



UNIVERSIDAD DE MARGARITA

ALMA MATER DEL CARIBE

VICERRECTORADO ACADÉMICO

DECANATO DE INGENIERIA

UNIDAD CURRICULAR: ESTRUCTURA DE DATOS

**INFORME PROYECTO 2: SISTEMA COSTAZUL**

Profesor:

Cesar Requena

Realizado por:

José Ferreira C.I: V-28.315.655

Paola Márquez C.I: V-27.125.784

El Valle del Espíritu Santo, 5 de diciembre del 2023

Para las etapas iniciales del proyecto se revisaron cuales son los datos que debían ser solicitados al usuario, y se consideró que lo más lógico debía ser empezar por la fecha y hora actual. Para ello, se creó un JFrame Form llamado “Bienvenido” que además de darle la bienvenida al usuario, le solicita los datos de fecha y hora que serán usados posteriormente, mediante listas desplegables o JComboBox con la información.

Luego de eso, se abre un nuevo JFrame Form llamado “DatosUsuario” donde se solicitan mediante JTextField y JComboBox los datos como nombre, apellido, tipo y numero de documento y la hora de salida del centro comercial. Adicionalmente el sistema registra si el usuario entró con carro o no, por lo que se usó un JCheckBox para habilitar o deshabilitar las opciones de los datos del vehículo. Otro dato que se solicita es que en caso de que la persona entró con vehículo, se le pregunta si ingresó con acompañantes, lo que se usará más adelante para agilizar el registro de personas.

Para el manejo y almacenamiento interno de esos datos, se crearon clases llamadas “Persona” y “Vehículo”. La primera almacena datos de las personas que se registran en el sistema, como su ID única, el nombre, apellido, el documento, horas de entrada y salida, junto con la información del objeto de la clase vehículo en caso de tenerlo. La segunda, almacena datos del vehículo, como su dueño (que es una instancia de Persona), tipo de vehículo, marca, color y placa. Cuando el usuario ingresa los datos en el form, se crea un objeto de la clase Persona y Vehículo (en caso de tenerlo) con esa información y se introduce a una pila de personas, pasando a ser su tope y manipulándose su información.

Dicha pila es una instancia de la clase “Pila”, que contiene nodos con valores de Personas. Se usó una pila porque la idea es manipular el ultimo elemento creado he introducido en la estructura, algo que no podría hacerse de forma tan cómoda con una cola, por ejemplo. Cada usuario en sesión se va almacenando y manipulando a medida que entra en la pila.

De haber ingresado con un vehículo, se abre un JFrame Form llamado “vEstacionamiento”, donde se le solicita al usuario que ingrese el puesto de estacionamiento a ocupar. Para el manejo del estacionamiento, se dividió en 7 sectores, 2 de motos (sector Palma y Bambú 2, 100 puestos c/u) y 5 de carros (sector Mangle, Samán, Roble, Guayacán, y Bambú 1, 500 puestos c/u) para un total de 2700 puestos. Se creó una clase llamada “puestoEst” para almacenar la información de deben contener los puestos: su sector, el número y una lista de ocupantes para llevar registro de las

personas que lo están usando. Evidentemente que se creó una clase “Nodo” y “Lista” con sus respectivos métodos para almacenar y manipular esos datos.

Como la cantidad de puestos del estacionamiento es fija, se decidió usar una estructura estática para guardarlos, un arreglo bidimensional de puestos de 5x500 para los carros y de 2x100 para las motos. En el momento en que el usuario selecciona un puesto usando los JComboBox destinados para ello, se accede al puesto en el arreglo en ese sector y con ese número, y usando el método “estaOcupado” de la clase Lista, se recorre la lista de ocupantes de ese puesto, comparando sus horas de entrada y salida con el que se desea ingresar para determinar si existen choques de hora. En caso de no haberlo, retorna “true” y la persona es ingresada al final de la lista.

Posteriormente de seleccionar un puesto, o en caso de que el usuario no usó el estacionamiento, se le pregunta si desea realizar una compra. De ser positiva la respuesta, se abre un JFrame Form llamado “vCompras” que contiene el mapa de tiendas. Para el manejo de las tiendas, se creó una clase llamada “Tienda” que almacena el nombre, encargado, ubicación (la entrada más cercana), el número del local y una lista con los productos que tiene a la venta. Para los productos se creó una clase “Producto” que contiene el nombre, descripción y el precio de cada instancia.

Cuando el usuario selecciona una tienda, toda su información aparece en un JPanel y al confirmarla, se abre un JFrame Form llamado “MenuCompra” que contiene todas las opciones y datos para la compra de un producto en la tienda. Es por esto que se creó una clase llamada “Compra”, que contiene la información del comprador (instancia de Persona), tienda (instancia de Tienda), producto (instancia de Producto), unidades compradas, precio total, el tipo de compra (online o presencial) y el método de pago (efectivo, TDD, TDC, Pagomovil, transferencia o Zelle). Una vez confirmada la compra, se crea el objeto de la clase Compra y se almacena en la lista de compras que posee cada Persona como atributo.

Si la persona con vehículo selecciona que trajo acompañantes, se repite el proceso anterior con los acompañantes (los cuales no poseen vehículo, sólo el conductor) hasta llenar la pila de personas con los usuarios en sesión, para luego ser vaciada, introduciendo cada elemento en un árbol AVL de personas, abriéndose nuevamente la ventana de Bienvenido para usuarios nuevos.

Tanto las personas registradas en el sistema, como las tiendas, son almacenadas y manipuladas usando un árbol AVL (un árbol binario que se autobalancea), debido a su efectividad en la búsqueda

y acceso a la información cuando se manejan muchos datos. Esto se consigue con la clase “AVL”, que contiene su nodo raíz con la información de su hijo izquierdo y derecho, además de los métodos necesarios para su funcionamiento.

Por otro lado, es necesario llevar un registro de toda la información manejada, por lo que se usaron archivos de texto que son manipulados mediante la clase “Archivos”. Al iniciar el programa, se chequea la existencia de 3 archivos: “Tiendas.txt”, “Personas.txt” y “Puestos.txt”. En el primero, están almacenados todos los datos de las 300 tiendas que posee el centro comercial, que es leída para crear las 300 instancias de la clase Tienda y ser almacenadas en el árbol AVL para su fácil acceso. El 2do guarda todos los datos de los usuarios existentes, incluyendo su vehículo y compras, que es leído para crear las instancias de Persona que se van introduciendo en el árbol AVL de personas. El 3ro contiene la información de cada puesto, incluyendo si posee ocupantes y quienes son, el cual es leído para crear las 2700 instancias de la clase puestoEst y ser introducidas en su respectiva posición de los arreglos de puestos. Cabe destacar que esta información en los archivos se actualiza cada vez que se abre la ventana de “Bienvenida”, después de terminar el registro de todos los usuarios de una sesión.

Respecto a las funciones de modificar y eliminar la información de un usuario ya registrado, hay un apartado en el JFrame Form “Bienvenido” de “Editar información”, que al ser accedido se abre un nuevo JFrame Form llamado “menuEditar”. Éste permite buscar un usuario ya sea por nombre y apellido, o cédula. Si consigue coincidencias, muestra en un JComboBox las fechas de los registros de las mismas, mostrando toda la información de datos personales, vehículo y compras para poder ser editada. También aparece la opción “Eliminar” que borra por completo la información de ese registro del sistema.