

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA DE SISTEMAS

BENCHMARK PARA EL USO DE TECNOLOGÍAS RELACIONADAS A SERVIDOR DE APLICACIONES GLASSFISH Y JBOSS

**PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO ESPECIALISTA
EN SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN**

WASHINGTON RODRIGO ALMEIDA MOYANO

washo_am@hotmail.com

DANIEL FERNANDO IMBACUÁN LÓPEZ

danobts@hotmail.com

DIRECTOR: MSc. FRANCISCO VILLAVICENCIO OTAÑEZ

fvillavi@yahoo.com

QUITO, Mayo 2009

DECLARACIÓN

Nosotros, WASHINGTON RODRIGO ALMEIDA MOYANO y DANIEL FERNANDO IMBACUÁN LÓPEZ, declaramos bajo juramento que el trabajo aquí descrito es de nuestra autoría; que no ha sido previamente presentada para ningún grado o calificación profesional; y, que hemos consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedemos nuestros derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

WASHINGTON RODRIGO
ALMEIDA MOYANO

DANIEL FERNANDO IMBACUÁN
LÓPEZ

CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por WASHINGTON RODRIGO ALMEIDA MOYANO y DANIEL FERNANDO IMBACUÁN LÓPEZ, bajo mi supervisión.

FRANCISCO VILLAVICENCIO

DIRECTOR DE PROYECTO

AGRADECIMIENTO

En primer lugar al Divino Niño Jesús del cual soy devoto por todo el apoyo brindado durante la realización del presente trabajo brindando me esta inmensa alegría y el permitirme crecer en el campo profesional.

A mi padres que con su apoyo constante me impulsaron a alcanzar esta meta, darles gracias que siempre estuvieron pendientes durante toda mi vida estudiantil del cual estoy muy agradecido por su esfuerzo y sacrificio.

A mis hermanos que confiaron en mi y me tiene como su punto de referencia para lograr las metas que uno se proponga, su apoyo y amor que me brindan e impulsan a crecer.

A Daniel mi compañero de mil batallas, que durante todo este tiempo se convirtió en un hermano mas el cual le agradezco por compartir su confianza, respeto y que con su esfuerzo y apoyo se logro cumplir este objetivo.

Por ultimo a mis compañeros de universidad que son mi segunda familia que estuvieron en las buenas y en las malas, todo esa ayuda brindada durante el tiempo que estuvimos juntos en la facultad.

WASHO

DEDICATORIA

A mi padre Jorge, mi madre Marina mis hermanos Diego, Andres y Mishelle por todo su amor y confianza entregada a este humilde servidor.

Y para todas las personas que durante todo el tiempo que hemos compartido me brindaron su amor, respeto y apoyo.

WASHO

AGRADECIMIENTO

Infinitamente a Dios quien ha sido la guía primordial para todas las actividades cotidianas, y entre otras cosas para la realización del trabajo contenido en estas páginas

A mi esposa Rosquillita, quien ha estado junto a mí en los momentos más difíciles y más hermosos, en el día a día, apoyando con fuerza cuando todo parecía decaer.

A mis padres por su gran sacrificio inquebrantable durante todos mis años de vida estudiantil, su amor, paciencia y fe.

A mi compañero y amigo de Tesis quien demostró fortaleza para culminar con éxito esta y todas las actividades que se nos han puesto adelante.

A mi hermano y a toda mi familia de Ecuador y México, gracias por su valiosísimo apoyo.

DANIEL

DEDICATORIA

A todos quienes se han esmerado por darme todo el amor y apoyo

Mi esposa Rosquillita, Mis padres Wilson y Consuelito y mi hermano Alejandro.

DANIEL

INTRODUCCIÓN

Durante los últimos años ha resurgido con fuerza renovada el interés por parte de las empresas actuales de poner en producción Sistemas de Información, que provean mayor valor a sus negocios. La amplia difusión y fácil acceso a Internet han permitido generar no solo una gran infraestructura, sino que han sido de central importancia para concebir nuevas tecnologías, las cuales permitan desarrollar Sistemas de Información: flexibles, escalables, y de rápido desarrollo.

La plataforma J2EE es una especificación que intenta resolver los problemas antes presentados, brindando servicios de persistencia, seguridad, manejo de transacciones, presentación web, entre otros. Por ende todas estas características están presentes en los Servidores de Aplicaciones que manejan esta especificación.

Lo que da a manera de estudio el determinar cual de ellos tiende a brindar una mejor administración de los recursos como CPU, MEMORIA, RED y TIEMPOS DE RESPUESTA a las solicitudes o peticiones del usuario, da la pauta para generar un sistema de comparación (BENCHMARK) que por medio de los datos obtenidos de mediciones nos den en forma más clara cual es la mejor opción a la hora de desplegar una aplicación

En la actualidad los Servidores de Aplicaciones basados en especificación J2EE de más aceptación y que fueron el motivo para este estudio son GLASSFISH y JBOSS, ambos se mantienen en la misma línea de administración de aplicaciones Web. Además proveen servicios típicamente reservados para EJB's como soporte transaccional, clustering, etc. con el fin de que los administradores se beneficien de alguna de las características de escalabilidad de estos servidores.

El presente trabajo muestra la experiencia de desarrollar un Sistema Informático (Portal Web y Benchmark) basado completamente en la especificación J2EE y su puesta en marcha utilizando los Servidores de Aplicaciones mencionados, los cuales servirán a modo

de comparación y permitirán determinar que servidor gestiona las operaciones del Portal Web de manera más eficiente.

Independientemente de los resultados que las mediciones del Benchmark arrojen del despliegue del portal Web, de antemano conocemos que ambos tienden a brindar los mismos servicios aunque poseen características particulares por lo que el análisis comparativo entre ambos servidores abarcará desde las especificaciones técnicas, la arquitectura de cada uno y la manera de administrar los componentes.

El benchmark para su desarrollo sigue un estándar genérico que al igual que el desarrollo del sistema se rige por etapas, las mismas que deben tener un seguimiento para conseguir un software que obtenga resultados reales para obtener conclusiones fiables acerca de los objetivos propuestos.

Estudiar los resultados del benchmark conlleva a tener un conocimiento específico de los parámetros estadísticos que son tomados en cuenta para interpretar los resultados al igual que el tipo de gráficos para representarlos, ya que estos son los que explican de mejor manera el rendimiento de los servidores, todos los Benchmarks realizados para discernir sobre el rendimiento de herramientas informáticas utilizan este conocimiento como base para poder explicar los comportamientos y concluir de manera veraz las interpretaciones.

Del gran conjunto de procedimientos y metodologías que hoy en día existen para desarrollar de manera eficiente software es necesario seleccionar uno adecuado, tomando muy en cuenta la tendencia de la mayoría de las empresas dedicadas al desarrollo, sumándole a esto que se trata de un software que contiene 2 partes, la primera que involucra un prototipo de Portal Web y la otra correspondiente a las Mediciones de Benchmark en sí, además es necesario una metodología que comprenda fases e iteraciones que permita ir puliendo capa por capa y resultando en un entregable de calidad, finalmente se debe tratar esta metodología con un lenguaje acorde al desarrollo del sistema que sea comprensible al momento de entender la evolución en cada una de las fases.

El Portal Web es un software prototipo que pretende simular un gran conjunto de tareas o servicios que gran cantidad de empresas prestan a través del Internet, es decir la simulación

de un e-commerce dedicado a la venta de productos como en el caso concreto de la venta de computadoras y partes, tomando ejemplos prácticos de entre la gran cantidad de artículos que sector mercantil representa.

Por lo que podemos entender de manera global que se trata de un Portal en donde el tratamiento de clientes es fundamental, ya que son ellos quienes mantienen con vida a la empresa a través de la compra de los productos esta provee. Siendo así el cliente debe mantenerse satisfecho al momento de encontrarse con un sitio que le brinde un buen servicio pero sobre todo de manera rápida, es allí donde radica la importancia del e-commerce para la empresa, la velocidad con la que el cliente es atendido, es decir, en términos de Benchmark, el tiempo de respuesta de cada petición Web que el cliente realice.

La concurrencia de clientes del Portal Web es otro factor importante para tomar en cuenta cuando existe en un ambiente real de trabajo, en empresas de gran éxito miles y miles de clientes acceden a las páginas Web de las empresas para consultar o realizar compras lo que genera gran concurrencia de clientes accediendo a todas las opciones que la empresa maneja, se establece un ambiente en donde la carga del sistema crece y se pone a prueba la capacidad de los Servidores de Aplicaciones tanto a nivel de Hardware, como de la gestión propia en el tratamiento de tareas y procesos que implica atender a los clientes.

El benchmark debe considerar principalmente estos factores ya que si se necesita elegir entre dos Servidores de Aplicaciones como JBOSS o GLASSFISH para la gestión de una aplicación o Portal Web, esa elección debe basarse en una comparación real con datos que una empresa pueda brindar para satisfacer de manera eficiente a sus clientes, esto conlleva a la simulación de este ambiente, lo que se logra sobrecargando a los Servidores a través de procesos que generen clientes concurrentes realizando tareas simultáneamente las cuales son atendidas de manera individual bajo la responsabilidad del Servidor de Aplicaciones.

Para el desarrollo en sí del Sistema que permitirá evaluar a los Servidores de Aplicaciones necesitamos guiarnos en el conjunto de especificaciones que nos da la metodología mencionada anteriormente, generando una gran cantidad de información a partir del análisis de los requerimientos para esta aplicación, con la ayuda de diagramas y modelos, por ejemplo de los casos de uso, podremos establecer el camino que se dirige a la

obtención de un software que cumpla las expectativas y objetivos iniciales para poder realizar las pruebas de benchmark.

Para poner en marcha el Software, es necesario de una planificación de las pruebas que se realizará a los servidores, en donde se prevean los procedimientos que se va a tomar en cuenta, los parámetros y lineamientos en los cuales las pruebas son realizadas, entre los cuales encontramos usuarios del sistema, tiempo de realización de las pruebas, tareas del usuario, registros en la base de datos. Estos parámetros están planificados según índices de medición de rendimiento. Los índices de medición del rendimiento nos indican el comportamiento del Servidor de Aplicaciones al gestionar una aplicación Web, por lo que cuando las pruebas se realizan el benchmark toma muestras de cada uno de estos índices y se los evalúa de acuerdo con las capacidades y características propias del Hardware. Cpu, memoria y red por separado nos dan una idea clara en todos los ámbitos de la gestión que el Servidor está realizando con la aplicación.

La medición de uso del CPU es un índice fundamental que nos indica la Gestión de los procesos que el procesador ejecuta. La Memoria nos indica la Gestión de valores y códigos que la aplicación requiere almacenar para ejecutar dichos procesos y la red, el flujo de información con respecto a los usuarios y a la base de datos.

El sistema operativo juega un papel importante, ya que si bien es cierto la aplicación se ejecuta en una Máquina Virtual de Java, el Sistema Operativo se encuentra al mismo tiempo gestionando otras aplicaciones y por tanto dando uso de los elementos CPU, memoria y tarjetas de red, por lo que es necesario considerarlo al momento de sacar las conclusiones de las mediciones.

Los datos obtenidos de las mediciones son tratados, analizados, calculados y finalmente comparados. Para cada uno de los índices que necesitamos evaluar la toma de las muestras se debe realizarlo independientemente, pero realizándolo en un mismo ambiente de pruebas. Las muestras de los índices CPU, memoria y red, son tomadas en un tiempo planificado limitado por el medidor o gestor de mediciones en el momento de preveer los datos para las pruebas. Para el tiempo de respuesta en cambio no se lo puede medir, debido a que es necesario que toda la información de la petición realizada

finalmente haya llegado al usuario y este pueda continuar con su navegación. Es este el tiempo que debe ser medido por este índice.

Otro de los factores que se debe planificar son las tareas que debe realizar el usuario cliente del Portal Web cuando se están realizando las mediciones, debido igualmente a la generación correcta del ambiente de trabajo a evaluar, estas tareas están acorde a los objetivos tanto de usuarios como de la empresa en un ambiente real, registrarse como un usuario para poder realizar una compra de productos. Tomando en cuenta estas premisas los momentos más importantes a evaluar justamente son cuando se realizan estas dos tareas, debido a lo cual se toman muestras en un intervalo donde la mayoría de los casos puedan finalizar completamente estas tareas.

En la ejecución de un Portal Web es difícil saber el número exacto de clientes que interactúan con el sistema o que acciones realizarán, por lo que también es necesario establecer varios ambientes en donde interactúen varios usuarios de tal manera que genere un trabajo o carga mayor para el Servidor de Aplicaciones.

Finalmente una vez evaluadas las muestras es fundamental para conseguir los objetivos propuestos para el presente trabajo comparar cada uno de los resultados en cuanto a rendimientos y a promedios en porcentajes, lo que nos dará de manera más clara y objetiva la respuesta de cual de los Servidores brinda mejores beneficios para los administradores de las empresas que cuentan con este tipo de e-commerce y finalmente poder concluir y recomendar la solución más adecuada.

CAPÍTULO 1. PLANTEAMIENTO DEL PROBLEMA.

1.1 DESCRIPCIÓN DEL PROBLEMA.

En la actualidad una empresa que realiza varias operaciones y tareas a nivel informático requiere que estos sean ejecutados al menor tiempo posible y con una carga baja de consumo de recursos.

La gestión de las operaciones de una aplicación informática debe consistir en contar con una planeación adecuada en el desarrollo de dichas operaciones con el fin de obtener el mejor rendimiento del procesamiento, alcanzar los objetivos en tiempos de respuesta, facilidad de uso, complejidad de transacciones, recurrencia masiva en las funcionalidades de la aplicación, bajo costo y satisfacer cualquier otro objetivo de dichas operaciones.

La selección de las herramientas de desarrollo y aplicación es de los principales factores que determinan la eficacia en que los objetivos de las aplicaciones se cumplan y minimizar los riesgos en que se ven sometidos los usuarios de dichas aplicaciones. Por este motivo las decisiones de que instrumentos, así como los servidores de aplicaciones se deba destinar para la gestión de las operaciones, merecen especial atención por parte de los desarrolladores y jefes de proyectos.

Para poder establecer la eficacia en la funcionalidad de los servidores de aplicaciones, debemos tomar en cuenta los diferentes servicios que la empresa brinda, como por ejemplo correo electrónico, base de datos, manejo de archivos, etc.

Debido a que el campo de los servidores de aplicaciones es muy amplio y la creciente demanda del uso de aplicaciones desarrolladas bajo plataforma J2EE comprenden un campo de estudio importante dentro del sector empresarial en su área de desarrollo, da la pauta para tomar como referencia servidores de tecnología más reciente como son GLASSFISH Y JBOSS.

La selección entre los servidores GLASSFISH y JBOSS para gestionar las operaciones de una aplicación según sus características de desarrollo es el punto de partida para una buena administración en el desenvolvimiento de una aplicación en un entorno empresarial. Por lo cual es necesario de un método o herramienta que permita interpretar en base a la medición de parámetros cual de estos servidores elegir para dicha gestión.

El Benchmark como solución para la obtención de los resultados y su análisis representan una alternativa confiable al momento de determinar que servidor gestiona las operaciones de una aplicación de manera más eficiente según las características de la aplicación que se desarrolle.

1.2 ANÁLISIS COMPARATIVO DE CARACTERÍSTICAS ENTRE JBOSS Y GLASSFISH BAJO ARQUITECTURA J2EE.

Un servidor de aplicaciones o (servidor APP) es una herramienta basada en componentes que residen en la capa intermedia de las tres capas de la arquitectura, que permite el construir, desplegar y administrar nuevas aplicaciones a los usuarios, de una forma más ordenada debido a que contienen la lógica propia del negocio, separando la presentación de la operación, lo que hace más flexible la construcción de aplicaciones, y escalable ya que la lógica de negocio, se encuentra en un solo lugar (servidor) y es la misma para todas las presentaciones. ^{[6], [14]}

Los servidores de aplicación son la parte integral de arquitectura de tres capas lo que permite que estos se sincronicen con el servidor de Web para procesar la petición hecha por el cliente. Dichos clientes pueden ser una variedad de dispositivos (computadores personales o portátiles, PDAs, celulares, etc.), pero la concordancia es que ellos soporten protocolos y tecnologías orientadas a Web, la comunicación entre el servidor de aplicaciones y el dispositivo se lo puede realizar ya sea utilizando un servidor de Web que a su vez se comunica con el servidor de aplicaciones, sea este el caso el dispositivo del usuario final se apoya en base a los varios protocolos que soportan los servidores Web y navegadores Web, y en otros casos la conexión puede ser directa entre el servidor y el dispositivo dependiendo sobre cual de las tecnologías son apoyadas por el servidor de aplicaciones, estos dispositivos podrían estar ejecutando Java Applets o aplicaciones, ActiveX, programas que se comunican usando el protocolo basado en Corba o programas que utilizan un protocolo propietario sobre TCP/IP. ^{[6], [18]}

^[6] LINDGREN, M. Lisa. Application Server for E-Business

^[14] BARRY & Associates, Application Server Definition.

^[18] JAVA BOUTIQUE. Explore the World of Application Servers

Lo mencionado anteriormente da la pauta necesaria para que al momento de seleccionar que tecnología es la que brinde más prestaciones al poder establecer una comunicación entre el servidor de aplicaciones y el dispositivo (usuario final), tienda más al uso de desarrollo de aplicaciones empresariales multicapa basadas en J2EE.

“La plataforma J2EE es un conjunto de tecnologías Java que al usarlas en conjunto proveen de herramientas para el desarrollo y ejecución de aplicaciones de Red IP, además incluye estándares para el middleware (EJB, JMS), conectividad a base de datos (JDBC), transacciones (JTA/JTS), presentación (Servlets y Java Server Pages), y servicios de directorios (JNDI).”^[24]

Con dichos argumentos mencionados existen en el mercado varios servidores de aplicaciones que siguen la tendencia de esta plataforma, pero nos especificaremos únicamente en dos de estos para su estudio los cuales son GLASSFISH. y JBOSS.

1.2.1 ¿QUÉ SON GLASSFISH Y JBOSS?

GLASSFISH y JBOSS son servidores de aplicaciones desarrollados bajo el lenguaje JAVA, y pueden correr en plataforma J2EE, ofrecen fuente abierta para los usuarios y desarrolladores, y se distribuyen de manera libre, se mantienen en la misma línea de administración de aplicaciones Web. Estos servidores proveen contenedores para los EJB, y dan la infraestructura para las extensiones de administración (JMX). A demás componentes esenciales para la mensajería en JMS, transacciones JTA/JST, persistencia CMP, seguridad JAAS, conectividad JCA. A demás presentan características como modularidad, acoplabilidad.^[23] ^[36]

^[24] LUNA, Octavio. JBOSS Servidor de Aplicaciones de fuente abierta.

^[23] LUI, Jenny; University Of Sydney. An Analysis of JBoss Architecture.

^[36] TOMICH, Jill, Refcardz. Glassfish Application Server Refcard.

El soporte para los componentes WEB, tal como Servlets y JSPs se provee por medio de una capa de integración. Las Implementaciones del servicio de integración son provistas por terceros como Tomcat y Jetty. ^[27]

Ambos servidores de aplicaciones permiten que objetos java de bajo nivel tengan acceso a soporte transaccional y de clustering típicamente reservado para EJB´s con el fin de que se beneficien de alguna de las características de escalabilidad de estos servidores de aplicaciones J2EE. ^[37], ^[36]

1.2.1.1 Arquitectura JBoss

El diseño del servidor de aplicaciones JBOSS utiliza componentes basados en plug –ins brindándole a este, características de modularidad (“conectar” o ”desconectar” componentes de acuerdo a la necesidad). JBOSS utiliza como su núcleo de arquitectura el JMX (Gestión de Extensiones de Java) para poder gestionar sus componentes y también posee la característica de despliegue extensible lo que permite adicionarle componentes al JMX. ^[4]

Como punto de partida para poder entender la arquitectura JBOSS, se empezará con la introducción a JMX y los distintos módulos que interactúan con este.

1.2.1.1.1 JMX

JMX es un Framework que representa la “columna vertebral” que permite gestionar y monitorizar los componentes de JBOSS (módulos, contenedores, y plug-ins) mediante un conjunto de APIs. ^[8]

Mediante JMX se puede gestionar recursos de tecnología introduciendo un modelo de MBeans, los ManagedBeans (Gestión de Beans) son los objetos JMX que exponen la

^[27] PÉREZ, Mariñán Martín. Arquitectura empresarial y Software Libre.

^[37] UNISYS. JBoss Application Server Migration Strategies.

^[36] TOMICH, Jill, Refcardz. Glassfish Application Server Refcard.

^[4] FLEURY, Mark. JBoss 4.0 The Official Guide.

^[8] PERRY, J Steven. Java Managment Extensions

información manejable en forma de propiedades y operaciones necesarias para un determinado recurso.^[26]

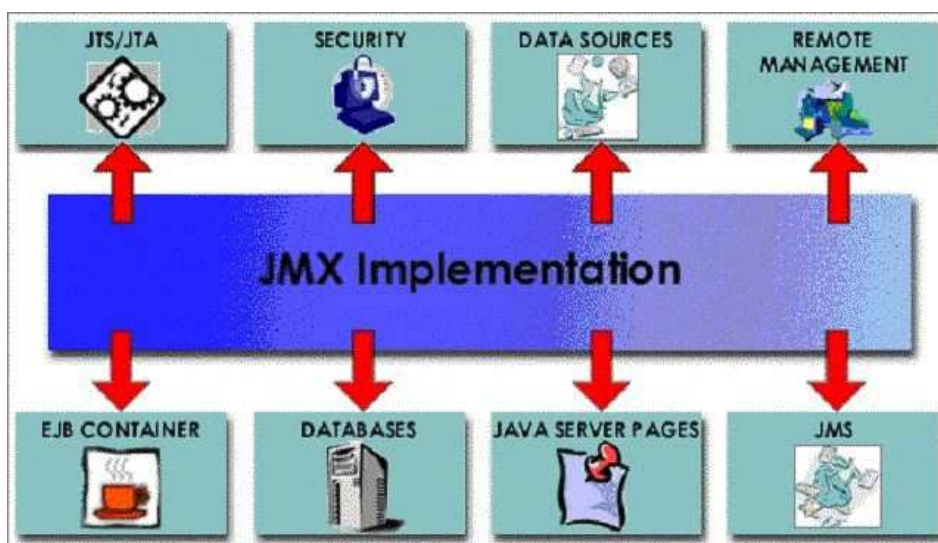


Figura 1. Papel de integración de JMX como una columna vertebral, hacia componentes (módulos, contenedores y plug-ins).^[20]

La arquitectura JMX está conformada por tres capas o niveles las cuales son:

Nivel de Instrumentación: Este nivel o capa incluye todos los recursos o componentes que facilitan la información necesaria para la gestión de aplicaciones, en este se definen los requerimientos para implementar un recurso gestionable (aplicaciones, componentes de los servicios, dispositivos, etc.) mediante JMX.

En este nivel se presentan cuatro tipos de MBeans:

- MBeans Estandar: Es un java bean simple y definido estáticamente (son los más comunes dentro de JBOSS).
- MBeans Dinámicos: Estos exponen su interfaz en tiempo de ejecución.
- MBeans Abiertos: Estos son una extensión de los anteriores.
- MBeans de Modelo: Este tipo también son una extensión de los MBean Dinámicos, simplifican la instrumentación de los recursos dándole a estos un comportamiento por defecto. Los XMBeans de JBOSS son un ejemplo de este tipo de MBeans^[8]

^[26] PEREIRA, Santiago. Instrumentación de componentes Java usando JMX.

^[20] JBOSS.org. The JBoss 4 Application Server Guide.

Nivel de Agente: En este nivel se proveen los requerimientos para implementar un agente. Un agente es aquel en donde es responsable de controlar y hacer disponibles los recursos manejados en el nivel de instrumentación, como también de administrar las relaciones entre ellos.

Un componente esencial en este nivel es el MBeanServer servidor de MBeans, este MBeanServer es un registro de MBeans que permite ser accesibles a otras aplicaciones. Un MBeanServer ofrece un servicio de consulta para los MBeans, tras una consulta retorna los nombres de los MBeans. Debido a que sólo se devuelvan los nombres, todas las operaciones en todos los MBeans deben pasar por el MBeanServer.^[8]

Nivel de Servicios Distribuidos: Nivel destinado a permitir una gestión cooperativa de redes de agentes y de sus recursos, en otras palabras ayuda a que las aplicaciones de administración interactúan con los agentes y sus objetos gestionados a través de adaptadores. En general, existe al menos un adaptador específico para cada protocolo de manejo o tecnología requerida para apoyar los diferentes sistemas de gestión. Permitiendo a estos componentes ampliarse para una completa aplicación de gestión.

^[8] PERRY, J. Steven. Java Management Extensions.

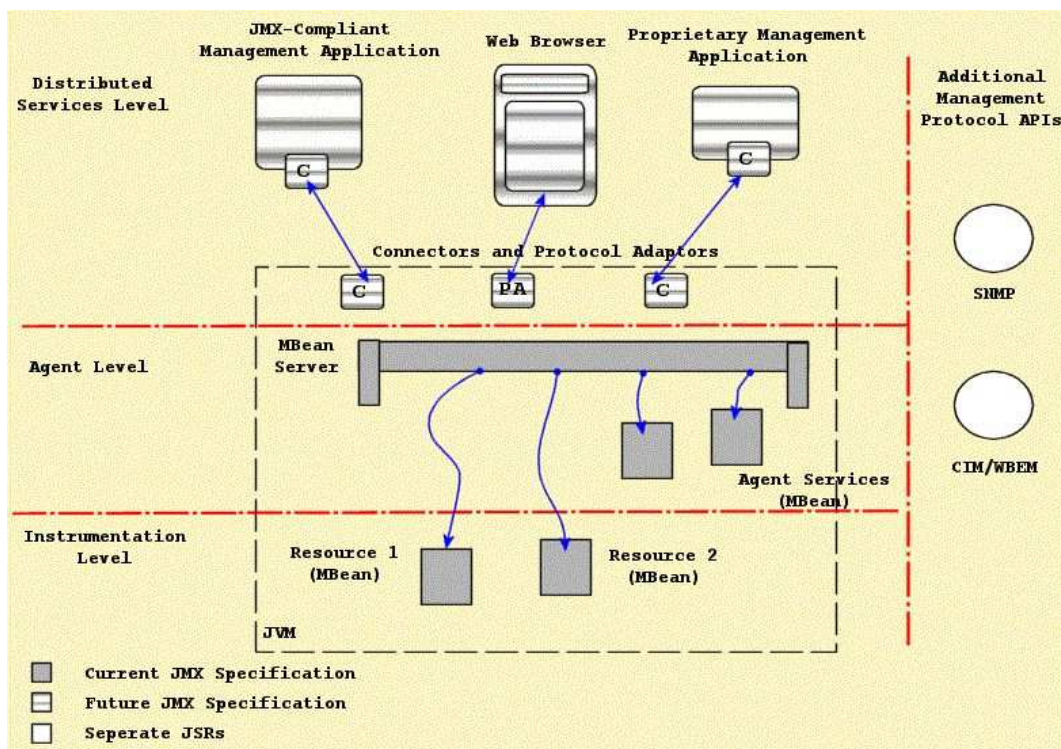


Figura 2 Interacción entre los distintos niveles de la arquitectura JMX ^[20]

1.2.1.1.2 Servicios Proporcionados Por JBoss

Como parte integral de la arquitectura, JBOSS cuenta con servicios que permiten la precarga de clases facilitando la comunicación entre unidades de desarrollo y servicios de la aplicación. Estos servicios son detallados a continuación. ^[29]

1.2.1.1.2.1 Componente JBoss NS

El componente (servicio de nombres) JBossNS es una implementación de Java Naming Directory Interface (JNDI), el cual proporciona un servicio de nombres que le permite a los usuarios asignar un nombre a un objeto o servicio para facilitar su búsqueda, además proporciona una interfaz común para servicios de directorios existentes tales como DNS, LDAP, Active Directory, RMI registry, COS registry, NIS y sistemas de archivos. El API JNDI está dividido lógicamente en un API cliente que se utiliza para acceder a los

^[20] JBOSS.org. The JBoss 4 Application Server Guide.

^[29] REDHAT. Jboss Enterprise Application Plataforma Configuration Guide.

servicios de nombres, y una interfaz proveedor de servicios (SPI) que permite al usuario crear implementaciones JNDI para el uso de los servicios de nombre.^[19]

1.2.1.1.2.2 Componente JBoss TX

El componente JBossTX está diseñado para usar cualquier gestor de transacciones que implementen el JTA (Java Transacción API). Este servicio garantiza el cumplimiento de las cuatro importantes propiedades de las transacciones las cuales son: atomicidad, consistencia, aislamiento y durabilidad (A.C.I.D.). JBossTX depende externamente del JBossNS debido a que su implementación esta ligada al manejo de nombres proporcionada por JNDI cuando este es inicializado por el servicio de administración del servidor JBOSS.^[19]

1.2.1.1.2.3 Componente Contenedor EJB

El componente contenedor EJB es una arquitectura que hace hincapié en el diseño modular plug-in, en la cual reside toda la lógica del negocio implementados en los EJB (Enterprise Java Beans). Los cuales proporcionan un conjunto de características o servicios como (mensajería, soporte a transacciones distribuidas, acceso a base de datos, servidores de correo, seguridad, concurrencia, etc.) todos estos aspectos clave del contenedor EJB podrán ser implementados y gestionados por versiones personalizadas de un plug-in realizado por un desarrollador. Por ultimo una gran ventaja de manejo de servicios por medio de EJBs es que estos permiten flexibilidad y también ser reutilizados.^[4]

1.2.1.1.2.4 Componente JBoss MQ

El componente JBossMQ se compone de varios subsistemas que trabajan juntos para proporcionar el servicio JMS API a las aplicaciones cliente El JMS API Java Message Service Application Programming Interface, utilizado por las aplicaciones para enviar mensajes a otras aplicaciones de manera asíncrona. En modo asíncrono el emisor del mensaje no tiene por que esperar a que éste llegue a su destino. El emisor emite el mensaje y continua trabajando, el receptor lo recibe y a partir de ahí puede realizar numerosas tareas con el. Para dar una calidad de característica de fiabilidad el JMX utiliza como

^[19] JBOSS.com. Jboss Enterprise Portal Plataforma.

^[4] FLEURY, Mark; STARK, Scott; NORMAN, Richards. JBoss 4.0 The Official Guide.

recurso intermediario al MOM (Messaging Oriented Middleware) quien receipta los mensajes enviados por un emisor y los redirecciona a los destinos adecuados, garantizando que el mensaje llegue al destino aunque no se encuentre disponible el receptor a través de un método de persistencia.^[19]

1.2.1.1.2.5 Componente JBoss CX

El componente JBossCX proporciona la arquitectura para manejar de manera adecuada el API JCA el cual tiene la función de integrar o conectar los EJBs que se ejecutan en el servidor de aplicaciones con sistemas externos de información conocidos como EIS (Enterprise Information System) mediante mecanismos homogéneos que da resultado a ser transparente y portable con respecto al EIS. Dichos EIS proporcionan un adaptador de recursos que sirven como intermediarios para establecer una conexión con el servidor de aplicaciones siempre y cuando soporte JCA esto en base a “contratos” establecidos por la arquitectura JCA que estandarizan la conexión estos tipos de contratos son de Connection Management, Transaction Management y Security, los cuales proporcionan escalabilidad, acceso transaccional y seguridad respectivamente. En forma general los conectores JCA han sido elaborados para resolver el acceso a sistemas legados en una modalidad sincrónica (petición/respuesta), lo cual se refleja en la mayoría de los conectores existentes.^[20]

1.2.1.1.2.6 Componente JBoss SX

El componente JBossSX diseñado para el control de acceso y autorización de usuarios mediante el API JAAS, el cual es un conjunto de paquetes con servicios para autenticar y controlar el acceso de una manera centralizada (en un descriptor.) además la característica primordial que posee JAAS es la inclusión de un dominio de seguridad que su principal utilidad es la de servir como una especie de JMS (Java Message Service). El objetivo principal de JAAS es delegar el mecanismo de autenticación y autorización en el servidor de aplicaciones mediante la asignación de roles a los usuarios de la aplicación J2EE, dichos roles de asignación no deben ser codificados sino mas bien especificados en un archivo XML llamado descriptor de despliegue (deployment descriptor) es así como JAAS

^[19] JBOSS.com. Jboss Enterprise Portal Plataform.

^[20] JBOSS.org. The JBoss 4 Application Server Guide.

maneja el acceso a recursos y servicios de acuerdo al perfil configurado por el propio usuario.^[19]

1.2.1.1.2.7 Componente Contenedor Web

El componente contenedor WEB es una implementación que especifica un entorno de ejecución para componentes Web que son capaces de manejar conexiones HTTP, es decir maneja la ejecución de Servlets y páginas JSP (Java Server Pages) lo que permite que estas tecnologías ayuden a la creación de páginas Web dinámicas es decir, páginas web que se generan en el servidor en base a la lógica de negocio.^[27]

La forma de comunicación entre el contenedor web y el usuario esta basado en un sistema de peticiones y respuestas lo que significa que el usuario inicia siempre una conexión y el servidor retorna una respuesta y cierra la conexión una vez realizada la respuesta, dependiendo de los recursos que el usuario maneje, el contenedor utilizará conectores que sirven de intermediarios entre la comunicación con elementos externos como el uso de otros servicios ya sea para sistema de mensajería o gestión de base de datos, etc.

1.2.1.1.2.8 Deployment

El servicio de DEPLOYMENT apoya los despliegues de los EJB jars, wars y ears. Este vigila las urls para los archivos J2EE y despliega los archivos tal como aparecen o cambian sin la necesidad de reiniciar el servidor JBoss si se encuentra en ejecución.^[23]

1.2.1.1.3 Otros Servicios Proporcionados Por JBoss

El servidor de aplicaciones JBOSS proporciona nuevos servicios que se adaptan a las necesidades empresariales particulares de ciertos usuarios, por lo cual estos paquetes no están incluidos en una instalación “típica” entre estos servicios se detalla a breves rasgos los siguientes.

1.2.1.1.3.1 Hibernate

^[19] JBOSS.com. Jboss Enterprise Portal Plataform.

^[27] PÉREZ, Mariñán Martín, Arquitectura empresarial y Software Libre.

^[23] LUI, Jenny; University Of Sydney. An Analysis of JBoss Architecture.

Es un servicio de persistencia que proporciona una simple, pero poderosa, alternativa a los “beans” estándares. Facilita la creación de clases de persistencia utilizando el lenguaje Java, incluyendo la asociación, herencia, polimorfismo y composición. Hibernate es muy simple, pero cuando se ejecuta en JBoss, puede optar por desplegar su aplicación como archivo Hibernate, llamado archivo HAR.^[4]

1.2.1.1.3.2 Clustering

Para mejorar el funcionamiento de una aplicación si esta abarca una dimensión considerada JBOSS permite el manejo de nodos los cuales conforman un cluster para que dicha aplicación se ejecute en paralelo. Cada nodo es un servidor JBOSS físico lo que garantiza la recuperación de la aplicación si uno de los nodos falla.^[20]

1.2.1.1.3.3 JGroups

Proporciona el servicio peer-to-peer para que se puedan llevar a cabo comunicaciones entre nodos que formen parte de un mismo cluster.^[21]

1.2.1.1.3.4 JBoss Cache

Es un servicio Diseñado para almacenar en caché los objetos Java a los que se accede más frecuentemente de manera que se aumente el rendimiento de las aplicaciones. Eliminando accesos innecesarios a la base de datos, reduce el tráfico de red e incrementa la escalabilidad de las aplicaciones.^[34]

1.2.1.1.3.5 JBoss Portal

Es una Plataforma de código abierto para albergar y servir una interfaz de portales Web, publicando y gestionando el contenido así como adaptando el aspecto de la presentación.
[19]

1.2.1.1.3.6 JBoss AOP

^[4] FLEURY, Mark; JBoss 4.0 The Official Guide.

^[20] JBOSS.org. The JBoss 4 Application Server Guide.

^[21] JGROUPS. A Toolkit for Reliable Multicast Communication.

^[34] SURTANI, Manik. Ban Bela. JBOSS.org. JBoss Cache User Guide.

^[19] JBOSS.com. Jboss Enterprise Portal Platform.

Este servicio AOP (Aspect-Oriented Programming) es una innovación relevante en JBOSS ya que soporta el desarrollo orientado a aspectos, el **aspecto** es un comportamiento común en el que se identifica claramente métodos, clases y objetos, por lo que permite mayor modularidad que en la programación orientada a objetos y la reutilización de código para un mejor rendimiento en la aplicación. ^[4]

1.2.1.2 Arquitectura Glassfish

La arquitectura de GLASSFISH V2 o Sun Java System Application Server 9.1, está basado en una estructura de clusters flexibles, además brinda el soporte para gestión remota multi-máquina y multi-dominio segura, lo cual pretende mejorar el grado de disponibilidad y la escalabilidad en el rendimiento del servidor de aplicaciones para sistemas de alto volumen y Servicios Web. ^[38]

1.2.1.2.1 Componentes De La Arquitectura Glassfish

Varios elementos conforman la infraestructura en la arquitectura de GLASSFISH, elementos que se organizan y configuran permitiendo el trabajo en paralelo de varias instancias de servidor, las cuales están agrupadas en los distintos clusters, que a su vez pertenecen a nodos que en este caso representan un host distinto del servidor, todos estos elementos están gestionados por un Dominio (DAS) que ofrece un buen grado de servicio administrativo, cuyos conceptos se detallan a continuación.

^[4] FLEURY, Mark. JBoss 4.0 The Official Guide.

^[38] WHITE, Larry, Kumar, Abhijit, SUN Microsystems. Clústeres de la version 2 de Glassfish.

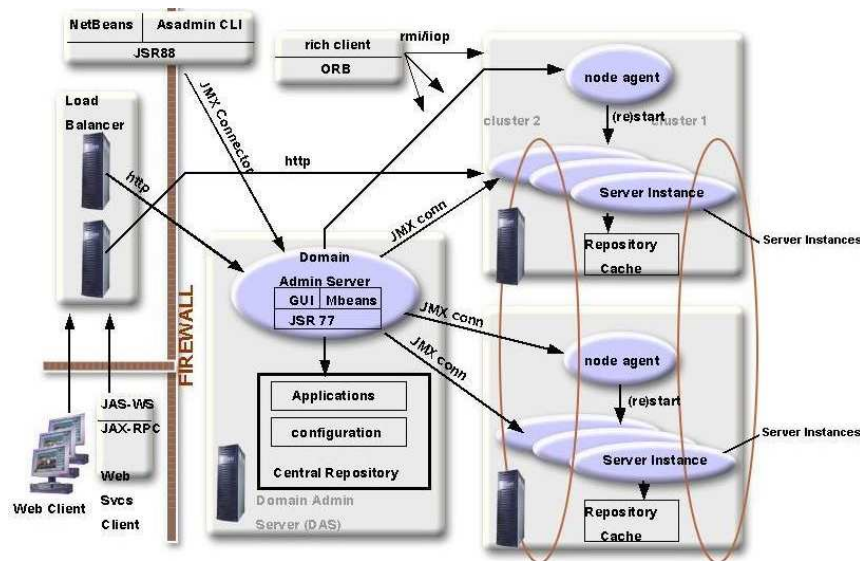


Figura 3 Muestra la arquitectura del servidor de aplicaciones GLASSFISH. ^[16]

1.2.1.2.1.1 Instancia

Una instancia de servidor de aplicaciones es un motor del servidor de aplicaciones JavaEE donde se alojan las aplicaciones Java ejecutándose sobre una única Java Virtual Machine (JVM). Cada instancia funcionará como un servidor JMX MBean, es decir cuenta con los recursos administrativos para todos los servicios que brinde el servidor, puede existir dos tipos de instancias, las instancias que pertenecen a un cluster y las instancias independientes; las instancias que pertenecen a un cluster son homogéneas recibiendo los mismos recursos y configuración de su cluster padre en cambio la instancia independiente tiene su propio set de aplicaciones, recursos y configuración. ^{[38] [31]}

1.2.1.2.1.2 Cluster

Un cluster es una entidad virtualizada que pertenece exactamente a un solo dominio y que contiene un grupo de instancias homogéneas trabajando en conjunto, es decir trabajan con las mismas aplicaciones, recursos e información de configuración. Un cluster es una importante entidad de administración ofreciendo a los usuarios un servicio transparente en el manejo de las instancias y sus recursos, por ejemplo una aplicación puede ser desplegada

^[16] GLASSFISH WIKI. Glassfish V2 Architecture.

^[38] WHITE, Larry, Kumar, Abhijit, SUN Microsystems. Clústeres de la version 2 de Glassfish.

^[31] SUN Microsystems. Application Server Commands and Concepts.

en un cluster en una operación agregada sin tener la necesidad de desplegar la aplicación en cada instancia perteneciente al cluster, el usuario no necesita estar consciente de cuantas instancias provean el servicio. Además un cluster facilita el equilibrio de la carga al ser escalable horizontalmente ya que se pueden agregar instancias adicionales de servidor.^[22]

1.2.1.2.1.3 Dominio

Un dominio es un punto unificado de la administración para cualquier número de clusters o instancias del servidor así como sus configuraciones y aplicaciones que son administradas en conjunto. Este actúa como un único punto de autenticación administrativa y autorización, además en una instalación del servidor pueden agregarse múltiples dominios, actuando como mínimo un dominio que trabaje en una máquina, cada dominio puede definirse como un ente independientemente manejado, restringiendo a un desarrollador la administración de cada instancia de dominio. Por último cada dominio cuenta con un Servidor de Administración de Dominio (DAS) que se emplea para administrar instancias del servidor de aplicaciones en el dominio.^[16]

1.2.1.2.1.4 DAS

El Servidor de Administración de Dominio DAS (Domain Administrative Server) es una instancia del servidor de aplicaciones que administra la configuración del dominio y alberga dicha configuración en un Repositorio (ficheros en el File System). El DAS utiliza una consola para la administración del deposito central que es una interfaz basada en explorador.

Las operaciones administrativas que el DAS configura y los mantiene en el Repositorio son:

1. Configuraciones de:
 - Dominio
 - Instancia y su JVM
 - Cluster y sus instancias
2. Aplicaciones desplegadas sobre cada instancia

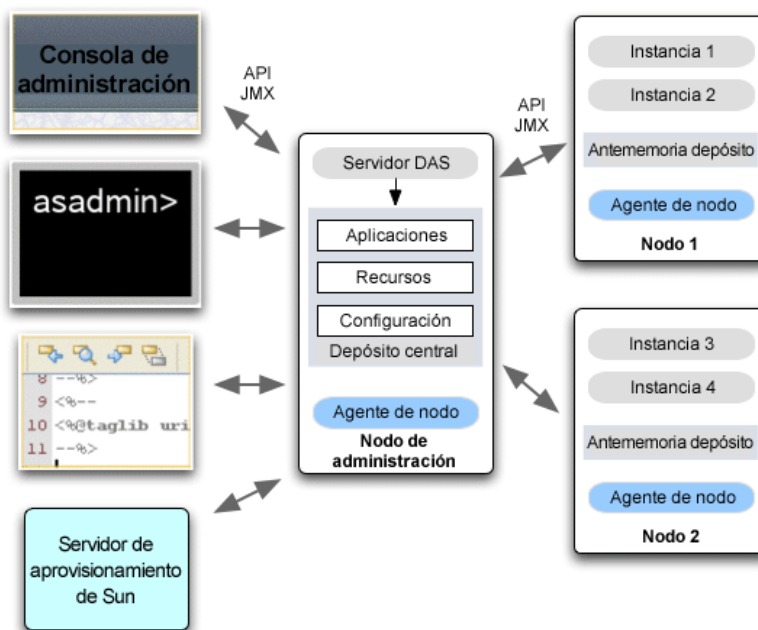
^[22] KALALI, Masoud; GlassFish Application Server.

^[16] GLASSFISH WIKI. Glassfish V2 Architecture.

3. Recursos (JDBC, MAIL, etc.) disponibles a cada instancia del dominio, previa asignación del administrador.

Todos los cambios de la configuración almacenada en el repositorio es replicada en las cache de configuración local de las instancias administradas por el dominio, es decir el DAS será responsable de todas las operaciones de escritura en el repositorio lo que permite una gestión centralizada de múltiples instancias ubicadas en diferentes servidores.

Otra característica a tomar en cuenta por el DAS es que la infraestructura de administración se basa en tecnología JMX (Java Management Extensions) por lo que utiliza la información de administración en forma de MBeans. Los usuarios que tiene acceso al DAS deben tener un perfil administrativo, ya sea el propio administrador del servidor de aplicaciones así como todos los clientes administrados que utilizan la API JMX. Cuando el DAS no esta en ejecución no es posible procesar operaciones administrativas, sin embargo las instancias y los clusters en el dominio pueden funcionar normalmente con su repositorio cache y su nodo agente, es decir el DAS no tiene características de alta disponibilidad.^{[5] [31]}



^[5] HEFFELFINGER, David. Java EE 5 Development using GlassFish Application Server.

^[31] SUN Microsystems. Application Server Commands and Concepts.

Figura 4. Arquitectura administrativa del DAS.^[38]

1.2.1.2.1.5 Nodo Agente

El nodo agente es un proceso ligero que facilita la gestión del ciclo de vida remoto de las instancias, se ejecuta en cada Host físico o nodo en el cual funciona por lo menos una instancia del servidor. El propósito del nodo agente será administrar características de las instancias como: arrancar, parar, crear y reiniciar instancias caídas, así como lectura de logs y especialmente sincronizar la configuración de todo el dominio en conexión con el DAS, por último el nodo agente corre sobre una JVM recibiendo solamente una API JMX en tiempo de ejecución.,^{[12][32]}

1.2.1.2.1.6 Balanceador De Carga

Es un subsistema que distribuye los requerimientos de varios servidores de aplicaciones y servidores Web dentro de las instancias del servidor es decir provee balanceo de carga para clientes HTTP, RMI/IIOP y JMS, revisando sus operaciones si es necesario y manteniendo una conexión con un host particular con el que se ha establecido una sesión.^[3]

1.2.1.2.1.7 Repositorio Central Y Cache Local

Estos son utilizados para almacenar la información escrita por el DAS a cerca de la configuración del dominio y de las aplicaciones desplegadas en las instancias.

El repositorio central contiene toda la información de las instancias administradas bajo el mismo dominio usando un API JMX.

El cache local es un repositorio individual para cada instancia del servidor permite mantener la información de cada aplicación disponible sin utilizar el DAS dando una

^[38] WHITE, Larry, Kumar, Abhijit, SUN Microsystems. Clústeres de la version 2 de Glassfish.

^[12] SUN Microsystems; Sun Java Enterprise System 5 Monitoring Guide

^[32] SUN Microsystems; Configuring Node Agents

^[3] CID, Jaime; Servidores de Aplicación Arquitectura y planificación Análisis de Mercado

mayor velocidad al arranque y despliegue de las aplicaciones, el cache esta sincronizado con el repositorio central y se restaura cuando se reinicie la instancia. ^[32]

1.2.1.2.1.8 Tipos De Usuarios

Para efectos de administración el servidor de aplicaciones GLASSFISH ha dividido los clientes en tres tipos.

Usuarios Administrativos: Incluyen la utilidad **asadmin** CLI (Command Line Interface) y se comunican exclusivamente con el DAS vía JMX en Beans.

Usuarios Web: Se comunican con las instancias del servidor utilizando el protocolo HTTP a través de un Web browser o una invocación del servicio Web, estos clientes son lo que más utilizan el balanceador de carga

Usuarios RMI/IIOP: Este cliente reside dentro del limite del firewall del servidor se comunican con las instancias del servidor utilizando el protocolo RMI/IIOP, tiene acceso a todos los recursos JNDI definidos en el servidor, por lo tanto tiene características de administrador para ciertas actividades de configuración. ^{[16] [33]}

1.2.2 ANALISIS COMPARATIVO DE CARACTERISTICAS DE JBOSS Y GLASSFISH

Una vez descritas las arquitecturas que forman la estructura de cada uno de los servidores de aplicaciones GLASSFISH y JBOSS se procederá a realizar un cuadro comparativo en la cual se tratará de mostrar las diferentes características que cada uno de estos servidores poseen, las cuales son tomadas de las referencias bibliográficas y documentos que son propias de las empresas que las desarrollan

^[32] SUN Microsystems; Configuring Node Agents

^[16] GLASSFISH WIKI. Glassfish V2 Architecture.

^[33] SUN Microsystems; Usage Profiles

	JBOSS	GLASSFISH	COMENTARIO
TIPO DE SERVIDOR	Servidor de Aplicaciones	Servidor de Aplicaciones	
VERSIÓN	4.0	Sun Java System Application Server 9.1 (Glassfish V2).	Actualmente Sun esta desarrollando una versión 3 de GLASSFISH, en el caso de JBOSS RedHat está desarrollando una versión 5.
PROVEEDOR	Red hat	Sun Microsystems	
DISTRIBUCIÓN	Libre	Libre	
OPEN SOURCE	✓	✓	Contienen la fuente de software la cual es configurable (a medida).
DESCRIPCIÓN	Implementa y ejecuta aplicaciones J2EE y tiene licencia GPL/LGPL.	Implementa y ejecuta aplicaciones J2EE y tiene licencia CDDL GNUGPL.	
ADMINISTRACIÓN			
CONFIGURACIÓN GUI	X	X	Ambos servidores poseen GUIs

			estáticos.
ADMINISTRACIÓN REMOTA	X	✓	GLASSFISH implementa varios tipos de usuarios uno de ellos el cliente administrador.
PROTOCOLO SNMP CONFIGURABLE / MONITOREABLE	X	X	El protocolo SNMP tiene funcionalidad externa respecto a los servidores de aplicaciones.
ESCALABILIDAD			
COMPATIBLE .NET	X	X	No es compatible ya que ambos se basan en arquitectura J2EE.
PUERTO 64-BIT	✓	✓	Dependiendo de la extensión de la aplicación se pueden ocupar o no los 64 bits de mapeo de la JVM.
SOPORTE CLUSTER	✓	✓	Permiten que sea más tolerante a fallos y mantenga alta disponibilidad.
SOPORTE IPV6	X	X	No esta establecido ya que el protocolo básico de

			comunicación se mantiene en IPV4.
COMPATIBLE J2EE 1.4	✓	✓	Basados en la arquitectura J2EE 1.4 o superior.
OTRAS CARACTERISITICAS			
MULTIPLES REGISTROS	✓	✓	Permiten registrar eventos de la aplicación.
SERVIDORES VIRTUALES	✓	✓	Ofrece la capacidad de mostrar al usuario una instancia del servidor que simula al servidor de aplicaciones.
CAPACIDAD JSP 2.1 Y SERVLET 2.5	✓	✓	Ambas utilizan para el desarrollo de aplicaciones Web las últimas versiones de JSP y Servlets.
CAPACIDAD EJB 3.0	✓	✓	Ambos utilizan la ultima versión del contenedor para el manejo de lo Enterprise Java Beans.
JAX-WS / JAX-B 2.x	✓	✓	Es la más actualizada arquitectura de

			manejo de código que enlaza XML con componentes estándar J2EE.
SEGURIDAD			
ADAPTABILIDAD DE FIREWALL	X	✓	Dentro de la arquitectura de GLASSFISH está incluido el manejo de firewall para el ingreso de usuarios.
AUTENTICACIÓN LDAP	✓	✓	Ambos poseen métodos de autenticación para brindar seguridad en la administración utilizando el servicio de nombres propios de cada servidor.
(SOFTWARE) SSL	✓	✓	Utilizan SSL para el transporte seguro de datos a través de la encriptación de los mismos garantizando una conexión segura.
SOPORTE			
SOPORTE COMERCIAL DISPONIBLE	✓	✓	Ambos cuentan con los mejores servicios contratados para

			soporte disponible así como soporte gratuito basado en foros.
SOPORTE A LISTA DE CORREO	✓	✓	De acuerdo a especificaciones de soporte solicitados por los usuarios estos son organizados mediante listas de correos que solventan problemas eventuales.
SOPORTE JAVA SERVER FACES 1.2	✓	✓	Para manejar de una forma estandarizada el desarrollo de interfaces con el uso de JSP, ambos utilizan el Framework JSF que administra y ayuda a realizarlas.
SOPORTE PLUG-IN PERSONALIZADO	✓	✓	Tanto JBOSS como GLASSFISH utilizan la característica de plugin para brindar servicios extras de acuerdo al ambiente de prestaciones que

			la aplicación desarrollada brinde a los usuarios.
SOPORTE HIBERNATE 3.X	✓	✓	Otra característica soportada para la seguridad de la lógica del negocio es implementada por estos servidores con el uso de Hibernate como alternativa de persistencia.
SOPORTE JBOSS SEAM	✓	X	Por medio de este Framework ambos servidores están en la capacidad de unir varias tecnologías y estándares que brinden nuevas y distintas funcionalidades para desarrollar una aplicación Web con características que brinden en otros Frameworks como JSF y EJB.
SOPORTE DE CONEXIÓN ECLIPSE IDE	✓	✓	A través de plugins los servidores se pueden conectar al IDE ECLIPSE lo

			que facilita su configuración por medio de una interfaz brindando a algunos desarrolladores un manejo mas fácil lo que con lleva el omitir el uso de líneas de comando para ejecución de tareas.
MÁS DE UNA INSTANCIA POR NODO	X	✓	JBOSS permite la creación de más instancias pero únicamente una instancia por nodo, por lo que son únicamente instancias físicas, mientras GLASSFISH permite a su vez creación de más instancias lógicas dentro de un mismo nodo

Tabla 1. Comparación de características básicas de los Servidores de Aplicación GLASSFISH y JBOSS^[30]

^[30] SERVER Watch; Server Compare;

Una vez contempladas las características técnicas entre los servidores de aplicaciones GLASSFISH y JBOSS se puede llegar a determinar de forma general que ambos servidores están diseñados para brindar al desarrollador similares servicios como los servicios de nombres, mensajería, seguridad, manejo de EJBs, etc. , sin embargo la estructura de la arquitectura de cada uno de estos difiere conceptualmente en la gestión de esos servicios y de usuarios permitiendo finalmente que GLASSFISH sea un servidor de aplicaciones veloz y que JBOSS sea un servidor de aplicaciones más robusto.

Cabe recalcar que la versión considerada para el estudio de GLASSFISH maneja la tecnología J2EE y además la tecnología JEE5 que es una versión evolutiva de J2EE. La versión considerada por JBOSS para su estudio, en cambio esta basada completamente en la tecnología J2EE.

1.3 SELECCIÓN DE BENCHMARK ESTANDAR APLICADO A CASO DE ESTUDIO

Después de haber considerado las características y servicios que brindan los servidores de aplicaciones GLASSFISH y JBOSS se debe tomar en cuenta que el rendimiento es clave en la comparación de los servidores de aplicaciones. Por lo tanto la implementación de un benchmark que mida el desempeño de los servidores de aplicaciones cuando una aplicación se encuentre en tiempo de ejecución nos dará una clara visión de cual de estos servidores de aplicaciones se adapten mejor a los requerimientos de los desarrolladores de aplicaciones.

1.3.1 BENCHMARK

La referenciación conocida como “Benchmark”, llevado al campo de los sistemas computacionales, es un proceso usado para comparar diferentes sistemas informáticos o componentes de un sistema basándose en cual tiende a ofrecer el mejor rendimiento o el acceso a más recursos como por ejemplo bases de datos o sistemas de ficheros por parte de los usuarios, eso si bajo un mismo entorno de ejecución.^{[17] [15]}

^[17] INTXAUBURU, Miren Gurutze. ¿Es el Benchmark una herramienta de aprendizaje organizacional?

^[15] CSI.map. Consejo Superior de Administración Electrónica; Recomendaciones Aplicables a la contratación de Equipos Informáticos

Para poder garantizar los resultados obtenidos. Dicho proceso se deben efectuar siguiendo un estándar genérico considerado por Robert C. Camp que contiene las siguientes etapas:

1. Etapa de Planeación.- El objetivo de esta etapa es planear las investigaciones de Benchmarking respondiendo a cuestiones como son: quien, que y como.
2. Etapa de Elaboración de Benchmark.- Ejecutada la primera etapa se lleva a cabo la realización del sistema que nos permita evaluar las herramientas que vamos a medir.
3. Etapa de Medición de Parámetros y Generación de Resultados.- Esta etapa requiere de los datos obtenidos de la implementación del sistema Benchmark desarrollado en la segunda etapa, los cuales son organizados según poscriterios de la planeación.
4. Etapa de Análisis de Resultados.- Esta etapa comprende la interpretación de los datos organizados los cuales determinan las brechas de desempeño de cada herramienta analizada.
5. Etapa de Conclusiones y Recomendaciones.- Una vez analizados los datos se tiene una clara idea del desempeño de las herramientas, lo cual permite recomendar una elección adecuada y fundamentada entre ellas.^[2]

1.3.1.1 Planeación

La fase de planeación consiste en tener una clara comprensión de lo que se va a medir, y bajo qué circunstancias estas mediciones se van a llevar a cabo, así como también especificar de cada una de las etapas del Benchmark que factores se toman en cuenta para lograr los objetivos establecidos de determinar la mejor opción entre los servidores de aplicaciones estudiados.

Para poder evaluar de una manera más real el rendimiento del servidor de aplicaciones es preciso que se realicen múltiples peticiones Web las cuales pongan en funcionamiento los servicios que ofrece dicho servidor, por lo cual es factible realizar una simulación de varias entradas al sistema que debe ser implementado en el software del benchmark, el número de usuarios simulados debe tener un significado relativo al entorno de trabajo, de tal manera

^[2] CAMP, Robert; Benchmarking

que se compare el comportamiento de los servidores analizados bajo las mismas circunstancias, y no pretender medir la capacidad máxima de usuarios que pueden utilizar simultáneamente los servicios.

No podemos elegir al azar los tipos de métricas para establecer el rendimiento de cualquier aplicación, sino que se debe especificar las que proporcionarán la información adecuada para realizar un análisis de rendimiento que englobe el trabajo realizado por el servidor de aplicaciones.

Considerando que el objetivo principal de un servidor de aplicaciones es brindar al usuario de una aplicación características como respuestas rápidas a peticiones en ambiente Web, ejecución de múltiples tareas, administración de recursos y servicios, y cumplimiento de propiedades ACID en transacciones, se ha realizado la elección de las métricas que abarcan el rendimiento de estos servicios basados en tres puntos básicos:

- Tiempo de Respuesta (R)
- Rendimiento (X)
- La Utilización de los Recursos (U) ^[35]

1.3.1.1.1 Tiempo De Respuesta

Es el tiempo transcurrido desde que es realizada una petición Web hasta que la respuesta sea totalmente desplegada., lo que nos indica de forma clara la satisfacción del usuario como un parámetro de rendimiento al recibir en el menor tiempo la respuesta a una o varias peticiones. Se debe tomar en cuenta que el tiempo de respuesta es la medición estrictamente de una petición ya que en una carga real pueden existir varias peticiones al mismo tiempo.^[25]

1.3.1.1.2 Rendimiento

Es el parámetro que mide el trabajo realizado por el servidor de aplicaciones independientemente del tiempo de respuesta, este puede ser medido dependiendo del

^[35] THE World's Leading Java Resource; MICKLEA, Brad. Using Metrics to Optimize J2EE Application Performance in Production;

^[25] Oracle Application server; Oracle Application Server Performance Guide.

parámetro (cpu, memoria y red), lo que nos indica la capacidad que tiene el sistema para manejar la carga correspondiente a recursos, usuarios y transacciones.^[13]

1.3.1.1.3 Utilización De Los Recursos

Este parámetro es vital para entender el rendimiento del servidor de aplicaciones con respecto al uso de recursos críticos tales como cpu, memoria, ancho de banda de red, los cuales afectan el rendimiento del servidor de aplicaciones. Los principales factores que deben analizarse respecto al uso de los recursos son: los porcentajes de uso de los recursos (cpu y red) así como el tiempo que las Peticiones Web utilizan los recursos cuando una aplicación se encuentra en ejecución.^[35]

La medición de estos indicadores, nos permitirá establecer el rendimiento general de un servidor de aplicaciones cuando una aplicación se esté ejecutando y realizando peticiones, simulando una sobrecarga real a través de la concurrencia de usuarios que generan dichas peticiones.

