



Sistemas Informáticos

Curso 2007-08

Diseño e Implementación de un Sistema
de Control de la Construcción de una
Instalación Industrial

M^a del Mar Gutiérrez Fernández
César Marqués Collado
Raquel Rodríguez Casero

Dirigido por:
Prof. Yolanda García Ruiz

Facultad de Informática
Universidad Complutense de Madrid

INDICE

<i>Palabras clave</i>	<i>1</i>
<i>Resumen</i>	<i>2</i>
<i>Summary</i>	<i>3</i>
1 <i>Introducción.</i>	4
1.1 Marco / Actividad Empresarial	4
1.2 Objetivo del Sistema	5
2 <i>Ingeniería</i>	7
2.1 Fase de Especificación de Requisitos	7
2.2 Fase de Construcción del Software	20
3 <i>Ampliaciones y Conclusiones</i>	54
3.1 Sistema Gestor de Base de Datos	54
3.2 Visual Basic 6	55
3.3 Visualizador 3D	55
3.4 Integración con herramientas de planificación de proyectos	56
4 <i>Bibliografía</i>	57
Libros	57
Páginas Web	57
<i>Anexo A. Ejemplo de Informe de Avance</i>	<i>58</i>
1 <i>Informe de avance en Kg</i>	<i>58</i>
<i>Anexo B. Ejemplo de Isométrica</i>	<i>59</i>
<i>Anexo C. Manual de Usuario</i>	<i>60</i>
1 <i>Introducción</i>	<i>60</i>
2 <i>Consideraciones previas</i>	<i>61</i>
2.1 Estructura de carpetas	61
2.2 Configuración regional	62
3 <i>Menú Principal</i>	<i>63</i>
4 <i>Menú Administración</i>	<i>65</i>
4.1 Gestión de Usuarios	65
4.2 Configuración del Proyecto	68
4.3 Borrar Vales	70
4.4 Borrar Isométrica	71
5 <i>Menú Taller</i>	<i>72</i>

5.1	Marcar Realizado	72
6	<i>Menú Herramientas</i>	80
6.1	Importar Isométricas	80
6.2	Importar Vale de Salida	81
6.3	Importar Vales de Forma Masiva	82
6.4	Gestión de cargas de Isométricas	83
6.5	Configurar Colores	83
7	<i>Menú Dir. Construcción</i>	84
7.1	Costes	84
7.2	Informes	86

Palabras clave

Navisworks
Microsoft Access
Planta Industrial
Isométrica
Construcción
Maqueta 3D
ADO (ActiveX Data Objects)
UML (Unified Modeling Language)
PDS (Plant Design System)
PDMS (Plant Design Management System)

Resumen

El proyecto consiste en realizar el desarrollo de un Sistema que permita calcular el porcentaje de avance de la Construcción de una Instalación Industrial. Para lograr este objetivo, es necesario desarrollar también una herramienta gráfica que, apoyada en un visualizador 3D (**Navisworks JetStreamv5**), permita al usuario seleccionar los objetos en la maqueta 3D que hayan sido construidos, de forma que se sustituya así el método manual que se sigue actualmente.

Para ello, ha sido necesario estudiar en detalle los flujos de trabajo entre la fase de Diseño y la fase de Construcción, que se llevan a cabo a lo largo de un proyecto industrial, con el fin de determinar qué información resulta necesaria para el sistema y establecer así las interfaces externas entre el sistema a desarrollar y las herramientas existentes.

Con el objetivo de facilitar el proceso de selección de los objetos en la maqueta 3D, se ha optado por implementar una integración con la herramienta de visualización elegida, de manera que desde el sistema se puedan utilizar los comandos propios de dicho visualizador. Por este motivo, ha sido necesario realizar un análisis de la interfaz de programación (API) de **Navisworks JetStreamv5** con el fin de poder utilizar sus comandos desde el nuevo Sistema.

Para realizar el desarrollo se ha utilizado como lenguaje de programación **Visual Basic 6.0**, debido a que la interfaz de programación (API) de los visualizadores 3D está implementada en este lenguaje. El sistema de base de datos elegido para almacenar la información necesaria es **Microsoft Access**. La elección del gestor de la base de datos ha estado motivada por un lado, por el lenguaje de programación utilizado y, por otro lado, por la facilidad de manejo y mantenimiento que ofrece, teniendo en cuenta que se utilizará en Obra, donde no siempre se cuenta con las mejores infraestructuras tecnológicas. El visualizador de la maqueta 3D es **Navisworks JetStreamv5**, que es una de las herramientas más extendidas en el mercado por permitir visualizar la mayoría de formatos de maquetas electrónicas 3D.

