

**ITCR – Escuela de Ing. en Computación – Base de Datos 1 – Prof. fquiros –
Setiembre de 2025**

Segunda tarea programada.

Uno. Objetivo.

Implementar una aplicación sencilla que realiza:

- validaciones de entrada de datos,
- consultas sobre varias tablas en la BD con despliegue de información en una interfaz de usuario,
- la captura, edición y validación de valores que se salvan en una tabla,
- el registro de movimientos en una aplicación transaccional,
- el registro de actividades en la BD a través de una bitácora de eventos,
- la implementación de manejo de errores en procedimientos almacenados; y.
- la escritura de transacciones de BD.

Dos. Descripción General.

Se trata de implementar una aplicación sencilla para hacer CRUD de una tabla de Empleado (como parte de un módulo de control de vacaciones en sistema de recursos humanos), y realizar algunas consultas con filtros por nombre del empleado o valor de documento de identidad, así como el registro de movimientos de vacaciones, etc. La aplicación incluye validación de ingreso de usuarios mediante clave y contraseña, así como un registro en una bitácora de todas las operaciones que un usuario realiza en la BD.

Tres. Modelo Físico.

Respecto del modelo conceptual visto en clases para el problema de Control de vacaciones, se implementan los siguientes Conjuntos Entidad:

Nombre	Atributos
Puesto	Id PK, Nombre, SalarioxHora
Empleado	Id PK, IdPuesto FK, ValorDocumentoidentidad, Nombre, FechaContratacion, SaldoVacaciones, EsActivo
TipoMovimiento	Id PK, Nombre, TipoAccion
Movimiento	Id PK, IdEmpleado FK, IdTipoMovimiento FK, Fecha, Monto, NuevoSaldo, IdPostByUser FK, PostInIP, PostTime
Usuario	Id PK, Username, Password
TipoEvento	Id PK, Nombre
BitacoraEvento	Id PK, idTipoEvento FK, Descripcion, IdPostByUser FK, PostInIP, PostTime

DBError	ID PK, UserName, Number, State, Severity, Line, Procedure, Message, DateTime
Error	Id PK, Codigo, Descripcion

Cuando hay una condición de error que es encontrada en los SPs, el SP envía a capa lógica un código de error (código de error es 0 si no hubo error), la capa lógica consulta en la tabla Error usando ese código para obtener el texto que se envía a capa de presentación.

Cuatro. Requerimientos funcionales.

R1. Login, Logout.

Para ingresar a la aplicación se muestra una página que captura el UserName y el Password. Si durante los últimos 5 minutos el usuario ha hecho más de 5 login no exitosos, se deshabilita el la acción de login (aunque no el botón), y se envía un mensaje ("Demasiados intentos de login, intente de nuevo dentro de 10 minutos). Implica que al mostrar la página de login hay que consultar la bitácora de eventos a ver si ese usuario en esa ip fue deshabilitado hace menos 5 minutos.

R2. Listar empleados, filtro por nombre o valor de documento de identidad, y selección de un empleado para realizar operaciones.

Esta opción lista todos los empleados ordenados por nombre ascendente, permite seleccionar un empleado de la lista, tal que sobre el mismo se pueda hacer Consulta, Borrado o Modificación (requerimiento R4), listar movimientos (requerimiento R5) o insertar un movimiento (requerimiento R6). La selección del empleado para realizar alguna operación NO DEBE hacerse mediante la captura de su id. En esta tarea los ids estarán escondidos.

La interfaz que muestra esta lista debe contener una caja de texto al inicio de la lista junto con un botón que se llama **Filtrar** (o botón de buscar), en la cual, el usuario ingresa un texto que puede ser es un nombre de empleado aproximado o un valor de documento de identidad aproximado. Al dar click en <Filtrar> la aplicación detecta si el filtro contiene solo letras o espacios en blanco, hace una búsqueda aproximada por nombre; si contiene solo números, hace una búsqueda aproximada por valor de documento de identidad, si solo contiene números.

Ejemplos:

Valor de filtro	Acción
Perez	Lista todos los empleados que tiene el string 'Perez' como parte del nombre.
603	Lista todos los empleados que tienen el string '603' en el valor de documento de identidad.
<vacío o espacios en blanco>	Se listan todos los empleados.

La interfaz que muestra la lista de empleados es la principal, desde la cual hay un botón de insertar o botones para Consultar, eliminar o actualizar un empleado; o listar movimientos de un empleado.

