

alidar(BE.Secuencia): List(Of BE.Error)

### BLL.Config

- + Guardar(BE.Config): void
- + ObtenerConfig(): BE.Config

### BROperation

- DVH: DV
- Fecha: Datetime
- ID\_Operacion: Long
- Path: String

- + Ejecutar(): void

### Backup

- + Cargar(BROperation): o: BROperation
- + Ejecutar(): void
- + Eliminar(): void
- + Grabar(BROperation): void
- + Obtener\_Backups(List(Of BROperation)): List(Of BROperation)

### Restore

- + Cargar(BROperation): o: BROperation
- + Ejecutar(): void
- + Guardar(BROperation)
- + ObtenerRestores(List(Of BROperation)): List(Of BROperation)

### Digitos

#### DV

- Valor

- + Calcular(object): long

#### Digito\_Vertical

- + Calcular(object): long

#### Digito\_Horizontal

- tabla: String

- + Calcular(object): long

### Cifrado

#### Cifrado

- + desencriptar\_fec(date): Date
- + desencriptar\_num(double): Double
- + desencriptar\_str(string): String
- + encriptar\_fec(date): Date
- + encriptar\_num(double): Double
- + encriptar\_str(string): String

DAL.PuntodeVenta
+ Guardar(List( Of BE.PuntodeVenta)): void + ObtenerPDV(List( Of BE.PuntodeVenta)): List( Of BE.PuntodeVenta)

DAL.Sesion
+ Validar_DV(): List(Of Be.Error)

DAL.Secuencia
+ Guardar(BE.Secuencia): void + ObtenerSecuencias(List( Of BE.Secuencia)): List( Of BE.Secuencia)

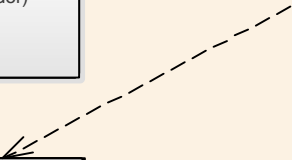
DAL.Config
+ Guardar(BE.Config): void + ObtenerConfig(): BE.Config

DAL.Idioma
+ Grabar(BE.Idioma): void + Obtener_Idiomas(List(Of BE.Idioma)): List(Of BE.Idioma)

DAL.Vendedor
+ Guardar(f: List(Of BE.Vendedor)): void + ObtenerVendedores(List(Of BE.Vendedor)): List(Of BE.Vendedor)

DAL.Elemento
+ AgregarPermiso(Elemento, Elemento, BE.Usuario) + Cargar(Elemento): Elemento + Guardar(Elemento) + ObtenerGrupos(Elemento, BE.Usuario, BE.Pantalla): Listof(Elemento) + ObtenerPermisos(Elemento, BE.Usuario, BE.Pantalla): Listof(Elemento) + QuitarPermiso(Elemento, Elemento, BE.Usuario)

DAL.Bitacora
+ Cargar(): Long + Guardar(): void + ObtenerEntradas(BE.Filtro): List(of Entrada)



DAL.Concurso

+ Eliminar(BE.Concurso): void  
+ Guardar(BE.Concurso): void  
+ ObtenerConcursos(List(Of BE.Concurso)): List(Of BE.Concurso)  
+ ObtenerDistConcurso(List(Of BE.Concurso)): List(Of BE.Dist\_Concurso)

DAL.Usuario

+ Agregar\_elemento(Elemento): void  
+ Guardar(BE.Usuario): void  
+ ObtenerUsuarios(List(Of BE.Usuario)): List(Of BE.Usuario)

DAL.Restore

+ Guardar(BROperation)  
+ ObtenerRestores(List(Of BROperation)): List(Of BROperation)

DAL.Stock

+ Guardar(List(Of BE.Stock)): void  
+ ObtenerStocks(List(Of BE.Stock)): List(Of BE.Stock)

DAL.Backup

+ Eliminar(BROperation): void  
+ Grabar(BROperation): void  
+ Obtener\_Backups(List(Of BROperation)): List(Of BROperation)

DAL.Leyenda

+ Guardar\_Leyendas(BE.Idioma): void  
+ ObtenerLeyendas(BE.Idioma): List(Of BE.Leyenda)  
+ ObtenerLeyendas(BE.Idioma, List(Of BE.Error)): List(Of BE.Leyenda)

DAL.Detalle\_Factura

+ Guardar(List( Of BE.Detalle\_Factura)): void  
+ ObtenerDetalles(List( Of BE.Factura)): List( Of BE.Detalle\_Factura)

DAL.Error

+ Guardar(BE.Error): void  
+ ObtenerErrores(List( Of BE.Error)): List( Of BE.Error)

DAL.Factura

+ Guardar(BE.Secuencia): void  
+ ObtenerFacturas(List(Of BE.Factura)): List(Of BE.Factura)

DAL.Provincia

+ ObtenerProvincias(List( Of BE.Provincia)): List( Of BE.Provincia)



