

Proyecto #3 - PacketTEC

Instituto Tecnológico de Costa Rica
Área de Ingeniería en Computadores
Algoritmos y Estructuras de Datos I (CE 1103)
II Semestre 2017
Valor 15%



Objetivo General

- Implementar una solución a un problema mediante la utilización de estructuras de datos de propósito general

Objetivos Específicos

- Implementar las diferentes estructuras de datos de propósito general
- Implementar algoritmos sobre estructuras de datos de propósito general.
- Desarrollar algoritmos para dar solución a un problema.
- Implementar un mecanismo para manipular las estructuras de datos implementadas.
- Fomentar la creatividad mediante el análisis y diseño de algoritmos
- Utilizar diagramas de clases UML para modelar una solución a un problema.
- Aplicar patrones de diseño en la elaboración de una solución a un problema.

Descripción del Problema

Se desea desarrollar una aplicación que permita gestionar el tráfico de información en una red de dispositivos (Similar a Packet Tracer).

La idea general de este proyecto es crear una red de dispositivos móviles que permitan la distribución de mensajes (texto, archivos, videos, imágenes) entre cualquiera de los nodos, siempre y cuando exista al menos una ruta entre los nodos.

Los componentes requeridos para implementar esta aplicación son:

- **PacketTECApp:** Aplicación mobile por medio de la cual los usuarios pueden enviar y recibir información.
- **PacketTECWebSite:** Aplicación Web por medio de la cual interactúan los administradores de la red.
- **PacketTECServer:** Encargado de gestionar toda la lógica del sistema.

La comunicación en el sistema debe ser utilizando JSON y toda información que deba ser almacenada debe ser utilizando archivos XML.

PacketTECApp

Esta es una aplicación para dispositivos mobile, que utilizan los usuarios para agregar sus dispositivos mobile a la red para enviar y/o recibir información. Puede ser desarrollada en Android o IOS, queda a criterio de cada grupo.

PacketTECApp debe permitir las siguientes funcionalidades:

- **Registrarse en el sistema:** Se debe permitir el registro de nuevos dispositivos en la red. Debe permitir identificar de manera única a cada dispositivo y definir una ruta de almacenamiento en cada dispositivo para guardar los mensajes.

- **Almacenamiento de mensajes:** Cada dispositivo de la red debe almacenar la siguiente información:
 - ◆ Información y adjuntos de los mensajes que ha enviado y llegado satisfactoriamente a su destino (**Árbol Splay**).
 - ◆ Información y adjuntos de los mensajes que ha enviado y no han llegado satisfactoriamente a su destino (**Árbol AVL**).
 - ◆ Lista de mensajes que han pasado por el dispositivo y no es el receptor final (**Árbol Binario de Búsqueda**).
 - ◆ Información y adjuntos de los mensajes que ha recibido, siendo el receptor final (**Árbol B**).
- **Envío de mensajes:** Para enviar mensajes se debe indicar: destinatario, texto (obligatorio), adjuntos (no es obligatorio). El remitente lo determina la aplicación de acuerdo al nombre de la persona con que se registró el dispositivo. Una vez que se envía el mensaje este llega a PacketTECServer y le asigna un ID único y le pasa este ID al dispositivo. Para el destinatario el usuario selecciona alguno de la lista de todos los dispositivos (no sabe con cuales tiene o no comunicación).
- **Recepción de mensajes:** Un dispositivo puede recibir un mensaje en dos casos: es parte de la ruta para entregar el mensaje al destinatario final o es el destinatario (**Patrón cadena de responsabilidad**). En ambos casos el dispositivo debe confirmar la recepción del mensaje a PacketTECServer. Cuando recibe un mensaje donde no es el receptor final debe verificar la ruta que viene dentro del msj para saber a cuál nodo debe enviarlo. Si el nodo al que debe pasarlo no tiene comunicación debe enviar una solicitud a PacketTECServer para que recalcule y le envíe una nueva ruta.
- **Búsqueda de Mensajes enviados:** La aplicación debe permitir buscar un mensaje en el dispositivo. La búsqueda puede ser por tres criterios: algún atributo de la información general del mensaje, nombre del adjunto o tipo de adjunto. Para esto se debe buscar sobre las 4 estructuras de almacenamiento del dispositivo y debe desplegar:
 - ◆ **Resumen de la búsqueda** indicando para cada estructura si encontro algún mensaje que coincida y la cantidad de iteraciones que tuvo que realizar sobre la estructura.
 - ◆ **Desplegar todos los mensajes** que coincidan con la búsqueda, ordenados por el ID de cada mensaje. Al dar click sobre alguno muestra el detalle de toda la información con respecto al mensaje.

