

Facultad de Ingeniería – Instituto de Computación

Introducción al middleware

Evaluación escrita - Diciembre 2010

Duración: 3 horas

Preguntas (60 puntos)

1. Explique las similitudes y diferencias entre los tipos de middleware: RPC, ORB-CORBA y ESB.
2. Describa brevemente los conceptos de seguridad a nivel de transporte y seguridad a nivel de mensaje. Mencione dos ejemplos y realice una breve comparación entre ellos, presentando ventajas y desventajas
3. Explique qué problemas intenta resolver la categoría de patrones de Mensajería Endpoints Patterns y comente brevemente dos patrones de la misma
4. La capacidad de ruteo intermediario de los ESBs permite determinar dinámicamente el destino de un mensaje de acuerdo a diferentes factores. Describa al menos dos tipos de ruteo intermediario presentando ejemplos de cómo podrían utilizarse.
5. Describa brevemente qué es la JSR 168, qué define y qué beneficios tuvo su creación. Mencione algunas de las mejoras introducidas en la JSR 286.
6. Describa por lo menos dos sitios públicos que hagan Mashups de contenidos.

Problema (40 puntos)

Una organización necesita construir un proceso de facturación siguiendo una Arquitectura Orientada a Servicios en donde los sistemas participantes no podrán modificarse.

En la organización, se tiene un sistema legado (Cobol) el cual deja diariamente un archivo con información necesaria para la invocación del proceso pero no suficiente. La información faltante para la ejecución del proceso puede obtenerse a partir de un Web Service externo a la organización.

El proceso de facturación es llevado a cabo por una aplicación que se ejecuta asincrónicamente sobre un Application Server Java EE. La forma de interacción es a través de mensajes. Cuando se desea invocar al proceso, se genera un mensaje con la información requerida y se lo envía al sistema. Este mensaje es leído por la aplicación y una vez que el proceso termina impacta el resultado en su Base de Datos. Luego, avisa su culminación con un mensaje de respuesta indicando si el proceso ejecutó exitosamente o no debido a errores. En caso de que el proceso falle, se deberá invocar a un sistema externo cuya responsabilidad es de Logging y generación de alertas. A este sistema se lo puede acceder a través de Sockets TCP.

La aplicación que ejecuta el proceso tiene una interfaz Web de donde se puede ejecutar. Dado que se trata de un proceso asincrónico, una vez iniciado se le avisa al usuario que se le notificará vía email cuando culmine. Claramente, quien recibirá dicho mail será el usuario que lanzó el proceso, el cual luego de recibir el mismo podrá ingresar nuevamente al sistema y ver el resultado.

Se pide:

a.- Proponer una arquitectura para la solución y describir los componentes de software que interactuarán en la misma, indicando responsabilidades y protocolos de comunicación.

b.- Describir en alto nivel el diseño de la interacción entre los componentes.

Notas:

Si en su solución utiliza algún tipo de middleware, estándares, y/o patrones de diseño vistos en el curso, puede usarlos sin detallarlos.