**Sistema Artec**

**Resumen de actividades desarrolladas**

1. Lenguaje de programación:C#
2. Desarrollo en 5 capas: Entidades, BLL, DAL, GUI y Servicios
3. Acceso único a la Base de Datos a través de la capa de servicios
4. La clase de acceso a la BD es reutilizable y permite a otras capas iniciar y finalizar una transacción
5. Mapeador resuelve mediante reflection
6. Servicios:
   * Persistencia y recupero de datos
   * Backup y Restore BD
   * Idioma
   * Login implementando patrón Singleton
   * Encriptación y aplicación de hash
   * (Bitácora)
   * (Digitos Verificadores)
   * (Manejo de excepciones)
   * (Autorización mediante Familia Patente)
7. Interfaces y lógica programada:
   * Login (Autenticación)
   * Crear Solicitud
   * Backup y Restore
   * Crear Proveedor

Descartes:

26/06/2017

1. Las compras de software son únicamente de licencias permanentes, porque las que son por suscripción requieren una erogación de gastos continua, y eso conlleva a la elevación del pedido hacia la procuradora y el trámite de adquisición lo realiza el área de administración (se iría a una cuasi Licitación)
   * Eliminé fechasus y fehafinsusc de inventario
   * Eliminé IdTipoLicencia de Bien
   * Elimine relación Bien con Tipolicencia
   * Elimine Tabla TipoLicencia
2. Adquisicion
   * Elimine NroExpediente de Adquisicion
   * Eliminé NroOrdenCompra de Adquisicion
3. Tabla AsigDetalle
   * Eliminé TiempoAsignación de AsigDetalle porque no se contemplan los prestamos
4. Tabla TipoAsignación, la eliminé porque solo hay un tipo de asignación (ya que no se contemplan prestamos)
   * Eliminé IdTipoAsignacion en AsigDetalle
5. Idioma
   * Eliminé la columna IdFormulario de Etiqueta
   * Modifique la clave primaria en etiqueta quitando de la misma IdFormulario
   * Eliminé la tabla Formulario
6. Store [dbo].[CotizacionTraerPorSolicitud]
   * Modifiqué el select

Estaba: --select Coti.IdCotizacion, Coti.MontoCotizado, Coti.FechaCotizacion, Prov.IdProveedor, Prov.AliasProv, Coti.IdPartidaDetalle, Coti.IdPartida, Det.IdSolicitudDetalle, Det.IdSolicitud

Quedó: select Coti.IdCotizacion, Coti.MontoCotizado, Coti.FechaCotizacion, Prov.IdProveedor, Prov.AliasProv, Det.IdSolicitudDetalle, Det.IdSolicitud

1. Porque pinchaba cuando quería agregar una cotización a un detalle que tenía cero cotizaciones:
   * Agregue lo resaltado en frmsolicitudmodificar:

frmCotizaciones UnFrmCotizaciones = new frmCotizaciones(unaSolicitud.unosDetallesSolicitud[e.RowIndex].unasCotizaciones, unDetSolic);

* + Agregue lo resaltado (y quite lo comentado) en frmCotizaciones:

unaCotiz.unDetalleAsociado.IdSolicitud = unDetSolic.IdSolicitud;//unasCotizaciones[0].unDetalleAsociado.IdSolicitud;

unaCotiz.unDetalleAsociado.IdSolicitudDetalle = unDetSolic.IdSolicitudDetalle;//unasCotizaciones[0].unDetalleAsociado.IdSolicitudDetalle;

1. Se eliminó table TipoAdquisicion y referencia desde Adquisición
   * Modifique AdquisicionCrear en consecuencia
2. Agregue campo IdPartidaDetalle en tabla Inventario, para poder relacionar el inventario, con la compra y la solicitud
3. Agregar prop Bien en categoría para poder acceder al tipo de bien desde el detalle de solic, creo q ya no}
4. Eliminar IdAdquisicion de tabla PartidaDetalle (posiblemente no lo use nunca)
5. Quitar el agente relacionado al software
6. Quitar la tabla depósito, porque no hay varios depósitos, además se compra lo solicitado y se lo entrega, no se stockea, lo de consultar stock era para cuando se entrega en forma parcial algo
7. IdInventario quedo como pk de Inventario y hay un índice unique entre IdBienEspecif y SerieKey en Inventario
8. Elimine columna IdBien en AsigDetalle, porque la pk de inventario ya no contiene a IdBien
9. Quitar el campo Observacion de AsigDetalle (poner solo uno en asignación quizas)
10. ~~//FALTA QUE Ponga en estado "Compradas" las cosas de un SolicDetalle~~

~~//Para hacerlo se puede consultar de nuevo la cantidad comprada y comparar con la cantidad solicitada~~

~~//Todo dentro del negocio o dal.. El código donde empzaria esto esa en frmregistrar línea 66~~