PacketTECServer (RESTful)

Este es el encargado de gestionar la red de PacketTEC, siendo el responsable del manejo de toda la lógica del sistema, dentro de sus responsabilidades se encuentran:

- **Gestionar Red de Dispositivos:** Debe permitir almacenar la información general de cada dispositivo (Nombre, ruta de almacenamiento, etc). Es el encargado de gestionar la red, de manera que cualquier cambio en la red debe ser comunicado a todos los dispositivos. Cada vez que se agrega un dispositivo o se elimina alguno se debe notificar el nuevo mapa de la red a todos los dispositivos (**Patrón Observer**).
Considerando la poca disponibilidad de dispositivos para poblar la red se debe permitir crear dispositivos bots.
Cada vez que se agrega un dispositivo se le asigna de manera aleatoria la relación con los otros dispositivos y el costo de cada enlace.
- **Gestionar rutas para entrega de paquetes:** Recibe la solicitud de envío de algún mensaje y

debe determinar mediante algún algoritmo visto en clase si existe comunicación entre el remitente y el destinatario utilizando algún algoritmo visto en clase para este objetivo.

Debe permitir establecer la ruta para la entrega de cada mensaje utilizando algún algoritmo visto en clases para este objetivo.

Adicionalmente, en cualquier momento se pueden presentar cierres en las rutas que comunican los dispositivos, por lo que puede recibir solicitudes para recalcular rutas.

- **Visualizador de la Red:** Debe permitir mostrar en tiempo real la red de Dispositivos, de manera que pueda visualizarse el desplazamientos de cada uno de los mensajes.

PacketTECWebSite

Esta es la aplicación mediante la cual se gestiona la red y solo pueden acceder usuarios administradores (Autenticación con Facebook o Twitter):

- **Visualizador de la Red:** Debe permitir mostrar en tiempo real la red de Dispositivos, de manera que pueda visualizarse el desplazamientos de cada uno de los mensajes. También deberían aplicarse cierres autogenerados por el sistema de forma aleatoria.
- **Cierres de ruta:** Los administradores pueden cerrar caminos entre dispositivos y también se deben generar cierres automáticos y aleatorios por parte del sistema.
- **Almacenamiento de mensajes:** Almacena todos los mensajes que han sido enviados en la red en una sola estructura.
- **Búsqueda de Mensajes enviados:** La aplicación debe permitir buscar un mensaje. La búsqueda puede ser por tres criterios: algún atributo de la información general del mensaje, nombre del adjunto o tipo de adjunto. Los mensajes que coincidan con el criterio de búsqueda deben desplegarse de manera ordenada aplicando alguno de los siguientes algoritmos de ordenamiento:
 - ◆ Selection Sort
 - ◆ Bubble Sort
 - ◆ Insertion Sort
 - ◆ Shell Sort
 - ◆ Merge Sort
 - ◆ Quick Sort
 - ◆ Radix Sort

Aspectos Generales

- Toda la lógica del sistema debe ser realizada por PacketTECServer, mientras que PacketTECApp y PacketTECWebSite solamente se usan como interfaz de despliegue de información y ejecución de acciones.

