del problema. Se conformó un panel integrado por especialistas de la Dirección de Calidad de Software de la universidad, especialistas de DATEC con experiencia en temas relacionados con la investigación.

Para seleccionar el grupo de expertos se analizaron las siguientes características.

- ✓ Conocimiento acerca de las Bases de Datos.
- ✓ Conocimientos y habilidades en actividades de pruebas de software.
- ✓ Conocimiento sobre la realización de procedimientos.
- ✓ Conocimiento sobre pruebas Benchmark.

Elegir los expertos atendiendo a las características mencionadas propició obtener resultados con calidad, junto a otras cualidades propias de estos como fueron: la seriedad, la honestidad, la sinceridad, la responsabilidad y otras en este sentido, que hicieron que las opiniones brindadas fueran confiables y válidas para el objetivo propuesto.

### 3.3.2 Cálculo del coeficiente por competencia

Lo que se obtiene del formulario de autovaloración es el coeficiente de competencia que se determina mediante la fórmula:  $K = \frac{1}{2} (k_c + k_a)$ , donde  $k_c$  es el coeficiente de conocimiento y  $k_a$  es el coeficiente de argumentación.

El coeficiente de conocimiento se obtiene de la primera pregunta del cuestionario y el coeficiente de argumentación de la segunda pregunta, el coeficiente de conocimiento se multiplica por 0.1 para que los resultados estén ajustados a probabilidades. El coeficiente de argumentación viene dado por la siguiente tabla de valores, (Ver tabla #8) según las respuestas del experto se suman todos los valores obtenidos dando como resultado  $k_a$ .

No	Fuentes de Argumentación	Grado de influencia		
		Alto	Medio	Bajo
1	Análisis realizado por usted	0.3	0.2	0.1
2	Experiencia	0.5	0.4	0.2
3	Trabajo de autores nacionales	0.05	0.05	0.05
4	Trabajo de autores extranjeros	0.05	0.05	0.05
5	Su propio conocimiento del tema	0.05	0.05	0.05
6	Su intuición	0.05	0.05	0.05
7	Certificaciones que ha obtenido en esa área	0.5	0.4	0.2

**Tabla #8:** Valores del coeficiente de Argumentación

El código para la interpretación de los coeficientes de competencia es el siguiente:

Si  $0.8 < k < 1.0$ , el coeficiente de competencia es alto

Si  $0.5 < k < 0.8$ , el coeficiente de competencia es medio.

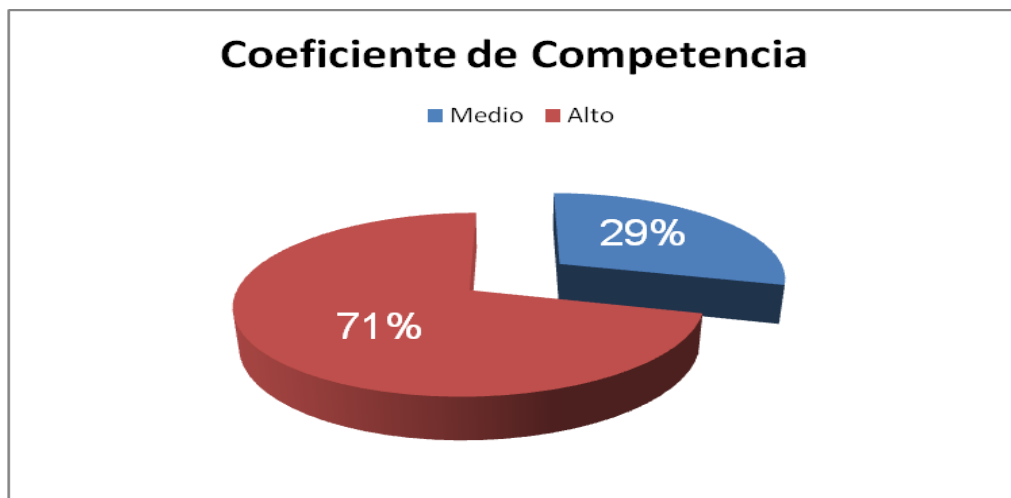
Si  $k < 0.5$  el coeficiente de competencia es bajo.

El resultado del coeficiente de competencia de todos los cuestionarios de autovaloración de los 7 expertos puede verse en el Anexo 6. A continuación se muestra el resultado general del coeficiente de competencia de los expertos que colaboraron (Ver tabla #9).

No. Exp.	P1	P2						Ka	Kc	K	Competencia
	Con	P2.1	P2.2	P2.3	P2.4	P2.5	P2.6				
1	7	0.3	0.4	0.05	0.05	0.05	0.05	0.9	0.7	0.8	Medio
2	8	0.2	0.4	0.05	0.05	0.05	0.05	0.8	0.8	0.8	Medio
3	9	0.2	0.4	0.05	0.05	0.05	0.05	0.8	0.9	0.85	Alto
4	8	0.3	0.4	0.05	0.05	0.05	0.05	0.9	0.8	0.85	Alto
5	9	0.2	0.4	0.05	0.05	0.05	0.05	0.8	0.9	0.85	Alto
6	9	0.3	0.4	0.05	0.05	0.05	0.05	0.9	0.9	0.9	Alto
7	9	0.2	0.5	0.05	0.05	0.05	0.05	0.9	0.9	0.9	Alto
<b>Total</b>	8,42	0.24	0.41	0.05	0.05	0.05	0.05	0.85	0.84	0.99	<b>ALTO</b>

**Tabla #9:** Resultados del coeficiente de competencia de los expertos.

A continuación se muestra el resultado general del coeficiente de competencia de los expertos que colaboraron (Ver figura #13).



