

5 Aplicaciones empresariales con tecnología java EE.

Esta tesis aborda la creación de una aplicación empresarial, pero, ¿a qué se refiere el término de aplicación empresarial? En esencia, las aplicaciones empresariales son aquellas que permiten a una empresa gestionar sus actividades económicas como planificación de recursos, inventarios y catálogos de productos, preparación de facturas, etc. Todo esto es un reto, y algunos factores que contribuyen a ese reto y su complejidad son los siguientes [15]:

Diversidad de necesidades de información

En una empresa, la información es creada y utilizada por varios usuarios en formas diferentes, dependiendo de necesidades específicas. Es muy común encontrar que cada actividad económica procese la información de un modo diferente.

Complejidad de procesos económicos

La mayoría de procesos económicos de las empresas recaban información compleja, la procesan y la distribuyen. Con mucha frecuencia, se enfrentan a una lógica compleja para recabar y procesar información. Esto tiene como resultado unos requisitos técnicos complejos y requisitos de arquitectura para crear dichas aplicaciones.

Diversidad de aplicaciones

Debido a la compleja naturaleza de los procesos económicos en las empresas, es frecuente encontrar que una empresa desarrolle un gran número de aplicaciones, cada una de ellas creada en distintos momentos para satisfacer las necesidades de distintos procesos económicos que fueron surgiendo. Esto desemboca con frecuencia en la presencia de aplicaciones creadas utilizando diferentes arquitecturas y tecnologías. Uno de los retos a los que se enfrentan las empresas hoy en día es la necesidad de conseguir que dichas aplicaciones se comuniquen entre sí de modo que los procesos económicos puedan llevarse a cabo sin interrupciones.

Estos factores son muy comunes y las empresas invierten grandes sumas de dinero para crear y gestionar aplicaciones que hagan frente a estos retos esperando cumplir los siguientes requisitos [15]:

Productividad de programación

La adopción directa de nuevas tecnologías es insuficiente si estas no son utilizadas adecuadamente aprovechando su potencial al máximo y si no son integradas apropiadamente con otras tecnologías relevantes. De este modo, la habilidad para desarrollar y desplegar las aplicaciones tan rápida y efectiva-

mente como sea posible es muy importante. Conseguir este objetivo puede resultar complicado debido a la gran variedad de tecnologías y normas que requieren capacidades altamente desarrolladas; adquirirlas y mantenerse actualizado, además, es un problema en sí mismo. Además, el cambio tan rápido en las mismas normas dificulta la tarea de asegurar una unión eficiente entre las tecnologías.

Fiabilidad y disponibilidad

En Internet el tiempo de inactividad puede resultar fatal para una empresa. La habilidad para organizar y poner en marcha sus operaciones de base Web, y mantenerlas activas es crucial. Y por si no fuera suficiente, también debe ser capaz de garantizar la fiabilidad de sus transacciones económicas de modo que sean procesadas por completo y con exactitud.

Seguridad

Internet no sólo ha aumentado exponencialmente el número de usuarios potenciales sino también el valor de la información de una compañía, de modo que la seguridad de esta información se ha convertido en una preocupación. Es más, a medida que las tecnologías son más avanzadas, las aplicaciones más sofisticadas y las empresas más complejas, implantar un modelo de seguridad efectivo es cada vez más difícil.

Escalabilidad

La capacidad de una aplicación para crecer y ajustarse a una nueva demanda, tanto en su operación como en el número de usuarios que atiende, es esencial. Esta condición se cumple especialmente cuando consideramos que los usuarios potenciales de una aplicación pueden ser de millones de usuarios individuales, vía Internet. Su capacidad para escalar¹² depende no sólo de la habilidad para manejar un amplio aumento en el número de clientes sino también del uso efectivo de los recursos de un sistema.

Integración

La información se ha convertido en un punto clave para las empresas, mucha de esta información existe como datos en sistemas de información viejos y obsoletos. Para maximizar la utilidad de esta informa-

¹² Escalabilidad es la propiedad deseable de un sistema, una red o un proceso, que indica su habilidad para extender el margen de operaciones sin perder calidad, o bien manejar el crecimiento continuo de trabajo de manera fluida, o bien para estar preparado para hacerse más grande sin perder calidad en los servicios ofrecidos. Wikipedia 2010-10-11, “Escalabilidad”, <http://es.wikipedia.org/wiki/Escalabilidad>.

ción, las aplicaciones deben poder ser integradas en los sistemas de información ya existentes, lo cual no es necesariamente una tarea fácil ya que las tecnologías actuales con frecuencia han avanzado hasta superar algunos de estos sistemas. En este aspecto, la habilidad para combinar las nuevas y las viejas tecnologías es fundamental para el desarrollo de las empresas de hoy en día.