1.3.1.1.4 Parámetros Estadísticos.

Para realizar un conjunto de mediciones debemos conocer ciertos aspectos que son fundamentales al momento de ejecutar dicho proceso las cuales de una manera u otra incidirán al momento de dar una válida interpretación de los datos recogidos.

Después de obtener resultados lo siguiente es diseñar tablas y gráficas en las que vamos a depositar la información de las mediciones en donde se recopilará todos los datos necesarios para el análisis numérico y gráfico que nos llevará a la obtención de resultados. Estos datos los evaluaremos tomando en cuenta conceptos de estadística que nos permitirán emitir un juicio real y concluir con acierto acerca de los objetivos planteados.

^[13] TRANSACTION Processing Performance Council (TPC); TPC Benchmark App

^[35] THE world's leading java resource; MICKLEA, Brad. Using Metrics to Optimize J2EE Application Performance in Production

Puesto que las representaciones gráficas sin un uso de estudio de estadística no siempre consiguen ofrecer una información completa de una serie de datos, es necesario utilizar procedimientos numéricos que permitan resumir toda la información del experimento en unos números llamados parámetros estadísticos. Los parámetros que tendrán más énfasis y se pondrán en aplicación al caso de estudio se detallan a continuación.

a) **Medidas de tendencia central.**- Que representan a toda la distribución. Entre las más importantes y que se presenta una breve descripción son la mediana y la moda.

- **Mediana.**- Es el valor que divide a la población o muestra en dos partes, cada una con igual número de datos, también es considerada un fractil, pues es el valor que divide la probabilidad en dos partes.
- **Moda.**- Es el valor que se presenta más veces, es decir, que tiene la frecuencia mayor, si dos o más valores les corresponde la misma frecuencia mayor, la distribución se llama bimodal o multimodal.

b) **Medidas de dispersión.**- Que indican si los valores están agrupados o dispersos. La más importante y que se presenta una breve descripción es la desviación estándar.

- **Desviación Estándar.**- La raíz cuadrada de la varianza se la denomina desviación estándar o típica que al restar este valor a la media (límite inferior) y sumarlo a la media (límite superior) ayuda a obtener un intervalo y permite asegurar que el 60% de los datos se encuentran dentro de él.

1.3.1.1.5 Tipos De Diagramas

Para la representación gráfica una forma útil de presentar una serie de datos que se repiten en una unidad de tiempo se lo hace por medio de histogramas y polígonos las cuales serán utilizadas para el caso de estudio para una mejor interpretación.

Por ende al momento de interpretar un gráfico lo anteriormente visto brindará ciertos conocimientos útiles sobre como un grupo de datos puede influir en las conclusiones finales y en la interpretación misma.

Para realizar las mediciones, en los capítulos 2, 3 y 4 del presente proyecto, se realizarán detalladamente las etapas del benchmark mencionadas, etapas de elaboración, medición, análisis y conclusiones, que en resumen se trata de la confección del sistema que realice la sobrecarga de los servidores a través de interacciones Web con manejo de transacciones, utilización de múltiples recursos y al mismo tiempo mida cuantitativamente los parámetros

o indicadores, con los cuales se analizará el rendimiento de cada uno de los servidores de aplicaciones como solución a los objetivos planteados.

CAPÍTULO 2. ELABORACIÓN DEL BENCHMARK.

2.1. SELECCIÓN DE LA METODOLOGÍA DE DESARROLLO.

Para la selección de una metodología de desarrollo de software es necesario establecer el modelo de proceso de ingeniería, según la naturaleza del proyecto, tomando en consideración los métodos y herramientas a utilizarse, así como el tipo de resultados que se requieren del producto.

Existen varios modelos de procesos de software que describen la estructura y organización de procesos y actividades para lograr los objetivos del desarrollo de software. Cada uno de los modelos abarcan los mismos procesos básicos de desarrollo como análisis, diseño, implementación y pruebas, pero ellos son organizados de manera distinta siguiendo un enfoque particular que guía al equipo de desarrollo. La selección de un modelo adecuado como guía es un punto importante debido a que este permite que los desarrolladores y jefes de proyecto realicen sus actividades coordinadas y con un alto desempeño para la consecución de los objetivos.^[7]

Los enfoques que presentan los modelos de procesos se pueden dividir en dos importantes grupos, los modelos clásicos y los modelos evolutivos. Entre los modelos clásicos según autores como Pressman, encontramos el Modelo Líneal Secuencial, de Cascada, el Modelo de Prototipo y el modelo de Desarrollo Rápido de Aplicaciones. Otros modelos evolutivos presentan un enfoque iterativo, es decir organizado por capas o regiones entre los cuales podemos encontrar el Modelo en Espiral, el Modelo Basado en Componentes, Métodos formales, y de cuarta generación.^[9]

Debido a que el sistema esta realizado bajo especificación J2EE es recomendable utilizar un modelo de desarrollo evolutivo o iterativo que incorpore el manejo de componentes, actividades y tareas para la creación del software. Se necesita de un modelo que permita al administrador de proyecto asignar las tareas y componentes que estén organizadas bajo un lenguaje unificado como puede ser el lenguaje UML.

^[7] PALACIO, Juan Bañeres. Compendio de Ingeniería de Software I

^[9] PRESSMAN, Roger; "Ingeniería de Software un enfoque práctico

Un modelo adecuado, que comprende fases e iteraciones que se adapta de mejor manera al proceso de desarrollo que se pretende lograr es el Proceso Unificado de Desarrollo de software, que consta de fases las cuales son organizadas en ciclos y que permiten adecuadamente llevar a cabo el proyecto de software

2.1.1. RATIONAL UNIFIED PROCESS (RUP) Y UNIFIED MODELING LANGUAGE (UML)

Es un modelo desarrollado por Rational, provee un enfoque que permite asignar tareas dentro de una organización de desarrollo, asegurando alta calidad en la producción de software y resolviendo las necesidades del usuario en base a presupuestos y un calendario establecidos, similar al modelo en Espiral consta de varios ciclos, los cuales terminan con un producto entregable al usuario, los ciclos denotan las iteraciones del modelo, cada ciclo consta de 4 fases.^[10]

- **Inicio:** define los objetivos del proyecto y su ámbito, a través de la identificación de los casos de uso que reflejan la visión general de los principales requisitos del proyecto sus características y limitaciones.
- **Elaboración:** fase en la cual se realiza la planificación, estableciendo una base sólida en la arquitectura, la asignación de recursos disponibles y eliminación de elementos de mayor riesgo del proyecto, para lo cual se han identificado la mayor parte de casos de uso y actores en el sistema
- **Construcción:** Se realiza el desarrollo de la aplicación basada en la gestión de las fases anteriores, la arquitectura, y los recursos que se utilizan, abarca la codificación y elaboración de documentación dando como resultado un producto de software preliminar que representa la iteración con tareas de análisis, diseño e implementación.
- **Transición:** En esta fase se realiza la instalación de una versión formal del sistema, además toma en cuenta la corrección de errores y se analiza el ambiente en donde el sistema se pondrá en producción, se finalizan toda la documentación de apoyo del sistema.

^[10] RUMBAUGH, James; Jacobson, Ivar. El Proceso Unificado de Desarrollo.

Para la preparación y realización de todos los “planos” del sistema, el proceso unificado de desarrollo de Rational utiliza el Lenguaje de Modelado Unificado, UML, como parte integral del proceso. Este lenguaje proporciona la notación común que se utiliza para realizar modelos de software, visualizar, especificar, construir y documentar sistemas, se focaliza en representar conceptual y físicamente el sistema a través de los métodos de solución del software. El Lenguaje UML provee distintas visualizaciones de la arquitectura del sistema, mostrando como ella evoluciona a través del ciclo de vida de desarrollo, por lo cual es necesario definir los artefactos que va a producir, que actividades realizan, y las personas que los crean, usan y administran. ^[28]

Para poder desarrollar el sistema benchmark que nos permitirá medir el rendimiento de los servidores de aplicación, se debe llevar a cabo los procesos inmersos en ciclos, los cuales refinarán el producto por cada iteración. Tomando en consideración que lo primordial para la fase inicial de un sistema es conocer el qué se quiere lograr, se debe seguir un análisis exhaustivo de los requerimientos con la elaboración del modelo conceptual con la ayuda de casos de uso para cada una de las actividades, para luego poder diseñar el producto mediante el modelo de diseño que nos presenta la arquitectura completa del sistema, estos modelos se llevarán a cabo utilizando los métodos que UML presenta para solución del software en los siguientes capítulos

2.2. ANÁLISIS DEL BENCHMARK.

Se debe empezar con un conocimiento total de los requerimientos del sistema, con esto se pretende identificar y documentar las necesidades reales acerca del producto, de tal manera que se cubra todas las expectativas del software^[11], entre las características que deben ser especificadas para tener una idea de los requerimientos del sistema a continuación se definen aspectos como:

2.2.1. PANORAMA GENERAL (PRESENTACIÓN GENERAL)

Este proyecto tiene por objeto crear un software que nos permita realizar mediciones del rendimiento de los servidores de aplicaciones Glassfish y JBOSS, cuando una aplicación

^[28] RATIONAL corporation. Rational Unified Process, Best Practices for Software Development Teams.

^[11] SHLARMAN, Craig; UML y Patrones. Introducción al Diseño orientado a objetos.

(portal Web como prototipo de prueba) se este ejecutando y almacenar dicha información para su posterior análisis.

2.2.2. USUARIOS

Los usuarios en este caso puede ser cualquier desarrollador de software que esté interesado en la medición de rendimiento de ambos servidores de aplicaciones, utilizando este prototipo de pruebas.

2.2.3. METAS

1. Presentar una aplicación Web que maneje funcionalidades de un Portal Web como son: consultas, registros y modificaciones de datos de clientes, así como la compra de productos de una empresa de ventas de computadoras y suministros, utilizando la administración del Servidor de Aplicaciones asignado (Glassfish o JBoss)
2. Simular la sobrecarga del servidor de aplicaciones para evaluar su comportamiento, cuando se realiza varias peticiones Web simultáneas.
3. Medir procesos realizados por los servidores de aplicación, tomando en cuenta los parámetros ya mencionados en la planeación como son: tiempo de respuesta, Rendimiento, utilización de recursos, cuando se realiza una o varias peticiones Web.
4. Calcular y almacenar datos estadísticos que permiten realizar un estudio del comportamiento en el rendimiento general.
5. Representar estos datos en forma gráfica para su entendimiento y análisis RESPECTIVO.

2.2.4. FUNCIONES DEL BENCHMARK

A continuación definimos una lista de funcionalidades del software Benchmark

2.2.4.1. Funciones de Portal Web

Referencia #	Función	Categoría
R 1.1	Permitir inicio de sesión de usuario registrado o visita de usuario no registrado	Evidente
R 1.2	Manejar perfil de usuario cliente registrado	Oculto

R 1.3	Manejar perfil de usuario cliente no registrado	Superflua
R 1.4	Manejar perfil de usuario medidor	Oculto
R 1.5	Registrar datos de cliente que requiera realizar una compra	Evidente
R 1.6	Emitir Factura de compra	Evidente
R 1.7	Realizar registro de ventas	Oculto
R 1.8	Actualizar el stock de productos	Oculto
R 1.9	Mostrar detalle de productos como descripción y precios	Evidente

Tabla 1. Funciones del Sistema respecto al Portal Web

2.2.4.2. Funciones de Medición y Representación gráfica

Referencia #	Función	Categoría
R 2.1	Medir tiempo de respuesta de una petición Web (tomando en cuenta que las peticiones Web puede ser cualquier acción que el usuario realice en el sistema, se tomarán varias referencias)	Oculto
R 2.2	Medir el rendimiento del servidor de aplicaciones (el número de operaciones ejecutadas por el sistema durante un intervalo de tiempo)	Oculto
R 2.3	Medir porcentaje de uso de Procesador durante el intervalo de petición Web	Oculto
R 2.4	Medir porcentaje de uso de memoria durante el intervalo de petición Web	Oculto
R 2.5	Medir el porcentaje de uso de red durante el intervalo de petición Web	Oculto
R 2.6	Representar gráficamente las mediciones de	Evidente

	tiempo de respuesta	
R 2.7	Representar gráficamente las mediciones de porcentajes de uso de procesador	Evidente
R 2.8	Representar gráficamente las mediciones de porcentajes de uso de memoria	Evidente
R 2.9	Representar gráficamente las mediciones de porcentajes de uso de red	Evidente
R 2.10	Calcular datos	Oculto

Tabla 2. Funciones del Sistema respecto al Benchmark

En las categorías de la funcionalidad se establecen 3 tipos:

Evidente: Debe realizarse y el usuario debería saber que se ha realizado

Oculto: Debe realizarse, aunque no es visible para los usuarios

Superfluas: Opcionales, no repercuten en la consecución de los objetivos planteados

2.2.5. ATRIBUTOS DEL BENCHMARK

Los atributos del benchmark son las características específicas que se requieren para la funcionalidad del sistema

2.2.5.1. Atributos generales del sistema

Sistema Operativo: El servidor de Aplicación estará instalado en el Sistema Operativo Windows 2003 Server, y los usuarios usan sistema operativo Windows XP Professional utilizando cualquier browser o navegador, con una versión mínima de Java Virtual Machine, así como habilitar la acción del browser para poder observar applets.

Base de Datos: Se utiliza la Base de Datos Oracle 9i

Especificación de proyecto: El proyecto está realizado utilizando la plataforma J2EE 1.4, lo cual implica utilización de las capas propias de J2EE vistas anteriormente, las versiones están sujetas a las compatibles con esta versión de J2EE.

2.2.5.2. Atributos respecto de las funciones:

De Portal Web

Ref. #	Función	Atributo	Detalles y Restricciones	Categoría de atributo
R 1.1	Permitir inicio de sesión de usuario registrado o visita de usuario no registrado	Usuario por sesión	Un usuario por sesión, Varios usuarios concurrentes	Obligatorio
R 1.5	Registrar datos de clientes que requieran realizar una compra	Usuario por sesión	Un usuario por sesión, Varios usuarios concurrentes	Opcional
R 1.6	Emitir Factura de compra	Usuario por sesión	Un usuario por sesión, Varios usuarios concurrentes	Opcional
R 1.7	Realizar registro de ventas	Número de peticiones	Una o Varias peticiones simultáneas	Opcional

Tabla 3. Atributos de las funciones del Portal Web

2.2.5.3. Atributos de Medición y Representación gráfica

Ref. #	Función	Atributo	Detalles y Restricciones	Categoría de atributo
R 2.1	Medir tiempo de respuesta de una petición Web (tomando en cuenta que la petición Web puede ser cualquier acción que el usuario realice en el sistema)	Número de mediciones	Las que se den en un intervalo de tiempo de 10 minutos	Obligatorio
R 2.2	Medir el rendimiento del servidor de	Número de	Las que se den en un intervalo de	Obligatorio

	aplicaciones (el número de operaciones ejecutadas por el sistema durante un intervalo de tiempo)	operaciones	tiempo de 10 minutos	
R 2.3	Medir porcentaje de uso de Procesador durante el intervalo de petición Web	Porcentajes	Tomada en 30 intervalos de 20 segundos durante 10 minutos	Obligatorio
R 2.4	Medir porcentaje de uso de memoria durante el intervalo de petición Web	Porcentajes	Tomada en 30 intervalos de 20 segundos durante 10 minutos	Obligatorio
R 2.5	Medir el porcentaje de uso de red en bytes transmitidos durante el intervalo de petición Web	Porcentajes	Tomada en 30 intervalos de 20 segundos durante 10 minutos	Obligatorio

Tabla 4. Atributos del Benchmark

Ref = referencia de la función

En las categorías de los atributos se establecen 2 tipos:

Obligatoria: Debe realizarse.

Opcional: no repercuten en la consecución de los objetivos planteados

2.2.6. MODELOS DE ANÁLISIS

La fase de análisis nos da como resultado varios artefactos que en conjunto son llamados modelo de análisis, estos son esenciales para la fase de diseño ya que nos permitirán conocer la estructura y el funcionamiento general del sistema.

Entre los artefactos contenidos en el modelo de análisis tenemos los siguientes:

1. Modelo de Casos de uso del Análisis
2. Diagrama de Casos de Uso
3. Modelo Conceptual
4. Modelo de Comportamiento del sistema

2.2.6.1. Modelo de Casos de Uso

Después de haber definido las funcionalidades del sistema, es preciso definir la frontera, identificar los actores y finalmente identificar y definir los casos de uso

2.2.6.1.1. Frontera del Software de Benchmark

La frontera del sistema define lo que es interno y externo para el sistema, así como las responsabilidades del sistema. En el caso del Software Benchmark podemos definir las fronteras del software excluyendo relativamente las acciones que realiza el Servidor de Aplicaciones es decir el sistema comprometerá sobre todo la funcionalidad del portal Web, así como también las funciones de medición y representación gráfica.

2.2.6.1.2. Actores del Software de Benchmark

El actor representa las entidades externas del sistema que participa en la historia del caso de uso, es quién estimula al sistema con eventos de entrada o recibe algo de él. En el caso del software del benchmark los actores fácilmente los podemos identificar:

- Servidor de Aplicaciones el cual va a ser evaluado
- Cliente
- Gestor de mediciones

2.2.6.1.3. Casos de uso de Análisis

Para identificar los casos de uso partimos de los actores, los cuales interactúan con el sistema:

2.2.6.1.3.1. Acciones de los actores

Servidor de Aplicaciones	Administra o gestiona procesos de la aplicación
Cliente	Ejecuta petición Web de Cliente
Gestor de Mediciones	Ejecuta la gestión de mediciones realizadas al usar el portal Web

Tabla 5. Actores del Sistema

2.2.6.1.3.2. Peticiones Web de Cliente

- Inicia sesión Cliente
- Registra sus datos
- Busca Productos
- Selecciona Productos de compra
- Compra Productos
- Paga Compra
- Imprime Factura
- Cierra sesión Cliente

2.2.6.1.3.3. Tipos de casos de uso:

Por prioridad

- Primario
- Secundario

Por nivel de detalle

- Esencial
- Real

2.2.6.1.3.1. Diagrama General de los Casos de Uso

El diagrama General de los Casos de Uso nos permitirá partir para establecer la estructura de cada uno de ellos y plantearlos individualmente. Para este diagrama partimos de las interacciones de los usuarios con las Peticiones Web antes expuestas

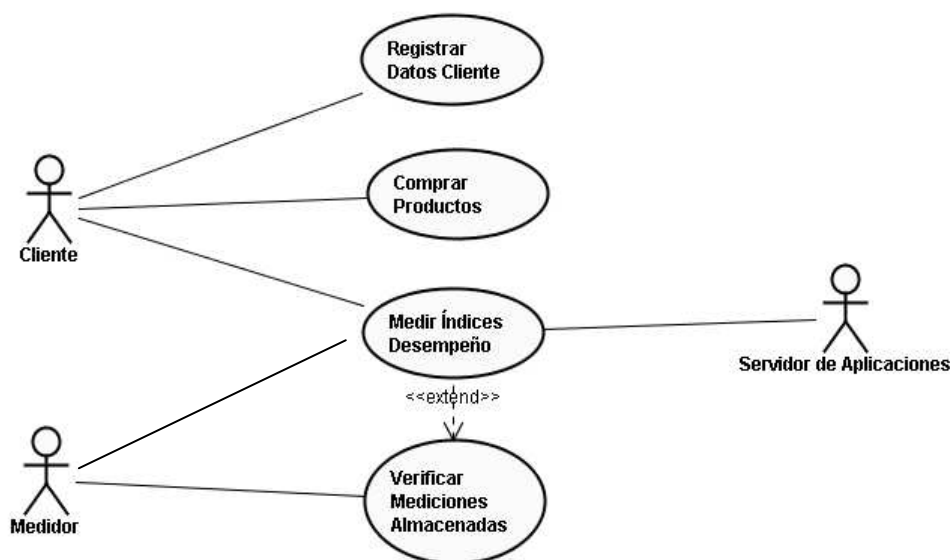


Figura 4. Diagrama General de los casos de uso

Como podemos observar se han agrupado los principales casos de uso para la interacción con el usuario Cliente: Registrar Datos Cliente y Comprar Productos, que contienen de manera general las peticiones Web que serán medidas por el Benchmark cuando esté interactuando con el Servidor de Aplicaciones.

Los casos de uso se los revisará no en el orden del Diagrama sino siguiendo la importancia del Software Benchmark en sí, es decir dando prioridad a las mediciones.

2.2.6.1.3.2. Caso de Uso: C1. Medir Índices de desempeño

Actores: Cliente, Servidor de aplicaciones, Medidor (iniciador).

Tipo: Primario, esencial

Propósito: Capturar mediciones de los diferentes índices para medir el desempeño

Diagrama:

El Caso de uso que representa el accionar del benchmark es Medir índices de Desempeño ya que en él se lleva a cabo a la dirección de medidas de los índices y sus parámetros; su diagrama está representado:

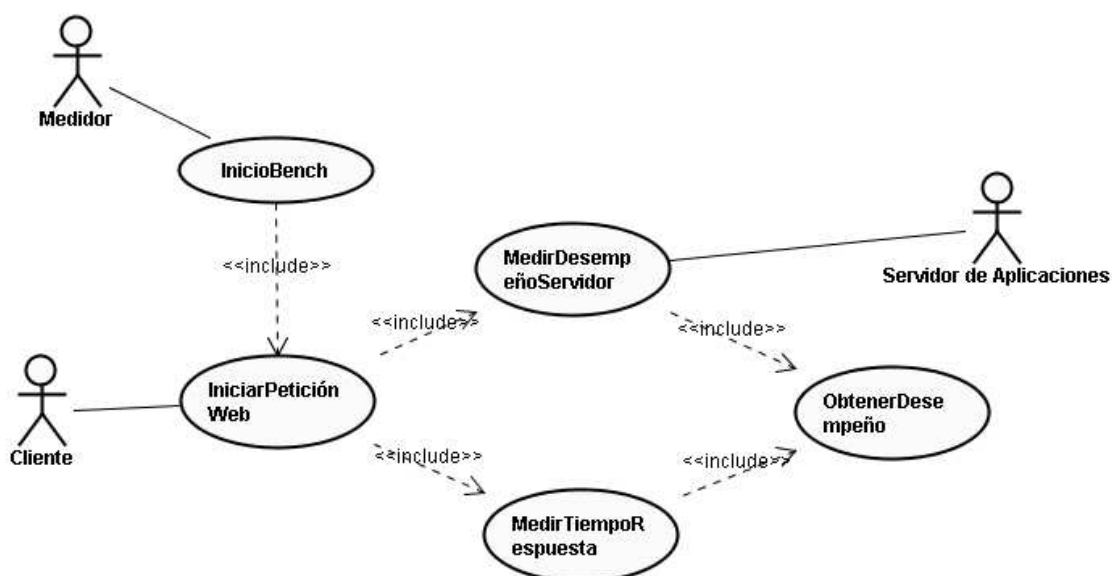


Figura 5. Diagrama de casos de uso Benchmark

Descripción: El Medidor activa la funcionalidad del Benchmark para que comience a medir las acciones del Cliente, en esta parte el Medidor puede pasar de ser usuario medidor a usuario Cliente, el usuario Cliente realiza una petición Web, el Servidor de Aplicaciones ejecuta administración y se comienza a medir un índice de rendimiento, al completar la respuesta al Cliente o un intervalo de tiempo asignado, el Servidor concluye su administración y se termina de medir el índice. Finalmente los datos son almacenados. Si el medidor no realiza el inicio del Benchmark, las mediciones no comienzan.

Referencias Cruzadas:

Casos de Uso: C3, C4, C5

Curso Normal de los eventos

Acción del Actor	Respuesta del Sistema
1. Este caso de uso comienza cuando el Medidor inicia el Benchmark y el Cliente realiza una petición Web (el	

medidor puede tornarse usuario cliente en este caso	
2. El Servidor de Aplicaciones comienza la administración, seleccionando el servicio que requiere la petición	
	3. Inicia la medición del parámetro o índice de desempeño
	4. Realiza la petición Web, entrega los resultados de la petición al Cliente
5. El Cliente recibe todas las respuestas del sistema, el estado de la conexión llega a estado terminado. O el intervalo de tiempo es cumplido.	
6. El Servidor de Aplicaciones termina la administración.	
	7. Finaliza la medición del parámetro o índice de desempeño.
	8. Almacena datos de medidas

Cursos Alternos:

Línea 5: El Cliente no recibe todas las respuestas a la o las peticiones realizadas al sistema o el estado de la conexión no llega a estado terminado. Se cancela medición, se indica existencia de error.

Tomando en cuenta que el caso de uso de Medir Índices de Desempeño es central para el proceso de desarrollo de software este lo podemos detallar en varios casos de uso que lo conforman, ellos son:

1. Caso de Uso: Iniciar Petición Web

2. Caso de Uso: Medir Desempeño del Servidor

3. Caso de Uso: Medir Tiempos de respuesta

4. Caso de Uso: Obtener desempeño

2.2.6.1.3.2.1. C1.1. Caso de uso Iniciar Petición Web

Actores: Cliente (iniciador)

Tipo: Secundario, esencial

Propósito: Proponer al Servidor de Aplicaciones una transacción, para que esta sea administrada y medir su desempeño

Diagrama:

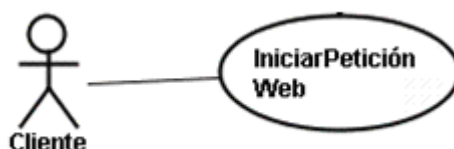


Figura 6. Diagrama de caso de uso Iniciar Petición Web

Descripción: Cliente realiza una petición Web, la cual puede ser cualquiera de las peticiones presentadas en los casos de uso: Comprar Productos, Registrar Datos de Cliente, Navegar por el portal

Referencias Cruzadas:

Casos de Uso: C3, C4, C5

Curso Normal de los eventos

Acción del Actor	Respuesta del Sistema
1. Este caso de uso comienza cuando el Cliente realiza una petición Web, esta petición involucra a uno de los casos de uso C3, C4 o C5	

2. El cliente en este caso simulará la realización de una o varias peticiones Web simultáneas (planificadas) para crear un ambiente realista de trabajo	
	3. El sistema recibe la petición y administra los recursos Web para realizar dicha petición

Cursos Alternos:

Línea 2. El cliente puede simular una o varias peticiones a la vez, el número de peticiones es planificado según se requiera para establecer el rendimiento del servidor.

2.2.6.1.3.2.2. C1.2 Caso de uso Medir Desempeño de Servidor

Actores: Servidor de aplicaciones.

Tipo: Secundario, esencial

Diagrama:



Figura 7. Diagrama de caso de uso Medir desempeño del Servidor

Propósito: Capturar mediciones de los índices respecto al servidor de aplicaciones y sus recursos, por lo que en este caso se tomarán en cuenta los índices: rendimiento, utilización de recursos y operaciones ejecutadas por el Servidor.

Descripción: Este caso de uso comienza cuando el Servidor de Aplicaciones comienza a administrar las peticiones realizadas, asignando recursos y tareas, creando instancias y dominios para cada petición, se mide parámetros como tiempo total de uso de servidor, porcentajes de uso de los componentes: CPU, memoria y red.

Referencias Cruzadas:**Casos de Uso: C1.1****Curso Normal de los eventos**

Acción del Actor	Respuesta del Sistema
1. Este caso de uso comienza cuando el Servidor de Aplicaciones comienza la administración de la primera petición Web, asignándole recursos, creando instancias y dominios	
	2. Inicia la medición del tiempo desde que comienza la administración Web
	3. Contabiliza el número de peticiones que son administradas
	4. Inicia medición de porcentaje de uso de CPU, y su almacenamiento en documento
	5. Inicia medición de porcentaje de uso de memoria y su almacenamiento en documento
6. Inicia medición de porcentaje de uso de red y su almacenamiento en documento	7. El Servidor de Aplicaciones finaliza la administración o se cumple el intervalo de tiempo máximo de medición para Servidor de Aplicaciones (planificado)

Línea 1. Si el Servidor de aplicaciones falla en su administración, el proceso se detiene, y se envía error

Línea 7. El Servidor de Aplicaciones puede finalizar la administración o la medición termina cuando se cumple un intervalo de tiempo

2.2.6.1.3.2.3. C1.3. Caso de uso Medir Tiempos de Respuesta

Actores: Cliente

Tipo: Secundario, esencial

Propósito: Capturar mediciones de los índices respecto a la Petición Web realizada por el Cliente

Descripción: Una vez que el Cliente inicia una petición Web determinada se inicia la medición del tiempo que esta tarda en ser totalmente atendida y los resultados son presentados en su totalidad al Cliente, por lo tanto el parámetro tiempo de respuesta es medido en su totalidad. Las peticiones Web que se determinarán para ser medidas son aquellas que el Servidor de Aplicaciones genere trabajo como son el Registro de Cliente y la compra de productos.

Referencias Cruzadas:

Casos de Uso: C1.1

Curso Normal de los eventos

Acción del Actor	Respuesta del Sistema
1. Este caso de uso comienza después de iniciada la petición Web.	
	2.. Inicia la medición del parámetro tiempo de respuesta
3. El Cliente recibe todas las respuestas del sistema a la petición Web, el estado de la conexión llega a estado terminado.	

	4. Finaliza la medición del parámetro tiempo de respuesta
--	---

Línea 3: El cliente no recibe todas las respuestas del sistema por falla de conexión por lo tanto no se llega a un estado terminado, se finaliza la medición.

2.2.6.1.3.2.4. C1.4. Caso de uso Obtener Desempeño

Actores: Cliente (iniciador), Servidor de aplicaciones.

Tipo: Secundario, esencial

Propósito: Capturar mediciones de los diferentes índices para medir el desempeño

Diagrama:

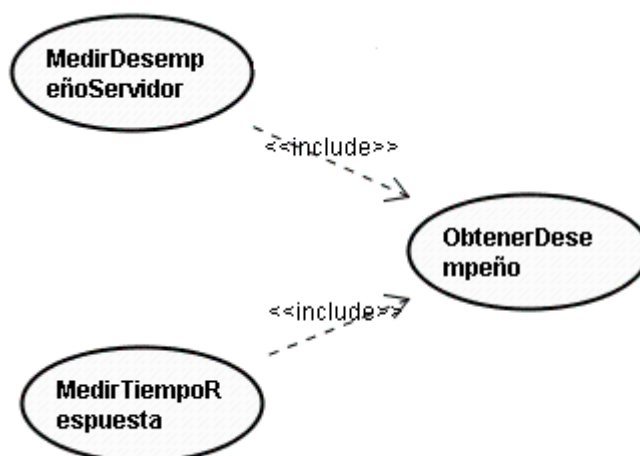


Figura 8. Diagrama de casos de uso obtener desempeño

Descripción: Cliente realiza una petición Web, el Servidor de Aplicaciones ejecuta administración y se comienza a medir un índice de rendimiento, al completar la respuesta al Cliente, el Servidor concluye su administración y se termina de medir el índice. Finalmente los datos son almacenados.

Referencias Cruzadas:

Casos de Uso: C1.2, C.3

Curso Normal de los eventos

Acción del Actor	Respuesta del Sistema
1. Este caso de uso comienza cuando se ha finalizado la medición de los parámetros	
	2. Se almacenan los datos obtenidos en documentos manejables para más adelante ser analizados
3. El Cliente puede realizar una nueva Petición Web inicial o finalizar acción	

2.2.6.1.3.3. Caso de Uso: C2.Verificar Mediciones almacenadas

Actores: Medidor

Tipo: Primario, esencial

Propósito: Comprobar validez de mediciones y concluir resultados generales de medición

Diagrama:



Figura 9. Diagrama de caso de uso Verificar Mediciones Almacenadas

Descripción: El medidor requiere datos de los índices almacenados, selecciona los índices, verifica datos almacenados del parámetro indicado y genera un gráfico, verifica gráficos y datos y selecciona otro parámetro o sale del sistema.

Referencias Cruzadas:

Casos de Uso: C1.

Curso Normal de los eventos

Acción del Actor	Respuesta del Sistema
1. Este caso de uso comienza cuando el Medidor inicia sección de comparación de servidores	
	2. Muestra los índices e intervalos de los cuales se requiere la información
3. El medidor selecciona un índice y un intervalo de mediciones (previamente planificado)	
	4. Calcula datos estadísticos necesarios para gráficos y mediciones
	5. Muestra gráficos y datos estadísticos correspondientes al índice y al intervalo seleccionados
6. El Medidor finaliza sección de mediciones	

Cursos Alternos:

Línea 5. Error en los datos analizados, se indica el error por tratarse de falta de datos en el intervalo

Línea 6. El Medidor puede pasar directamente al paso 8

Línea 6. El Medidor puede volver al paso 3

2.2.6.1.3.3. Caso de Uso: C3. Comprar Producto

Actores: Cliente

Tipo: Primario, esencial

Propósito: Simular petición Web de comprar producto capturando la venta y su pago, petición común para medir índices de desempeño

Diagrama:



Figura 10. Diagrama de caso de uso Comprar Producto

Descripción: El Cliente inicia sesión, selecciona el o los productos que desee adquirir, los que virtualmente son agregados a un carrito de compras, cuando finalice la selección, se muestra al Cliente el detalle de compras en donde están incluidos los datos de los productos y valores para la compra que realizará, el Cliente acepta la factura y procede a realizar el pago, realizado el pago el Cliente cierra sesión, los productos son enviados al domicilio del Cliente virtualmente.

Referencias Cruzadas:

Casos de Uso: C1, C4

Curso Normal de los eventos

Acción del Actor	Respuesta del Sistema
1. Este caso de uso comienza cuando el Cliente inicia sesión en el sistema, se identifica con sus datos	
	2. Se muestra una variedad de productos divididos en Laptos, Impresoras y Mainboards, que la “empresa” ofrece a sus “Clientes”

3. El Cliente selecciona el o los productos que desea adquirir	
	4. Se almacena la lista de productos agregados en un carrito de compras
5. El Cliente finaliza la selección de los productos agregándolos al carrito virtual	
	6. Muestra la factura previa con los productos que el Cliente adquiere, los precios individuales y el precio final
7. El Cliente acepta la factura	
	8. Se pide confirmación de datos de tarjeta de crédito para pago
9. El Cliente realiza el pago y acepta	
	10. Se muestra confirmación de venta e información de envío
11. El Cliente cierra sesión	
	12. Se envían productos a domicilio y se actualiza stock

Cursos Alternos:

Línea 1: El Cliente no registrado puede observar los productos pero para agregarlos debe registrarse o si ya lo hizo iniciar su sesión.

Línea 3. El Cliente puede solamente observar los productos y luego continúa con el paso 11, sin realizar el paso 12

Línea 7. El Cliente puede eliminar productos de la factura y aceptar

Línea 7: El Cliente puede regresar al paso 2

Línea 9: El Cliente puede cancelar el proceso o regresar a punto 6

Línea 10: El Cliente puede imprimir información

Línea 11: El Cliente continúa al paso 2.

2.2.6.1.3.4. Caso de Uso: C4. Registrar Datos de Cliente

Actores: Cliente

Tipo: Secundario, esencial

Propósito: Simular petición Web de registro real de Cliente almacenando datos de Clientes.

Diagrama:



Figura 11. Diagrama de caso de uso Registrar Dato de clientes

Descripción: el Cliente realiza petición de registrar datos personales y de compra, ingresa datos personales y datos de envío, los datos son almacenados de forma segura y se emite una bienvenida.

Referencias Cruzadas:

Casos de Uso: C1,

Curso Normal de los eventos

Acción del Actor	Respuesta del Sistema
1. Este caso de uso comienza cuando un Cliente no registrado decide realizar su registros en el sistema para poder realizar	

compras	
	2. Pide datos personales (obligatorios)
	3. Pide datos de envío dirección, teléfono
4. El Cliente proporciona los datos	
	5. Se almacenan los datos
	6. Se muestra una bienvenida al usuario
7. El Cliente continúa inicia sesión	

Cursos Alternos:

Línea 7. El Cliente puede Salir del sistema.

Una vez establecidos los casos de uso es preciso establecer las prioridades de los casos de uso para establecer los ciclos o iteraciones que se llevarán a cabo para el proyecto.

Por lo cual tenemos tres tipos de prioridad. Alto, mediano, bajo.

CASO DE USO	PRIORIDAD
Medir índices de desempeño	Alta
Verificar mediciones almacenadas	Alta
Comprar Producto	Mediana
Registrar Datos de Cliente	Mediana
Iniciar Petición Web	Mediana
Medir Desempeño Servidor	Alta
Medir Tiempo Respuesta	Alta
Obtener Resultados de Desempeño	Mediana

Tabla 6. Clasificación de Prioridad de casos de uso

Según la clasificación es factible dividir la realización del proyecto en iteraciones las cuales se realizarán según el tipo de complejidad de los casos de uso, así como también versiones y capas que serán implementadas en cada caso de uso hasta completar las iteraciones y el ciclo de vida del proyecto.

2.2.6.2. Modelo Conceptual

El modelo conceptual nos permitirá visualizar al proyecto desde una perspectiva de conceptos con atributos y asociaciones entre los conceptos, basados en el análisis de casos de uso podemos obtener el siguiente modelo conceptual que posteriormente nos permitirá realizar el esquema entidad relación para la base de datos.

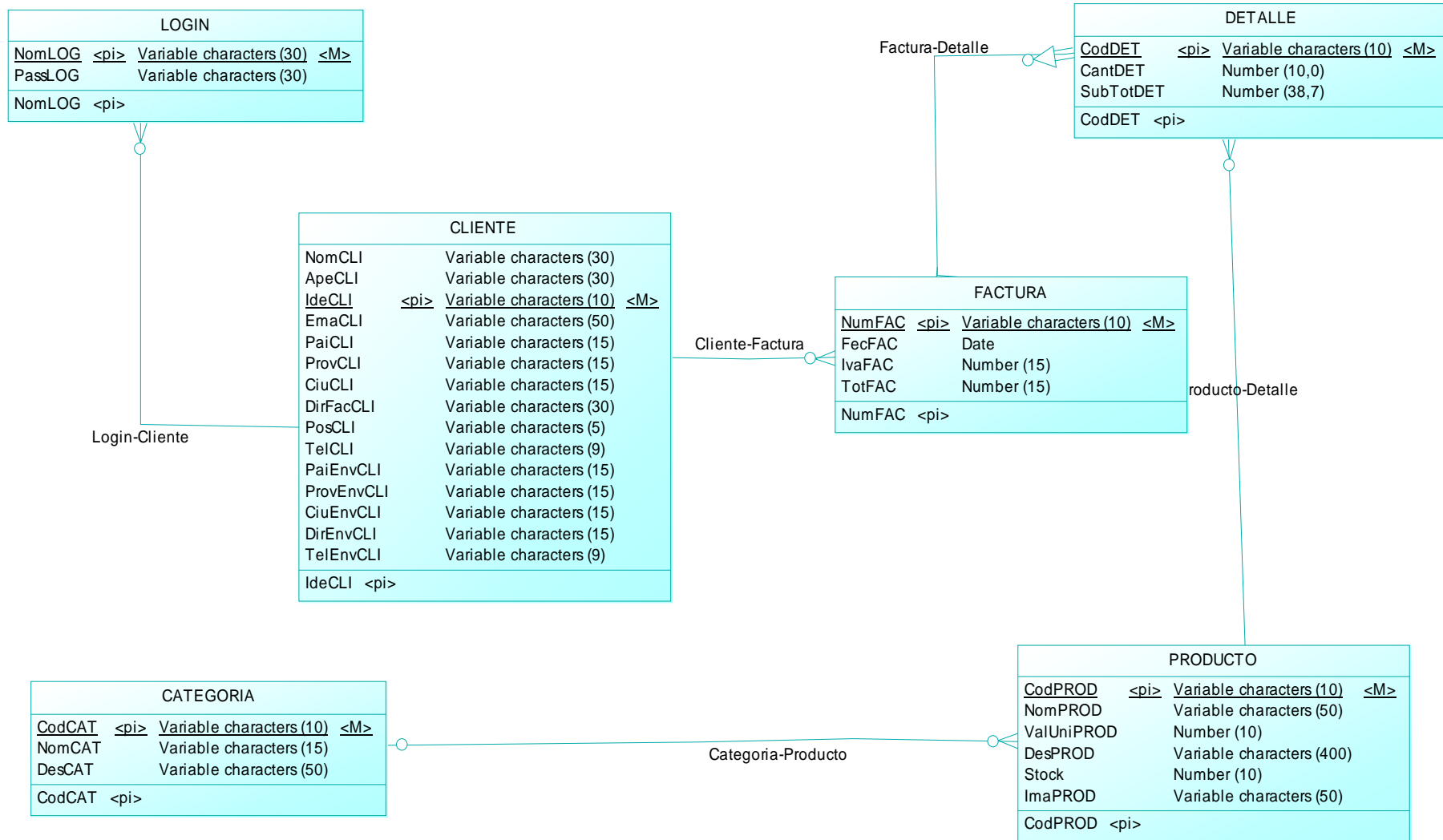


Figura 12. Diagrama Modelo Conceptual Benchmark

El Modelo Conceptual nos permite visualizar los conceptos que rodean la solución del software, con los atributos que cada uno presenta, así como sus relaciones con los otros conceptos en el Modelo Conceptual los conceptos son Entidades.

Como podemos observar los conceptos claves para tener una visión clara del planteamiento general son el cliente y el producto, los cuales se encuentran relacionados por los datos de detalle de compra, y rodeados de conceptos que abarcan la descripción de ellos como marca de producto, tipo, detalle, así como también la tarjeta de crédito y datos de acceso con respecto al cliente.

Una vez visualizado el modelo conceptual nos corresponde establecer un modelo de Comportamiento en donde se muestra lo que hace el sistema sin explicar la manera en como lo hace.

2.2.6.3. Modelo de Comportamiento

El modelo de comportamiento muestra como el sistema reaccionará ante eventos externos suscitados mostrándose externamente como una caja negra, posteriormente se añadirá las funcionalidades del sistema en un diagrama más amplio

2.2.6.3.1. Diagrama de secuencias

Con el diagrama de secuencias se explica los eventos que inducen a que el sistema realice una respuesta y están basados en los casos de uso, por lo tanto existe un diagrama por caso de uso.

2.2.6.3.1.1. Diagrama de secuencia: Medir índices de desempeño

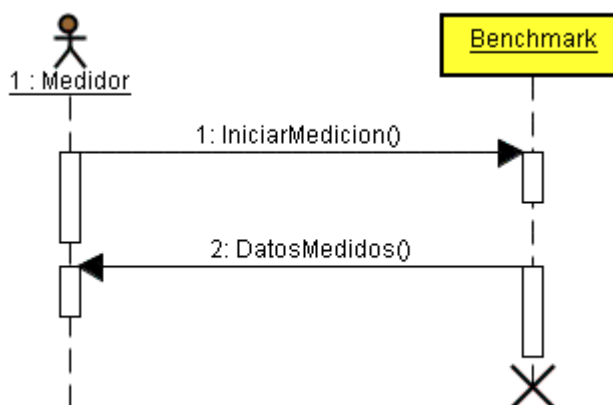


Figura 13. Diagrama de secuencia Medir Índice de desempeño

2.2.6.3.1.2. Diagrama de secuencia: Verificar Mediciones Almacenadas

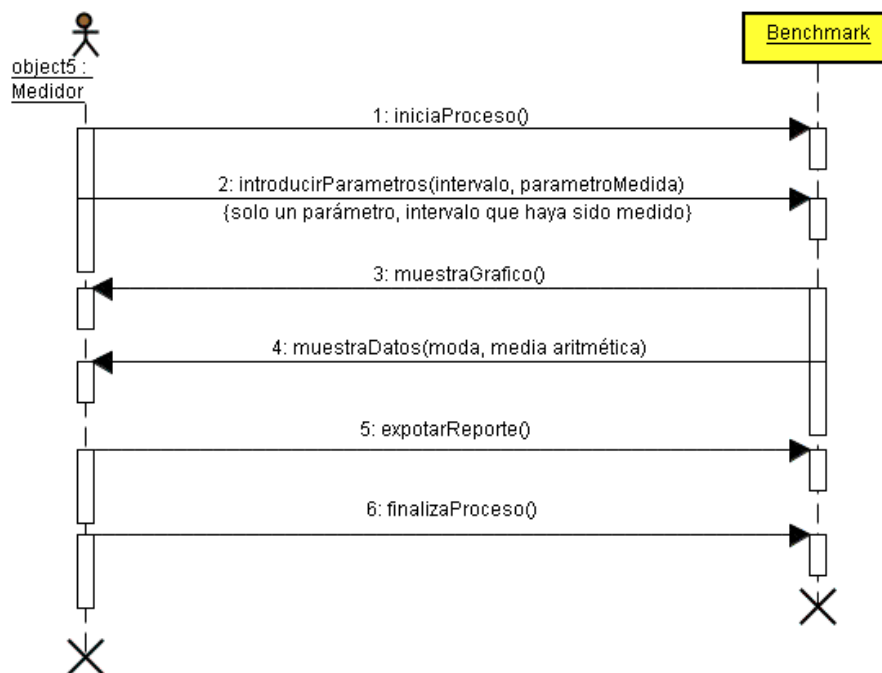


Figura 14. Diagrama de secuencia verificar mediciones almacenadas

2.2.6.3.1.3. Diagrama de secuencia: Comprar Productos

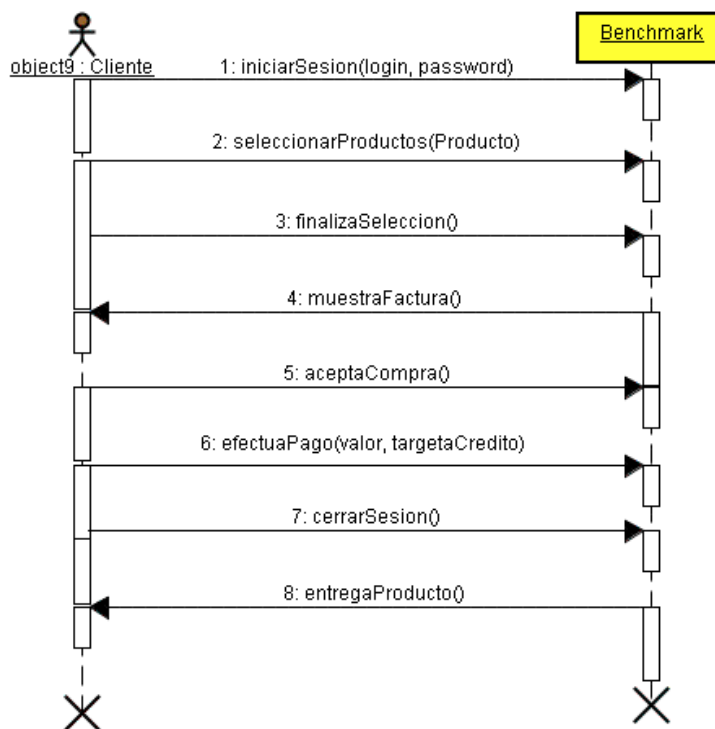


Figura 15. Diagrama de Secuencia comprar productos

2.2.6.3.1.4. Diagrama de secuencia: Registrar Datos de Cliente

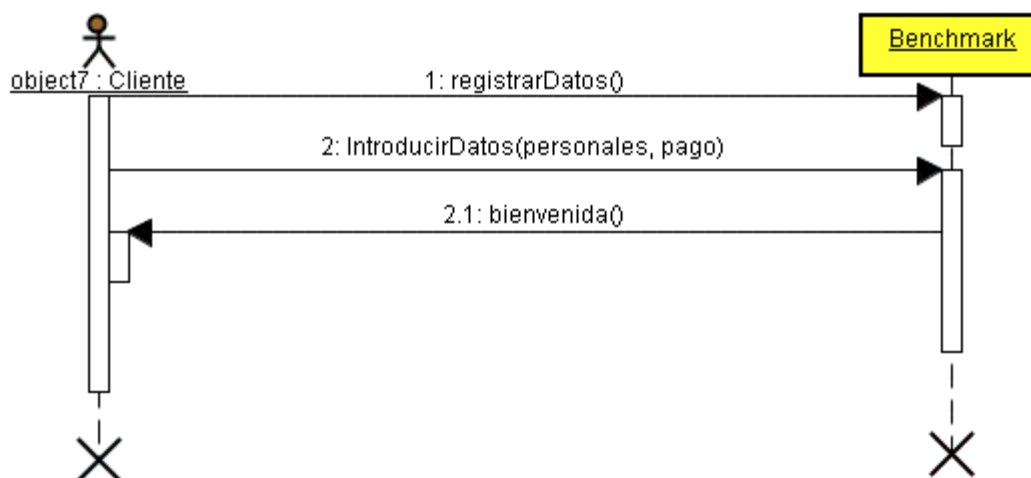


Figura 16. Diagrama de secuencia Registrar datos de cliente

Una vez analizados las secuencias tomando en cuenta al benchmark como caja cerrada podemos ampliar el análisis de la secuencia de los casos de uso que corresponden a la medición de los índices de desempeño, que es la parte medular del benchmark los cuales son especificados a continuación:

2.2.6.3.1.5. Diagrama de secuencia: Medir Desempeño Servidor de Aplicaciones

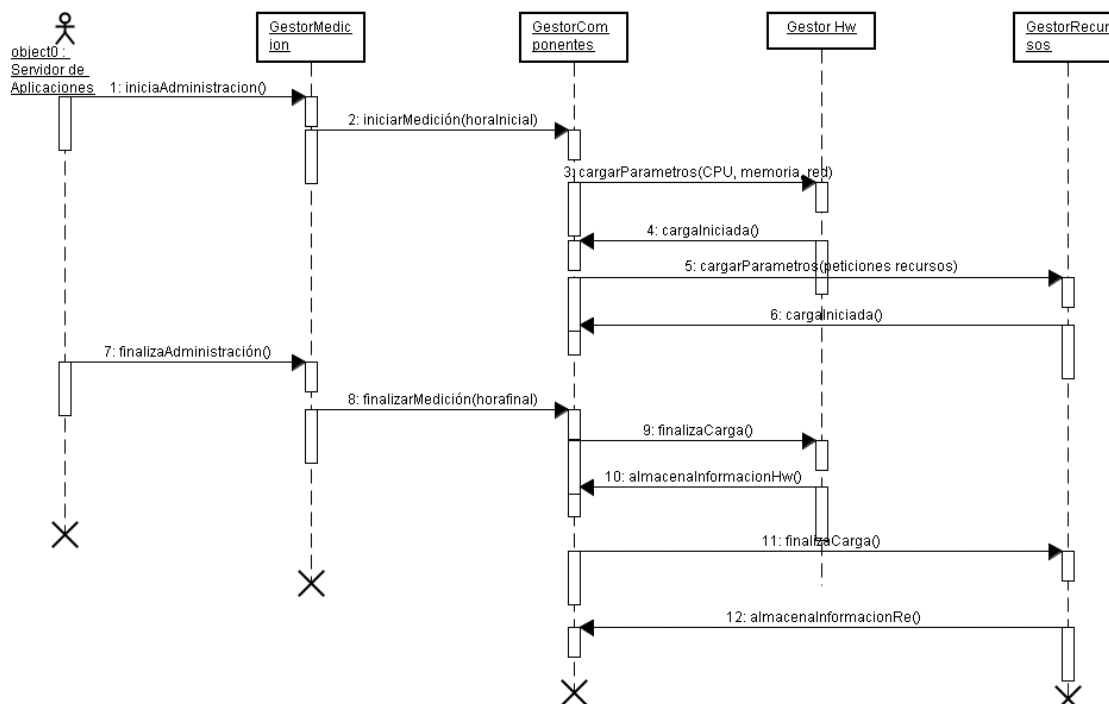


Figura 17. Diagrama de secuencia Medir Desempeño de servidor

Como podemos observar para controlar la medición necesitamos de nuevos artefactos gestores quienes se encargan de la organización de las mediciones los cuales son:

Gestor de Medición: Se encarga del control de inicio de medición y finalización de medición, entregando la hora inicial y final de medición según lo planificado

Gestor de Componentes: Administra la carga de parámetros que comienzan a ser medidos, se encarga de las relaciones con el sistema operativo para carga de parámetros Hardware como CPU, memoria y red y rendimiento.

2.2.6.3.1.6. Diagrama de secuencia: Medir Tiempo de respuesta Petición Web

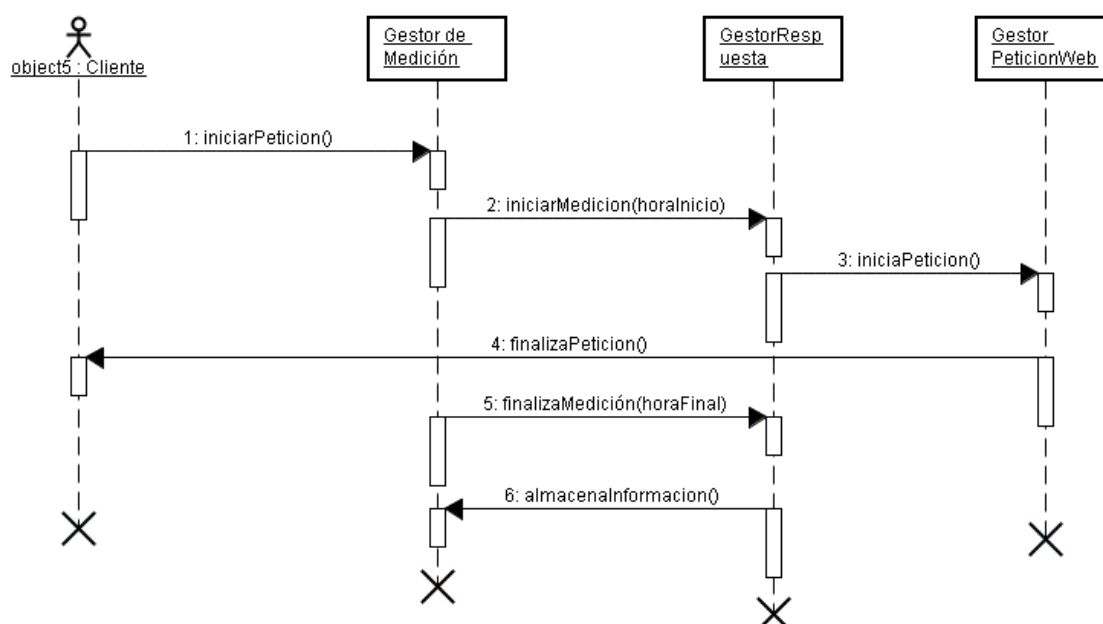


Figura 18. Diagrama de secuencia Medir tiempo de respuesta Petición Web

De igual manera en este caso tenemos la aparición de nuevos modelos que nos ayudarán a visualizar el envío de mensajes entre ellos para obtener las mediciones de los parámetros:

Gestor Tiempo de Respuesta: Permite el control de una petición Web la cual es medida desde que el Cliente la inicia hasta que la respuesta es totalmente recibida por el Cliente

Gestor Petición Web: No es más que la representación de la realización de una petición Web, que para el efecto de nuestro análisis están representados por Registro de Cliente y Compra de Productos.

2.3. DISEÑO DEL BENCHMARK.

El diseño conlleva a aplicar los artefactos, diagramas y modelos que se han producido del análisis del benchmark, lo cual nos permite establecer nuevos modelos de la arquitectura, los que nos guiarán en la construcción del Benchmark y posterior implementación:

1. Modelo de Casos de Uso de Diseño: Casos de uso reales.
2. Modelo de Comportamiento: Diagrama de interacción.
3. Modelo de clases: Diagrama de clases de diseño
4. Modelo de la arquitectura

2.3.1. MODELO DE CASOS DE USO DE DISEÑO

Este modelo nos permite formalizar el modelo realizado en la fase de análisis, diseñando a detalle la funcionalidad del software del benchmark, el modelo consiste en los casos de uso con sus diagramas los cuales no variarán respecto a los de análisis por lo cual el aspecto primordial es el diseño de los casos de uso finales

2.3.1.1. Casos de Uso de Diseño.

2.3.1.1.1. Caso de Uso: C1. Medir Índices de desempeño

El caso de uso medir índices de desempeño se subdivide en cuatro nuevos casos de uso como son Iniciar Petición Web, Medir Desempeño del Servidor, Medir Tiempos de Respuesta y Obtener resultados, todos ellos iniciados después de que el Medidor ha activado o iniciado las mediciones del Benchmark. Por lo tanto son ellos quienes deben especificar las funcionalidades reales del benchmark.

2.3.1.1.1.1. C1.1. Caso de uso Iniciar Petición Web

Actores: Cliente

Tipo: Secundario, real

Propósito: Proponer al Servidor de Aplicaciones una transacción, para que esta sea administrada y medir su desempeño

Descripción: Este caso de uso inicia realmente cuando un Cliente realiza la primera o primeras peticiones Web simuladas que en este caso la llamaremos petición de inicio, la

cual comenzará con el proceso de medición del rendimiento general del Servidor de Aplicaciones, como anteriormente fue explicado estas peticiones se mantienen dentro del ámbito de los casos de uso que pertenecen a Peticiones Web como son: Comprar Productos, Registrar Datos Cliente, a demás la navegación por la página Web.

Curso Normal de los eventos

Acción del Actor	Respuesta del Sistema
1. Este caso de uso comienza cuando el Cliente realiza una petición Web, esta petición involucra a uno de los casos de uso C3, C4 o C5	
2. El cliente en este caso simulará la realización de una o varias peticiones Web simultáneas (planificadas) para crear un ambiente realista de trabajo	
	3. El sistema recibe la petición y administra los recursos Web para realizar dicha petición con la posterior acción del Servidor de Aplicaciones.

Cursos Alternos:

Línea 2. El cliente puede simular una o varias peticiones a la vez, el número de peticiones es planificado según se requiera para establecer el rendimiento del servidor. La planificación de las simulaciones se realizará en un capítulo posterior de planificación de mediciones.

2.3.1.1.1.2. C1.2 Caso de uso Medir Desempeño de Servidor

Actores: Servidor de aplicaciones.

Tipo: Secundario, real

Propósito: Capturar mediciones de los índices respecto al Servidor de Aplicaciones y sus recursos, por lo que en este caso se tomarán en cuenta los índices: rendimiento, utilización de recursos y demanda de servicios.

Descripción: Este caso de uso comienza cuando el Servidor de Aplicaciones comienza a administrar las peticiones realizadas, asignando recursos y tareas, creando instancias y dominios para cada petición, se mide parámetros como tiempo total de uso de servidor, número de peticiones atendidas, recursos utilizados del servidor, porcentajes de uso de los componentes: CPU, memoria y red.