**Figura #13:** Resultados del coeficiente de competencia de los expertos

### 3.3.3 Elaboración del cuestionario para la validación propuesta

La encuesta elaborada (Ver Anexo 5) consta de diez preguntas de tipos contables y abiertos, estas últimas permiten a los encuestados hacer una valoración crítica del tema lo cual es muy importante para conocer posibles limitaciones o ineficiencias del procedimiento propuesto. La encuesta se llevó a cabo de una manera anónima (se realizó haciendo uso del correo electrónico) para evitar influenciar en la opinión de los expertos.

Por otra parte, las preguntas de tipo contables como bien describe su nombre, permiten contabilizar ciertos aspectos que el autor considere importantes y que sirvan para mostrar gráficamente dichos resultados. Asimismo en este tipo de preguntas la escala para sus respuestas está dividida en 10, sin importar si esta división está propuesta en forma cuantitativa (%) o cualitativa. El nivel máximo representaría un 10 y el nivel más bajo representaría un 1. De esta forma, la escala sería única y permitiría mostrar con facilidad los resultados de las mismas.

### 3.3.4 Desarrollo práctico y explotación de resultados.

Resultado Final de la Validación de los indicadores propuestos (Ver tabla #10).

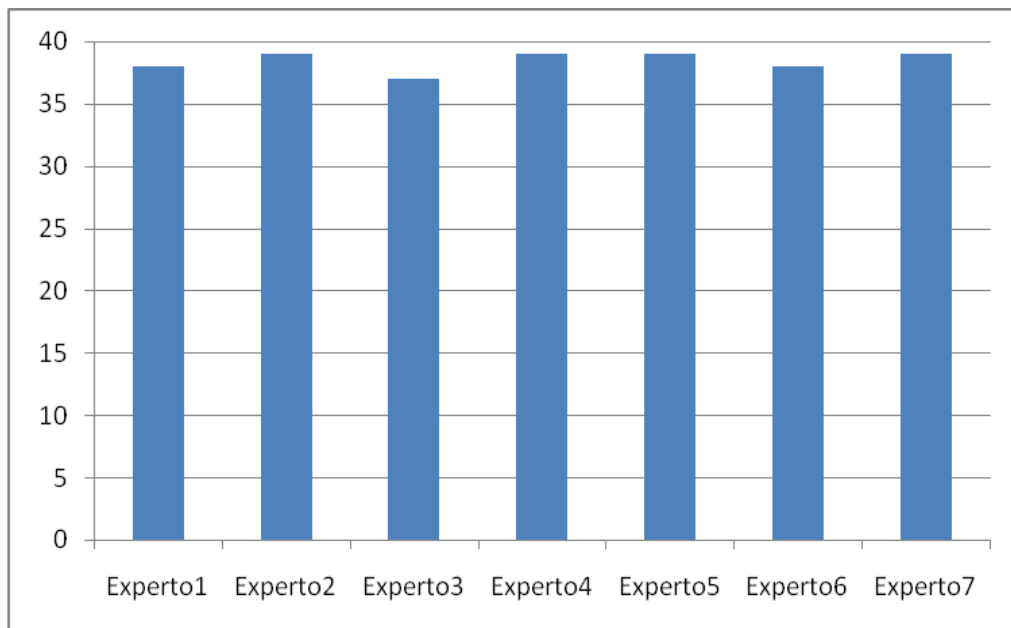
G	C/E	E1	E2	E3	E4	E5	E6	E7	EP
40	C1	10	9	10	10	10	10	10	9,85
	C2	9	10	9	9	9	9	9	9,14
	C3	10	10	9	10	10	9	10	9,71
	C4	9	10	9	10	10	10	10	9,71
20	C5	9	9	10	8	9	10	9	9,14
	C6	9	10	9	9	9	9	10	9,71
20	C7	10	9	10	9	10	8	10	9,42
	C8	10	8	9	9	10	9	9	9,14



