

---

## PROYECTO 2 - SISTEMA DE ATENCIÓN A CLIENTES

---

202109754 – Aldo Saúl Vásquez Moreira

### Resumen

En el presente ensayo se hará mención de diversos procesos que fueron necesarios para desarrollar el programa prototipo que la empresa “Soluciones Guatemaltecas, S.A.” solicitó para mejorar la experiencia de usuario y de las empresas. Básicamente, la solución consiste en una aplicación móvil que los clientes de las diversas empresas registradas en el sistema, podrán escoger una empresa, seguidamente, podrán escoger el punto de atención de dicha empresa para proceder a gestionar las diversas transacciones que este requiera. En cuanto al beneficio que las empresas tendrán gracias a esta solución, es que podrán tener un control más amplio de cada uno de sus puntos de atención, así como de los tiempos de espera y atención debido a que tendrán la posibilidad de activar y desactivar escritorios. Es importante resaltar, que únicamente se desarrolló el programa prototipo para analizar el funcionamiento y decidir si la aplicación final realmente brindará todos los beneficios que se mencionaron anteriormente.

### Palabras clave

Punto de atención: local perteneciente a una empresa.

Transacción: Trato o convenio por el cual dos partes llegan a un acuerdo comercial.

### Abstract

*This essay will mention several processes that were necessary to develop the prototype program that the company "Soluciones Guatemaltecas, S.A." requested to improve the user and business experience. Basically, the solution consists of a mobile application that the clients of the different companies registered in the system will be able to choose a company, then, they will be able to choose the point of attention of that company to proceed to manage the different transactions that it requires. As for the benefit that companies will have thanks to this solution, is that they will be able to have a better control of each of their service points, as well as the waiting and service times due to the possibility of activating and deactivating desks of service. It is important to mention that only the prototype program was developed to analyze the operation and decide if the final application will really provide all the benefits mentioned before.*

### Keywords

*Point of service: Service point belonging to a company.*

*Transaction: Deal or agreement by which two parties reach a commercial agreement.*

## Introducción

En el presente ensayo se hará mención de los detalles más importantes y relevantes en el desarrollo del programa prototipo desarrollado para la empresa nacional “Soluciones Guatemaltecas, S.A.”. El objetivo principal de dicha empresa era construir un sistema de atención a clientes diseñado para cualquier organización que necesite brindar servicios presenciales a sus clientes. Por lo tanto, esta empresa deseaba innovar la experiencia del cliente con la solución propuesta sobre la cual se hará mención posteriormente.

## Desarrollo del tema

Para el desarrollo de este proyecto se utilizaron documentos con formato XML para el ingreso y salida de información del sistema haciendo uso de ElementTree. Sabiendo que XML es un formato de datos inherentemente jerárquico, y la forma más natural de representarlo es con un árbol. ElementTree tiene dos características para este propósito: ElementTree el cual representa todo el documento XML como un árbol y Element el cual representa un solo nodo en este árbol. Las interacciones con todo el documento (leer y escribir en/ desde archivos) se realizan en el nivel ElementTree. Ahora bien, las interacciones con un solo elemento XML y sus subelementos se realizan por medio del nivel Element.

Era importante hacer mención sobre los archivos XML puesto que los archivos de este tipo fueron utilizados como medio principal para el ingreso de datos al sistema. Se manejaron dos archivos, el primero se encargaba de configurar el sistema de atención al cliente. Es decir, era el medio por el cual

las empresas que brindan atención presencial a sus clientes se suscriben al servicio de atención al cliente de “Soluciones Guatemaltecas, S.A.”. Este cargaba los puntos de atención (ubicación de los lugares donde la empresa atiende presencialmente a sus clientes), los escritorios de servicio (cada punto de atención debe definir uno o más escritorios de servicio, estos representan los lugares físicos dentro del punto de atención donde el personal de la empresa atiende a los clientes) y las transacciones (es el tipo de operación o servicio que se puede dar a los clientes en los puntos de atención de la empresa.

