ÍNDICE

[**1. PROBLEMÁTICA 1**](#_4eyzygqy3pzp)

[**2. IDENTIFICACIÓN DE ENTIDADES 1**](#_mtf0mrotdqkw)

[**3. MODELO ER 2**](#_iro8yxhl2kqx)

[**4. DICCIONARIO DE DATOS 3**](#_1s1a0b8zoc85)

[Tabla: Usuario 3](#_lr7b9pkfi8h)

[Tabla: Rol 4](#_qw1vv2wmal11)

[Tabla: AsignacionRol 4](#_83do8436mij8)

[Tabla: Tabla 4](#_7k69citygeg9)

[Tabla: Permiso 5](#_tf4w1rp6jsgy)

[Tabla:Departamento 5](#_wwc2jyxpgyno)

[Tabla:PeridoPago 5](#_ltg6y5ct0rsl)

[Tabla: Empleado 5](#_4ue0ykh3y098)

[Tabla: Pago 6](#_j5carirovc5o)

[Tabla: AuditoriaPago 6](#_oldkbcu3jbmw)

[Tabla: AuditoriaEmpledo 7](#_8p6429jlz3p)

[**5. PROCEDIMIENTOS ALMACENADOS 7**](#_67xtbcohmfpw)

[a. ObtenerPermisosUsuario 7](#_35nvb6vrax1i)

[b. ObtenerCamposYTablaPorRol 8](#_54non3mtwwgz)

[c. EjecutarConsultaPorRolYTabla 9](#_va2ookbdclp7)

[d. EjecutarConsultaPorRolYTabla 10](#_29ulmcyyya2a)

[e. sp\_GenerarCSV\_ReporteNominaPorEmpleado 11](#_8gu3xdtskubd)

[**6. Triggers (Disparadores) 12**](#_czs4zkf6rtsw)

# **PROBLEMÁTICA**

Ayudar a Buy n Large, una empresa de tecnología, a desarrollar un Sistema de Control de Permisos en una API.

Aunque se usará un Software de Nómina como escenario, el verdadero objetivo es implementar un sistema de permisos granular, donde quien interactúe con el sistema tenga acceso restringido a nivel de tabla y registro, según sus roles o usuario.

# **IDENTIFICACIÓN DE ENTIDADES**

* 1. **Entidades de Nómina**

| **ENTIDAD** | **DESCRIPCIÓN** |
| --- | --- |
| Empleado | Contiene la información básica de cada empleado, incluyendo su relación con un departamento. |
| Departamento | Define los departamentos dentro de la empresa a los que pueden estar asignados los empleados. |
| Pago | Registra los pagos hechos a cada empleado en un periodo determinado, con el monto correspondiente. |
| PeriodoPago | Define los intervalos de tiempo para la ejecución de pagos de nómina. |

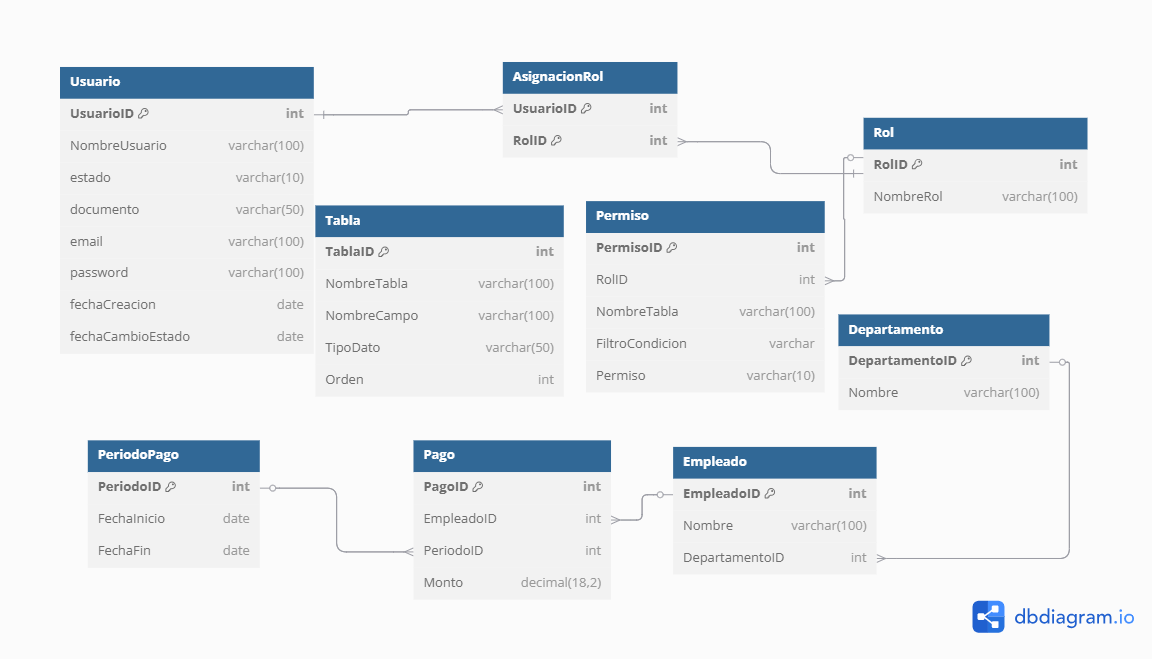
* 1. **Entidades de manejo de Permisos**