Se puede decir entonces que las empresas están buscando una tecnología y una infraestructura permisiva que simplifique algunos de los temas técnicos complejos. Y no es que estos problemas sean nuevos para los desarrolladores, si no que resolverlos de forma rentable y exhaustiva sigue siendo algo crucial. Se puede decir entonces que lo que se necesita es una plataforma con una infraestructura rica y numerosas posibilidades de arquitectura que al mismo tiempo prevea un rápido entorno de desarrollo.

5.1 Introducción a la Tecnología Java EE.

Java EE es una de las mejores soluciones encontradas hasta el momento para satisfacer las demandas de las empresas de hoy en día. Java EE especifica tanto la infraestructura para gestionar sus aplicaciones, como los servicios API para construir sus aplicaciones.

La plataforma Java EE es esencialmente un entorno distribuido aplicación-servidor, un entorno Java que ofrece:

- Un conjunto de varios API de extensión Java para construir aplicaciones. Estos API definen un modelo de programación para aplicaciones Java EE.
- Una infraestructura de entornos de ejecución para albergar y gestionar aplicaciones.

Las aplicaciones que puede desarrollar con estos dos elementos pueden ser programas para controlar páginas Web o componentes para implementar transacciones complejas de bases de datos, o incluso applets de Java, todos ellos distribuidos por la red.

5.1.1 Arquitectura

La plataforma Java EE está compuesta por cuatro contenedores, un conjunto de entornos de ejecución para gestionar los componentes de la aplicación y destinados a proporcionar acceso a los API de Java EE.

La figura 3 muestra la arquitectura de Java EE en términos de sus contenedores y sus API [14]. Las flechas representan los protocolos usados por un contenedor para acceder a otro.

