

UNIVERSIDAD EAFIT

INGENIERÍA DE SISTEMAS

St0263: Tópicos Especiales en Telemática, grupo 031, 2012-1, Prof.: Edwin Montoya

Reto 04 – Análisis, Diseño e Implementación de un Chat Distribuido Web / Cliente Grueso

Descripción:

Realizar el modelado, diseño e implementación de un Chat con múltiples salas. Un usuario puede enviar mensajes a todos los miembros de una sala (por defecto) o a un usuario en particular en línea en el sistema. Se debe enviar mensajes sencillos de texto de máximo 160 caracteres (tipo twitter o SMS).

Los siguientes son los requerimientos:

- Implementar un middleware de invocación remota. Utilizar uno orientado a procedimientos, objetos o servicios (web services).
 - Debe implementar un cliente Grueso y un Cliente Web. Y entre ambos clientes poder ejecutar la aplicación.
 - Definir el modelo y la arquitectura.
 - Desde el punto de vista del sistema distribuido y teniendo en cuenta el modelo/arquitectura/middleware a utilizar, considere:
 - Mecanismo de interacción (tipo de sincronización, simetría, estado y sesión).
 - Modelo de manejo de fallos.
 - Modelo de seguridad.
 - Niveles de transparencia.
 - Multiusuario / Concurrencia
 - Consideraciones de escalabilidad, extensibilidad y otros criterios de diseño de la arquitectura de la aplicación.
 - Desplegar la aplicación en el servidor “sistemas.eafit.edu.co”.
1. Realice la implementación en el lenguaje de programación de su preferencia. No se distraiga con la interfaz gráfica.
 2. Realice un documento que contenga los elementos de análisis, diseño e implementación de la aplicación distribuida.

Fecha de entrega:

- **Marzo 25 de 2012** por Entrega de trabajos de EI y en el servidor de producción respectivo.

Criterios de evaluación:

- Informe escrito que contiene el análisis, diseño (global y detallado) además de normas del documento (portada, índice, desarrollo, referencias, etc).
 - En la documentación claramente debe especificar como se instala, que requerimientos de software tiene, como se ejecuta.
 - Desde el punto de vista de análisis y diseño debe apoyarse de los diagramas UML que considere pertinentes.
- Que el reto funcione (corra) por fuera de un editor y que corra bien, con todas las funcionalidades solicitadas.

- Debe correr en el servidor de despliegue asignado para la materia.
 - Se probará ejecutando más de 2 clientes desde diferentes máquinas.
 - Que no tenga parámetros quemados en código (host, ip o puertos).
- Se seleccionará de manera aleatoria 5 grupos, y dentro de ellos, un integrante de manera aleatoria. Dicho estudiante debe responder por toda la sustentación y de acuerdo a la sustentación de dicho estudiante será la nota del grupo.