Summary

This Project consists in developing a System which allows calculating the progress percentage of the construction of an Industrial Facility. In order to achieve this goal, it is necessary to develop a graphic tool which, based on a 3D model viewing software (**Navisworks JetStreamv5**), allows the user to select the items of the 3D model which have been created. This is how the present manual method would be replaced.

This is why it has been necessary to study in great detail the work flows from the Design phase to the Construction phase, carried out along an industrial project, so that the information needed for the system can be determined and that it can be established the external interfaces between the system to be developed and the existing software tools.

With the aim of making the process of selecting the items of the 3D model easier, we have opted to implement integration with the chosen 3D model viewing tool, so that its own commands can be used. This is why it has been needed to carry out an analysis of the **Navisworks JetStreamv5** API, so that its commands can be used from the new System.

In order to execute the development, it has been **Visual Basic 6.0** the programming language used, since the Application Programming Interface (API) of the 3D model viewing software is implemented in this language. The database system chosen in order to store the needed data has been **Microsoft Access**. The election of the database manager has been brought about on the one hand by the programming language used, and, on the other hand, by its easiness in use and maintenance, taking into account that it will be used in the Field, where not the best technological infrastructures can always be counted on. 3D model viewing software is **Navisworks JetStreamv5**, which is one of the more extended tools in market, due to the fact that it allows visualising most of the formats of 3D electronic models.

1 Introducción.

1.1 Marco / Actividad Empresarial

El proyecto consiste en el desarrollo de una herramienta software que se ajusta a las necesidades de una empresa privada que desarrolla su actividad en el ámbito de la ingeniería y construcción de todo tipo de plantas industriales. A continuación, se explica con más detalle en qué consiste su área de trabajo y qué tipo de proyectos lleva a cabo.

TÉCNICAS REUNIDAS, TR, es un contratista general con actividad internacional que se dedica a la ingeniería, diseño y construcción de todo tipo de instalaciones industriales para un amplio espectro de clientes de todo el mundo que incluye muchas de las principales compañías petroleras estatales (national oil companies o NOCs en inglés) y multinacionales, además de grandes grupos españoles.

Desarrolla sus diseños en sistemas 3D mediante "maquetas electrónicas", que garantizan mayor precisión que las herramientas 2D convencionales, y eliminan problemas de inconsistencias entre las diferentes disciplinas, generando un producto de alta calidad con mayor valor añadido.

El siguiente cuadro ilustra cómo se clasifica la actividad empresarial que desarrolla en áreas de negocio perfectamente definidas:

Petróleo y Gas Natural		Generación de Energía	Infraestructuras e Industrias
<p>Refino y Petroquímica</p> <ul style="list-style-type: none"> • Refino • Petroquímica <p>Know how propio:</p> <ul style="list-style-type: none"> • Unidades de Refino • Transferencia de Calor • Fertilizantes y Ácido Nítrico 	<p>Producción de Crudo y Gas Natural</p> <ul style="list-style-type: none"> • Desarrollo de Campos de Petróleo y Gas (Onshore/Offshore) • GNL, Licuefacción, Regasificación y Tanques • Oleoductos-Gasoductos 	<ul style="list-style-type: none"> • Ciclos combinados • Nuclear • Térmica convencional • Renovables • Cogeneración 	<ul style="list-style-type: none"> • Tratamiento de Agua • Desalinización • Gestión de Residuos • Instalaciones de Transporte Aéreo • Transporte Terrestre • Puertos <p>Know how propio:</p> <ul style="list-style-type: none"> • Hidrometalurgia

En todos estos sectores, TR ofrece a sus clientes todo tipo de servicios de ingeniería y construcción de instalaciones industriales, que contempla desde estudios de viabilidad o ingenierías básicas y conceptuales, hasta la ejecución completa de grandes y complejos proyectos "llave en mano" incluyendo ingeniería y diseño, gestión de aprovisionamiento y entrega de equipos y materiales, y la construcción de instalaciones y otros servicios relacionados o vinculados, como la asistencia técnica, supervisión de construcción, dirección de obra, dirección facultativa, puesta en marcha, entrenamiento, etc.