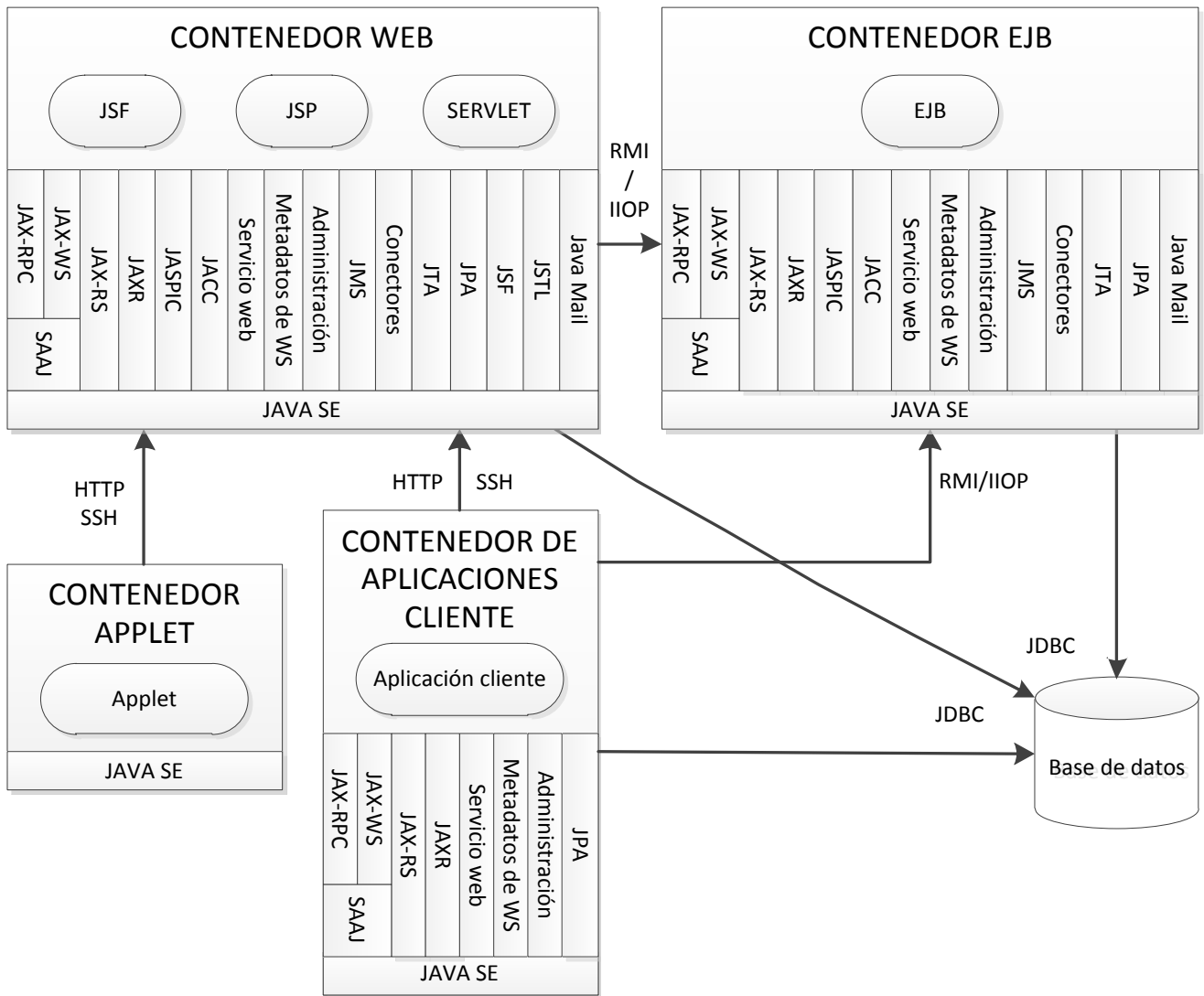


Figura 3 – Arquitectura de Java EE

Esta arquitectura muestra cuatro contenedores [14] [15]:

Un **contenedor Web** para albergar *servlets* Java, filtros, páginas JSP y páginas JSF. Es responsable de instanciar, inicializar e invocar *servlets*, apoyo a los protocolos HTTP y HTTPS y es utilizado para alimentar con páginas web a los navegadores web del cliente.

Un **contenedor EJB** para albergar componentes Enterprise Java Bean¹³. Da soporte a los componentes que implementan la capa de lógica de negocio y ofrece servicios de control de acceso y seguridad, control de transacciones y ejecución concurrente, etc.

Un **contenedor applet** para albergar applets¹⁴ Java.

¹³ Componentes de negocio que se ejecutan en el servidor. Permiten al programador abstraerse de los problemas generales de una aplicación empresarial (conurrencia, transacciones, persistencia, seguridad, etc.) para centrarse en el desarrollo de la lógica de negocio en sí.

¹⁴ Un *applet* es una clase Java de cliente que se ejecuta normalmente en un navegador Web.

Un **contenedor de aplicaciones cliente** para albergar aplicaciones Java estándar. Incluye un conjunto de librerías, clases y otro tipo de archivos necesarios para brindar inyección, seguridad y servicio de nombres a aplicaciones Java SE. Se puede comunicar tanto con el contenedor web como con el conde-nador de EJBs.

Cada uno de los contenedores proporciona un conjunto de entornos de ejecución para los respectivos componentes. Los componentes de Java EE también son denominados objetos gestionados, ya que es-tos objetos son creados y gestionados en el interior del entorno de ejecución del contenedor.

En la figura anterior, los bloques verticales situados en la parte inferior de cada contenedor representa los API de Java EE.

Además del acceso a estos API a nivel infraestructura, cada contenedor también implementa el corres-pondiente API específico a cada contenedor (API Java Servlet para el contenedor Web y el API EJB para el contenedor EJB).

Los rectángulos (*servlets*, paginas JSP y los EJB) de la figura son los programas que se desarrollan y se albergan en estos contenedores. En Java EE, estos programas se conocen como componentes de aplica-ciones.

5.1.2 Componentes de aplicación

En el entorno de ejecución Java EE se definen cuatro tipos de componentes con los que es compatible [14].

Los applets son aplicaciones gráficas que se ejecutan en un navegador web. Estos usan los API Swing para proporcionar potentes interfaces de usuario.

Las aplicaciones son programas que se ejecutan en un cliente. Son típicamente GUIs o programas de procesamiento por lotes que tienen acceso a todas las comodidades que proporciona Java EE nivel medio.