| **ENTIDAD** | **DESCRIPCIÓN** |
| --- | --- |
| **Usuario** | Representa a los usuarios del sistema, incluyendo datos personales, credenciales y estado. |
| **Rol** | Define los diferentes roles que pueden existir en el sistema, los cuales agrupan permisos específicos. |
| **AsignacionRol** | Representa la asignación de uno o más roles a cada usuario, permitiendo una relación muchos a muchos. |
| **Permiso** | Define el tipo de acceso que un rol tiene sobre una tabla específica, incluyendo condiciones de filtro opcionales. |
| **Tabla** | Contiene metadatos de las tablas y campos de la base de datos, útil para configuración dinámica o control de permisos. |

* 1. **Entidades Auditoria**

| **ENTIDAD** | **DESCRIPCIÓN** |
| --- | --- |
| **AuditoriaPago** | Registra todos los cambios realizados en la tabla Pago, incluyendo los valores anteriores y nuevos, quién los hizo y cuándo. |
| **AuditoriaEmpleado** | Registra los cambios en los datos del empleado, como nombre y departamento, incluyendo información del usuario que hizo el cambio. |

# **MODELO ER**

****

# **DICCIONARIO DE DATOS**

## **Tabla: Usuario**

| **Campo** | **Tipo** | **Clave** | **Nulo** | **Descripción** |
| --- | --- | --- | --- | --- |
| UsuarioID | INT | PK | NO | Identificador único del usuario |
| NombreUsuario | VARCHAR(100) |  | NO | Nombre visible o alias del usuario |
| estado | VARCHAR(10) |  | SÍ | Estado del usuario (ACTIVO o INACTIVO) |
| documento | VARCHAR(50) |  | NO | Documento de identidad del usuario |
| email | VARCHAR(100) |  | NO | Correo electrónico |
| password | VARCHAR(100) |  | NO | Contraseña encriptada o hash |
| fechaCreacion | DATE |  | NO | Fecha de creación del usuario |
| fechaCambioEstado | DATE |  | SÍ | Fecha del último cambio de estado |

## **Tabla: Rol**

| **Campo** | **Tipo** | **Clave** | **Nulo** | **Descripción** |
| --- | --- | --- | --- | --- |
| RolID | INT | PK | NO | Identificador del rol |
| NombreRol | VARCHAR(100) |  | NO | Nombre descriptivo del rol |

## **Tabla: AsignacionRol**

| **Campo** | **Tipo** | **Clave** | **Nulo** | **Descripción** |
| --- | --- | --- | --- | --- |
| UsuarioID | INT | PK/FK | NO | Relación al usuario |
| RolID | INT | PK/FK | NO | Relación al rol |