Curso Normal de los eventos

Acción del Actor	Respuesta del Sistema
<p>1. Este caso de uso comienza una vez que se ha realizado una o varias peticiones Web y el Servidor de Aplicaciones comienza la administración de la petición de inicio, asignándole recursos, creando instancias y dominios según lo requiera la petición, estas acciones son las que van ser cuantificadas por lo que son tratadas como acción del Servidor de Aplicaciones como actor</p>	
	<p>2. Se ejecuta un Gestor de Mediciones, el cual se encarga de constatar que los procesos de mediciones se realice de manera sincrónica, se maneja conjuntamente con el Sistema operativo el reloj que medirá tiempos (planificado) del trabajo del Servidor de Aplicaciones.</p> <p>3. Se ejecuta un Gestor de componentes que controlará el tiempo total, medido en</p>

	milsegundos, del intervalo, y realizará la carga de componentes individuales para medir el trabajo del Servidor
	4. Se ejecuta un Gestor de Hardware previamente cargados los parámetros: CPU, memoria y red: se comienza por separado la medición de dichos parámetros en datos porcentuales obtenidos con la ayuda del Sistema Operativo
5. El Servidor de Aplicaciones finaliza la administración o se cumple el intervalo de tiempo máximo de medición para Servidor de Aplicaciones (planificado)	
	6. Una vez cumplido el intervalo de medición los gestores de Hardware y Recursos finalizan las mediciones y almacenan la información recavada pasando al gestor de componentes la finalización satisfactoria
	7. Finalizan el control de medición por parte del Gestor de mediciones

Línea 1. Si el servidor de aplicaciones falla en su administración, el proceso se detiene, y se envía error

Línea 5. El Servidor de Aplicaciones puede finalizar la administración o la medición termina cuando se cumple un intervalo de tiempo controlado por el Gestor de componentes.

2.3.1.1.1.3. C1.3. Caso de uso Medir Tiempos de Respuesta

Actores: Cliente

Tipo: Secundario, real

Propósito: Capturar mediciones de los índices respecto a la Petición Web realizada por el Cliente

Descripción: Una vez que el Cliente inicia una petición Web se inicia la medición del tiempo que esta tarda en ser totalmente atendida y los resultados son presentados en su totalidad al Cliente, por lo tanto el parámetro tiempo de respuesta es medido en su totalidad.

Curso Normal de los eventos

Acción del Actor	Respuesta del Sistema
1. Este caso de uso comienza después de iniciada la petición Web por parte del Cliente.	
	2. Se ejecuta un Gestor de Mediciones, el cual se encarga de constatar que los procesos de mediciones se realice de manera adecuada, se maneja conjuntamente con el Sistema operativo el reloj que medirá tiempos (planificado) de una petición Web.
	3. Se ejecuta el Gestor de Respuesta comienza a medir el tiempo en realizarse completamente una petición Web (planificada), tomando en cuenta que simultáneamente se realizan otras peticiones
	4. Se ejecuta el Gestor de Petición Web, que en este caso simplemente realiza un

	caso de uso C3, C4 o C5
5. El Cliente recibe todas las respuestas del sistema a la petición Web, el estado de la conexión llega a estado terminado.	
	6. Finaliza la medición del parámetro tiempo de respuesta, se almacena la información en el Gestor de Respuesta
	8. Finalizan el control de medición por parte del Gestor de mediciones

Línea 3: El cliente no recibe todas las respuestas del sistema por falla de conexión por lo tanto no se llega a un estado terminado, se finaliza la medición.

2.3.1.1.1.4. C1.4. Caso de uso Obtener Desempeño

Actores: Cliente (iniciador), Servidor de aplicaciones.

Tipo: Secundario, real

Propósito: Capturar mediciones de los diferentes índices para medir el desempeño

Descripción: Cliente realiza una petición Web, el Servidor de Aplicaciones ejecuta administración y se comienza a medir un índice de rendimiento, al completar la respuesta al Cliente, el Servidor concluye su administración y se termina de medir el índice. Finalmente los datos son almacenados.

Curso Normal de los eventos

Acción del Actor	Respuesta del Sistema
1. Este caso de uso comienza cuando se ha finalizado la medición de los parámetros	

	2. Se almacenan los datos obtenidos en documentos manejables como Microsoft Excel para más adelante ser analizados y realizar los gráficos y cálculos respectivos
3. El Cliente puede realizar una nueva Petición Web inicial o finalizar acción	

2.3.1.1.2 Caso de Uso: C2.Verificar Mediciones almacenadas

Actores: Medidor

Tipo: Primario, real

Propósito: Comprobar validez de mediciones y concluir resultados generales de medición

Descripción: El Medidor requiere datos de los índices almacenados, ingresa intervalo de tiempo, y selecciona índices, verifica datos almacenados del parámetro indicado y genera un gráfico, verifica gráficos y datos y selecciona otro parámetro o sale del sistema.

Curso Normal de los eventos

Acción del Actor	Respuesta del Sistema
1. Este caso de uso comienza cuando el Medidor inicia sección de comparación de servidores	
	2. Se muestra los índices e intervalos de los cuales se requiere la información, los cuales están divididos en las 4 secciones que indican los parámetros para cada uno: tiempo de respuesta de una petición Web

	y la utilización de recursos (CPU, memoria o red)
3. El Medidor selecciona un índice y un intervalo de mediciones de las opciones presentadas (previamente planificado)	
	4. Calcula datos estadísticos necesarios para gráficos y mediciones, dependiendo del índice y parámetro seleccionados.
	5. Muestra gráficos y datos estadísticos correspondientes al índice y al intervalo seleccionados
6. El Medidor finaliza sección de mediciones	

Cursos Alternos:

Línea 5. Error en los datos analizados, se indica el error por tratarse de falta de datos en el intervalo

Línea 6. El Medidor puede volver al paso 3

Para los casos de uso realizados para las peticiones Web se cumple los pasos como está señalado en la fase de análisis, por lo que no es necesario especificar los detalles. Por lo cual procedemos directamente a realizar el siguiente modelo.

2.3.2. MODELO DE COMPORTAMIENTO DE OBJETOS

Mediante este modelo podemos visualizar gráficamente como los objetos interactúan a través de mensajes para realizar las tareas que darán la solución de software.

2.3.2.1. Diagramas de Interacción.

El diagrama de interacción nos permite tener una perspectiva colaborativa de los objetos funcionales que formarán parte de las clases, basados en la secuencias, damos una visión espacial del Benchmark. De este modelo surgen las interfaces de usuario.

2.3.2.1.1. Interacción de Medir Desempeño de Servidor de Aplicaciones

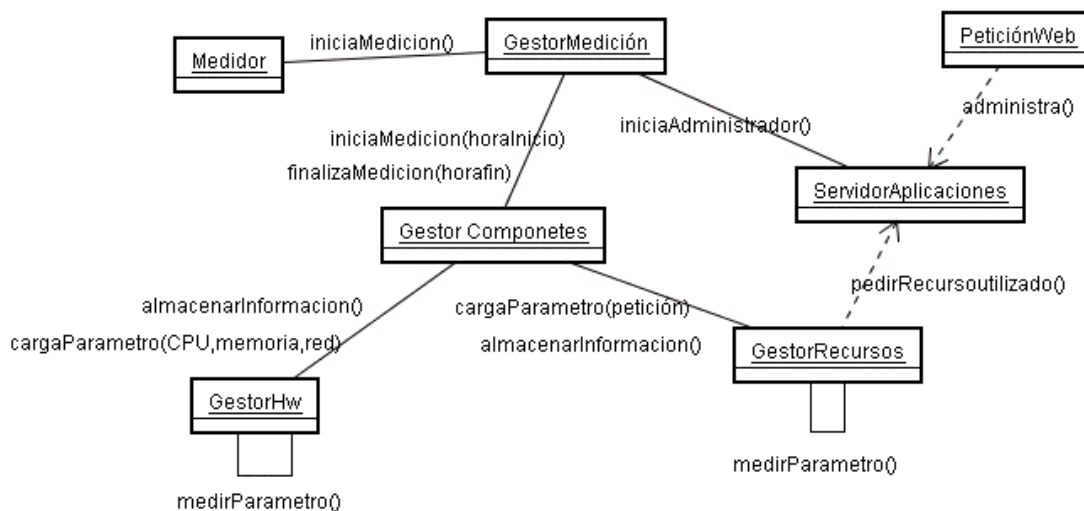


Figura 19. Diagrama de interacción Medir Desempeño de Servidor

2.3.2.1.2. Interacción de Medir Tiempo de Respuesta

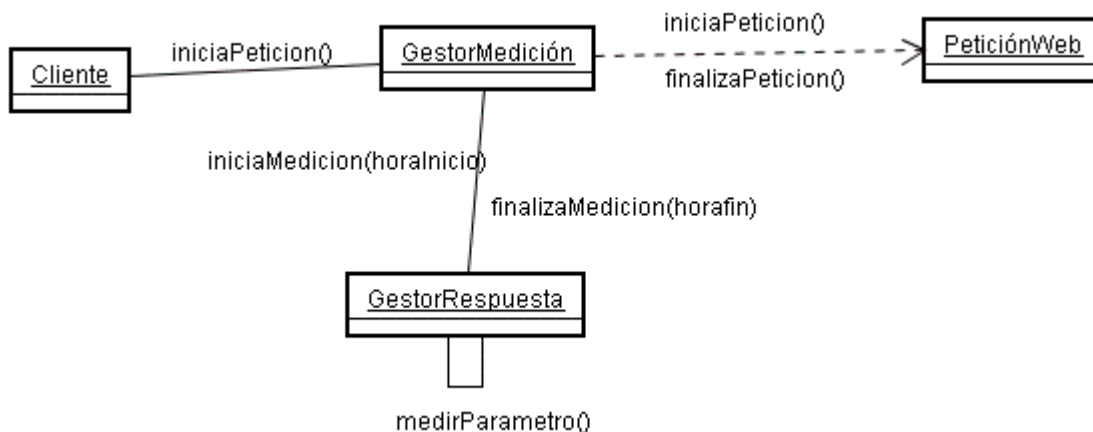


Figura 20. Diagrama de Interacción Medir Tiempo de Respuesta

2.3.2.1.3. Interacción de Verificar Mediciones Almacenadas

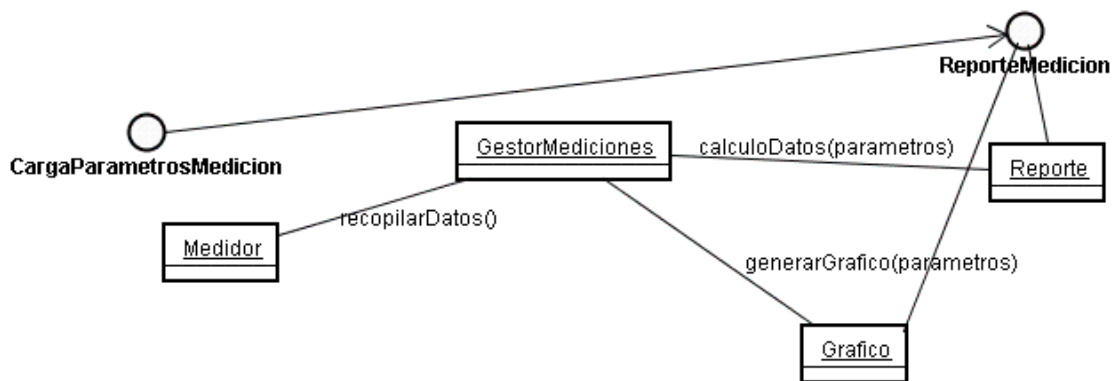


Figura 21. Diagrama de Interacción Verificar Mediciones Almacenadas

2.3.2.1.4. Interacción de Comprar Productos

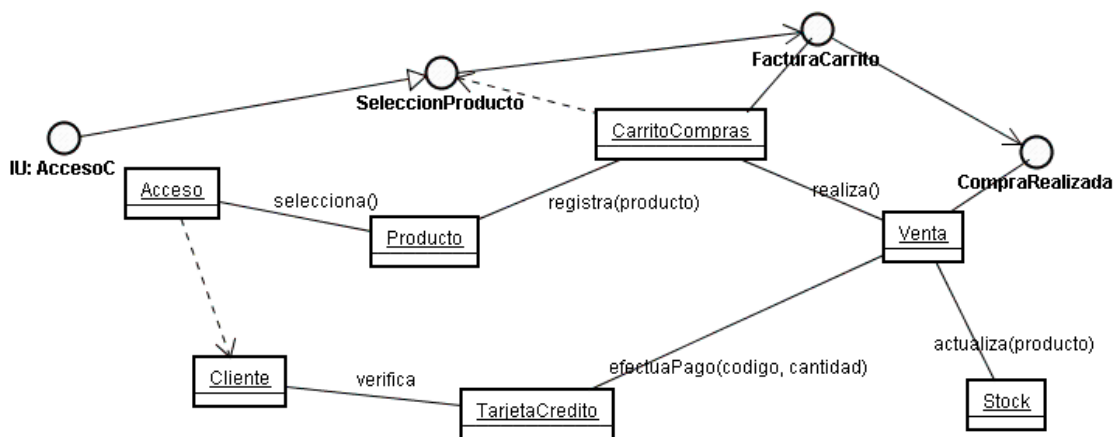


Figura 22. Diagrama de Interacción comprar Productos

2.3.2.1.5. Interacción de Registrar Datos de Cliente

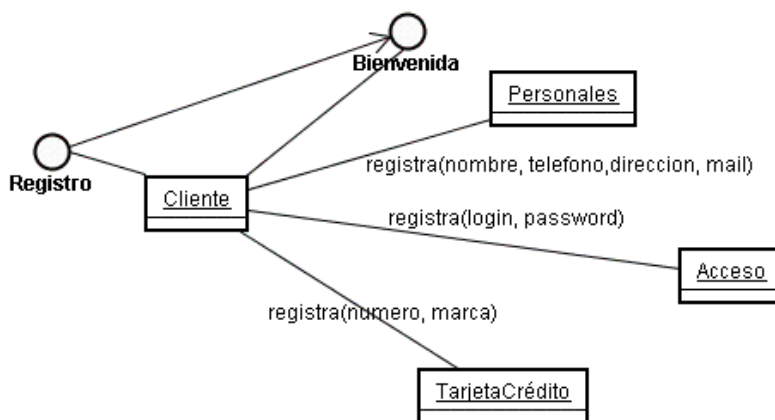


Figura 23. Diagrama de Interacción Registrar Datos De Cliente

Una vez finalizada la realización de los diagramas de interacción podemos contemplar nuevos artefactos que de ella obtenemos como las interfaces para los usuarios y también algunas de las clases necesarias para la implementación del benchmark.

Entre las interfaces que podemos apreciar se encuentran:

2.3.2.2. Interfaces de Benchmark

Tenemos 2 interfaces que necesita el usuario Medidor:

- Selección de los Parámetros de Medición
- Reporte de Medición (Benchmark)

2.3.2.3. Interfaces de Peticiones Web

Tenemos interfaces para los usuarios Clientes

- Interfaz de acceso
- Selección de Productos (Varios productos)
- Carrito de compras
- Tarjeta de crédito
- Factura de carro de compras
- Compra Realizada
- Registro de Datos Cliente
- Bienvenida a Usuario General
- Bienvenida a Usuario Registrado
- Empresa

2.3.3. MODELO DE CLASES

En esta sección se profundiza en la definición de la arquitectura de clases de la aplicación, con la finalidad de establecer de manera apropiada un nivel adecuado que permitirá

realizar la implementación, al tratarse de un modelo evolutivo se lo ha realizado por fases hasta lograr finalmente un diseño adecuado.

2.3.3.1. Definición de clases

La definición de cada una de las clases del sistema ha partido del análisis de los procesos indicados en los diagramas de interacción que nos permite visualizar el comportamiento del sistema, así como también se ha tomado en cuenta el análisis del modelo conceptual, generado en la fase de análisis para obtener de ella información acerca de muchas entidades, específicamente de los atributos que se manejará en las clases.

A continuación se especifica un listado de las clases que se ha tomado en cuenta las cuales se establecerán en un Diagrama de clases UML:

- Producto
- Categoría de Productos
- Cliente
- Login
- Carro de Compras
- Factura
- Detalle de Factura
- Conexión a la Base de Datos
- Parámetro de Medición (Cpu, Memoria, red)
- Gestor de Medición
- Gestor de Respuesta
- Gestor de Componente
- Gestor de Recursos
- Reportes de Medidas

- Gráfico de Reporte

La aplicación de las clases representa la lógica del negocio, lo cual implica que ellas son implementadas a través de EJBs, por lo cual se crean también clases propias de los EJBs dependiendo de que tipo sea, como por ejemplo EJBs de sesión o EJBs de entidad de los que se conectan a la Base de datos, locales o remotos. Así también las validaciones de la presentación son implementados a través de Servlets y de las propias JSPs

2.3.3.2. Atributos

Para especificar los atributos de las clases se debe tomar en cuenta la funcionalidad de cada clase, los atributos provienen del análisis del modelo conceptual así como también de diagrama de interacción.

Los atributos identifican las cualidades de las clases, las cuales nos permiten manejar la información almacenada o generada con los métodos.

2.3.3.3. Métodos.

Los métodos son la parte medular del funcionamiento del sistema, en este se especifican claramente los procesos que se seguirán en cada una de las clases para poder lograr los objetivos propuestos para el Benchmark.

Los métodos son obtenidos del Diagrama de interacción y de la lógica del funcionamiento del Benchmark como por ejemplo la estructuración de obtención de resultados, gráficos, condiciones, índices y parámetros, así también el procedimiento del portal Web.

2.3.3.4. Relaciones.

Las relaciones especifican los diálogos realizados entre clases las cuales permiten generación de instancias, creación y alteración de procesos y manipulación de información las cuales son de gran importancia para entender la funcionalidad de la arquitectura del Benchmark

El detalle de atributos, métodos y relaciones en cada una de las clases son expuestas en el siguiente Diagrama el cual está fundamentado en la especificación UML.

2.3.3.5: Diagrama de clases

Diagrama que nos indica las clases utilizadas por el Benchmark en las dos partes diferenciadas: Mediciones y Portal Web.

2.3.3.5.1. Diagrama de clases Portal Web

Clases correspondientes a las clases a partir de los casos de uso de Peticiones Web

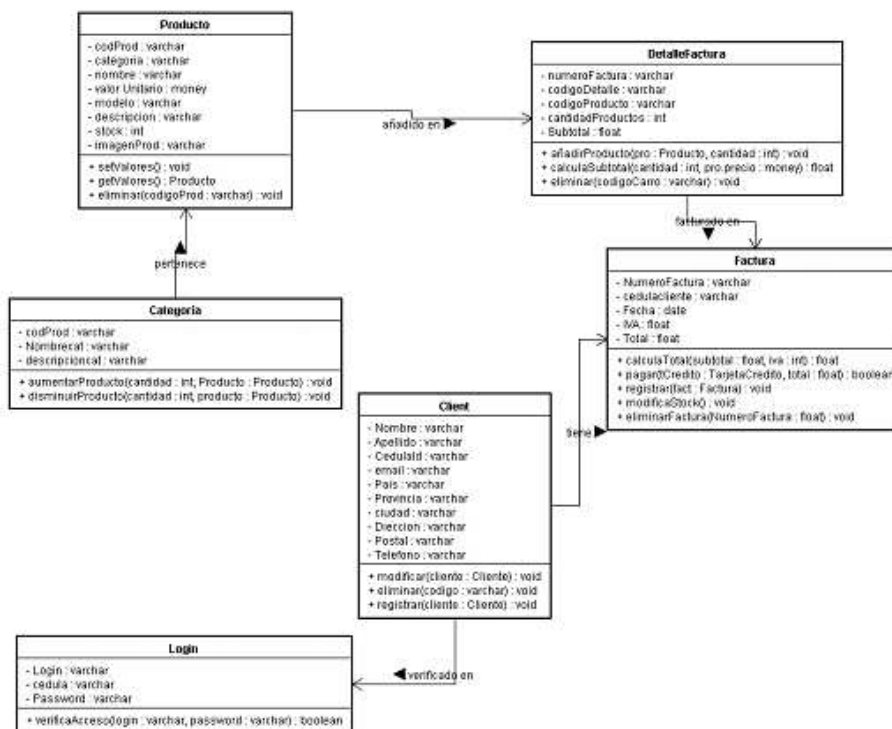


Figura 24. Diagrama de clases del Portal Web

2.3.3.5.2. Diagrama de clases de Mediciones del Benchmark

Clases correspondientes a las clases a partir de los casos de uso de Peticiones Web

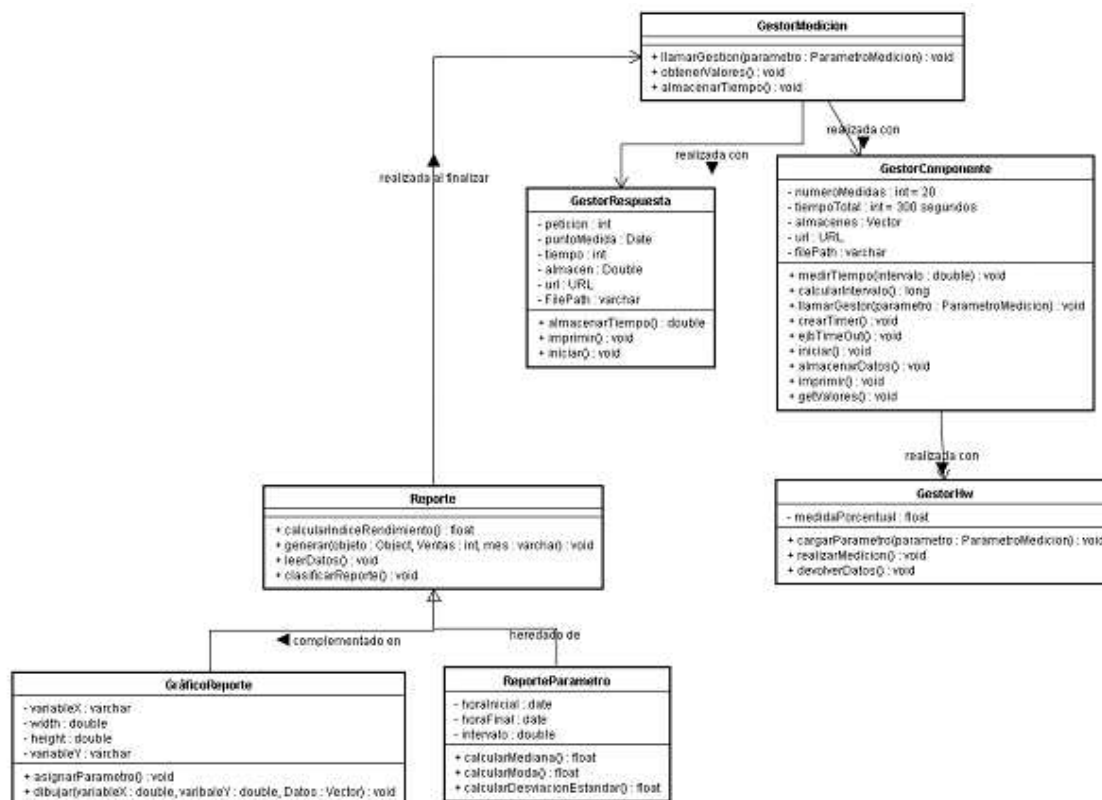


Figura 25. Diagramas de clases de Mediciones del Benchmark

2.3.4. MODELO DE LA ARQUITECTURA

Definir la arquitectura del sistema es un proceso que se realiza normalmente tras un análisis de los artefactos, los cuales nos indicarán de que manera estos se establecerán en el diseño de la arquitectura.

El caso del Benchmark, nos permite establecer las capas que la especificación de la arquitectura J2EE define, por lo cual, se utilizarán 4 capas: de presentación, de lógica del negocio, de Servidor Web y de almacenamiento, a demás agregamos a esta arquitectura la capa de mediciones.

La presentación es la cara con la que el usuario interactúa las cuales son definidas según los diagramas de navegabilidad.

La capa de Servidor Web es la que se encarga de la relación de la lógica del negocio con la presentación

La lógica del negocio contempla todas las clases que se han obtenido para el diagrama de clases las cuales son agrupadas en paquetes, estos paquetes son representados según la funcionalidad que dentro del sistema se desarrolle, se utiliza ejbs de sesión quienes contienen dicha lógica que modifica al negocio.

La capa de almacenamiento es contemplada en la Base de Datos y documentos que almacenan la información generada por el sistema y por el usuario, así como las clases que comunican al sistema con dicha base.

La capa de presentación pueden ser utilizadas bajo un solo tipo de componentes Web como son los archivos JSP, así como también el uso de Servlets para la capa de servidor Web, para la implementación del Benchmark se utiliza ambos aspectos.

Una **capa exterior o capa de Mediciones** que se ocupa de la generación de Reportes de las medidas utiliza también para su presentación JSP, adherido internamente un componente **Applet** de acuerdo al Reporte Gráfico que se requiera

2.3.4.1. Diagrama de Paquetes, visión del sistema.

Para tener una visión general de la arquitectura del benchmark necesitamos agrupar en paquetes las clases definidas en el Modelo de Clases lo que nos permitirá realizar la implementación.

Es posible establecer paquetes que engloban la funcionalidad del sistema, los cuales podemos separarlos en las diferentes capas como se muestra en la siguiente tabla:

CAPA	PAQUETE	DESCRIPCIÓN
Lógica del Negocio de la Empresa	com.empresa.bdd	Contiene clases remotas de comunicación del sistema con la Base de Datos
	com.empresa.entidades	Contiene las clases tipo entidad locales con la estructura de las tablas de la base de datos.

	com.empresa.sesion	Contiene las clases remotas que prestan persistencia a las entidades para comunicarse con la base de datos manteniendo la estructura de las tablas de la base de datos.
	com.empresa.resources	Contiene las clases de recursos que necesita el portal para su funcionamiento
Lógica del Negocio del Benchmark	com.ebench.conexion	Contiene las clases remotas que se encargan de la interacción con cliente y capa de Presentación para medir valores
	com.ebench.sesion	Contiene las clases que realizan las funciones de almacenamiento y control de la medición de los parámetros
	componentes	Contiene las Clases que se encargan de la interacción directa con los componentes Hw del Sistema operativo y de la Máquina Virtual de Java para que sean medidos los parámetros, y obtinene valores en tiempo real
Presentación	Web	Contiene todas las clases de la presentación que mantiene la interacción con el usuario final, incluyen la presentación para el usuario Medidor, quien las utiliza para obtener los reportes de las mediciones.

Reporte Gráfico	Reporte	Contiene las clases que generan los gráficos de los datos almacenados en las mediciones.
-----------------	---------	--

Tabla 7. Descripción de los paquetes utilizados en el desarrollo del Benchmark

2.3.4.2. Navegabilidad

El proyecto de Benchmark es muy sencillo con respecto a las interfaces utilizadas, debido a que este no es un proyecto que se especializa en mostrar al usuario interfaces con gran complejidad, sino más bien, simples interfaces que permitirán navegar al usuario en un común portal Web. Con respecto a las mediciones del Benchmark las interfaces básicas se limitan a mostrar los datos necesarios para que puedan analizarse.

En el modelo de Comportamiento de Objetos hemos obtenido varias interfaces según las acciones realizadas por los usuarios del sistema, las cuales se mantienen siguiendo un esquema de navegabilidad consecuente al análisis de la lógica del Benchmark.

2.3.4.2.1. Navegabilidad del Benchmark

Correspondiente al Benchmark podemos establecer el siguiente diagrama que muestra la interacción de las interfaces con el usuario Medidor.

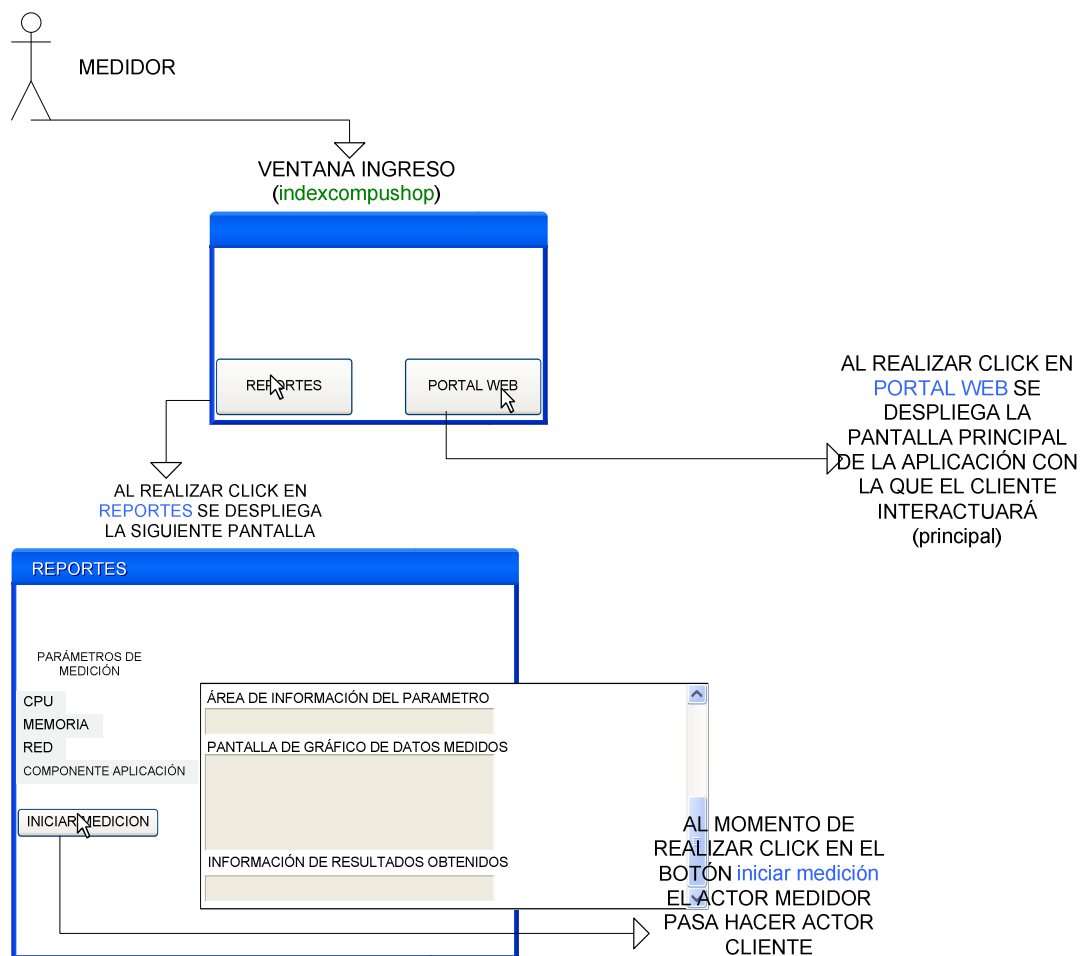


Figura 26. Navegabilidad del Mediciones del Benchmark

2.3.4.2.2. Navegabilidad de Portal Web

Correspondiente al Portal Web podemos establecer el siguiente diagrama que muestra la interacción de las interfaces con el usuario Cliente y Administrador.



Figura 27. Navegabilidad del Portal Web

2.3.4.2.2.1. Descripción De Navegabilidad

Para el actor cliente la página principal de la aplicación está constituida de varios elementos como (menús, enlaces, etc.) en las que el actor al interactuar con estos tendrá acceso a varias pantallas en las que mostrará información pertinente a usuarios, productos, información del sistema y mucho más.

A continuación se especificará cada una de las pantallas que se encuentran en el sistema y a las cuales el usuario tiene acceso, las especificaremos de acuerdo al diagrama anteriormente realizado.

2.3.4.2.2.2. Sección De Interacción E Información

Esta sección de la pantalla principal del sistema es un sitio en donde todas las pantallas pertenecientes al área del portal web serán mostradas evitando así, un uso indebido de apertura de páginas nuevas por lo que provee una mejor comunicación del sistema con el cliente.

La página por defecto que se despliega en esta sección es la página de información del sitio web para conocimiento del visitante y se detallará.

2.3.4.2.2.3. Página De Información De La Empresa

Es esta se especifica información de la empresa como su misión y visión descrita por sus desarrolladores, además en esta página se puede encontrar dos enlaces gráficos que son Registro de Usuarios y Producto en Promoción.

Registro de Clientes

Es la página que permite al usuario ingresar sus datos y registrarse lo que le da la posibilidad de poder adquirir los productos que se expenden.

Producto en Promoción

Es la página que permite desplegar información acerca de un producto que se encuentre en promoción para conocimiento del usuario.

2.3.4.2.2.4. Sección Sesión De Usuario

Esta sección de la pantalla principal del sistema contiene elementos de información requeridos para el ingreso del usuario como son login y password , si los datos ingresados son correctos y se presiona el botón ingresar al usuario se le desplegará la página de bienvenida al sistema.

Por otro lado si los datos ingresados no son correctos o si se presiona el botón ingresar al cliente se le despliega la página de registro de usuario.

El enlace Regístrese Ahora!!! es la forma más rápida que tiene el visitante de poder registrarse en el sistema y poder adquirir los diferentes productos que se expenden.

Y por último Cerrar Sesión permite al usuario terminar su interacción con el sistema.

2.3.4.2.2.5. Sección Menú Principal

Esta sección de la pantalla principal del sistema es la más importante ya que contiene todos los enlaces posibles que maneja el sistema y a continuación se los detalla.

Inicio

Esta opción permite desplegar la página correspondiente a Información del Sistema en la cual se detallo anteriormente.

Productos

Esta opción desplegable contiene los tipos de productos que el sistema ofrece para su venta al usuario y existen tres tipos que son: Laptops, Mainboards e Impresoras cada una con su página correspondiente.

Laptops.- En esta página se despliega toda la gama de laptops que el usuario puede adquirir, además posee información detallada de cada producto como es nombre, descripción, stock y precio.

Mainboards.- En esta página se despliega toda la gama de mainboards que el usuario puede adquirir, además posee información detallada de cada producto como es nombre, descripción, stock y precio.

Impresoras.- En esta página se despliega toda la gama de impresoras que el usuario puede adquirir, además posee información detallada de cada producto como su nombre, descripción, stock y precio.

Si el usuario se encuentra registrado y si el usuario escogió algún producto o productos y presiono el botón agregar aparece una pantalla de Carro de Compras.

Carro de Compras

En esta página se indica de forma explícita el producto o productos adquiridos, esta brinda al usuario diferentes opciones que pueden ser Recalcular, Eliminar y Comprar todas estas opciones están sobreentendidas por lo que al marcar la opción de elección de un producto en forma individual se podrá ejecutar la opción eliminar o si en el texto de cantidad la cambia se puede recalcular el valor cada una de estas , para el caso de comprar al presionar este botón se despliega la página Pago de Productos.

Pago de Productos

En esta página se hace un registro de la forma de pago de los productos adquiridos, cuando se tiene todo ingresado y al pulsar en el botón pagar se despliega la pantalla Gracias.

Gracias

La página gracias da un aviso de confirmación al cliente por la compra realizada indicándole el valor real a pagar además hay un botón imprimir factura que al presionarlo se puede ver de forma explícita los productos.

Factura

Esta página al cargarse brinda la opción de imprimir para que el usuario tenga en respaldo el detalle de la factura realizada.

Si el usuario no se encuentra registrado y escogió algún producto o productos y presiona el botón comprar directamente es enviado a la página Registro de Clientes.

Promociones

Es la página que permite desplegar información acerca de un producto que se encuentre en promoción para conocimiento del usuario.

La Empresa

En esta página se describe una pequeña presentación de las actividades que realiza la empresa para beneficio de sus usuarios.

Servicios en Línea

Esta opción desplegable contiene dos submenús que son Registrarse y Carrito de Compras de las cuales estas son sitios de referencia para que un usuario acceda al registro de sus datos o visualizar como su carro de compras toma forma de acuerdo a los productos adquiridos esto es cuando el usuario haya iniciado su sesión.

Contáctenos

Es la opción que permite al usuario enviar un correo electrónico a la empresa para conocer alguna novedad referente a la compra realizada.

2.3.4.2.2.6. Sección Menú Opciones

Esta sección de la pantalla principal del sistema contiene enlaces importantes que son de referencia rápida al cliente como son las páginas: carrito de compras y regístrese aquí que previamente ya se las indico, la otra página de referencia se la detalla.

2.3.4.2.2.7. Sugerencias y Comentarios

En esta página se le brinda al usuario el poder enviar alguna sugerencia o comentario para poder ser tomada en cuenta por sus desarrolladores.

2.4. IMPLEMENTACIÓN DEL BENCHMARK.

La fase de implementación corresponde a la construcción dentro del marco de ciclo de vida RUP, lo cual nos indica la realización del software las herramientas que se utilizarán basándose en los modelos y diagramas realizados en las fases anteriores de análisis y diseño, por lo cual este capítulo está compuesto de las siguientes partes:

2.4.1. DESCRIPCIÓN DE LAS HERRAMIENTAS PARA LA CONFECCIÓN DEL BENCHMARK

La descripción de las herramientas es un proceso necesario ya que nos permitirá establecer un ambiente adecuado en el cual se ha fundamentado la construcción del Benchmark.

Para la construcción del Benchmark fijado en los objetivos se necesita establecer las versiones de los Servidores de Aplicaciones, la plataforma para el desarrollo, y la base de datos; cada una representa las capas utilizadas para el desarrollo revisadas anteriormente en la fase de diseño

2.4.1.1. Plataforma de desarrollo Netbeans

Debido a su gran amplitud, aceptación, y la integración de las herramientas como los servidores de aplicación que son la base de nuestro análisis, así como también su libre distribución se ha seleccionado esta herramienta para la construcción del benchmark, por lo cual los procesos revisados en la fase de Diseño se acoplan en esta plataforma.

La versión de Netbeans que integra los módulos necesarios para el acoplamiento de Glassfish y de JBoss es la 6.5 la cual soporta las siguientes tecnologías:

- Java SE
- JavaFX
- Java Web y EE
- Java ME
- Glassfish V2 UR2
- Glassfish V3 Prelude
- Apache Tomcat 6.0.18

- Jboss 4.0

Esta herramienta puede ser obtenida directamente de la página Web propia de Netbeans:

<http://www.netbeans.org/downloads/index.html>

2.4.1.2. Servidor de Aplicaciones JBOSS

El Servidor de Aplicaciones JBOSS que se utiliza para realizar la comparación, debido a su acoplamiento con la plataforma de desarrollo y su gran uso en la actualidad se ha seleccionado su versión 4.0.4,GA

JBOSS es factible obtenerlo gratis de: <http://jboss.org/jbossas>

2.4.1.3. Servidor de Aplicaciones GLASSFISH

El servidor de aplicaciones Glassfish en la actualidad se encuentra en producción la versión 2 y la 3 en desarrollo. De la página de Netbeans se puede obtener directamente la versión 2 acoplada con la plataforma de desarrollo.

2.4.1.4. Base de Datos Oracle

Debido a su gran uso y amplitud de funciones, así como su facilidad de administración se ha seleccionado la base de datos Oracle en su versión 9i.

2.4.2. CONSTRUCCIÓN BASADA EN J2EE

2.4.2.1. J2EE

Habíamos analizado que la plataforma J2EE permite crear aplicaciones empresariales, utilizando un modelo multicapas, dividiendo a la aplicación en diferentes niveles, cada uno basándose en una tarea en particular. Podemos diferenciar 2 partes importantes en la arquitectura inicial de J2EE, una interna y otra externa; la parte externa está compuesta por los JavaBeans, Applets, SQL, etc, mientras que la parte más interna tenemos un Contenedor que maneja los EJB, JSP y servlets, también transacciones, mensajes y servicio de mail, todos estos administrados a través del Servidor de Aplicaciones.

De modo resumido podemos definir:

- EJB: módulo encargado de manejar la lógica de la programación
- JSP: maneja las interfaces de comunicación con el usuario

- Servlets: Programas que permiten la comunicación y ejecución de programas a través de una Aplicación Web
- SQL: maneja la comunicación con el Gestor de la Base de Datos.

Tomando en cuenta que el Benchmark medirá el rendimiento de una aplicación J2EE, básicamente el Portal Web debe mantener la estructura que utiliza todos estos componentes, para lograr capturar un comportamiento verdadero.^[1]

2.4.2.2. Bibliotecas y librerías a utilizar

Para la realización del Benchmark son necesarias varias librerías externas que permiten realizar actividades necesarias para el correcto funcionamiento del Software. Estas librerías son:

- Jdbc: permite la conexión con la Base de datos Oracle 9i, la versión es ojdbc1.4 para Oracle, este se lo puede obtener de la página:
http://www.oracle.com/technology/software/tech/java/sqlj_jdbc/htdocs/jdbc9201.html
- Sigar: Permite realizar las mediciones de los recursos generales del Hardware, ya que realiza la interacción directa con el Sistema Operativo, esta herramienta fue originalmente realizada en lenguaje C, pero para efectos de interacción tiene módulos configurados para que funcione en lenguaje JAVA. Pertenece al software de Hyieric Sigar y se lo puede bajar de la página:
<http://support.hyeric.com/display/SIGAR/Home;jsessionid=F9319CDB8079CE7E1FC9BCF1FEB1EA83#Home-download>

Así mismo es necesario de componentes internos para la funcionalidad del sistema:

- EJBTimer: permite controlar acciones bajo intervalos de tiempo en la utilización de los EJB.
- Pool de conexiones: Permite realizar la conexión de la base de datos para que pueda ser directamente administrada por el Servidor de Aplicaciones

^[1] ABARCA. C, Miguel, Donoso, Gustavo. Manual de Desarrollo Básico de Aplicaciones En la Plataforma J2EE En Ubuntu 7.x.

- Data Resource: Recupera los datos de la base de datos y los administra en un contenedor temporal de sesión.

2.4.3. PRUEBAS DEL BENCHMARK

Se realizan pruebas de funcionamiento del Benchmark, de cada una de los módulos, así como las conexiones a la Base de Datos.

Las pruebas más importantes se las realiza a las mediciones de los parámetros cuando el Portal es visitado por más de un usuario a la vez de tal manera que se genere las medidas correctas y en forma más real. Se las realiza con un usuario en un ambiente personal y luego en un ambiente real, en una red Lan utilizando el servicio http desde las terminales.

2.5. APLICACIÓN DEL BENCHMARK.

La aplicación del Benchmark es un proceso fundamental para lograr las metas propuestas, en este capítulo se planifica la medición de los parámetros correspondientes al Benchmark que nos dará una solución de rendimiento de Glassfish y JBoss, a demás de los detalles respecto a la instalación y configuración de los Servidores de Aplicación así como de Netbeans y la base de Datos Oracle.

2.5.1. PLANIFICACIÓN DE LAS PRUEBAS DE MEDICIÓN Y AMBIENTE

Las pruebas de medición se las realiza bajo un mismo ambiente, de manera que el servicio que ofrece el Servidor de Aplicaciones no sea diferenciado por el trabajo del Hardware o del Sistema Operativo en el cual se encuentra implementado.

Los puntos principales a considerar para la planificación de las pruebas son:

- Parámetro o índice de Medición
- Tipos de mediciones
- Número de Mediciones
- Número de Usuarios
- Intervalos de tiempo

Estos son clasificados durante todo el proceso de medición según el parámetro que se vaya a medir, del cual dependerán los otros parámetros.

2.5.1.1. Medición de los componentes Hardware

Los componentes Hardware (CPU, memoria y red), reflejarán el comportamiento del Sistema utilizando los recursos que se requiera de Hardware con la ayuda del Sistema Operativo. Las especificaciones de las mediciones dependen de las especificaciones descritas anteriormente, permitiéndonos tener una idea clara de cual de los 2 Servidores de Aplicaciones, ofrece un mejor rendimiento en su uso. Para establecer un ambiente particular y único para ambos Servidores de Aplicación, se ha establecido un conjunto específico de tareas que se realizarán por un número determinado de usuarios simultáneamente.

Tiempo Total	Intervalos	Parámetros	Número de Usuarios	Tareas	Medidas Totales
600 segundos	20 segundos	CPU	100	<ul style="list-style-type: none"> • Registro • 2 compras 	30
			1000		
			5000		
			10000		
		MEMORIA	100		30
			1000		
			5000		
			10000		
		RED	100		30
			1000		
			5000		
			10000		

Tabla 8. Descripción de los parámetros de medición para índices Hardware, cpu, memoria y red

Número total de medidas 30 por cada parámetro y por cada usuario = 340 mediciones, divididas en 3 de 120 por parámetro.

2.5.1.2. Medición de los tiempos de Respuesta

Los tiempos de respuesta se refiere a la medición del tiempo que tarda el Servidor de Aplicaciones en realizar una petición Web dentro del Portal Web, por un usuario del Sistema, para lo cual se debe especificar cuales operaciones van a ser medidas.

Las Peticiones Web que se van a medir en la utilización de ambos servidores son:

- Comprar Producto
- Registro de Cliente

Estas peticiones han sido seleccionadas debido a que representan el objetivo del portal Web, es decir que un Cliente “registrado” realice una “compra” de los productos que ofrece la empresa.

Para obtener datos que representen un significativo número de medidas tomadas se tomarán en cuenta los siguientes argumentos:

Número de Usuarios	Número de Medidas por petición			Peticiones Totales			
	Registro	Compras			M1	M2	M3
1	1	1	2	3	2	3	4
50	1	1	2	3	100	150	200
100	1	1	2	3	200	300	400
500	1	1	2	3	1000	1500	2000
1000	1	1	2	3	2000	3000	4000
5000	1	1	2	3	10000	15000	20000
10000	1	1	2	3	20000	30000	40000

Tabla 9. Descripción del Tiempo de respuesta para la medición del Portal Web

Total mediciones realizadas: 21

Para el número de medidas la tabla está dividida en las Peticiones Web Registro y Compras, en todos los casos los usuarios realizarán un solo registro, y se realizarán 3 tipos de variaciones en Compras, cada usuario realizará 3 tipos de mediciones para 1, 2 y 3 compras respectivamente, lo que nos dará los valores en las tablas correspondientes a M1, M2 y M3 que son las medidas totales en cada medición.

2.5.2. INSTALACIÓN Y CONFIGURACIÓN DE LAS HERRAMIENTAS

La instalación de las herramientas es uno de los puntos importantes para el buen funcionamiento del software por lo que es necesaria una especificación detallada de cada una de las herramientas.

2.5.2.1. Instalación Del JDK

Como paso previo a la instalación de los servidores de aplicaciones es necesario instalar la herramienta JDK útil para poder utilizar los servidores, se instalará una de las últimas versiones la `jdk-6u10-windows-i586-p.exe`.

Luego de haber instalado el JDK se pasa a configurar las variables de entorno Path y Classpath.

Para la variable Path nos dirigimos a Variables del [Sistema](#) se la ubica y hacemos click en modificar. Al final del campo llamado "valor de variable" escribimos la ruta de ubicación del compilador de JAVA, como se indica en la figura.

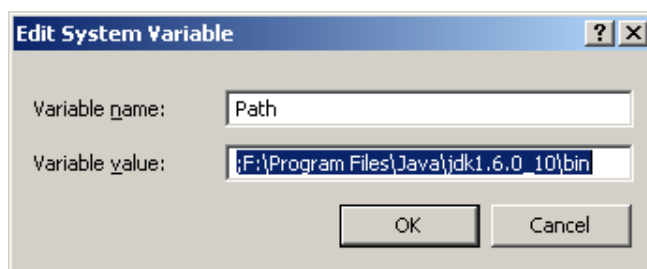


Figura 28. Edición de la variable de entorno Path

En la misma ventana de variables de entorno, creamos una variable llamada Classpath, en valor de variable se tiene que ingresar la ruta de ubicación, como se indica en la figura.

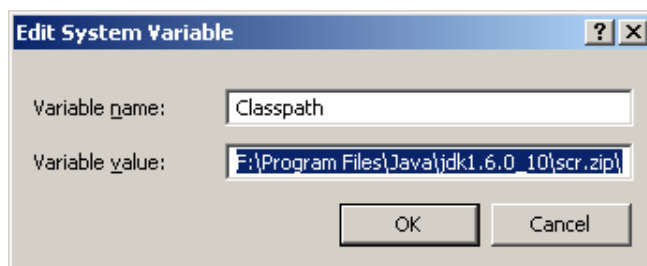


Figura 29. Creación de la variable de entorno Classpath

2.5.2.2. Instalación del Servidor de Aplicaciones Glassfish.

Se ejecuta el asistente de instalación y nos despliega la pantalla de bienvenida y presionamos el botón de personalizar para instalar los paquetes necesarios.

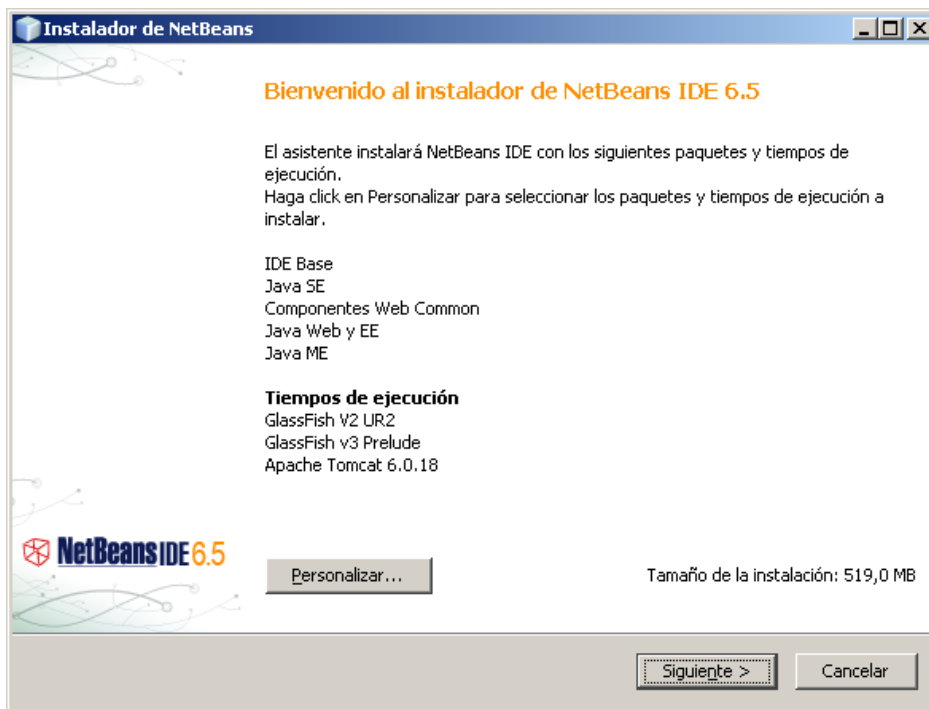


Figura 30. Pantalla de bienvenida de Netbeans 6.5

Lo personalizamos y se escoge los componentes para su instalación el más importante para este caso es el GlassFish V2 y luego presionamos en aceptar.

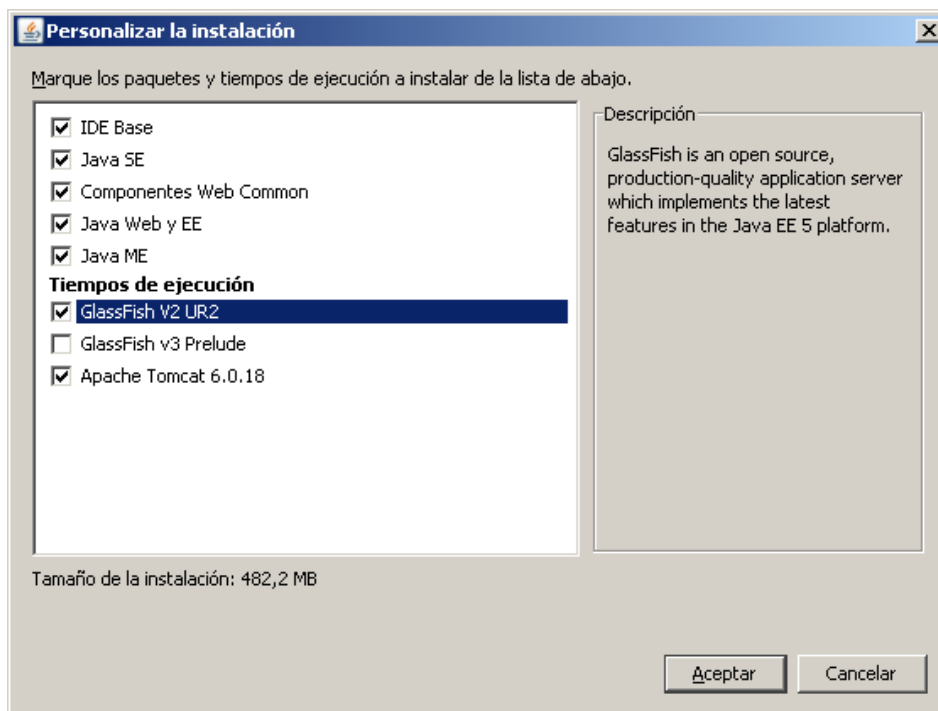


Figura 31. Pantalla de selección de componentes a instalar entre ellos el servidor de aplicaciones.

La siguiente pantalla es de la aceptación de acuerdo de licencia por lo que se acepta y se presiona en siguiente.

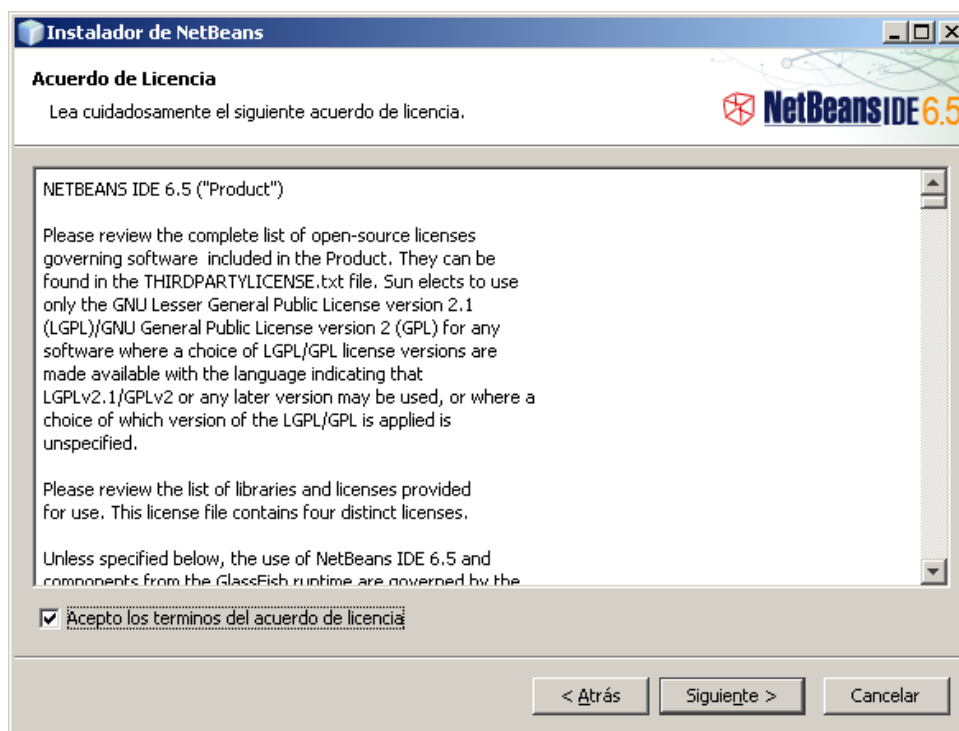


Figura 32. Pantalla de acuerdo de licencia

Luego se selecciona la ubicación de instalación del IDE NetBeans y se localiza el JDK (paso automático).

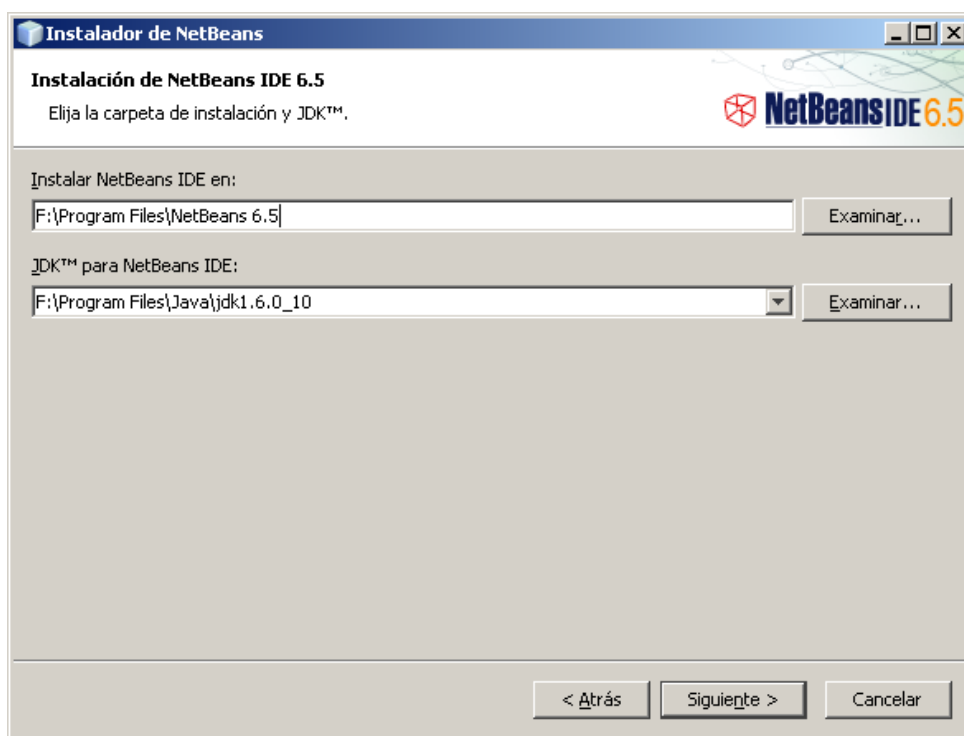


Figura 33. Pantalla de ruta de instalación de Netbeans 6.5 y del JDK.

En esta pantalla se indica la carpeta de instalación y propiedades del servidor GlassFish por lo que primero se indica la ruta de instalación del servidor, luego se indica el JDK este de manera automática ubica la ruta.

Luego las demás opciones se muestran por defecto por lo que no representa mayores dudas con la excepción de que esta información posteriormente será útil para el manejo del servidor y se recomienda tener un respaldo de esta.

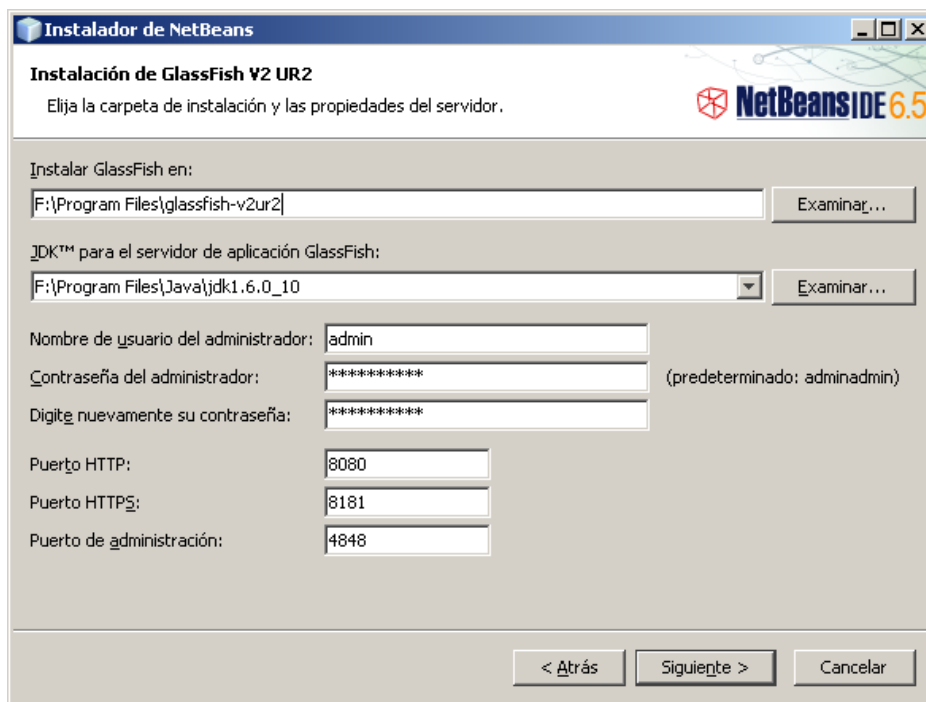


Figura 34. Pantalla de ruta de instalación del servidor Glassfish incluido características propias del servidor.

