Documentación

November 6, 2010

Descripción de los diferentes módulos del programa

TTipoLocalidad

TTipoLocalidad es un nuevo tipo que definirá mediante un enumerado los distintos tipos de localidad de los que dispone un teatro. Estos son:

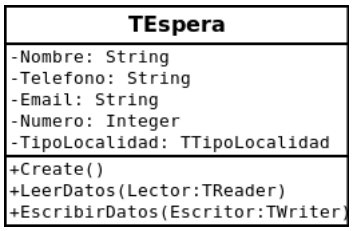
- Patio
- Primera planta
- Palco
- Indiferente

TEstadoLocalidad

TEstadoLocalidad será un enumerado que definirá el estado de la localidad que pueden ser:

- Libre
- Comprada
- Reservada

TEspera



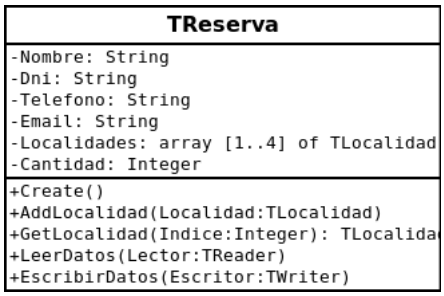
TEspera es una estructura de datos que implementa cada uno de los elementos que componen la lista de esperas, que nos servirá para gestionar los datos de las personas que están en ella.

Para ello necesitaremos unos atributos privados que serán: Nombre, Telefono, Email, Numero y Tipo de Localidad. Estos datos son los que el software guarda para identificar a los usuarios, poder contactar con ellos y saber el tipo de localidad deseada.

Las funciones y procedimientos que posee esta estructura de datos son:

- Create() : Es un constructor que crea una nueva espera vacía.
- LeerDatos(Lector: TReader) : Es un procedimiento que obtiene mediante parámetros un buffer por el cual va a poder leer los datos de una espera de un fichero.
- EscribirDatos(Escritor: TWriter): Es un procedimiento que obtiene mediante parámetros un buffer por el cual va a poder escribir los datos de una espera en un fichero.

TReserva



TReserva es una estrucutra de datos que implementa cada uno de los elementos que componen la lista de reservas. Mediante esta estructura podremos gestionar los datos de las personas que han realizado una reserva.

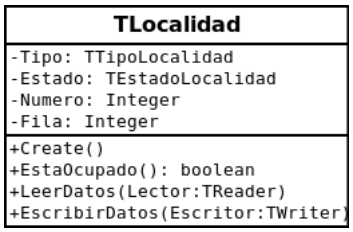
TReserva posee una serie de atributos privados que son: Nombre,DNI, Telefono, Email, Localidades, Cantidad. El nombre y dni nos servirán para identificar a los clientes. El teléfono y el email para contactar con los cilentes si es necesario. El atributo Localidades es un array de 4 posiciones que contiene un

tipo TLocalidad. Por último cantidad contiene el número de localidades que contiene la reserva.

Esta estructura implementa una serie de funciones y procedimientos para poder gestionar adecuadamente las reservas:

- Create() : Es un constructor que crea una nueva resreva vacía.
- AddLocalidad(Localidad: TLocalidad) : Es un procedimiento que recibe un TLocalidad y lo añade a la reserva.
- GetLocalidad(Indice: Integer):TLocalidad : Se trata de una función que recibe un índice(entre 1 y 4), y busca en el array que contiene las localidades de la reserva y devuelve el TLocalidad que se encuentra en la posición del índice dado.
- LeerDatos(Lector: TReader) : Es un procedimiento que obtiene mediante parámetros un buffer por el cual va a poder leer los datos de una reserva de un fichero.
- EscribirDatos(Escritor: TWriter): Es un procedimiento que obtiene mediante parámetros un buffer por el cual va a poder escribir los datos de una reserva en un fichero.

TLocalidad



TLocalidad es una estructura de datos que implementa cada una de las localidades del teatro.

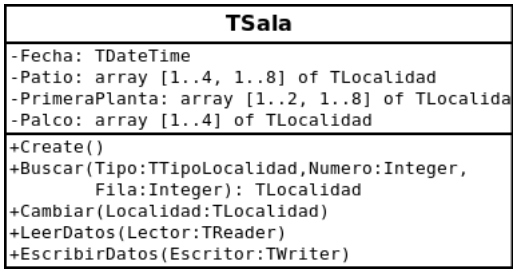
Esta estructura posee una serie de atributos que permiten identificar la localidad. Mediante el atributo Tipo sabemos de que tipo de localidad se trata, con Estado sabemos si se encuentra ocupada o no, y con Numero y Fila podemos identificarla dentro del teatro.

Las funciones y procedimientos son los siguientes:

- Create(): Definirá el constructor
- EstaOcupado(): Devolverá un boolealo, que será verdadero en el caso de que la localidad esté ocupada por algún usuario.
- LeerDatos(Lector: TReader): Es un procedimiento que obtiene mediante parámetros un buffer por el cual va a poder leer los datos de una localidad en un fichero.