R3. Insertar empleado.

Esta acción se activa desde un botón en la interfaz principal (R2).

Se levanta una ventana en la cual se solicita el Valor de documento de Identidad (i.e. número de cédula), nombre, idPuesto (que debe seleccionarse de una lista – drop down list con los puestos en orden alfabético, o sea NO se captura el Id). Se debe validar que no exista otro empleado con mismo nombre ni valor de documento de identidad. El valor de documento de identidad debe ser numérico, el nombre debe ser alfabético y puede contener espacios entre nombre y apellidos.

R4. ABC de Empleados (Actualización, Borrado y Consulta).

Se selecciona un empleado para el ABC. Una es que haya botones de ABC, a la derecha de cada empleado en la lista que muestra la interfaz principal (R2).

Se trata de hacer delete, update o consulta de un empleado.

Operación	Descripción
Update (Actualización)	Permite la edición del Valor de Documento de Identidad, nombre y idPuesto de un empleado existente. El saldo de vacaciones no se puede editar. Igual se debe validar que al modificar el documento de identidad o el nombre, no exista otro empleado con mismo documento de identidad o mismo nombre.
Delete (Borrado)	Se levanta una alerta que muestra Valor de documento de identidad y nombre del empleado y un mensaje indicando “Esta seguro de eliminar este empleado?”, y si el usuario confirma, el campo de activo se pone en “0” (borrado Lógico).
Consulta	Se muestra el valor de documento de identidad, nombre, nombre del puesto y SaldoVacaciones.

R5. Listar Movimientos.

En una nueva ventana se muestra el Valor de documento de identidad del empleado, su nombre, el saldo de vacaciones actual y una lista con todos sus movimientos ordenados en orden descendente de fecha (el movimiento más reciente se muestra de primero). La lista debe mostrar para cada movimiento: fecha, Nombre del tipo de movimiento, monto del movimiento, nuevo saldo despues de aplicar el movimiento, nombre del usuario que registró el movimiento, la IP y la estampa de tiempo.

La interfaz de Listar Movimientos debe contener un botón que lleva a Insertar un movimiento al empleado cuyos movimientos se estan listando.

R6. Insertar Movimiento.

Se levanta una ventana que muestra valor de documento de identidad, el nombre del empleado y, el saldo vacaciones. Solicita el tipo de movimiento (debe seleccionarse de una lista) y monto. Debe validarse que la aplicación del monto no haga que el saldo se haga negativo.

R7. Trazabilidad.

Toda operación que realiza un usuario en el sistema debe quedar registrado en una bitácora, a continuación se muestran los tipos de evento y la información que se almacena en el campo de descripción, además de que los campos IdPostByUser, IPPostIn, y PostTime deben ir llenados también.

Tipo de Evento	Información que se guarda en BitacoraEvento en campo Descripción.
Login Exitoso	Nada
Login No exitoso	Numero de intento en los últimos 5 minutos yCodigo de Error
Login deshabilitado	Nada
Logout	Nada
Inserción no exitosa	Descripción del error (tomado del catálogo de errores), valor documento identidad, Nombre del empleado, Nombre del Puesto, respecto de los valores que se intentó insertar sin éxito.
Inserción exitosa	Valor documento identidad, Nombre del empleado, y Nombre del Puesto, de lo insertado.
Update No exitoso	Descripción del error, y los valores de documento identidad, Nombre del empleado, Nombre del Puesto, valor documento identidad que se intentó modificar. El saldo de vacaciones se agrega aunque no es editable.
Update Exitoso	Valor documento identidad antes de editarse, Nombre del empleado antes de editarse, Nombre del Puesto antes de editarse, valor documento identidad despues de editarse, Nombre del empleado despues de editarse, Nombre del Puesto despues de editarse, y el SaldoVacaciones.
Intento de Borrado (el usuario no confirmo el borrado)	Valor documento identidad, Nombre del empleado, Nombre del Puesto, y SaldoVacaciones.
Borrado Exitoso	Valor documento identidad, Nombre del empleado, identidad, Nombre del Puesto, y SaldoVacaciones
Consulta con filtro de nombre	Valor del filtro del nombre
Consulta con filtro de cedula	Valor del filtro de cedula
Intento de insertar movimiento	Descripción del error, Valor de documento identidad del empleado, nombre, Saldo actual, Nombre de tipo de movimiento, y monto.
Insertar movimiento exitoso	Valor de documento identidad del empleado, nombre, Nuevo Saldo, nombre de tipo de movimiento, y monto.