El segundo archivo XML se encarga de inicializar la prueba del sistema de atención a clientes. Básicamente, este se encarga de activar de forma masiva escritorios de servicio de un punto de atención específico. Asimismo, se encarga de asignar clientes a cada uno de los puntos de atención y las respectivas transacciones de cada de ellos.

Con esta información clara y luego de tomar los datos necesarios del archivo con formato XML se almacenó la información. Para esto se hizo uso de listas enlazadas ya que una consideración importante es que el software haga un uso óptimo de los recursos que el equipo ofrece.

Dando un breve concepto de las listas enlazadas, estas son unas de las estructuras de datos fundamentales la cual puede ser usada para implementar otras estructuras de datos. Estas consisten en una secuencia de nodos, en los que se guardan campos de datos arbitrarios y una o dos referencias, enlaces o punteros al nodo anterior y posterior. En este caso en

específico únicamente se utilizaron punteros hacia el nodo siguiente ya que el caso se podía manejar perfectamente de esa manera y era ideal para optimizar el rendimiento del equipo.

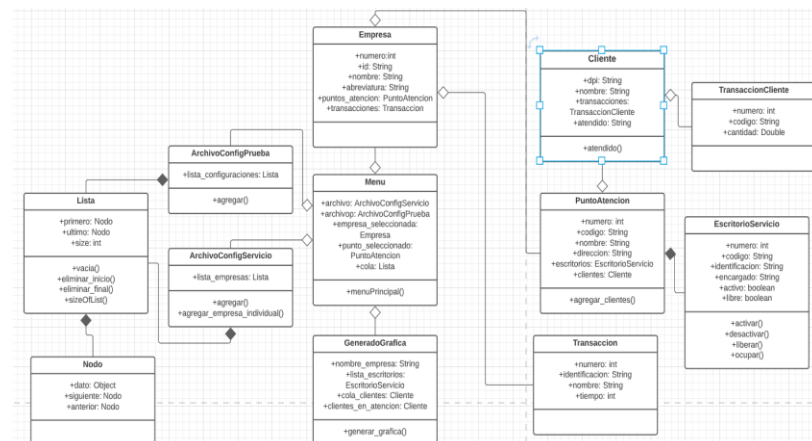
Además, el principal beneficio de las listas enlazadas respecto a los vectores convencionales es que el orden de los elementos enlazados puede ser diferente al orden de almacenamiento en la memoria o el disco, permitiendo que el orden de recorrido de la lista sea diferente al de almacenamiento. Es decir, que a pesar de que los elementos enlazados se encuentren uno a otro en la lista, esto no quiere decir que se almacenen contiguamente en la memoria física de la computadora, sino que, cada puntero hace referencia a una posición en memoria específica para recuperar la información correspondiente al nodo que se desea.

También es importante destacar que una lista enlazada es un tipo de dato autor referenciado porque contienen un puntero o enlace a otro dato del mismo tipo. Las listas enlazadas permiten inserciones y eliminación de nodos de cualquier punto de la lista en tiempo constante, pero no permiten un acceso aleatorio. Existen diferentes tipos de listas enlazadas: listas enlazadas simples, listas doblemente enlazadas, listas enlazadas circulares y listas enlazadas doblemente circulares.

Estas listas pueden ser implementadas en muchos lenguajes. Aunque también es importante recalcar que muchos lenguajes ya implementan este tipo de listas, por lo tanto, no es necesario volver a diseñarlas.

Era importante comprender cómo funcionan dichas listas ya que estas fueron utilizadas para el almacenamiento de las empresas junto a los puntos de atención, escritorios y clientes respectivos. Además, que fueron de total utilidad para la manipulación de estos archivos y así, obtener el funcionamiento esperado de la solución.

Ahora que quedó claro el concepto de qué es una lista enlazada y sabiendo que la información recibida de los archivos con formato XML se almacenó de esta forma se mostrará a continuación la forma en que se diseñaron las clases del programa:



Tal como se muestra en el diagrama de clases utilizado, podemos observar que para darle solución al problema fue necesario implementar seis clases. A continuación, se explicará la función que cumple dentro del sistema cada una de estas.