Posteriormente se procede a la instalación del último paquete seleccionado que es Apache Tomcat y se ingresa la ruta de instalación.

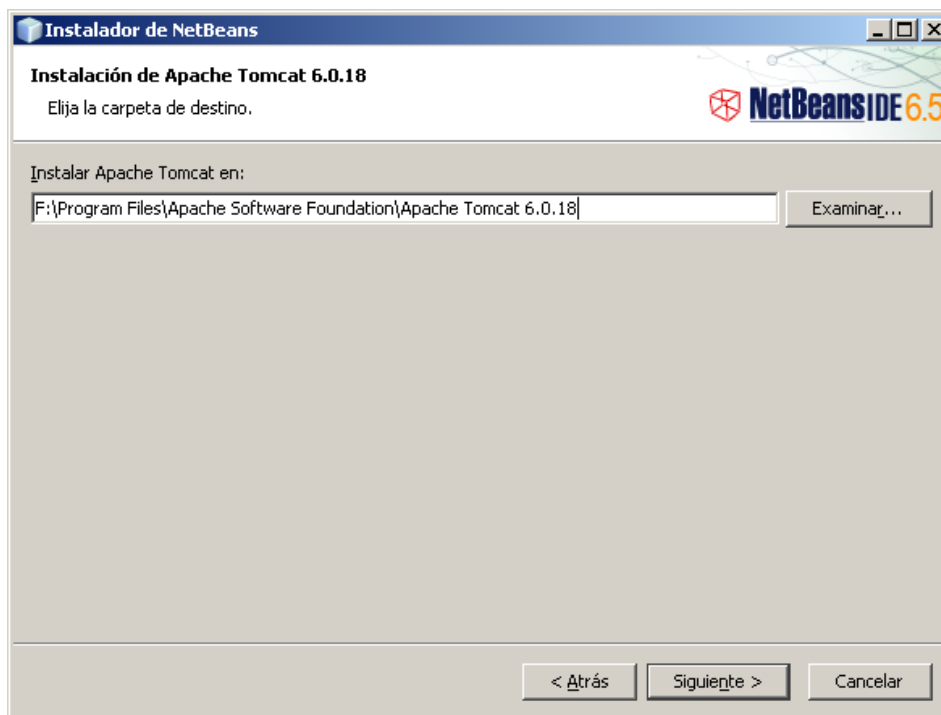


Figura 35. Pantalla de ruta de instalación del servidor Web

En esta pantalla se detalla un resumen de los componentes que se instalarán, sus respectivas rutas de instalación y un total de espacio a ocupar en disco.

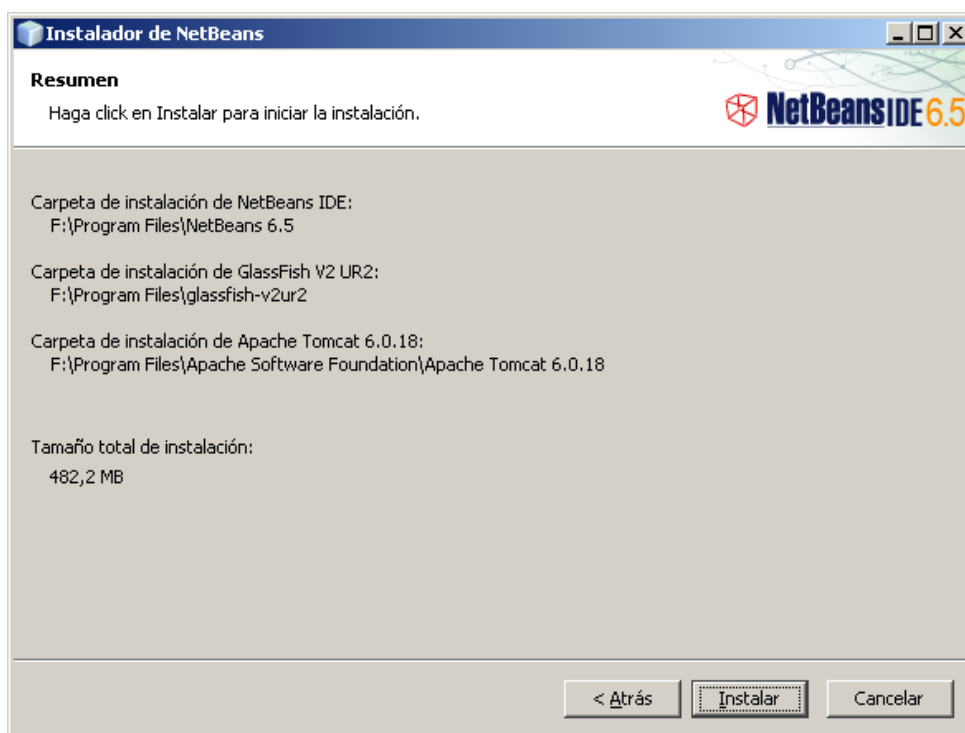


Figura 36. Pantalla de rutas resumidas de instalación de cada uno de los componentes.

En esta se detalla el proceso de instalación.

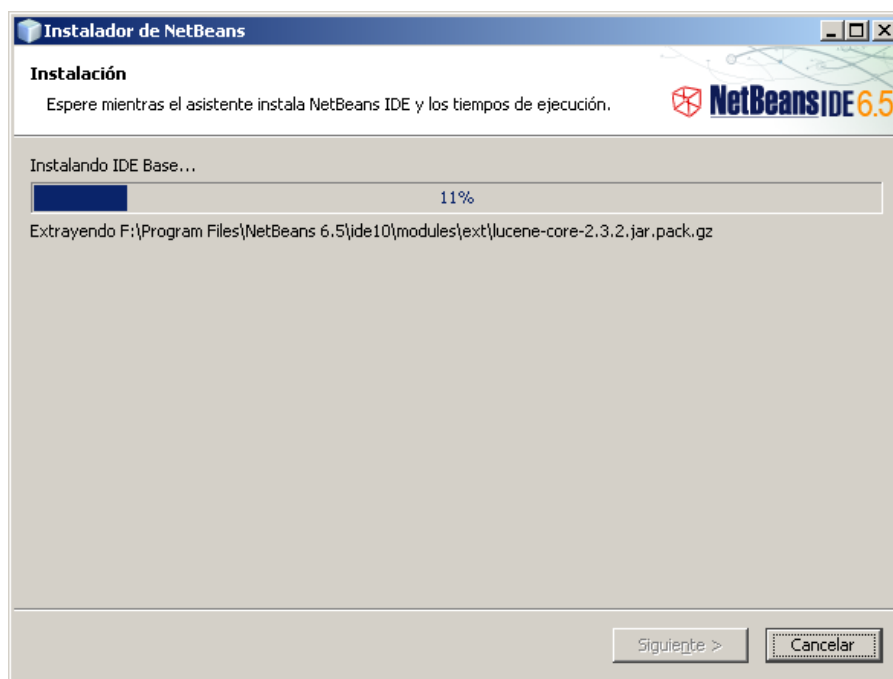


Figura 37. Pantalla de proceso de instalación.

Por último se indica que el proceso de instalación a finalizado y a manera personal se puede escoger las diferentes opciones que se detallan.

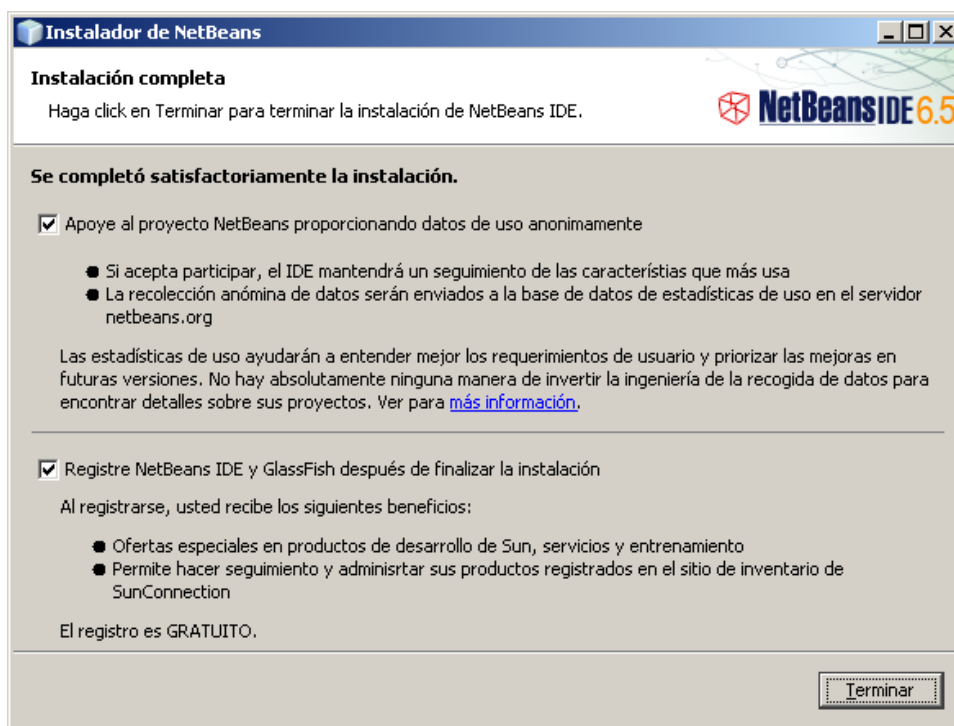


Figura 38. Pantalla de finalización de instalación.

Para poder utilizar el servidor GlassFish se deben crear dos variables de entorno para este caso son la ANT_HOME en donde se indica la ruta de la carpeta ant del servidor, como se indica en la figura.

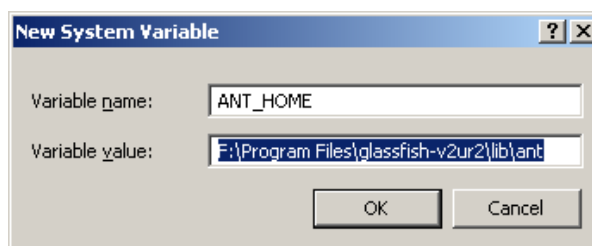


Figura 39. Pantalla de creación de variable de entorno ANT HOME.

Y la variable JAVA_HOME en donde se indica la ruta de ubicación del compilador de java, como se indica en la figura.

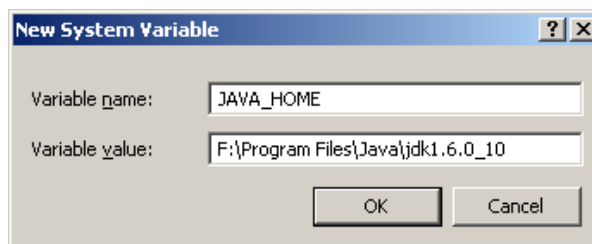


Figura 40. Pantalla de creación de variable de entorno JAVA HOME.

Hecho los pasos previos se procede a abrir el browser y colocamos <http://localhost:4848>, esta dirección es para la administración total del servidor por parte del administrador.

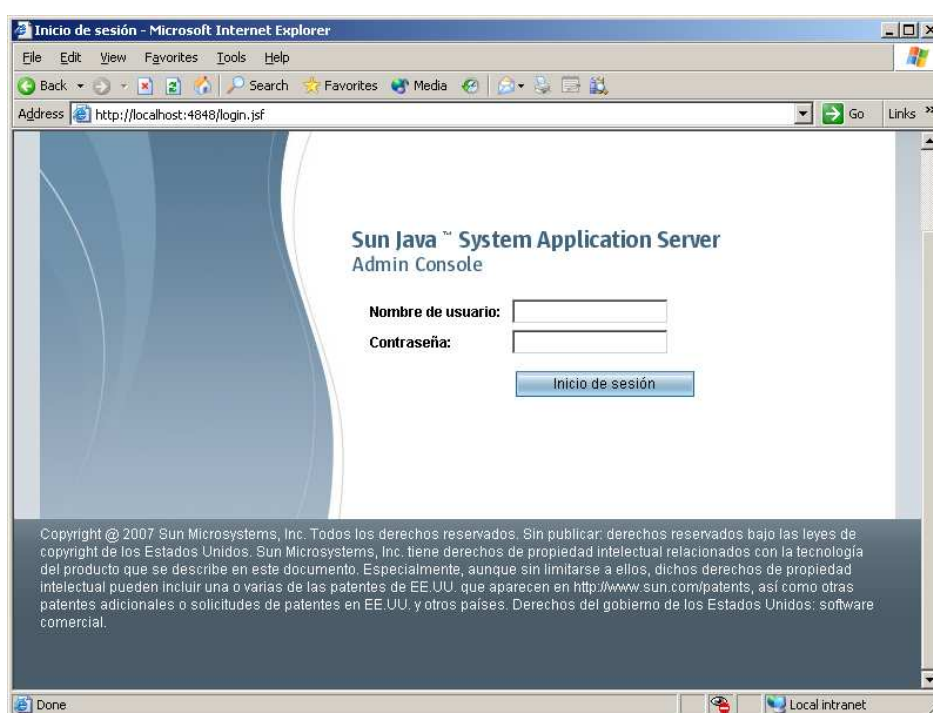


Figura 41. Pantalla de ingreso al servidor por medio de consola web

Se ingresa el nombre de usuario y contraseña por defecto son

Nombre Usuario: admin

Contraseña: adminadmin

Y nos da acceso a la pantalla principal del servidor de aplicaciones y sus diferentes opciones para el manejo de aplicaciones basadas en plataforma java.

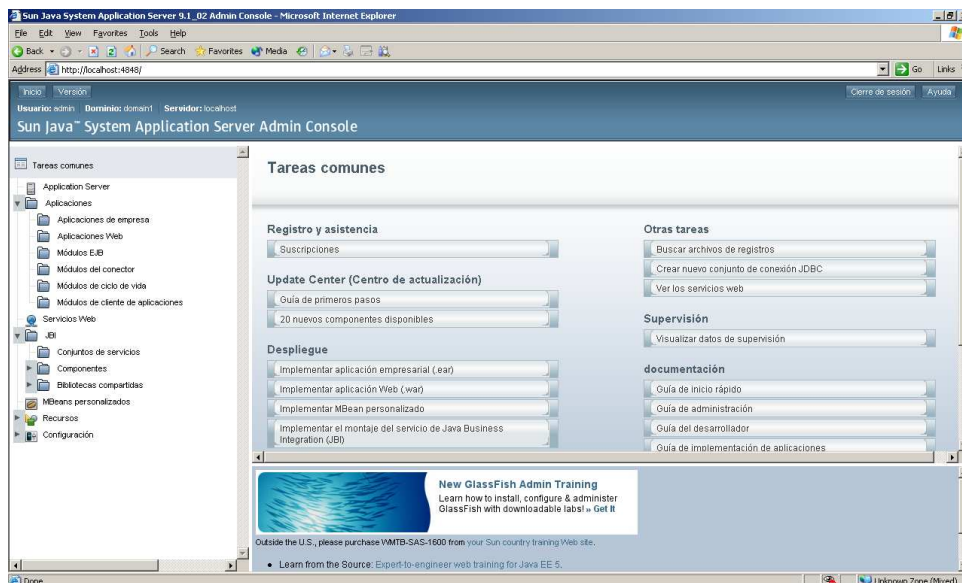


Figura 42. Pantalla de interfaz de administración del servidor.

2.5.2.3. Instalación del Servidor de Aplicaciones JBoss.

Primero descargamos el servidor de aplicaciones a instalar en este caso el JBoss versión 4.0.4. y ejecutamos el instalador con extensión .jar y comenzará el proceso de instalación del servidor.



Figura 43. Pantalla de selección de lenguaje de instalación.

Se procede a la selección del lenguaje de instalación por defecto se encuentra en ingles por lo que presionamos OK y se despliega la pantalla de inicio.

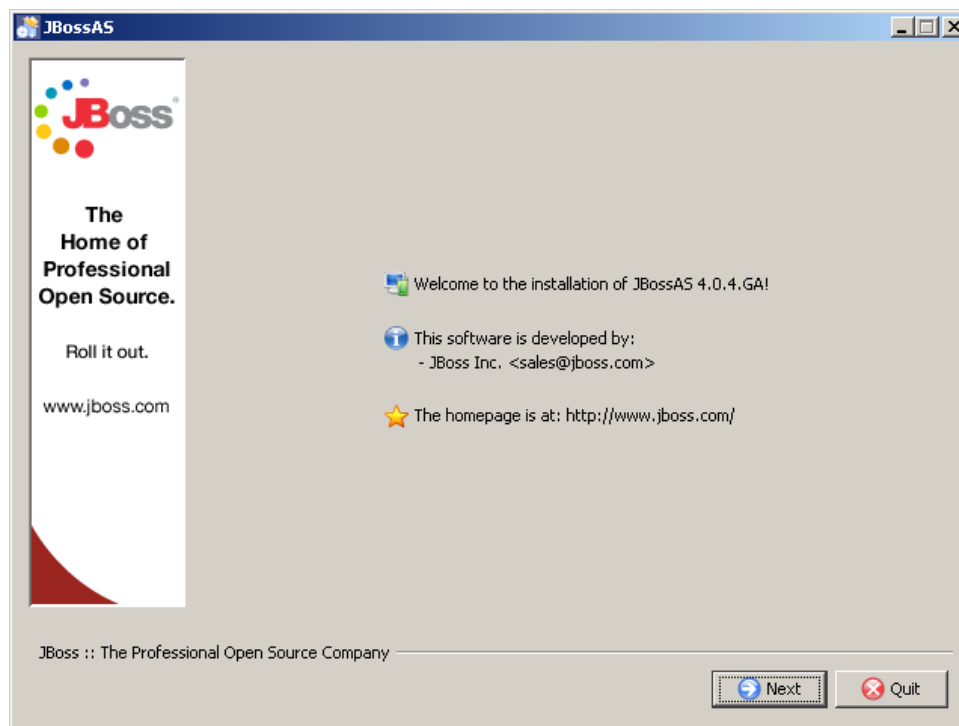


Figura 44. Pantalla de bienvenida.

Pantalla referente a datos informativos sobre el servidor de aplicaciones antes de su proceso de instalación.

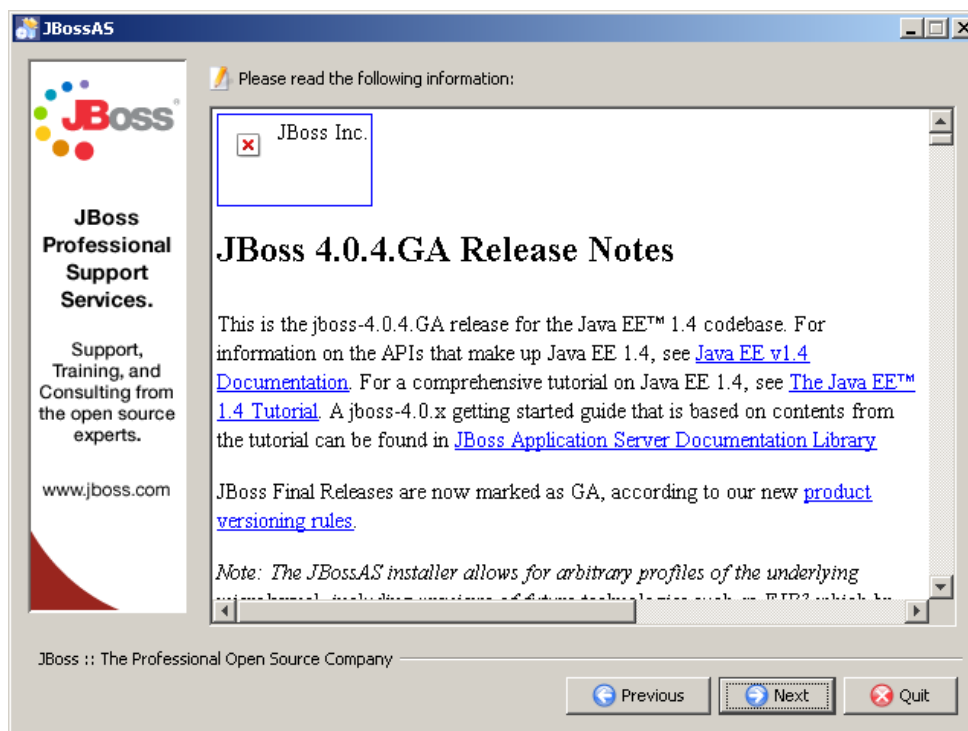


Figura 45. Pantalla de información del producto.

Luego para continuar con la instalación se debe aceptar el acuerdo de licencia GNU

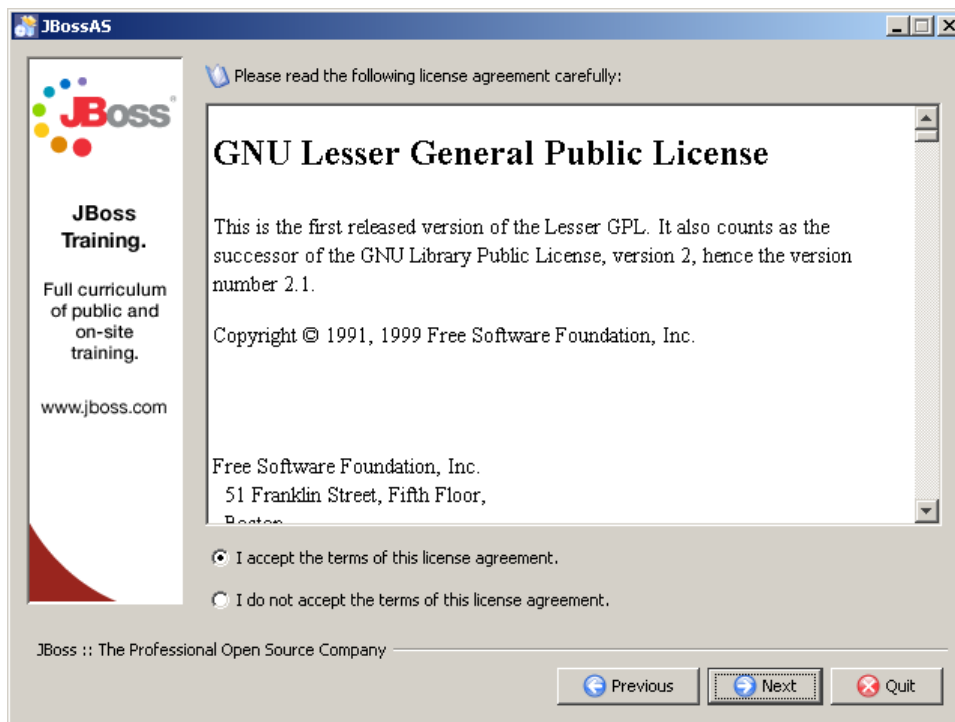


Figura 46. Pantalla de acuerdo de licencia.

En esta pantalla se selecciona la ruta de instalación para el servidor de aplicaciones.

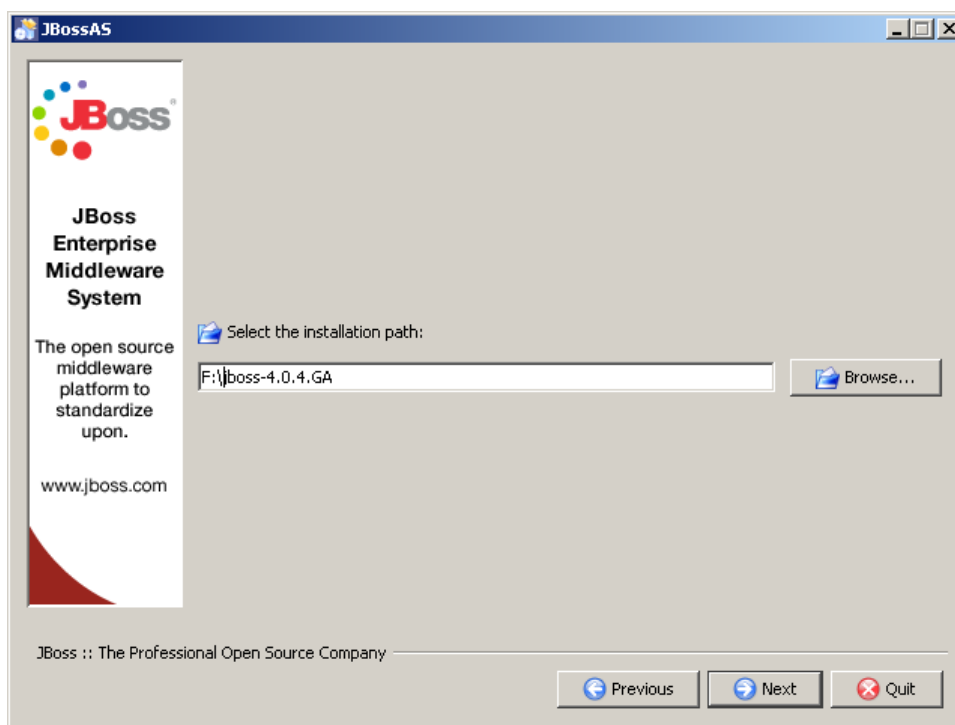


Figura 47. Pantalla de ruta de instalación del servidor.

Se muestra una pantalla donde se indica que si la carpeta en el directorio a instalar no existe lo creará.

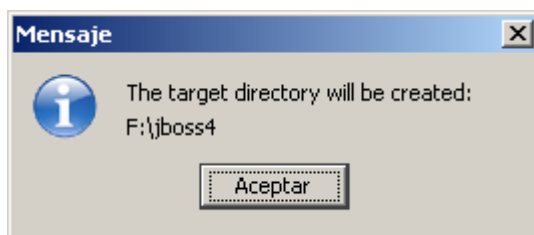


Figura 48. Pantalla de creación de folder en la ruta seleccionada.

Acto seguido se deberá definir el tipo de instalación a realizar en este caso se selecciona la instalación total y se indica el tamaño en disco de la misma.

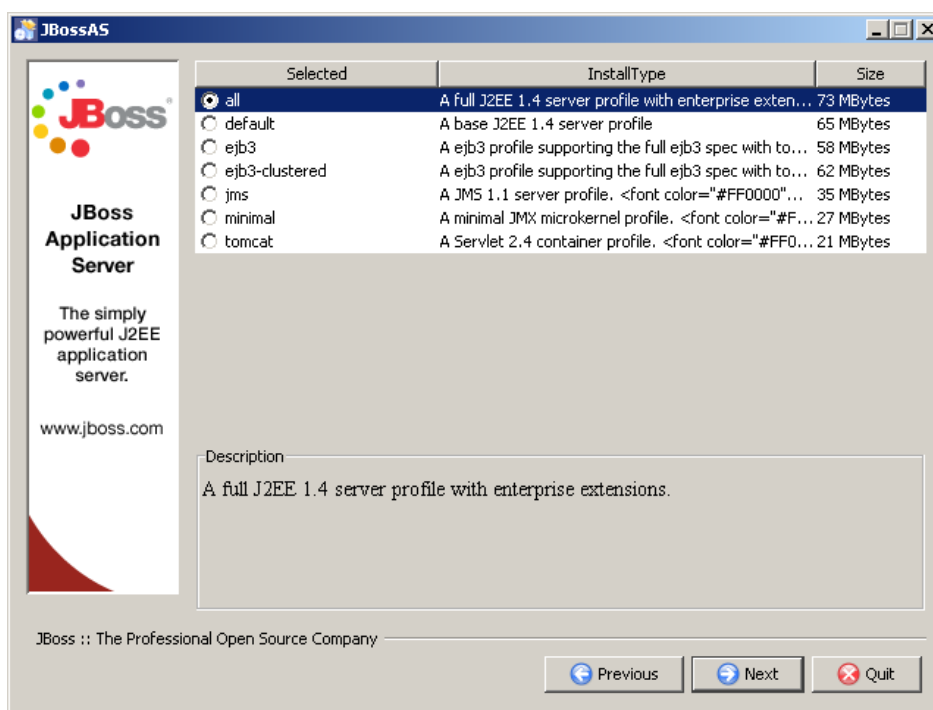


Figura 49. Pantalla de selección del tipo de instalación.

En esta pantalla se despliega una lista de paquetes que se instalarán según el tipo de instalación escogida anteriormente.

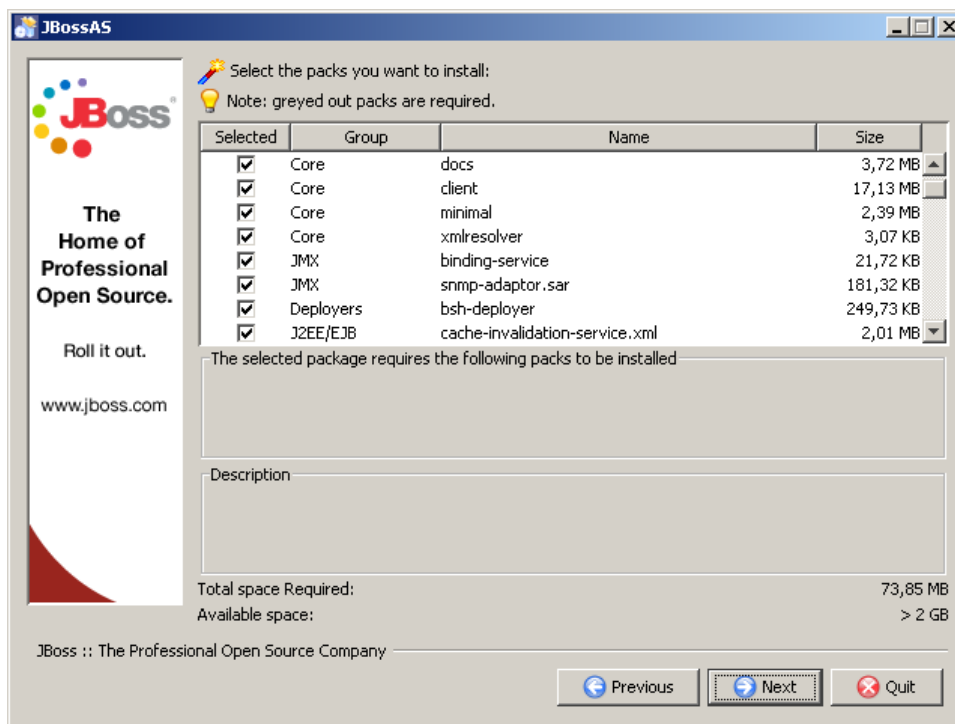


Figura 50. Pantalla de selección de paquetes a instalar.

A continuación se debe indicar el nombre de la instancia del servidor por defecto su valor es **default**.

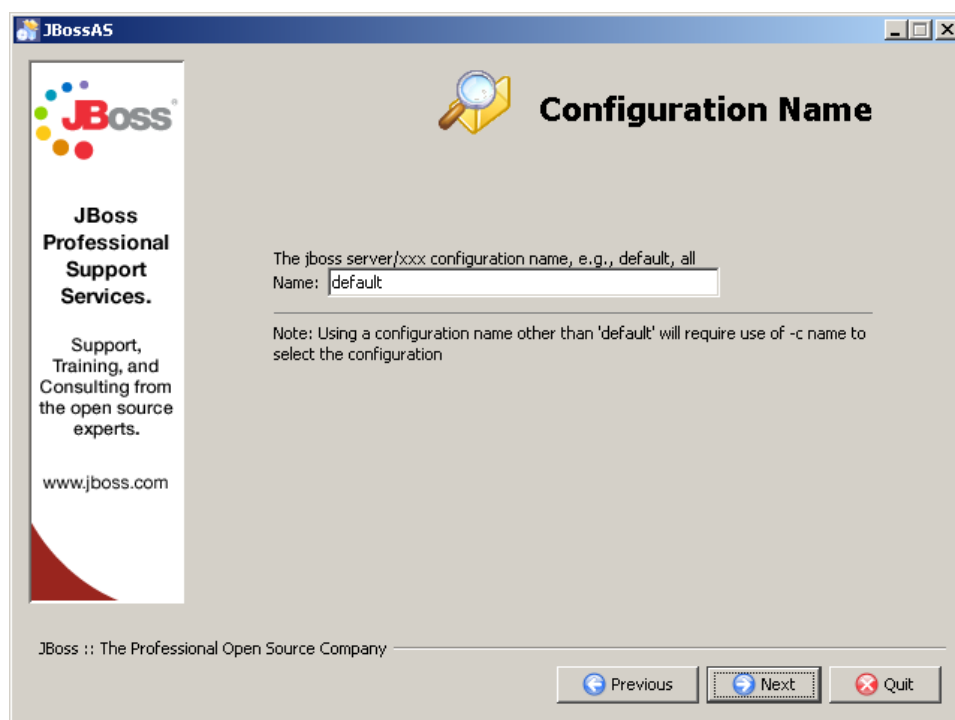


Figura 51. Pantalla del nombre de configuración del servidor.

En esta pantalla se selecciona el tipo de seguridad para el control de cualquier interfaz JMX que interactuará con el servidor, esta opción es a criterio personal si se la escoge el nombre de usuario y password para cada una de estas son especificadas, como se indica en la pantalla.

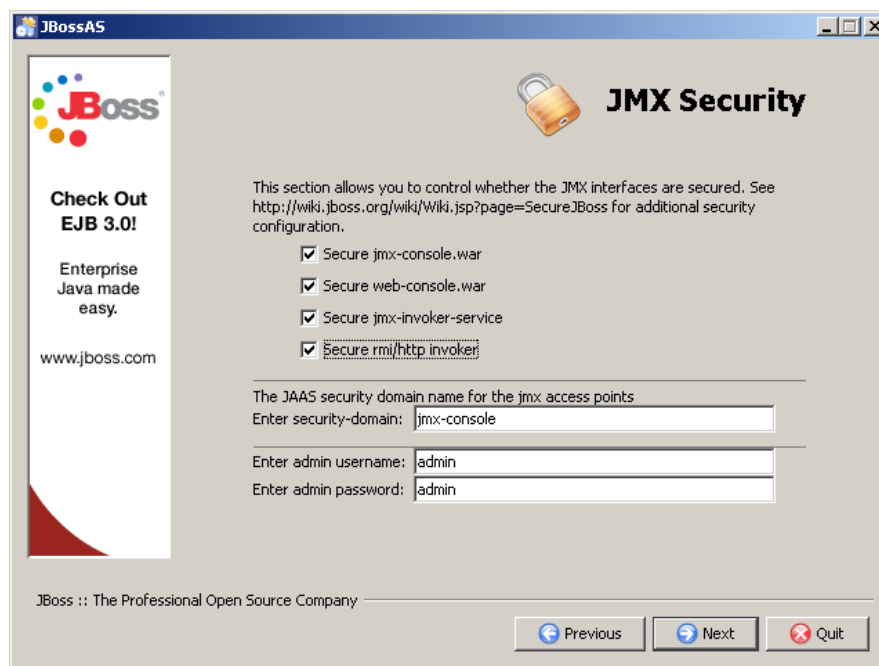


Figura 52. Pantalla de seguridad de la interfaz JMX.

En esta pantalla se indica la ruta de instalación y todos los componentes a instalar.

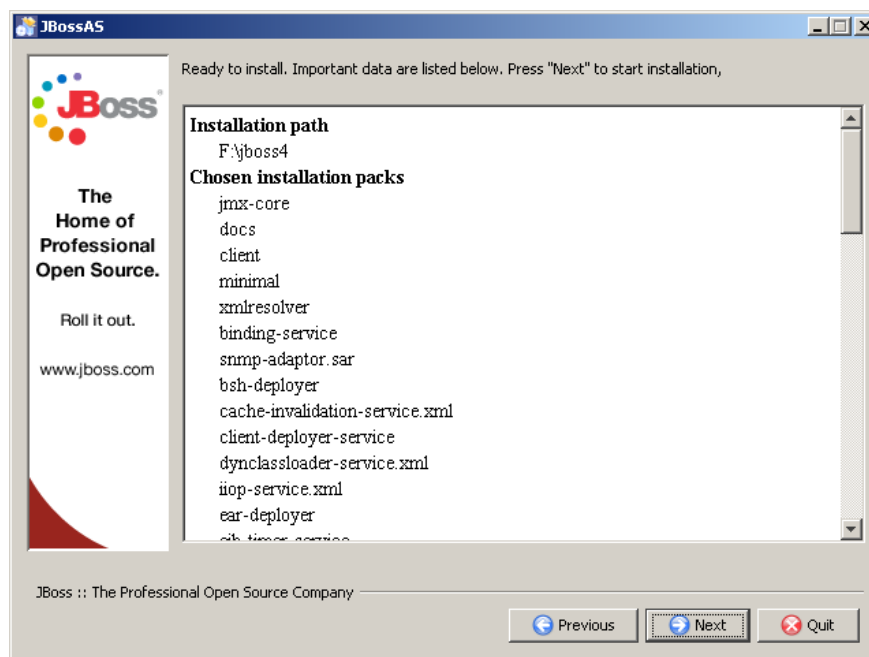


Figura 53. Pantalla de ruta de instalación del servidor y sus paquetes.

A continuación se muestra el progreso de la instalación.

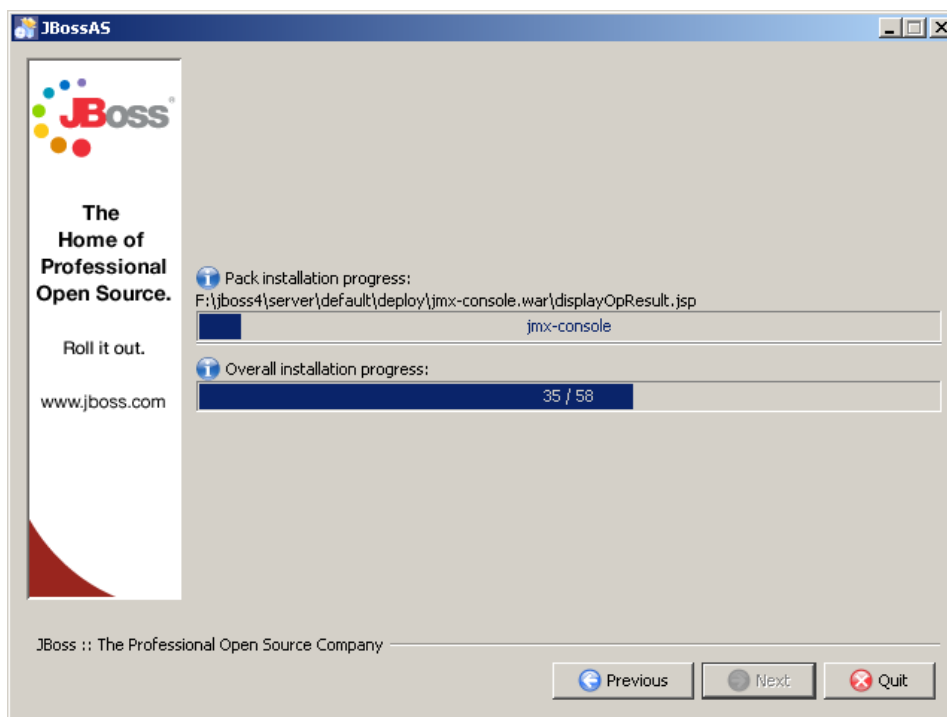


Figura 54. Pantalla de proceso de instalación del servidor.

En esta pantalla se indica que se puede almacenar un archivo resumen de la instalación realizada y un script para replicar la instalación, esto es muy útil para cuando se necesite instalar masivamente el programa en equipos con similares características

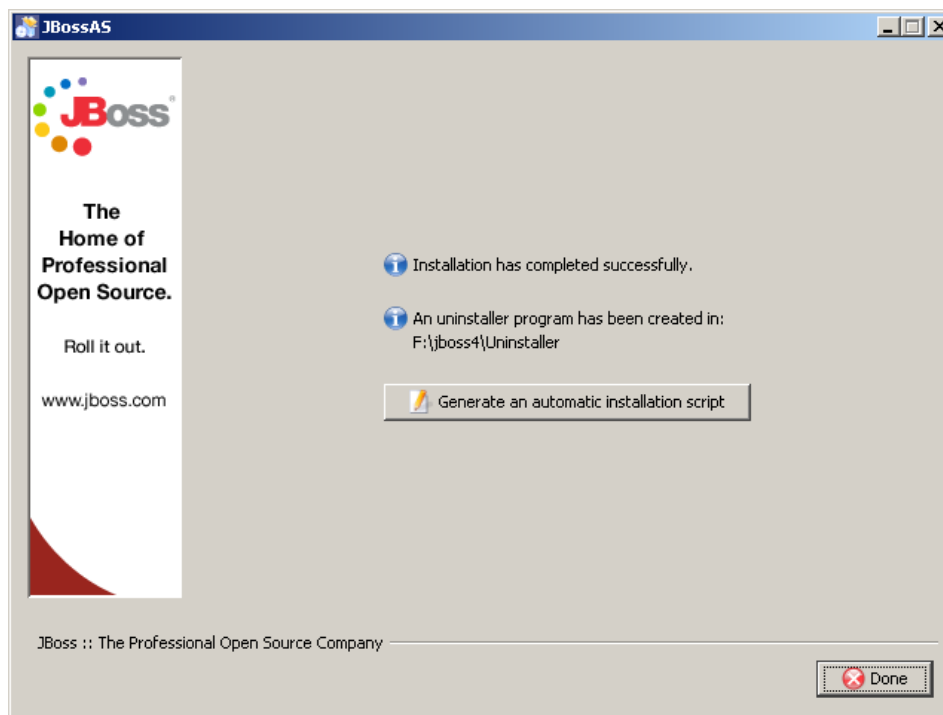


Figura 55. Pantalla de finalización de instalación.

Luego de haber instalado el servidor es necesario configurar su variable de entorno la cual es `JBOSS_HOME` y se indica su ruta de instalación, como lo muestra la figura.

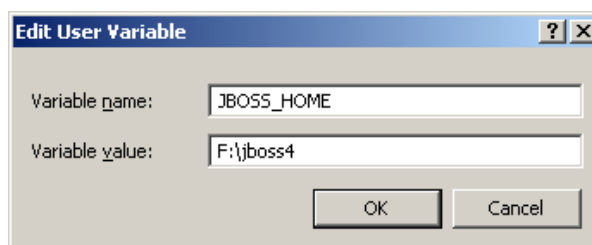
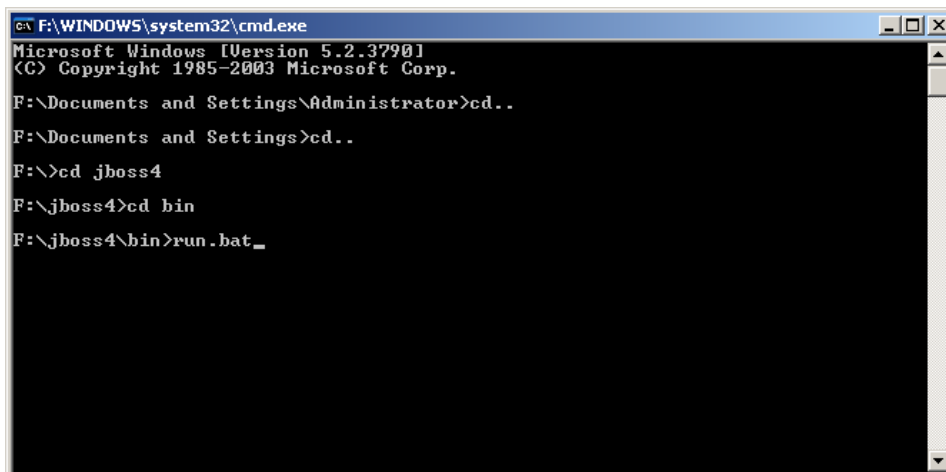


Figura 56. Pantalla de creación de variable de entorno JBOSS HOME.

Para correr el servidor de aplicaciones ingresamos a modo consola y nos ubicamos en la ruta de acceso en donde se encuentra el archivo `run.bat` y se lo ejecuta como se indica en la pantalla.



```

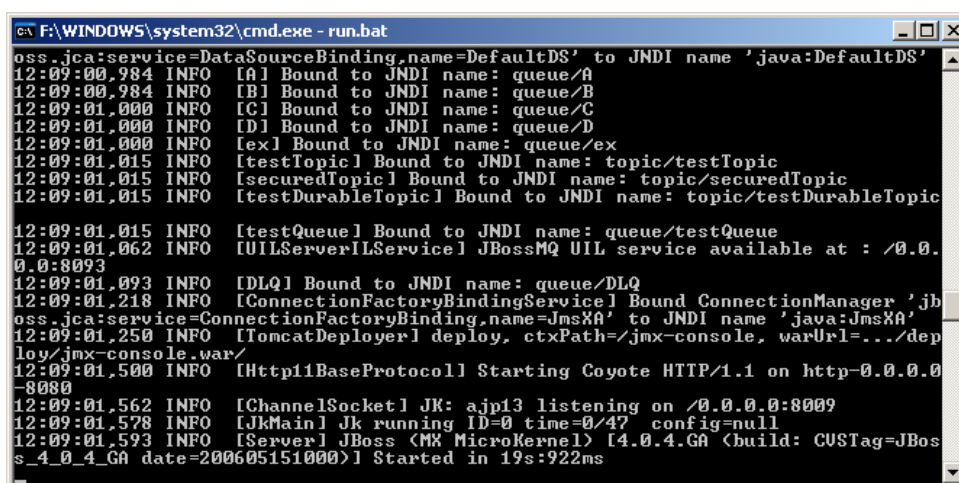
F:\WINDOWS\system32\cmd.exe
Microsoft Windows [Version 5.2.3790]
(C) Copyright 1985-2003 Microsoft Corp.

F:\Documents and Settings\Administrator>cd..
F:\Documents and Settings>cd..
F:\>cd jboss4
F:\jboss4>cd bin
F:\jboss4\bin>run.bat_

```

Figura 57. Pantalla de inicialización del servidor por medio de consola del sistema.

Esta ventana que indica la ejecución del servidor de aplicaciones permanecerá congelada mientras JBOSS esté activo. Se visualizarán, entre otras cosas, los errores de ejecución y los resultados de agregar EJB's al directorio **deploy** de JBOSS.



```

F:\WINDOWS\system32\cmd.exe - run.bat
oss.jca:service=DataSourceBinding,name=DefaultDS' to JNDI name 'java:DefaultDS'
12:09:00,984 INFO [A] Bound to JNDI name: queue/A
12:09:00,984 INFO [B] Bound to JNDI name: queue/B
12:09:01,000 INFO [C] Bound to JNDI name: queue/C
12:09:01,000 INFO [D] Bound to JNDI name: queue/D
12:09:01,000 INFO [ex] Bound to JNDI name: queue/ex
12:09:01,015 INFO [testTopic] Bound to JNDI name: topic/testTopic
12:09:01,015 INFO [securedTopic] Bound to JNDI name: topic/securedTopic
12:09:01,015 INFO [testDurableTopic] Bound to JNDI name: topic/testDurableTopic

12:09:01,015 INFO [testQueue] Bound to JNDI name: queue/testQueue
12:09:01,062 INFO [UILServerILService] JBossMQ UIL service available at : /0.0.0.0:8093
12:09:01,093 INFO [DLQ] Bound to JNDI name: queue/DLQ
12:09:01,218 INFO [ConnectionFactoryBindingService] Bound ConnectionManager 'jboss.jca:service=ConnectionFactoryBinding,name=JmsXA' to JNDI name 'java:JmsXA'
12:09:01,250 INFO [TomcatDeployer] deploy, ctxPath=/jmx-console, warUrl=.../deploy/jmx-console.war/
12:09:01,500 INFO [Http11BaseProtocol] Starting Coyote HTTP/1.1 on http-0.0.0.0-8080
12:09:01,562 INFO [ChannelSocket] JK: ajp13 listening on /0.0.0.0:8009
12:09:01,578 INFO [JkMain] Jk running ID=0 time=0/47 config=null
12:09:01,593 INFO [Server] JBoss (MX MicroKernel) [4.0.4.GA (build: CUSTag=JBoss_4_0_4_GA date=200605151000)] Started in 19s:922ms

```

Figura 58. Pantalla de arranque del servidor de aplicaciones.

Para verificar que JBOSS está arrancado se accede a través del browser a <http://localhost:8080/>. Hay que asegurarse de que en el sistema no haya nada que esté utilizando el puerto 8080. También se puede acceder a [JBoss JMX Management Console](#) (consola de administración de JBoss), en la que se muestran todos los JMX MBean que se están ejecutando en el servidor.

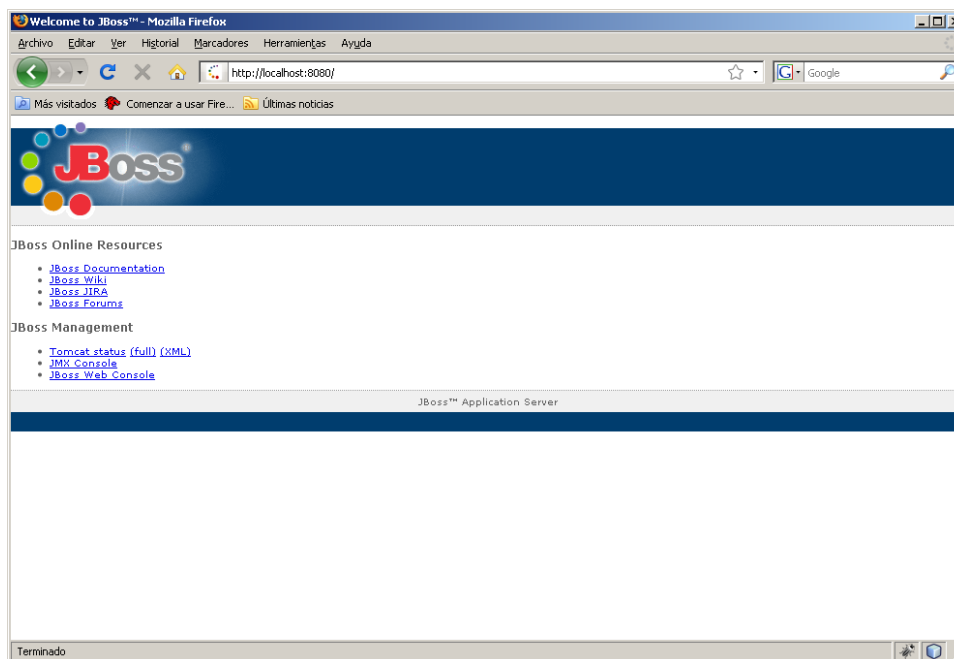


Figura 59. Pantalla de la consola principal del servidor en ambiente web.

2.5.2.4. Configuración de Glassfish

El siguiente procedimiento define la manera que se debe seguir para levantar el proyecto Portal Web sobre el servidor de aplicaciones GLASSFISH independientemente del estado de desarrollo que se encuentre la aplicación, por lo que se debe realizar los siguientes pasos:

1. De la fuente (cd de proyecto) de los datos, copiar en la raíz toda la carpeta del Proyecto llamada “ProyectoFinal”, el cual contiene todos los archivos necesarios para el funcionamiento correcto del Software.

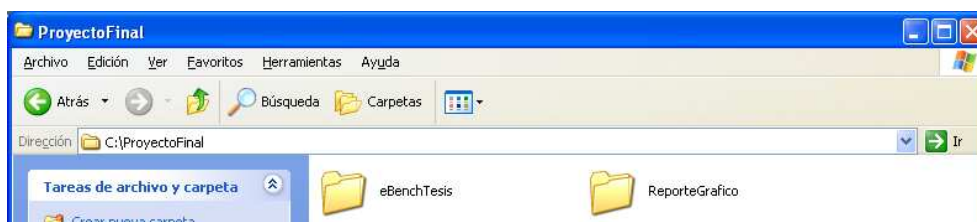
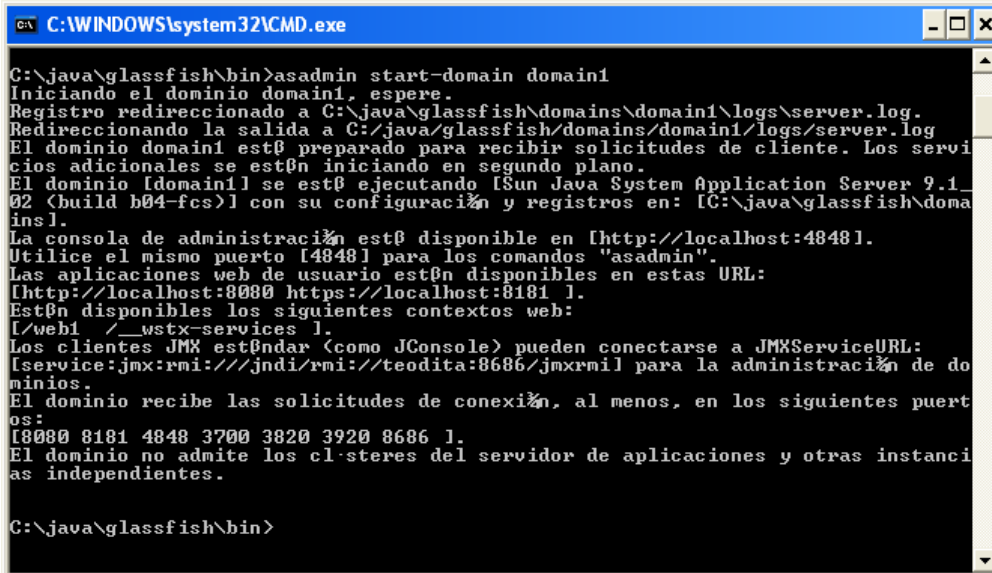


Figura 60. Pantalla de archivos del Portal Web ubicados en la raíz del directorio C.

2. Copiamos los drivers necesarios en el directorio de GLASSFISH y en el de jdk, los archivos que se deben copiar son:

C:\java\glassfish\lib	sigar-x86-winnt.dll
	ojdbc14
C:\Archivos de programa\Java\jdk1.6.0_10\bin	sigar-x86-winnt.dll

3. Después de Instalar correctamente el Servidor de Aplicaciones GLASSFISH, debemos iniciarlo cargándolo desde consola para lo cual ingresamos en el directorio de GLASSFISH perteneciente a las configuraciones, es decir, el directorio bin: en este introducimos el comando: `asadmin start-domain domain1` (en donde: domain es el nombre del dominio y domain1 es el nombre del usuario del dominio), presionamos enter y el servidor se levanta.



```

C:\WINDOWS\system32\CMD.exe
C:\java\glassfish\bin>asadmin start-domain domain1
Iniciando el dominio domain1, espere.
Registro redireccionado a C:\java\glassfish\domains\domain1\logs\server.log.
Redireccionando la salida a C:\java\glassfish\domains\domain1\logs\server.log
El dominio domain1 est&eacute; preparado para recibir solicitudes de cliente. Los servi
cios adicionales se est&eacute;n iniciando en segundo plano.
El dominio [domain1] se est&eacute; ejecutando [Sun Java System Application Server 9.1_
02 (build b04-fcs)] con su configuraci&eacute;n y registros en: C:\java\glassfish\doma
ins\
La consola de administraci&eacute;n est&eacute; disponible en [http://localhost:4848].
Utilice el mismo puerto [4848] para los comandos "asadmin".
Las aplicaciones web de usuario est&eacute;n disponibles en estas URL:
[http://localhost:8080 https://localhost:8181 ].
Est&eacute;n disponibles los siguientes contextos web:
[/web1 /_wstx-services ].
Los clientes JMX est&eacute;ndar (como JConsole) pueden conectarse a JMXServiceURL:
[service:jmx:rmi:///jndi/rmi://teodita:8686/jmxrmi] para la administraci&eacute;n de do
minos.
El dominio recibe las solicitudes de conexi&eacute;n, al menos, en los siguientes puert
os:
[8080 8181 4848 3700 3820 3920 8686 ].
El dominio no admite los clusters del servidor de aplicaciones y otras instanci
as independientes.

C:\java\glassfish\bin>

```

Figura 61. Pantalla de inicio del servidor de aplicaciones.

4. Se inicia la configuracién del servidor abriendo la consola de Administracién para lo que se requiere de un browser, se ubica el servidor en la direccién: <http://localhost:4848>, el nombre del usuario admin y la contraseña adminadmin.

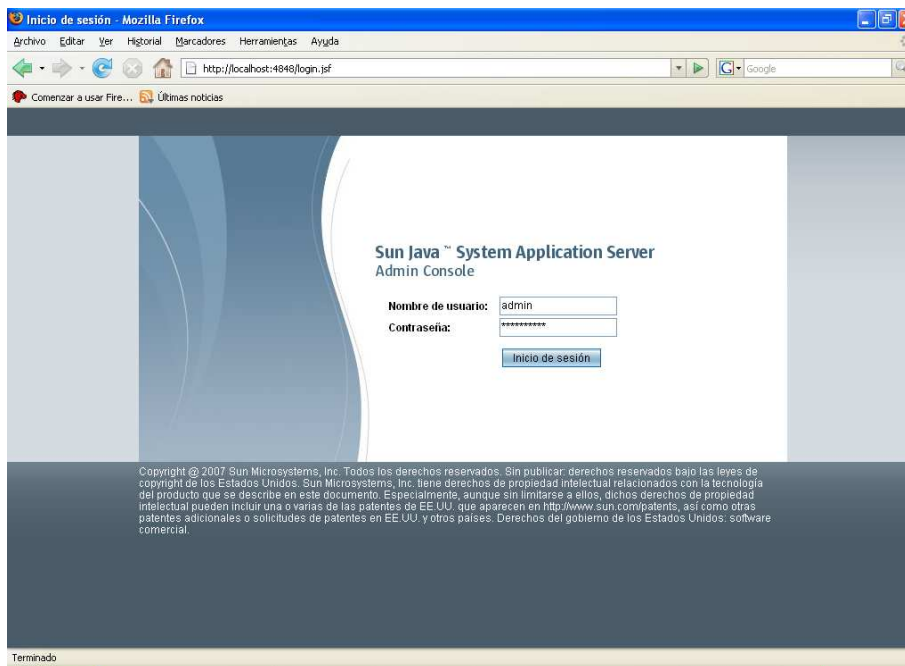


Figura 62. Pantalla de acceso a la consola Web del servidor de aplicaciones.

5. Abierta la consola nos dirigimos a Aplicaciones y a Aplicaciones de empresa, en esta sección pulsamos Implementar y buscamos el archivo generado ebench.ear que se encuentra en la carpeta dist de nuestro proyecto.

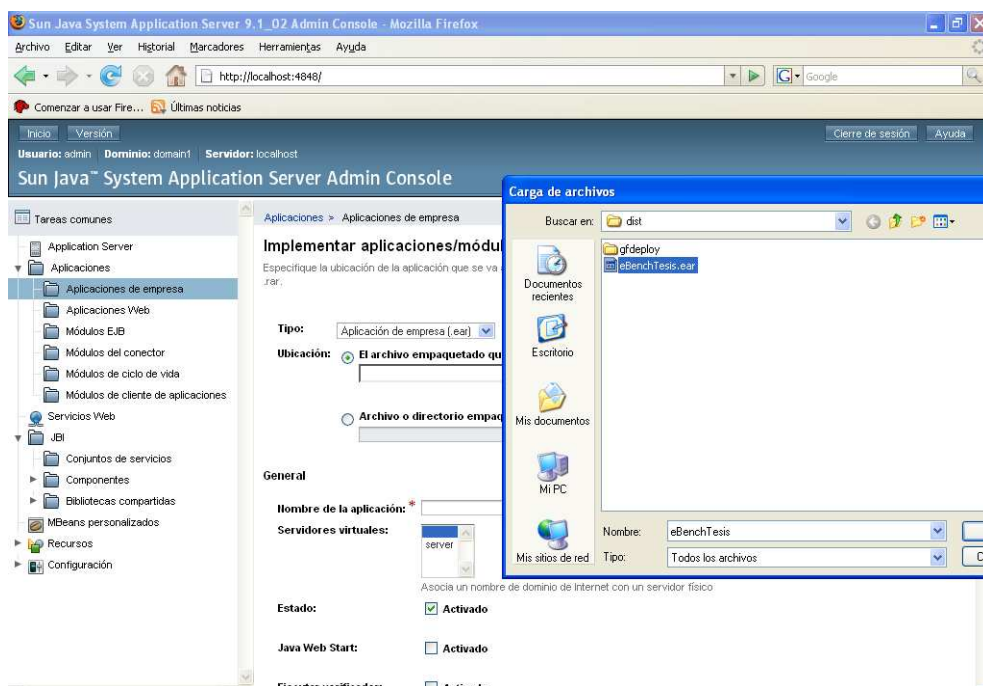


Figura 63. Pantalla de selección del archivo ear ubicado en la carpeta colocada en la raíz del directorio C.