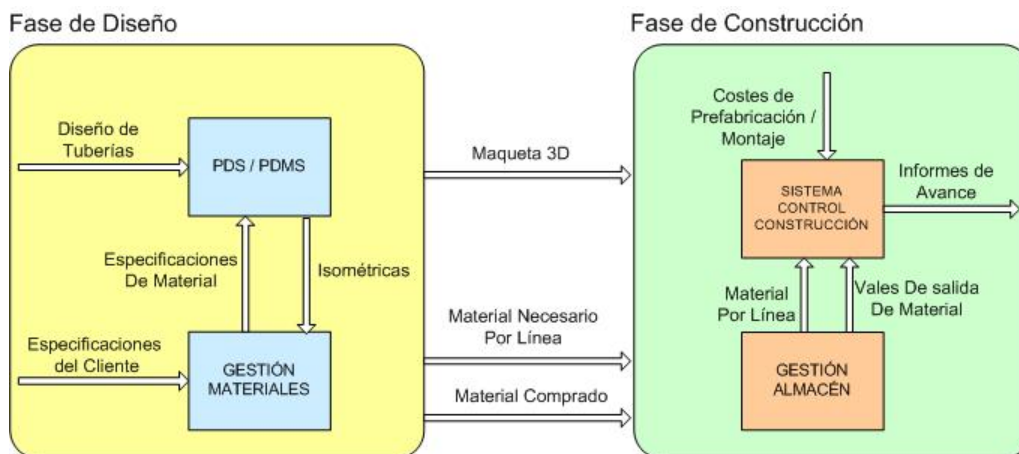
Desde su creación, las compañías del grupo TR han diseñado y construido más de 1000 plantas industriales.

Entre los principales clientes y licenciantes de TR figuran las primeras empresas del mundo, habiendo realizado proyectos en más de 50 países en los cinco continentes.

1.2 Objetivo del Sistema

El desarrollo del Sistema de Control de Construcción, en adelante SCC, se centra en la fase de construcción de una instalación industrial. Consiste en desarrollar una herramienta que permita controlar el avance de la obra de forma automatizada sustituyendo así el proceso manual que se sigue actualmente y que resulta lento e ineficiente. Para ello, es necesario disponer de la información que se genera en la fase de diseño con el fin de establecer una relación entre lo que se ha diseñado y lo que se ha construido.

En el siguiente gráfico se muestra la información generada en la fase de diseño y en la fase de construcción que es necesaria utilizar en el SCC para calcular el avance de la obra.



En la fase de diseño, se utilizan herramientas CAD 3D (PDS/PDMS) que permiten a los proyectistas construir una maqueta electrónica 3D de acuerdo a lo diseñado. En este tipo de sistemas, la información se organiza por disciplinas y, en este caso, la disciplina que se va a tratar y que resulta la más importante en este tipo de plantas es la de Tuberías. En esta disciplina, los datos se tratan por línea de tubería. Por cada una de las líneas de tubería existentes en la maqueta electrónica, se genera un plano constructivo que consiste en una vista isométrica de la línea diseñada y la lista de material necesario para su construcción. Este documento emitido para construcción se denomina **isométrica** (ver Anexo B).

Con el fin de optimizar la construcción de la planta, los materiales necesarios de cada línea se clasifican en materiales de **prefabricación**, que son aquellos que van soldados y es posible realizar su construcción en un taller, y de **montaje**, que son aquellos que se incorporan a la línea en la obra. El coste de prefabricación y de montaje es diferente y depende del diámetro de la tubería y del proyecto.

Por tanto, con la información generada en diseño: isométricas, material necesario por línea, y la maqueta electrónica; así como con la información generada por el sistema de control de almacén en Obra, el desarrollo del SCC deberá cumplir los siguientes objetivos:

- Establecer una interfaz con el sistema de gestión de materiales (MATUB) que permita importar la información necesaria correspondiente a isométricas, material necesario y vales de salida de material.
- Integrar las funcionalidades del visualizador 3D, que permita realizar el proceso de marcado, sobre la maqueta 3D, de los materiales que ya han sido prefabricados o montados, por cada una de las isométricas generadas en el proyecto y emitidas para construcción (IFC).
- Crear una herramienta que permita a la Dirección de Construcción, en adelante DC, incorporar los costes de prefabricación y montaje.
- Desarrollar una opción que permita generar los informes de avance de la construcción en términos de peso y coste económico.