1. Clase Nodo: Esta se encarga de almacenar la información de cualquier dato que se ingrese en una lista. Cuenta con los dos punteros, uno para el nodo anterior y otro para el nodo siguiente.

2. Clase Lista: Esta se encarga de enlazar cada uno de los nodos que se vayan creando. Además, permite eliminar nodos, así como recorrerse para mostrar cada uno de los nodos que la conforman.

3. Clase ArchivoConfigServicio: Se encarga de configurar el sistema con la información recibida en el primer documento XML.

4. Clase ArchivoConfigPrueba: Se encarga de configurar cada una de las empresas y sus respectivos puntos de atención por medio de la información contenida en el segundo archivo XML.

5. Clase Cliente: Hace referencia a un cliente que puede ser cliente de cualquier empresa en el sistema.

6. Clase Empresa: Hace referencia a cada una de las entidades registradas en el sistema.

7. Clase EscritorioServicio: Hace referencia a cada uno de los escritorios de servicio pertenecientes a un punto de atención.

8. Clase GeneradorGrafica: Se encarga de generar el archivo con extensión “.dot” haciendo uso de la librería de Python “Graphviz” para simular el funcionamiento de cada uno de los puntos de atención de las respectivas empresas.

9. Clase Menu: Se encarga de brindar una buena experiencia de usuario mostrando de manera

ordenada y clara cada una de las opciones que el sistema ofrece.

10. Clase PuntoAtencion: Hace referencia a cada uno de los puntos de atención que una empresa determinada pueda contener.

11. Clase Transaccion: Hace referencia a cada una de las transacciones que se puedan realizar en una empresa determinada.

12. Clase TransaccionCliente: Hace referencia a las transacciones que desee hacer un cliente en específico.

### **Funciones o métodos principales**

1. método agregar en clase ArchivoConfigServicio: Este método se encarga de iniciar cada una de las empresas por medio de listas enlazadas.

2. método agregar\_empresa\_individual en clase ArchivoConfigServicio: Este método se encarga de crear una sola empresa.

3. método agregar en clase ArchivoConfigPrueba: Este método se encarga de agregar la información correspondiente a cada una de las empresas.

4. método activar\_escritorio en clase EscritorioServicio: Se encarga de iniciar el funcionamiento de un escritorio en un determinado punto de atención.

5. método `desactivar_escritorio` en clase `EscritorioServicio`: Se encarga de finalizar el funcionamiento de un escritorio en un determinado punto de atención.

6. método `liberar_escritorio` en clase `EscritorioServicio`: Se encarga indicar que se un determinado escritorio está atendiendo a un cliente en ese momento.

7. método `ocupar_escritorio` en clase `EscritorioServicio`: Se encarga indicar que se un determinado escritorio ha terminado de atender a un cliente.

## Conclusiones

Finalmente, se puede decir que la implementación de todas estas técnicas de programación y estructuras de datos proporcionan una solución adecuada para el problema presentado. Así, como la implementación de la librería `Graphviz` es una excelente herramienta para representar gráficamente cualquier tipo de estructura.

## Referencias bibliográficas

Dimeisons, N. (2014, 29 julio). *Listas Enlazadas*. Monografias.com. Recuperado 5 de septiembre de 2022, de <https://www.monografias.com/trabajos101/las-istas-enlazadas/las-istas-enlazadas>

Fuentes, J. (2021, 8 septiembre). *¿Qué son las estructuras de datos y por qué son tan útiles?* OpenWebinars.net. Recuperado 5 de septiembre de

2022, de <https://openwebinars.net/blog/que-son-las-estructuras-de-datos-y-por-que-son-tan-utiles/>

*xml.etree.ElementTree* — La API XML de *ElementTree* — documentación de Python - 3.10.6. (s. f.). Recuperado 5 de septiembre de 2022, de <https://docs.python.org/es/3/library/xml.etree.elementtree.html>