6. Una vez que aparezca el proyecto implementado, procedemos a configurar el pool de conexiones de la base de datos y el Datasource que nos permitirán realizar las consultas e interacciones con Oracle: en el menú izquierdo del servidor, en la sección de recursos tenemos los recursos JDBC y los conjuntos de conexiones.
7. En primer lugar creamos el conjunto o pool de conexiones siguiendo las siguientes especificaciones para nuestro proyecto: En conjuntos seleccionamos Nuevo, ubicamos los siguientes datos:
 - a. Nombre: oracle-thin_TESIS_benchmarkPool
 - b. Tipo de Recurso: javax.sql.DataSource
 - c. Proveedor de la Base de datos: Oracle
8. Después de hacer clic en siguiente nos dirigimos a Propiedades Adicionales y agregamos los siguientes datos:
 - User: benchmak
 - Password: benchmark
 - driverClass: oracle.jdbc.OracleDriver
 - URL: jdbc:oracle:thin:@ratoncita:1521:TESIS

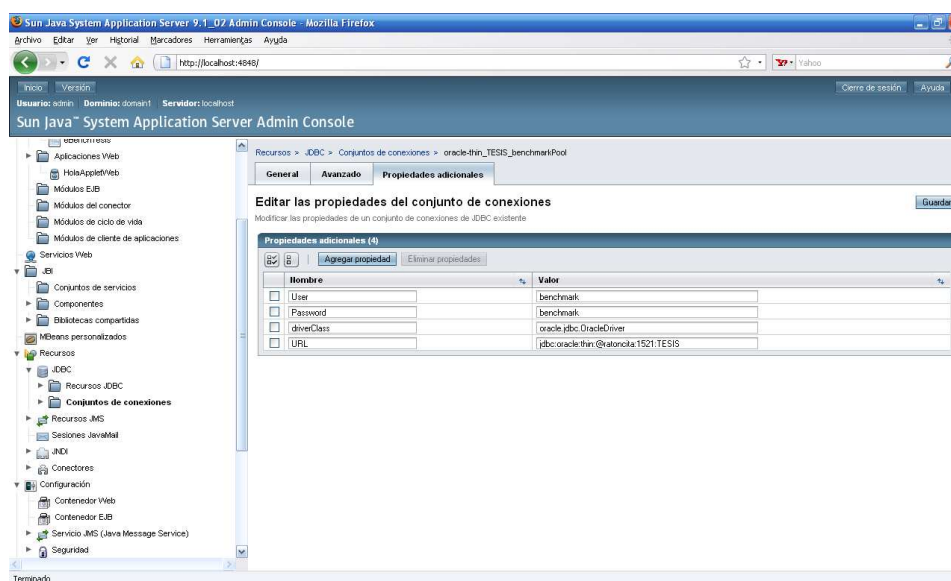


Figura 64. Pantalla de selección de propiedades definidas para el pool de conexiones.

9. Configuramos la fuente de Datos DataSource, dirigiéndonos a la sección de Recursos JDBC, igualmente creamos un nuevo Datasoruce con los siguientes datos:
 - Nombre JNDI: jdbc/myDataB-D
 - Nombre del Conjunto: oracle-thin_TESIS_benchmarkPool
10. Finalmente aceptamos y arrancamos la página del proyecto

2.5.2.5. Configuración de JBoss

Para levantar el proyecto Portal Web en el servidor de aplicaciones JBOSS se debe realizar los siguientes pasos dentro del directorio del servidor de aplicaciones:

Toda configuración de los archivos mencionados a continuación será posteriormente indicada en la sección ANEXOS.

1. De la fuente (cd de proyecto) de los datos, copiar en la raíz toda la carpeta del Proyecto llamada “PROYECTOTESIS”, el cual contiene todos los archivos necesarios para el funcionamiento correcto del Software.
2. En dicha carpeta que se coloco todos los archivos referentes al sistema hay que ubicar el archivo con extensión EAR en este caso es TESEISEBENCH y copiarlo en la carpeta rutainstalaciónservidor\server\default\deploy, en esta misma carpeta se crea un archivo que servirá para la conexión a la base de datos llamado jboss-ds.xml.
3. Luego en la carpeta rutainstalaciónservidor\server\default\conf hay que realizar la configuración necesaria de dos archivos para que la conexión a la base de datos tenga éxito estos son: standardjaws.xml y standardjbosscomp-jdbc.xml.
4. A continuación en la carpeta rutainstalaciónservidor\server\default\lib se colocan los archivos externos que son parte necesaria para funcionamiento de la aplicación estos son: ojdbc14.jar util para conectar JBOSS con la base de datos Oracle y sigar-x86-winnt.dll librería utilizada para obtener información de los recursos del parámetro a prueba.

	ojdbc14
C:\JBOSS\server\default\lib	sigar-x86-winnt.dll
C:\Archivos de programa\Java\jdk1.6.0_10\bin	sigar-x86-winnt.dll

2.6. MEDICIÓN DE PARÁMETROS Y GENERACIÓN DE RESULTADOS.

La medición de los parámetros se los realiza según lo planificado siguiendo las especificaciones, en donde se plantea los detalles de cada parámetro y tiempo de respuesta de las peticiones Web que los usuarios cliente realizan desde la aplicación.

Los resultados son almacenados en archivos planos que pueden ser representados por la misma aplicación en las interfaces de los usuarios medidores.

Las tablas de los resultados se los pueden encontrar en la sección de anexos.

Con los resultados obtenidos se pueden comparar las mediciones tanto de promedios como de rendimientos y utilizando los parámetros de medida como moda, media, desviación estándar, con los que se puede concluir la capacidad de los servidores.

CAPÍTULO 3. ANÁLISIS DE RESULTADOS

En este capítulo se describe el análisis de los datos obtenidos por los servidores de aplicaciones de manera individual de acuerdo a como cada uno de estos emplea el uso de los recursos planteados para su medición y los tiempos de respuesta que conlleva a la realización de las tareas a ejecutar. Por último se realiza un análisis comparativo de los resultados obtenidos de cada servidor para poder determinar bajo que criterios y normas se presentan dichos resultados.

3.1 ANÁLISIS DE LOS DATOS OBTENIDOS POR JBOSS

El análisis de los datos que el servidor de aplicaciones JBOSS obtuvo durante el intervalo de medición de los diferentes recursos será el estudio en concreto de las mediciones referentes a 1000 usuarios con las tareas 1 registro y 3 compras generando 4000 registros. Todo el intervalo de medición es aprovechado para realizar esas tareas y entender como el servidor actúa, y además el estudio de estos datos en concreto no influirá en la tendencia que marca el servidor, cada uno de los resultados de los parámetros serán detallados a continuación

Cabe indicar previamente que el parámetro CPU generara resultados de acuerdo al uso del sistema operativo y al uso por parte del servidor de aplicaciones, en base a eso lo más conveniente es un estudio separado de cada uno de estos.

3.1.1. ANÁLISIS DE DATOS DEL USO DEL PARÁMETRO CPU POR PARTE DEL SISTEMA OPERATIVO

La gráfica y su análisis respectivo de acuerdo a las medidas obtenidas por parte del sistema operativo son detallados.

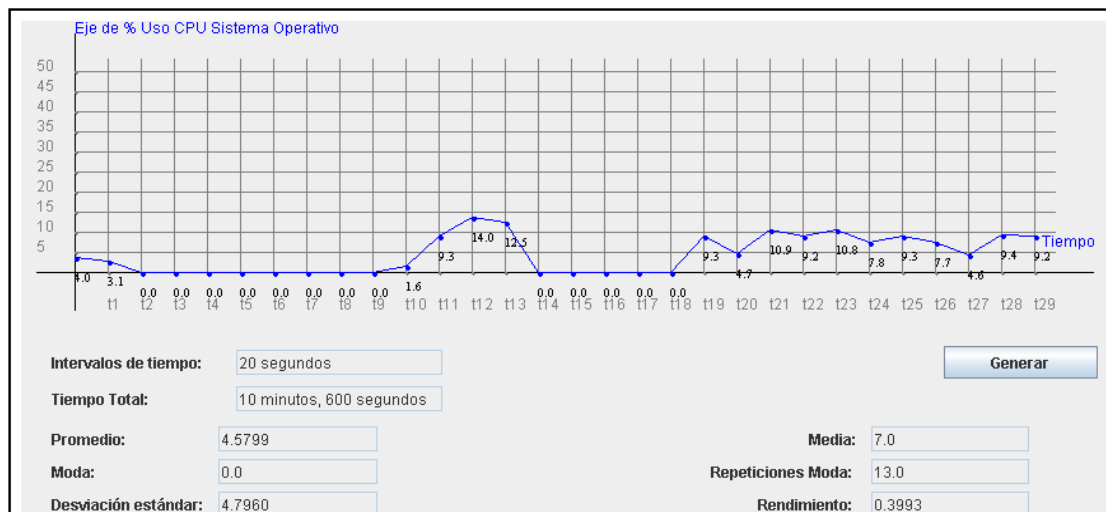


Figura 65. Porcentaje de Uso CPU Sistema Operativo JBOSS

Como se puede apreciar el total de uso de CPU por el sistema operativo es mínimo llegando a un valor máximo del 14% y así también observamos que existen 13 resultados que arrojan valor de cero en la medición lo cual nos explica que casi la mitad del tiempo de medición el sistema operativo no procesa otro tipo de tareas, por otro lado existen dos periodos de tiempo en los que se nota un aumento en el porcentaje de uso esto se da en todas las mediciones en lo que respecta al CPU lo cual indica la utilización de los threads y recursos necesarios para procesar las peticiones.

3.1.2. ANÁLISIS DE DATOS DEL USO DEL PARÁMETRO CPU POR PARTE DEL SERVIDOR DE APLICACIONES

La gráfica y su análisis respectivo de acuerdo a las medidas obtenidas por parte del servidor de aplicaciones son detallados.

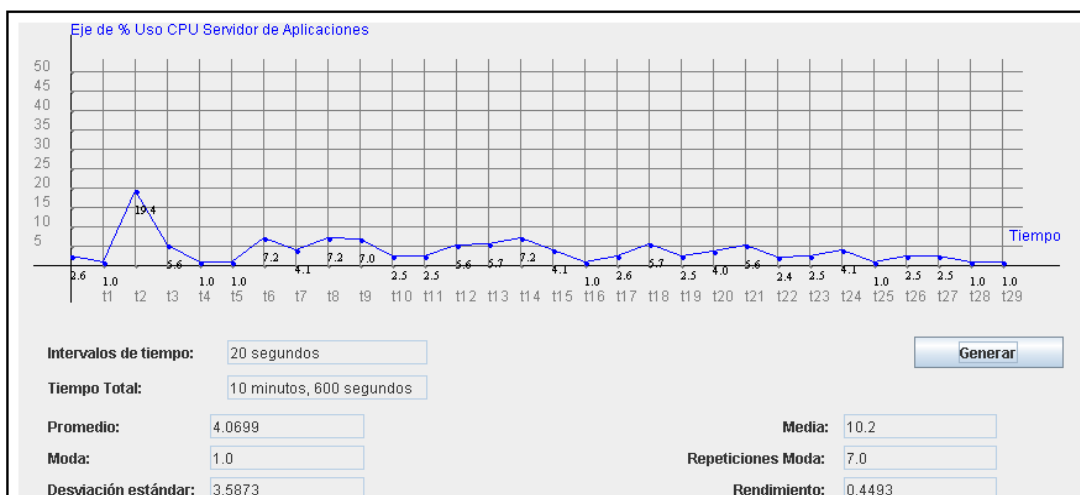


Figura 66. Porcentaje de Uso CPU Servidor Aplicaciones JBOSS

Como se puede apreciar en la gráfica el servidor de aplicaciones genera un trabajo constante de los threads en el CPU debido a su permanente comunicación con el usuario y la gestión de recursos propia del servidor. Los resultados de las medidas generadas nos indican que el porcentaje varía generalmente entre 1 y 7 % encontrando además un valor atípico de 19.4 %. Se observa también que existe una tendencia o moda del 1% de uso del CPU, este valor es el mínimo que puede ser medido ya que pertenece a la gestión de recursos en la instancia del servidor. El valor atípico corresponde probablemente a la gestión del servidor realizando cargas correspondientes al pool de conexiones y a la carga de los EJBs para que ellos puedan interactuar con la base de datos, este valor como se puede ver se lo realiza por una sola vez al cargar el propio servidor.

3.1.3. ANÁLISIS DE DATOS DEL USO DEL PARÁMETRO MEMORIA

La gráfica y su análisis respectivo de acuerdo a las medidas obtenidas por parte del uso de este parámetro son detallados.

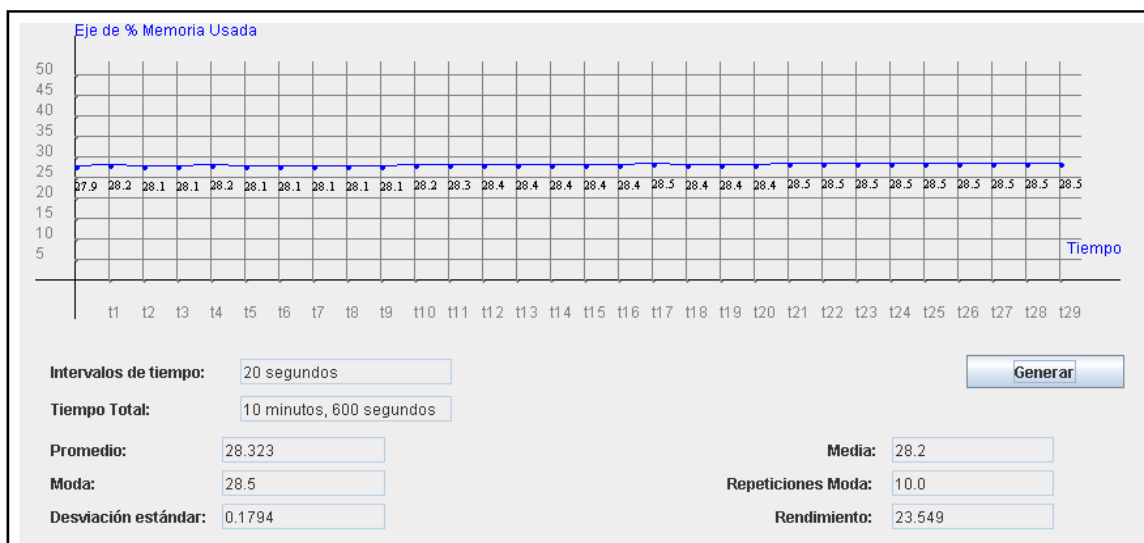


Figura 67. Porcentaje de Uso Memoria JBOSS

A simple vista se puede observar una horizontal constante en el uso de la memoria, lo que indica que esta no es afectada por el aumento en la carga de las peticiones Web.

Se debe tomar en cuenta que según las especificaciones de Red Hat acerca del servidor de aplicaciones JBOSS, el consumo de la memoria no sobrepasa los 50 MBs por aplicación; Además la capacidad de memoria del equipo en la que se despliega la aplicación es de 2048 MBs por lo tanto el promedio de memoria usada es de 580 MBs lo que implica que más de 500 MBs son usados por el sistema operativo.

Las gráficas con respecto a memoria libre indican lo opuesto a lo anteriormente explicado.

3.1.4. ANÁLISIS DE DATOS DEL USO DEL PARÁMETRO RED EN KBYTES TRANSMITIDOS

La gráfica y su análisis respectivo de acuerdo a las medidas obtenidas por parte del uso de este parámetro son detallados.

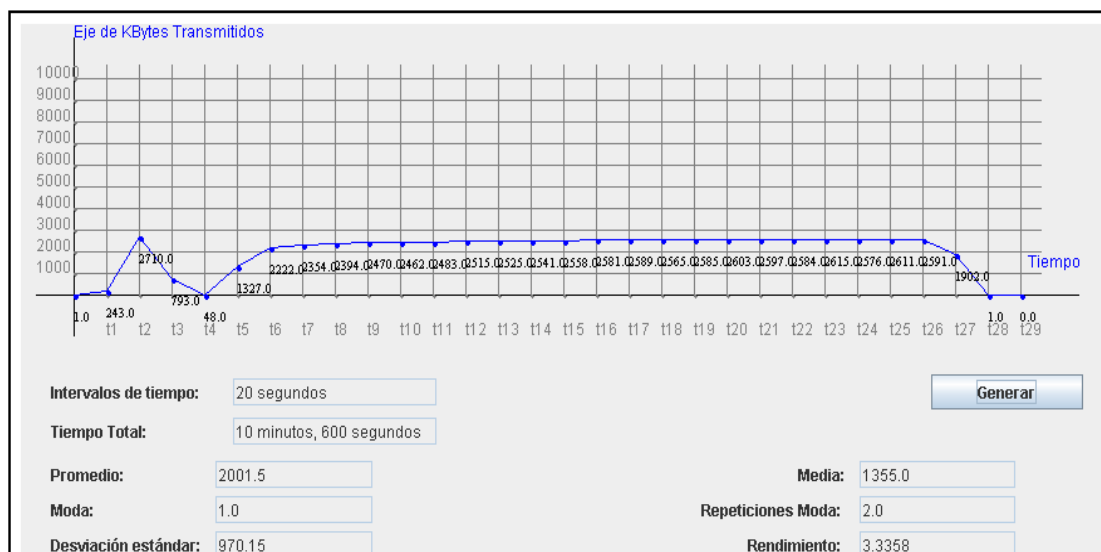


Figura 68. Medida de kbytes transmitidos JBOSS

Para generalizar las gráficas obtenidas de los datos de red en KBytes Transmitidos tanto como en recibidos se visualizan dos momentos en los que el servidor presenta un mayor flujo de información que reflejan la transmisión de datos cuando el usuario ha enviado la petición. El primer momento corresponde al registro del cliente y el segundo a la compra de productos.

Para esta gráfica en particular se observa claramente que el flujo de los paquetes del servidor de aplicaciones para el registro del cliente se lo realiza en un lapso aproximado de 70 segundos, mientras que para la compra de productos ocupa aproximadamente 420 segundos es decir la mayor cantidad de tiempo del intervalo de prueba, existiendo un intervalo intermedio entre ambos momentos de aproximadamente 30 segundos en donde el flujo de datos disminuye debido a que en este tiempo el usuario interactúa con la aplicación para elegir la compra.

3.1.5. ANÁLISIS DE DATOS DEL USO DEL PARÁMETRO RED EN KBYTES RECIBIDOS

La gráfica y su análisis respectivo de acuerdo a las medidas obtenidas por parte del uso de este parámetro son detallados.

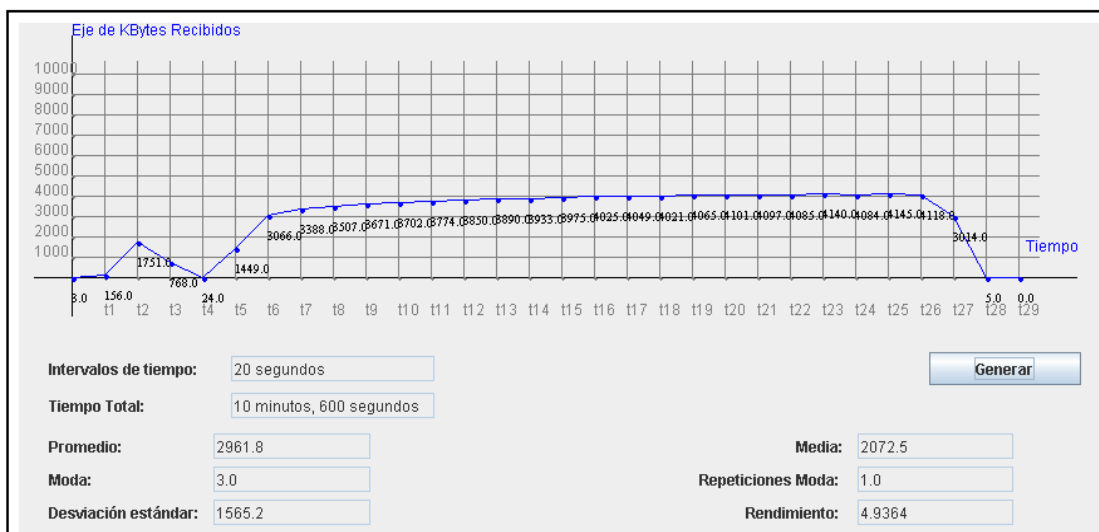


Figura 69. Medida de Kbytes Recibidos JBOSS

Se puede observar la misma estructura que en la transmisión de bytes existiendo los dos momentos importantes en donde hay mayor flujo de información por lo que se obtiene resultados de medidas muy similares que las transmitidas. La variación en el aumento de bytes recibidos se debe a la gestión del servidor de aplicaciones y su constante recepción de información tanto del usuario como de la base de datos.

3.2 ANÁLISIS DE LOS DATOS OBTENIDOS POR GLASSFISH

Siguiendo el criterio tomado para el servidor JBOSS se lo hará de igual manera para el servidor de aplicaciones GLASSFISH, es decir bajo las mismas premisas, 1000 usuarios con las tareas 1 registro y 3 compras generando 4000 registros, ya que de igual manera en todo el intervalo de medición es aprovechado para realizar esas tareas y entender como el servidor actúa, cada uno de los resultados de los parámetros serán detallados a continuación

3.2.1. ANÁLISIS DE DATOS DEL USO DEL PARÁMETRO CPU POR PARTE DEL SISTEMA OPERATIVO

La gráfica y su análisis respectivo de acuerdo a las medidas obtenidas por parte del sistema operativo son detallados.

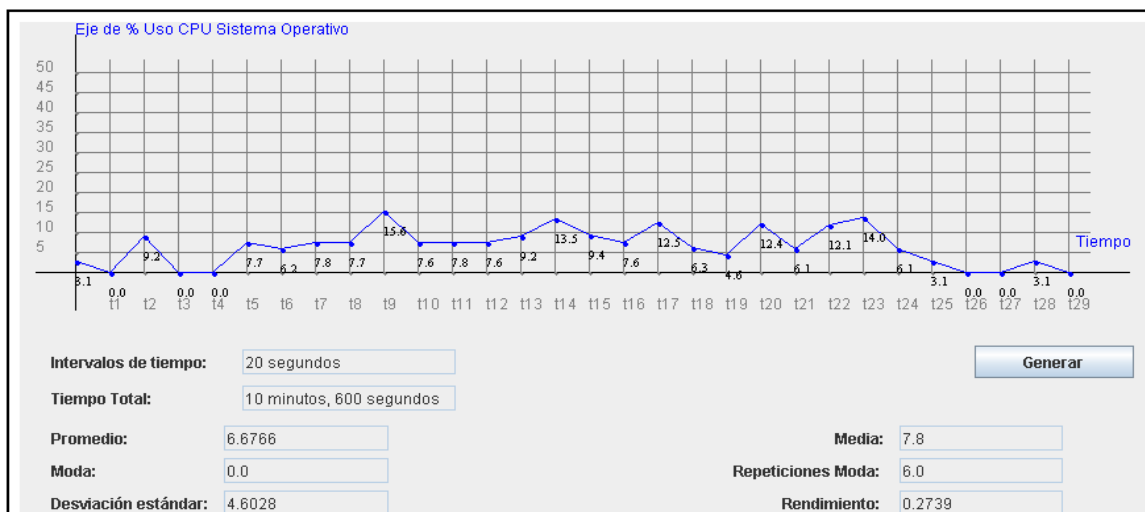


Figura 70. Porcentaje de Uso CPU Sistema Operativo GLASSFISH

Se puede notar que el servidor desde que inicia su gestión tiene una constante interacción con el sistema operativo manteniendo al procesador ejecutando las operaciones que requiera para la gestión de la aplicación, se debe tomar en cuenta que el porcentaje de uso tiene un valor máximo de 15,6 % y una desviación estándar de 4,60 lo que indica la estabilidad de los datos aunque no en su constancia, de la misma manera se puede observar que el valor que indica la moda es cero en siete ocasiones demostrando que existe un intervalo considerable del tiempo de espera del CPU.

3.2.2. ANÁLISIS DE DATOS DEL USO DEL PARÁMETRO CPU POR PARTE DEL SERVIDOR DE APLICACIONES

La gráfica y su análisis respectivo de acuerdo a las medidas obtenidas por parte del servidor de aplicaciones son detallados.

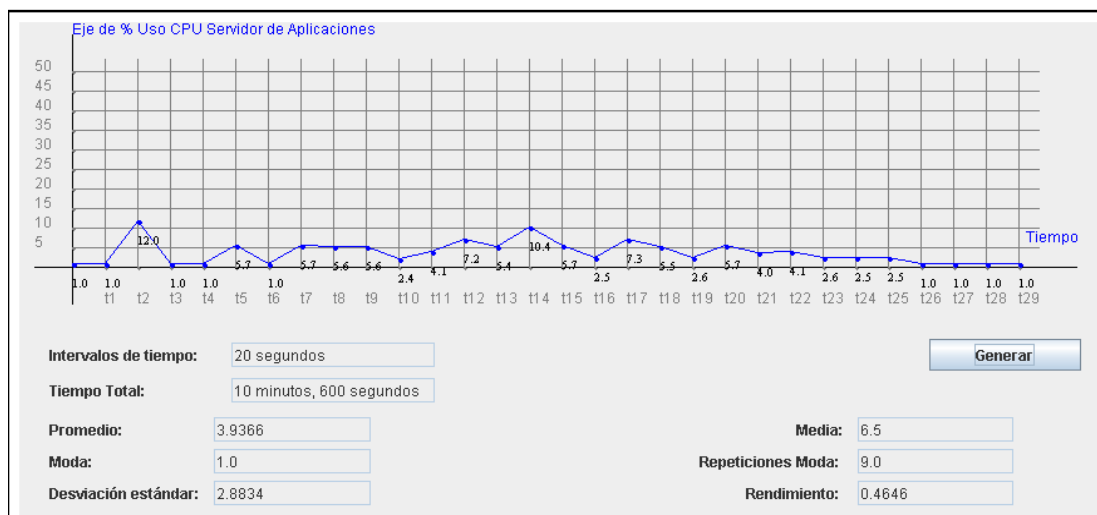


Figura 71. Porcentaje de Uso CPU Servidor Aplicaciones GLASSFISH

De los datos representados en la gráfica se aprecia nuevamente la misma tendencia de los datos sin embargo se nota que la moda en nueve de las muestras totales presentan un valor de uno, que es el valor mínimo del servidor interactuando con el procesador debido a la gestión, en el resto de las medidas este valor aumenta ya que se genera un mayor número de threads, además las especificaciones propias del servidor indican que los porcentajes de uso de CPU para este tipo de aplicaciones no varían en rangos muy elevados cuando se aumenta el número de usuarios debido al tratamiento de los procesos de usuario en threads individuales; Por lo tanto el uso del CPU no representa un problema para ninguno de los servidores en cuanto al rendimiento de la aplicación ya que no afectará el tiempo de respuesta con el usuario

3.2.3. ANÁLISIS DE DATOS DEL USO DEL PARÁMETRO MEMORIA

La gráfica y su análisis respectivo de acuerdo a las medidas obtenidas por parte del uso de este parámetro son detallados.

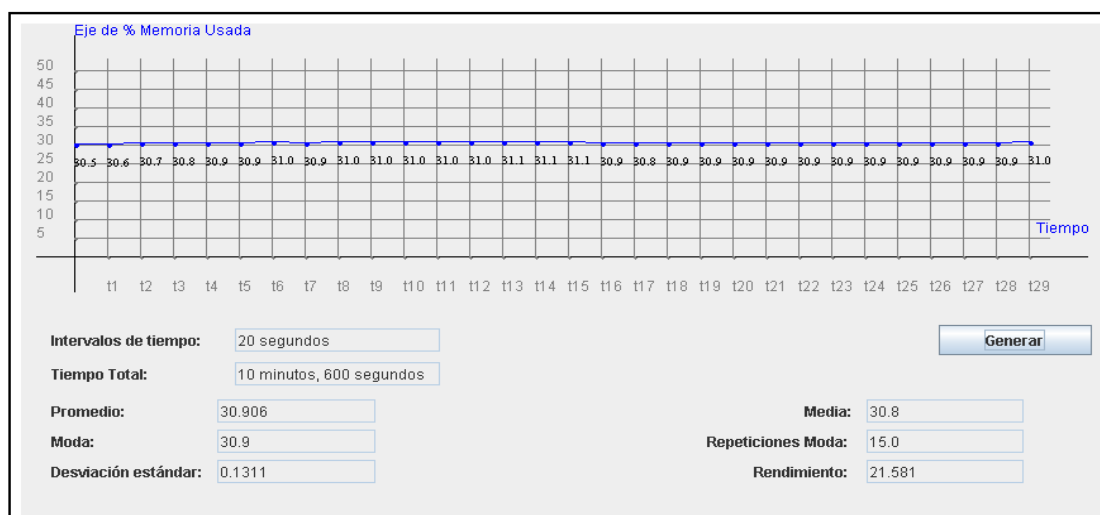


Figura 72. Porcentaje de Uso Memoria GLASSFISH

Las especificaciones del servidor GLAAFISH al igual que se lo hizo con el servidor JBOSS indican que el uso de la memoria no supera los 50MB, variando de acuerdo a la aplicación que este desplegada con una diferencia mínima esto se deba a la gestión realizada por los servidores y la utilización de una memoria virtual por lo que tanto este parámetro tampoco representa repercusión en los tiempos de respuesta del cliente tomando en cuenta las características del hardware de la maquina del servidor es decir 2048MB. Los valores observados es el resultado de la suma del porcentaje de uso de memoria por parte del servidor de aplicaciones más el porcentaje de uso por parte del sistema operativo lo que aproximadamente da un valor de 632MBs de total. Si se toma en cuenta que el servidor de aplicaciones utiliza 50MB aproximados de este total, 580MBs son utilizados para gestión del sistema operativo.

Las gráficas con respecto a memoria libre indican lo opuesto a lo anteriormente explicado.

3.2.4. ANÁLISIS DE DATOS DEL USO DEL PARÁMETRO RED EN KBYTES TRANSMITIDOS

La gráfica y su análisis respectivo de acuerdo a las medidas obtenidas por parte del uso de este parámetro son detallados.

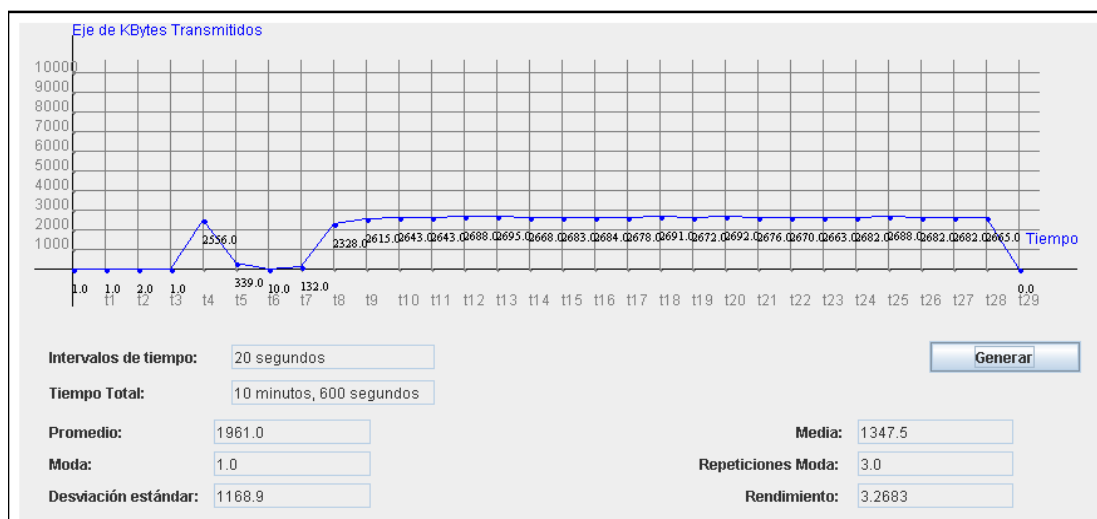


Figura 73. Medida de kbytes transmitidos GLASSFISH

De igual manera que el servidor de aplicaciones JBOSS, GLASSFISH presenta la misma tendencia de datos es decir, se visualizan dos momentos de mayor flujo de bytes correspondientes al registro del cliente y a la compra de productos. Se observa que mayor cantidad de tiempo GLASSFISH ocupa en transmisión y recepción de bytes cuando se realiza una compra.

Para esta gráfica de transmisión de datos se observa un valor elevado en la desviación estándar esto se debe a dos intervalos de tiempo, uno de aproximadamente un minuto antes de realizar el registro de cliente y otro de 40 segundos después de la realización del registro, con valores que descienden debido a que el usuario no envía peticiones Web al servidor, sin embargo se mantiene la comunicación constante cliente-servidor por lo que se envían datos desde el servidor al cliente.

Otro factor importante a tomar en cuenta son los dos momentos de mayor flujo de bytes el uno muy corto el cual llega un valor máximo de 2556 kbytes transmitidos y el otro de una duración aproximada de 400 segundos de un flujo que varía entre 2615 y 2690 Kbytes transmitidos cada 20 segundos.

3.2.5. ANÁLISIS DE DATOS DEL USO DEL PARÁMETRO RED EN KBYTES RECIBIDOS

La gráfica y su análisis respectivo de acuerdo a las medidas obtenidas por parte del uso de este parámetro son detallados.

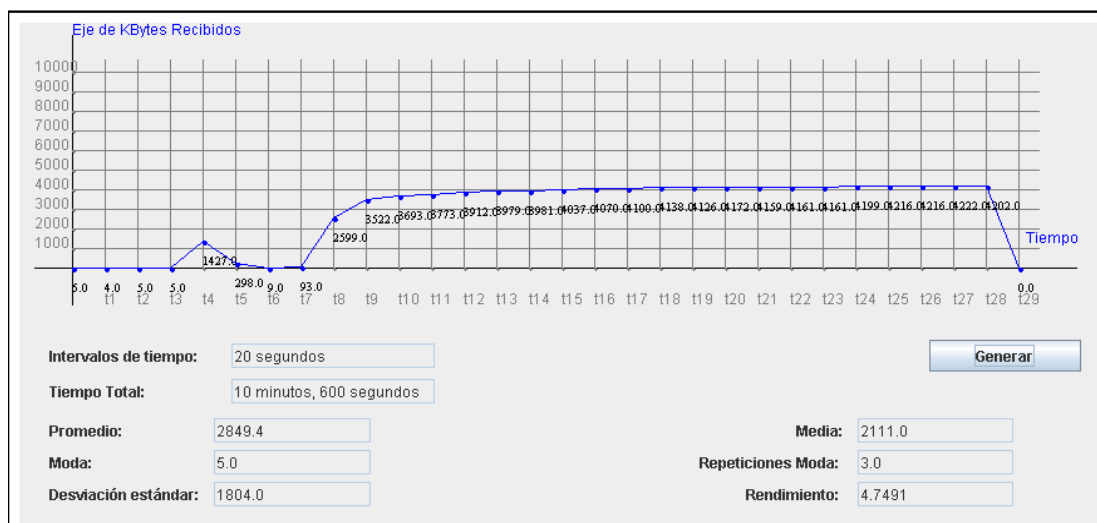


Figura 74. Medida de Kbytes Recibidos GLASSFISH

Como se ha observado durante todas las mediciones de red de kbytes recibidos GLASSFISH no es la excepción ya que muestra los dos intervalos de interacción con el usuario: registro y compra; Con una disminución notable cuando se realiza el registro llegando a un máximo de 1497 kbytes, el segundo momento mantiene la tendencia de medidas anteriores. Por último se muestra también mayor flujo de bytes recibidos que transmitidos esto de igual manera es debido a la administración del servidor de aplicaciones y su interacción con los usuarios y la base de datos.

3.3 COMPARACIÓN DE RESULTADOS

El análisis de Resultados de las mediciones nos ha entregado un grupo variado de gráficas las cuales nos permiten analizar y comparar un servidor con otro de acuerdo a los parámetros e índices seleccionados como a continuación lo realizamos

Los gráficos utilizados para la comparación de las medidas de ambos servidores se encuentran en la sección de anexos. Podemos observar tanto para promedios de porcentajes y valores de Rendimientos así como los cálculos utilizados.

3.3.1. COMPARACIÓN DEL ÍNDICE CPU

Los servidores de aplicaciones generalmente no usan gran cantidad de CPU comparado con otros programas, el mayor porcentaje de su uso es debido a la serialización de los datos a través de la red o al almacenamiento persistente, tomando en cuenta que mayor trabajo lo realiza el CPU en lectura y escritura de disco.

Existen otras causas que influyen en el uso del CPU:

- El trabajo del CPU del servidor se ve reflejado cuando necesita transformar gran cantidad documentos XSL, CSS, u otros tipos de hojas de estilo.
- El servidor de aplicaciones es capaz de escalar en uso del CPU, debido al número de peticiones que esté procesando, aunque es relativo al trabajo que la petición requiera.
- Una mala configuración de los Servidores en lo que respecta a la administración de procesos, la configuración por defecto de cada uno de los servidores permite que se utilice eficientemente el servidor, pero estos pueden ser modificados según lo que la aplicación requiera, ya que esta puede requerir mayor cantidad de recursos o dejar de utilizar aquellos que no requiera la aplicación
- Manejo de threads especialmente los de pool de conexiones, los cuales generan mayor cantidad de trabajo del CPU cuando se trata de una aplicación Web como la que estamos utilizando de prototipo. El manejo de threads los servidores de aplicación lo manejan individualmente y siguiendo distintos algoritmos, aunque en ambos casos siguen una misma estructura de multihilos con colas de procesos.

3.3.1.1. Análisis de Resultados del % de uso de CPU con respecto al Sistema Operativo

Los promedios de porcentajes de uso de CPU aumentan según el número de usuarios

Cuando comparamos los promedios de % de uso CPU con respecto al Sistema Operativo podemos observar que este aumenta debido al uso de recursos de Sistema Operativo utilizados para efectuar las peticiones Web concurrentes que ejercen los usuarios, estos resultados están en concordancia al trabajo realizado por el Servidor de Aplicaciones,

conociendo de antemano la relación de ambos servidores a través de una Máquina Virtual de Java (JVM) con el Sistema operativo quien aún se encarga de administrar los protocolos de comunicación, uso de memoria, y en general recursos Hardware, por lo tanto se ve reflejado el aumento en el porcentaje de uso mientras actúan estos recursos.

Particularmente la carga del porcentaje de uso de CPU con respecto al Sistema operativo se ve reflejado por el incremento de usuarios que realizan las mismas actividades por lo que el servidor a través de la Máquina Virtual de Java, no presenta gran interacción con el Sistema Operativo

Podemos observar que a pesar de que GLASSFISH mantiene una mayor interacción con el Sistema Operativo, durante el intervalo de medición de peticiones Web, los valores de porcentaje de uso no mantienen el equilibrio total en los resultados, esto se debe a que el procesador es utilizado obligatoriamente por procesos en background pertenecientes a la misma administración del Sistema Operativo que no dependen del Servidor de Aplicaciones.

La Máquina Virtual de Java se encarga de la traducción de las operaciones Java a lenguaje de comunicación con el Hardware el cual administra el Sistema Operativo, por lo tanto no afecta el ambiente de los Servidores ya que es el mismo lenguaje utilizado para soportar la aplicación en ambos servidores.

Si tomamos en cuenta las especificaciones de línea base el porcentaje de uso de CPU mantiene un buen rendimiento mientras no suba del 75 % por lo tanto se mantiene en un buen nivel de rendimiento

El porcentaje de uso del CPU en lo que respecta al Sistema Operativo, nos muestra que el trabajo realizado efectivamente crece al aumentar el número de usuarios, sin embargo de que el porcentaje es relativamente bajo ya que no llega a utilizarse el 11%, podemos notar una importante variación entre 100 y 1000 usuarios, así como entre 1000 y 5000 usuarios. Entre 5000 y 10000 usuarios tiende este promedio a estabilizarse.

El rendimiento por su parte se obtiene de la relación entre los ciclos de reloj que utiliza el CPU promedio por programa (valor reflejado en la frecuencia de velocidad del CPU en MHz) sobre la tasa de reloj (medida del promedio en el tiempo de medida)

El rendimiento mostrado entre los servidores en el porcentaje de uso del Sistema Operativo se mantiene en rangos regulares muy parecidos con ligeras variaciones que no inciden en el tiempo de respuesta de las peticiones Web. Están directamente relacionados con los valores del promedio ya que crecen de manera inversamente proporcional por lo tanto se deben a los mismos factores.

El rendimiento es obtenido de la relación entre la frecuencia resultante de los ciclos del reloj del CPU y la tasa medida en un intervalo de tiempo (el intervalo planificado), donde podemos observar que el rendimiento crece de manera inversamente proporcional al promedio, de manera lógica podemos concluir que a mayor uso de CPU, el Servidor de Aplicaciones presenta un menor rendimiento según este parámetro.

3.3.1.2. Análisis de Resultados del % de uso de CPU con respecto al Servidor de Aplicaciones

Las medidas correspondientes al promedio de uso de CPU por parte de los servidores de aplicaciones no sobrepasan del 7%, en las gráficas se nota que hay un leve aumento por parte del servidor JBOSS al momento de interactuar con una cantidad de 100 y 1000 usuarios, en cuanto a 5000 y 10000 usuarios JBOSS tiende a separarse del patrón normal de aumento que mantiene con GLASSFISH, ahora dependiendo del número de usuarios entre 100 y 1000 se visualiza que existe un gran cambio en ambos servidores, entre 1000 a 5000 y de 5000 a 10000 el cambio es gradual por lo que existe un uso óptimo de los servidores de aplicaciones con la CPU.

Para poder dar un análisis de cómo estos valores representan el rendimiento hay que tomar en cuenta que cuando se trata de servidores de aplicaciones existe un valor que limita un uso óptimo de este parámetro equivalente al 75%, esto a valor numérico en la gráfica está por los 0.024, que son representados por la frecuencia del procesador sobre la tasa de ese porcentaje, si los datos fueran menores a este valor umbral no sería óptimo su uso,

siguiendo esta norma los resultados de las gráficas demuestran que los datos obtenidos para rendimiento son altos lo que da lugar a entender que con un número máximo de 10000 usuarios tanto GLASSFISH como JBOSS rinden de manera óptima. De acuerdo a la carga de usuarios se visualiza que la variación entre 100 a 1000 y de 1000 a 5000 su diferencia tiende a ser constante, pero de 5000 a 10000 usuarios su variación es mínima, por lo que el servidor de aplicaciones tendrá la posibilidad de atender sin ningún problema las solicitudes generadas por los usuarios.

Los cambios en estas medidas se deben a los factores que a continuación se detallan.

El promedio del % de uso de la CPU por parte de los servidores de aplicaciones tiende a ser proporcional al número de usuarios por lo que aumenta, pero su aumento no demanda mucha utilización de este recurso debido a que los servidores de aplicaciones administran de manera anticipada cada uno de los servicios que necesitara ejecutar a su debido tiempo a través de múltiples hilos los cuales mantienen un estado expectante de procesos requeridos por la petición Web evitando un uso indebido o una sobrecarga de servicios.

Las gráficas reflejan el comportamiento que tiene los servidores de aplicaciones con el uso de CPU por lo que características como máquina virtual, el uso de EJBs y el uso adecuado del JDBC influyen para que su uso sea óptimo.

En el caso de la máquina virtual las configuraciones por defecto que manejan los servidores de aplicaciones aprovechan de gran manera el tamaño de la pila para generar resultados eficientes, debido a que normalmente fijan de igual valor los tamaños mínimo y máximo de la pila lo que garantiza que cuando se inicia la máquina virtual exista suficiente memoria disponible, evitando sorpresas más tarde, cuando se trata de asignar la totalidad de memoria.

Dentro de la memoria virtual una característica que se debe tomar en cuenta es el uso del Garbage Collectors este parámetro permite dividir a la pila y clasificar los objetos de acuerdo a su creación y uso constante, estos objetos almacenan datos tales como clases y e instancias de los métodos, cuando uno de estos objetos ya no puede ser ubicado por un puntero por el programa en ejecución este permite que estos objetos pasen a formar parte

de colectores antiguos, haciendo que estos objetos ya no sean más utilizados y además que exista más memoria disponible para los objetos más jóvenes.

Otro punto a tomar en cuenta en los servidores de aplicaciones es el tipo de EJBs a manejar, esto está ligado al contenedor de EJBs, dependiendo del tipo de aplicación su elección tiende al final a influir en el rendimiento, esto es debido a que según el tipo que se elija, manejarán de diferente manera los servicios que requieran, un ejemplo es el uso del pool de conexiones no es la misma configuración para un session beans sin estado local que para uno remoto, en este caso se utiliza básicamente EJBs de conexión a la base de datos o CMP Entity Beans, de sesión locales y remotos sin estado según las características de la lógica del negocio y sus respectivas conexiones con JSPs y servlets

Con respecto al JDBC lo más recomendable es verificar y tomar en cuenta los valores de los parámetros que manejan los servidores un ejemplo de estos pueden ser el tamaño mínimo en ambos casos los servidores tienen un valor de 5 threads y máximo del pool de conexiones cuya valor para los servidores es 20 threads, el período de tiempo de una conexión inactiva que permanecerá en el pool antes de usarse en ambos casos es 300 segundos. Estos valores con más relevancia siempre estarán a criterio de los administradores y en cuanto a código al momento de obtener una conexión cercana a los recursos de la operación como sea posible lo más óptimo es cerrar tan pronto sea completada la consulta con el fin de devolver la conexión al pool.

3.3.2. COMPARACIÓN DEL ÍNDICE MEMORIA

Los resultados de las medidas del porcentaje de Uso de la memoria es el total de memoria que tanto Sistema Operativo como Servidor de Aplicaciones utilizan durante el intervalo medido. Los porcentajes en ambos casos se mantienen con una relación:

$$\frac{\%UsoMemoriaS.O.}{\%UsoMemoriaS.A} = \frac{27,5\%}{2,5\%}$$

Por lo que tomando en cuenta que la capacidad de la Memoria Total del Hardware Servidor medido en Mega Bytes es de 2048, los porcentajes llegan a un 30 % de uso de memoria aproximadamente.

Podemos deducir que el porcentaje de uso de la memoria del Servidor de aplicaciones es un valor que casi no varía y mantiene un rango de 45 a 50 Bytes, sin depender del incremento en el número de usuarios, sino más bien de la cantidad de instrucciones de la aplicación. Por lo que la variación que se refleja en las medidas está relacionada con la Administración del Sistema Operativo y no del Servidor de Aplicaciones.

Las medidas correspondientes a la memoria demuestran que la variación es mínima, el uso de la memoria arroja valores que oscilan entre el 26 y el 31% de uso lo que denota en ambos casos que utilizan de manera adecuada la memoria.

Cabe tomar en cuenta, sin embargo, que las medidas demuestran que JBOSS obtiene un mejor rendimiento en este parámetro ya que en todas las medidas, excepto en las obtenidas para 5000 y 10000 usuarios cuando se realizan 2 compras, JBOSS la utiliza en menor porcentaje. Es decir para las medidas tomadas para JBOSS, el porcentaje se aproxima más a los 45 MBytes de uso exclusivo de Servidor de Aplicaciones

La diferencia dada entre los valores medidos en % de uso de memoria entre 5000 y 10000 usuarios cuando se realizan 2 compras y un registro no sobrepasa el 1.1% por lo que podemos concluir que no es muy significativo.

Cabe mencionar que la Java Virtual Machine, utiliza también una memoria virtual la cual trabaja con la memoria real pero en un porcentaje similar pero ofreciendo un mayor porcentaje de memoria libre asignada.

Los valores medidos mantienen un rango que varía entre 31,566 y 26,688, lo que nos da una diferencia de 4,88%, en todas las mediciones realizadas, tanto GLASSFISH como JBOSS, por lo que podemos advertir que el trabajo de memoria no variará con el incremento de usuarios, esto se debe a que los códigos y direccionamientos en los recursos utilizados se mantienen en la memoria, mientras que se realicen las mismas actividades, actuando en mayor cantidad el nodo que contiene a la Base de Datos.

Los servidores de Aplicaciones tienen un valor umbral correspondiente al 80% de uso de uso, el cual es administrado eficientemente por sus consolas de administración.

Podemos observar también que en la mayoría de los resultados el Servidor de Aplicaciones GLASSFISH presenta ligeramente un mayor porcentaje de uso de memoria, esto se da debido a la cantidad de parámetros que se mantienen de recursos, especialmente configuraciones del servicio de Nombre JNDI en la carga de los EJBs, así como también en lo que respecta al pool de conexiones para JBOSS las invocaciones son realizadas una sola vez y estos son almacenados en archivos que son leídos de disco, GLASSFISH los sube a memoria en cada llamada a los EJBs.

3.3.3. COMPARACIÓN DEL ÍNDICE RED

Para el análisis de los datos correspondientes a la red debemos tomar en cuenta sucesos que acontecieron durante las mediciones:

- Todos los datos fueron tomados en el mismo intervalo de duración, 10 minutos, aunque no todas las transacciones pudieron culminar su labor debido a la cantidad de peticiones en el intervalo indicado
- Los procesos necesarios para 100 usuarios tardaron un intervalo menor a 10 minutos, aproximadamente 4, en finalizar.
- Los procesos para 1000 usuarios ocuparon prácticamente la totalidad del intervalo de 10 minutos
- Los procesos para 5000 y 10000 usuarios tardaron una cantidad de tiempo muy superior a los 10 minutos, por lo que solo se pudo concluir una de las tareas, el registro, mientras que para la compra de productos no lograron ser medidos en su totalidad.
- Los valores calculados para el parámetro estadístico de desviación estándar es demasiado elevado, esto es debido a que está realmente representado y evaluado el tiempo de espera del servidor cuando no se hace petición Web, en este tiempo el usuario está realizando la compra, por lo que da valores demasiado bajos en la transferencia de bytes.

Existen 2 enfoques que se deben analizar dados debido al tiempo que tardan ambos servidores en finalizar las tareas. Cuando observamos los resultados obtenidos entre 100 y 1000 usuarios, debemos tomar en cuenta que las tareas si finalizan antes de cumplirse el intervalo de medición de 10 minutos, mientras que si observamos los resultados obtenidos

entre 5000 y 10000 usuarios este tiempo es sobre pasado. Por lo que el enfoque es distinto en las dos perspectivas.

Los datos obtenidos en cuanto a promedio de kbytes transmitidos y recibidos por parte de los servidores de aplicaciones reflejan a manera general que tiende a presentar la misma tendencia respecto al aumento de usuarios y tareas a ejecutar, se puede ver de manera clara como mientras el número de usuarios va en aumento los kbytes transmitidos y recibidos lo hace de igual manera y nunca superan los 3000 kbytes promedio en el tiempo establecido para la medición (20 segundos), la diferencia marcada entre la carga de 100 y 1000 usuarios por lo servidores varia muy poco, con una carga de 1000 a 5000 usuarios el cambio es más notable pero conservando el estilo de aumento, con 5000 y 10000 usuarios ese estilo no cambiaria y es lo más lógico, pero debido a que el tiempo establecido para la medición no permite tomar los datos totales solo se representan los que están en ese rango.

Desde el punto de vista de los resultados obtenidos entre 100 y 1000 usuarios, que son las medidas que finalizan antes del intervalo de medición, podemos analizar el crecimiento y la relación entre los datos obtenidos por ambos servidores.

Tomando en cuenta que los resultados muestran que JBOSS tiende a enviar más KBytes en promedio por intervalo, establecemos una relación de promedios:

$$\frac{KbytesTransmitidosJBOSS}{KbytesTransmitidosGlassfish} \quad y \quad \frac{KbytesRe recibidosJBOSS}{KbytesRe recibidosGlassfish}$$

Nos da una relación de crecimiento de:

Kbytes Recibidos:

		TAREAS		
		1r - 1c	1r - 2c	1r - 3c
Usuarios	Relación	Kbytes Rx JB/ Kbytes Rx GF		
100		1,278	1,24	1,21
1000		1,1	1,055	1,04

Tabla 10. Relación de Kybtes Recibidos JBOSS frente a GLASSFISH

Kbytes Transmitidos:

		TAREAS		
		1r - 1c	1r - 2c	1r - 3c
Usuarios	Relación	Kbytes Tx JB/ Kbytes Tx GF		
100		1,12	1,08	1,072
1000		1,04	1,02	1,019

Tabla 11. Relación de Kybtes Transmitidos JBOSS frente a GLASSFISH

Donde

1r - 1c: carga = 1 registro – 1 compra

1r - 1c: carga = 1 registro – 2 compras

1r - 1c: carga = 1 registro – 3 compras

Analizando las relaciones dadas entre GLASSFISH y JBOSS con estos parámetros notamos que mientras aumenta la carga en una misma cantidad de usuarios la relación disminuye aunque en un mínimo porcentaje lo que indica que GLASSFISH aumenta el flujo de Kbytes en los intervalos, llegando en algún momento a superar el número de Kbytes tanto Recibidos como Transmitidos cuando la carga sea aún mucho mayor. Estos resultados los vemos reflejados cuando el número de peticiones bordee los 5000 usuarios y es más notorio cuando GLASSFISH supera en flujo de bytes a JBOSS cuando medimos a 10000 usuarios, donde la relación será menor que 1.

Estas relaciones como vemos se mantienen en todas las instancias tomando en cuenta que el aumento de carga. y que los datos resultantes son mediciones de cada 20 segundos.

Por lo tanto podemos concluir también que:

Utilizando los picos máximos la velocidad de transferencia en la red es:

GLASSFISH (1r - 3c, 5000 usuarios)

Velocidad tx = Kbytes transmitidos / tiempo transferencia

Velocidad tx = 2390,44 / 20 segundos

Velocidad tx = 119,522 kbytes por segundo

JBOSS (1r - 3c, 5000 usuarios)

Velocidad tx = Kbytes transmitidos / tiempo transferencia

Velocidad tx = 2166,6 / 20 segundos

Velocidad tx = 108,33 kbytes por segundo

Por lo que podemos concluir que la velocidad máxima que bordea la transferencia de datos esta aproximadamente entre los 120 kbytes por segundo para GLASSFISH y 110 kbytes por segundo en JBOSS

Utilizando los picos máximos la velocidad de recepción en la red es:

GLASSFISH (1r - 3c, 5000 usuarios)

Velocidad tx = Kbytes transmitidos / tiempo transferencia

Velocidad tx = 3117,82 / 20 segundos

Velocidad tx = 155,89 kbytes por segundo

JBOSS (1r - 3c, 5000 usuarios)

Velocidad tx = Kbytes transmitidos / tiempo transferencia

Velocidad tx = 2746,9/ 20 segundos

Velocidad tx = 137,34 kbytes por segundo

Por lo que podemos concluir que la velocidad máxima que bordea la transferencia de datos esta aproximadamente entre los 156 kbytes por segundo para GLASSFISH y 138 kbytes por segundo en JBOSS.

Por lo tanto según este parámetro el Servidor de Aplicaciones GLASSFISH es mejor en el flujo de Bytes.

3.3.3.1. Promedio Kbytes Transmitidos Y Recibidos

Los datos obtenidos en cuanto a promedio de kbytes transmitidos y recibidos por parte de los servidores de aplicaciones reflejan a manera general que tiende a presentar la misma tendencia respecto al aumento de usuarios y tareas a ejecutar, se puede ver de manera clara como mientras el número de usuarios va en aumento los kbytes transmitidos y recibidos lo hace de igual manera y nunca superan los 3000 kbytes promedio en el tiempo establecido para la medición (20 segundos), la diferencia marcada entre la carga de 100 y 1000 usuarios por lo servidores varia muy poco, con una carga de 1000 a 5000 usuarios el

cambio es más notable pero conservando el estilo de aumento, con 5000 y 10000 usuarios ese estilo no cambiaría y es lo más lógico, pero debido a que el tiempo establecido para la medición no permite tomar los datos totales solo se representan los que están en ese rango.

3.3.3.2. Rendimiento Kbytes Transmitidos Y Recibidos

El interpretar los valores obtenidos para el rendimiento de la red esta de acuerdo al promedio de Kbytes transmitidos y recibidos entre el intervalo planificado (10 minutos) lo cual significa que obtenemos un rendimiento de los valores obtenidos en rangos de 20 segundos cuantificados en el total de la medición.

$$RendimientoTX = \frac{\text{promedio}(KBytesTx_en_20segundos)}{600segundos}$$

$$RendimientoRX = \frac{\text{promedio}(KBytesRx_en_20segundos)}{600segundos}$$

Por lo tanto el rendimiento es directamente proporcional al promedio, es decir mientras más KBytes se logre enviar en un intervalo de tiempo el servidor obtiene un mayor rendimiento, es por eso que se puede observar que mayor rendimiento se genera cuando se obtiene los valores de 5000 y 10000 usuarios donde se requiere un mayor flujo de KBytes, se puede observar también que cuando se recoge los valores de 5000 usuarios la tendencia cambia en los resultados observando que GLASSFISH obtiene un mayor rendimiento debido al aumento en el flujo de KBytes por parte del servidor de aplicaciones.

Los cambios en estas medidas se deben a factores que a continuación se detallan.

Un rendimiento optimo en cuanto a los servidores de aplicaciones esta en mayor porcentaje vinculado a como el sistema operativo maneja de manera predeterminada el recurso de red, en sistemas operativos UNIX existen varios parámetros que permiten mejorar el rendimiento por ende tal configuración lo asumirá los servidores, para sistemas operativos WINDOWS la configuración de la red definida en los servidores de aplicaciones es la más optima por lo que al final esto es reflejado con los datos recolectados en las medidas.

Otro factor que influye son los servicios que manejan los servidores por defecto, esto es ya que cada servicio tiene un espacio asignado de bits en el paquete de transmisión de los datos si dicho servicio es cargado por defecto y sin un uso por parte de este, al final en la transmisión de los datos habrá diferencia en el consumo del canal de transferencia o ancho de banda de la red, pero si la carga de datos es menor no tendera a marcar mucha disconformidad.

De lo anteriormente explicado la otra posibilidad es que si la carga de datos acrecentara la relación de transmisión y recepción tiende a acotarse entre ambos servidores, llegara a un punto que el servidor GLASSFISH enviara y recibirá más datos que el servidor JBOSS y esto se debe a que en el caso del servidor JBOSS la forma de cómo cada paquete de datos lo maneja depende de los bits de control, con este servidor a cada paquete se le asigna uno o varios como método de control y seguimiento, con el servidor GLASSFISH la situación cambia a cada paquete de datos no se coloca bits de control más bien el servidor delega paquetes de control a un determinado conjunto de paquetes enviados lo que conlleva a que con una agrupación grande de datos este paquete pese mucho más que lo haría los bits de manera individual, por lo que JBOSS a manera global es eficiente en el manejo y control de los paquetes, aunque GLASSFISH presenta un mejor rendimiento en el flujo de información.

