UNIDAD 1 FUNDAMENTOS

OBJETIVOS GENERALES DEL CURSO

El alumno podrá desarrollar aplicaciones con interfaz web de tres o más capas utilizando la tecnología "Oracle Java Enterprise Edition JEE", por medios entornos de desarrollo apropiados (IDE´s).

Elaboro:prof.ª Angélica María Ortiz Gaucín.

- 1.1 Panorama de las aplicaciones con interfaz web, características y retos
- 2.1 Plataforma Oracle Java y sus diferentes ediciones.
- 1.2 Diferentes entornos de desarrollo de aplicaciones empresariales
- 1.3 Java Edición Empresarial(JEE)
- 1.3.1 Arquitectura
- 1.3.1 Cliente
- 1.3.2 Web
- 1.3.3 Negocios
- 1.3.4 Datos
- 1.4 Componentes
- 1.5 Contenedores
- 1.6 Servicios



Contesta las siguientes preguntas de la información que se presenta a continuación.

- ¿Cuáles son las tendencias de desarrollo web?
- ¿Cuáles son las tendencias de empleo en el desarrollo web?
- · Menciona las tendencias en el diseño web
- ¿Que sitios web conoces que apliquen el Diseño Plano & Material?
- · Maneras para un sitio web se muestre mas rápido.
- ¿Cuáles son las ediciones de la plataforma Java, a que se orienta cada una de ellas?
- Cita la evolución del Java estándar

- ¿Que características tiene Java Enterprise Edition?
- Objetivos de Java Enterprise Edition
- ¿Cuáles son los requisitos de las aplicaciones empresariales?
- Menciona la función de cada una de las 4 capas en una arquitectura web.
- Realiza en diagrama multicapa de la arquitectura J2EE
- ¿Cuáles son los componentes de cada una de las capas de la arquitectura J2EE?

- ¿Qué es un componente?
- · Menciona la clasificación de los componentes
- Programación basada en conmponentes vs. Programacion o-o
- Que es un contenedor?
- · Eplica la relación entre un componente y un contenedor

Menciona los tipos de contenedores

Menciona Tipos de servicios



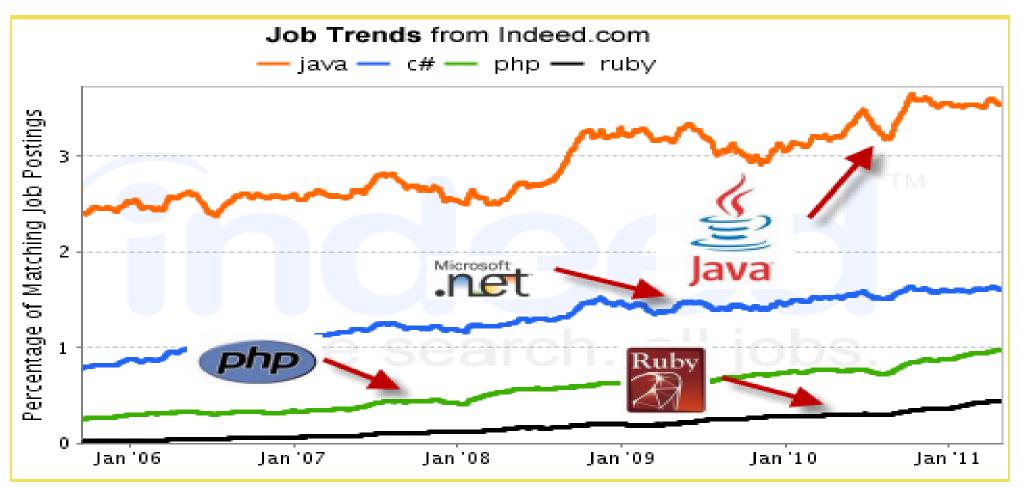
1.1 PANORAMA DE LAS APLICACIONES CON INTERFAZ WEB, CARACTERÍSTICAS Y RETOS

- A principios de la década se hablaba del conjunto de tecnologías agrupadas en la etiqueta HTML5 (HTML5, CSS3 y Javascript) como el futuro de Internet.
- Las tecnologías responsive se convertirán en un estándar para el Diseño y Desarrollo de la Web.