Con esta finalidad, como ya se ha explicado, será necesario automatizar los procesos manuales que se siguen actualmente para obtener los informes de avance de forma inmediata.

2 Ingeniería

2.1 Fase de Especificación de Requisitos

2.1.1 Introducción

Mediante esta Especificación de Requisitos se pretende describir las funcionalidades que debe implementar un sistema que permita controlar el avance de la construcción de una planta industrial.

Hasta el momento, el control del avance de la construcción de una planta industrial se llevaba a cabo partiendo de las isométricas, generadas en papel, sobre las que el contratista iba marcando la parte que ya había sido prefabricada o montada. Teniendo en cuenta que una planta industrial de estas características puede haber del orden de diez mil isométricas, la transcripción manual de la información generada por el contratista a un fichero en formato electrónico resulta muy costosa tanto en tiempo como en términos económicos. Con este método de trabajo, resulta muy complicado llevar el control, por parte de la dirección de la construcción, de lo que realmente ha construido ya el contratista y es necesario, en muchos casos, trabajar con estimaciones con el riesgo que ello supone.

Por todo esto, lo que pretende el SCC es unificar el proceso de marcado de lo construido y el traspaso de esa información hacia un formato electrónico, de manera que el contratista cuente con una herramienta software con la que pueda realizar este proceso de forma directa. Para ello, se sustituirá el papel por la maqueta 3D, generada en la fase de diseño del proyecto industrial, y se implementará una base de datos asociada que permitirá almacenar, al mismo tiempo, toda la información necesaria para permitir a la DC generar los informes de avance más adecuados a las necesidades del proyecto.

Asimismo, será necesario facilitar la documentación de utilización necesaria para asegurar el correcto funcionamiento del sistema por los usuarios finales de la aplicación.

2.1.1.1 Objetivo

El objeto de esta Especificación de Requisitos es definir cuáles son las funcionalidades necesarias para una aplicación que permita controlar el avance de la Construcción de una instalación industrial.

2.1.1.2 Alcance

El sistema permitirá, partiendo de la información generada en la fase de diseño de la ingeniería de una planta industrial, obtener los informes necesarios sobre el avance de la fase de construcción de la planta.

Por tanto, es alcance del sistema desarrollar y documentar lo siguiente:

- La interfaz adecuada con la herramienta software que gestiona los materiales (MATUB) en la fase de diseño, con el fin de poder importar el recuento de material por línea.
- La interfaz adecuada con la herramienta software que gestiona los materiales (MATUB, módulo de Obra) en la fase de construcción, con el fin de poder importar los Vales de Salida de Material.
- Una interfaz gráfica de usuario que enlace la información del diseño con la maqueta 3D y permita al contratista marcar los elementos prefabricados o montados.
- Una herramienta que permita introducir los costes de prefabricación y montaje, y sea capaz de generar los informes de avance definidos por la DC.

No está dentro del alcance de este sistema realizar lo siguiente:

- Gestionar los recuentos de material de las líneas de tubería. Para este cometido se utilizará la herramienta habitual que, en este caso, es MATUB.
- Gestionar las compras del material necesario para llevar a cabo la construcción de la planta; tampoco es alcance del sistema gestionar el almacén de material en Obra. Para ello, se utilizará el módulo correspondiente del programa MATUB.

2.1.1.3 Acrónimos y términos

MATUB	Sistema de Gestión de Materiales de Tubería.
PDS	Plant Design System.
PDMS	Plant Design Management System.
SPR	Smart Plant Review.
NavisWorks	Visualizador de la maqueta 3D.
SCC	Sistema de Control de Construcción.
DC	Dirección de Construcción.
IFC	Emitido para Construcción.
API	Application Programming Interface.
Isométrica	Plano constructivo de una línea (o tramo) de tubería (Anexo B).

2.1.2 Descripción General

2.1.2.1 Perspectiva del producto

En los últimos años, se ha extendido la utilización de sistemas 3D para el diseño de plantas industriales debido a que permiten representaciones reales de las plantas diseñadas, de forma que se garantiza una mayor precisión y se eliminan problemas de inconsistencia entre las diferentes disciplinas.

