

TP Gestion de Datos 2C-2016
Grupo Siegfried
Estrategia

Integrantes:

Romero, Pablo Antonio

Crocco, Julian

DER: Adjuntado en otro archivo para que sea más visible.

Entidades:

- **Usuarios:** Entidad que contiene la información que comparten tanto Afiliados como profesionales
- **Afiliados:** Entidad que contiene la información específica de los afiliados.
- **Profesional:** Entidad que contiene la información específica de los profesionales
- **Roles_Usuarios:** Entidad asociativa entre roles y usuarios para modelar la relación muchos a muchos que tienen
- **Roles:** Entidad que contiene la información específica de los roles
- **Funcionalidades_Roles:** Entidad asociativa entre roles y funcionalidades para modelar la relación muchos a muchos que tienen
- **Funcionalidades:** Entidad que contiene la información específica de las funcionalidades
- **Especialidades:** Entidad que contiene la información específica de las especialidades
- **Profesional_Especialidad:** Entidad asociativa entre profesionales y especialidades para modelar la relación muchos a muchos que tienen.
- **Tipos_Especialidades:** Entidad que contiene la información específica de los tipos de especialidades
- **Plan:** Entidad que contiene la información específica de los planes
- **Historial_Usuarios_Planes:** Entidad que contiene la información específica del evento en el que un afiliado cambia de un plan a otro por determinado motivo.
- **Turnos:** Entidad que contiene la información específica de un turno de un afiliado.
- **Agenda:** Entidad que contiene la información específica de los horarios disponibles para solicitar turnos a un profesional.
- **Bonos:** Entidad que contiene la información específica de los bonos que tiene cada afiliado para si mismo y para sus familiares.

- **Compra_Bonos:** Entidad donde se almacenan los eventos de compras de bonos por parte de los afiliados.
- **Consultas:** Entidad que contiene la información específica de las consultas de afiliados a profesionales
- **Cancelacion:** Entidad que contiene la información específica de las cancelaciones de turnos por parte de afiliados o profesionales
- **Tipo_Cancelacion:** Entidad que contiene la información específica de los tipos de cancelaciones existentes en el sistema

Procedures:

- **Alta_Afiliado_Familiar:** Dar de alta un familiar de un afiliado
- **Alta_Afiliado_Titular:** Dar de alta un afiliado titular
- **Alta_Rol:** Dar de alta un rol
- **AumentarIntentosFallidos:** Aumentar los intentos fallidos de login de un usuario al sistema
- **Baja_Rol:** Baja logica de un rol
- **Baja_Usuario:** Baja logica de un usuario
- **Borrar_Funcionalidades_Rol:** Desasignar todas las funcionalidades de un rol específico.
- **Cambiar_Nombre_Rol:** Cambiar el nombre de un rol
- **Cancelar_Dias:** Cancela para un profesional todos los turnos que posee y coloca como cancelado su agenda.
- **Cancelar_Turno:** Cancela un turno específico para un afiliado.
- **ComprarBonos:** Registrar el evento de compra de un bono e insertar cada uno de los bonos creados en la tabla correspondiente.
- **CrearDiaAgenda:** Crea un día en la agenda del medico
- **Create_Indexes:** Crea los indices necesarios para cada tabla
- **Create_Tables_And_Fill:** Crea las tablas del sistema
- **Generar_Consulta:** Crea una consulta en la tabla para el turno especificado
- **Grabar_Turno:** Graba un turno solicitado por un afiliado.
- **Guardar_Consulta:** Se agregan a la consulta específica la hora de atención, los síntomas y diagnóstico del medico.
- **Habilitar_Rol:** Alta lógica de un rol.

- **InsertFuncionalidad:** Alta de una funcionalidad del sistema.
- **LimpiarIntentos:** Limpiar los intentos de login de un usuario
- **Load_Bonos:** Cargar los bonos de la tabla maestra a la base de datos normalizada.
- **Load_Cancelaciones:** Carga los tipos de cancelaciones
- **Load_Consultas_Turnos:** Carga los turnos y consultas de la tabla maestra a la base de datos normalizada
- **Load_Especialidades:** Cargar las especialidades de la tabla maestra a la base de datos normalizada.
- **Load_Estados_Civiles:** Cargar los estados civiles de los afiliados.
- **Load_Planes:** Cargar los planes de la tabla maestra a la base de datos normalizada.
- **Load_Roles_Y_Funcionalidades :** Cargar los roles y funcionalidades.
- **Load_Users:** Cargar los usuarios afiliados y profesionales de la tabla maestra a la base de datos normalizada.
- **Modificacion_Roles:** Modificación de roles
- **Modificar_Afiliado:** Modificación de afiliado.
- **Set_Admin:** Carga el usuario admin al sistema

Funciones:

- **GetUser:** Obtiene el usuario que se quiere loguear

Decisiones de diseño:

- Se considera que todos los usuarios de la tabla maestra no tienen familiares a cargo.
- Como no están las matrículas de los profesionales en la tabla maestra se decidió que dejar la matrícula de los profesionales migrados en NULL
- Como no está especificado el sexo de los usuarios en la tabla maestra, se dejó en NULL
- Se persistirán los turnos cancelados en la tabla Cancelaciones mientras que la agenda tendrá un campo cancelado para mostrar su baja.

