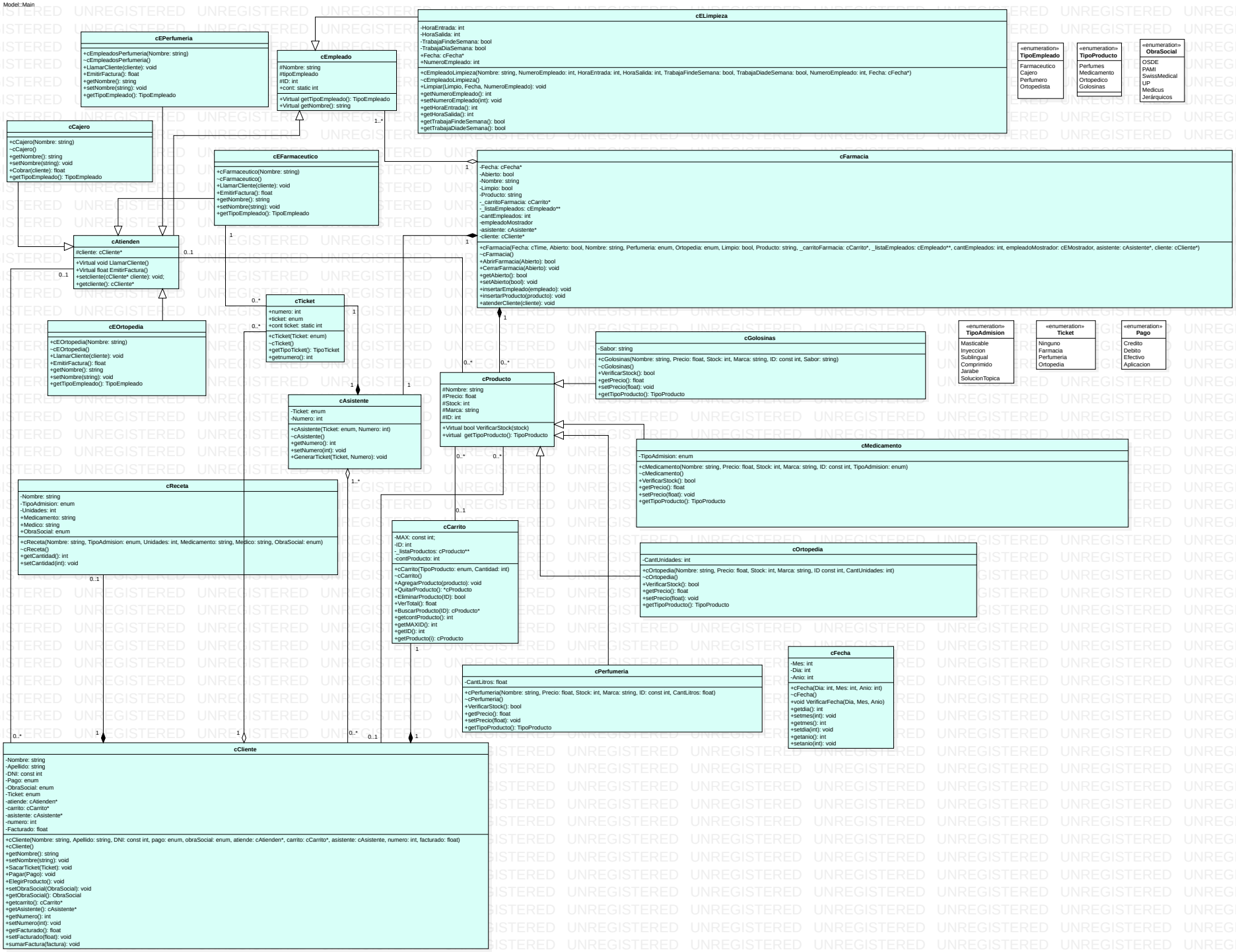
Model::Main


cEPerfumeria
+cEmpleadosPerfumeria(Nombre: string)
~cEmpleadosPerfumeria()
+LlamarCliente(cliente): void
+EmitirFactura(): float
+getNombre(): string
+setNombre(string): void
+getTipoEmpleado(): TipoEmpleado
cEmpleado
#Nombre: string
#tipoEmpleado
#ID: int
+cont: static int
+Virtual getTipoEmpleado(): TipoEmpleado
+Virtual getNombre(): string
cELimpieza
-HoraEntrada: int
-HoraSalida: int
-TrabajaFindeSemana: bool
-TrabajaDiaSemana: bool
+Fecha: cFecha*
+NumeroEmpleado: int
+cEmpleadoLimpieza(Nombre: string, NumeroEmpleado: int, HoraEntrada: int, HoraSalida: int, TrabajaFindeSemana: bool, TrabajaDiadeSemana: bool, NumeroEmpleado: int, Fecha: cFecha*)
~cEmpleadoLimpieza()
+Limpiar(Limpio, Fecha, NumeroEmpleado): void
+getNumeroEmpleado(): int
+setNumeroEmpleado(int): void
+getHoraEntrada(): int
+getHoraSalida(): int
+getTrabajaFindeSemana(): bool
+getTrabajaDiadeSemana(): bool
«enumeration» TipoEmpleado
Farmaceutico
Cajero
Perfumero
Ortopedista
«enumeration» TipoProducto
Perfumes
Medicamento
Ortopedico
Golosinas
«enumeration» ObraSocial
OSDE
PAMI
SwissMedical
UP
Medicus
Jerárquicos
cCajero
+cCajero(Nombre: string)
~cCajero()
+getNombre(): string
+setNombre(string): void
+Cobrar(cliente): float
+getTipoEmpleado(): TipoEmpleado
cEFarmaceutico
+cFarmaceutico(Nombre: string)
~cFarmaceutico()
+LlamarCliente(cliente): void
+EmitirFactura(): float
+getNombre(): string
+setNombre(string): void
+getTipoEmpleado(): TipoEmpleado
cFarmacia
-Fecha: cFecha*
-Abierto: bool
-Nombre: string
-Limpio: bool
-Producto: string
-_carritoFarmacia: cCarrito*
-_listaEmpleados: cEmpleado**
-cantEmpleados: int
-empleadoMostrador
-asistente: cAsistente*
-cliente: cCliente*
+cFarmacia(Fecha: cTime, Abierto: bool, Nombre: string, Perfumeria: enum, Ortopedia: enum, Limpio: bool, Producto: string, _carritoFarmacia: cCarrito*, _listaEmpleados: cEmpleado**, cantEmpleados: int, empleadoMostrador: cEMostrador, asistente: cAsistente*, cliente: cCliente*)