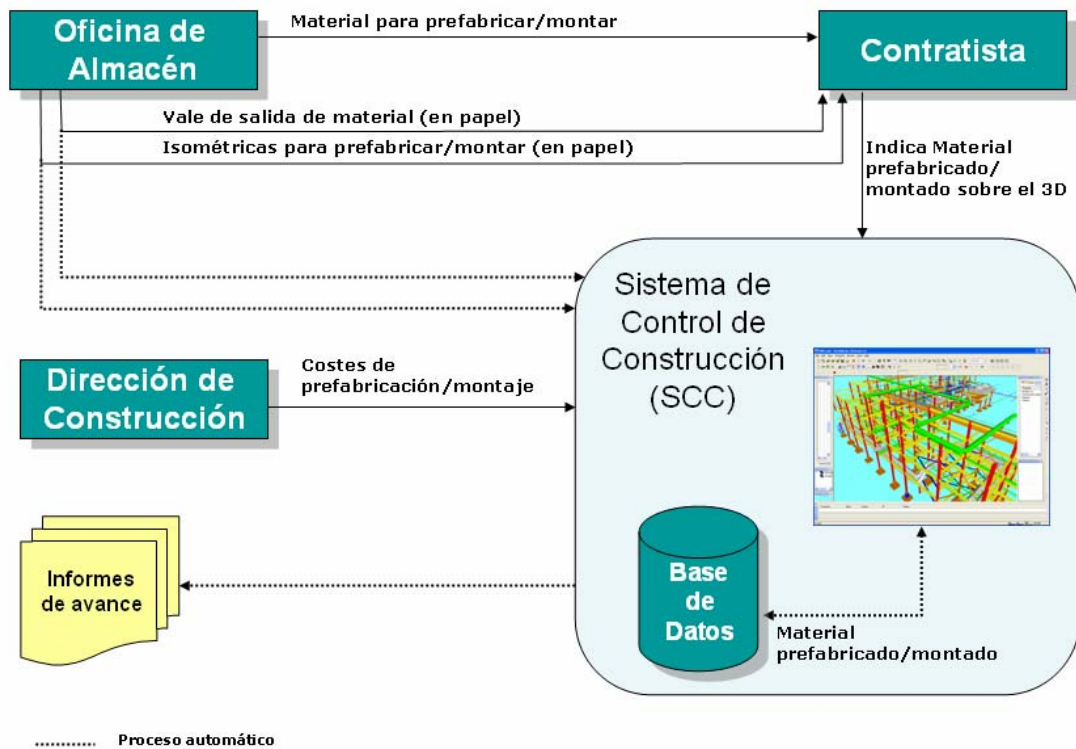
Con la utilización de estos nuevos sistemas surgen nuevas necesidades en el trabajo que facilitan la automatización de algunos procesos que en la actualidad se realizan de forma manual y que suponen una gestión de recursos ineficiente.

Los proyectos “llave en mano” que lleva a cabo TR suponen una gran inversión y, por consiguiente, un importante riesgo. En este tipo de proyectos, la fase de construcción resulta crucial por lo que es imprescindible gestionarla de forma eficiente. Una de las tareas que es necesario realizar en esta fase es la que corresponde al cálculo del avance de la obra. El desarrollo de esta herramienta plantea automatizar esta tarea sustituyendo el proceso manual, que se lleva a cabo en la actualidad, por la utilización del sistema SCC optimizando así los recursos y consiguiendo mayor precisión en el resultado.

En primer lugar, es necesario desarrollar una interfaz con el sistema de gestión de materiales en Obra (MATUB), en el que se almacena la información sobre el material necesario, por isométrica o línea de tubería, y el estado del almacén. La interfaz permitirá importar dicha información a nuestro sistema y controlar así el material necesario por línea y el material entregado al contratista por vale de salida.