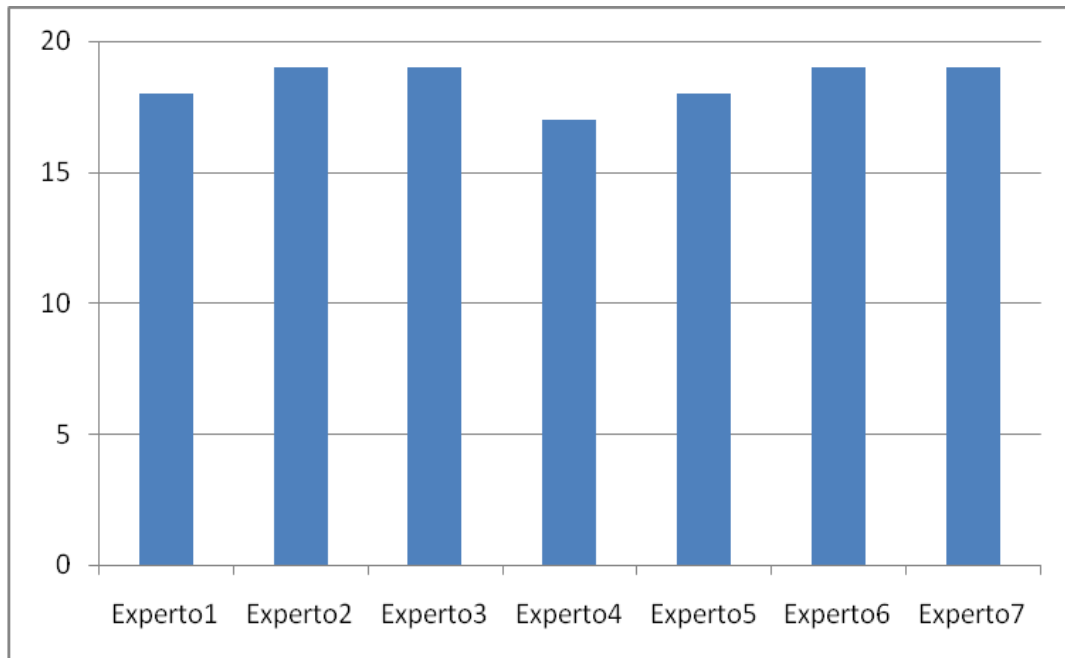
G	C/E	E1	E2	E3	E4	E5	E6	E7	EP
20	C9	10	9	10	10	8	9	9	9,28
	C10	10	9	9	10	10	10	10	9,71
T		96	93	94	94	95	92	96	94,28

**Tabla #10:** Criterios de los expertos

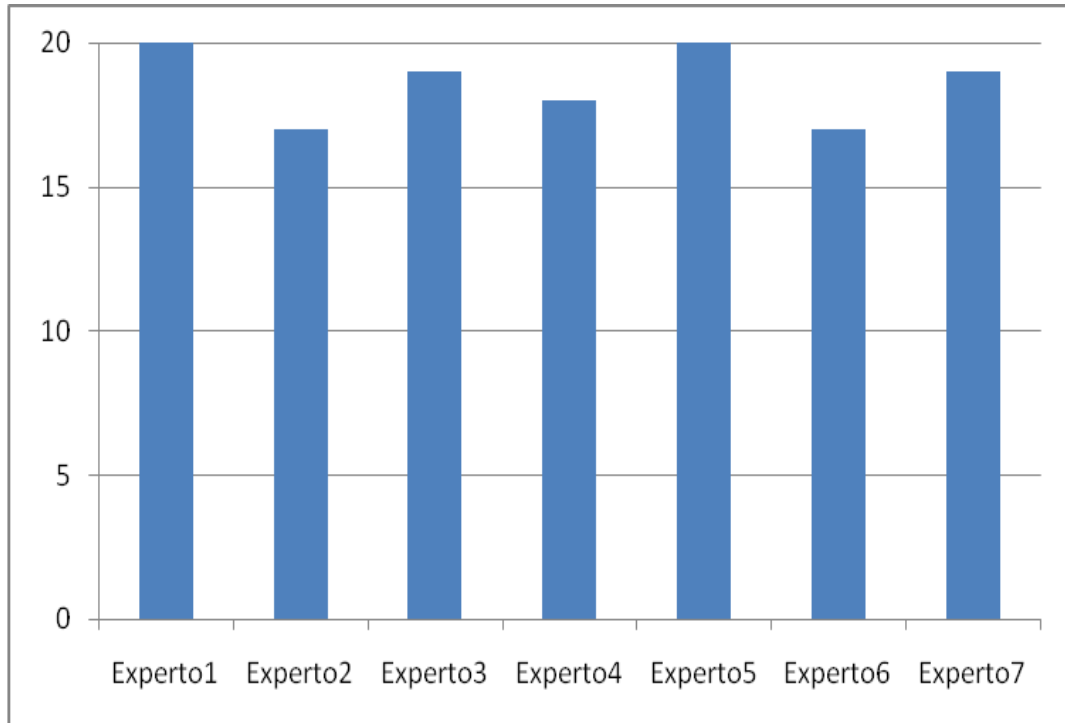
A continuación se muestran las gráficas de los grupos por criterios evaluados de cada uno de los expertos.



