
PROYECTO 1 – FASE 1 – UDrawing Paper

202000119 – Sergie Daniel Arizandieta Yol

Resumen

El proyecto se enfoca en la gestión de Tipo de Dato Abstracto (TDA) para el uso de memoria dinámica mediante listas y sus derivados en Java, siendo así cada lista siendo aplicada a distintos propósitos.

Donde se aplican para simular la gestión de los clientes, impresoras, ventanillas, imágenes y registros de la empresa, inicialmente usando Json para la carga de los primeros clientes, al mismo tiempo mostrando el estado del sistema de manera insidiar de forma gráfica mediante Graphviz.

Dicho proyecto hace utilización de estas tecnologías de una manera novedosa para la búsqueda, lecturas y escrituras optimas de los datos a gestionar para la UDrawing Paper requerido en el 2022 (Se recalca que el proyecto no es una fase final).

Palabras clave

TDA, JAVA, JSON, Graphviz, Listas

Abstract

The project focuses on the management of Abstract Data Type (ADT) for the use of dynamic memory through lists and their derivatives in Java, thus each list being applied to different purposes.

Where they are applied to simulate the management of clients, printers, counters, images and company records, initially using Json to load the first clients, at the same time showing the status of the system graphically through Graphviz.

This project makes use of these technologies in a novel way for the optimal search, reading and writing of the data to be managed for the UDrawing Paper required in 2022 (It is emphasized that the project is not a final phase).

Keywords

TDA, JAVA, JSON, Graphviz, Lists

Introducción

En el ámbito cotidiano a nivel laboral y empresarial es indispensable una buena gestión de memoria, por lo cual se recurre a la memoria dinámica que cambian constantemente el tamaño de esta, para ser menos cargados en los softwares utilizados por diferentes equipos al tener que procesar demasiados registros por lo cual se recurre a las TDA's, almacenado cada registro en sus tuplas así guardando toda la información.

Al mismo tiempo su uso e implementación suele ser confuso, por lo cual también el manejo de estas es de alta importancia, una de las implementaciones de TDA's.

Estas aparte de almacenar a la información es la que presenta de dichos datos para poder generar una salida mediante archivos de extensión JPG para presentación de la estructura y DOT para los grafos.

Desarrollo del tema

Las tecnologías requeridas y fundamentales para el desarrollo del software son:

- ❖ Implementación de TDA
 - Listas
 - Manipulación de las tuplas de las TDA
- ❖ Archivos Json
 - Definición de los archivos Json
- ❖ Generación de grafos mediante Graphviz

Conceptos generales

- Tipo de dato abstracto (TDA)

Son el nivel más alto de abstracción y son independientes del lenguaje de programación, básicamente son una colección de valores y de operaciones que se definen mediante una especificación en el presente proyecto implementado en listas dinámicas.

- TDA Lista:

Se define como una tupla de n elementos los cuales son asignados dinámicamente como se requiera, donde se pueden crear operaciones para cada una de las tuplas como:

- Obtener datos
- Mostrar
- Eliminar
- Agregar
- Otros

- Manipulación de las tuplas de una lista TDA

Es muy similar a la POO (Programación Orientada Objetos), ya que cada tupla cuenta con atributos específicos, los cuales se pueden hacer referencia gracias a que son definidos en su constructor el cual se encarga de definir los atributos de las tuplas cuando llamadas para crear una instancia de estas.

- Archivos JSON

JSON sus siglas en ingles son “JavaScript Object Notation” el cual traducido seria “Notación de Objetos de JavaScript”. Se utiliza para estructurar datos en forma de texto y permite el intercambio

de información entre aplicaciones de manera sencilla, liviana y rápida. Es una alternativa al XML.

- **Grafos en Graphviz:**

Graphviz es un software de visualizador de gráficos de código abierto, el cual implementa su extensión DOT para sus grafos.

- **DOT**

Es un lenguaje descriptivo de texto plano el cual es una forma simple de describir los grafos.

Representación de los módulos del programa

Tabla I.

Módulos del proyecto.

<i>Clases</i>
Main
General
ReadMasiveData
Client
Circular_Doble_espera
Cola_Print
Cola_Reception
Pila_Images
Simple_Clients_Served
Simple_Windows

Descripción de módulos:

- **Main**
Se encarga solo de la instancia de la clase de general para comenzar todo el gestionamiento.

- **General**

Se encarga de desplegar las funcionalidades para el usuario, así como los requisitos para iniciar el sistema.

- **ReadMasiveData**

Se encarga de leer el Json requerido para el sistema y almacenarlo, así mismo como la cantidad de ventanas disponibles y se implementan.

- **Client**

Se encarga de la creación de objetos para todos los clientes que ingresen al sistema,

- **Circular_Doble_espera**

Es una lista circular doblemente enlazada donde se almacenan los clientes que están esperando sus imágenes.

- **Cola_Print**

Es una cola de las impresoras de color y blanco y negro, en las cuales se gestiona dicha impresión para entregar al respectivo cliente.

- **Cola_Reception**

Es la cola de los clientes en espera para entrar al sistema, el cual inicia en las ventanillas.