3.3.4. TIEMPOS DE RESPUESTA

Los resultados obtenidos en tiempos de respuesta reflejan el tiempo que tarda el servidor de aplicaciones en responder completamente una petición Web, por lo tanto podemos relacionarlo directamente este parámetro con la satisfacción del usuario.

Como demuestran las gráficas es lógico observar que el tiempo de respuesta aumenta mientras más usuarios realicen concurrentemente una petición Web.

Para efectos de comparación en los tiempos de respuesta se han tomado medidas con 1, 50, 100, 500, 1000, 5000, y 10000 usuarios para observar la variación y poder comparar ambos servidores

Si nos ubicamos en un rango de satisfacción del usuario esperamos que el servidor muestre una respuesta lo más pronto posible, es decir que la respuesta sea entregada hasta en un

intervalo de 15 segundos como máximo, a partir de este tiempo la satisfacción del usuario decrece perjudicando los objetivos de una aplicación Web. En el caso de nuestro prototipo se observa que si se atiende concurrentemente a una cantidad de 500 usuarios realizando peticiones Web comunes (un registro y una compra) este tiempo llega a los 18 segundos aproximadamente, lo que sobrepasa las expectativas de un usuario por lo tanto se puede concluir que el mayor rendimiento que el servidor de aplicaciones ofrece se da entre 100 y 500 usuarios para este tipo de tareas. Cuando se analiza una variación en las tareas (un registro dos compras) se observa que en un número de 500 usuarios, lo que representa 1500 peticiones atendidas el tiempo de respuesta asciende a los 37,3 y 45,7 segundos de los servidores GLASSFISH y JBOSS respectivamente, valores que se encuentran fuera del rango de las expectativas de los usuarios por lo que se puede asumir que la satisfacción del usuario es mayor aproximadamente cuando se atienden a 100 usuarios es decir 300 peticiones Web. Finalmente en los valores obtenidos para las tareas (un registro tres compras) se observa la misma tendencia es decir aproximadamente cuando se atiende a 100 usuarios concurrentes, es decir 400 peticiones Web se da el mayor rendimiento en lo que a tiempos de respuesta concierne.

Cuando se observa los resultados desde 1000 usuarios en adelante realizando una petición Web concurrente se observa un exagerado aumento de tiempo de respuesta en ambos servidores lo que demuestra que para atender un número significativo de usuarios se debe realizar nuevas configuraciones administrativas a los servidores o incluso contar con un servidor con unas características más adecuadas en Hardware para soportar dicha carga; Dado que nuestros objetivos están basados en la comparación de los servidores manteniendo un mismo ambiente no se han realizado dichas configuraciones y han sido realizadas en un solo servidor con características mencionadas en el capítulo II.

En términos de comparación de tiempos de respuesta las gráficas obtenidas demuestran que el servidor de aplicaciones GLASSFISH responde en menor tiempo a todas las peticiones Web (registro y compras) sin importar la cantidad de tareas a realizar y la carga de usuarios que interactúan concurrentemente en comparación con el servidor JBOSS.

Para demostrar que el servidor de aplicaciones GLASSFISH responde de manera más rápida que JBOSS se utilizara muestras de los resultados obtenidos para determinar una

velocidad aproximada de los servidores con respecto a la carga, es decir obtendremos la velocidad con la siguiente formula:

$$\text{Velocidad}_{\text{ del servidor}} = \frac{\text{peticiones}_{\text{ atendidas}}}{\text{tiempo}}$$

			SERVIDORES DE APLICACIONES	
			GLASSFISH	JBOSS
Usuarios	Total Peticiones Atendidas	Tareas	Peticiones Atendidas/ Tiempo(seg.)	
50	100	1r-1c	51,21	52,45
	150	1r-2c	58,71	48,49
	200	1r-3c	58,58	48,57
100	200	1r-1c	65,31	57,79
	300	1r-2c	59,36	53,70
	400	1r-3c	57,08	48,08
500	1000	1r-1c	55,56	42,15
	1500	1r-2c	40,21	32,79
	2000	1r-3c	29,93	25,76
5000	10000	1r-1c	16,15	15,98
	15000	1r-2c	7,07	6,69
	20000	1r-3c	4,55	4,38

Tabla 12. Velocidad de Recepción de Peticiones Web, Peticiones atendidas en un intervalo de tiempo

Donde

1r - 1c: carga = 1 registro – 1 compra

1r - 2c: carga = 1 registro – 2 compras

1r - 3c: carga = 1 registro – 3 compras

Con el conjunto de datos obtenidos de la comparación entre los servidores por medio de la carga de usuarios y el tipo de tareas que se ejecutan se puede comprender como actúa la velocidad del servidor, con una carga mayor a 50 y menor a 100 usuarios se aprecia que la

velocidad realizada por los servidores de aplicaciones tiende a aumentar, con una carga de 100 usuarios se presenta el máximo de peticiones que los servidores pueden atender y con carga de 500 usuarios o más tiende a disminuir por lo que acorde al numero total de peticiones los servidores disminuyen su tiempo de respuesta y eso es lo más lógico que suceda.

Comparando los datos entre los servidores de aplicaciones se da cuenta que la velocidad de GLASSFISH en un rango mínimo es superior a lo que realiza JBOSS, y con la carga de 100 usuarios e independiente de las tareas se visualiza lo máximo que rinden en ambos y por ende GLASSFISH es más rápido.

CAPÍTULO 4. CONCLUSIONES Y RECOMENDACIONES

4.1. CONCLUSIONES

1. En cuanto a la instalación de los servidores de aplicaciones se concluye que JBOSS presenta un cuadro más detallado referente a las características que este brinda, mientras que GLASSFISH lo hace de manera más genérica evitando que el administrador tome dediciones específicas que las podrá modificar en la configuración de la aplicación, haciendo que este proceso sea más fácil de manejar.
2. En la configuración de los servidores los administradores se ven envueltos en distintos ambientes. JBOSS requiere de conocimiento y seguimiento de un conjunto de archivos planos los que contienen variables que una aplicación requiere para ser desplegada, en contraparte GLASSFISH presenta un ambiente amigable que si requiere de conocimiento de variables aunque ellas se encuentran organizadas y de fácil acceso. Por lo tanto el tiempo que un administrador requiere para configurar un ambiente de despliegue de aplicaciones es menor con el servidor GLASSFISH que con JBOSS.
3. Ambos servidores presentan diferentes esquemas en sus arquitecturas para el despliegue de aplicaciones, de tal manera que soportan la especificación J2EE lo que mantiene un mismo conjunto de servicios como mensajes, manejo de transacciones, contenedores EJBs y muchos más por lo que ambos servidores enfrentan de manera optima grandes cargas de aplicaciones en ambientes reales de trabajo.
4. Los servidores de aplicaciones se pueden obtenerlos gratuitamente bajándolos de Internet, en cuanto a soporte se da una diferencia: GLASSFISH posee una mayoritaria comunidad de expertos y diseñadores que ofrecen un soporte de fácil acceso y en tiempo real tanto para el desarrollo de nuevas versiones del servidor como para la elaboración de aplicaciones, en forma gratuita. JBOSS en cambio al ser parte de Red Hat brinda asistencia técnica que representa un costo para el usuario, dependiendo del tipo de asistencia que este requiera se ve reflejado el costo.
5. Al basarse en una plataforma bajo la especificación J2EE el desarrollo del portal Web como prototipo de pruebas es el mismo para ambos servidores, la diferencia se da en cuanto a la gestión de dicha aplicación.
6. En base a los análisis realizados de los resultados del benchmark en cuanto al parámetro de % de uso de la CPU por parte del Sistema Operativo se concluye que el servidor de aplicaciones JBOSS presenta una ligera superioridad en el rendimiento, se

debe tomar en cuenta que debido al bajo porcentaje de uso de este parámetro, este no afecta en el tiempo de respuesta de la aplicación.

7. Los resultados arrojados por el benchmark en cuanto al parámetro de % de uso de la CPU por parte del servidor de aplicaciones demuestran que GLASSFISH en un margen muy pequeño supera a JBOSS, de igual manera se toma en cuenta que este parámetro no afecta en gran medida al despliegue de la aplicación ya que como se pudo observar en los respectivos análisis esta por debajo de los valores umbrales.
8. Por la constancia en los valores del porcentaje de uso de la memoria se concluye que este no es afectado por el incremento de usuarios. JBOSS genera un mayor rendimiento en memoria no muy notorio que GLASSFISH que no variara mucho por el tamaño de la aplicación. Dado que estos valores son casi constantes el rendimiento en este parámetro dependerá de la capacidad del hardware del equipo más no en si de la gestión misma que pueda ejecutar los servidores.
9. Dado el notorio cambio de los resultados obtenidos por el benchmark para la medición del parámetro red en transmisión de bytes por ambos servidores, este resulta ser el de mayor complejidad en cuanto a su análisis ya que sus resultados finales muestran en unos casos a JBOSS brindar mejor rendimiento donde la carga de usuarios es menor pero con un aumento de estos la tendencia cambia y el mejor es GLASSFISH. Por ende GLASSFISH trata de mejor manera la transmisión de datos cuando la carga es mayor y esto repercute en tiempo de respuesta a los usuarios finales.
10. En cuanto al parámetro red en recepción de bytes se tiene un mismo comportamiento del rendimiento que en la transmisión de datos lo que demuestra que cada servidor tiene un tratamiento distinto para el flujo de bytes en general y este tratamiento hará la diferencia al momento que exista mayor concurrencia de usuarios.
11. Para tiempos de respuesta aunque la diferencia obtenida por ambos servidores es mínima, el que obtiene mejor rendimiento de respuesta de las peticiones Web ya sean registro de usuarios y compra de productos es GLASSFISH. Este parámetro influye directamente en el grado de satisfacción del usuario lo que implica que a menor tiempo de respuesta en cada petición Web mayor grado de satisfacción por lo tanto se ve que GLASSFISH es el que mejor cumple con las expectativas del usuario.

4.2. RECOMENDACIONES

1. Considerar el o los equipos destinados para la instalación de los servidores de aplicaciones es muy importante debido a que cada uno de estos necesariamente deben cumplir ciertos requisitos mínimos en cuanto a hardware, además se debe tener en cuenta en el ambiente que operarán, el uso de estos tendrá relevancia y podrá marcar mucha diferencia a futuro.
2. Se recomendaría para investigaciones futuras realizar benchmark que se enfoquen en la nueva plataforma de especificación JEE5 que es la evolución de J2EE con servidores de aplicaciones acordes a esta nueva tendencia.
3. Llevar a cabo una investigación de rendimientos en servidores de aplicaciones utilizando software libre en su totalidad arrojaría nuevos resultados factibles de comparación, lo que daría una nueva perspectiva para beneficio del administrador.
4. Otra fuente de investigación podría orientarse a los servicios que prestan cada uno de los servidores en particular ya que muchas veces los administradores requieren la utilización de ciertos servicios más que otros.
5. Para realizar Benchmarks como cargas mucho más significativas se recomienda utilizar granjas de servidores para aprovechar los beneficios de cluster, instancias y nodos que estos servidores brindan.
6. Cuando se trate del manejo de una aplicación por parte del servidor de aplicaciones puede existir dos escenarios dependiendo de la cantidad de usuarios y aplicaciones que vayan a interactuar, si en el caso de que la interacción la carga a manejar será de grandes proporciones lo más aconsejable es utilizar GLASSFISH, la otra situación es que si esta carga es moderada lo más factible es JBOSS, esto es aplicable cuando el manejo del sistema dependa exclusivamente de una instancia del servidor.
7. Se recomienda a los administradores utilizar cada uno de los servicios que brindan los servidores para aprovechar su bajo uso en CPU y MEMORIA cuando se tiene una aplicación más robusta o se necesite de otras aplicaciones interactuando en el mismo servidor.

BIBLIOGRAFIA

Libros

- [1] ABARCA. C; Miguel, Donoso, Gustavo. Manual de Desarrollo Básico de Aplicaciones En la Plataforma J2EE En Ubuntu 7.x.
- [2] CAMP, Robert; Benchmarking; Primera edición; Editorial Panorama Editorial, S.A.; 1993.
- [3] CID, Jaime; Servidores de Aplicación Arquitectura y planificación Análisis de Mercado; S/EM; Febrero, 2006.
- [4] FLEURY, Mark; STARK, Scott; NORMAN, Richards. JBoss 4.0 The Official Guide. Editorial Sams Publishing. USA. Año 2005.
- [5] HEFFELFINGER, David. Java EE 5 Development using GlassFish Application Server. Editorial Packt Publishing. USA. Año 2007.
- [6] LINDGREN, M. Lisa. Application Server for E-Business. 1ra edición. Editorial Auerbach. USA. Año 2001.
- [7] PALACIO, Juan Bañeres. Compendio de Ingeniería de Software I. Rev 0.04. Junio 2006
- [8] PERRY, J Steven. Java Managment Extensions. Editorial O'Reilly. Año 2002
- [9] PRESSMAN, Roger; Ingeniería de Software un enfoque práctico. Mc Graw Hill, 5ta Edición, Febrero 2002.
- [10] RUMBAUGH, James; JACOBSON, Ivar; BOOCH, Grady. El Proceso Unificado de Desarrollo. 1ra edición. Editorial Addison-Wesley. España. Año 2000.

- [11] SHLARMAN, Craig; UML y Patrones. Introducción al Diseño orientado a objetos. Editorial Prentice Hall USA. Año 2003.
- [12] SUN Microsystems; Sun Java Enterprise System 5 Monitoring Guide; United States; 2006.
- [13] TRANSACTION Processing Performance Council (TPC); TPC Benchmark App; Versión 1.3; febrero, 2008

Referencias Web

- [14] BARRY & Associates, Application Server Definition. http://www.service-architecture.com/application-servers/articles/application_server_definition.html
- [15] CSI. Consejo Superior de Administración Electrónica; Recomendaciones Aplicables a la contratación de Equipos Informáticos. http://www.csi.map.es/csi/pdf/recomendaciones_contratacion.pdf; 2008
- [16] GLASSFISH WIKI. Glassfish V2 Architecture. <http://wiki.glassfish.java.net/Wiki.jsp?page=GlassFishV2Architecture>.
- [17] INTXAUBURU, Miren Gurutze; OCHOA, Carlos; VELASCO, Eva; DIALNET; ¿Es el Benchmark una herramienta de aprendizaje organizacional?. http://dialnet.unirioja.es/servlet/fichero_articulo?codigo=2499425&orden=; Septiembre, 2005.
- [18] JAVA Boutique. Explore the World of Application Servers. <http://javaboutique.internet.com/articles/AppServers/>. 2007
- [19] JBOSS.com. Jboss Enterprise Portal Platform. http://www.jboss.com/pdf/jb_portal_platform_04_07.pdf.
- [20] JBOSS.org. The JBoss 4 Application Server Guide. <http://docs.jboss.org/jbossas/jboss4guide/r5/html/index.html>. AÑO 2006

- [21] JGROUPS. A Toolkit for Reliable Multicast Communication.
<http://www.jgroups.org/overview.html>. 2002
- [22] KALALI, Masoud; GlassFish Application Server.
<http://refcardz.dzone.com/announcements/glassfish>.
- [23] LUI, Jenny; University Of Sydney. An Analysis of JBoss Architecture.
<http://www.cse.unsw.edu.au/~yliu/jboss/jBoss.htm>. Abril, 2002
- [24] LUNA, Octavio. JBOSS Servidor de Aplicaciones de fuente abierta.
http://www.rosenblueth.mx/InterFAR/Vol1Num5/extdoc/Octavio_Luna.htm.
2002
- [25] ORACLE Application server; Oracle® Application Server Performance Guide.
http://otndnld.oracle.co.jp/document/products/as10g/101300/B25221_03/core.1013/b25213.pdf
- [26] PEREIRA, Santiago. Instrumentación de componentes Java usando JMX.
<http://www.adictosaltrabajo.com/tutoriales/tutoriales.php?pagina=jmx>.
Noviembre 2005.
- [27] PÉREZ, Mariñán Martín, Molpeceres Alberto, JavaHispano. Arquitectura empresarial y Software Libre.
<http://www.willydev.net/InsiteCreation/v1.0/willicrawler/2008.05.02.articulo.arquitectura%20empresarial%20y%20software%20libre,%20j2ee.pdf>.
Noviembre 2002
- [28] RATIONAL Corporation. Rational Unified Process, Best Practices for Software Development Teams.
http://www.augustana.ab.ca/~mohrj/courses/2000.winter/csc220/papers/rup_best_practices/rup_bestpractices.html. 2004

- [29] REDHAT. Jboss Enterprise Application Platform Configuration Guide. http://www.redhat.com/docs/manuals/jboss/jboss-eap-4.3/doc/Server_Configuration_Guide/html/index.html. Noviembre, 2007
- [30] SERVER Watch; Server Compare. <http://www.serverwatch.com/stypes/compare/index.php/compare218239,18225>; 2008
- [31] SUN Microsystems. Application Server Commands and Concepts. <http://docs.sun.com/app/docs/doc/819-3671/ablaz?a=view>. 2008.
- [32] SUN Microsystems; Configuring Node Agents ; <http://docs.sun.com/source/819-0215/nodeagent.html>; Octubre, 2005
- [33] SUN Microsystems; Usage Profiles; <http://docs.sun.com/app/docs/doc/819-3671/gelvk?a=view>
- [34] SURTANI, Manik. Ban Bela. JBOSS.org. JBoss Cache User Guide. http://www.jboss.org/file-access/default/members/jboss-cache/freezone/docs/2.2.1.GA/userguide_en/html_single/index.html. Julio 2008.
- [35] THE World's Leading Java Resource; Micklea, Brad; VONA, Geoff; Using Metrics to Optimize J2EE Application Performance in Production; <http://java.sys-con.com/node/117726>; Agosto, 2005.
- [36] TOMICH, Jill, Refcardz. Glassfish Application Server Refcard. <http://refcardz.dzone.com/announcements/glassfish>, Julio 2008.
- [37] UNISYS. JBoss Application Server Migration Strategies. http://unisys.com/eprise/main/admin/corporate/doc/JBoss_Application_Server_Migration_Strategies_White_Paper.pdf. Octubre, 2005

- [38] WHITE, Larry, Kumar, Abhijit, SUN Microsystems. Clústeres de la version 2 de Glassfish.
<http://www.sun.com/bigadmin/hubs/multilingual/spanish/content/glassfishcluster.jsp>. 2008

ANEXOS

ANEXO A. TABLAS Y GRÁFICOS DE LA MEDICIÓN DE LOS RECURSOS

En esta sección se visualiza todos los datos obtenidos en el intervalo de medición asignado cuando el número de usuarios incrementa al igual que el número de tareas a ejecutar. Se comenzará por los datos y gráficos que representan el Promedio de cada uno de los recursos sometidos a la medición, posteriormente se representan los datos referentes a Rendimiento y se finaliza con la información referente a Tiempos de Respuesta.

Tareas: 1 Registro – 1 Compra

PROMEDIO								
	GLASSFISH				JBOSS			
Total Transacciones	200	2000	10000	20000	200	2000	10000	20000
USUARIOS	100	1000	5000	10000	100	1000	5000	10000
% Uso Sistema Operativo	0,1699	4,3066	8,5433	8,64	0,1566	2,01	8,3099	10,066
% Uso Servidor Aplicaciones	1,1066	1,9766	4,0533	4,9133	1,6066	2,2433	5,3633	5,8133
Memoria % en Uso	29,833	29,766	29,733	28,57	27,536	28,276	28,62	27,173
Memoria % Libre	70,066	70,133	70,166	71,329	72,363	71,623	71,28	72,726
Kbytes Transmitidos	33,025	399,46	1771,5	2216,9	37	415	2129,8	1937,3
Kbytes Recibidos	21,833	429,66	2140,4	2334,8	27,9	473,2	2678,8	1798,8

Tabla 13. Resultados de Promedios obtenidos por los servidores de aplicaciones al momento que los usuarios ejecutan las tareas 1 Registro y 1 Compra.

INFORMACION REFERENTE A PROMEDIOS

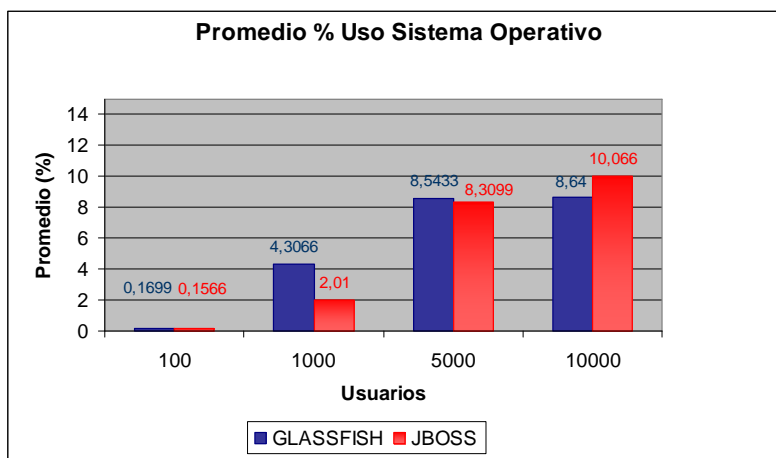


Figura 75 Valores referentes al promedio % Uso por parte del Sistema Operativo expuestos en la tabla 13.

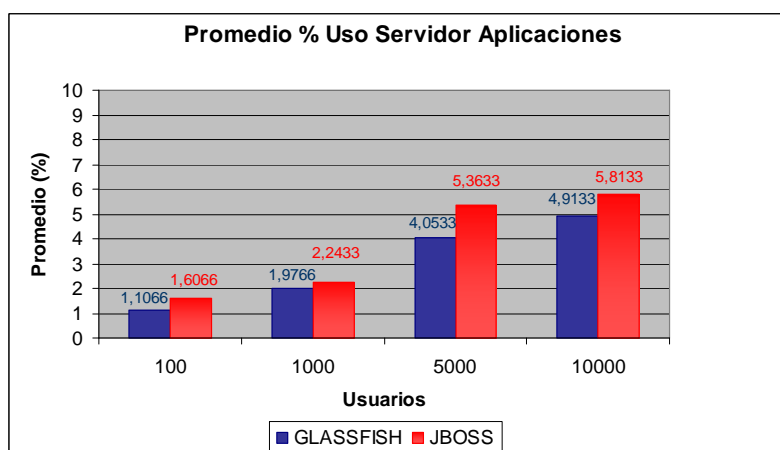


Figura 76. Valores referentes al promedio % Uso por parte del Servidor Aplicaciones 13.

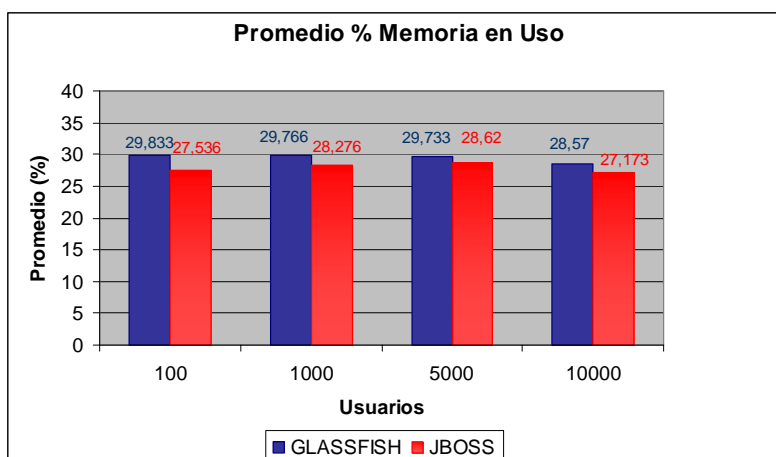


Figura 77 Valores referentes al promedio % Memoria en Uso expuestos en la tabla 13.

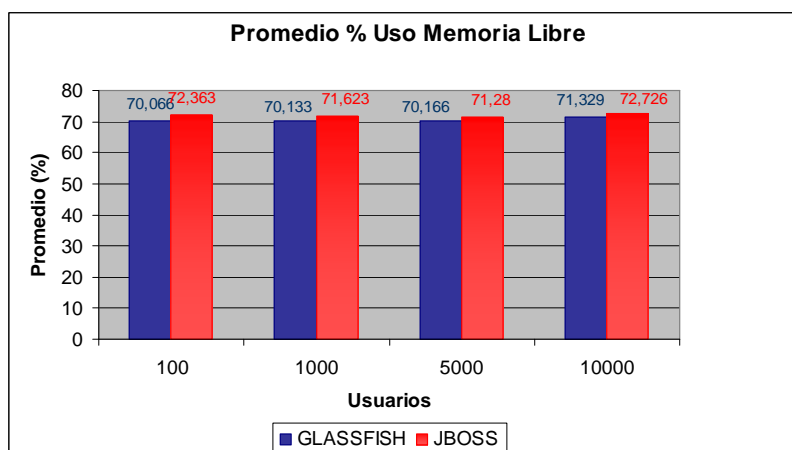


Figura 78. Valores referentes al promedio % Memoria Libre expuestos en la tabla 13.

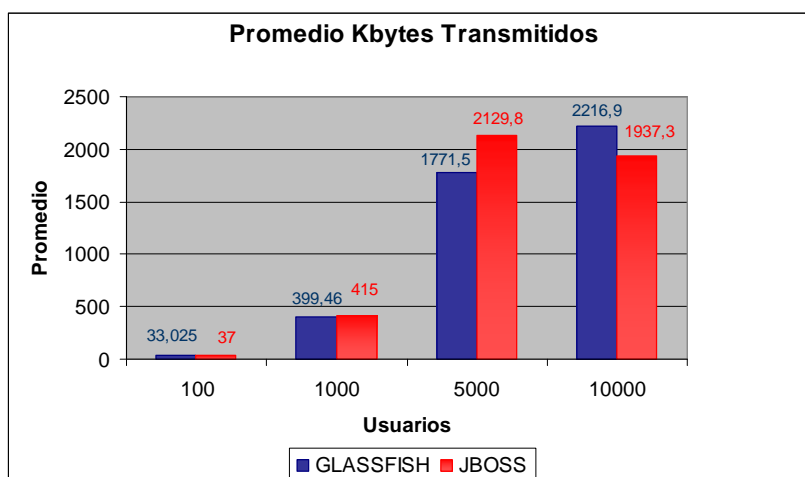


Figura 79. Valores referentes al promedio de KBytes Transmitidos expuestos en la tabla 13.

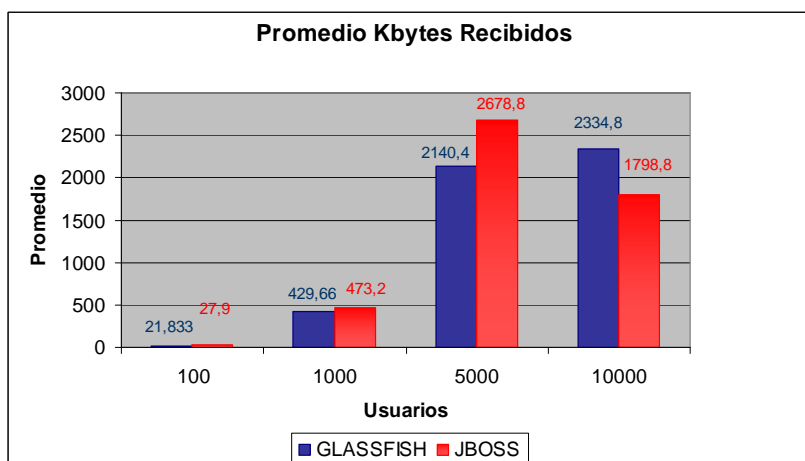


Figura 80. Valores referentes al promedio de KBytes Recibidos expuestos en la tabla 13.

Tareas: 1 Registro – 2 Compras

PROMEDIO								
	GLASSFISH				JBOSS			
Total Transacciones	300	3000	15000	30000	300	3000	15000	30000
USUARIOS	100	1000	5000	10000	100	1000	5000	10000
% Uso Sistema Operativo	0,2866	4,7466	9,76	9,9833	0,2033	4,3366	9,1733	10,836
% Uso Servidor Aplicaciones	1,3633	2,7399	4,6533	5,35	1,41	3,2666	5,4166	6,2099
Memoria % en Uso	31,576	31,043	26,976	30,699	28,789	27,669	28,95	30,85
Memoria % Libre	68,323	68,856	72,923	69,199	71,11	72,23	70,95	69,049
Kbytes Transmitidos	39,3	1015,7	2227,4	1995,4	42,833	1035	2159,9	1972,1
Kbytes Recibidos	35,9	1344,2	2849,6	1987,7	44,433	1418	2733,6	1860,2

Tabla 14. Resultados de Promedios obtenidos por los servidores de aplicaciones al momento que los usuarios ejecutan las tareas 1 Registro y 2 Compras.

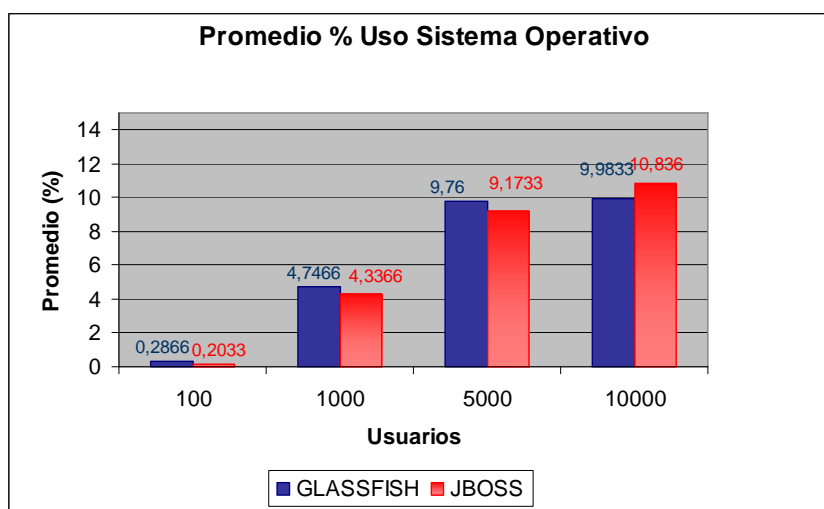


Figura 81. Valores referentes al promedio % Uso por parte del Sistema Operativo expuestos en la tabla 14.

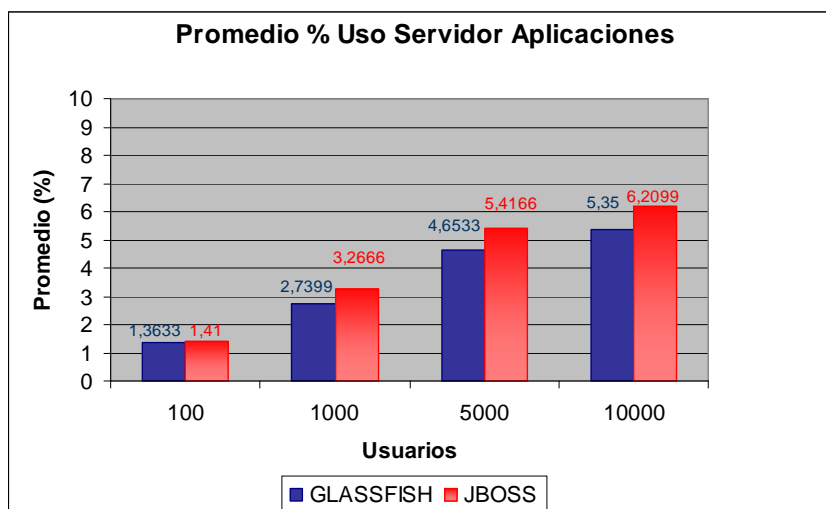


Figura 82. Valores referentes al promedio % Uso por parte del Servidor Aplicaciones expuestos en la tabla 14.

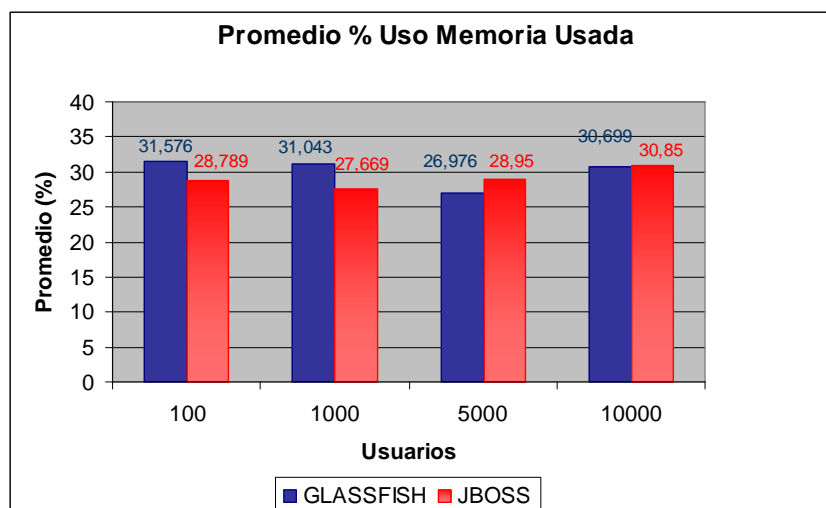


Figura 83. Valores referentes al promedio % Memoria en Uso expuestos en la tabla 14.

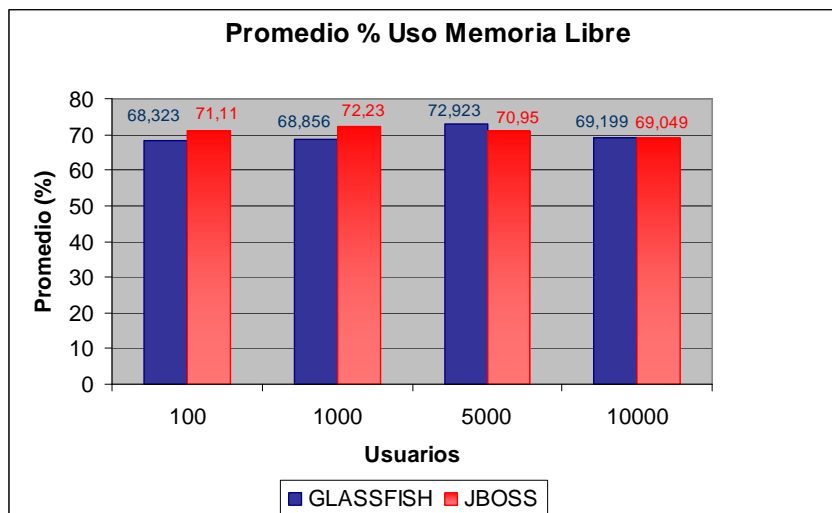


Figura 84. Representación gráfica de los valores referentes al promedio % Memoria Libre expuestos en la tabla 14.

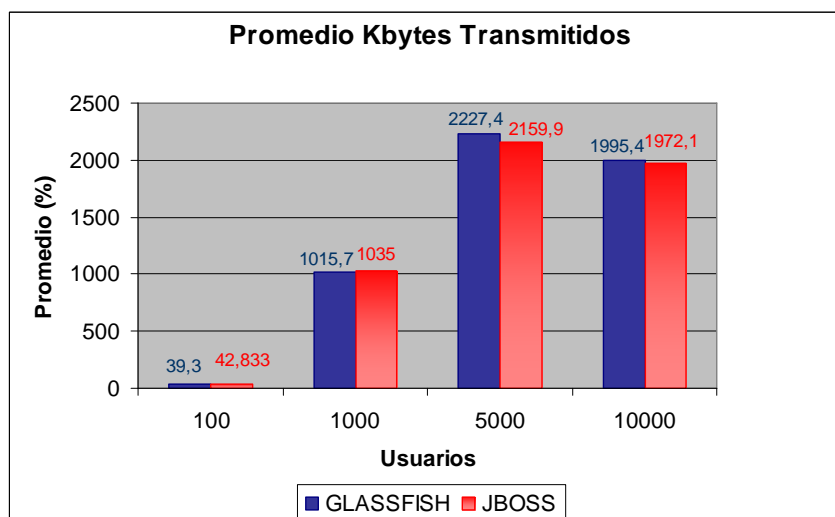


Figura 85. Representación gráfica de los valores referentes al promedio de KBytes Transmitidos expuestos en la tabla 14.

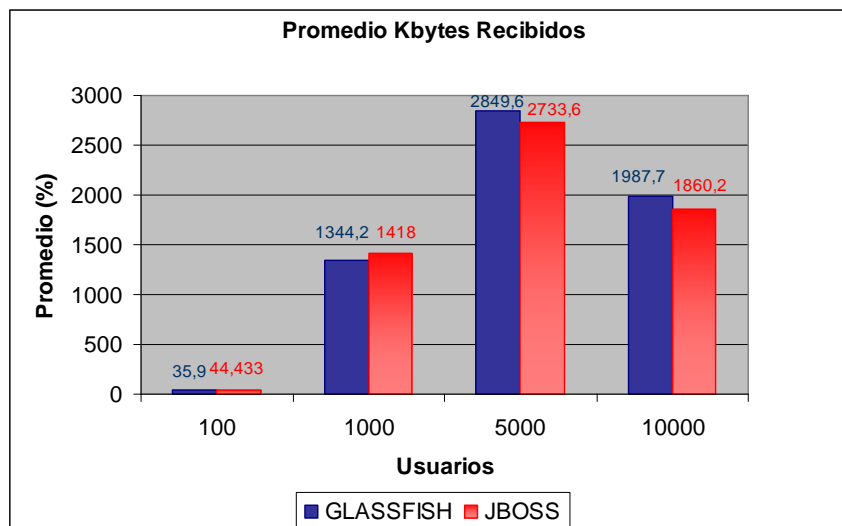


Figura 86. Valores referentes al promedio de KBytes Recibidos expuestos en la tabla 14.

Tareas: 1 Registro – 3 Compras

PROMEDIO								
	GLASSFISH				JBOSS			
Total Transacciones	400	4000	20000	40000	400	4000	20000	40000
USUARIOS	100	1000	5000	10000	100	1000	5000	10000
% Uso Sistema Operativo	0,5166	6,6766	9,2866	8,9599	0,5	4,5799	7,64	10,22
% Uso Servidor Aplicaciones	1,0466	3,9366	4,2833	5,44	1,1533	4,0699	5,0266	6,1899
Memoria % en Uso	31,62	30,906	30,916	28,64	30,723	28,323	26,683	27,549
Memoria % Libre	68,28	68,993	68,983	71,26	69,176	71,576	73,216	72,349
KBytes Transmitidos	56,366	1961	2390,44	2289,49	60,4	2001,5	2166,6	2036,8
KBytes Recibidos	56,033	2849,4	3117,82	2445,56	67,666	2961,8	2746,9	1961,4

Tabla 15. Resultados de Promedios obtenidos por los servidores de aplicaciones al momento que los usuarios ejecutan las tareas 1 Registro y 3 Compras.

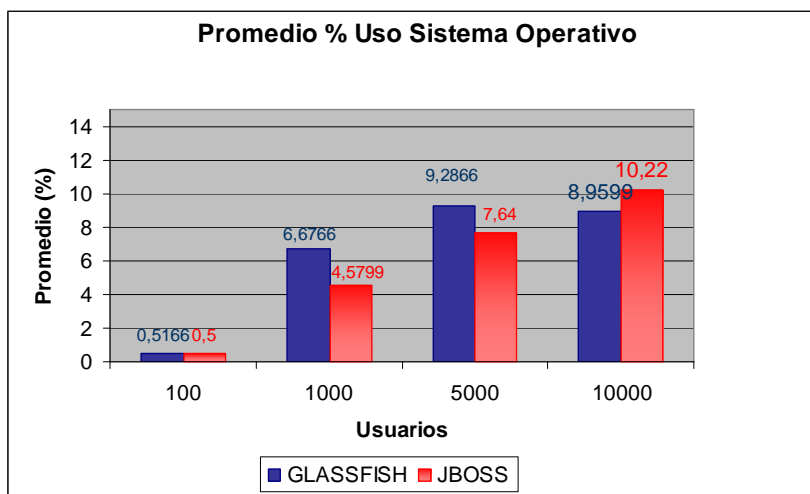


Figura 87. Valores referentes al promedio % Uso por parte del Sistema Operativo expuestos en la tabla 15.

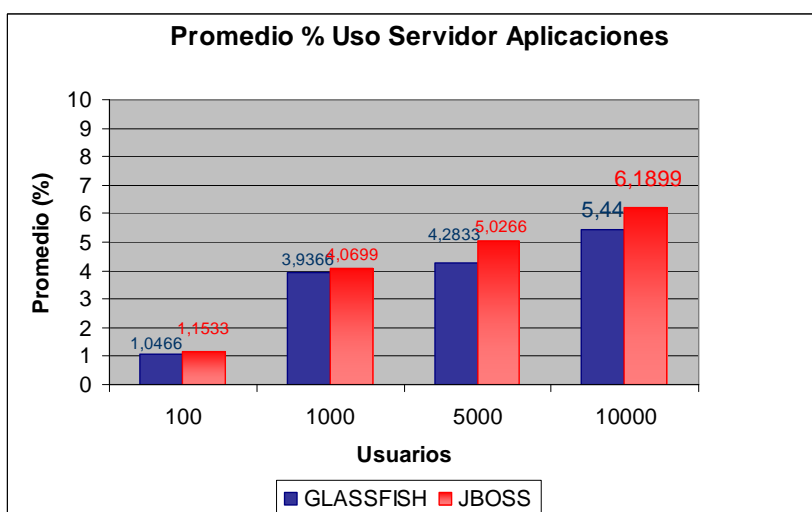


Figura 88. Valores referentes al promedio % Uso por parte del Servidor Aplicaciones expuestos en la tabla 15.

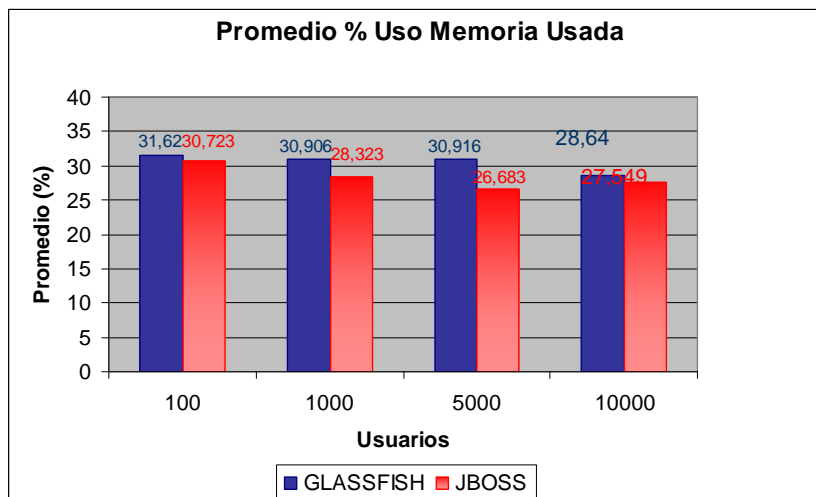


Figura 89. Valores referentes al promedio % Memoria en Uso expuestos en la tabla 15.

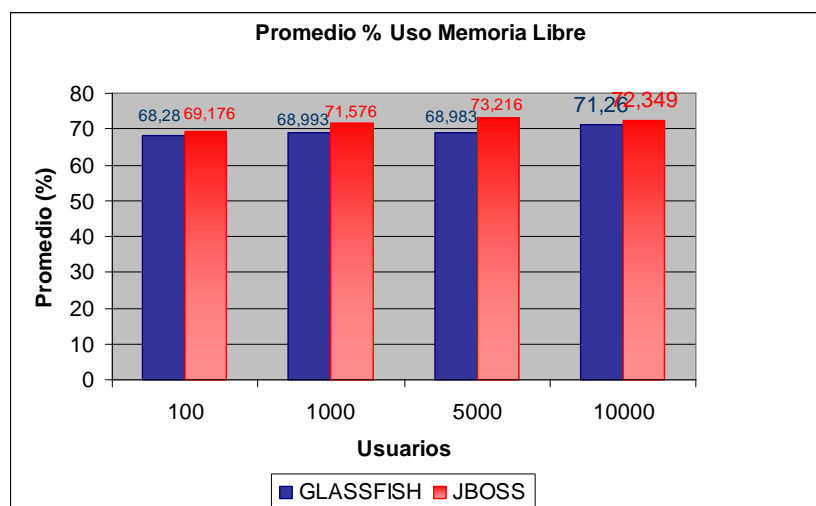


Figura 90. Valores referentes al promedio % Memoria Libre expuestos en la tabla 15.

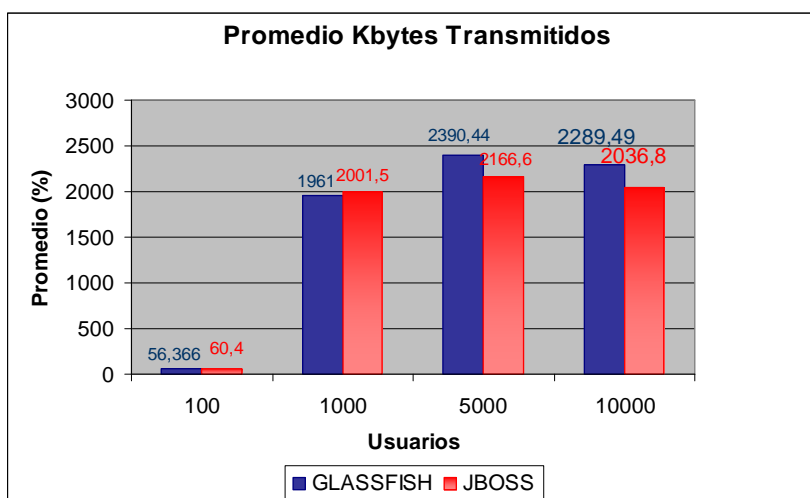


Figura 91. Valores referentes al promedio de KBytes Transmitidos expuestos en la tabla 15.

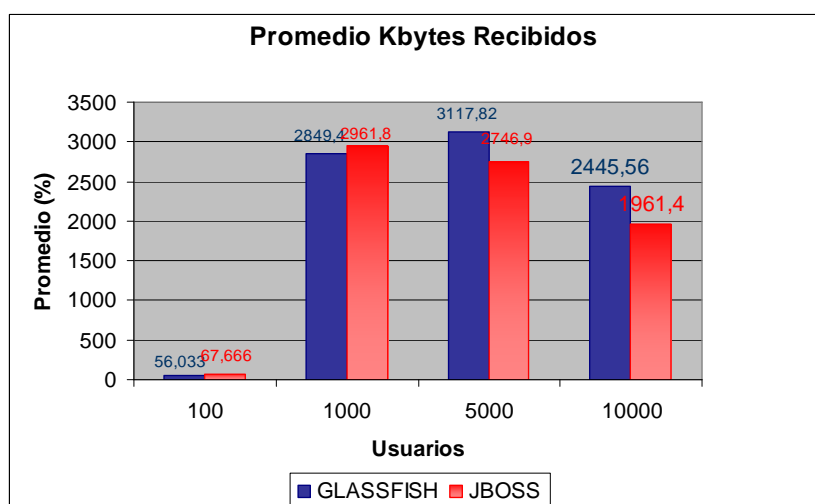


Figura 92. Valores referentes al promedio de KBytes Recibidos expuestos en la tabla 15.

Al igual que lo expuesto para la medida del promedio se indican los datos y gráficas que presentaran el rendimiento que cada uno de los recursos mostró al ser medidos.

INFORMACION REFERENTE A RENDIMIENTOS

Tareas: 1 Registro –1 Compra

RENDIMIENTO								
	GLASSFISH				JBOSS			
Total Transacciones	200	2000	10000	20000	200	2000	10000	20000
USUARIOS	100	1000	5000	10000	100	1000	5000	10000
% Uso Sistema Operativo	10,758	0,4246	0,214	0,2116	11,674	0,9099	0,22	0,1816
% Uso Servidor Aplicaciones	1,6527	0,9252	0,4512	0,3722	1,1383	0,8153	0,341	0,3146
Memoria % en Uso	22,357	22,407	22,432	23,346	24,222	23,588	23,305	24,546
Memoria % Libre	9,5195	9,5104	9,5059	9,3509	9,2173	9,3126	9,3574	9,1713
KBytes Transmitidos	0,0555	0,6657	2,9526	3,6949	0,0616	0,6916	3,5497	3,2288
KBytes Recibidos	0,0363	0,7161	3,5673	3,8913	0,0465	0,7886	4,4646	2,998

Tabla 16. Resultados de Rendimientos obtenidos por los servidores de aplicaciones al momento que los usuarios ejecutan las tareas 1 Registro y 1 Compra.

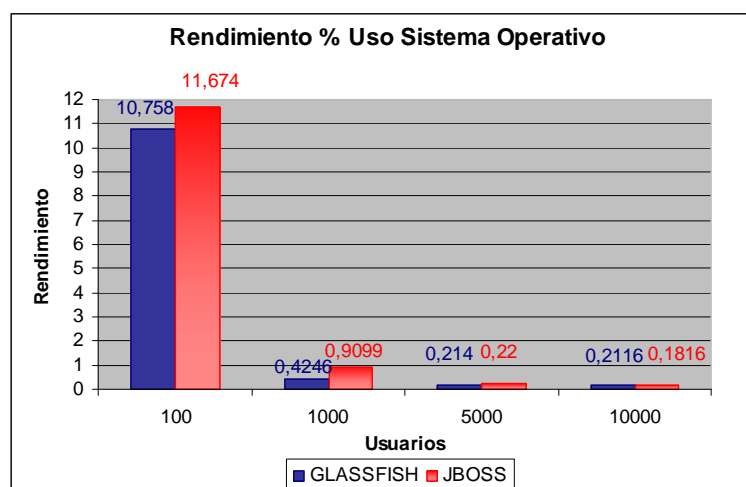


Figura 93. Valores referentes al rendimiento % Uso por parte del Sistema Operativo expuestos en la tabla 16.

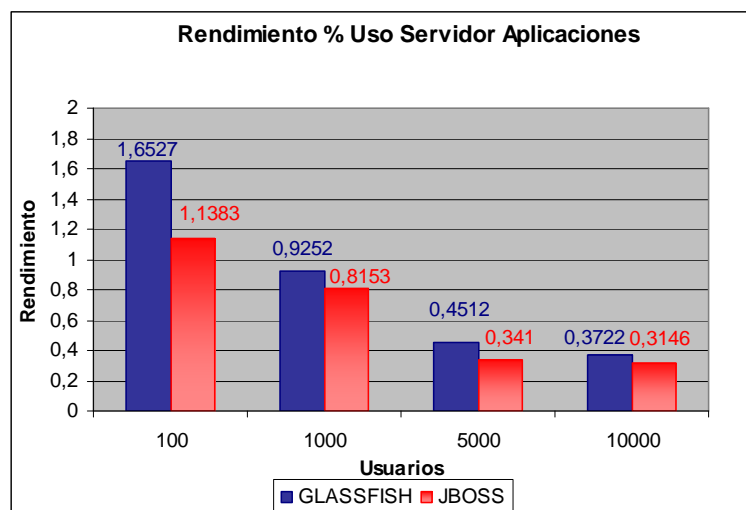


Figura 94. Valores referentes al rendimiento % Uso por parte del Servidor Aplicaciones expuestos en la tabla 16.

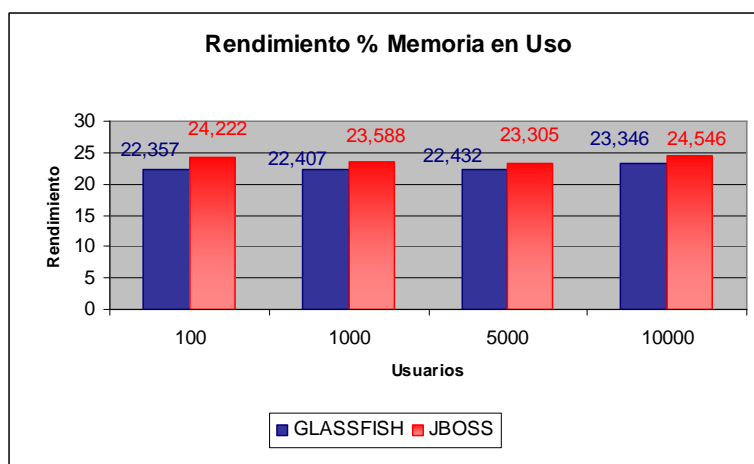


Figura 95. Valores referentes al rendimiento % Memoria en Uso expuestos en la tabla 16.

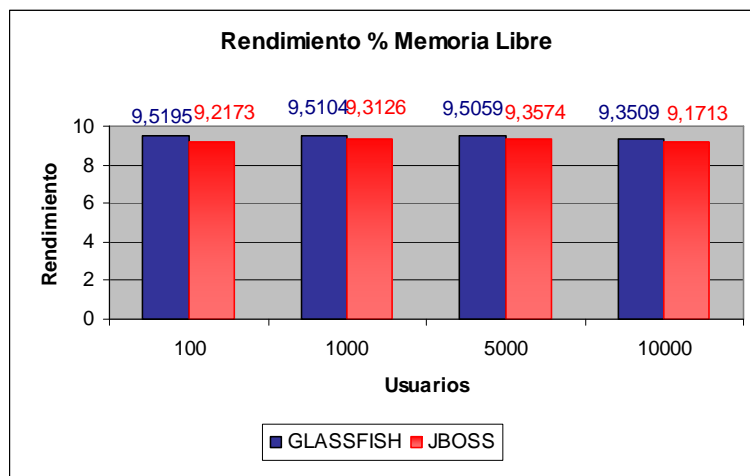


Figura 96. Valores referentes al rendimiento % Memoria Libre expuestos en la tabla 16.

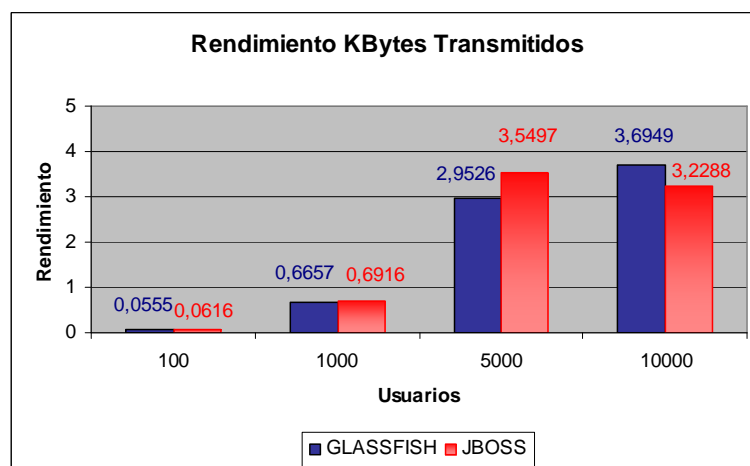


Figura 97. Valores referentes al rendimiento de KBytes Transmitidos expuestos en la tabla 16.

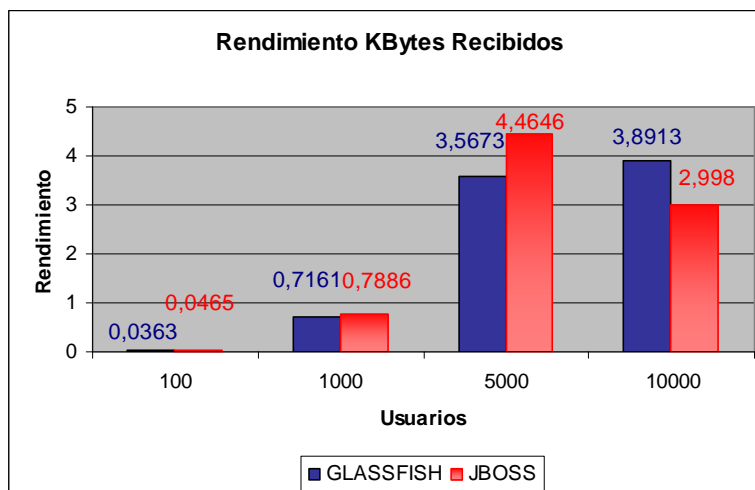


Figura 98. Valores referentes al rendimiento de KBytes Recibidos expuestos en la tabla 16.

Tareas: 1 Registro – 2 Compras

RENDIMIENTO								
	GLASSFISH				JBOSS			
Total Transacciones	300	3000	15000	30000	300	3000	15000	30000
USUARIOS	100	1000	5000	10000	100	1000	5000	10000
% Uso Sistema Operativo	6,3802	0,3853	0,1873	0,1832	8,995	0,4217	0,1993	0,1687
% Uso Servidor Aplicaciones	1,3415	0,6675	0,393	0,3418	1,2971	0,5598	0,3376	0,2945
Memoria % en Uso	21,123	21,486	24,725	21,726	23,167	24,105	23,039	21,62
Memoria % Libre	9,7624	9,6867	9,1465	9,6387	9,3798	9,2343	9,4009	9,6596
Kbytes Transmitidos	0,0654	1,6928	3,7124	3,3257	0,0713	1,7251	3,5998	3,2869
Kbytes Recibidos	0,0598	2,2404	4,7493	3,3128	0,074	2,3634	4,556	3,1003

Tabla 17. Resultados de Rendimientos obtenidos por los servidores de aplicaciones al momento que los usuarios ejecutan las tareas 1 Registro y 2 Compras.

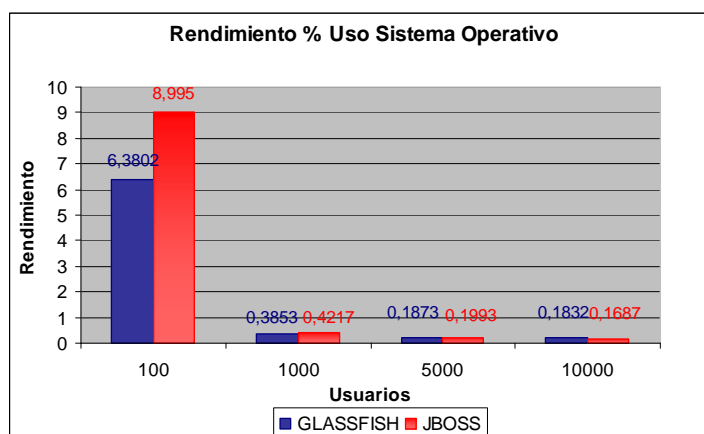


Figura 99. Valores referentes al rendimiento % Uso por parte del Sistema Operativo expuestos en la tabla 17.

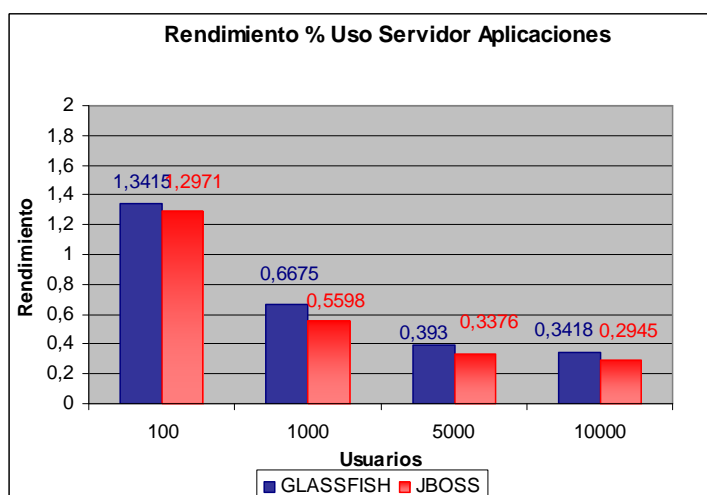


Figura 100. Valores referentes al rendimiento % Uso por parte del Servidor Aplicaciones expuestos en la tabla 17.