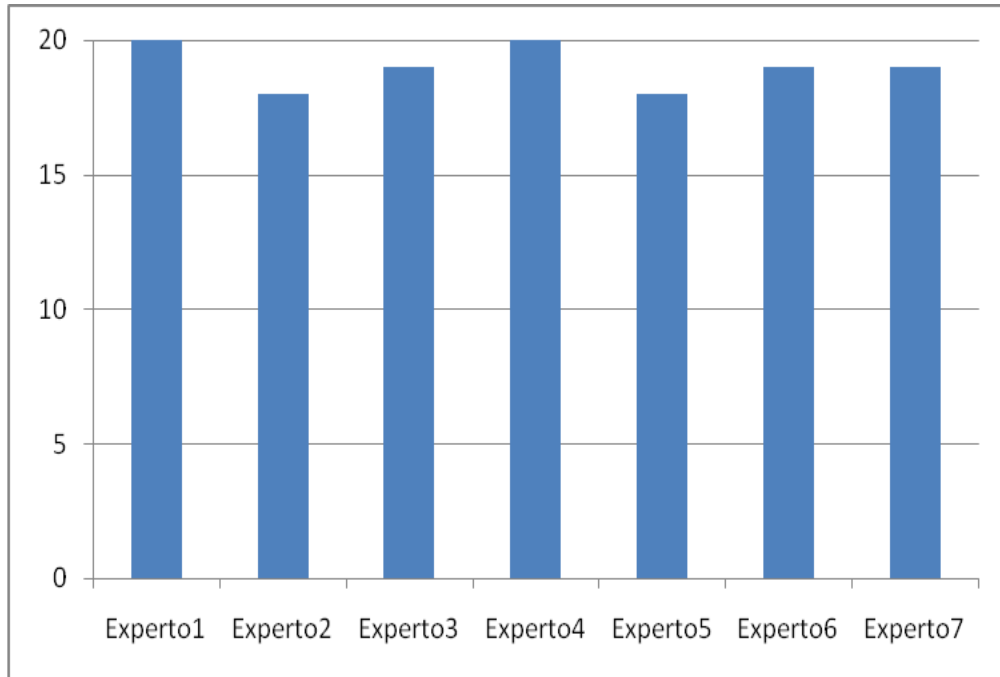
**Gráfica #3:** Grupo No.1 Criterios de mérito científico.



**Gráfica #4:** Grupo No.2 Criterios de implantación.



**Gráfica #5:** Grupo No.3 Criterios de flexibilidad.



**Gráfica #6:** Grupo No.4 Criterio de Impacto

De los 7 expertos seleccionados 2 evaluaron el procedimiento de regular y los 5 restantes lo evaluaron de bueno (Ver figura 14).



**Figura #14:** Evaluación final

### **Conclusiones**

En este capítulo se aplicó el procedimiento a través de la prueba TPC-B Benchmark, se emplearon además, los Casos de pruebas y la planilla de evaluación de las pruebas realizadas por los autores del trabajo investigativo, con todos estos resultados se validó la propuesta del procedimiento para pruebas Benchmark para Bases de Datos. Utilizando el método de validación Delphi, se seleccionaron 7 expertos y se realizaron 10 preguntas relacionadas con el procedimiento planteado.

## CONCLUSIONES GENERALES

Con el desarrollo del presente trabajo de diploma se cumplieron los objetivos trazados, logrando arribar a las siguientes conclusiones:

- ✓ Se desarrolló un procedimiento para evaluar el rendimiento de las Bases de Datos mediante las pruebas Benchmark definiendo actividades, roles y materiales.
- ✓ Se realizó la validación del procedimiento mediante su aplicación y la utilización el método de Delphi, por un conjunto de 7 expertos arrojando como resultado final una evaluación Buena.

Lo antes planteado le da cumplimiento al objetivo principal: Procedimiento para pruebas de software Benchmark para evaluar el rendimiento de las Bases de Datos en la facultad 6.

## RECOMENDACIONES

Existen algunos aspectos que se desprenden de este trabajo investigativo en los cuales serían útiles seguir profundizando y otros que se recomiendan tener en cuenta como son:

- ✓ Aplicar el procedimiento en los proyectos de la facultad 6, en otras facultades y en CALISOF.
- ✓ Colaborar con las búsquedas de otras herramientas para la automatización de las pruebas Benchmark o la implementación de una herramienta que realice éste tipo de prueba.
- ✓ Automatizar el procedimiento propuesto por la investigación para así lograr una mejora en el proceso de la prueba.

## REFERENCIAS BIBLIOGRÁFICAS

1. **1990, IEEE.** IEEE Standard Glossary of Software Engineering Terminology. 28 de septiembre de 1990. p. <http://www.idi.ntnu.no/grupper/su/publ/ese/iee>.
2. **(1990), IEEE.** Computer Dictionary. Computer Society.
3. **ISO, (2000).** Sistema de gestion de la calidad.Requisitos. 2000
4. **Pressman, R.S.** “Ingeniería del Software. Un enfoque práctico”. s.l. : I Mc Graw Hill, Quinta ed. 2002.
5. **Piattini, Mario G.** “Análisis y diseño de aplicaciones informáticas de gestión, una perspectiva de ingeniería del software”. 2004.
6. **Myers, Glenford , .** “The Art of Software Testing”. s.l. : John Wiley & Sons. ISBN 0-471-04328-1, segunda edición 2004.
7. **Miguel-Angel, Estándar ISO 9126 del IEEE y la Mantenibilidad.** Descripción del concepto de Mantenibilidad según el estándar ISO 9126 del IEEE. <http://cnx.org/content/m17461/latest/>.
8. **Garcerant, I.** Niveles de Pruebas. 2008.
9. **Lores Sánchez, Linet and Monné Roque, Diana.** Aplicación de las pruebas de liberación al Sistema Informático De Genética Médica. Trabajo de Diploma, Universidad de las Ciencias Informáticas, Ciudad de La Habana : s.n., 2009.
10. **Espinoza Bárbara, Quinta Vanessa, Vegas Alexandra.** Pruebas de rendimiento. Julio, 2005. [http://carolina.terna.net/ingsw3/datos/Pruebas\\_de\\_Desempe%F1o.pdf](http://carolina.terna.net/ingsw3/datos/Pruebas_de_Desempe%F1o.pdf).
11. **Soriano, Amelia.** Tipos de Prueba. [http://carolina.terna.net/ingsw3/datos/Tipos\\_Prueba.pdf](http://carolina.terna.net/ingsw3/datos/Tipos_Prueba.pdf)
12. **(AVL). Vega Jara, y otros.** De la prueba de carga. 2004.
13. **Van Veenendaal, Erik.** Standard glossary of terms used in Software Testing. Version 2.0 (dd. December, 2nd 2007).