 El entorno de programación y librería Node.js (basado en Javascript) continúa madurando, al facilitar el desarrollo de gran cantidad de aplicaciones en "tiempo real". Las utilidades de esta herramienta son variadas, destacando su uso en servicios web donde sea necesaria una respuesta inmediata: chat, edición en línea, redes sociales y correo electrónico.

• El lanzamiento del nuevo software de diseño web Adobe Brackets cambia el resultado visual de un HTML al instante. Esto utilizando las posibilidades que entrega HTML5, sin tener que cargar la página o guardar el trabajo realizado.

TENDENCIAS DE EMPLEO



T.1 PANORAMA DE LAS APLICACIONES CON INTERFAZ WEB, CARACTERÍSTICAS Y RETOS

 Dado que la gama de dispositivos en los que se visualizan los sitios Web sigue creciendo, los diseñadores Web deben adaptarse constantemente para cumplir con los cambios que se producen en el diseño web.

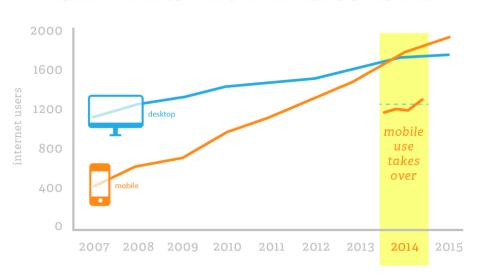
• A la gente le encanta la **sencillez y el minimalismo**. En la actualidad los sitios Web de formas sencillas son los más eficientes.

SI NUESTRO SITIO WEB CUENTA CON UN <u>DISEÑO WEB RESPONSIVE</u>, LAS EXPERIENCIAS DE NAVEGACIÓN A TRAVÉS DE DISPOSITIVOS MÓVILES HARÁN MÁS FÁCIL QUE UN FUTURO CLIENTE SALGA SATISFECHO DE NUESTRO SITIO WEB Y NOSOTROS CONSIGAMOS NUESTROS OBJETIVOS.



2014: YEAR OF THE Mobile Revolution

MOBILE INTERNET USE WILL OVERTAKE DESKTOPS THIS YEAR.



Source: ComScore

Más desplazamiento y menos clicks

Los usuarios prefieren tener toda la información visible en un único lugar, en vez de ir saltando de página en página.

Más imágenes vs. menos texto

A esta tendencia de usar menos texto, más imágenes y más interactividad, será muy importante elegir una tipografía agradable para el usuario.

DISEÑO PLANO & MATERIAL

 Las principales características del diseño material es que es más limpio, predominan animaciones y transiciones de respuesta, y se incorporan efectos de profundidad tales como la iluminación y las sombras.



El **diseño de tarjeta** (card design), se caracteriza por mostrar interfaces limpios y sencillos con Elab**mucha se ersatilidad, exactamente lo que necesita la web.**

EJEMPLO DE DISEÑO PLANO & MATERIAL



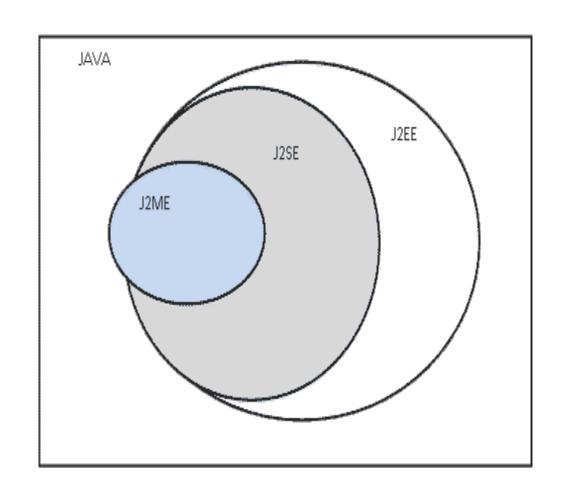
RENDIMIENTO

- A pesar de que la interactividad y las animaciones pueden hacer que un sitio web cargue más lento, hay muchas maneras de ayudar a que este se muestre más rápido:
- Compresión de imágenes y código.
- Plugins de carga en el momento de uso.
- O servidores más veloces, serán una tendencia de diseño web en 2015.