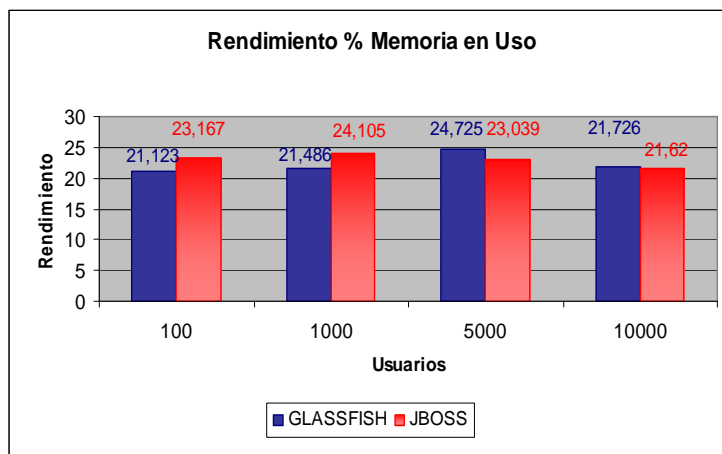


Figura 101. Valores referentes al rendimiento % Memoria en Uso expuestos en la tabla 17.

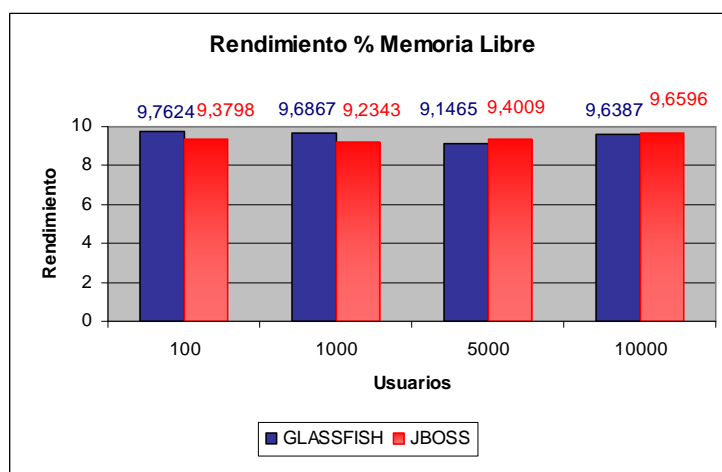


Figura 102. Valores referentes al rendimiento % Memoria Libre expuestos en la tabla 17.

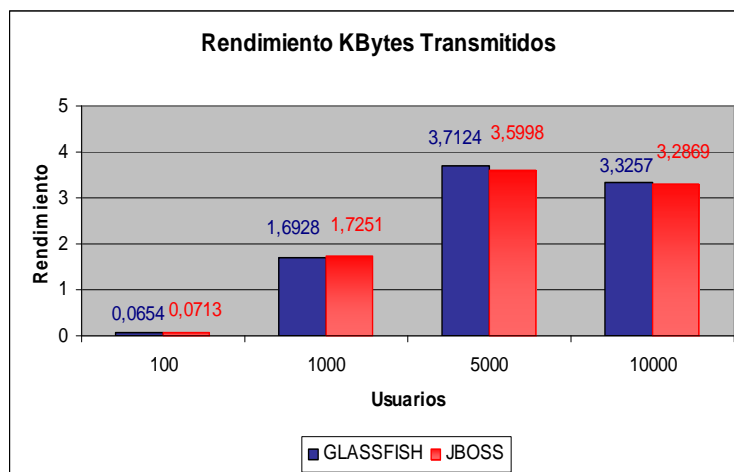


Figura 103. Valores referentes al rendimiento de KBytes Transmitidos expuestos en la tabla 17.

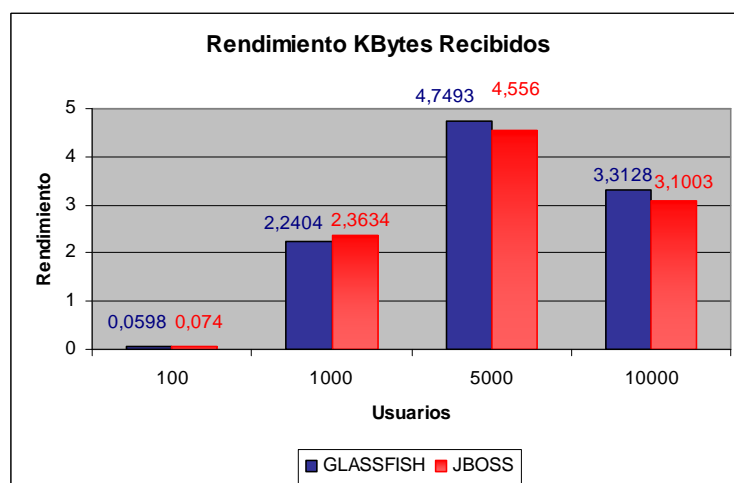


Figura 104. Valores referentes al rendimiento de KBytes Recibidos expuestos en la tabla 17.

Tareas: 1 Registro – 3 Compras

RENDIMIENTO								
	GLASSFISH				JBOSS			
Total Transacciones	400	4000	20000	40000	400	4000	20000	40000
USUARIOS	100	1000	5000	10000	100	1000	5000	10000
% Uso Sistema Operativo	3,5399	0,2739	0,1969	0,2041	3,658	0,3993	0,2393	0,1789
% Uso Servidor Aplicaciones	1,7474	0,4646	0,427	0,3362	1,5858	0,4493	0,3638	0,2954
Memoria % en Uso	21,094	21,581	21,574	23,289	21,709	23,549	24,996	24,21
Memoria % Libre	9,7685	9,6676	9,669	9,36	9,6419	9,3186	9,1099	9,219
KBytes Transmitidos	0,0939	3,2683	3,984	3,8158	0,1006	3,3358	3,6111	3,3947
KBytes Recibidos	0,0933	4,7491	5,1963	4,0759	0,1127	4,9364	4,578	3,269

Tabla 18. Resultados de Rendimientos obtenidos por los servidores de aplicaciones al momento que los usuarios ejecutan las tareas 1 Registro y 3 Compras.

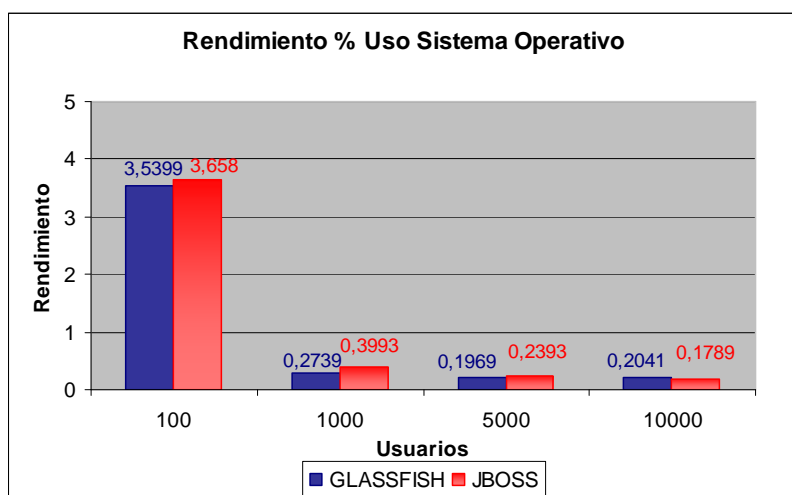


Figura 105. Representación gráfica de los valores referentes al rendimiento % Uso por parte del Sistema Operativo expuestos en la tabla 18.

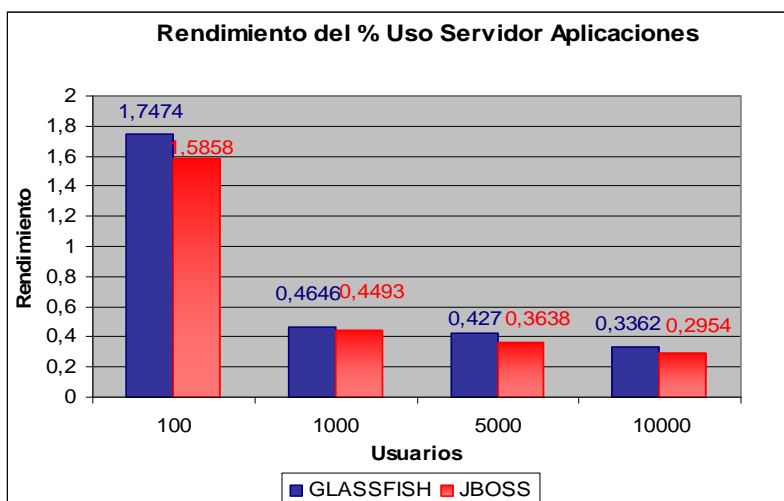


Figura 106. Valores referentes al rendimiento % Uso por parte del Servidor Aplicaciones expuestos en la tabla 18.

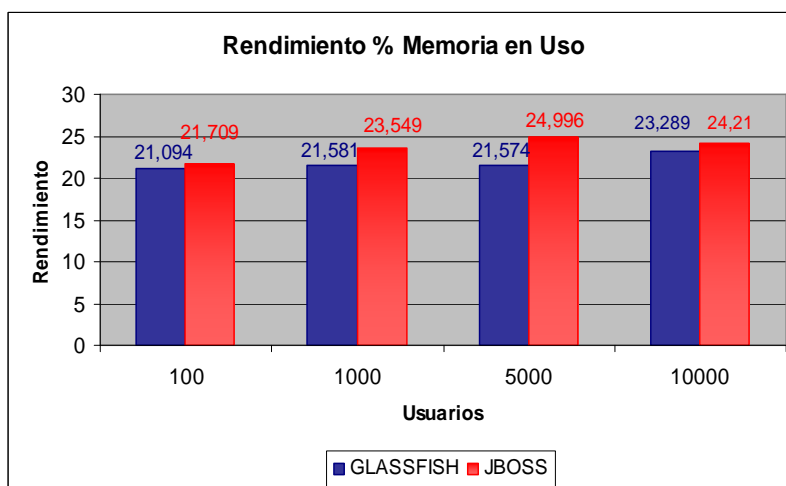


Figura 107. Valores referentes al rendimiento % Memoria en Uso expuestos en la tabla 18.

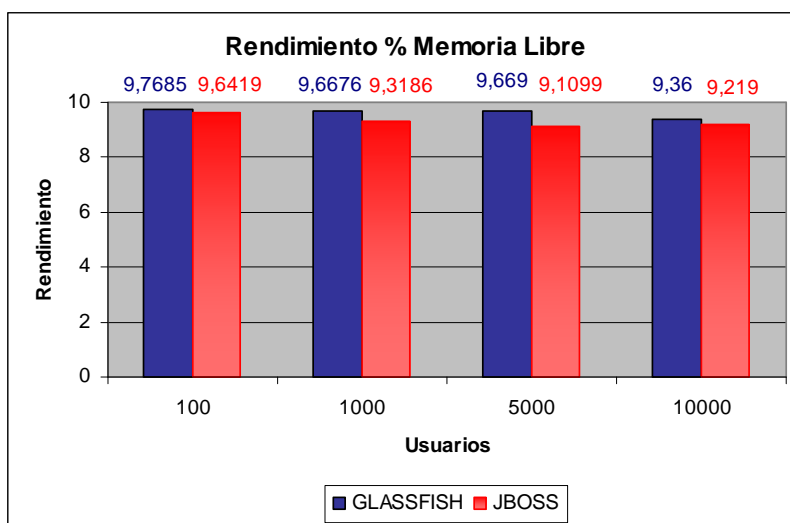


Figura 108. Valores referentes al rendimiento % Memoria Libre expuestos en la tabla 18.

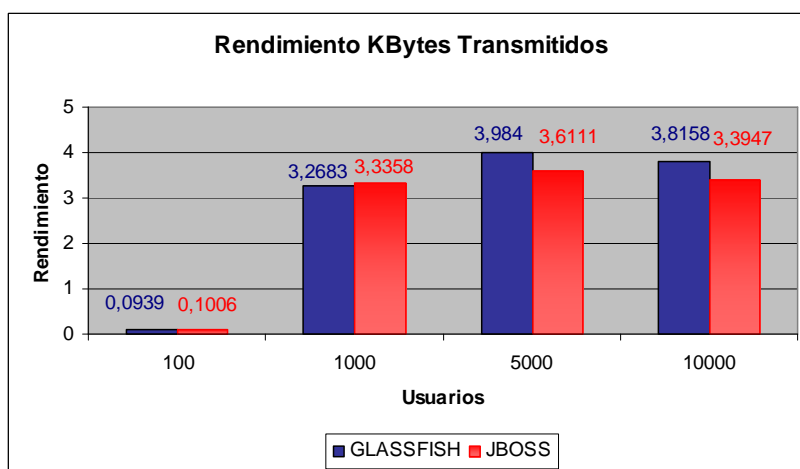


Figura 109. Valores referentes al rendimiento de KBytes Transmitidos expuestos en la tabla 18.

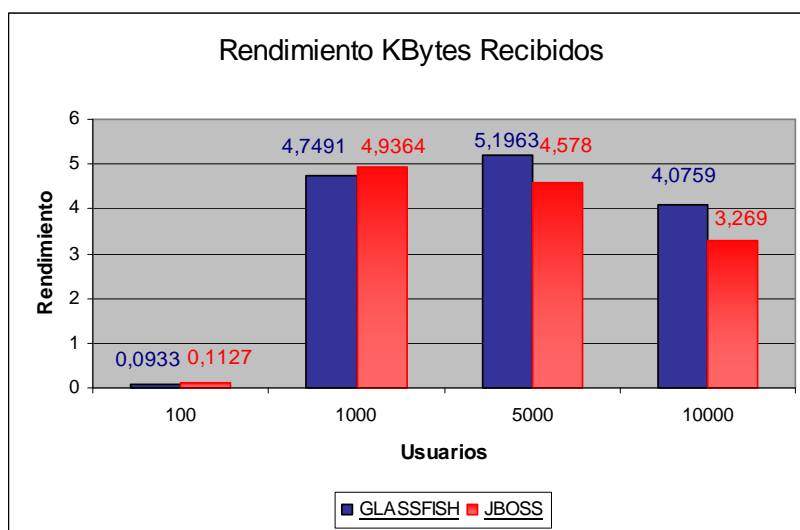


Figura 110. Valores referentes al rendimiento de KBytes Recibidos expuestos en la tabla 18.

INFORMACIÓN REFERENTE A TIEMPOS DE RESPUESTA.

TIEMPOS DE RESPUESTA														
	GLASSFISH							JBOSS						
USUARIOS	1	10	100	500	1000	5000	10000	1	10	100	500	1000	5000	10000
Promedio	0,078	1,9525	3,0625	17,999	43,625	619,11	2089,7	0,078	1,9065	3,461	23,726	52,438	625,93	2157,2
Rendimiento	12,820	25,608	32,653	27,778	22,922	8,076	4,785	12,82	26,226	28,893	21,073	19,07	7,9881	4,6354

Tabla 19. Resultados de Tiempos de Respuesta obtenidos por los Servidores de Aplicaciones de acuerdo al número de usuarios que interactúan y que ejecutan las tareas 1 Registro y 1 Compra.

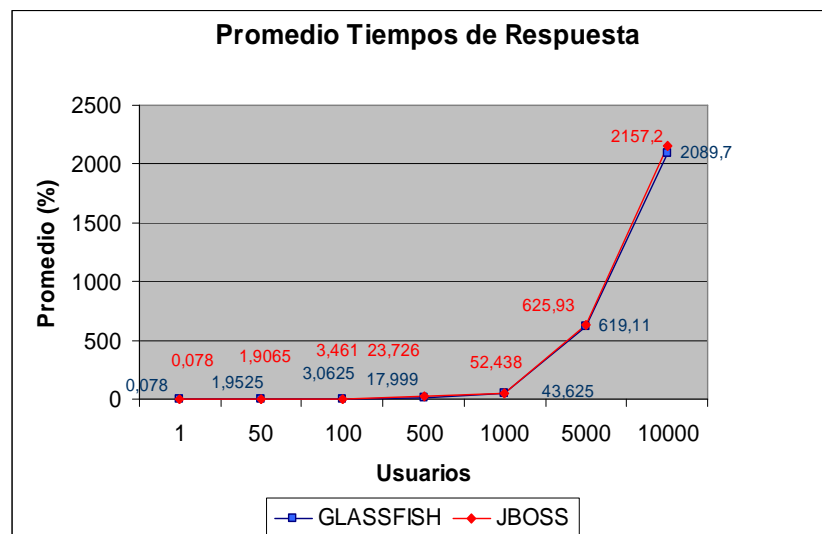


Figura 111. Valores referentes a Promedios obtenidos por los Tiempos de Respuesta expuestos en la tabla 19.

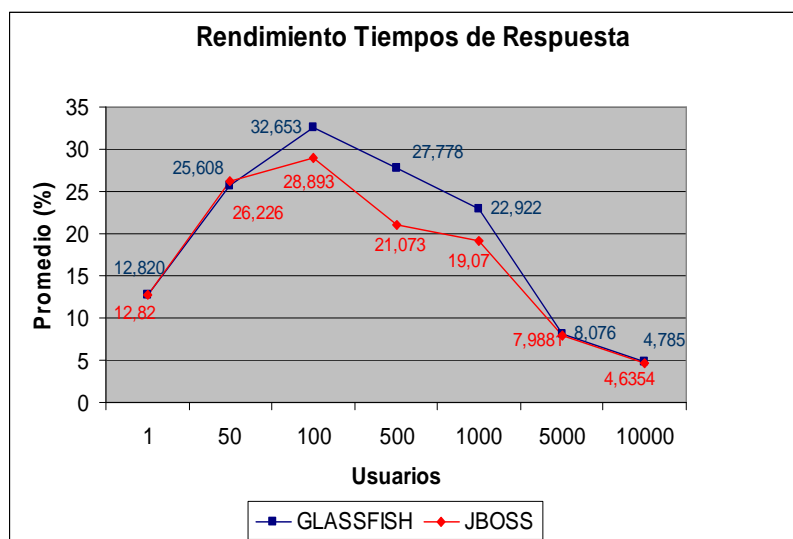


Figura 112. Valores referentes a Rendimientos obtenidos por los Tiempos de Respuesta expuestos en la tabla 19.

TIEMPOS DE RESPUESTA														
	GLASSFISH							JBOSS						
USUARIOS	1	10	100	500	1000	5000	10000	1	10	100	500	1000	5000	10000
Promedio	0,0705	2,5549	5,0545	37,304	113,59	2121,7	7622	0,094	3,0934	5,586	45,742	118,35	2240,3	8145,3
Rendimiento	14,184	19,569	19,784	13,403	8,8033	2,3565	1,3119	10,638	16,162	17,901	10,93	8,4488	2,2318	1,2276

Tabla 20. Resultados de Tiempos de Respuesta obtenidos por los Servidores de Aplicaciones de acuerdo al número de usuarios que interactúan y que ejecutan las tareas 1 Registro y 2 Compras.

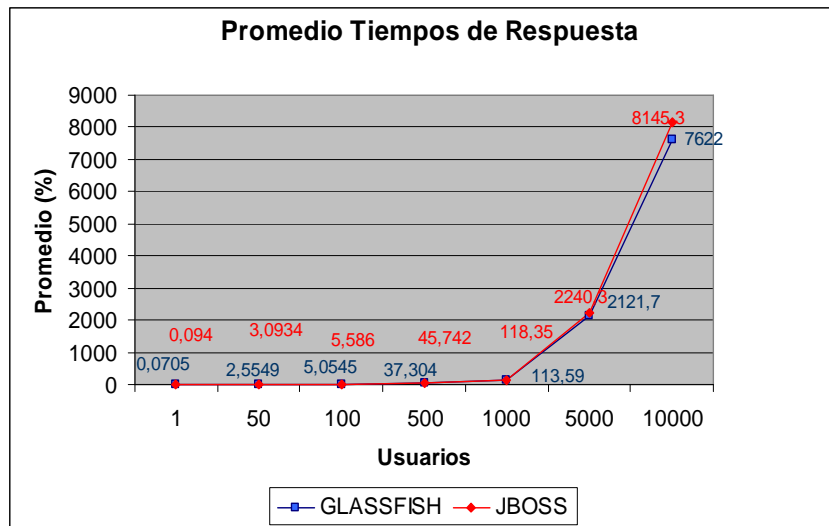


Figura 113. Valores referentes a Promedios obtenidos por los Tiempos de Respuesta expuestos en la tabla 20.

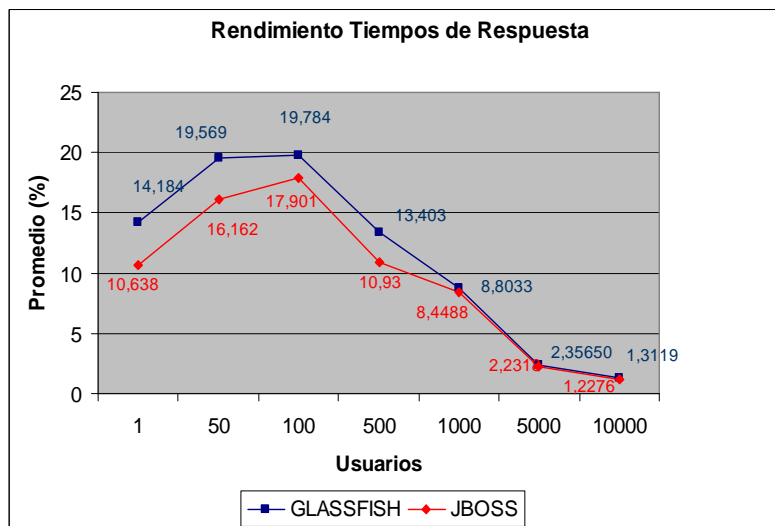


Figura 114. Valores referentes a Rendimientos obtenidos por los Tiempos de Respuesta expuestos en la tabla 20.

TIEMPOS DE RESPUESTA														
	GLASSFISH							JBOSS						
USUARIOS	1	10	100	500	1000	5000	10000	1	10	100	500	1000	5000	10000
Promedio	0,078	3,4139	7,008	66,805	222,66	4392,4	-----	0,1169	4,1175	8,32	77,625	242,08	4563,7	-----
Rendimiento	12,820	14,645	14,269	7,484	4,491	1,1383	-----	8,547	12,143	12,019	6,4412	4,1307	1,0955	-----

Tabla 21. Resultados de Tiempos de Respuesta obtenidos por los Servidores de Aplicaciones de acuerdo al número de usuarios que interactúan y que ejecutan las tareas 1 Registro y 3 Compras.

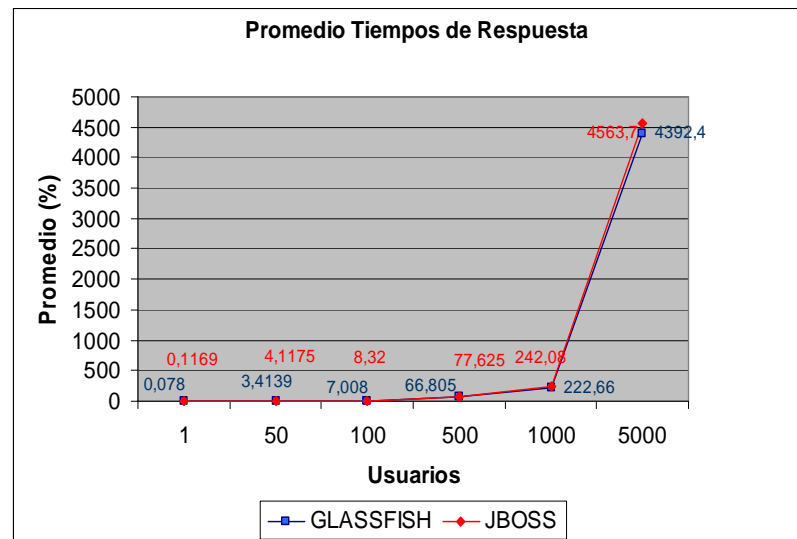


Figura 115. Valores referentes a Promedios obtenidos por los Tiempos de Respuesta expuestos en la tabla 21.

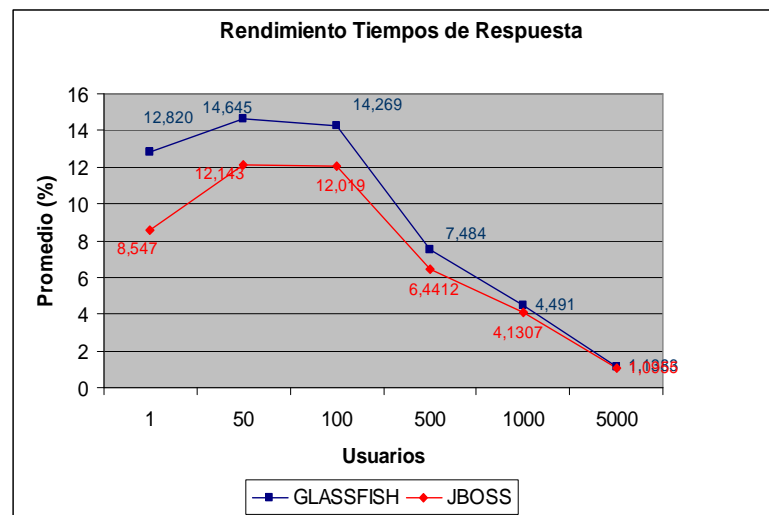


Figura 116. Valores referentes a Promedios obtenidos por los Tiempos de Respuesta expuestos en la tabla 21.

Intervalo de Medición (Segundos)	Servidor de Aplicaciones GLASSFISH						Servidor de Aplicaciones JBOSS					
	Recurso CPU		Recurso MEMORIA		Recurso RED		Recurso CPU		Recurso MEMORIA		Recurso RED	
	% Uso Sistema Operativo	% Uso Servidor Aplicaciones	% Memoria Libre	% Memoria en Uso	KBytes TX	KBytes RX	% Uso Sistema Operativo	% Uso Servidor Aplicaciones	% Memoria Libre	% Memoria en Uso	KBytes TX	KBytes RX
0-20	0.0	4.2	69.9	30.0	10897.0	18063.0	1.5	2.6	73.0	26.9	15314.0	103034.0
21-40	0.5	1.0	70.1	29.8	10899.0	18068.0	0.0	1.0	73.0	26.9	15316.0	103308.0
41-60	0.0	1.0	70.0	29.9	10901.0	18073.0	0.0	1.0	73.0	26.9	15318.0	103331.0
61-80	0.0	1.0	69.9	30.0	11209.0	18266.0	0.0	1.0	72.6	27.3	15407.0	103734.0
81-100	0.0	1.0	70.2	29.7	11470.0	18289.0	0.0	1.0	72.6	27.3	15728.0	103749.0
101-120	0.0	1.0	70.2	29.7	11836.0	18608.0	0.0	8.5	72.4	27.5	15998.0	103754.0
121-140	0.0	1.0	70.2	29.7	11865.0	18617.0	0.0	1.0	72.4	27.5	16375.0	103759.0
141-160	1.5	1.0	70.1	29.8	11866.0	18622.0	0.0	1.0	72.3	27.6	16377.0	103763.0
161-180	0.0	1.0	70.1	29.8	11868.0	18626.0	0.0	4.1	72.3	27.6	16378.0	103767.0
181-200	0.0	1.0	70.1	29.8	11869.0	18630.0	0.0	1.0	72.2	27.7	16380.0	103772.0
201-220	0.0	1.0	70.1	29.8	11870.0	18634.0	0.0	1.0	72.3	27.6	16381.0	103776.0
221-240	0.0	1.0	70.0	29.9	11872.0	18639.0	0.0	2.5	72.2	27.7	16382.0	103780.0
241-260	0.0	1.0	70.1	29.8	11873.0	18643.0	0.0	1.0	72.2	27.7	16385.0	103785.0
261-280	0.0	1.0	70.1	29.8	11875.0	18648.0	0.0	1.0	72.2	27.7	16386.0	103789.0
281-300	3.1	1.0	70.2	29.7	11876.0	18652.0	0.0	2.5	72.1	27.8	16387.0	103793.0
301-320	0.0	1.0	70.1	29.8	11877.0	18657.0	0.0	1.0	72.2	27.7	16389.0	103798.0
321-340	0.0	1.0	70.1	29.8	11879.0	18661.0	0.0	1.0	72.1	27.8	16391.0	103801.0
341-360	0.0	1.0	70.1	29.8	11880.0	18665.0	0.0	1.0	72.3	27.6	16392.0	103806.0
361-380	0.0	1.0	70.1	29.8	11881.0	18669.0	0.0	1.0	72.3	27.6	16393.0	103810.0
381-400	0.0	1.0	70.1	29.8	11883.0	18674.0	3.2	1.0	72.2	27.7	16395.0	103816.0
401-420	0.0	1.0	70.1	29.8	11884.0	18678.0	0.0	1.0	72.3	27.6	16397.0	103821.0
421-440	0.0	1.0	70.1	29.8	11885.0	18682.0	0.0	4.0	72.3	27.6	16399.0	103826.0
441-460	0.0	1.0	70.1	29.8	11887.0	18687.0	0.0	1.0	72.3	27.6	16400.0	103830.0
461-480	0.0	1.0	70.0	29.9	11888.0	18692.0	0.0	1.0	72.3	27.6	16402.0	103835.0
481-500	0.0	1.0	70.1	29.8	11889.0	18695.0	0.0	1.0	72.3	27.6	16403.0	103839.0
501-520	0.0	1.0	70.0	29.9	11891.0	18700.0	0.0	1.0	72.3	27.6	16418.0	103843.0
521-540	0.0	1.0	70.0	29.9	11893.0	18705.0	0.0	1.0	72.3	27.6	16420.0	103848.0
541-560	0.0	1.0	70.0	29.9	11894.0	18709.0	0.0	1.0	72.3	27.6	16421.0	103862.0
561-580	0.0	1.0	69.9	30.0	11895.0	18713.0	0.0	1.0	72.3	27.6	16423.0	103866.0
581-600	0.0	1.0	69.9	30.0	11897.0	18718.0	0.0	1.0	72.3	27.6	16424.0	103871.0

Tabla 22. RESULTADOS DE MEDICIÓN DE RECURSOS CON CARGA DE 100 USUARIOS Y TAREAS 1 REGISTRO Y 1 COMPRA DURANTE 10 MINUTOS (600 segundos).

Intervalo de Medición (Segundos)	Servidor de Aplicaciones GLASSFISH						Servidor de Aplicaciones JBOSS					
	Recurso CPU		Recurso MEMORIA		Recurso RED		Recurso CPU		Recurso MEMORIA		Recurso RED	
	% Uso Sistema Operativo	% Uso Servidor Aplicaciones	% Memoria Libre	% Memoria en Uso	KBytes TX	KBytes RX	% Uso Sistema Operativo	% Uso Servidor Aplicaciones	% Memoria Libre	% Memoria en Uso	KBytes TX	KBytes RX
0-20	3.0	4.2	69.9	30.0	10897.0	18063.0	3.0	1.0	71.5	28.4	11294.0	103930.0
21-40	0.0	1.0	70.1	29.8	10899.0	18068.0	0.0	1.0	71.4	28.5	11684.0	104203.0
41-60	0.0	1.0	70.0	29.9	10901.0	18073.0	0.0	13.3	71.4	28.5	11754.0	104237.0
61-80	0.0	1.0	69.9	30.0	11209.0	18266.0	0.0	1.0	71.3	28.6	12535.0	105140.0
81-100	0.0	1.0	70.2	29.7	11470.0	18289.0	0.0	1.0	71.3	28.6	12536.0	105143.0
101-120	0.0	1.0	70.2	29.7	11836.0	18608.0	0.0	1.0	71.3	28.6	12537.0	105148.0
121-140	0.0	1.0	70.2	29.7	11865.0	18617.0	0.0	1.0	71.2	28.7	12539.0	105153.0
141-160	0.0	1.0	70.1	29.8	11866.0	18622.0	0.0	1.0	71.2	28.7	12540.0	105157.0
161-180	0.0	1.0	70.1	29.8	11868.0	18626.0	0.0	1.0	71.2	28.7	12541.0	105161.0
181-200	0.0	1.0	70.1	29.8	11869.0	18630.0	0.0	1.0	71.2	28.7	12542.0	105166.0
201-220	0.0	1.0	70.1	29.8	11870.0	18634.0	0.0	1.0	71.2	28.7	12544.0	105169.0
221-240	0.0	1.0	70.0	29.9	11872.0	18639.0	0.0	1.0	71.1	28.8	12545.0	105174.0
241-260	3.1	1.0	70.1	29.8	11873.0	18643.0	3.1	1.0	71.0	28.9	12546.0	105178.0
261-280	0.0	1.0	70.1	29.8	11875.0	18648.0	0.0	1.0	71.0	28.9	12547.0	105183.0
281-300	0.0	1.0	70.2	29.7	11876.0	18652.0	0.0	1.0	71.0	28.9	12548.0	105187.0
301-320	0.0	1.0	70.1	29.8	11877.0	18657.0	0.0	1.0	71.0	28.9	12549.0	105191.0
321-340	0.0	1.0	70.1	29.8	11879.0	18661.0	0.0	1.0	71.0	28.9	12551.0	105195.0
341-360	0.0	1.0	70.1	29.8	11880.0	18665.0	0.0	1.0	71.0	28.9	12552.0	105199.0
361-380	0.0	1.0	70.1	29.8	11881.0	18669.0	0.0	1.0	71.0	28.9	12553.0	105204.0
381-400	0.0	1.0	70.1	29.8	11883.0	18674.0	0.0	1.0	71.0	28.9	12554.0	105207.0
401-420	0.0	1.0	70.1	29.8	11884.0	18678.0	0.0	1.0	71.0	28.9	12555.0	105212.0
421-440	0.0	1.0	70.1	29.8	11885.0	18682.0	0.0	1.0	71.0	28.9	12556.0	105217.0
441-460	0.0	1.0	70.1	29.8	11887.0	18687.0	0.0	1.0	71.0	28.9	12571.0	105233.0
461-480	0.0	1.0	70.0	29.9	11888.0	18692.0	0.0	1.0	71.0	28.9	12572.0	105238.0
481-500	0.0	1.0	70.1	29.8	11889.0	18695.0	0.0	1.0	71.0	28.9	12573.0	105242.0
501-520	0.0	1.0	70.0	29.9	11891.0	18700.0	0.0	1.0	71.0	28.9	12574.0	105246.0
521-540	0.0	1.0	70.0	29.9	11893.0	18705.0	0.0	1.0	71.0	28.9	12575.0	105250.0
541-560	0.0	1.0	70.0	29.9	11894.0	18709.0	0.0	1.0	71.0	28.9	12577.0	105255.0
561-580	0.0	1.0	69.9	30.0	11895.0	18713.0	0.0	1.0	71.0	28.9	12578.0	105259.0
581-600	0.0	1.0	69.9	30.0	11897.0	18718.0	0.0	1.0	71.0	28.9	12579.0	105263.0

Tabla 23. RESULTADOS DE MEDICIÓN DE RECURSOS CON CARGA DE 100 USUARIOS Y TAREAS1 REGISTRO Y 2 COMPRAS DURANTE 10 MINUTOS (600 segundos).

Intervalo de Medición (Segundos)	Servidor de Aplicaciones GLASSFISH						Servidor de Aplicaciones JBOSS					
	Recurso CPU		Recurso MEMORIA		Recurso RED		Recurso CPU		Recurso MEMORIA		Recurso RED	
	% Uso Sistema Operativo	% Uso Servidor Aplicaciones	% Memoria Libre	% Memoria en Uso	KBytes TX	KBytes RX	% Uso Sistema Operativo	% Uso Servidor Aplicaciones	% Memoria Libre	% Memoria en Uso	KBytes TX	KBytes RX
0-20	1.5	1.0	68.8	31.1	13214.0	19971.0	3.0	1.0	69.5	30.4	12682.0	15963.0
21-40	0.0	1.0	68.8	31.1	13245.0	19979.0	0.0	1.0	69.5	30.4	13074.0	15968.0
41-60	0.0	1.0	68.7	31.2	13246.0	19983.0	0.0	1.0	69.2	30.7	13127.0	15974.0
61-80	0.0	1.0	68.5	31.4	13247.0	19987.0	0.0	1.0	69.3	30.6	14340.0	15978.0
81-100	12.5	1.0	68.4	31.5	13557.0	20181.0	0.0	1.0	69.3	30.6	14451.0	16250.0
101-120	0.0	1.0	68.3	31.6	13590.0	20193.0	0.0	5.6	69.2	30.7	14452.0	16263.0
121-140	1.5	2.4	68.2	31.7	14703.0	21375.0	0.0	1.0	69.2	30.7	14453.0	16434.0
141-160	0.0	1.0	68.2	31.7	14875.0	21553.0	0.0	1.0	69.2	30.7	14454.0	17883.0
161-180	0.0	1.0	68.2	31.7	14876.0	21557.0	0.0	1.0	69.2	30.7	14456.0	17889.0
181-200	0.0	1.0	68.2	31.7	14878.0	21562.0	0.0	1.0	69.2	30.7	14457.0	17894.0
201-220	0.0	1.0	68.2	31.7	14879.0	21566.0	7.5	1.0	69.2	30.7	14458.0	17898.0
221-240	0.0	1.0	68.2	31.7	14881.0	21571.0	3.1	1.0	69.2	30.7	14459.0	17902.0
241-260	0.0	1.0	68.2	31.7	14882.0	21575.0	0.0	1.0	69.2	30.7	14460.0	17907.0
261-280	0.0	1.0	68.2	31.7	14883.0	21579.0	0.0	1.0	69.2	30.7	14461.0	17911.0
281-300	0.0	1.0	68.2	31.7	14885.0	21584.0	0.0	1.0	69.0	30.9	14462.0	17915.0
301-320	0.0	1.0	68.2	31.7	14886.0	21588.0	0.0	1.0	69.0	30.9	14464.0	17919.0
321-340	0.0	1.0	68.2	31.7	14887.0	21593.0	0.0	1.0	69.0	30.9	14465.0	17924.0
341-360	0.0	1.0	68.2	31.7	14888.0	21597.0	0.0	1.0	69.0	30.9	14466.0	17928.0
361-380	0.0	1.0	68.2	31.7	14890.0	21602.0	0.0	1.0	69.0	30.9	14467.0	17942.0
381-400	0.0	1.0	68.2	31.7	14892.0	21607.0	0.0	1.0	69.1	30.8	14468.0	17947.0
401-420	0.0	1.0	68.2	31.7	14893.0	21611.0	0.0	1.0	69.1	30.8	14470.0	17951.0
421-440	0.0	1.0	68.2	31.7	14894.0	21616.0	0.0	1.0	69.1	30.8	14470.0	17956.0
441-460	0.0	1.0	68.2	31.7	14896.0	21620.0	0.0	1.0	69.1	30.8	14486.0	17959.0
461-480	0.0	1.0	68.2	31.7	14897.0	21624.0	0.0	1.0	69.2	30.7	14487.0	17964.0
481-500	0.0	1.0	68.2	31.7	14898.0	21629.0	0.0	1.0	69.2	30.7	14488.0	17969.0
501-520	0.0	1.0	68.2	31.7	14900.0	21634.0	0.0	1.0	69.2	30.7	14489.0	17973.0
521-540	0.0	1.0	68.2	31.7	14901.0	21638.0	1.4	1.0	69.2	30.7	14490.0	17980.0
541-560	0.0	1.0	68.2	31.7	14902.0	21642.0	0.0	1.0	69.2	30.7	14491.0	17985.0
561-580	0.0	1.0	68.2	31.7	14904.0	21647.0	0.0	1.0	69.2	30.7	14492.0	17989.0
581-600	0.0	1.0	68.3	31.6	14905.0	21652.0	0.0	1.0	69.1	30.8	14494.0	17993.0

Tabla 24. RESULTADOS DE MEDICIÓN DE RECURSOS CON CARGA DE 100 USUARIOS Y TAREAS 1 REGISTRO Y 3 COMPRAS DURANTE 10 MINUTOS (600 segundos).

Intervalo de Medición (Segundos)	Servidor de Aplicaciones GLASSFISH						Servidor de Aplicaciones JBOSS					
	Recurso CPU		Recurso MEMORIA		Recurso RED		Recurso CPU		Recurso MEMORIA		Recurso RED	
	% Uso Sistema Operativo	% Uso Servidor Aplicaciones	% Memoria Libre	% Memoria en Uso	KBytes TX	KBytes RX	% Uso Sistema Operativo	% Uso Servidor Aplicaciones	% Memoria Libre	% Memoria en Uso	KBytes TX	KBytes RX
0-20	4.6	1.0	70.9	29.0	15064.0	60399.0	0.0	2.6	71.9	28.0	1270569.0	2061533.0
21-40	3.1	1.0	70.8	29.1	15065.0	60403.0	0.0	1.0	71.9	28.0	1270593.0	2061539.0
41-60	1.5	1.0	70.8	29.1	15067.0	60407.0	0.0	1.0	71.9	28.0	1270594.0	2061543.0
61-80	1.6	1.0	70.8	29.1	15068.0	60413.0	0.0	1.0	71.9	28.0	1270596.0	2061549.0
81-100	1.6	1.0	70.8	29.1	15069.0	60417.0	0.0	1.0	71.9	28.0	1270598.0	2061553.0
101-120	9.3	18.1	70.4	29.5	16413.0	60945.0	9.3	21.3	71.7	28.2	1272709.0	2062867.0
121-140	1.6	1.0	70.3	29.6	17912.0	62131.0	0.0	1.0	71.6	28.3	1274316.0	2064224.0
141-160	1.5	1.0	70.3	29.6	17939.0	62154.0	0.0	1.0	71.7	28.2	1274336.0	2064233.0
161-180	23.1	1.0	70.1	29.8	17969.0	62172.0	0.0	1.0	71.7	28.2	1274379.0	2064250.0
181-200	12.3	4.0	70.1	29.8	18847.0	62944.0	12.5	4.0	71.5	28.4	1274653.0	2064458.0
201-220	15.1	2.6	70.1	29.8	21411.0	66204.0	9.4	7.1	71.6	28.3	1276696.0	2067024.0
221-240	13.7	7.1	70.2	29.7	24055.0	69850.0	15.3	1.0	71.6	28.3	1279104.0	2070415.0
241-260	1.6	1.0	70.0	29.9	26150.0	72805.0	13.8	5.7	71.6	28.3	1281597.0	2074023.0
261-280	1.5	1.0	70.0	29.9	26168.0	72811.0	0.0	1.0	71.6	28.3	1282665.0	2075563.0
281-300	1.6	1.0	69.9	30.0	26209.0	72837.0	0.0	1.0	71.6	28.3	1282680.0	2075570.0
301-320	18.7	1.0	69.9	30.0	26258.0	72866.0	0.0	1.0	71.6	28.3	1282686.0	2075576.0
321-340	1.5	2.5	69.9	30.0	26312.0	72894.0	0.0	1.0	71.6	28.3	1282758.0	2075609.0
341-360	0.0	1.0	69.9	30.0	26362.0	72916.0	0.0	1.0	71.5	28.4	1282798.0	2075644.0
361-380	1.5	1.0	69.9	30.0	26389.0	72929.0	0.0	1.0	71.6	28.3	1282822.0	2075651.0
381-400	1.5	1.0	69.9	30.0	26424.0	72953.0	0.0	1.0	71.6	28.3	1282835.0	2075665.0
401-420	1.5	1.0	69.9	30.0	26468.0	72984.0	0.0	1.0	71.5	28.4	1282837.0	2075670.0
421-440	0.0	1.0	69.9	30.0	26548.0	73039.0	0.0	1.0	71.5	28.4	1282838.0	2075676.0
441-460	1.5	1.0	69.9	30.0	26623.0	73090.0	0.0	1.0	71.5	28.4	1282863.0	2075682.0
461-480	1.6	1.0	69.9	30.0	26693.0	73137.0	0.0	1.0	71.5	28.4	1282878.0	2075689.0
481-500	1.6	1.0	69.9	30.0	26694.0	73142.0	0.0	1.0	71.5	28.4	1282894.0	2075696.0
501-520	1.5	1.0	69.9	30.0	26775.0	73166.0	0.0	1.0	71.5	28.4	1282919.0	2075702.0
521-540	1.6	1.0	69.9	30.0	26850.0	73217.0	0.0	1.0	71.5	28.4	1282920.0	2075707.0
541-560	1.5	1.0	69.9	30.0	26914.0	73259.0	0.0	1.0	71.5	28.4	1282945.0	2075714.0
561-580	1.5	1.0	69.9	30.0	26998.0	73282.0	0.0	2.6	71.5	28.4	1282995.0	2075723.0
581-600	0.0	1.0	69.9	30.0	27048.0	73289.0	0.0	1.0	71.6	28.3	1283019.0	2075729.0

Tabla 25. RESULTADOS DE MEDICIÓN DE RECURSOS CON CARGA DE 1000 USUARIOS Y TAREAS 1 REGISTRO Y 1 COMPRA DURANTE 10 MINUTOS (600 segundos).

Intervalo de Medición (Segundos)	Servidor de Aplicaciones GLASSFISH						Servidor de Aplicaciones JBOSS					
	Recurso CPU		Recurso MEMORIA		Recurso RED		Recurso CPU		Recurso MEMORIA		Recurso RED	
	% Uso Sistema Operativo	% Uso Servidor Aplicaciones	% Memoria Libre	% Memoria en Uso	KBytes TX	KBytes RX	% Uso Sistema Operativo	% Uso Servidor Aplicaciones	% Memoria Libre	% Memoria en Uso	KBytes TX	KBytes RX
0-20	1.5	4.1	69.5	30.4	2388367.0	3858041.0	1.5	1.0	72.6	27.3	1283559.0	2076079.0
21-40	3.1	1.0	69.3	30.6	2388428.0	3858052.0	0.0	1.0	72.6	27.3	1283583.0	2076085.0
41-60	1.6	1.0	69.3	30.6	2388430.0	3858057.0	0.0	1.0	72.6	27.3	1283585.0	2076089.0
61-80	1.6	1.0	69.3	30.6	2388431.0	3858061.0	7.7	7.1	72.4	27.5	1284821.0	2076862.0
81-100	1.6	1.0	69.4	30.5	2388432.0	3858065.0	13.6	5.6	72.3	27.6	1287262.0	2078721.0
101-120	12.3	11.7	68.9	31.0	2390795.0	3859313.0	0.0	1.0	72.3	27.6	1287320.0	2078767.0
121-140	1.5	1.0	68.9	31.0	2391294.0	3859780.0	6.2	1.0	72.2	27.7	1287371.0	2078790.0
141-160	1.6	2.6	68.9	31.0	2391547.0	3859804.0	4.6	5.7	72.2	27.7	1287850.0	2079179.0
161-180	4.7	4.1	68.8	31.1	2392868.0	3861079.0	6.2	10.2	72.2	27.7	1290223.0	2082309.0
181-200	6.3	2.4	68.8	31.1	2395525.0	3864551.0	12.5	10.4	72.2	27.7	1292809.0	2085993.0
201-220	3.1	11.7	68.8	31.1	2398242.0	3868328.0	7.8	4.1	72.2	27.7	1295442.0	2089831.0
221-240	17.1	1.0	68.8	31.1	2400966.0	3872209.0	11.0	5.7	72.2	27.7	1298062.0	2093703.0
241-260	20.0	7.3	68.7	31.2	2403714.0	3876209.0	6.2	7.2	72.2	27.7	1300698.0	2097661.0
261-280	7.8	1.0	68.7	31.2	2406432.0	3880228.0	14.0	1.0	72.2	27.7	1303314.0	2101627.0
281-300	7.6	4.0	68.7	31.2	2409215.0	3884392.0	10.9	7.2	72.2	27.7	1305951.0	2105659.0
301-320	12.3	1.0	68.7	31.2	2411961.0	3888538.0	6.2	5.6	72.2	27.7	1308591.0	2109717.0
321-340	4.6	4.1	68.7	31.2	2414708.0	3892715.0	9.2	7.2	72.2	27.7	1311243.0	2113813.0
341-360	7.6	8.6	68.8	31.1	2417432.0	3896881.0	12.5	4.0	72.2	27.7	1313901.0	2117936.0
361-380	1.5	1.0	68.8	31.1	2418279.0	3898172.0	0.0	1.0	72.2	27.7	1314284.0	2118506.0
381-400	1.6	1.0	68.8	31.1	2418281.0	3898176.0	0.0	1.0	72.2	27.7	1314321.0	2118514.0
401-420	9.4	1.0	68.8	31.1	2418294.0	3898184.0	0.0	1.0	72.2	27.7	1314372.0	2118531.0
421-440	1.6	1.0	68.8	31.1	2418443.0	3898196.0	0.0	1.0	72.1	27.8	1314444.0	2118564.0
441-460	1.6	1.0	68.7	31.2	2418548.0	3898224.0	0.0	1.0	72.1	27.8	1314495.0	2118574.0
461-480	1.5	1.0	68.7	31.2	2418593.0	3898238.0	0.0	1.0	72.2	27.7	1314497.0	2118578.0
481-500	1.6	1.0	68.7	31.2	2418632.0	3898260.0	0.0	1.0	72.1	27.8	1314498.0	2118584.0
501-520	1.5	1.0	68.7	31.2	2418671.0	3898283.0	0.0	1.0	72.1	27.8	1314500.0	2118588.0
521-540	1.5	1.0	68.7	31.2	2418698.0	3898299.0	0.0	1.0	72.1	27.8	1314519.0	2118597.0
541-560	1.6	1.0	68.7	31.2	2418733.0	3898320.0	0.0	1.0	72.1	27.8	1314521.0	2118602.0
561-580	1.5	2.6	68.7	31.2	2418777.0	3898344.0	0.0	1.0	72.2	27.7	1314596.0	2118613.0
581-600	1.6	1.0	68.6	31.3	2418839.0	3898369.0	0.0	1.0	72.1	27.8	1314611.0	2118621.0

Tabla 26. RESULTADOS DE MEDICIÓN DE RECURSOS CON CARGA DE 1000 USUARIOS Y TAREAS 1 REGISTRO Y 2 COMPRAS DURANTE 10 MINUTOS (600 segundos).

Intervalo de Medición (Segundos)	Servidor de Aplicaciones GLASSFISH						Servidor de Aplicaciones JBOSS					
	Recurso CPU		Recurso MEMORIA		Recurso RED		Recurso CPU		Recurso MEMORIA		Recurso RED	
	% Uso Sistema Operativo	% Uso Servidor Aplicaciones	% Memoria Libre	% Memoria en Uso	KBytes TX	KBytes RX	% Uso Sistema Operativo	% Uso Servidor Aplicaciones	% Memoria Libre	% Memoria en Uso	KBytes TX	KBytes RX
0-20	3.1	1.0	69.4	30.5	57167.0	113723.0	4.0	2.6	72.0	27.9	81082.0	128197.0
21-40	0.0	1.0	69.3	30.6	57168.0	113728.0	3.1	1.0	71.7	28.2	81083.0	128200.0
41-60	9.2	12.0	69.2	30.7	57169.0	113732.0	0.0	19.4	71.8	28.1	81326.0	128356.0
61-80	0.0	1.0	69.1	30.8	57171.0	113737.0	0.0	5.6	71.8	28.1	84036.0	130107.0
81-100	0.0	1.0	69.0	30.9	57172.0	113742.0	0.0	1.0	71.7	28.2	84829.0	130875.0
101-120	7.7	5.7	69.0	30.9	59728.0	115169.0	0.0	1.0	71.8	28.1	84877.0	130899.0
121-140	6.2	1.0	68.9	31.0	60067.0	115467.0	0.0	7.2	71.8	28.1	86204.0	132348.0
141-160	7.8	5.7	69.0	30.9	60077.0	115476.0	0.0	4.1	71.8	28.1	88426.0	135414.0
161-180	7.7	5.6	68.9	31.0	60209.0	115569.0	0.0	7.2	71.8	28.1	90780.0	138802.0
181-200	15.6	5.6	68.9	31.0	62537.0	118168.0	0.0	7.0	71.8	28.1	93174.0	142309.0
201-220	7.6	2.4	68.9	31.0	65152.0	121690.0	1.6	2.5	71.7	28.2	95644.0	145980.0
221-240	7.8	4.1	68.9	31.0	67795.0	125383.0	9.3	2.5	71.6	28.3	98106.0	149682.0
241-260	7.6	7.2	68.9	31.0	70438.0	129156.0	14.0	5.6	71.5	28.4	100589.0	153456.0
261-280	9.2	5.4	68.8	31.1	73126.0	133068.0	12.5	5.7	71.5	28.4	103104.0	157306.0
281-300	13.5	10.4	68.8	31.1	75821.0	137047.0	0.0	7.2	71.5	28.4	105629.0	161196.0
301-320	9.4	5.7	68.8	31.1	78489.0	141028.0	0.0	4.1	71.5	28.4	108170.0	165129.0
321-340	7.6	2.5	69.0	30.9	81172.0	145065.0	0.0	1.0	71.5	28.4	110728.0	169104.0
341-360	12.5	7.3	69.1	30.8	83856.0	149135.0	0.0	2.6	71.4	28.5	113309.0	173129.0
361-380	6.3	5.5	69.0	30.9	86534.0	153235.0	0.0	5.7	71.5	28.4	115898.0	177178.0
381-400	4.6	2.6	69.0	30.9	89225.0	157373.0	9.3	2.5	71.5	28.4	118463.0	181199.0
401-420	12.4	5.7	69.0	30.9	91897.0	161499.0	4.7	4.0	71.5	28.4	121048.0	185264.0
421-440	6.1	4.0	69.0	30.9	94589.0	165671.0	10.9	5.6	71.4	28.5	123651.0	189365.0
441-460	12.1	4.1	69.0	30.9	97265.0	169830.0	9.2	2.4	71.4	28.5	126248.0	193462.0
461-480	14.0	2.6	69.0	30.9	99935.0	173991.0	10.8	2.5	71.4	28.5	128832.0	197547.0
481-500	6.1	2.5	69.0	30.9	102598.0	178152.0	7.8	4.1	71.4	28.5	131447.0	201687.0
501-520	3.1	2.5	69.0	30.9	105280.0	182351.0	9.3	1.0	71.4	28.5	134023.0	205771.0
521-540	0.0	1.0	69.0	30.9	107968.0	186567.0	7.7	2.5	71.4	28.5	136634.0	209916.0
541-560	0.0	1.0	69.0	30.9	110650.0	190783.0	4.6	2.5	71.4	28.5	139225.0	214034.0
561-580	3.1	1.0	69.0	30.9	113332.0	195005.0	9.4	1.0	71.4	28.5	141127.0	217048.0
581-600	0.0	1.0	68.9	31.0	115997.0	199207.0	9.2	1.0	71.4	28.5	141128.0	217053.0

Tabla 27. RESULTADOS DE MEDICIÓN DE RECURSOS CON CARGA DE 1000 USUARIOS Y TAREAS1 REGISTRO Y 3 COMPRAS DURANTE 10 MINUTOS (600 segundos).

Intervalo de Medición (Segundos)	Servidor de Aplicaciones GLASSFISH						Servidor de Aplicaciones JBOSS					
	Recurso CPU		Recurso MEMORIA		Recurso RED		Recurso CPU		Recurso MEMORIA		Recurso RED	
	% Uso Sistema Operativo	% Uso Servidor Aplicaciones	% Memoria Libre	% Memoria en Uso	KBytes TX	KBytes RX	% Uso Sistema Operativo	% Uso Servidor Aplicaciones	% Memoria Libre	% Memoria en Uso	KBytes TX	KBytes RX
0-20	0.0	1.0	70.7	29.2	116284.0	199552.0	1.5	1.0	71.8	28.1	1375231.0	2207713.0
21-40	1.5	1.0	70.5	29.4	116309.0	199559.0	0.0	1.0	71.8	28.1	1375233.0	2207717.0
41-60	1.6	1.0	70.5	29.4	116310.0	199563.0	15.6	8.7	71.4	28.5	1375668.0	2207992.0
61-80	1.6	1.0	70.5	29.4	116312.0	199569.0	7.7	10.3	71.6	28.3	1378867.0	2209974.0
81-100	12.4	10.4	70.3	29.6	118534.0	200423.0	10.8	2.4	71.5	28.4	1382270.0	2212080.0
101-120	12.5	5.6	70.2	29.7	122692.0	201993.0	7.6	10.4	71.4	28.5	1385652.0	2214176.0
121-140	10.7	11.9	70.2	29.7	125956.0	203595.0	7.6	5.6	71.3	28.6	1389017.0	2216273.0
141-160	11.0	4.1	70.2	29.7	127655.0	205243.0	7.8	5.7	71.3	28.6	1390400.0	2217646.0
161-180	10.9	7.3	70.1	29.8	129398.0	206931.0	10.9	4.1	71.3	28.6	1391813.0	2219049.0
181-200	1.6	1.0	70.2	29.7	130538.0	208036.0	15.6	12.0	71.3	28.6	1393211.0	2220435.0
201-220	1.5	1.0	70.2	29.7	130557.0	208043.0	0.0	1.0	71.2	28.7	1393852.0	2221058.0
221-240	12.5	4.1	70.1	29.8	130798.0	208216.0	0.0	2.6	71.1	28.8	1393907.0	2221080.0
241-260	7.8	12.0	70.1	29.8	132450.0	209517.0	14.0	5.6	71.2	28.7	1394960.0	2221879.0
261-280	14.0	4.1	70.1	29.8	134718.0	212008.0	16.9	8.6	71.2	28.7	1396384.0	2223175.0
281-300	7.8	5.6	70.1	29.8	137335.0	215544.0	12.5	1.0	71.2	28.7	1398740.0	2226363.0
301-320	9.2	5.7	70.1	29.8	139977.0	219254.0	10.9	7.2	71.2	28.7	1401253.0	2229951.0
321-340	12.0	4.0	69.9	30.0	142628.0	223057.0	12.5	8.6	71.2	28.7	1403849.0	2233733.0
341-360	15.6	5.7	70.1	29.8	145309.0	226981.0	4.6	7.3	71.2	28.7	1406500.0	2237656.0
361-380	12.2	4.0	70.1	29.8	147952.0	230905.0	6.2	1.0	71.2	28.7	1409173.0	2241664.0
381-400	14.0	1.0	70.1	29.8	150635.0	234931.0	6.2	5.7	71.2	28.7	1411838.0	2245700.0
401-420	7.8	4.0	70.1	29.8	153322.0	238993.0	7.8	4.1	71.2	28.7	1414545.0	2249833.0
421-440	1.6	1.0	70.0	29.9	155177.0	241800.0	7.6	7.0	71.2	28.7	1417251.0	2253980.0
441-460	1.6	1.0	70.0	29.9	155178.0	241804.0	7.8	4.1	71.2	28.7	1419960.0	2258156.0
461-480	4.6	1.0	70.1	29.8	155179.0	241808.0	13.9	5.6	71.2	28.7	1422683.0	2262371.0
481-500	10.9	2.6	70.0	29.9	155984.0	243030.0	4.6	7.1	71.1	28.8	1425402.0	2266595.0
501-520	12.3	7.2	70.1	29.8	158658.0	247105.0	12.5	8.6	71.1	28.8	1428131.0	2270845.0
521-540	13.8	1.0	70.1	29.8	161352.0	251246.0	3.1	5.6	71.2	28.7	1430886.0	2275150.0
541-560	9.1	2.5	70.1	29.8	164028.0	255380.0	7.8	4.1	71.2	28.7	1433622.0	2279433.0
561-580	13.4	4.1	70.1	29.8	166716.0	259544.0	9.2	2.5	71.2	28.7	1436363.0	2283732.0
581-600	10.8	5.7	70.1	29.8	169431.0	263764.0	6.1	2.4	71.2	28.7	1439127.0	2288077.0

Tabla 28. RESULTADOS DE MEDICIÓN DE RECURSOS CON CARGA DE 5000 USUARIOS Y TAREAS1 REGISTRO Y 1 COMPRA DURANTE 10 MINUTOS (600 segundos).