## 

## **Tabla: Tabla**

| **Campo** | **Tipo** | **Clave** | **Nulo** | **Descripción** |
| --- | --- | --- | --- | --- |
| TablaID | INT | PK | NO | Identificador interno |
| NombreTabla | VARCHAR(100) |  | NO | Nombre de la tabla referenciada |
| NombreCampo | VARCHAR(100) |  | NO | Nombre del campo en esa tabla |
| TipoDato | VARCHAR(50) |  | NO | Tipo de dato del campo |
| Orden | INT |  | NO | Orden de aparición del campo |

## **Tabla: Permiso**

| **Campo** | **Tipo** | **Clave** | **Nulo** | **Descripción** |
| --- | --- | --- | --- | --- |
| PermisoID | INT | PK | NO | Identificador del permiso |
| RolID | INT | FK | NO | Relación al rol que recibe el permiso |
| NombreTabla | VARCHAR(100) |  | SÍ | Tabla a la que aplica el permiso |
| FiltroCondicion | VARCHAR(MAX) |  | SÍ | Condición opcional para restringir acceso |
| Permiso | VARCHAR(10) |  | NO | Tipo de permiso (ej. SELECT, UPDATE, etc.) |

## **Tabla:Departamento**

| **Campo** | **Tipo** | **Clave** | **Nulo** | **Descripción** |
| --- | --- | --- | --- | --- |
| DepartamentoID | INT | PK | NO | Identificador del departamento |
| Nombre | VARCHAR(100) |  | NO | Nombre del departamento |

## **Tabla:PeridoPago**

| **Campo** | **Tipo** | **Clave** | **Nulo** | **Descripción** |
| --- | --- | --- | --- | --- |
| PeriodoID | INT | PK | NO | Identificador del período |
| FechaInicio | DATE |  | NO | Fecha de inicio del período |
| FechaFin | DATE |  | NO | Fecha de finalización del período |

## **Tabla: Empleado**

| **Campo** | **Tipo** | **Clave** | **Nulo** | **Descripción** |
| --- | --- | --- | --- | --- |
| EmpleadoID | INT | PK | NO | Identificador del empleado |
| Nombre | VARCHAR(100) |  | NO | Nombre completo del empleado |
| DepartamentoID | INT | FK | NO | Departamento al que pertenece |

## **Tabla: Pago**

| **Campo** | **Tipo** | **Clave** | **Nulo** | **Descripción** |
| --- | --- | --- | --- | --- |
| PagoID | INT | PK | NO | Identificador del pago |
| EmpleadoID | INT | FK | NO | Empleado que recibe el pago |
| PeriodoID | INT | FK | NO | Período al que corresponde el pago |
| Monto | DECIMAL(18,2) |  | NO | Monto pagado |

## **Tabla: AuditoriaPago**

| **Campo** | **Tipo** | **Clave** | **Nulo** | **Descripción** |
| --- | --- | --- | --- | --- |
| AuditoriaID | INT | PK | NO | Identificador del registro de auditoría |
| PagoID | INT |  | SÍ | Pago modificado |
| EmpleadoID\_Ant | INT |  | SÍ | Valor anterior del EmpleadoID |
| EmpleadoID\_Nue | INT |  | SÍ | Valor nuevo del EmpleadoID |
| PeriodoID\_Ant | INT |  | SÍ | Valor anterior del PeriodoID |
| PeriodoID\_Nue | INT |  | SÍ | Valor nuevo del PeriodoID |
| Monto\_Ant | DECIMAL(18,2) |  | SÍ | Monto anterior del pago |
| Monto\_Nue | DECIMAL(18,2) |  | SÍ | Monto nuevo del pago |
| FechaHora | DATETIME |  | NO | Fecha y hora de la operación |
| UsuarioEjecuta | NVARCHAR(128) |  | SÍ | Usuario que realizó la operación |
| HostName | NVARCHAR(128) |  | SÍ | Host o estación de trabajo del usuario |
| TipoOperacion | CHAR(1) |  | NO | Tipo de operación: 'I', 'U', 'D' |