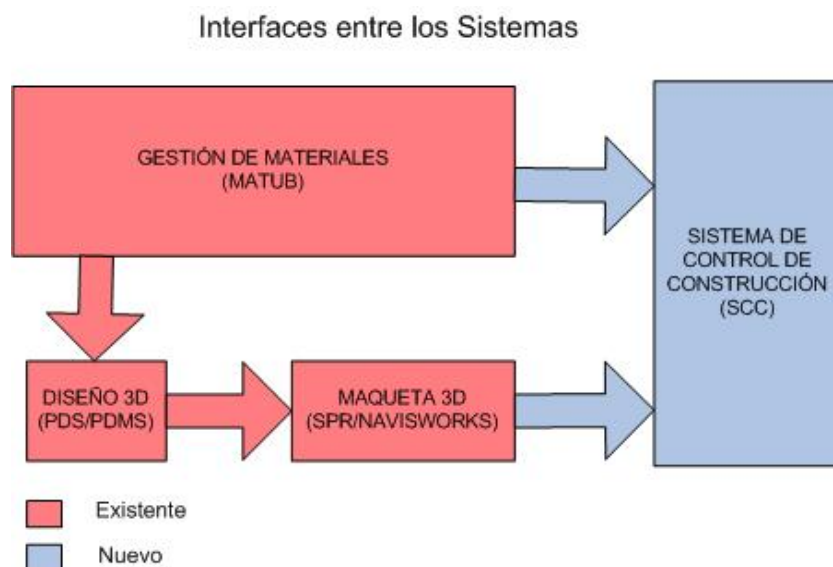
Partiendo de esta información como base, el contratista, desde el SCC, podrá seleccionar las isométricas emitidas para construcción (IFC), de las que se le ha entregado material para prefabricar o montar, sobre la maqueta 3D e indicar al mismo tiempo qué elementos de la isométrica están ya prefabricados o montados, según corresponda. Esta información será almacenada automáticamente en la base de datos asociada. Así, esta operación sustituye al proceso de marcado manual que el contratista lleva a cabo sobre el papel, para cada isométrica, que luego es necesario traspasar a un fichero electrónico (Excel) para poder obtener los listados deseados. Utilizando el nuevo sistema, la dirección de Construcción podrá obtener al instante, desde la base de datos, los informes requeridos acerca del avance de la construcción y del coste asociado.

El siguiente dibujo ilustra el flujo de información entre las entidades del sistema:



2.1.2.2 Interfaces del sistema

El sistema deberá implementar las interfaces correspondientes con MATUB para poder disponer de la información necesaria, y deberá ser capaz de visualizar la maqueta 3D permitiendo la interacción con el usuario, de forma que el sistema pueda leer los datos del 3D e interpretarlos adecuadamente con el fin de almacenarlos en la base de datos asociada al SCC.



2.1.2.2.1 Módulos del sistema

▪ Administración

En el módulo de Administración se implementarán las opciones restringidas al perfil de usuario de administrador del sistema. Las opciones corresponden a la gestión de los usuarios, así como aquellas opciones necesarias para la configuración de la herramienta con el fin de adaptarla a cada uno de los proyectos en los que se desee utilizar.

También se definirán funcionalidades que permitan acceder a la base de datos para realizar determinadas operaciones que el resto de usuarios tienen restringidas, como es el caso de eliminar datos en determinadas tablas.

▪ Interfaz con el Sistema de Gestión de Materiales en Obra

En este módulo se definirán dos interfaces con el Sistema de Gestión de Materiales que permitirán importar la información necesaria correspondiente a:

- Listado de material por isométrica o línea de tubería que haya sido emitidas como IFC.
- Vales de salida de material que se generan cuando desde el almacén se entregan materiales al contratista para construir isométricas.

Este módulo facilitará al usuario la opción de importar los datos anteriormente mencionados tantas veces como sea necesario de acuerdo al avance del diseño y a la llegada de materiales al almacén de la Obra.

▪ Módulo de Taller

En este módulo se definirán tanto las funcionalidades básicas de visualización que el usuario necesita para interactuar con la maqueta 3D como operaciones más avanzadas que le permitan realizar búsquedas sobre el 3D.

También en este módulo se implementarán las opciones que faciliten al usuario realizar el marcado de aquellos elementos de tubería que han sido prefabricados/montados por el contratista.

▪ Informes

Desde este módulo la DC será capaz de generar los informes de avance de la construcción que hayan sido definidos para el proyecto. Se generará un fichero en formato Excel al que el usuario podrá acceder para editarlo si fuera necesario.

También se definirá la interfaz gráfica de usuario que permita a la DC introducir los costes de prefabricación y montaje que permitan realizar el cálculo del coste del avance de la Construcción.

2.1.2.2 Interfaces Hardware

Los requisitos hardware mínimos que debe cumplir el equipo en el que se ejecute la aplicación para que el software funcione correctamente son los siguientes:

- PC o compatible Pentium 4 o superior.
- 1Gb de memoria RAM
- 1Gb de disco duro
- Tarjeta gráfica con 512 Mb de memoria RAM

2.1.2.3 Interfaces Software

La aplicación trabajará correctamente en un entorno que cumpla los siguientes requisitos:

- Sistema Operativo: Windows XP SP2.
- Base de datos: MS Access 2003.
- Informes de entrada: MS Word 2003.
- Informes de salida: MS Excel 2003.
- Visualizador maqueta 3D: Navisworks Jetstream v5.