Notas sobre el uso de la aplicación:

1. **ABM Afiliado:** Se utilizan los filtros para buscar los afiliados existentes, en caso de querer modificar alguno, hacer click en alguna de las columnas del usuario y se abra el form de modificación. Al crear un nuevo usuario, si se indica que tiene familiares, al terminar de crearlo, se debera crear al menos uno de los familiares.
2. **Agenda:** Se deberá ingresar primero el id del profesional para ingresar. Luego dentro se muestra la agenda para dicho médico y se pueden agregar para los días seleccionados y el rango horario la agenda en el rango de fechas que se coloque.
3. **Pedir Turno:** Se ingresa el ID del profesional y su especialidad y el id del afiliado al cual se le quiere asignar el turno. Se muestran los turnos disponibles para ese Profesional y seleccionando un turno disponible se lo puede anotar para el mismo.
4. **Cancelar Turno:** Se debe seleccionar el profesional o el afiliado a cual se le quiere cancelar. Si es el profesional deberá seleccionarse el rango de días a cancelar y para el afiliado seleccionar el turno a cancelar. Deberá seleccionarse el tipo de motivo y la explicación para poder cumplir con el proceso.

Queries Listado Estadístico:

- Especialidades con más cancelaciones:

```
SELECT descripcion from SIEGFRIED.ESPECIALIDADES where  
id_especialidad IN (  
    SELECT TOP 5 a.id_especialidad  
    from SIEGFRIED.AGENDA a  
    where a.cancelado = 1  
    AND YEAR(a.dia_hora) = @anio AND  
    MONTH(a.dia_hora) = @mes  
    GROUP BY a.id_especialidad  
    ORDER BY COUNT(a.id_turno) desc)
```

- Profesionales más consultados:

```
SELECT TOP 5 usuario.nombre+' '+usuario.apellido as medico,
plann.descripcion, especialidad.descripcion, count(*) as cantidad_consultas
FROM (
SELECT bono.id_plan as planid, agenda.id_profesional as
profesional,agenda.id_especialidad as especialidad FROM
SIEGFRIED.BONOS bono
LEFT JOIN SIEGFRIED.CONSULTAS consulta on bono.id_consulta =
consulta.id_consulta
LEFT JOIN SIEGFRIED.TURNOS turno on turno.id_turno = consulta.id_turno
LEFT JOIN SIEGFRIED.AGENDA agenda on turno.id_turno =
agenda.id_turno
WHERE bono.id_consulta is not null
AND YEAR(agenda.dia_hora) = @anio AND MONTH(agenda.dia_hora) =
@mes AND bono.id_plan = @plan
) as vista
LEFT JOIN SIEGFRIED.USUARIOS usuario ON vista.profesional =
usuario.id_usuario
LEFT JOIN SIEGFRIED.PLANES plann on plann.id_plan = vista.planid
LEFT JOIN SIEGFRIED.ESPECIALIDADES especialidad on
especialidad.id_especialidad = vista.especialidad
GROUP BY plann.descripcion, usuario.nombre+' '+usuario.apellido,
especialidad.descripcion ORDER BY 3 DESC
```

- Profesionales que menos trabajaron:

```
SELECT TOP 5 usuario.nombre+' '+usuario.apellido as
medico,especialidad.descripcion,count(*) / 2 as horas_trabajadas
FROM (
SELECT agenda.id_profesional as profesional,agenda.id_especialidad as
especialidad
FROM SIEGFRIED.BONOS bono LEFT JOIN SIEGFRIED.CONSULTAS
consulta on bono.id_consulta = consulta.id_consulta LEFT JOIN
SIEGFRIED.TURNOS turno on turno.id_turno = consulta.id_turno LEFT JOIN
SIEGFRIED.AGENDA agenda on turno.id_turno = agenda.id_turno
WHERE bono.id_consulta is not null
AND YEAR(agenda.dia_hora) = @anio AND MONTH(agenda.dia_hora) =
@mes AND bono.id_plan = @plan AND agenda.id_especialidad =
@especialidad ) as vista
LEFT JOIN SIEGFRIED.USUARIOS usuario ON vista.profesional =
usuario.id_usuario
```

```

LEFT JOIN SIEGFRIED.ESPECIALIDADES especialidad on
especialidad.id_especialidad = vista.especialidad
GROUP BY usuario.nombre+' '+usuario.apellido, especialidad.descripcion
ORDER BY 3 ASC

```

- Afiliados con mas bonos comprados:

```

SELECT TOP 5 usuario.nombre + ' ' + usuario.apellido, COUNT(*), CASE
WHEN bono.id_afiliado % 10 = 1 THEN 'No' ELSE 'Si' END as
es_de_grupo_familiar
FROM SIEGFRIED.BONOS bono
LEFT JOIN SIEGFRIED.USUARIOS usuario on bono.id_afiliado =
usuario.id_usuario
WHERE YEAR(bono.fecha_compra) = @anio AND
MONTH(bono.fecha_compra) = @mes
GROUP BY usuario.nombre + ' ' + usuario.apellido, bono.id_afiliado ORDER
BY 2 DESC

```

- Especialidades con mas bonos usados:

```

SELECT especialidad.descripcion, vista.cantidad_consultas FROM (
SELECT TOP 5 bono.id_bono as id_bono, agenda.id_especialidad as
id_especialidad, COUNT(*) as cantidad_consultas
FROM SIEGFRIED.BONOS bono LEFT JOIN SIEGFRIED.CONULTAS
consulta on bono.id_consulta = consulta.id_consulta LEFT JOIN
SIEGFRIED.TURNOS turno on turno.id_turno = consulta.id_turno LEFT JOIN
SIEGFRIED.AGENDA agenda on turno.id_turno = agenda.id_turno
WHERE bono.id_consulta is not null AND YEAR(agenda.dia_hora) = @anio
AND MONTH(agenda.dia_hora) = @mes
GROUP BY bono.id_bono, agenda.id_especialidad ORDER BY COUNT(*)
DESC ) vista
left join siegfried.especialidades especialidad on especialidad.id_especialidad
= vista.id_especialidad

```