14. **Ramalinga Moorthy, Ramya.** Software Performance Testing Handbook.
15. **Lanzillotta, Analía.** Master Magazine Archivo 2004 <http://www.mastermagazine.info/termino/4012.php>.
16. **Gutiérrez, Javier Jesús.** (2005). "Generación de pruebas de sistema a partir de la especificación funcional". [http://www.lsi.us.es/~javierj/investigacion\\_ficheros/NDEAv25.pdf](http://www.lsi.us.es/~javierj/investigacion_ficheros/NDEAv25.pdf).



## BIBLIOGRAFÍA

- Aseguramiento de la Calidad. [En línea] Disponible en:  
<http://www.csae.map.es/csi/metrica3/calidad.pdf>
- Biblioteca (Universidad de las Ciencias Informáticas). [En línea] Disponible en:  
<http://biblioteca.uci.cu/>
- Benchmark – Best Practices.[ En línea] 2009 Disponible en:  
<http://www.softwaremetrics.com/benchmark.htm>
- Benchmarking Software and Discussion. [En línea] 2009 Disponible en:  
<http://www.overclock.net/benchmarking-software-discussion/>
- IEEE Standard for Software Reviews – IEEE Std 1028-199. [En línea] [Citado el: 8 de marzo de 2010.] Disponible en:  
<http://pesona.mmu.edu.my/~wruslan/SE2/Readings/detail/Reading-6.pdf>
- La Calidad en general y los Modelos de Calidad. Modelos de Gestion de la Calidad del Software . [En línea] [Citado el: 16 de noviembre de 2009.] Disponible en:  
<http://modelosdegestiondelacalidad.blogspot.com/>
- Plataforma Moddle UCI. [En línea] 2009 Disponible en:  
<http://teleformacion.uci.cu/>
- Pagina oficial de las pruebas TPC. [En línea] 2009 Disponible en:  
<http://www.tpc.org/>
- Pruebas Benchmark de soluciones cliente/servidor. [En línea] 2009 Disponible en:  
[http://www.ideoe.upm.es/jideoe05/descargas/sesion\\_07\\_03.pdf](http://www.ideoe.upm.es/jideoe05/descargas/sesion_07_03.pdf)

- Punto de Referencia. [ En línea] 2009 Disponible en:  
<http://www.bitpipe.com/tlist/Benchmark-Testing.html>
- Parallel Universe. [En línea] 2009 Disponible en:  
<http://universoparalelo-en.blogspot.com/2008/06/one-of-these-recent-days-i-was-reading.html>
- Pruebas de Rendimiento. [En línea] 2009 Disponible en:  
<http://www.csi.map.es/csi/silice/Hw-wst40.html>
- **Pressman, Roger S.** Ingeniería de Software. : Mc Graw Hill. 1995.
- **Pressman, Roger S.** Ingeniería de Software. Un enfoque práctico. 1998.
- **Pressman, Roger S.** Ingeniería de Software. : Mc Graw Hill. 2002.
- **Pressman, Roger S.** Ingeniería de Software. Un enfoque práctico. 2005.
- **Pressman, Roger S.** Ingeniería de Software. Un enfoque práctico. 2006.
- Quest Software(En Español). [En línea] 2009 Disponible en:  
[http://www.alta-tecnologia.com/sybase/syb\\_bench.php](http://www.alta-tecnologia.com/sybase/syb_bench.php)
- Revista Benchmark. [En línea] 2009 Disponible en:  
<http://www.revistabenchmark.com/>
- Tutorial de que es un Benchmark. [En línea] 2009 Disponible en:  
<http://www.emagister.com/que-es-benchmark-cursos-636811.htm#programa>
- The Evolution of TPC Benchmarks [En línea] 2009 Disponible en:  
<http://www.hpl.hp.com/techreports/tandem/TR-93.1.pdf>

**ANEXOS****ANEXO 1:** Planilla de solicitud de evaluación**Nombre del proyecto:****Nombre y apellidos del solicitante:****Teléfonos:****E-mail del solicitante:****Etapa de Evaluación:**

Aspectos Necesarios.	Etapa de Evaluación	Responsable.