2.1 PLATAFORMA ORACLE JAVA Y SUS DIFERENTES EDICIONES.

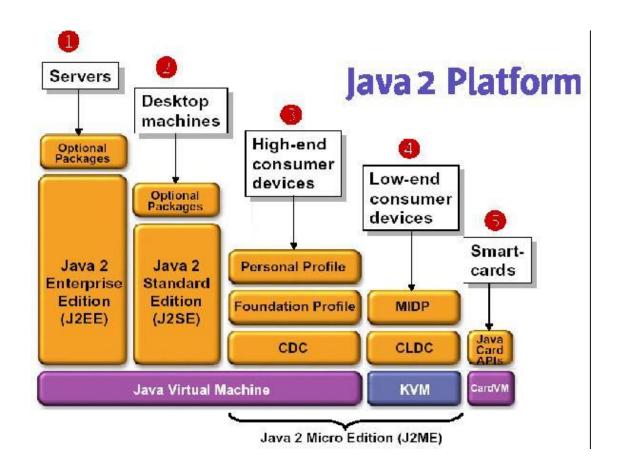
- a) J2SE o simplemente Java SE: Java 2 Standard Edition o Java Standard Edition. Orientado al desarrollo de aplicaciones cliente / servidor. No incluye soporte a tecnologías para internet.
- b) J2EE: Java 2 Enterprise Edition.

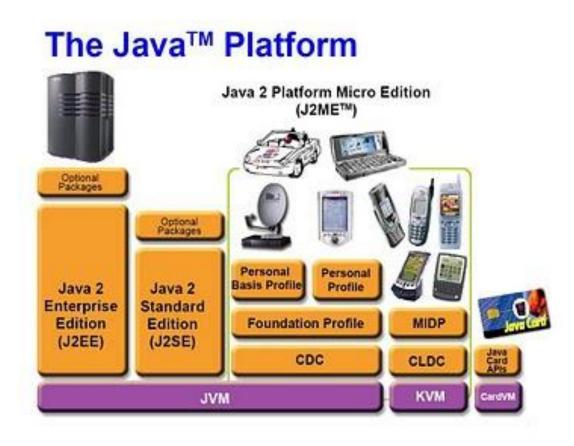
 Orientado a empresas y a la integración entre sistemas. Incluye soporte a tecnologías para internet. Su base es J2SE.
- c) **J2ME**: Java 2 Micro Edition. **Orientado a pequeños dispositivos móviles** (teléfonos, tabletas, etc.).



· Java Card es la versión de Java enfocada a aplicaciones que se ejecutan en tarjetas de crédito con chip. Es una versión muy recortada de Java. Una Java Card es una tarjeta capaz de ejecutar mini-aplicaciones Java. En este tipo de tarjetas el sistema operativo es una pequeña máquina virtual Java (JVM) y en ellas se pueden cargar dinámicamente aplicaciones desarrolladas específicamente para este entorno.







2.T PLATAFORMA ORACLE JAVA Y SUS DIFERENTES EDICIONES.

- Podemos citar en la evolución del Java estándar:
- JDK 1.0 (1996): primer lanzamiento del lenguaje Java.
- · JDK 1.1 (1997): mejora de la versión anterior.
- · **J2SE 1.2** (1998): ésta y las siguientes versiones fueron recogidas bajo la denominación Java 2 y el nombre "J2SE" (Java 2 Platform, Standard Edition), reemplazó a JDK para distinguir la plataforma base de J2EE (Java 2 Platform, Enterprise Edition) y J2ME (Java 2 Platform, Micro Edition). Incluyó distintas mejoras.
- J2SE 1.3 (2000): mejora de la versión anterior.
- Jabor J25E T.4 (2002): mejora de la versión anterior.