R8. Mensajes de error.

Respecto de los SP que realizan operaciones y de algunos errores que se detectan en capa de lógica, cuando el error es detectado, se envía un código de error a algún método que consultará la descripción del error en una tabla de la base de datos, esta tabla es entonces un catálogo de errores.

Habrán códigos de error por los siguientes conceptos

Usuario inválido
Login deshabilitado
Empleado con ValorDocumentoidentidad ya existe en inserción
Empleado con mismo nombre ya existe en inserción
Empleado con ValorDocumentoidentidad ya existe en actualización
Empleado con mismo nombre ya existe en actualización
Error de base de datos (detectado en catch)
Nombre de empleado no alfabético
Valor de documento de identidad no alfabético
Monto del movimiento rechazado pues si se aplica, el saldo sería negativo.

Cinco. Datos de prueba.

Un equipo de trabajo hará los datos de prueba, los cuales vendrán en formato XML.

Los catálogos se cargarán desde archivos XML que contienen los ids, nombre y otros valores de todos los catálogos. En estos catálogos, el id no será autoincremental, o sea viene dado en el XML, para asegurarnos que todos los Ids son los mismos para todos los equipos de trabajo, esto será así excepto para todos los catalogos excepto catálogo de puestos.

Igualmente para poder probar consultas con empleados y movimientos, el equipo de trabajo aportará archivos xml con datos de prueba de empleados y movimientos.

El siguiente segmento es un ejemplo del XML para carga de datos:

```
<Datos>
<Puestos>
  <Puesto Nombre="Cajero" SalarioxHora="11.00"/>
  <Puesto Nombre="Camarero" SalarioxHora="10.00"/>
  <Puesto Nombre="Cuidador" SalarioxHora="13.50"/>
  <Puesto Nombre="Conductor" SalarioxHora="15.00"/>
  <Puesto Nombre="Asistente" SalarioxHora="11.00"/>
  <Puesto Nombre="Recepcionista" SalarioxHora="12.00"/>
  <Puesto Nombre="Fontanero" SalarioxHora="13.00"/>
  <Puesto Nombre="Niñera" SalarioxHora="12.00"/>
  <Puesto Nombre="Conserje" SalarioxHora="11.00"/>
  <Puesto Nombre="Albañil" SalarioxHora="10.50"/>
</Puestos>
<TiposEvento>
  <TipoEvento Id="1" Nombre="Login Exitoso"/>
  <TipoEvento Id="2" Nombre="Login No Exitoso"/>
  <TipoEvento Id="3" Nombre="Login deshabilitado"/>
  <TipoEvento Id="4" Nombre="Logout"/>
</TiposEvento>
```

```

    <TipoEvento Id="5" Nombre="Insercion no exitosa"/>
    <TipoEvento Id="6" Nombre="Insercion exitosa"/>
    <TipoEvento Id="7" Nombre="Update no exitoso"/>
    <TipoEvento Id="8" Nombre="Update exitoso"/>
    <TipoEvento Id="9" Nombre="Intento de borrado"/>