\_\_\_\_\_  
**Aceptada**\_\_\_\_\_  
**No Aceptada**\_\_\_\_\_  
**Firma del Líder de la Comisión**\_\_\_\_\_  
**Firma del Jefe de Proyecto**

**ANEXO 2:** Planilla plan de prueba**Plan de Pruebas Benchmark**

&lt;Versión 1.0&gt;

**Control de versiones**

<i>Fecha</i>	<i>Versión</i>	<i>Descripción</i>	<i>Autor</i>
2/04/10	<1.0>	Plan de Prueba.	Elieny Rodriguez Yoel Sánchez

**Reglas de Confidencialidad****Clasificación:** <USO INTERNO>**Forma de distribución:** <PDF Digital>

Este documento contiene información propietaria de **Calisoft “Centro para la Excelencia en el Desarrollo de Proyectos Tecnológicos”**, y es emitido confidencialmente para un propósito específico.

El que recibe el documento asume la custodia y control, comprometiéndose a no reproducir, divulgar, difundir o de cualquier manera hacer de conocimientos público su contenido, excepto para cumplir el propósito para el cual se ha generado.

Estas reglas son aplicables a las **5** páginas de este documento.

**Introducción**

Este documento se confecciona con el objetivo de definir el Plan de Pruebas Benchmark para los sistemas de Base de Datos de la facultad 6.

**Objetivos**

Este documento describe el plan de pruebas para evaluar las pruebas Benchmark en los sistemas de Bases de Datos de la facultad 6. Concretamente persigue:

- Definir los recursos necesarios para ejecutar la evaluación.
- Describir la estrategia de pruebas a seguir en el proceso de evaluación.
- Definir el cronograma de las pruebas.
- Documentar los resultados que se obtienen de las actividades de prueba.

### Alcance

Tipo de Prueba	Descripción del tipo de pruebas	Herramientas de Pruebas
Benchmark	Es un tipo de prueba que compara el rendimiento de un elemento nuevo o desconocido a uno de carga de trabajo de referencia conocido.	Benchmark Factory for Database

### Estrategia de Evolución del Plan

El plan será chequeado diariamente y se realizarán los ajustes que sean pertinentes al cronograma si alguna situación en particular modifica los eventos previstos. Al finalizar cada jornada de trabajo, el equipo de pruebas se reúne para verificar el cumplimiento del plan y realizar dichos ajustes de ser pertinentes.

Se realizará un documento con las modificaciones al cronograma, el cual será distribuido entre los participantes de las pruebas para con ello comunicar los cambios.

Las propuestas de modificaciones al plan serán presentadas para su análisis en reunión de chequeo conjunto, participando los involucrados: equipo de proyecto y el Laboratorio Industrial de Prueba de Software (LIPS).

### Recursos

## Roles y Responsabilidades

<b>Rol</b>	<b>Cantidad</b>	<b>Responsabilidad</b>
Probadores	2	Es el principal ejecutor en el proceso de pruebas en cada Turno de Trabajo. Es su responsabilidad garantizar el registro de las No Conformidades de la manera más clara, concisa e íntegra posible.
Representante del proyecto	1	Facilitar la documentación para las pruebas. Garantizar que se respondan las NC. Mantener relación con el equipo de pruebas. Tiene que tener poder de decisión.
Asesor de Calidad. (facultad)	1	Coordinar las pruebas con el equipo de desarrollo del proyecto y estar pendiente de la ejecución de las mismas. Además de apoyar las pruebas con los estudiantes del grupo de calidad de la facultad.