## **Tabla: AuditoriaEmpledo**

| **Campo** | **Tipo** | **Clave** | **Nulo** | **Descripción** |
| --- | --- | --- | --- | --- |
| AuditoriaID | INT | PK | NO | Identificador del registro de auditoría |
| EmpleadoID | INT |  | NO | Empleado afectado |
| Nombre\_Ant | VARCHAR(100) |  | SI | Nombre anterior del empleado |
| Nombre\_Nue | VARCHAR(100) |  | SI | Nuevo nombre del empleado |
| DepartamentoID\_Ant | INT |  | SI | Departamento anterior |
| DepartamentoID\_Nue | INT |  | SI | Departamento nuevo |
| FechaHora | DATETIME |  | NO | Fecha y hora de la operación |
| UsuarioEjecuta | NVARCHAR(128) |  | NO | Usuario que realizó el cambio |
| HostName | NVARCHAR(128) |  | NO | Host de donde se ejecutó el cambio |
| TipoOperacion | CHAR(1) |  | NO | Tipo de operación: 'I', 'U', 'D' |

# **PROCEDIMIENTOS ALMACENADOS**

## **ObtenerPermisosUsuario**

Retorna los roles y permisos asociados a un usuario específico mediante su ID. Este procedimiento verifica si el usuario tiene permisos asignados y, en caso afirmativo, devuelve los roles, nombres de las tablas y permisos correspondientes. Si no tiene permisos, lanza un error.

**Entradas:**

| **Nombre** | **Tipo** | **Descripción** |
| --- | --- | --- |
| @UsuarioID | INT | Identificador del usuario para consultar sus permisos. |

### 

**Salida exitosa:**

| **Nombre del Campo** | **Tipo** | **Descripción** |
| --- | --- | --- |
| RolID | INT | ID del rol asignado al usuario. |
| NombreRol | VARCHAR | Nombre del rol asignado. |
| NombreTabla | VARCHAR | Nombre de la tabla sobre la cual se tiene permiso. |
| Permiso | VARCHAR | Tipo de permiso otorgado (SELECT, INSERT, etc.). |

#### 

**Salidas fallidas:**

| **Código de Error** | **Mensaje** | **Descripción** |
| --- | --- | --- |
| 50001 | El usuario no tiene permisos asignados. | Se lanza mediante THROW si no existen permisos para el usuario. |

## **ObtenerCamposYTablaPorRol**

Devuelve los nombres de los campos, tipos de datos concatenados y permisos asociados a una tabla específica sobre la cual un rol tiene permisos. Utiliza funciones de agregación para presentar la información de forma compacta.

**Entradas:**

| **Nombre** | **Tipo** | **Descripción** |
| --- | --- | --- |
| @RolID | INT | Identificador del rol cuyos permisos se desean consultar. |
| @NombreTabla | VARCHAR(100) | Nombre de la tabla sobre la cual se desea obtener la información. |

**Salida exitosa:**

| **Nombre del Campo** | **Tipo de Dato** | **Descripción** |
| --- | --- | --- |
| NombreTabla | VARCHAR | Nombre de la tabla consultada. |
| CamposConcatenados | VARCHAR(MAX) | Lista de nombres de campos, separados por comas, ordenados por posición. |
| TiposConcatenados | VARCHAR(MAX) | Lista de tipos de datos correspondientes a los campos, también ordenados. |
| Permiso | VARCHAR | Tipo de permiso otorgado (por ejemplo: SELECT, UPDATE, etc.). |

**Salida Fallida:**

| **Código de Error** | **Mensaje** | **Descripción** |
| --- | --- | --- |
| N/A | N/A | Este procedimiento no lanza errores explícitos si no hay datos; simplemente retorna 0 filas si no hay coincidencias. |

## **EjecutarConsultaPorRolYTabla**

Ejecuta dinámicamente una consulta SELECT sobre una tabla específica, utilizando los campos definidos para la tabla y las condiciones de filtro asignadas al rol. Verifica que el rol tenga permisos sobre la tabla antes de ejecutar. En caso de error en la consulta, lanza una excepción controlada.