    <TipoEvento Id="10" Nombre="Borrado exitoso"/>
    <TipoEvento Id="11" Nombre="Consulta con filtro de nombre"/>
    <TipoEvento Id="12" Nombre="Consulta con filtro de cedula"/>
    <TipoEvento Id="13" Nombre="Intento de insertar movimiento"/>
    <TipoEvento Id="14" Nombre="Insertar movimiento exitoso"/>
</TiposEvento>
<TiposMovimientos>
    <TipoMovimiento Id="1" Nombre="Cumplir mes" TipoAccion="Credito"/>
    <TipoMovimiento Id="2" Nombre="Bono
vacacional" TipoAccion="Credito"/>
    <TipoMovimiento Id="3" Nombre="Reversion
Debito" TipoAccion="Credito"/>
    <TipoMovimiento Id="4" Nombre="Disfrute de
vacaciones" TipoAccion="Debito"/>
    <TipoMovimiento Id="5" Nombre="Venta de
vacaciones" TipoAccion="Debito"/>
    <TipoMovimiento Id="6" Nombre="Reversion de
Credito" TipoAccion="Debito"/>
</TiposMovimientos>
<Empleados>
    <empleado Puesto="Camarero" ValorDocumentoIdentidad="6993943" Nomb
re="Kaitlyn Jensen" FechaContratacion="2017-12-07"/>
    <empleado Puesto="Albañil" ValorDocumentoIdentidad="1896802" Nomb
re="Robert Buchanan" FechaContratacion="2020-09-20"/>
    <empleado Puesto="Cajero" ValorDocumentoIdentidad="5095109" Nomb
re="Christina Ward" FechaContratacion="2015-09-13"/>
    <empleado Puesto="Fontanero" ValorDocumentoIdentidad="8403646" Nomb
re="Bradley Wright" FechaContratacion="2020-01-27"/>
    <empleado Puesto="Conserje" ValorDocumentoIdentidad="6019592" Nomb
re="Robert Singh" FechaContratacion="2017-02-01"/>
    <empleado Puesto="Asistente" ValorDocumentoIdentidad="4510358" Nomb
re="Ryan Mitchell" FechaContratacion="2018-06-08"/>
    <empleado Puesto="Asistente" ValorDocumentoIdentidad="7517662" Nomb
re="Candace Fox" FechaContratacion="2013-12-17"/>
    <empleado Puesto="Asistente" ValorDocumentoIdentidad="8326328" Nomb
re="Allison Murillo" FechaContratacion="2020-04-19"/>
    <empleado Puesto="Cuidador" ValorDocumentoIdentidad="2161775" Nomb
re="Jessica Murphy" FechaContratacion="2017-04-12"/>
    <empleado Puesto="Fontanero" ValorDocumentoIdentidad="2918773" Nomb
re="Nancy Newton PhD" FechaContratacion="2016-11-22"/>
</Empleados>
<Usuarios>
    <usuario Id="1" Nombre="UsuarioScripts" Pass="UsuarioScripts"/>
    <usuario Id="1" Nombre="mgarrison" Pass=")*2LnSr^lk"/>
    <usuario Id="2" Nombre="jgonzalez" Pass="3YSI0Hti&I"/>
    <usuario Id="3" Nombre="zkelly" Pass="X4US4aLam@"/>
    <usuario Id="4" Nombre="andersondeborah" Pass="732F34xo%S"/>
    <usuario Id="5" Nombre="hardingmicheal" Pass="himB9Dzd%_"/>
    <usuario Id="6" Nombre="Usuario No validado" Pass="himB9Dzd%_"/>
</Usuarios>
<Movimientos>