Documentación requerida

1. Se deberá documentar el código fuente utilizando JavaDoc.
2. Se deberá entregar un documento que contenga:
 - a. Todas las partes estándar: Portada, índice, introducción, conclusión, bibliografía y anexos.
 - b. El cuerpo del documento debe contener:
 - Breve descripción del problema
 - Planificación y administración del proyecto
 - ◆ Lista de features e historias de usuario identificados de la

especificación

- ◆ Distribución de historias de usuario por criticalidad y secuencia de uso
- ◆ Minimal system span
- ◆ Plan de iteraciones que agrupen cada bloque de historias de usuario de forma que se vea un desarrollo incremental
- ◆ Asignación de user stories por miembro del equipo
- ◆ Descomposición de cada user story en tareas.
- ◆ Bitácora de trabajo que muestre tiempo invertido para cada actividad y la fecha en que se realizó

→ Diseño

- ◆ Diagrama de arquitectura de la solución
- ◆ Diagrama de componentes
- ◆ Diagrama de secuencia (Seleccionar al menos 4 Historias de Usuario)
- ◆ Diagrama de clases

→ Implementación

- ◆ Descripción de las bibliotecas utilizadas
- ◆ Descripción de las estructuras de datos desarrolladas.
- ◆ Descripción detallada de los algoritmos desarrollados.
- ◆ Problemas encontrados: En esta sección se detalla cualquier problema que no se ha podido solucionar en el trabajo. Incluye descripción detallada, intentos de solución sin éxito, solución encontradas con su descripción detallada, recomendaciones, conclusiones y bibliografía consultada para este problema específico.

Aspectos operativos y evaluación:

1. Fecha de entrega de acuerdo al cronograma definido al inicio del semestre.
2. El proyecto tiene un valor de 15% de la nota del curso
3. Se puede realizar en grupos de 4 o 5 personas
4. Es obligatorio utilizar un manejador de versiones del código. Puede ser **git o svn**. Se revisará que cada commit lleve comentarios relevantes relacionados con alguna tarea identificada en la sección de planificación
5. Los proyectos que no cumplan con los siguientes requisitos **no serán revisados**:
 - a. Toda la solución debe estar integrada
 - b. La interfaz de usuario debe estar implementada e integrada
6. El código tendrá un valor total de 75%, la documentación 15% y la defensa 10%.
7. De las notas mencionadas en el punto anterior se calculará la Nota Final del Proyecto.
8. Se debe enviar el código (preliminar) y la documentación a más tardar a las 23:59 del día de la fecha de entrega a el email del profesor. El formato del email del subject del email es **CE1103 - Proyecto III**, en el body del email se debe ingresar los carnet y nombres de cada uno de los estudiantes, si el proyecto es en grupos se debe copiar en el email a todos los miembros del grupo.
9. Las citas de revisión oficiales serán determinadas por el profesor durante las lecciones o mediante algún medio electrónico.
10. Aún cuando el código y la documentación tienen sus notas por separado, se aplican las siguientes restricciones

- a. Si no se entrega documentación, automáticamente se obtiene una nota de 0.
 - b. Si no se utiliza un manejador de código se obtiene una nota de 0.
 - c. Si el código y la documentación no se entregan en la fecha indicada se obtiene una nota de 0.
 - d. Si el código no compila se obtendrá una nota de 0, por lo cual se recomienda realizar la defensa con un código funcional.
 - e. El código debe ser desarrollado en Java, en caso contrario se obtendrá una nota de 0.
 - f. Si no se siguen las reglas del formato de email se obtendrá una nota de 0.
11. La revisión de la documentación será realizada por parte del profesor, no durante la defensa del proyecto. El único requerimiento que se consultará durante la defensa del proyecto es el diagrama de clases, documentación interna y la documentación en el manejador de código.
 12. Cada grupo tendrá como máximo 20 minutos para exponer su trabajo al profesor y realizar la defensa de éste, es responsabilidad de los estudiantes mostrar todo el trabajo realizado, por lo cual se recomienda tener todo listo antes de ingresar a la defensa.
 13. Cada excepción o error que salga durante la ejecución del proyecto y que se considere debió haber sido contemplada durante el desarrollo del proyecto, se castigará con 2 puntos de la nota final del proyecto.
 14. Cada grupo es responsable de llevar los equipos requeridos para la revisión, si no cuentan con estos deberán avisar al menos 2 días antes de la revisión a el profesor para coordinar el préstamo de estos.
 15. Durante la revisión únicamente podrán participar los miembros del grupo, asistentes, otros profesores y el coordinador del área.