**Entorno de Prueba Ideal**

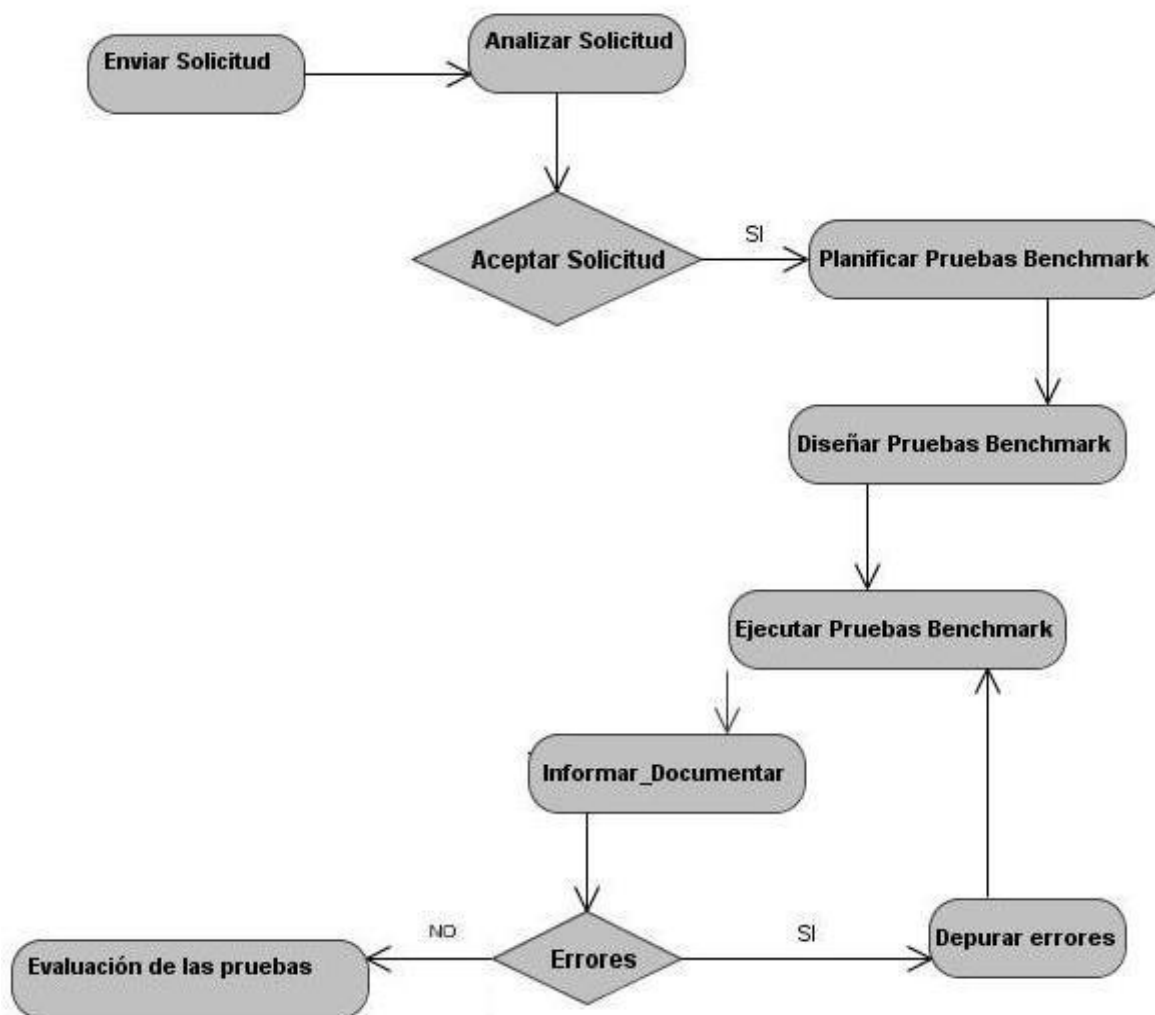
El software que se necesita para un funcionamiento correcto son las Bases de Datos Oracle, SQL Server, DB2 UDB, Sybase ASE, MySQL y otras que usen conectividad ODBC. Además, se necesita un PC

Pentium 160 o superior (Pentium II recomendado), 32 MB de RAM, 100 MB de espacio libre en disco y un sistema operativo Windows 95, 98 o NT (NT recomendado)

### Entorno de Prueba Real

La realización de estas pruebas se ejecutará en un PC Pentium 4, 504 MB de RAM, 25 GB de espacio libre en disco, Windows NT. El software de Base de Datos Postgres.

### Estrategia de Prueba



**Cronograma de Trabajo**

Recoge las fechas (inicio y fin) en las cuales se llevará a cabo el proceso de prueba.

<b>N o.</b>	<b>Tarea</b>	<b>Fecha</b>	<b>Responsable</b>	<b>Participantes</b>	<b>Etap a</b>	<b>Observacion es</b>
1	Enviar Solicitud	10/5/2010	Elieny Rodriguez Yoel Sánchez	Elieny Rodriguez Yoel Sánchez		
2	Analizar Solicitud	10/5/2010	Elieny Rodriguez Yoel Sánchez	Elieny Rodriguez Yoel Sánchez		
3	Aceptar Solicitud	10/5/2010	Elieny Rodriguez Yoel Sánchez	Elieny Rodriguez Yoel Sánchez		
4	Planificar Pruebas Benchmark	10/5/2010	Elieny Rodriguez Yoel Sánchez	Elieny Rodriguez Yoel Sánchez		
5	Diseñar Pruebas Benchmark	10/5/2010	Elieny Rodriguez Yoel Sánchez	Elieny Rodriguez Yoel Sánchez		
6	Ejecutar Pruebas Benchmark	10/5/2010	Elieny Rodriguez Yoel Sánchez	Elieny Rodriguez Yoel Sánchez		
7	Informar documentar	10/5/2010	Elieny Rodriguez Yoel Sánchez	Elieny Rodriguez Yoel Sánchez		
8	Evaluación de las Pruebas	10/5/2010	Elieny Rodriguez Yoel Sánchez	Elieny Rodriguez Yoel Sánchez		
9	Depurar Errores	10/5/2010	Elieny Rodriguez Yoel Sánchez	Elieny Rodriguez Yoel Sánchez		



**ANEXO 3: Planilla Caso de Prueba****Descripción General**

Base de datos del proyecto Ambiente Integrado para la Modelación y Ejecución de Procesos de la facultad 6.