- · **J2SE 5.0** (2004): originalmente numerada 1.5, esta notación aún es usada en ocasiones. Mejora de la versión anterior.
- · Java SE 6 (2006): en esta versión, Sun cambió el nombre "J2SE" por Java SE y eliminó el ".0" del número de versión. Mejora de la versión anterior.
- · Java SE 7 (2011): nueva versión que mejora la anterior.
- Java SE 8 (2014): nueva versión que mejora la anterior. Incluye la posibilidad de embeber JavaScript con Java y mejoras en la gestión de fechas y tiempo.
- Java SE 9: nueva versión que mejora la anterior (en difusión).
- · Java SE 10: nueva versión que mejora la anterior (todavía sin uso comercial).

2.3 JAVA EDICIÓN EMPRESARIAL (JEE)

• Es una plataforma que ofrece un modelo de aplicación distribuido y multicapa, formado por componentes reutilizables, con el modelo unificado de seguridad, control transaccional flexible y soporte de servicios web a través del intercambio integrado de información en XML y otros estándares y protocolos abiertos.

2.3 JAVA EDICIÓN EMPRESARIAL (JEE)

 PlataformaJava2EnterpriseEdition(J2EE)(SunMicrosystems)→inde pendencia de plataforma, portabilidad e interoperabilidad entre aplicaciones.

Objetivos básicos:

• Abstracción de tareas críticas y repetitivas mediante servicios con una interfaz uniforme.

 Preparación de una infraestructura uniforme y de una arquitectura de software basada en ella para aplicaciones empresariales:

REQUISITOS DE LAS APLICACIONES EMPRESARIALES

- Almacenamiento y acceso de datos (Back-end integratión):
 Empleo de sistemas de bases de datos (DBMS)
- Mapping de datos y persistencia: Representación de los datos en los programas (Clases) y correspondencia (mapping) con su representación en la base de datos, actualización de la base de datos tras cambios por el programa.
- Consistencia de datos: control de acceso concurrente a los datos, monitores de transacción.

• Interacción con el usuario: autentificación, control de acceso, coordinación de accesos concurrentes.

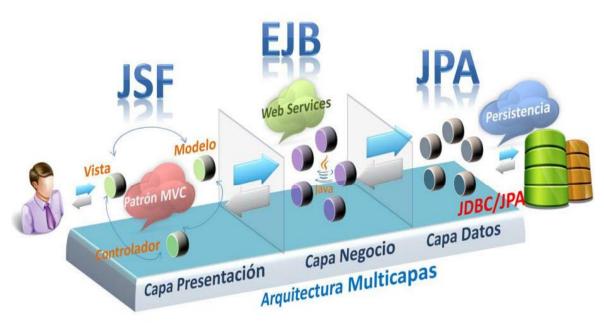
 Acceso a datos comunes: aislamiento de los distintos acceso, cache de datos.

• Performance: tiempo de respuesta, interacción eficiente entre los distintos componentes.

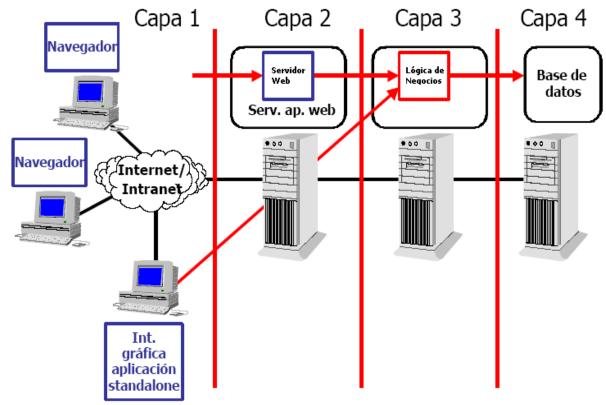
- Escalabilidad: Posibilidad de incorporar nuevos servidores, distribución de carga.
- **Disponibilidad:** seguridad frente a caídas de la aplicación –(ideal disponibilidad 24 x 7), sistemas de tolerancia a fallos, clusterin de servidores y datos.
- Diseño software: mantenibilidad y portabilida → modularidad, Diseño en niveles, reducción de dependencias externas (por ejemplo, en la base de datos).