El sistema será desarrollado utilizando el lenguaje de programación de Microsoft Visual Basic 6.0. Se ha elegido este entorno de programación debido, en gran parte, a que era necesario desarrollar una interfaz con el visualizador de la maqueta 3D, en este caso Navisworks, e integrar sus comandos en nuestro sistema. Para ello, era necesario utilizar la interfaz de programación (API) que proporciona Navisworks, que está implementada para Visual Basic. Además, la utilización de este lenguaje también facilita la tarea de trabajar con el paquete MS Office que se utilizará para generar informes, para cargar datos en el sistema y como base de datos para almacenar toda la información que requiera la aplicación. Una vez finalizado el desarrollo del sistema, se creará un paquete de distribución que permitirá instalarlo en las máquinas que se requiera, de forma que no será necesario tener instalado el entorno de desarrollo de Visual Basic para su correcta ejecución

El gestor de base de datos que se ha elegido es MS Access. Esta decisión está motivada debido a que se trata de un formato que resulta fácil de mantener, de transportar de un lugar a otro y de configurar. Es necesario tener en cuenta que el sistema se va a utilizar en Obras donde puede darse el caso de que las infraestructuras no permitan utilizar arquitecturas más complejas. En un futuro, podría plantearse el uso de un gestor de bases de datos más robusto y más consistente, como puede ser Oracle.

El visualizador de la maqueta electrónica 3D que se ha elegido, corresponde al que está siendo utilizado actualmente por la empresa.

2.1.2.3 Características de los usuarios

En este punto se van a definir los perfiles de usuario básicos que, por defecto, debe tener el sistema. Aparte de los perfiles aquí mencionados y con el fin de cubrir las necesidades que puedan surgir en el futuro, el sistema debe permitir crear nuevos perfiles y modificar las opciones a las que éstos tienen acceso.

La gestión de usuarios, por tanto, se realizará utilizando perfiles de forma que cada usuario que se cree estará asociado a un perfil que determinará las opciones a las que tendrá acceso dentro del sistema.

Los perfiles básicos que se han definido se detallan a continuación.

2.1.2.3.1 Administrador del sistema

El administrador del sistema se encargará de realizar la configuración inicial de la aplicación definiendo las opciones adecuadas a las particularidades del proyecto concreto: definición de usuarios y perfiles, carga de la información de inicialización del proyecto, configuración de la estructura de carpetas asociada, etc.

Este usuario deberá tener conocimientos avanzados de bases de datos (MS Access) ya que se encargará de la configuración de la base de datos del sistema así como de realizar las tareas de mantenimiento que sean necesarias.

2.1.2.3.2 Usuario del taller

El usuario del taller es el encargado de incorporar al sistema la información correspondiente a los elementos prefabricados/montados de cada una de las isométricas Emitidas Para Construcción (IFC) y entregadas al contratista.

Este usuario sólo tendrá acceso a las opciones de importar isométricas e importar vales de salida así como a la interfaz con el visualizador de la maqueta 3D que le permitirá realizar el marcado de elementos sobre la maqueta electrónica.

2.1.2.3.3 Usuario de DC

El usuario de Dirección de Construcción se encargará de introducir los costes de prefabricación y montaje correspondientes al proyecto que permitirán realizar el cálculo del avance del proyecto en coste.

Será el encargado de generar los informes de avance que ofrece el sistema. Con el fin de supervisar el trabajo que realiza el usuario del taller, también deberá tener acceso a las mismas opciones que dicho usuario.

2.1.3 Requisitos Específicos

2.1.3.1 Gestionar opciones de Proyecto

Cada Proyecto debe estar identificado por un código externo al sistema, así como por un nombre de proyecto, los cuales se podrán cargar o modificar por el Administrador del sistema tanto al comienzo como durante la vida del Proyecto.

El sistema debe poder establecer una imagen de fondo del menú principal personalizada para cada proyecto. El cambio de la cual solo debe ser accesible desde el administrador del sistema.

El sistema debe tener identificado un Cliente para el conjunto del Proyecto, estando accesible para las diversas opciones de la aplicación en modo consulta y en modo edición solo para el Administrador del Sistema.

El sistema deberá utilizar una maqueta en 3D en el formato de datos que utiliza el programa “NavisWorks JetStream”, de modo que se pueda ubicar en el Sistema la ruta en la que se encuentra dicha maqueta.