- **Pila_Images**

Es una pila donde las ventanillas almacenan temporalmente las imágenes de los clientes para enviarlas a las impresoras.

- **Simple_Clients_Served**

Es una lista simple donde se almacenan los datos de los clientes que ya fueron atendidos y salieron del sistema.

- **Simple_Windows**

Es lista simple donde están en disposición las ventanillas para atender a los clientes en cola.

dinámica a grandes rasgos es fundamental, la utilización y operaciones de las TDA son flexibles a como lo pueda lograr el desarrollador del software.

El presente software solo es una de miles de tipos de implementación que uno requiera, siendo así, ¿cómo sete ocurriría a ti implementar memoria dinámica para la optimización de procesos muy rígidos que conozcas?

Referencias bibliográficas

Arias Guerra, D. (2008) Estructura de datos Avanzadas (Revisado, ed., Vol. 9). Universidad de las Ciencias Informáticas. <https://cutt.ly/eWdkzvt>

Ellson, J. (2005) Graphviz. Graphviz org. <https://graphviz.org/doc/info/lang.html>

Ferris Castell, R. (2004) Algoritmos y Estructura de Datos I (Revisado ed., Vol. 1) Universidad de Valencia http://informatica.uv.es/iiguia/AED/oldwww/2001_02/Teoria/Tema_10.pdf

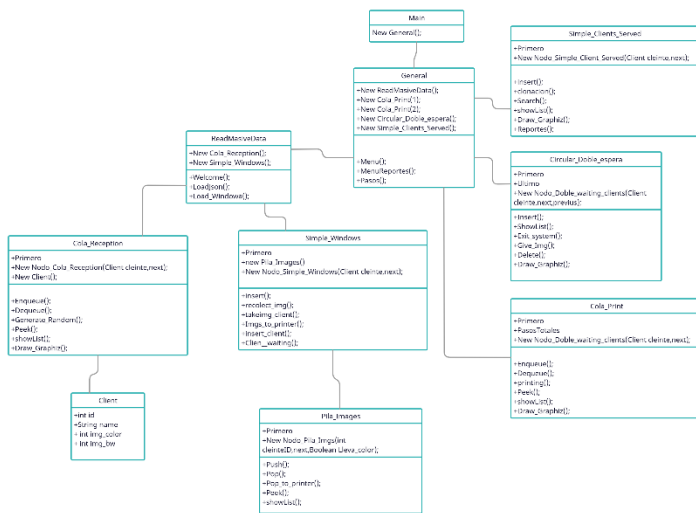


Figura 1. Diagrama de Clases.

Fuente: elaboración propia.

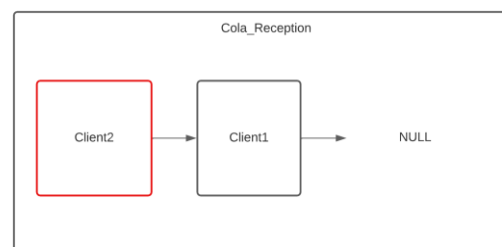
Conclusiones

El uso de nuevas tecnologías, implementaciones inclusive librerías que puedan hacer el trabajo de desarrollo es vital para una empresa viéndolo a gran escala, como son las TDA lista, ya que la mayoría de las veces el cliente siempre requerirá que se tenga el menor estrés en sus computadoras al ejecutar los softwares necesarios.

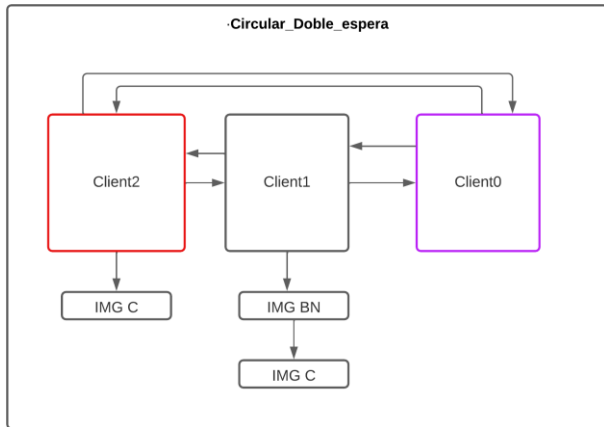
Además, no siempre podemos dar por hecho que se tendrá datos estáticos, por lo cual el uso de memoria

Anexos

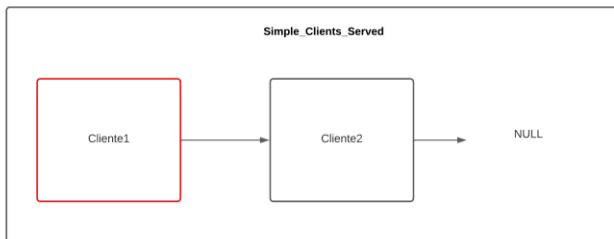
Modelo “Cola_Reception”



Modelo “Circular_Doble_espera”



Modelo “Simple_Clients_Served”



Modelo “Simple_Windows”