Aplicaciones Web (*servlets*, *servlets* filtro, páginas JSP y JSF) se ejecutan en un contenedor web y res-ponden a las peticiones HTTP de clientes web.

Enterprise Java Beans son componentes controlados por el contenedor para el procesamiento transac-cional de la lógica de negocio.

5.1.3 Los API de Java EE

Las aplicaciones distribuidas requieren acceso a una serie de servicios de empresa. Los servicios típicos incluyen procesamiento de transacciones, acceso a bases de datos, mensajería, multihilos (*multithreading*), etc. La arquitectura de java EE unifica el acceso a estos servicios en un API de servicios de empresa. Sin embargo, en lugar de tener que acceder a estos servicios a través de interfaces de propietario o no estándar, los programas de aplicación en Java EE pueden acceder a estos API mediante el contenedor.

Una plataforma Java EE típica comercial (o servidor de aplicación Java EE) incluye uno o más contenedores y el acceso a los API de empresa viene especificado por la Java EE. Cabe mencionar que servidores de aplicación Java EE no necesitan implementar estos servicios ellos mismos; los contenedores sólo deben proporcionar acceso a cada implementación de servicio a través de un API de Java EE [14] [15] [29].

La especificación de la plataforma Java EE incluye un grupo de extensiones estándar Java que cada plataforma Java EE debe mantener:

Java Transaction API (JTA): Este servicio ofrece una interfaz para la demarcación de transacciones utilizadas por el contenedor y la aplicación. También proporciona una interfaz entre el administrador de transacciones y el administrador de recursos a nivel de la interfaz del proveedor de servicio (SPI, por sus siglas en inglés).

Java Persistence API (JPA): Es una API estándar para el Mapeo Objeto-Relacional (ORM, por sus siglas en inglés) que proporciona la conversión automática de tablas relacionales a objetos en memoria; cabe destacar que es solamente una especificación, no un software utilizable y que requiere una implementación particular. JPA utiliza el Java Persistence Query Language (JPQL) para consultar datos almacenados en la base de datos subyacente, un dialecto de SQL portable utilizado para evitar dependencias de SQL en aplicaciones Java.

Java Message Service (JMS): Permite comunicarse de forma asíncrona a través de mensajes, ya sea punto a punto o Publicador/Suscriptor.

Java Naming and Directory Interface (JNDI): Es una API que provee interfaces estándar para localizar usuarios, máquinas, recursos y objetos en la red a través de servicios de Nombre y Directorio.

Java Mail: Es una API de Java que facilita el envío y recepción de e-mail desde código java.

JavaBeans Activation Framework (JAF): Es una API que proporciona un *framework*¹⁵ de activación utilizado por otros paquetes. El desarrollador puede usar JAF para determinar el tipo de un trozo arbitrario de datos, acceder a él, descubrir las operaciones disponibles e instanciar el Bean apropiado para ejecutar esas operaciones.

Procesamiento XML: La mayoría de los componentes de Java EE pueden desplegarse con ayuda de los descriptores de despliegue XML, para esto, las aplicaciones tienen que manipular a menudo dichos documentos y en este sentido API Java de procesamiento de XML (JAXP) presta apoyo con ayuda de las API SAX¹⁶ y DOM¹⁷, así como para XSLT.

Java Connector Architecture (JCA): Define una arquitectura estándar para conectar la plataforma Java EE con recursos externos y sistemas empresariales de información (EIS, por sus siglas en inglés) heterogéneos. Pudiendo ser desde bases de datos, mainframes o programas de Planificación de Recursos Empresariales (ERP, por sus siglas en inglés).

Web Services: Java EE proporciona soporte para SOAP¹⁸ y servicios web RESTful¹⁹. Java EE incluye la API Java XML Web Services (JAX-WS) en sustitución de la API de Java para XML basado en RPC²⁰ (JAXRPC) que proporciona soporte a servicios Web utilizando el protocolo SOAP / HTTP y la API Java REST Web Services (JAX-RS) que proporciona soporte para servicios web utilizando la arquitectura REST²¹.

Servicios de gestión: Java EE define APIs para la gestión de los contenedores y servidores utilizando EJBs administrados especiales. El Java Management Extensions (JMX) API se utiliza para proporcionar apoyo a la gestión.