**Entradas:**

| **Nombre** | **Tipo** | **Descripción** |
| --- | --- | --- |
| @RolID | INT | Identificador del rol que ejecutará la consulta. |
| @NombreTabla | VARCHAR(100) | Nombre de la tabla sobre la cual se ejecutará la consulta. |

**Salida exitosa:**

| **Resultado** | **Descripción** |
| --- | --- |
| SELECT dinámico | Devuelve los registros seleccionados de la tabla especificada según los campos y condición del rol. |

**Salida fallida:**

| **Código de Error** | **Mensaje** | **Descripción** |
| --- | --- | --- |
| 50001 | El rol no tiene permisos sobre la tabla especificada. | Se lanza si no existen registros de permisos para ese rol y tabla en la tabla Permiso. |
| 50002 | Error al ejecutar la consulta dinámica: <detalle del error> | Se lanza si ocurre un error en la ejecución dinámica del SQL generado con sp\_executesql. |

## **EjecutarConsultaPorRolYTabla**

Ejecuta de forma dinámica una sentencia DML (como INSERT, UPDATE, DELETE, MERGE, etc.) que recibe como parámetro. Utiliza sp\_executesql e implementa manejo de errores.

**Entradas:**

| **Nombre** | **Tipo** | **Descripción** |
| --- | --- | --- |
| @cadenaDML | NVARCHAR(MAX) | Cadena de texto con la sentencia DML a ejecutar. |

**Salida exitosa:**

| **Campo** | **Tipo** | **Descripción** |
| --- | --- | --- |
| Mensaje | NVARCHAR(100) | Devuelve el texto: 'Procedimiento ejecutado con éxito.' |

**Salida fallida:**

| **Código de Error** | **Mensaje** | **Descripción** |
| --- | --- | --- |
| 50010 | Error al ejecutar la sentencia DML: <detalle del error> | Se lanza si ocurre un error durante la ejecución dinámica con sp\_executesql. |

## **sp\_GenerarCSV\_ReporteNominaPorEmpleado**

Genera un archivo .csv con el reporte de pagos de nómina para un empleado específico, utilizando bcp y xp\_cmdshell. Válida si existen datos antes de exportar.

**Entradas:**

| **Nombre** | **Tipo** | **Descripción** |
| --- | --- | --- |
| @EmpleadoID | INT | Identificador del empleado para el cual se genera el reporte. |
| @RutaArchivo | NVARCHAR(255) | Ruta completa donde se guardará el archivo .csv. Ejemplo: 'C:\reportes\reporte\_nomina.csv' |

**Salida exitosa:**

| **Resultado** | **Descripción** |
| --- | --- |
| Archivo generado | Se crea un archivo .csv en la ruta indicada con los datos de nómina del empleado. |
| Ejecución correcta | Si todo funciona correctamente, no hay salida explícita; el archivo es la prueba del éxito. |

**Salida fallida:**

| **Condición** | **Mensaje de Error** |
| --- | --- |
| No existen pagos registrados para el empleado | No hay registros de pagos para el EmpleadoID proporcionado. |
| Error al ejecutar xp\_cmdshell o bcp | Error al generar el archivo CSV: <mensaje específico del sistema> |

# **Triggers (Disparadores)**

| **Nombre** | **Descripción** |
| --- | --- |
| trg\_AuditoriaEmpleado\_Delete | Inserta en la tabla AuditoriaEmpleado cuando hay updates sobre registros en la tabla Empleados. |
| trg\_AuditoriaEmpleado\_Insert | Inserta en la tabla AuditoriaEmpleado cuando hay Inserts sobre registros en la tabla Empleados. |
| trg\_AuditoriaEmpleado\_Update | Inserta en la tabla AuditoriaEmpleado cuando hay deletes sobre registros en la tabla Empleados. |
| trg\_AuditoriaPago\_Delete | Inserta en la tabla AuditoriaPago cuando hay deletes sobre registros en la tabla Pago. |
| trg\_AuditoriaPago\_Insert | Inserta en la tabla AuditoriaPago cuando hay inserts sobre registros en la tabla Pago. |
| trg\_AuditoriaPago\_Update | Inserta en la tabla AuditoriaPago cuando hay updates sobre registros en la tabla Pago. |