###### Licenciatura en Ingeniería de Software

###### Taller Programación Visual III

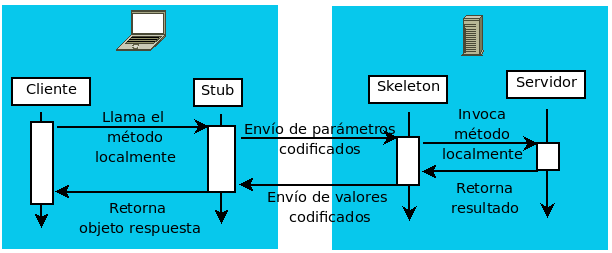
|  |  |
| --- | --- |
| UNIDAD III | Tarea # 9 |
| Nombre: Sergio Herrera Rivera | Fecha: 24/01/2019 |

###### Java RMI (Remote Method Invocation)

RMI es una tecnología desarrollada por Sun para permitir la colaboración de objetos que están localizados remotamente. Esta tecnología se enmarca en la idea de permitir colaboración entre Objetos Remotos. La idea no es que los objetos se comuniquen a través de la programación del usuario de protocolos estándares de red, la idea es tener un objeto cliente, donde podamos completar un requerimiento de datos. El cliente luego prepara el requerimiento que envía a un objeto ubicado en un servidor. El objeto remoto prepara la información requerida (accediendo a bases de datos, otros objetos, etc.). Finalmente el objeto remoto envía la respuesta al cliente. En lo posible esta interacción debería ser lo más semejante posible a requerimientos hechos localmente.\

En principio se puede anhelar la colaboración de objetos escritos en cualquier lenguaje (no es el caso de RMI). Esta idea no es simple de lograr, corresponde al esfuerzo del grupo OMG (Object Management Group, www.omg.org) los cuales propusieron CORBA (Common Object Request Broker Architecture), el cual define un mecanismo común para descubrir servicios e intercambiar datos. CORBA usa Object Request Broker (ORB) como traductores universales para la comunicación entre objetos.

La opción propuesta por Microsoft para comunicar objetos remotos es COM (Component Object Model). Hoy este modelo parece haber sido superado por la tecnología .NET.  
Cuando el cliente y servidor son escritos en Java, la generalidad y complejidad de CORBA no es requerida. En este caso Sun desarrolló RMI, un mecanismo más simple especialmente pensado para comunicación entre aplicaciones Java.



###### Spring MVC

Es una alternativa de framework basado en el patrón modelo-vista-controlador, después de haber aprendido de errores de frameworks anteriores a estos como Jakarta Struts y otras alternativas.

Particularmente, el equipo Spring siente que no hay suficiente separación entre las capas de presentación y la capa de manejo de request, y entre la capa de manejo de request y el modelo.

El framework tiene un conjunto de interfaces que después se implementan para proporcionar la funcionalidad correspondiente. Las interfaces están acopladas claramente al Servlet Api.  
La clase DispatcherServlet está en el front controller y es responsable de delegar y coordinar el control entre varias interfaces en la fase de ejecución durante una petición Http.  
Las interfaces más importantes definidas en Spring Mvc, y sus responsabilidades, son las siguientes:

* HandlerMapping: permite manejar peticiones de entrada.
* HandlerAdapter: ejecución de objetos que permiten manejar las peticiones entrantes.
* Controller: está entre el modelo y la vista, y permite manejar peticiones entrantes y redirigirlas a la respuesta adecuada.
* Vista: responsable de retornar una respuesta al cliente.
* ViewResolver: selecciona una vista basada en un nombre lógico de la vista.
* HandlerInterceptor: intercepta las peticiones entrantes , es comparable pero no igual a los filtros de Servlet.
* LocaleResolver: resuelve y opcionalmente salva el locale de un usuario individual.
* MultipartResolver: facilita trabajar con ficheros de subida wrapping peticiones de entrada.



###### WCF (Windows Communication Foundation)

Windows Communication Foundation (WCF) es un framework para crear aplicaciones orientadas a servicios. Con WCF, puede enviar datos como mensajes asíncronos de un punto final de servicio a otro. Un punto final de servicio puede ser parte de un servicio disponible continuamente alojado por IIS, o puede ser un servicio alojado en una aplicación. Un punto final puede ser un cliente de un servicio que solicita datos de un punto final de servicio. Los mensajes pueden ser tan simples como un solo carácter o palabra enviados como XML, o tan complejos como un flujo de datos binarios. Algunos ejemplos de escenarios incluyen:

* Un servicio seguro para procesar transacciones comerciales.
* Un servicio que suministra datos actuales a otros, como un informe de tráfico u otro servicio de monitoreo.
* Un servicio de chat que permite a dos personas comunicarse o intercambiar datos en tiempo real.
* Una aplicación de panel de control que sondea uno o más servicios en busca de datos y los presenta en una presentación lógica.
* Exponer un flujo de trabajo implementado usando Windows Workflow Foundation como un servicio WCF.
* Una aplicación de Silverlight para sondear un servicio para las últimas fuentes de datos.

Si bien la creación de tales aplicaciones era posible antes de la existencia de WCF, WCF hace que el desarrollo de puntos finales sea más fácil que nunca. En resumen, WCF está diseñado para ofrecer un enfoque manejable para la creación de servicios web y clientes de servicios web.



###### ASP.NET

**E**s un entorno para aplicaciones web desarrollado y comercializado por Microsoft. Es usado por programadores y diseñadores para construir sitios web dinámicos, aplicaciones web y servicios web XML. Apareció en enero de 2002 con la versión 1.0 del .NET Framework, y es la tecnología sucesora de la tecnología Active Server Pages (ASP). ASP.NET está construido sobre el Common Language Runtime, permitiendo a los programadores escribir código ASP.NET usando cualquier lenguaje admitido por el .NET Framework.

El tipo de servidores que emplean este lenguaje son, evidentemente, todos aquellos que funcionan con sistema Windows NT, aunque también se puede utilizar en un PC con windows 98 si instalamos un servidor denominado Personal Web Server. Incluso en sistemas Linux podemos utilizar las ASP si instalamos un componente denominado Chilisoft, aunque parece claro que será mejor trabajar sobre el servidor web para el que está pensado: Internet Information Server.