Intervalo de Medición (Segundos)	Servidor de Aplicaciones GLASSFISH						Servidor de Aplicaciones JBOSS					
	Recurso CPU		Recurso MEMORIA		Recurso RED		Recurso CPU		Recurso MEMORIA		Recurso RED	
	% Uso Sistema Operativo	% Uso Servidor Aplicaciones	% Memoria Libre	% Memoria en Uso	KBytes TX	KBytes RX	% Uso Sistema Operativo	% Uso Servidor Aplicaciones	% Memoria Libre	% Memoria en Uso	KBytes TX	KBytes RX
0-20	3.1	1.0	73.3	26.6	3394400.0	1247291.0	0.0	2.5	71.5	28.4	1543102.0	2453738.0
21-40	1.5	1.0	73.2	26.7	3394402.0	1247297.0	0.0	1.0	71.4	28.5	1543104.0	2453743.0
41-60	14.0	10.4	72.9	27.0	3396063.0	1247948.0	6.1	4.1	71.3	28.6	1544861.0	2454838.0
61-80	10.7	7.2	72.9	27.0	3400244.0	1249549.0	15.5	13.5	71.1	28.8	1548185.0	2456905.0
81-100	7.8	16.5	72.8	27.1	3403782.0	1251185.0	7.8	2.5	71.1	28.8	1551605.0	2459027.0
101-120	12.3	5.7	72.8	27.1	3405543.0	1252891.0	3.1	4.1	71.0	28.9	1555010.0	2461144.0
121-140	12.3	2.5	72.8	27.1	3407346.0	1254638.0	23.4	4.0	70.9	29.0	1557471.0	2462921.0
141-160	0.0	1.0	72.8	27.1	3408515.0	1255768.0	21.9	8.8	70.9	29.0	1558902.0	2464340.0
161-180	0.0	1.0	72.7	27.2	3408539.0	1255778.0	15.7	9.9	70.9	29.0	1560364.0	2465783.0
181-200	14.9	2.5	72.6	27.3	3408809.0	1255805.0	4.7	1.0	70.9	29.0	1561664.0	2467068.0
201-220	17.1	4.0	72.7	27.2	3410137.0	1256855.0	0.0	1.0	70.9	29.0	1561709.0	2467085.0
221-240	4.7	4.1	72.7	27.2	3412100.0	1258701.0	15.6	7.1	70.9	29.0	1562091.0	2467367.0
241-260	10.7	2.4	72.6	27.3	3414762.0	1262201.0	20.3	7.3	70.9	29.0	1563368.0	2468332.0
261-280	15.6	4.0	72.9	27.0	3417459.0	1265955.0	7.8	3.9	70.9	29.0	1565320.0	2470649.0
281-300	16.8	1.0	73.0	26.9	3420131.0	1269765.0	6.2	7.3	70.9	29.0	1567801.0	2474133.0
301-320	15.6	5.7	73.0	26.9	3422866.0	1273747.0	6.2	8.8	70.9	29.0	1570357.0	2477836.0
321-340	6.0	5.7	73.0	26.9	3425598.0	1277790.0	10.7	7.2	70.9	29.0	1572966.0	2481675.0
341-360	10.8	2.6	73.0	26.9	3428326.0	1281873.0	7.8	4.1	70.9	29.0	1575623.0	2485645.0
361-380	10.8	5.5	73.0	26.9	3431073.0	1286019.0	4.7	5.7	70.9	29.0	1578167.0	2489489.0
381-400	7.8	2.5	73.0	26.9	3433803.0	1290172.0	7.8	5.7	70.9	29.0	1580838.0	2493559.0
401-420	10.9	5.6	73.0	26.9	3436548.0	1294369.0	12.5	7.1	70.8	29.1	1583522.0	2497677.0
421-440	9.4	7.3	73.0	26.9	3439275.0	1298561.0	9.4	4.1	70.9	29.0	1586223.0	2501842.0
441-460	10.7	4.1	73.0	26.9	3442033.0	1302816.0	4.7	5.7	70.8	29.1	1588910.0	2505995.0
461-480	9.4	8.8	73.0	26.9	3444753.0	1307030.0	10.7	10.1	70.8	29.1	1591582.0	2510149.0
481-500	12.4	5.7	73.0	26.9	3447492.0	1311285.0	9.4	5.7	70.9	29.0	1594284.0	2514365.0
501-520	9.2	2.4	73.0	26.9	3450233.0	1315556.0	13.7	2.5	70.9	29.0	1596964.0	2518556.0
521-540	7.6	5.7	73.0	26.9	3452984.0	1319851.0	6.1	5.6	70.9	29.0	1599699.0	2522841.0
541-560	9.3	5.6	73.0	26.9	3455731.0	1324153.0	6.2	2.5	70.9	29.0	1602438.0	2527145.0
561-580	9.3	5.7	73.0	26.9	3458480.0	1328466.0	7.8	4.0	70.8	29.1	1605157.0	2531424.0
581-600	12.1	2.4	73.0	26.9	3461224.0	1332779.0	9.4	5.7	70.8	29.1	1607899.0	2535747.0

Tabla 29. RESULTADOS DE MEDICIÓN DE RECURSOS CON CARGA DE 5000 USUARIOS Y TAREAS 1 REGISTRO Y 2 COMPRAS DURANTE 10 MINUTOS (600 segundos).

Intervalo de Medición (Segundos)	Servidor de Aplicaciones GLASSFISH						Servidor de Aplicaciones JBOSS					
	Recurso CPU		Recurso MEMORIA		Recurso RED		Recurso CPU		Recurso MEMORIA		Recurso RED	
	% Uso Sistema Operativo	% Uso Servidor Aplicaciones	% Memoria Libre	% Memoria en Uso	KBytes TX	KBytes RX	% Uso Sistema Operativo	% Uso Servidor Aplicaciones	% Memoria Libre	% Memoria en Uso	KBytes TX	KBytes RX
0-20	1.5	1.0	69.2	30.7	246033.92	374744.19	0.0	7.3	73.9	26.0	112.0	147.0
21-40	4.7	1.0	69.1	30.8	247933.66	375486.58	0.0	1.0	73.8	26.1	1220.0	838.0
41-60	10.6	4.0	69.4	30.5	251939.99	377023.82	9.3	21.2	73.5	26.4	4348.0	2774.0
61-80	1.6	1.0	69.4	30.5	255358.56	378592.42	7.6	7.3	73.5	26.4	7556.0	4759.0
81-100	12.5	1.0	69.3	30.6	257159.82	380329.27	6.3	11.5	73.4	26.5	10773.0	6751.0
101-120	4.7	17.8	69.1	30.8	258951.45	382055.10	6.0	8.8	73.3	26.6	13923.0	8719.0
121-140	1.6	7.3	69.1	30.8	260168.98	383213.26	9.2	7.2	73.3	26.6	15262.0	10039.0
141-160	3.1	8.8	68.9	31.0	260200.25	383228.72	6.1	4.1	73.2	26.7	16612.0	11368.0
161-180	9.3	5.6	69.0	30.9	261495.00	384246.25	4.7	2.6	73.2	26.7	17979.0	12713.0
181-200	17.2	4.1	68.9	31.0	263477.71	386104.49	0.0	1.0	73.2	26.7	18783.0	13492.0
201-220	6.3	10.2	68.9	31.0	266091.89	389539.27	3.1	1.0	73.2	26.7	18829.0	13516.0
221-240	7.9	1.0	69.0	30.9	268719.80	393191.21	3.1	1.0	73.2	26.7	19715.0	14193.0
241-260	6.2	1.0	68.9	31.0	271415.72	397031.45	1.6	1.0	73.2	26.7	20981.0	15234.0
261-280	15.5	1.0	69.0	30.9	274131.02	400980.22	0.0	1.0	73.1	26.8	23161.0	18130.0
281-300	13.6	7.2	68.9	31.0	276863.33	405019.61	14.0	4.1	73.1	26.8	25587.0	21587.0
301-320	10.9	5.7	68.9	31.0	279578.85	409081.16	12.5	4.1	73.1	26.8	28092.0	25243.0
321-340	9.4	2.6	68.9	31.0	282291.88	413175.01	12.5	7.2	73.1	26.8	30641.0	29016.0
341-360	10.8	2.6	68.9	31.0	285013.22	417310.35	12.5	2.4	73.1	26.8	33240.0	32920.0
361-380	6.1	2.6	68.9	31.0	287737.39	421476.46	6.2	5.6	73.1	26.8	35843.0	36871.0
381-400	7.8	2.6	68.9	31.0	290454.41	425649.10	15.5	5.7	73.1	26.8	38463.0	40873.0
401-420	14.0	6.9	68.9	31.0	293178.05	429849.50	6.3	7.2	73.0	26.9	41086.0	44910.0
421-440	6.3	4.1	68.9	31.0	295913.29	434084.91	9.2	4.1	73.1	26.8	43724.0	48993.0
441-460	10.9	4.1	68.9	31.0	298662.72	438355.09	10.9	4.1	73.1	26.8	46365.0	53098.0
461-480	15.5	2.5	68.9	31.0	301373.24	442576.66	9.4	4.1	73.1	26.8	49027.0	57247.0
481-500	9.4	4.1	68.9	31.0	304095.95	446829.48	10.8	4.0	73.1	26.8	51693.0	61415.0
501-520	16.9	2.5	68.9	31.0	306808.55	451074.55	18.5	1.0	73.1	26.8	54372.0	65618.0
521-540	8.9	5.4	68.9	31.0	309526.31	455338.01	12.4	5.7	73.1	26.8	57050.0	69828.0
541-560	7.6	1.0	68.9	31.0	312262.80	459638.05	3.1	4.2	73.1	26.8	59731.0	74051.0
561-580	9.4	7.3	68.9	31.0	314995.10	463939.24	12.3	4.1	73.1	26.8	62413.0	78286.0
581-600	18.4	2.5	68.8	31.1	317747.26	468278.81	6.1	7.2	73.1	26.8	65112.0	82551.0

Tabla 30. RESULTADOS DE MEDICIÓN DE RECURSOS CON CARGA DE 5000 USUARIOS Y TAREAS1 REGISTRO Y 3 COMPRAS DURANTE 10 MINUTOS (600 segundos).

Intervalo de Medición (Segundos)	Servidor de Aplicaciones GLASSFISH						Servidor de Aplicaciones JBOSS					
	Recurso CPU		Recurso MEMORIA		Recurso RED		Recurso CPU		Recurso MEMORIA		Recurso RED	
	% Uso Sistema Operativo	% Uso Servidor Aplicaciones	% Memoria Libre	% Memoria en Uso	KBytes TX	KBytes RX	% Uso Sistema Operativo	% Uso Servidor Aplicaciones	% Memoria Libre	% Memoria en Uso	KBytes TX	KBytes RX
0-20	0.0	2.5	72.0	27.9	991595.0	3196250.0	0.0	1.0	74.4	25.5	14567.0	107416.0
21-40	0.0	1.0	71.9	28.0	991596.0	3196254.0	0.0	1.0	74.1	25.8	15145.0	107776.0
41-60	3.1	5.7	71.4	28.5	993426.0	3196969.0	9.2	8.6	73.5	26.4	18124.0	109620.0
61-80	4.7	7.2	71.4	28.5	997874.0	3198677.0	10.9	7.2	73.3	26.6	21162.0	111500.0
81-100	12.4	5.6	71.4	28.5	1002356.0	3200397.0	9.3	5.6	73.1	26.8	24270.0	113423.0
101-120	10.8	4.0	71.4	28.5	1006841.0	3202119.0	6.1	13.3	73.0	26.9	27334.0	115318.0
121-140	7.8	7.2	71.3	28.6	1010122.0	3203840.0	3.2	8.6	72.9	27.0	30439.0	117240.0
141-160	12.3	7.2	71.3	28.6	1011982.0	3205644.0	7.5	10.3	72.8	27.1	33486.0	119125.0
161-180	6.2	4.0	71.3	28.6	1013835.0	3207441.0	7.6	8.7	72.8	27.1	36547.0	121019.0
181-200	13.8	4.1	71.3	28.6	1015702.0	3209252.0	11.9	4.1	72.7	27.2	39648.0	122938.0
201-220	3.0	7.1	71.3	28.6	1017565.0	3211061.0	10.9	4.1	72.6	27.3	42358.0	124712.0
221-240	10.9	8.8	71.3	28.6	1019417.0	3212856.0	14.0	4.1	72.6	27.3	43650.0	125985.0
241-260	0.0	1.0	71.3	28.6	1019728.0	3213155.0	15.7	10.3	72.6	27.3	44972.0	127288.0
261-280	0.0	13.4	71.3	28.6	1019762.0	3213168.0	12.4	7.3	72.5	27.4	46277.0	128572.0
281-300	12.5	1.0	71.3	28.6	1019791.0	3213184.0	18.7	8.6	72.5	27.4	47597.0	129871.0
301-320	17.2	5.7	71.2	28.7	1021231.0	3214309.0	15.6	2.4	72.5	27.4	48916.0	131171.0
321-340	6.3	5.6	71.2	28.7	1023055.0	3215732.0	10.9	2.5	72.5	27.4	50243.0	132476.0
341-360	6.1	7.2	71.2	28.7	1025012.0	3217407.0	18.7	4.1	72.5	27.4	51563.0	133770.0
361-380	15.6	5.6	71.2	28.7	1027658.0	3220825.0	0.0	1.0	72.5	27.4	51888.0	134079.0
381-400	7.8	7.2	71.3	28.6	1030401.0	3224631.0	7.8	1.0	72.4	27.5	52468.0	134348.0
401-420	18.7	2.5	71.3	28.6	1033148.0	3228540.0	20.3	7.0	72.4	27.5	53612.0	135219.0
421-440	13.4	2.5	71.3	28.6	1035919.0	3232567.0	10.9	7.1	72.4	27.5	54775.0	136107.0
441-460	7.8	4.2	71.3	28.6	1038699.0	3236676.0	12.5	4.1	72.4	27.5	55911.0	136974.0
461-480	13.6	2.5	71.3	28.6	1041453.0	3240797.0	13.5	5.6	72.4	27.5	57743.0	139253.0
481-500	3.1	1.0	71.3	28.6	1044218.0	3244969.0	14.0	8.6	72.4	27.5	60078.0	142550.0
501-520	13.8	5.7	71.3	28.6	1046987.0	3249179.0	9.4	5.7	72.4	27.5	62485.0	146045.0
521-540	13.8	2.5	71.2	28.7	1049764.0	3253426.0	10.9	4.0	72.4	27.5	64989.0	149736.0
541-560	6.2	2.5	71.2	28.7	1052539.0	3257692.0	10.8	7.2	72.4	27.5	67540.0	153554.0
561-580	13.7	5.7	71.2	28.7	1055330.0	3261999.0	4.7	4.1	72.4	27.5	70098.0	157428.0
581-600	4.6	7.2	71.2	28.7	1058104.0	3266295.0	4.6	7.2	72.4	27.5	72686.0	161381.0

Tabla 31. RESULTADOS DE MEDICIÓN DE RECURSOS CON CARGA DE 10000 USUARIOS Y TAREAS 1 REGISTRO Y 1 COMPRA DURANTE 10 MINUTOS (600 segundos).

Intervalo de Medición (Segundos)	Servidor de Aplicaciones GLASSFISH						Servidor de Aplicaciones JBOSS					
	Recurso CPU		Recurso MEMORIA		Recurso RED		Recurso CPU		Recurso MEMORIA		Recurso RED	
	% Uso Sistema Operativo	% Uso Servidor Aplicaciones	% Memoria Libre	% Memoria en Uso	KBytes TX	KBytes RX	% Uso Sistema Operativo	% Uso Servidor Aplicaciones	% Memoria Libre	% Memoria en Uso	KBytes TX	KBytes RX
0-20	1.6	2.5	69.6	30.3	2587387.0	4134632.0	0.0	1.0	70.4	29.5	406436.0	812205.0
21-40	6.3	5.7	69.5	30.4	2587411.0	4134638.0	9.3	5.7	71.2	28.7	407782.0	813040.0
41-60	7.6	5.7	69.4	30.5	2587455.0	4134663.0	9.4	4.1	71.1	28.8	410783.0	814898.0
61-80	6.0	15.1	69.4	30.5	2587456.0	4134666.0	7.8	2.5	71.0	28.9	413821.0	816779.0
81-100	10.8	4.0	69.2	30.7	2590749.0	4135930.0	4.6	7.2	70.9	29.0	416859.0	818657.0
101-120	10.8	10.2	69.2	30.7	2595093.0	4137594.0	12.3	4.1	70.8	29.1	419903.0	820541.0
121-140	12.1	7.1	69.3	30.6	2599409.0	4139247.0	17.1	7.2	70.8	29.1	422962.0	822434.0
141-160	15.6	1.0	69.3	30.6	2603797.0	4140927.0	10.7	4.1	70.7	29.2	426029.0	824331.0
161-180	10.5	2.5	69.3	30.6	2606415.0	4142627.0	4.7	5.6	70.5	29.4	429061.0	826208.0
181-200	9.4	8.7	69.2	30.7	2608276.0	4144422.0	6.2	4.1	70.6	29.3	432091.0	828083.0
201-220	12.4	4.1	69.2	30.7	2610093.0	4146176.0	9.3	7.2	70.5	29.4	434467.0	829737.0
221-240	10.7	4.0	69.2	30.7	2611918.0	4147938.0	17.2	7.3	70.4	29.5	435802.0	831053.0
241-260	1.5	1.0	69.2	30.7	2613740.0	4149696.0	14.0	2.6	70.4	29.5	437151.0	832382.0
261-280	16.9	7.1	69.2	30.7	2615545.0	4151438.0	21.4	7.1	70.3	29.6	438496.0	833704.0
281-300	15.1	4.0	69.2	30.7	2615670.0	4151558.0	15.5	8.7	70.3	29.6	439851.0	835038.0
301-320	10.5	10.2	69.2	30.7	2615672.0	4151563.0	23.0	4.1	70.3	29.6	441201.0	836368.0
321-340	10.8	2.4	69.2	30.7	2615705.0	4151575.0	11.0	5.7	70.3	29.6	442542.0	837686.0
341-360	6.2	5.4	69.1	30.8	2616724.0	4152366.0	7.7	1.0	70.4	29.5	443746.0	838864.0
361-380	10.9	4.1	69.1	30.8	2618453.0	4153730.0	0.0	1.0	70.4	29.5	443788.0	838887.0
381-400	16.8	2.5	69.0	30.9	2620134.0	4155056.0	15.6	8.8	70.3	29.6	444571.0	839468.0
401-420	7.6	11.7	69.1	30.8	2622637.0	4157996.0	7.5	10.0	70.3	29.6	446073.0	840599.0
421-440	6.2	7.3	69.1	30.8	2625315.0	4161655.0	10.7	5.6	70.3	29.6	447551.0	841711.0
441-460	10.5	4.0	69.1	30.8	2628042.0	4165502.0	16.9	10.1	70.3	29.6	449552.0	844172.0
461-480	7.7	2.4	69.1	30.8	2630793.0	4169471.0	9.1	5.6	70.2	29.7	451914.0	847504.0
481-500	14.0	2.6	69.1	30.8	2633528.0	4173491.0	18.7	7.1	70.3	29.6	454341.0	851029.0
501-520	13.8	8.7	69.1	30.8	2636275.0	4177582.0	12.3	5.7	70.2	29.7	456769.0	854605.0
521-540	7.8	8.8	69.1	30.8	2639021.0	4181712.0	12.5	7.3	70.3	29.6	459227.0	858281.0
541-560	9.2	2.6	69.1	30.8	2641778.0	4185892.0	13.7	2.5	70.3	29.6	461692.0	862011.0
561-580	12.4	1.0	69.1	30.8	2644496.0	4190039.0	12.2	5.5	70.1	29.8	464203.0	865839.0
581-600	7.8	4.1	69.1	30.8	2647250.0	4194263.0	9.3	5.6	70.2	29.7	466668.0	869623.0

Tabla 32. RESULTADOS DE MEDICIÓN DE RECURSOS CON CARGA DE 10000 USUARIOS Y TAREAS 1 REGISTRO Y 2 COMPRAS DURANTE 10 MINUTOS (600 segundos).

Intervalo de Medición (Segundos)	Servidor de Aplicaciones GLASSFISH						Servidor de Aplicaciones JBOSS					
	Recurso CPU		Recurso MEMORIA		Recurso RED		Recurso CPU		Recurso MEMORIA		Recurso RED	
	% Uso Sistema Operativo	% Uso Servidor Aplicaciones	% Memoria Libre	% Memoria en Uso	KBytes TX	KBytes RX	% Uso Sistema Operativo	% Uso Servidor Aplicaciones	% Memoria Libre	% Memoria en Uso	KBytes TX	KBytes RX
0-20	1.6	2.6	71.7	28.2	469714.31	230895.52	0.0	1.0	70.4	29.5	406436.0	812205.0
21-40	9.4	11.9	71.4	28.5	472267.19	231084.75	9.3	5.7	71.2	28.7	407782.0	813040.0
41-60	10.9	7.2	71.2	28.7	476313.05	232683.08	9.4	4.1	71.1	28.8	410783.0	814898.0
61-80	4.5	7.1	71.3	28.6	480409.84	234361.84	7.8	2.5	71.0	28.9	413821.0	816779.0
81-100	10.9	4.1	71.3	28.6	484422.16	236037.88	4.6	7.2	70.9	29.0	416859.0	818657.0
101-120	12.4	5.7	71.3	28.6	487894.57	237681.82	12.3	4.1	70.8	29.1	419903.0	820541.0
121-140	10.9	8.8	71.3	28.6	489599.63	239398.43	17.1	7.2	70.8	29.1	422962.0	822434.0
141-160	9.3	7.2	71.3	28.6	491360.81	241159.25	10.7	4.1	70.7	29.2	426029.0	824331.0
161-180	13.8	2.6	71.3	28.6	493102.23	242915.59	4.7	5.6	70.5	29.4	429061.0	826208.0
181-200	6.2	4.0	71.3	28.6	494833.47	244676.05	6.2	4.1	70.6	29.3	432091.0	828083.0
201-220	9.4	4.1	71.3	28.6	496591.60	246426.05	9.3	7.2	70.5	29.4	434467.0	829737.0
221-240	0.0	1.0	71.3	28.6	497891.38	247814.43	17.2	7.3	70.4	29.5	435802.0	831053.0
241-260	0.0	1.0	71.3	28.6	497975.07	247819.70	14.0	2.6	70.4	29.5	437151.0	832382.0
261-280	4.7	1.0	71.3	28.6	498882.29	247841.53	21.4	7.1	70.3	29.6	438496.0	833704.0
281-300	17.1	4.0	71.2	28.7	500498.92	248441.99	15.5	8.7	70.3	29.6	439851.0	835038.0
301-320	9.3	10.4	71.2	28.7	502164.73	249784.87	23.0	4.1	70.3	29.6	441201.0	836368.0
321-340	6.2	7.2	71.2	28.7	504360.47	251098.29	11.0	5.7	70.3	29.6	442542.0	837686.0
341-360	9.0	10.4	71.2	28.7	506940.79	253623.03	7.7	1.0	70.4	29.5	443746.0	838864.0
361-380	15.1	4.1	71.2	28.7	509540.95	257342.53	0.0	1.0	70.4	29.5	443788.0	838887.0
381-400	10.7	4.0	71.2	28.7	512178.36	261263.34	15.6	8.8	70.3	29.6	444571.0	839468.0
401-420	17.2	5.6	71.2	28.7	514811.97	265354.30	7.5	10.0	70.3	29.6	446073.0	840599.0
421-440	7.8	5.7	71.2	28.7	517396.84	269510.33	10.7	5.6	70.3	29.6	447551.0	841711.0
441-460	6.1	8.5	71.2	28.7	520013.55	273739.46	16.9	10.1	70.3	29.6	449552.0	844172.0
461-480	6.2	2.5	71.2	28.7	522664.05	277995.55	9.1	5.6	70.2	29.7	451914.0	847504.0
481-500	10.9	10.3	71.2	28.7	525285.81	282310.59	18.7	7.1	70.3	29.6	454341.0	851029.0
501-520	10.7	4.1	71.2	28.7	527902.41	286648.87	12.3	5.7	70.2	29.7	456769.0	854605.0
521-540	7.6	5.7	71.2	28.7	530541.85	291025.03	12.5	7.3	70.3	29.6	459227.0	858281.0
541-560	11.0	4.1	71.2	28.7	533148.46	295413.42	13.7	2.5	70.3	29.6	461692.0	862011.0
561-580	12.1	7.3	71.2	28.7	535792.26	299814.48	12.2	5.5	70.1	29.8	464203.0	865839.0
581-600	7.8	1.0	71.2	28.7	538399.09	304262.34	9.3	5.6	70.2	29.7	466668.0	869623.0

Tabla 33. RESULTADOS DE MEDICIÓN DE RECURSOS CON CARGA DE 10000 USUARIOS Y TAREAS 1 REGISTRO Y 3 COMPRAS DURANTE 10 MINUTOS (600 segundos).

ANEXO B. VALORES ESTADÍSTICOS DE MEDICIÓN DE LOS COMPONENTES HARDWARE

A continuación se muestra la información de cada uno de los valores estadísticos obtenidos por los servidores de aplicaciones al interactuar un número definido de usuarios y el tipo de tareas que realizaron en el tiempo designado para la medición.

Estas medidas son el resultado de los datos obtenidos y presentados en las tablas referentes al ANEXO A.

Número Usuarios	Tiempo Total	Intervalos	Tareas	
			Registro	Compras
100	10 minutos	20 segundos	1	1

PARAMETRO		GLASSFISH					JBOSS				
Hardware	Valor Representativo	Promedio	Moda	Media	Desviación Estándar	Rendimiento	Promedio	Moda	Media	Desviación Estándar	Rendimiento
CPU	% Uso Sistema Operativo	0,1699	0,0	1,55	0,622	10,758	0,1566	0,0	1,6	0,6366	11,674
	% Uso Servidor Aplicaciones	1,1066	1,0	2,6	0,5842	1,6527	1,6066	1,0	4,75	1,5633	1,1383
MEMORIA	% en Uso	29,833	29,8	29,85	0,088	22,357	27,536	27,6	27,35	0,2413	24,222
	% Libre	70,066	70,1	70,050	0,088	9,5195	72,363	72,3	72,55	0,2413	9,2173
RED	KBytes Transmitidos	33,025	0,8085	764,13	95,503	0,0555	37,0	2,0	188,5	99,171	0,0616
	KBytes Recibidos	21,833	5,0	159,5	65,868	0,0363	27,9	5,0	201,5	86,219	0,0465

Tabla 34. Resultados de las medidas estadísticas obtenidas al momento que el número de usuarios es 100 y las tareas son 1 Registro y 1 Compra.

Número Usuarios	Tiempo Total	Intervalos	Tareas	
			Registro	Compras
100	10 minutos	20 segundos	1	2

PARAMETRO		GLASSFISH					JBOSS				
Hardware	Valor Representativo	Promedio	Moda	Media	Desviación Estándar	Rendimiento	Promedio	Moda	Media	Desviación Estándar	Rendimiento
CPU	% Uso Sistema Operativo	0,2866	0,0	2,25	1,0922	6,3802	0,2033	0,0	1,55	0,7739	8,9950
	% Uso Servidor Aplicaciones	1,3633	1,0	4,9	1,4585	1,3415	1,41	1,0	7,15	2,2456	1,2971
MEMORIA	% en Uso	31,576	31,7	31,4	0,1755	21,123	28,789	28,9	28,65	0,1539	23,167
	% Libre	68,323	68,2	68,5	0,1755	9,7624	71,11	71,0	71,25	0,1539	9,3798
RED	KBytes Transmitidos	39,3	1,0	377,5	147,44	0,0654	42,833	1,0	390,5	156,70	0,0713
	KBytes Recibidos	35,9	5,0	373,5	138,78	0,0598	44,433	4,0	451,5	169,44	0,0740

Tabla 35. Resultados de las medidas estadísticas obtenidas al momento que el número de usuarios es 100 y las tareas son 1 Registro y 2 Compras.

Número Usuarios	Tiempo Total	Intervalos	Tareas	
			Registro	Compras
100	10 minutos	20 segundos	1	3

PARAMETRO		GLASSFISH					JBOSS				
Hardware	Valor Representativo	Promedio	Moda	Media	Desviación Estándar	Rendimiento	Promedio	Moda	Media	Desviación Estándar	Rendimiento
CPU	% Uso Sistema Operativo	0,5166	0,0	6,25	2,2949	3,5399	0,5	0,0	3,75	1,546	3,658
	% Uso Servidor Aplicaciones	1,0466	1,0	1,7	0,2556	1,7474	1,1533	1,0	3,3	0,8398	1,5858
MEMORIA	% en Uso	31,620	31,7	31,4	0,1788	21,094	30,723	30,7	30,65	0,122	21,709
	% Libre	68,280	68,2	68,5	0,1788	9,7685	69,176	69,2	69,25	0,122	9,6419
RED	KBytes Transmitidos	56,366	1,0	556,5	209,36	0,0939	60,4	1,0	606,5	222,76	0,1006
	KBytes Recibidos	56,033	4,0	591,0	217,58	0,0933	67,666	4,0	724,5	266,93	0,1127

Tabla 36. Resultados de las medidas estadísticas obtenidas al momento que el número de usuarios es 100 y las tareas son 1 Registro y 3 Compras.

Número Usuarios	Tiempo Total	Intervalos	Tareas	
			Registro	Compras
1000	10 minutos	20 segundos	1	1

PARAMETRO		GLASSFISH					JBOSS				
Hardware	Valor Representativo	Promedio	Moda	Media	Desviación Estándar	Rendimiento	Promedio	Moda	Media	Desviación Estándar	Rendimiento
CPU	% Uso Sistema Operativo	4,3066	1,5	11,55	6,035	0,4246	2,01	0,0	7,65	4,6772	0,9099
	% Uso Servidor Aplicaciones	1,9766	1,0	9,55	3,2947	0,9252	2,2433	1,0	11,15	3,8913	0,8153
MEMORIA	% en Uso	29,766	30,0	29,5	0,3427	22,407	28,276	28,3	28,2	0,1406	23,588
	% Libre	70,133	69,9	70,4	0,3427	9,5104	71,623	71,6	71,7	0,1406	9,3126
RED	KBytes Transmitidos	399,46	1,0	1322,0	788,90	0,6657	415,0	24,0	1246,5	815,60	0,6916
	KBytes Recibidos	429,66	4,0	1823,0	1007,1	0,7161	473,2	6,0	1804,0	1023,3	0,7886

Tabla 37. Resultados de las medidas estadísticas obtenidas al momento que el número de usuarios es 1000 y las tareas son 1 Registro y 1 Compra.

Número Usuarios	Tiempo Total	Intervalos	Tareas	
			Registro	Compras
1000	10 minutos	20 segundos	1	2

PARAMETRO		GLASSFISH					JBOSS				
Hardware	Valor Representativo	Promedio	Moda	Media	Desviación Estándar	Rendimiento	Promedio	Moda	Media	Desviación Estándar	Rendimiento
CPU	% Uso Sistema Operativo	4,7466	1,6	10,75	5,0079	0,3853	4,3366	0,0	7,0	5,0888	0,4217
	% Uso Servidor Aplicaciones	2,7399	1,0	6,35	3,11	0,6675	3,2666	1,0	5,7	3,0878	0,5598
MEMORIA	% en Uso	31,043	31,2	30,85	0,2416	21,486	27,669	27,7	27,55	0,1417	24,105
	% Libre	68,856	68,7	69,05	0,2416	9,6867	72,23	72,2	72,35	0,1417	9,2343
RED	KBytes Transmitidos	1015,7	2,0	1391,5	1238,8	1,6928	1035,0	2,0	1329,0	1227,2	1,7251
	KBytes Recibidos	1344,2	4,0	2088,5	1797,7	2,2404	1418,0	4,0	2061,5	1811,7	2,3634

Tabla 38. Resultados de las medidas estadísticas obtenidas al momento que el número de usuarios es 1000 y las tareas son 1 Registro y 2 Compras.

Número Usuarios	Tiempo Total	Intervalos	Tareas	
			Registro	Compras
1000	10 minutos	20 segundos	1	3

PARAMETRO		GLASSFISH					JBOSS				
Hardware	Valor Representativo	Promedio	Moda	Media	Desviación Estándar	Rendimiento	Promedio	Moda	Media	Desviación Estándar	Rendimiento
CPU	% Uso Sistema Operativo	6,6766	0,0	7,8	4,6028	0,2739	4,5799	0,0	7,0	4,7960	0,3993
	% Uso Servidor Aplicaciones	3,9366	1,0	6,5	2,8834	0,4646	4,0699	1,0	10,2	3,5873	0,4493
MEMORIA	% en Uso	30,906	30,9	30,8	0,1311	21,581	28,323	28,5	28,2	0,1794	23,549
	% Libre	68,993	69,0	69,1	0,1311	9,6676	71,576	71,4	71,7	0,1794	9,3186
RED	KBytes Transmitidos	1961,0	1,0	1347,5	1168,9	3,2683	2001,5	1,0	1355,0	970,15	3,3358
	KBytes Recibidos	2849,4	5,0	2111,0	1804	4,7491	2961,8	3,0	2072,5	1565,2	4,9364

Tabla 39. Resultados de las medidas estadísticas obtenidas al momento que el número de usuarios es 1000 y las tareas son 1 Registro y 3 Compras.

Número Usuarios	Tiempo Total	Intervalos	Tareas	
			Registro	Compras
5000	10 minutos	20 segundos	1	1

PARAMETRO		GLASSFISH					JBOSS				
Hardware	Valor Representativo	Promedio	Moda	Media	Desviación Estándar	Rendimiento	Promedio	Moda	Media	Desviación Estándar	Rendimiento
CPU	% Uso Sistema Operativo	8,5433	1,6	7,8	4,9280	0,2140	8,3099	7,8	8,45	4,7040	0,2200
	% Uso Servidor Aplicaciones	4,0533	1,0	6,5	3,2454	0,4512	5,3633	1,0	6,5	3,1326	0,3410
MEMORIA	% en Uso	29,733	29,8	29,6	0,1728	22,432	28,620	28,7	28,450	0,1788	23,305
	% Libre	70,166	70,1	70,300	0,1728	9,5059	71,28	71,2	71,449	0,1788	9,3574
RED	KBytes Transmitidos	1771,5	1,0	2079,0	1771,5	2,9526	2129,8	2,0	1701,5	1057,6	3,5497
	KBytes Recibidos	2140,4	4,0	2110,0	1688	3,5673	2678,8	4,0	2172,5	1609,6	4,4646

Tabla 40. Resultados de las medidas estadísticas obtenidas al momento que el número de usuarios son 5000 y las tareas son 1 Registro y 1 Compra.

Número Usuarios	Tiempo Total	Intervalos	Tareas	
			Registro	Compras
5000	10 minutos	20 segundos	1	2

PARAMETRO		GLASSFISH					JBOSS				
Hardware	Valor Representativo	Promedio	Moda	Media	Desviación Estándar	Rendimiento	Promedio	Moda	Media	Desviación Estándar	Rendimiento
CPU	% Uso Sistema Operativo	9,7600	10,7	8,55	4,5625	0,1873	9,1733	7,8	11,7	6,0248	0,1993
	% Uso Servidor Aplicaciones	4,6533	5,7	8,75	3,2941	0,3930	5,4166	5,7	7,25	2,9504	0,3376
MEMORIA	% en Uso	26,976	26,9	26,950	0,1590	24,725	28,950	29,0	28,75	0,1696	23,039
	% Libre	72,923	73,0	72,949	0,1590	9,1465	70,950	70,9	71,15	0,1696	9,4009
RED	KBytes Transmitidos	2227,4	2747,0	2090,5	1040,4	3,7124	2159,9	2,0	1710,0	989,41	3,5998
	KBytes Recibidos	2849,6	4255,0	2156,5	1656	4,7493	2733,6	5,0	2161,5	1567,7	4,5560

Tabla 41. Resultados de las medidas estadísticas obtenidas al momento que el número de usuarios son 5000 y las tareas son 1 Registro y 2 Compras.

Número Usuarios	Tiempo Total	Intervalos	Tareas	
			Registro	Compras
5000	10 minutos	20 segundos	1	3

PARAMETRO		GLASSFISH					JBOSS				
Hardware	Valor Representativo	Promedio	Moda	Media	Desviación Estándar	Rendimiento	Promedio	Moda	Media	Desviación Estándar	Rendimiento
CPU	% Uso Sistema Operativo	9,2866	9,4	9,95	4,7162	0,1969	7,6400	0,0	9,25	4,9836	0,2393
	% Uso Servidor Aplicaciones	4,2833	1,0	9,4	3,6387	0,4270	5,0266	1,0	11,1	4,0777	0,3638
MEMORIA	% en Uso	30,916	31,0	30,8	0,1555	21,574	26,683	26,8	26,45	0,2102	24,996
	% Libre	68,983	68,9	69,1	0,1555	9,6690	73,216	73,1	73,45	0,2102	9,1099
RED	KBytes Transmitidos	2390,44	1899,7	2003,1	845,81	3,9840	2166,6	1108,0	1608,5	911,40	3,6111
	KBytes Recibidos	3117,82	742,39	2169,7	1488,2	5,1963	2746,8	691,0	2132,5	1505,0	4,578

Tabla 42. Resultados de las medidas estadísticas obtenidas al momento que el número de usuarios son 5000 y las tareas son 1 Registro y 3 Compras.

Número Usuarios	Tiempo Total	Intervalos	Tareas	
			Registro	Compras
10000	10 minutos	20 segundos	1	1

PARAMETRO		GLASSFISH					JBOSS				
Hardware	Valor Representativo	Promedio	Moda	Media	Desviación Estándar	Rendimiento	Promedio	Moda	Media	Desviación Estándar	Rendimiento
CPU	% Uso Sistema Operativo	8,6400	0,0	9,35	5,50	0,2116	10,066	10,9	10,15	5,3265	0,1816
	% Uso Servidor Aplicaciones	4,9133	2,5	7,2	4,9133	0,3722	5,8133	4,1	7,15	3,1241	0,3146
MEMORIA	% en Uso	28,570	28,6	28,299	0,1803	23,346	27,173	27,5	26,5	0,5051	24,546
	% Libre	71,329	71,3	71,6	0,1803	9,3509	72,726	72,4	73,4	0,5051	9,1713
RED	KBytes Transmitidos	2216,9	1,0	2242,5	1239,2	3,6949	1937,3	1320,0	1554,0	969,66	3,2288
	KBytes Recibidos	2334,8	4,0	2153,5	1565,9	3,8913	1798,8	360,0	1976,5	1117,9	2,9980

Tabla 43. Resultados de las medidas estadísticas obtenidas al momento que el número de usuarios son 10000 y las tareas son 1 Registro y 1 Compra.

Número Usuarios	Tiempo Total	Intervalos	Tareas	
			Registro	Compras
10000	10 minutos	20 segundos	1	2

PARAMETRO		GLASSFISH					JBOSS				
Hardware	Valor Representativo	Promedio	Moda	Mediana	Desviación Estándar	Rendimiento	Promedio	Moda	Mediana	Desviación Estándar	Rendimiento
CPU	% Uso Sistema Operativo	9,9833	10,8	9,2	3,8525	0,1832	10,836	9,3	10,95	5,9678	0,1687
	% Uso Servidor Aplicaciones	5,35	4,0	8,05	3,4868	0,3418	6,2099	5,4	6,4	3,0247	0,2945
MEMORIA	% en Uso	30,699	30,8	30,6	0,1339	21,726	30,850	31,1	30,200	0,4281	21,620
	% Libre	69,199	69,1	69,3	0,1339	9,6387	69,049	68,8	69,699	0,4281	9,6596
RED	KBytes Transmitidos	1995,4	24,0	2194,0	1340,6	3,3257	1972,1	964,0	1605,0	1008,5	3,2869
	KBytes Recibidos	1987,7	6,0	2112,0	1543,8	3,3128	1860,2	600,0	2001,0	1157,3	3,1003

Tabla 44. Resultados de las medidas estadísticas obtenidas al momento que el número de usuarios son 10000 y las tareas son 1 Registro y 2 Compras.

Número Usuarios	Tiempo Total	Intervalos	Tareas	
			Registro	Compras
10000	10 minutos	20 segundos	1	3

PARAMETRO		GLASSFISH					JBOSS				
Hardware	Valor Representativo	Promedio	Moda	Media	Desviación Estándar	Rendimiento	Promedio	Moda	Media	Desviación Estándar	Rendimiento
CPU	% Uso Sistema Operativo	8,9599	10,9	8,6	4,2799	0,2041	10,22	7,7	9,4	4,0359	0,1789
	% Uso Servidor Aplicaciones	5,44	4,1	6,45	3,0499	0,3362	6,1899	5,6	6,5	3,0040	0,2954
MEMORIA	% en Uso	28,640	28,7	28,45	0,1003	23,289	27,549	27,7	27,25	0,2968	24,210
	% Libre	71,26	71,2	71,45	0,1003	9,3600	72,349	72,2	72,65	0,2968	9,2190
RED	KBytes Transmitidos	2289,49	2552,8	2048,3	974,90	3,8158	2036,8	407,0	1683,0	1044,6	3,3947
	KBytes Recibidos	2445,56	189,23	2223,9	1588,7	4,0759	1961,4	257,0	2023,5	1211,8	3,269

Tabla 45. Resultados de las medidas estadísticas obtenidas al momento que el número de usuarios son 10000 y las tareas son 1 Registro y 3 Compras.

TIEMPOS DE RESPUESTA

Número de Usuarios	Número de Medidas por Petición		GLASSFISH				JBOSS			
	Registro	Compras	Medidas (ms)		Valores Tiempos de Respuesta		Medidas (ms)		Valores Tiempos de Respuesta	
			Registro	Compras	Promedio	Rendimiento	Registro	Compras	Promedio	Rendimiento
100	1	1	2875,0	3250,0	3,0625	32,653	3250,0	3672,0	3,4610	28,893
	1	2	3453,0	6656,0	5,0545	19,784	3547,0	7625,0	5,586	17,901
	1	3	3125,0	10891,0	7,008	14,269	3593,0	13047,0	8,32	12,019
1000	1	1	22547,0	64703,0	43,625	22,922	32110,0	72766,0	52,438	19,070
	1	2	22609,0	204578,0	113,59	8,8033	28609,0	208110,0	118,35	8,4488
	1	3	23031,0	422297,0	222,66	4,4910	33296,0	450875,0	242,08	4,1307
5000	1	1	107313,0	1130907,0	619,11	8,0761	151938,0	1099922,0	625,93	7,9881
	1	2	108828,0	4134718,0	2121,7	2,3565	168969,0	4311672,0	2240,3	2,2318
	1	3	99171,0	8685656,0	4392,4	1,1383	141141,0	8986391,0	4563,7	1,0955
10000	1	1	193047,0	3986422,0	2089,7	4,7852	279250,0	4035312,0	2157,2	4,6354
	1	2	196531,0	15047656,0	7622,0	1,3119	355750,0	15935000,0	8145,3	1,2276

Tabla 46. Medición de los Tiempos de Respuesta.

TIEMPOS DE RESPUESTA

Número de Usuarios	Número de Medidas por Petición		GLASSFISH				JBOSS			
	Registro	Compras	Medidas (ms)		Valores Tiempos de Respuesta		Medidas (ms)		Valores Tiempos de Respuesta	
			Registro	Compras	Promedio	Rendimiento	Registro	Compras	Promedio	Rendimiento
1	1	1	78,0	78,0	0,078	12,820	78,0	78,0	0,078	12,820
	1	2	62,0	79,0	0,0705	14,184	94,0	94,0	0,094	10,638
	1	3	63,0	93,0	0,078	12,820	109,0	125,0	0,1169	8,5470
50	1	1	2171,0	1734,0	1,9525	25,608	1844,0	1969,0	1,9065	26,226
	1	2	2141,0	2969,0	2,5549	19,569	2031,0	4156,0	3,0934	16,162
	1	3	2203,0	4625,0	3,4139	14,645	1844,0	6391,0	4,1175	12,143
500	1	1	12312,0	23687,0	17,999	27,778	17765,0	29688,0	23,726	21,073
	1	2	12453,0	62156,0	37,304	13,403	17625,0	73860,0	45,742	10,930
	1	3	12094,0	121516,0	66,805	7,4844	18063,0	137187,0	77,625	6,4412

Tabla 47. Medición de los Tiempos de Respuesta.

ANEXO C. CONFIGURACION DE ARCHIVOS XML PARA DESPLEGAR LA APLICACIÓN WEB EN EL SERVIDOR DE APLICACIONES JBOSS

Como se señaló previamente en el capítulo 2, la configuración de los siguientes archivos a detallar es necesaria para que Portal Web pueda ejecutarse bajo el Servidor de Aplicaciones JBOSS.

Archivo jboss-ds.xml

El siguiente archivo es útil para poder realizar el pool de conexiones que permitirá conectar el Portal Web al sistema de base de datos.

Cabe recalcar que el servidor JBOSS en su ruta de instalación docs\examples\jca a manera de ejemplo posee un listado del formato del tipo de pool de conexiones de acuerdo al motor de base de datos que se quiera utilizar en este caso la opción a tomar en cuenta y que se tomara como referencia es el archivo oracle-ds.xml.

Al ser un archivo XML este esta formado por etiquetas que cumplen una función específica de las cuales las más relevantes y necesarias son detalladas.

jndi-name: Esta etiqueta posee el nombre con el cual se identifica al motor de base de datos que se usara, en este caso y de la forma como esta escrita se lo debe colocar OracleDS.

connection-url: Esta etiqueta detalla la configuración necesaria de la conexión para el motor de base de datos de acuerdo a lo que se menciono con el uso apropiado del driver jdbc que servirá de puente de comunicación y el valor es jdbc:oracle:thin:@ratoncita:1521:TESIS.

Jdbc: indica el modo de conexión.

Oracle: es el nombre del motor de base de datos a conectar

Thin: especificación propia del driver jdbc.

@ratoncita: se indica el nombre de host de la maquina en la que reside el motor de base de datos.

1521: indica el puerto en donde escucha el motor de base de datos

TESIS: indica el nombre de la base de datos a conectarse.

Posteriormente la etiqueta driver-class, indica la características propias del driver jdbc definidas por el propio servidor de aplicaciones

Las etiquetas user-name y password indican los valores de acceso a la base de datos definidos por el usuario.

Las demás etiquetas como min-pool-size, max-pool-size, idle-timeout-minutes, etc. son características propias que se definen para el pool de conexiones todos estos valores son a criterio del administrador.

Cabe indicar que si al momento de desarrollar con la ayuda de un IDE que permita conectarse con el servidor JBOSS, la creación de este archivo se lo hace de una manera más amigable par que el usuario coloque la información detallada según su criterio.

Archivos standardjaws.xml y standardjbosscmp-jdbc.xml.

Estos dos archivos se los ubica en la ruta de instalación del servidor server\default\conf. Con el archivo standardjaws.xml se define la fuente de los datos por medio de la etiqueta datasource, definido esto se coloca java:/OracleDS esta forma de colocar tiene relevancia para que el proyecto no de errores de conexión, y por último en la etiqueta type-mapping se coloca la versión del motor de base de datos con la que se trabaja en este caso el valor es Oracle9.

En el archivo standardjbosscmp-jdbc.xml, es un archivo de información complementaria al anterior por lo que también se indica el nombre del datasource, con las etiquetas datasource el valor a colocar es java:/OracleDS y con la etiqueta datasource-mapping se coloca la versión del motor de base de datos Oracle9i.

Todo lo explicado permitirá que el servidor de aplicaciones pueda mantener una conectividad segura con el motor de base de datos, es importante que la forma como se colocan los valores de etiquetas tomarlas en cuenta ya que son formatos que el servidor define para este tipo de servicio.

ANEXO D. DETALLE DE OBTENCION DE DATOS PARA CADA UNO DE LOS RECURSOS A MEDIR Y TIEMPOS DE RESPUESTA

CPU

PROMEDIO= suma de todos los datos / numero total de mediciones

MODA= valor de las medición que se repiten mas veces

MEDIA= la suma del mayor valor con el de menor valor / 2

REPETICIONES MODA = el numero de veces que se repite el valor de MODA

RENDIMIENTO= velocidad cpu (Ghz)/ PROMEDIO

DESVIACIÓN ESTANDAR= en base a la definición dada para su cálculo sobre una población de datos.

MEMORIA

PROMEDIO= suma de todos los datos / numero total de mediciones

MODA= valor de las mediciones que se repiten mas veces

MEDIA= la suma del valor mas alto con el valor mas bajo / 2

REPETICIONES MODA = el numero de veces que se repite el valor de MODA

RENDIMIENTO= velocidad de memoria / PROMEDIO

DESVIACIÓN ESTANDAR= en base a la definición dada para su cálculo sobre una población de datos.

RED

PROMEDIO= suma de todos los datos / numero total de mediciones

MODA= valor de las mediciones que se repiten mas veces

MEDIA= la suma del valor mas alto con el valor mas bajo / 2

REPETICIONES MODA = el numero de veces que se repite el valor de MODA

RENDIMIENTO= PROMEDIO / tiempo total medición (600 seg.)

DESVIACIÓN ESTANDAR= en base a la definición dada para su cálculo sobre una población de datos.

TIEMPOS DE RESPUESTA

PROMEDIO= suma de todos los datos / numero total de mediciones

MODA= valor de las mediciones que se repiten mas veces

MEDIA= la suma del valor mas alto con el valor mas bajo / 2

REPETICIONES MODA = el numero de veces que se repite el valor de MODA

RENDIMIENTO= Número de usuarios de la medición / PROMEDIO

DESVIACIÓN ESTANDAR= en base a la definición dada para su cálculo sobre una población de datos.

CONTENIDO

INTRODUCCIÓN.....	1
CAPÍTULO 1. PLANTEAMIENTO DEL PROBLEMA.....	13
1.1 DESCRIPCIÓN DEL PROBLEMA.....	13
1.2 ANÁLISIS COMPARATIVO DE CARACTERÍSTICAS ENTRE JBOSS Y GLASSFISH BAJO ARQUITECTURA J2EE.....	14
1.2.1 ¿QUÉ SON GLASSFISH Y JBOSS?.....	15
1.2.1.1 Arquitectura JBoss.....	16
1.2.1.1.1 JMX.....	16
1.2.1.1.2 Servicios Proporcionados Por JBoss.....	19
1.2.1.1.3 Otros Servicios Proporcionados Por JBoss.....	22
1.2.1.2 Arquitectura Glassfish.....	24
1.2.1.2.1 Componentes De La Arquitectura Glassfish.....	24
1.2.2 ANALISIS COMPARATIVO DE CARACTERISTICAS DE JBOSS Y GLASSFISH.....	29
1.3 SELECCIÓN DE BENCHMARK ESTANDAR APLICADO A CASO DE ESTUDIO.....	37
1.3.1 BENCH MARK.....	37
1.3.1.1 Planeación.....	38
1.3.1.1.1 Tiempo De Respuesta.....	39
1.3.1.1.2 Rendimiento.....	39
1.3.1.1.3 Utilización De Los Recursos.....	40
1.3.1.1.4 Parámetros Estadísticos.....	40
1.3.1.1.5 Tipos De Diagramas.....	41
CAPÍTULO 2. ELABORACIÓN DEL BENCHMARK.....	43
2.1. SELECCIÓN DE LA METODOLOGÍA DE DESARROLLO.....	43
2.1.1. RATIONAL UNIFIED PROCESS (RUP) Y UNIFIED MODELING LANGUAGE (UML).....	44
2.2. ANÁLISIS DEL BENCHMARK.....	45
2.2.1. PANORAMA GENERAL (PRESENTACIÓN GENERAL).....	45
2.2.2. USUARIOS.....	46
2.2.3. METAS.....	46
2.2.4. FUNCIONES DEL BENCHMARK.....	46
2.2.4.1. Funciones de Portal Web.....	46
2.2.4.2. Funciones de Medición y Representación gráfica.....	47
2.2.5. ATRIBUTOS DEL BENCHMARK.....	48
2.2.5.1. Atributos generales del sistema.....	48
2.2.5.2. Atributos respecto de las funciones:.....	48
2.2.5.3. Atributos de Medición y Representación gráfica.....	49
2.2.6. MODELOS DE ANÁLISIS.....	50
2.2.6.1. Modelo de Casos de Uso.....	51
2.2.6.1.1. Frontera del Software de Benchmark.....	51
2.2.6.1.2. Actores del Software de Benchmark.....	51
2.2.6.1.3. Casos de uso de Análisis.....	51
2.2.6.2. Modelo Conceptual.....	67
2.2.6.3. Modelo de Comportamiento.....	69
2.2.6.3.1. Diagrama de secuencias.....	69

2.3. DISEÑO DEL BENCHMARK.....	73
2.3.1. MODELO DE CASOS DE USO DE DISEÑO.....	73
2.3.1.1. Casos de Uso de Diseño.....	73
2.3.1.1.1. Caso de Uso: C1. Medir Índices de desempeño.....	73
2.3.1.1.2 Caso de Uso: C2.Verificar Mediciones almacenadas.....	79
2.3.2. MODELO DE COMPORTAMIENTO DE OBJETOS.....	80
2.3.2.1. Diagramas de Interacción.....	81
2.3.2.1.1. Interacción de Medir Desempeño de Servidor de Aplicaciones.....	81
2.3.2.1.2. Interacción de Medir Tiempo de Respuesta.....	81
2.3.2.1.3. Interacción de Verificar Mediciones Almacenadas.....	81
2.3.2.1.4. Interacción de Comprar Productos.....	82
2.3.2.1.5. Interacción de Registrar Datos de Cliente.....	82
2.3.2.2. Interfaces de Benchmark.....	83
2.3.2.3. Interfaces de Peticiones Web.....	83
2.3.3. MODELO DE CLASES.....	83
2.3.3.1. Definición de clases.....	84
2.3.3.2. Atributos.....	85
2.3.3.3. Métodos.....	85
2.3.3.4. Relaciones.....	85
2.3.3.5: Diagrama de clases.....	86
2.3.3.5.1. Diagrama de clases Portal Web.....	86
2.3.3.5.2. Diagrama de clases de Mediciones del Benchmark.....	86
2.3.4. MODELO DE LA ARQUITECTURA.....	87
2.3.4.1. Diagrama de Paquetes, visión del sistema.....	88
2.3.4.2. Navegabilidad.....	90
2.3.4.2.1. Navegabilidad del Benchmark.....	90
2.3.4.2.2. Navegabilidad de Portal Web.....	91
2.4. IMPLEMENTACIÓN DEL BENCHMARK.....	96
2.4.1. DESCRIPCIÓN DE LAS HERRAMIENTAS PARA LA CONFECCIÓN DEL BENCHMARK.....	96
2.4.1.1. Plataforma de desarrollo Netbeans.....	96
2.4.1.2. Servidor de Aplicaciones JBOSS.....	97
2.4.1.3. Servidor de Aplicaciones GLASSFISH.....	97
2.4.1.4. Base de Datos Oracle.....	97
2.4.2. CONSTRUCCIÓN BASADA EN J2EE.....	97
2.4.2.1. J2EE.....	97
2.4.2.2. Bibliotecas y librerías a utilizar.....	98
2.4.3. PRUEBAS DEL BENCHMARK.....	99
2.5. APLICACIÓN DEL BENCHMARK.....	99
2.5.1. PLANIFICACIÓN DE LAS PRUEBAS DE MEDICIÓN Y AMBIENTE.....	99
2.5.1.1. Medición de los componentes Hardware.....	100
2.5.1.2. Medición de los tiempos de Respuesta.....	100
2.5.2. INSTALACIÓN Y CONFIGURACIÓN DE LAS HERRAMIENTAS.....	102
2.5.2.1. Instalación Del JDK.....	102
2.5.2.2. Instalación del Servidor de Aplicaciones Glassfish.....	103
2.5.2.3. Instalación del Servidor de Aplicaciones JBoss.....	110
2.5.2.4. Configuración de Glassfish.....	119
2.5.2.5. Configuración de JBoss.....	123
2.6. MEDICIÓN DE PARÁMETROS Y GENERACIÓN DE RESULTADOS.....	124

CAPÍTULO 3. ANÁLISIS DE RESULTADOS.....	125
3.1 ANÁLISIS DE LOS DATOS OBTENIDOS POR JBOSS.....	125
3.1.1. ANÁLISIS DE DATOS DEL USO DEL PARÁMETRO CPU POR PARTE DEL SISTEMA OPERATIVO.....	125
3.1.2. ANÁLISIS DE DATOS DEL USO DEL PARÁMETRO CPU POR PARTE DEL SERVIDOR DE APLICACIONES	126
3.1.3. ANÁLISIS DE DATOS DEL USO DEL PARÁMETRO MEMORIA.....	127
3.1.4. ANÁLISIS DE DATOS DEL USO DEL PARÁMETRO RED EN KBYTES TRANSMITIDOS	128
3.1.5. ANÁLISIS DE DATOS DEL USO DEL PARÁMETRO RED EN KBYTES RECIBIDOS	129
3.2 ANÁLISIS DE LOS DATOS OBTENIDOS POR GLASSFISH.....	130
3.2.1. ANÁLISIS DE DATOS DEL USO DEL PARÁMETRO CPU POR PARTE DEL SISTEMA OPERATIVO.....	130
3.2.2. ANÁLISIS DE DATOS DEL USO DEL PARÁMETRO CPU POR PARTE DEL SERVIDOR DE APLICACIONES	131
3.2.3. ANÁLISIS DE DATOS DEL USO DEL PARÁMETRO MEMORIA.....	132
3.2.4. ANÁLISIS DE DATOS DEL USO DEL PARÁMETRO RED EN KBYTES TRANSMITIDOS	133
3.2.5. ANÁLISIS DE DATOS DEL USO DEL PARÁMETRO RED EN KBYTES RECIBIDOS	134
3.3 COMPARACIÓN DE RESULTADOS	135
3.3.1. COMPARACIÓN DEL ÍNDICE CPU	136
3.3.1.1. Análisis de Resultados del % de uso de CPU con respecto al Sistema Operativo	136
3.3.1.2. Análisis de Resultados del % de uso de CPU con respecto al Servidor de Aplicaciones	138
3.3.2. COMPARACIÓN DEL ÍNDICE MEMORIA	140
3.3.3. COMPARACIÓN DEL ÍNDICE RED.....	142
3.3.3.1. Promedio Kbytes Transmitidos Y Recibidos	145
3.3.3.2. Rendimiento Kbytes Transmitidos Y Recibidos	146
3.3.4. TIEMPOS DE RESPUESTA.....	147
CAPÍTULO 4. CONCLUSIONES Y RECOMENDACIONES.....	151
4.1. CONCLUSIONES	151
4.2. RECOMENDACIONES.....	153
BIBLIOGRAFIA	154
ANEXOS	159
ANEXO A. TABLAS Y GRÁFICOS DE LA MEDICIÓN DE LOS RECURSOS	159
ANEXO B. VALORES ESTADÍSTICOS DE MEDICIÓN DE LOS COMPONENTES HARDWARE	198
ANEXO C. CONFIGURACION DE ARCHIVOS XML PARA DESPLEGAR LA APLICACIÓN WEB EN EL SERVIDOR DE APLICACIONES JBOSS	212
ANEXO D. DETALLE DE OBTENCION DE DATOS PARA CADA UNO DE LOS RECURSOS A MEDIR Y TIEMPOS DE RESPUESTA	214