

PROYECTO 2

SISTEMA DE ATENCIÓN SOLUCIONES GUATEMALTECAS, S.A

Carnet: 202101149 – Mariano Roberto Rac Noguera

Resumen

La empresa de Soluciones Guatemaltecas S.A ha consultado para el desarrollo de una aplicación para llevar el control de atención presencial a sus clientes en sus distintos puntos de atención y distintas configuraciones.

Para el desarrollo del software se utilizó principalmente la estructura de datos de listas enlazadas y su clase nodo, propiamente las pilas y colas en sus tipos FIFO y FILO, así también se auxilió de distintas librerías y lectura a los archivos de configuración por medio de un archivo XML. El usuario ingresa los archivos que configuran el sistema y después le es factible realizar una simulación o prueba del comportamiento de la atención al cliente y la cola visualizándolo gráficamente por estados.

Palabras clave

Listas , enlazada, nodo, pila, cola.

Abstract

Soluciones Guatemaltecas S.A has consulted for a software developing to lead the customer service attention for their clients in their different branch offices and different settings.

For the software developing the data structure linked list and its node class were mainly used, properly stacks and queues in their types FIFO and FILO have been implemented too, at the same time in aid of other libraries and lecture of the XML files. The user inputs the files that set up the system and later the simulation or trial is performed of the customer attendance it is viable to see its behavior and visualizing graphically by states.

Keywords

Lists, linked, node, stack, queue.

Introducción

Es indispensable que las empresas de hoy en día estén actualizadas e innoven constantemente en la forma en que atienden a sus clientes. El desarrollo de aplicaciones relacionadas a la atención al cliente es cada vez más indispensable, y por supuesto que es clave llevar un control en como, por quien y de que manera se da la atención al cliente.

El software desarrollado consiste en una lectura de archivos de configuración para el sistema de los cuales se pueden obtener las distintas empresas, puntos de atención y básicamente distintas configuraciones. Posterior a las cargas se puede realizar una prueba real de los clientes añadidos y como sería su comportamiento y atención en los distintos escritorios en varios puntos de atención que posean las empresas, permitiendo la visualización gráfica de una forma agradable de estas.

Desarrollo del tema

Se empleo el lenguaje multiparadigma de Python para la realización del software requerido creando una aplicación del tipo consola y aprovechando las distintas librerías que ofrece el lenguaje para realizar un código eficiente además de utilizar principalmente las estructuras de datos denominadas listas simples enlazadas, colas y pilas para la realización del programa.

El lenguaje de Python tiene demasiadas herramientas entre ellas las listas nativas de Python, ahora bien ¿Por qué utilizar listas enlazadas en vez de listas simples? Primero se debe conceptualizar correctamente las listas que son básicamente una colección de elementos sin embargo en muchos lenguajes estas listas o arreglos nativos tienen las dimensiones ya definidas a diferencia de las listas enlazadas que puede modificarse, se define a una lista enlazada como una colección lineal de estructuras que se auto referencian esto se denominan nodos, y estas están conectadas por medio de enlaces apuntadores.

¿Pero que tienen que ver las colas y pilas con una lista enlazada?

Bueno primero se define a una cola como aquella estructura de datos que almacena los datos en forma de lista en esta caso lista enlazada y permite la obtención de

los datos por el extremo superior o inferior de la lista, sabiendo como se comportan las colas en la realidad en el mundo de la programación puede hacerse de varias formas pero generalmente los objetos en el orden en que lleguen irán saliendo, es decir el primero en entrar será el primero en salir, a lo que se le denomina FIFO. Donde se encolan los objetos en la parte inferior y se obtienen o desencolan en la parte superior.

Por otro lado las pilas se comportan casi igual en esencia pero únicamente se pueden añadir y quitar datos u objetos por un único extremo.

En el desarrollo de la aplicación se utilizó lista simplemente enlazada para poder llenarla con los datos e información de las empresas obteniendo los datos del XML y por medio de métodos ir llenando la lista simple donde cada empresa apunta a la siguiente y esto facilita la manipulación y almacenamiento de los datos. Para acceder al XML se utilizó la librería minidom.

El enfoque principal de la aplicación desarrollada es tener un banco de empresas con distintos puntos de atención dándoles una nomenclatura e identificación respectiva, así también escritorios para atender por cada punto de atención y transacciones disponibles para ser realizadas.

En la lectura del 2do archivo nuevamente los datos fueron añadidos a una lista enlazada principalmente y después separando todo acorde al tipo de dato que se deseaba almacenar en otras listas enlazadas. La idea de este archivo es posteriormente realizar una ejecución, prueba o simulación con los clientes añadidos con sus respectivas transacciones, los escritorios activos y esta simulación siendo desarrollada en la empresa y punto de atención especificados, teniendo una visualización gráfica del comportamiento de la simulación o más bien la cola y atención en archivos .png generados por la aplicación con la librería graphviz, permitiendo verificar los clientes en cola, los clientes siendo atendidos y los distintos tiempos.