¹⁵ Se entiende por framework, “un conjunto estandarizado de conceptos, prácticas y criterios para enfocar un tipo de problemática particular”.

¹⁶ SAX (*Simple API for XML*) es un API de análisis del acceso serial para el XML.

¹⁷ DOM (Modelo de objetos de documento XM) es una representación donde se genera un árbol jerárquico en memoria del documento o información en XML de un documento XML

¹⁸ SOAP (Simple Object Access Protocol) es un protocolo estándar que define cómo dos objetos en diferentes procesos pueden comunicarse por medio de intercambio de datos XML.

¹⁹ Los RESTful Web Services son servicios basados en una técnica de arquitectura software llamada REST.

²⁰ Protocolo que permite a un programa ejecutar código en otra máquina remota sin tener que preocuparse por las comunicaciones entre ambos.

²¹ REST (Representational State Transfer) es un estilo de arquitectura que se utiliza, como modelo en los sistemas Web distribuidos. No es un estándar, sino un enfoque que muestra cómo desarrollar y proporcionar servicios en Internet.

Servicios de seguridad: Java Authentication and Authorization Service (JAAS) permite a los servicios autenticarse y hacer cumplir los controles de acceso a los usuarios.

Servicio de despliegue: La implementación de la especificación Java EE define un contrato entre las herramientas de despliegue y los productos Java EE para estandarizar el despliegue de aplicaciones.

5.1.4 Tecnologías de comunicación

Proporcionan los medios de comunicación para los diversos componentes y servicios de la aplicación Java EE.

5.1.4.1 Protocolos de Internet

Las solicitudes de un cliente y las respuestas de un servidor se comunican a través de tres protocolos principales.

HTTP o Hypertext Transfer Protocol es un protocolo genérico que funciona sobre una base solicitud/respuesta. Un cliente envía una solicitud al servidor, URI (Uniform Resource Identifier) y versión de protocolo, seguido por un mensaje tipo MIME que contiene notificaciones de solicitud, información del cliente y posible contenido de cuerpo en una conexión con un servidor. El servidor a su vez responde con una línea de estado seguida por un mensaje de tipo MIME que contiene información del servidor, meta información de entidad y posible contenido entidad-cuerpo [15].

TCP (TransmissionControl Protocol) sobre IP (Internet Protocol) son realmente dos protocolos distintos, que se combinan normalmente en una única entidad. IP es el protocolo responsable de asegurar que los datos son recibidos en ambos extremos de la comunicación a través de Internet. TCP es el protocolo que registra los paquetes y certifica que son reunidos en el mismo orden en el que fueron divididos y sin errores [30].

SSL (Secure Sockets Layer) utiliza la criptografía para cifrar el flujo de información entre el cliente y el servidor. También proporciona un medio para que ambas partes puedan autenticarse entre sí. *Secure http* (HTTPS) se distingue normalmente del *http* corriente sin cifrar al ser servido en diferente número de puertos [31].

5.1.4.2 Protocolos de objeto remoto

En aplicaciones donde los componentes se distribuyen normalmente en diferentes niveles y servidores, se requieren algunos mecanismos para utilizar los componentes de forma remota, dejando que el cliente ignore que el componente no es local respecto a sí mismo.

Remote Method Invocation (RMI) es uno de los mecanismos primarios en las aplicaciones de objetos distribuidos. Permite utilizar interfaces para definir objetos remotos [32].

RMI-IIOP permite invocar objetos remotos de forma independiente del protocolo subyacente [14].

5.1.5 ¿Java EE para cualquier aplicación?

El uso de Java EE libera al programador de la necesidad de implementar servicios de bajo nivel, tarea que queda para los implementadores de plataformas Java EE (Bea, Oracle, JBoss, etc.). Es estándar, independiente de la plataforma y puede comunicarse sin ningún tipo de problema con otras aplicaciones creadas en otros lenguajes diferentes de Java y viceversa. No obstante, requiere del uso de buenas prácticas y patrones de diseño para obtener resultados aceptables, lo que se traduce en programadores expertos en Java EE, siendo inaccesible a programadores Java. Dicho de otro modo, el desarrollo de una aplicación java EE se convierte en una pendiente empinada durante el desarrollo, pero como se ha visto en el presente capítulo es una buena opción a mediano y largo plazo para aplicaciones medianas y grandes, pensando en términos de usuarios potenciales y escalabilidad.