****

**Tecnicatura Universitaria en Programación**

**Base de Datos II**

**Equipo # 29**

**Integrantes:**

**Correa Gimenez, Leandro Sebastian - 23908**

**Jesus Esteban Ballico – 30884**

**Facundo Palomanes – 28799**

Explicación del Sistema

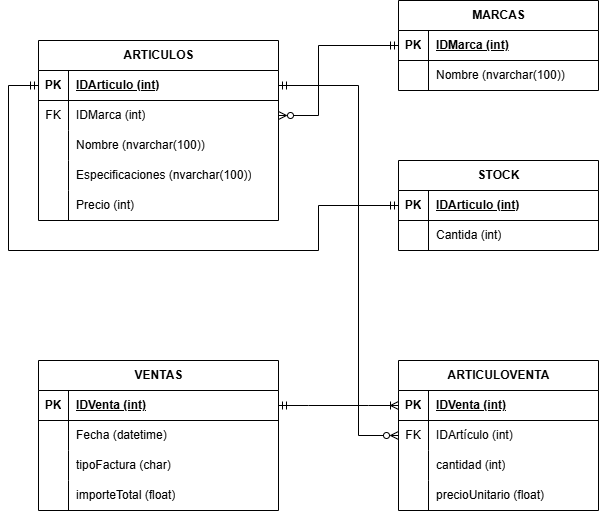
El sistema trata de un punto de venta de carácter general, orientado a pequeños y medianos comercios como kioskos, almacenes y locales de carácter general. El propósito del mismo es facilitar la administración del local, contando con diversas funcionalidades para registrar los movimientos del día a día, y en base a ellas, obtener métricas y reportes para llevar un control del modelo de negocio.

### Funcionalidades principales

* **Gestión del stock:** Permite realizar la carga de artículos al inventario disponible, gestionar su stock disponible del mismo para la venta, asignarle un nombre, marca, categoría, precio.
* **Registro de ventas:** Permite registrar una venta.
* **Reportes y métricas:** A través de las ventas registradas, permite obtener reportes claves para el negocio, como productos más vendidos, ingresos totales por día / semana / mes, reporte de stock actual.
* **Ordenes de compra:** En base al stock que va resultando, permite realizar ordenes de compras de los artículos que tengan un stock por debajo del umbral normal.

El sistema está construido con una base de datos relacional que almacena toda la información necesaria para gestionar estas funcionalidades. [Se han implementado tres vistas, dos procedimientos almacenados y dos triggers para mejorar la administración y la integridad de los datos.]

## Diagrama de Entidad Relación



## Objetos de Base de Datos clave en el sistema

### Vista 1

Se utiliza la vista VW\_SolicitudesPendientes que muestra todas las solicitudes de días libres pendientes de dictamen. Solamente podrán verla los administradores del sistema y la misma cuenta con la siguiente información:

IDEmpleado, Apellidos y nombres del empleado, Mail del Empleado, Antigüedad en la empresa, Nombre del Área y Fecha de la última solicitud de día libre.

|  |
| --- |
| CREATE VIEW SolicitudesPendientes AS SELECT  e.IDEmpleado,  e.Apellidos,  e.Nombres -- Resto de la lógica de la vista. Deberán adjuntar el código completo de su script. |

### Vista 2

Aquí se explica el propósito de la Vista 2 del sistema.

### Vista 3

Aquí se explica el propósito de la Vista 2 del sistema.

### Procedimiento Almacenado 1

Se utiliza el procedimiento almacenado llamado SP\_NuevoEmpleado que permite generar un nuevo empleado. Valida que las fechas de ingreso a la empresa y la fecha de nacimiento sean consistentes. También verifica que la dirección de correo electrónico sea una dirección válida.

|  |
| --- |
| CREATE PROCEDURE SP\_NuevoEmpleado  @IDArea INT,  @Apellidos NVARCHAR(100),  @Nombres NVARCHAR(100),  @Email NVARCHAR(100),  @Telefono NVARCHAR(20),  @Celular NVARCHAR(20),  @FechaNacimiento DATE,  @FechaIngreso DATE,  @CantidadDiasLibres INT,  @Sueldo DECIMAL(10,2) AS BEGIN  --Lógica completa del script para crear un nuevo empleado utilizando un bloque try-catch para personalizar los errores que pueden generarse. END; |

### Procedimiento Almacenado 2

Aquí se explica el propósito del Procedimiento Almacenado 2 de la base de datos.

### 

### Trigger 1

Se utiliza el trigger TR\_SolicitudDiaLibre que permite registrar una nueva solicitud de día libre por parte de un empleado aplicando la siguiente regla de negocio que automatiza el rechazo de las mismas dadas las siguientes condiciones:

* Se rechaza automáticamente si el empleado solicita un día libre en el mismo mes en el que fue sancionado.
* Se rechaza automáticamente si el empleado solicita un día libre con fecha consecutiva a otro día libre solicitado previamente y que haya sido aprobado.

|  |
| --- |
| CREATE TRIGGER TR\_SolicitudDiaLibre ON SolicitudDeDiasLibres AFTER INSERT AS BEGIN  DECLARE @IDEmpleado INT, @FechaSolicitud DATE;   -- Obtener los datos de la nueva solicitud  SELECT @IDEmpleado = IDEmpleado, @FechaSolicitud = FechaSolicitud  FROM INSERTED;  -- Resto de la lógica del trigger para aplicar los rechazos automáticos a partir de las reglas de negocio.  END; |

### Trigger 2

Aquí se explica el propósito del Trigger 2 de la base de datos.

## Links a los recursos

**Script de creación de base de datos con datos**

<https://github.com/Salchy/TP-integrador-dbii>

**Video demo del sistema (hasta 25 minutos)**

<https://youtube.com/usuario/VideoDemo>