2.3.1 ARQUITECTURA

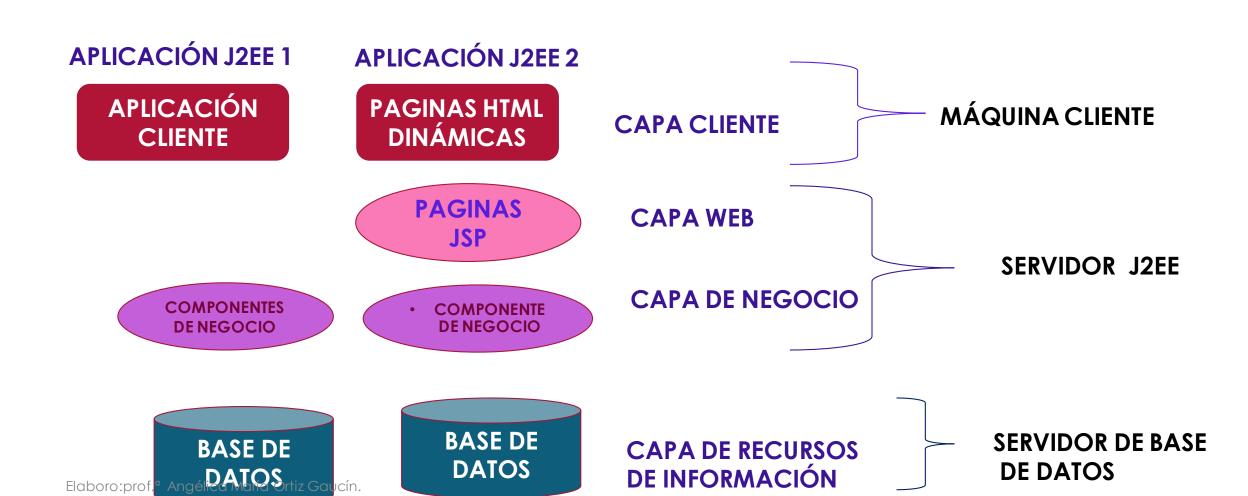
Arquitectura Multicapas



Servidor de Aplicaciones Java



2.3.1 ARQUITECTURA J2EE MULTICAPA



2.3.1 ARQUITECTURA J2EE MULTICAPA

CAPA CLIENTE:

Los componentes de esta capa se ejecutan en la máquina cliente. Se conoce también como capa de presentación. Esta compuesta por los programas que interactúan con el usuario. Estos programas piden al usuario los datos relacionados con la petición que este requiere realizar, envía la petición al componente que la tiene que procesar, el cual valida los datos y devuelve el resultado al cliente para que lo muestre al usuario. La mayoría de las peticiones van dirigidas a componentes de la capa web.

CAPA WEB

Los componentes de esta capa se ejecutan en el servidor J2EE y utilizan el protocolo HTTP para recibir las peticiones y enviar las respuestas a otros componentes. Esta capa proporciona a la aplicación J2EE funcionalidad de Internet, ya que permite proporcionar servicios a la capa cliente.

CAPA DE NEGOCIO

• Los componentes de esta capa se ejecutan en el servidor J2EE. Contiene la lógica de negocio (tareas especificas basadas en reglas que resuelven o cumplen las necesidades de un negocio particular, como la banca, la venta o la financiación) de las aplicaciones J2EE, que puede ser accedida concurrentemente por múltiples clientes de esta capa. Por lo general, un componente de esta capa interactúa con la capa de recursos de información.

CAPA DE RECURSOS DE INFORMACIÓN (MIDDELWARE)

 Los componentes de esta capa se ejecutan en el servidor que administra estos recursos. Conecta la aplicación J2EE con los sistemas y recursos que se encuentren en la red corporativa (sistemas heredados, bases de datos, sistemas de terceros); de esta forma, los desarrolladores de aplicaciones J2EE pueden aprovechar los recursos existentes con los que cuenta la empresa y no están obligados a replicarlos dentro de J2EE.