- EscribirDatos(Escritor: TWriter): Es un procedimiento que obtiene mediante parámetros un buffer por el cual va a poder escribir los datos de una localidad en un fichero.

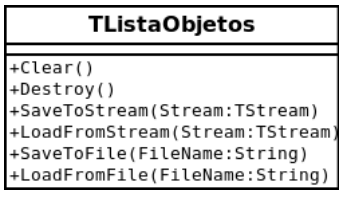
TSala



TSala es una estructura de datos que se encargará de gestionar las posibles operaciones de la sala. Para ello distinguimos los siguientes atributos con los que trabajaremos: Fecha, Patio, PrimeraPlanta y Palco. Nos serviremos de los siguientes métodos:

- Create(): Constructor que inicializará los variables de la sala
- Buscar(Tipo: TTipoLocalidad, Numero: Integer, Fila: Integer): Método que devolverá un tipo TLocalidad que permitirá buscar a través del tipo de localidad, el número y la fila.
- Cambiar(Localidad: TLocalidad): Método que cambiará la localidad
- LeerDatos(Lector: TReader): Es un procedimiento que obtiene mediante parámetros un buffer por el cual va a poder leer los datos de una sala en un fichero.
- EscribirDatos(Escritor: TWriter): Es un procedimiento que obtiene mediante parámetros un buffer por el cual va a poder escribir los datos de una sala en un fichero.

TListaObjetos



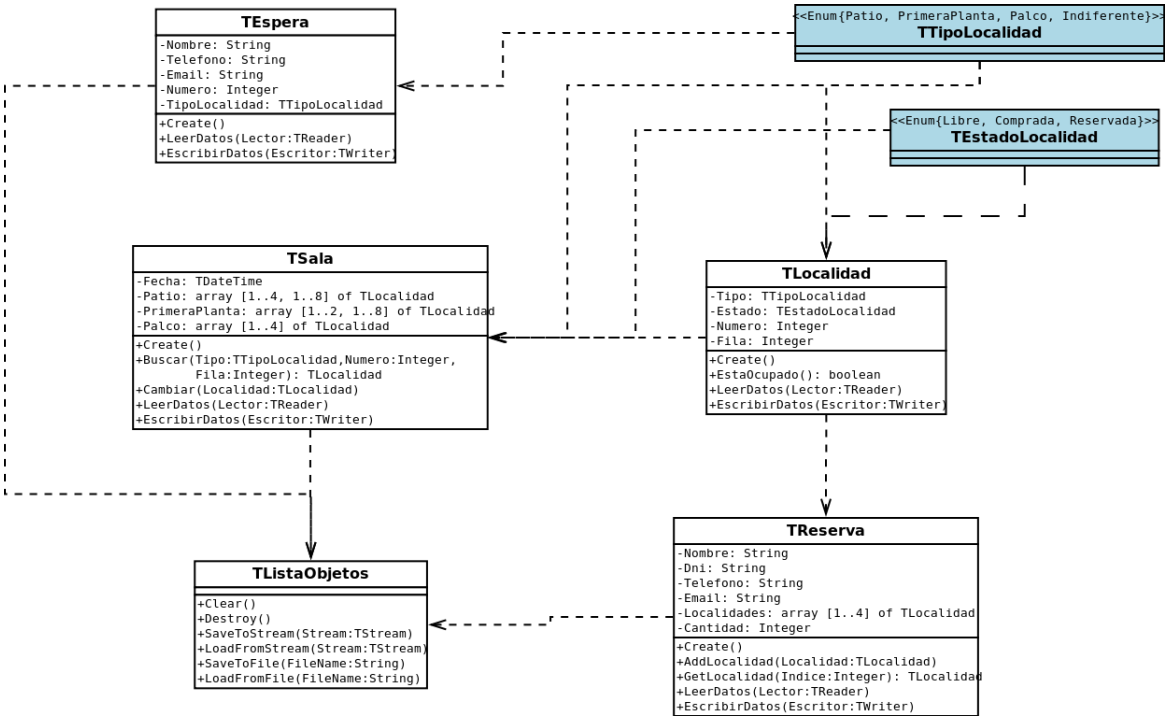
Es una lista de objetos que pueden adoptar diferentes tipos de datos: TSala, TReserva y TEspera

- Clear(): Limpiará la lista

- Destroy(): Liberará espacio en el disco
- SaveToStream(Stream: TStream): Guardará a Stream el objeto
- LoadFromStream(Stream: TStream) Cargará de Stream el objeto
- SaveToFile(FileName: String): Nos permite guardar en un fichero la lista de objetos
- LoadFromFile(FileName: String): Nos permite cargar de un fichero la lista de objetos

Visión General

En el siguiente esquema UML se muestra un resumen general de como se relacionan los modulos descritos anteriormente:



Herramientas de diseño

En el desarrollo de la práctica nos servimos de las siguientes herramientas:

1. Git: es un sistema de control de versiones.

2. Entorno de programación Lazarus: es un IDE de Free Pascal que ofrece una alternativa libre a Delphi.

3. Uso de diagramas UML con el programa de software libre DIA.