~cFarmacia()
+AbrirFarmacia(Abierto): bool
+CerrarFarmacia(Abierto): void
+getAbierto(): bool
+setAbierto(bool): void
+insertarEmpleado(empleado): void
+insertarProducto(producto): void
+atenderCliente(cliente): void
cAtienden
#cliente: cCliente*
+Virtual void LlamarCliente()
+Virtual float EmitirFactura()
+setcliente(cCliente* cliente): void;
+getcliente(): cCliente*
cTicket
+numero: int
+ticket: enum
+cont ticket: static int
+cTicket(Ticket: enum)
~cTicket()
+getTipoTicket(): TipoTicket
+getnumero(): int
cEOrtopedia
+cEOrtopedia(Nombre: string)
~cEOrtopedia()
+LlamarCliente(cliente): void
+EmitirFactura(): float
+getNombre(): string
+setNombre(string): void
+getTipoEmpleado(): TipoEmpleado
cGolosinas
-Sabor: string
+cGolosinas(Nombre: string, Precio: float, Stock: int, Marca: string, ID: const int, Sabor: string)
~cGolosinas()
+VerificarStock(): bool
+getPrecio(): float
+setPrecio(float): void
+getTipoProducto(): TipoProducto
«enumeration» TipoAdmision
Masticable
Inyeccion
Sublingual
Comprimido
Jarabe
SolucionTopica
«enumeration» Ticket
Ninguno
Farmacia
Perfumeria
Ortopedia
«enumeration» Pago
Credito
Debito
Efectivo
Aplicacion
cProducto
#Nombre: string
#Precio: float
#Stock: int
#Marca: string
#ID: int
+Virtual bool VerificarStock(stock)
+virtual getTipoProducto(): TipoProducto
cAsistente
-Ticket: enum
-Numero: int
+cAsistente(Ticket: enum, Numero: int)
~cAsistente()
+getNumero(): int
+setNumero(int): void
+GenerarTicket(Ticket, Numero): void
cMedicamento
-TipoAdmision: enum
+cMedicamento(Nombre: string, Precio: float, Stock: int, Marca: string, ID: const int, TipoAdmision: enum)
~cMedicamento()
+VerificarStock(): bool
+getPrecio(): float
+setPrecio(float): void
+getTipoProducto(): TipoProducto
cReceta
-Nombre: string
-TipoAdmision: enum
-Unidades: int
+Medicamento: string
+Medico: string
+ObraSocial: enum
+cReceta(Nombre: string, TipoAdmision: enum, Unidades: int, Medicamento: string, Medico: string, ObraSocial: enum)