1. Al registrar bienes, solo me tiene que permitir registrar los de un mismo SolicDetalle (sino el proveedor no concuerda con todos los datalles)
2. Seguir en DALAsignacion línea 58, lo de poner como finalizado un solicDetalle
3. ~~Al registrar bienes, todos aparecen con el mismo numero de serie, REVISAR~~
4. Fijarme en DALAsignacion línea 59 que con el METODO CIERRO LA CONEXION A LA BD Y DPS EN LA SEGUNDA VUELTA DEL FOR EACH PINCHA PORQ ESTA CERRADA LA BD
5. Agregue esta línea al final del store “InventarioHardTraerListosParaAsignar” para que traiga solo los inventarios que aun no fueron entregados (sino traía todos los inventarios que están dentro de un SolicDetalle que no está finalizado aún):
   * and inv.IdEstadoInventario = 1--Disponible
6. ~~Evlauar quitar la línea “and sdet.IdEstadoSolicDetalle = @IdEstadoSolicDetalle—Adquirido”, del store “InventarioHardTraerListosParaAsignar” para permitir asignar bienes sin que todos los inventarios de un mismo SolicDetalle hayan sido adquiridos~~
7. Al registrar un inventario me vuelve a hacer update del estadoSolicDetalle a “Adquirido” en caso de que se haya comprado todo (y aunque ya esté finalizado ese SolicDetalle porq se entregó todo), fijarme en frmRegistrar en el load, para que solo cargue los bienes que restan por entregar y no todos como está ahora q te los muestra y te pone la cantidad comprada

GUARDA:Para arreglarlo modifique mal, pero arece q sigue andando todo, la línea 114 de frmbienregistrar

Antes estaba esto: LisAUXDetalles = unosDetallesBienes.Select(x => new HLPDetallesAdquisicion() { DescripCategoria = x.unaCategoria.DescripCategoria, Cantidad = x.Cantidad, IdCategoria = x.unaCategoria.IdCategoria, IdSolicitudDetalle = x.IdSolicitudDetalle }).ToList();

EN LINEA 26 BLLAsignacion, poner que solo lo haga cuando ..asdf.asd



En store CategoriaDetBienesTraerPorIdPartida, modifique la línea:

--AND SolDet.IdEstadoSolicDetalle != 2--Distinto de Finalizado

Hacia:

AND sdet.IdEstadoSolicDetalle < 3--Que el estado del detalle no sea Adquirido ni Entregado

1. Quite relacion entre partidaDetalle y Adquisicion y eliminé campo IdAquisicion de tabla PartidaDetalle

Agregué tabla RelPDetAdq para tener un N a N entre PartidaDetalle y Adquisicion

1. .Agregué campo Costo en tabla Adquisicion
2. Asociar eventos a las funciones en solicmodificar y a lo último poner todas las validaciones comentadas (esta en proceso)
3. Empezar con el agregardetalle en modificarsolic
4. Al eliminar detalles el IdDetalle queda mal (en ModificarSolic)
5. Cargar el objeto que contiene el bien seleccionado, al hacer click en un detalle

31.1 Usar el store “InventariosTraerListosParaAsignar” para restringir que no se puedan asignar bienes sino hay inventarios listos para asignar

1. Agregar inv.costo en consulta “InventariosTraerListosParaAsignar”
2. No me muestra los agentes al seleccionar un software en detallessolic
   * Revisar store “SolicDetallesTraerPorNroSolicitud”, el problema viene en que en el load de frmsolicmodificar no cargo los agentesasociados al detalle que tiene un software

**EL PROBLEMA ESTABA PRINCIPALMENTE EN QUE NO GUARDO LOS AGENTES EN la tabla RelSolDetalleAgente. RESOLVER ESTO, lo hice en la línea 733**

* + **Agregue el store SolicDetallesTraerAgentesAsociados para traer los agentes dentro de unaSolicitud.unosDetallesSolicitud[LaCorrespondiente].unosAgentes cuando hago click en un detalle y es un software**
  + **Agregue el store “RelSolDetalleAgenteAgregar” para poder guardar los agentes asociados a un software**
  + **Ya asocia un nuevo agente a un detalle que ya tiene el software (yendo por el agregar nuevo detalle, falta verificar yendo por el de modificar). El problema ahora está en que al hacer click en el detalle del software, no me muestra todos los agentes, es decir solo me muestra los q ya estaban y no los q inclui recién.**
  + **Ahora no muestra los agentes al hacer click por primera vez (sin agregar un agente antes) en un detalle de software (Aclaro que si agrego un soft, al menos desde nuevodetalle, me muestra los agentes asociados correctamente). SOLUCION: Lo solucione con las líneas 337-344**
  + **Al estar viendo el detalle de un software y ver los agentes asociados… y al agregarle un agente, me borra de la lista de la grilla los agentes ya asociados y coloca el ingresado recién, seguro q lo arreglo como los puntos anteriores**