ARCHIVOS	CLASES
main.py	Main
Nodo.py	Nodo
listaempresa.py	Listaempresa
listasimulacion.py	Listasimulacion
listacola.py	Listacola
listaclientes.py	Listaclientes
escritoriossim.py	listadesks

Figura 1
Fuente: elaboración Propia

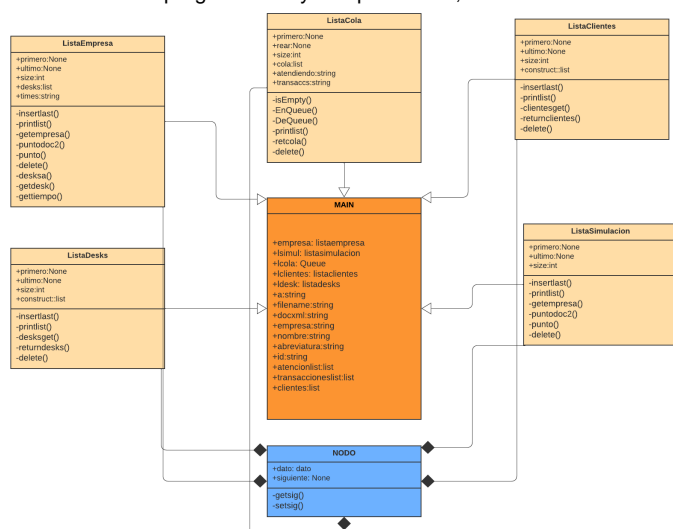


Figura 2

Fuente: elaboración Propia

En el uso de las listas simplemente enlazadas es la que ya ha sido definida es decir están compuestos por una clase nodo con los métodos inicializable, get que retorna los datos, set que pone. Esta clase Nodo es la única utilizada para todas las listas enlazadas en la aplicación

La clase de lista empresa cumple con el objetivo de almacenar todos los datos del primer archivo de entrada y tiene su inicializable acorde a los datos que se almacenan por cada empresa así como todos los gets(returns) de los distintos datos a los cuales se busque acceder, siendo así la esencia para todas las demás clases del tipo lista enlazada.

Siendo la excepción la clase de listacola que tiene como objetivo llamar a los clientes del

archivo de simulación y encolarlos de abajo hacia arriba según el orden en que vengan e ir desencolando desde el extremo superior teniendo así sus métodos enqueue() y dequeue() así también un método que retorna el estado de la cola .

Finalmente en la clase main a la hora de decidirse realizar la simulación se llaman a todos los objetos que se involucran en estas y los distintos métodos para finalmente con un algoritmo basado en bucles ir encolando y desencolando a la vez que se crean los digraphs(graphviz) para mostrar el estado de la atención en los escritorios que pueden modificarse en tiempo real y ver el estado de la cola.

Conclusiones

¿Es realmente más efectivo usar listas enlazadas? Si, indudablemente las listas enlazadas tienen demasiadas ventajas destacando primeramente que el orden de los elementos que han sido enlazados puede ser diferente al de almacenamiento, y cada nodo posee un espacio de almacenamiento en otras palabras, los nodos no tienen que estar contiguos en la memoria. Además se sigue manteniendo una secuencia ordenada y el acceso a los datos que se almacenan en los nodos es sumamente rápido y los algoritmos de búsqueda y ordenamiento son aplicables a estos además de que se pueden generar otro

tipo de estructuras como las matrices en base a estas listas enlazadas.

Los algoritmos de búsqueda siguen siendo un pilar de la programación y son la base del software desarrollado auxiliándose de la estructura de datos listas enlazadas y sus apuntadores pueden buscarse datos específicos en los nodos y obtener todos los datos que almacena ese nodo en base a un parámetro. Empleando diferentes apuntadores es decir con distintas referencias otros nodos es de suma utilidad pues la modificación o edición de los datos contenidos en los nodos es bastante efectiva y si fuese necesario añadir o eliminar también es bastante bueno.

Aplicando los conocimientos de este tipo de estructuras y auxiliándose con las librerías que ofrece este lenguaje de programación se desarrolló exitosamente un software que cumple a grandes rasgos con los objetivos planteados y un acceso bastante rápido y simplificado para el usuario.

Es indispensable el manejo de las estructuras de colas y pilas pues son amplias sus aplicaciones en el ámbito laboral para el desarrollo de ordenes o simplemente atención al cliente por ellos es primordial saber su estructura y posibles aplicaciones.

Referencias bibliográficas

García, M. E. (2015). *Estructurad de datos lineales: Lista Doblemente Ligada y doblemente ligada Circular*.

Rodríguez, R. R. (s.f.). Estrcuturas de Datos.

Wachenchauzer, R. (2014). *Uniwebsidad*.
Obtenido de Otras listas enlazadas.

Apéndice

Figura 2