PRÁCTICAS EN EL DESARROLLO DE ARQUITECTURA

El éxito de una aplicación depende fundamental de los siguientes factores:

- Que esté disponible 24 horas al día, 7 días a la semana, sin ninguna caída.
- Que tenga un tiempo de respuesta aceptable y que sea escalable, esto es, que se pueda ajustar a los aumentos y disminuciones de la demanda.
- Que pueda interactuar con los sistemas existentes.
- Debe ser extensible y mantenible de forma que sea fácil agregar nuevos requerimientos.
- Que utilice componentes existentes.
- Debe ser portable e independiente del proveedor.

COMPLETA LA MISIÓN DE LA CAPA CLIENTE

- El software de la capa cliente normalmente tiene como misión:
- Servir de interfaz gráfica al usuario para interactuar con la aplicación
- Requerir información del usuario y validarla. ...

DE LA CAPA WEB

- La capa web contiene componentes..... Que se comunican de forma directa con los clientes.
- Un componente de la capa Web se identifica a través de un... que se asocia a un enlace que se encuentra en alguna pagina web que se muestra en el cliente.
- La capa web debe controlar el acceso del cliente a los
- La capa web debe evitar peticiones duplicadas enviadas por los..

DE LA CAPA CLIENTE

- El software de la capa cliente normalmente tiene como misión:
- Servir de interfaz gráfica al usuario para interactuar con la aplicación: Esta interfaz normalmente estará basada en un explorador (un applet) o un una aplicación cliente.
- En una aplicación cliente, aunque sea más difícil de escribir, permite un control total sobre los elementos de la interfaz y la gestión de eventos; esto permite un control total sobre los elementos de la interfaz y la gestión de eventos; esto permitirá programar que el servidor sea llamado sólo cuando sea necesario lo que repercutirá favorablemente en el rendimiento final de aplicación.

LA CAPA CLIENTE

• Requerir información del usuario y validarla. Se sugiere realizar la mayor parte de la validación en el servidor ya que validar en el cliente nos obligaría a actualizar las rutinas de validación cada vez que se introduzca una nueva versión del explorador, lo que resulta difícil de mantener; fiarse de las validaciones en el cliente es arriesgado.

LA CAPA WEB

- Contiene componentes (servlets y JSP) que se comunican de forma directa con los clientes. Las aplicaciones que requieren servlets o JSP se env´n a un servidor de aplicaciones. Está técnica proporciona una seguridad adicional, ya que la ejecución de todos los procesos se localiza detrás del servidor de aplicaciones.
- Cuando utilizamos servlets y JSP, una buena programación supone separar el código de presentación y el de procesamiento, colocando el primero en una JSP y el segundo en un servlet.
- Un componente de la capa web se identifica a través de un URL que se asocia a un enlace que se encuentra en alguna página web que se muestra en el cliente.
- La capa web debe controlar el acceso del cliente a los recursos.
- La capa web debe evitar peticiones duplicadas enviadas por los clientes.

LA CAPA EJB

 Permite contruir componentes, que contengan lógica de negocio o que representen datos, reutilizables por distintas aplicaciones cliente y que normalmente estén distribuidos ente distintas máquinas.

2.4 COMPONENTES

 Las aplicaciones J2EE están hechas a base de componentes. Un componente J2EE es una unidad de software independiente que es ensamblado en la aplicación junto con sus ficheros y clases, y que se comunica con otros componentes a través de canales definidos.