**Secciones a probar**

ID del escenario	Escenarios de la sección	Usuarios Cargados	Resultado esperado		Resultado de la prueba		CUMPLE	
			Cantidad Transacción	Tiempo de transacción(s)	Cantidad Transacción	Tiempo de transacción(s)	Si	No
1	TPC-B	2	1	3-5	11020	61,22	X	
		4	1	3-5	7221	40,51	X	
		6	1	3-5	5725	31,81	X	
		8	1	3-5	3836	21,41	X	
		10	1	3-5	4794	26,66	X	
		12	1	3-5	3953	21,97	X	
		14	1	3-5	3742	20,81	X	
		16	1	3-5	3421	19,10	X	
		18	1	3-5	3094	17,21	X	
		20	1	3-5	3042	16,89	X	

**ANEXO 4: Encuesta para determinar el coeficiente de competencia de los expertos.**

Compañero (a): En la presente tesis, se desea someter a la valoración de un grupo de expertos una propuesta de procedimientos para pruebas Benchmark a Base de Datos, para garantizar un mejor rendimiento del software de Base de Datos que se producen en la facultad 6. Para ello se necesita conocer el grado de dominio que usted posee sobre el tema de las pruebas Benchmark y con ese fin se desea que responda lo que a continuación se le solicita.

**Nombre y Apellidos:** \_\_\_\_\_

**Labor que realiza:** \_\_\_\_\_

**Años de experiencia:** \_\_\_\_\_ **Especialidad:** \_\_\_\_\_

**Categoría docente:** \_\_\_\_\_ **Categoría científica:** \_\_\_\_\_

1.- Marque con una cruz (X) el grado de conocimiento que UD. tiene sobre la temática que se investiga:

0	1	2	3	4	5	6	7	8	9	10
---	---	---	---	---	---	---	---	---	---	----

2.- Marque con una cruz (X) las fuentes que le han servido para argumentar el conocimiento que tiene UD. de la temática que se investiga.

No.	Fuentes de argumentación	Grado de Influencia.		
		Alto	Medio	Bajo
1.-	Análisis realizado por Ud.			
2.-	Experiencia.			
3.-	Trabajos de autores nacionales.			
4.-	Trabajos de autores			
5.-	Su propio conocimiento del			
6.-	Su intuición			

**ANEXO 5:** Encuesta para los Expertos

Guía para informar el peso de los criterios.

Compañero (a): En la presente tesis, se desea someter a la valoración de un grupo de expertos una propuesta de procedimientos para pruebas Benchmark a Base de Datos, para garantizar un mejor rendimiento del software de Base de Datos que se producen en la facultad 6. Para ello se necesita conocer el grado de dominio que usted posee sobre el tema de las pruebas Benchmark y con ese fin se desea que responda lo que a continuación se le solicita.

**Nombre y Apellidos:** \_\_\_\_\_

**Labor que realiza:** \_\_\_\_\_

**Años de experiencia:** \_\_\_\_\_ **Especialidad:** \_\_\_\_\_

**Categoría docente:** \_\_\_\_\_ **Categoría científica:** \_\_\_\_\_

Le otorgará un peso a cada criterio de acuerdo a su opinión y el peso total de cada grupo debe sumar:

Grupo No.1..... 40

Grupo No.2..... 20

Grupo No.3.....20

Grupo No.4.....20

Para que el peso total asignado sea 100. Cada Factor que se mida tiene un valor de 10 puntos.

**Grupo No 1: Criterios de mérito científico.**

Valor científico de la propuesta.

Peso.....

Calidad de la investigación

Peso.....

Contribución científica.

Peso.....

Responsabilidad científica y profesionalidad de los investigadores.

Peso.....

**Grupo No 2: Criterios de implantación.**

Necesidad de empleo de la propuesta.

Peso.....

Posibilidades de aplicación.

Peso.....

**Grupo No 3: Criterios de flexibilidad.**

7. Adaptabilidad a entidades dedicadas a evaluar la calidad de los productos de software. Peso.....

8. Capacidad del proceso de evaluación para la admisión de cambios que impliquen mejoras.

Peso.....

**Grupo No 4. Criterios de impacto.**

9. Impacto en el área para la cual está destinada la guía.

Peso.....

10. Organización en el proceso de desarrollo.

Peso.....

**Categoría final del proyecto**

\_\_\_ Excelente: Alta novedad científica, con aplicabilidad y resultados relevantes.

\_\_\_ Bueno: Novedad científica, resultados destacados.

\_\_\_ Aceptable: Suficientemente bueno con reservas.

\_\_\_ Cuestionable: No tiene relevancia científica y los resultados son malos.

\_\_\_ Malo: No aplicable.

Valoración final, sugerencias del experto para mejorar la calidad del proyecto.

**GLOSARIO**

1. **ISO:** Organización Internacional para la Estandarización.
2. **UCI:** Universidad de Ciencias Informáticas.
3. **OLTP:** Procesamiento de Transacciones en Línea.
4. **TPC:** Transaction Procession Performance Council.
5. **SPEC:** Standard Performance Evaluation Corporation.
6. **DATEC:** Centro de Tecnologías de Gestión de Datos.
7. **ANSI:** Instituto Nacional de Normalización Estadounidense.
8. **NC:** Planilla de no conformidad.
9. **SQL:** Lenguaje de Consulta Estructurado.
10. **AS3AP:** ANSI SQL Escalable y Portátil.
11. **AIMEP:** Ambiente Integrado para la Modelación y Ejecución de Procesos.