~cReceta()
+getCantidad(): int
+setCantidad(int): void
cCarrito
-MAX: const int;
-ID: int
-_listaProductos: cProducto**
-contProducto: int
+cCarrito(TipoProducto: enum, Cantidad: int)
~cCarrito()
+AgregarProducto(producto): void
+QuitarProducto(): *cProducto
+EliminarProducto(ID): bool
+VerTotal(): float
+BuscarProducto(ID): cProducto*
+getcontProducto(): int
+getMAXID(): int
+getID(): int
+getProducto(i): cProducto
cOrtopedia
-CantUnidades: int
+cOrtopedia(Nombre: string, Precio: float, Stock: int, Marca: string, ID const int, CantUnidades: int)
~cOrtopedia()
+VerificarStock(): bool
+getPrecio(): float
+setPrecio(float): void
+getTipoProducto(): TipoProducto
cPerfumeria
-CantLitros: float
+cPerfumeria(Nombre: string, Precio: float, Stock: int, Marca: string, ID: const int, CantLitros: float)
~cPerfumeria()
+VerificarStock(): bool
+getPrecio(): float
+setPrecio(float): void
+getTipoProducto(): TipoProducto
cFecha
-Mes: int
-Dia: int
-Anio: int
+cFecha(Dia: int, Mes: int, Anio: int)
~cFecha()
+void VerificarFecha(Dia, Mes, Anio)
+getdia(): int
+setmes(int): void
+getmes(): int
+setdia(int): void
+getanio(): int
+setanio(int): void
cCliente
-Nombre: string
-Apellido: string
-DNI: const int
-Pago: enum
-ObraSocial: enum
-Ticket: enum
-atiende: cAtienden*
-carrito: cCarrito*
-asistente: cAsistente*
-numero: int
-Facturado: float
+cCliente(Nombre: string, Apellido: string, DNI: const int, pago: enum, obraSocial: enum, atiende: cAtienden*, carrito: cCarrito*, asistente: cAsistente, numero: int, facturado: float)
+cCliente()
+getNombre(): string
+setNombre(string): void
+SacarTicket(Ticket): void
+Pagar(Pago): void
+ElegirProducto(): void
+setObraSocial(ObraSocial): void
+getObraSocial(): ObraSocial
+getcarrito(): cCarrito*
+getAsistente(): cAsistente*
+getNumero(): int
+setNumero(int): void
+getFacturado(): float
+setFacturado(float): void
+sumarFactura(factura): void
1..*
1
0..1
0..*
1
1
1
1
0..*
0..*
0..*
0..*
1
1
1..*
0..1
0..1
1
0..*
1
0..*
0..1
1