**Solucion: Linea 343, agregue … = unosAgentesAsociados =… para que se guarde en esa lista los agentes asociados actuales**

1. **Arreglar las cosas en frmSolicCrear que fueron mejoradas en frmSolicModificar**
   * **Agregar soft, etc**
2. **Fijarme que la línea 267 de DALSolicitud funcione(lo del if)**
3. **Agregar el campo SolDetalleUID y ponerlo como PK de la tabla SolicDetalle; esto es para poder modificar y eliminar solicDetalles y que no se me generen inconsistencias.**
   * **Corregir todo en el programa**
   * **Corregir todo en los store**
4. **Mejorar el servicio Mail, para que los datos estén en la BD y encriptado o con hash la contraseña del remitente**
5. **Servicio Mail:** 
   * **Cargar los datos del host al iniciar el sistema (en el login), ahora está en el evento btnSolicitar de cotizaciones**
   * **Se coloco en bd la dirección de mail general para enviar los mails**
   * **Colocar en el destinatario el mail del proveedor seleccionado, etc**
   * **Colocar en el frm de solicitar cotizaciones el textbox con el cuerpo del msj pre-seteado para que se pueda modificar si así se quisiese**
   * **Puse un thread para que al enviar mails se pueda cerrar la ventana de las cotizaciones y no se bloquee ahí**
6. **Corregir Idioma para que las consultas a la bd se hagan desde DAL.Servicios.ServicioIdioma**
   * **Quedaría Framework.servicios.ServicioIdioma con lo mismo, pero quitando la parte de consultar a la bd. Y quedaría “EtiquetasTraerTodosPorIdioma (int elIdioma)” dentro de dal.servicios.servicioIdioma, y desde ahí se llamaría a framework.servicios.Traducir()**
   * **CambiarIdioma cambiaría en que el idioma queda relacionado con el usuario, hay que crear un campo en el usuario para relacionar el ultimo idioma. Y se modificaría toda la función cambiarIdioma**
7. **Realizar el modificar idioma en frmPrincipal con:**

//Replico la traducción a TODOS LOS FORMULARIOS ABIERTOS ESTO HAY QUE USARLO AL CAMBIAR EL COMBOBOX DEL IDIOMA

ServicioIdioma.CambiarIdioma(this.FindForm(), (int)cboIdioma.SelectedValue);

foreach (Control unForm in Application.OpenForms)

{

ServicioIdioma.Traducir(unForm, ServicioLogin.GetLoginUnico().UsuarioLogueado.IdiomaUsuarioActual);

}

1. Antes cambiar en bd el idioma actual del empleado: Esto se hace en la función cambiarIdioma dentro de ServicioIdioma
2. **Realizar el cbobox en Principal para modiicar el idioma por default, la función ya está hecha en ServicioIdioma**
3. **Tener cuidado con el brake línea 57 en ServicioIdioma, lo puse para que corte al encontrar coincidencia entre el nombre del control y la etiqueta. Funciona bien**
4. **frmPartidaSolicitar, empezado para que funcione desde Principal directamente; ya busca las solicitudes, hay q hacer que cargue los detalles y cotizaciones; en línea 535 frmpartidasolicitar hay que cargar los detalles dentro de unaSolicitud, porq en dicha variable no están esos datos.**

**Hay que revisar la carga de las cotizaciones, porque pincha al hacer click en un detalle (muy posiblemente sesa el orden de las columnas dentro de la grilla porq tira una excepci´n que dice algo así) Hecho, pero probar si todo funciona bien**

You need to create an event handler for the user control that is raised when an event from within the user control is fired. This will allow you to bubble the event up the chain so you can handle the event from the form.

When clicking Button1 on the UserControl, i'll fire Button1\_Click which triggers UserControl\_ButtonClick on the form:

User control:

public event EventHandler ButtonClick;

protected void Button1\_Click(object sender, EventArgs e)

{

//bubble the event up to the parent

if (this.ButtonClick!= null)

this.ButtonClick(this, e);

}

Form:

UserControl1.ButtonClick += new EventHandler(UserControl\_ButtonClick);

protected void UserControl\_ButtonClick(object sender, EventArgs e)

{

//handle the event

}

O sino::

***User Control***

public event EventHandler StatusUpdated;

private void FunctionThatRaisesEvent()

{

//Null check makes sure the main page is attached to the event

if (this.StatusUpdated != null)

this.StatusUpdated(this, new EventArgs());

}

***Main Page/Form***

public void MyApp()

{

//USERCONTROL = your control with the StatusUpdated event

this.USERCONTROL.StatusUpdated += new EventHandler(MyEventHandlerFunction\_StatusUpdated);

}

public void MyEventHandlerFunction\_StatusUpdated(object sender, EventArgs e)

{

//your code here

}