```

<movimiento ValorDocId="7517662" IdTipoMovimiento="Venta de vacaciones" Fecha="2024-01-18" Monto="2" PostByUser="hardingmicheal" PostInIP="42.142.119.153" PostTime="2024-01-18 18:47:14"/>
<movimiento ValorDocId="6993943" IdTipoMovimiento="Bono vacacional" Fecha="2024-10-31" Monto="1" PostByUser="mgarrison" PostInIP="156.92.82.57" PostTime="2024-10-31 12:43:18"/>
<movimiento ValorDocId="8326328" IdTipoMovimiento="Venta de vacaciones" Fecha="2024-11-22" Monto="7" PostByUser="andersondeborah" PostInIP="218.213.110.232" PostTime="2024-11-22 00:23:53"/>
<movimiento ValorDocId="4510358" IdTipoMovimiento="Reversion de Credito" Fecha="2024-07-03" Monto="3" PostByUser="hardingmicheal" PostInIP="143.42.131.166" PostTime="2024-07-03 17:07:39"/>
<movimiento ValorDocId="8403646" IdTipoMovimiento="Reversion de Credito" Fecha="2024-12-07" Monto="8" PostByUser="zkelly" PostInIP="155.44.100.105" PostTime="2024-12-07 15:44:30"/>
<movimiento ValorDocId="8326328" IdTipoMovimiento="Venta de vacaciones" Fecha="2024-11-26" Monto="10" PostByUser="hardingmicheal" PostInIP="141.163.255.56" PostTime="2024-11-26 09:33:41"/>
<movimiento ValorDocId="6993943" IdTipoMovimiento="Disfrute de vacaciones" Fecha="2024-11-20" Monto="6" PostByUser="hardingmicheal" PostInIP="4.176.52.1" PostTime="2024-11-20 23:31:41"/>
<movimiento ValorDocId="2918773" IdTipoMovimiento="Disfrute de vacaciones" Fecha="2024-10-30" Monto="10" PostByUser="zkelly" PostInIP="220.164.108.231" PostTime="2024-10-30 03:55:57"/>
<movimiento ValorDocId="2161775" IdTipoMovimiento="Reversion Debito" Fecha="2024-06-13" Monto="2" PostByUser="hardingmicheal" PostInIP="135.223.57.22" PostTime="2024-06-13 13:28:39"/>
<movimiento ValorDocId="8403646" IdTipoMovimiento="Bono vacacional" Fecha="2024-01-01" Monto="6" PostByUser="zkelly" PostInIP="150.250.94.62" PostTime="2024-01-01 05:17:10"/>
<movimiento ValorDocId="2918773" IdTipoMovimiento="Venta de vacaciones" Fecha="2024-07-12" Monto="6" PostByUser="hardingmicheal" PostInIP="218.191.123.15" PostTime="2024-07-12 09:10:16"/>
<movimiento ValorDocId="5095109" IdTipoMovimiento="Reversion de Credito" Fecha="2024-12-27" Monto="14" PostByUser="hardingmicheal" PostInIP="136.103.23.170" PostTime="2024-12-27 12:59:03"/>
<movimiento ValorDocId="6993943" IdTipoMovimiento="Venta de vacaciones" Fecha="2024-04-08" Monto="1" PostByUser="jgonzalez" PostInIP="158.48.100.86" PostTime="2024-04-08 01:24:38"/>
<movimiento ValorDocId="8403646" IdTipoMovimiento="Bono vacacional" Fecha="2024-08-25" Monto="8" PostByUser="jgonzalez" PostInIP="204.0.219.231" PostTime="2024-08-25 16:24:07"/>

```

        <movimiento ValorDocId="5095109" IdTipoMovimiento="Bono
vacacional" Fecha="2024-03-
07" Monto="7" PostByUser="andersondeborah" PostInIP="208.0.4.33" Po
stTime="2024-03-07 08:19:28"/>
    </Movimientos>
    <Error>
        <errorCodigo="50001" Descripcion="Username no existe"/>
        <errorCodigo="50002" Descripcion="Password no existe"/>
        <errorCodigo="50003" Descripcion="Login deshabilitado"/>
        <errorCodigo="50004" Descripcion="Empleado con
ValorDocumentoIdentidad ya existe en inserción"/>
        <errorCodigo="50005" Descripcion="Empleado con mismo nombre ya
existe en inserción"/>
        <errorCodigo="50006" Descripcion="Empleado con
ValorDocumentoIdentidad ya existe en actualización"/>
        <errorCodigo="50007" Descripcion="Empleado con mismo nombre ya
existe en actualización"/>
        <errorCodigo="50008" Descripcion="Error de base de datos"/>
        <errorCodigo="50009" Descripcion="Nombre de empleado no
alfabético"/>
        <errorCodigo="50010" Descripcion="Valor de documento de identidad
no alfabético"/>
        <errorCodigo="50011" Descripcion="Monto del movimiento rechazado
pues si se aplicar el saldo seria negativo."/>
    </Error>
</Datos>

```

Importante es considerar que al cargar empleados, su saldo inicial será 0. Los movimientos no se cargan en lote, se cargarán uno a uno, y conforme se almacenan en tabla movimiento debe actualizar el saldo de vacaciones.

Seis. ¿Qué se pide?

El código en capa lógica y de presentación para el funcionamiento de las páginas.

La base de datos creada, las tablas con información cargada y los store procedimientos para realizar las funcionalidades solicitadas.

Un script con los estatutos para cargar los datos de prueba.

La documentación que son dos documentos, una bitácora y un análisis de resultados.

Bitácora y análisis de resultados, tal como fue indicado en la rúbrica de la 1era tarea programada.

Siete. Reglas.

Todo el código de programación referido a base de datos, debe ser un procedimiento almacenado. No puede haber SQL incrustado en capa lógica, lo único permitido es invocar un procedimiento almacenado. Si el desarrollo se hace

en plataforma .net, NO puede usarse linq para procesar información en la base de datos.

No se puede usar expresiones o estatutos SQL que el profe no haya ilustrado en algún ejemplo en clase, o que explícitamente haya indicado que no debe usarse (por ejemplo no debe usarse cursores).

El uso del github es obligatorio, será una prueba de trabajo constante y colaborativo.

Se debe usar el estándar de codificación indicado en clase.

Grupos de 2 personas. Motor de base de datos: MS SQL. Código en capa lógica, en el lenguaje o framework de su preferencia.

El valor de esta tarea programada es 13.3% de la nota.

Fecha de entrega: 13 de Octubre.