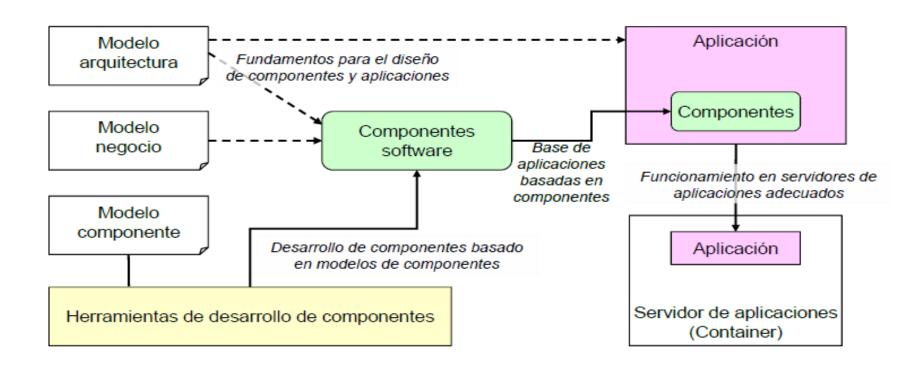
COMPONENTES

- Clientes:
 - clientes web (Suele denominarse ligero)
 - Applets
 - Aplicaciones cliente
- Componentes web:
 - Un Servlets
 - Un JSP
- Componentes de negocio .Es un elemento vital de la arquitectura ya que proporciona concurrencia, escalabilidad, gestión de ciclo de vida y tolerancia a fallos. Los componentes de esta capa son **Enterprise JavaBeans** (EJB).

PROGRAMACIÓN BASADA EN CONMPONENTES VS. PROGRAMACION O-O

- Objetivos comunes: Ocultación de información, abstracción, bajo acoplamiento externo y alta cohesión interna.
- Los objetos son de una granularidad más pequeña que los componentes.
- Los componentes son elementos más asociado a la aplicación

Relación entre componentes, modelos y entornos de ejecución



1.5 CONTENEDORES

• Un contenedor es una interfaz entre un componente y el sistema de inferior nivel que da soporte al componente.

 Mientras que un contenedor se encarga de la persistencia de los datos, la gestión de recurso, la seguridad, los hilos y otros servicios a nivel de sistema para los componentes asociados con él, estos son responsables de implementar la lógica de negocio. Comparativamente, la relación entre componente y un contenedor es muy similar a la relación que existe entre un programa y el sistema operativo; en este caso, el sistema operativo proporciona al programa servicios, por ejemplo, de E/S, y si se instala un nuevo dispositivo de E/S, no es necesario modificar el programa, basta reconfigurar el sistema operativo.

 En Resumen un cliente se refiere a un programa que solicita un servicio de un componente. Un componente es, a su vez, otro programa que realiza una función para proporcionar el servicio; para ello, en algunos casos, se puede necesitar obtener algún recurso. Y un contenedor es el software que gestiona al componente y le proporciona servicios de sistema.

TIPOS DE CONTENEDORES

- **Servidor J2EE**. Es el motor de ejecución de un producto J2EE. Proporciona los contenedores Web y EJB.
- Contenedor Web. Tiene el control sobre la ejecución de las páginas JSP y los servlets en una aplicación J2EE.
- Contenedor EJB. Tiene el control sobre la ejecución de los componentes en terprise beans en una aplicación J2EE.
- Contenedor aplicación cliente. Tiene el control sobre la ejecución de una aplicación cliente.
- Contenedor de applets. Tiene el control sobre la ejecución de los applets.

1.6 SERVICIOS

- Los servlets, las JSP y los JBE son los componentes clave de J2EE. Además de ellos, J2EE incluye otros servicios como:
- Compatibilidad con CORBA. Permite a las aplicaciones Java comunicarse con cualquier sistema empresarial compatible con esta tecnología.
- JavaMail. Correo electrónico. Permite a una aplicación J2EE enviar y recibir mensajes de correo electrónico.
- Java Massege service (JMS). Servicio de mensajería. Permite a los clientes (aplicaciones y componenetes) comunicarse entre si para solicitar y entregar servicios.

- Java Naming and Directrory Interface (JNDI) Interfaz de directorios y nombres de Java. Los objetos se pueden ubicar en distintos servidores. Un servicio de nombres se utiliza para asociar el nombre de un objeto con su ubicación y de esta forma permite una aplicación obtener acceso al objeto mediante su nombre.
- Java Transaction API (JTA). API Java para transacciones. Permite a los desarrolladores incluir código para gestionar las transacciones dentro de los componentes.

 Java DataBase Connectivity (JDBC). Conexión con bases de datos desde java. Permite a una aplicación conectarse con cualquier sistema gestor de base de datos.

 Deploytool. Una aplicación J2EE empaquetada incluye un conjunto de descriptores en formato XML para instalar la aplicación y sus componentes. Para facilitar el proceso de empaquetado.