El sistema debe tener la opción de personalizar el color utilizado en la maqueta para marcar los elementos montados y prefabricados, esta opción de cambio de color debe estar disponible para todos los Usuarios del Sistema

2.1.3.2 Gestionar usuarios

El sistema debe utilizar un sistema de gestión de usuarios con claves, las cuales deben cumplir con la Ley de Protección de Datos.

El sistema gestionará los usuarios como miembros de grupos o perfiles a los que el Administrador podrá añadir o eliminar opciones de acceso.

Se contarán con tres perfiles predeterminados, pero se podrán modificar estos perfiles, añadir nuevos o eliminarlos en funciones de las necesidades del proyecto.

2.1.3.3 Importar Isométricas

Las Isométricas del Sistema se cargarán por medio de ficheros “.rtf” generados por el Sistema de Gestión de Materiales existente en la Compañía (MATUB). En el que cada fichero corresponderá a una Isométrica o a un tramo de una Isométrica.

El sistema debe tener un mecanismo para cargar vales iniciales de forma masiva o vales de salida individuales que se generan a lo largo del proyecto en los que se listan varias Isométricas.

El Formato de estos ficheros Isométricas será el siguiente:

Material de la Isométrica

Identificación de la Isométrica

LISTADO DE ELEMENTOS

ISOMETRICA: H01 A 000101 01 / 01 REV.00 FECHA: 11/07/07 ESPECIFICACION: 12CG0U TIPO: APC

CLAVE INPUT	DESCRIPCION	P/M	CANTIDAD NECESARIA	CANTIDAD DESPACHADA	CANTIDAD PENDIENTE	TOTAL	PESO (EN KG) DESPACHADO
00000378	GATE VAL VE 125# FE ASTM A39 3"	5 M	1,00	0,00	1,00	32,60	0,00
00000539	PIPE Sch . S TD API 5L Gr. B SM 1"	LS (GALV.) M	3,00	0,00	3,00	7,50	0,00

TOTALES

P E S O S		M E T R O S		DIAMETROS MEDIO	
CANTIDAD NECESARIA	: 784,71	TUBERIA NECESARIA	: 0,00	TUBERIA NECESARIA	: 0,00"
CANTIDAD DESPACHADA	: 0,00	TUBERIA DESPACHADA	: 0,00	TUBERIA DESPACHADA	: 0,00"
CANTIDAD PENDIENTE	: 784,71	TUBERIA PENDIENTE	: 0,00	TUBERIA PENDIENTE	: 0,00"
PORCENTAJE DESPACHADO	: 0,00%				

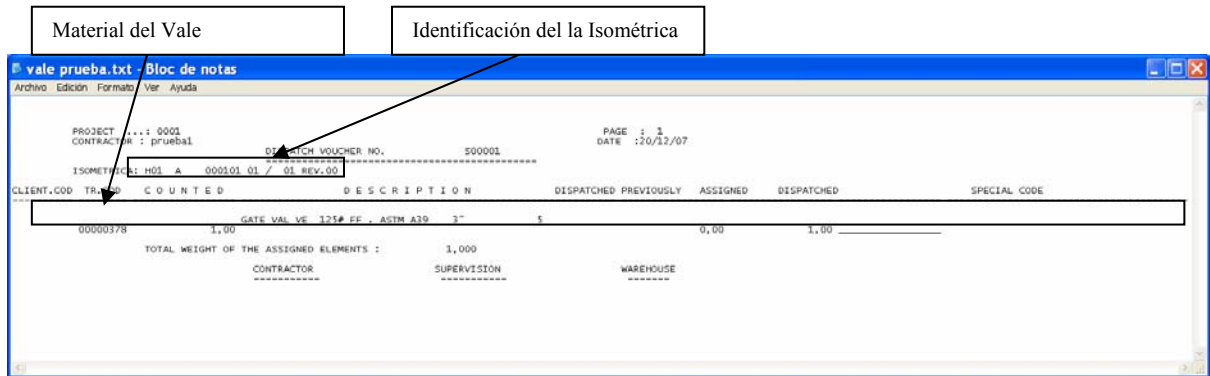
Resumen Final de la Isométrica

No se deberá mantener en el sistema la misma isométrica y en dos revisiones diferentes. En este caso, se deberá notificar y dejar constancia de la actualización de la Isométrica a la última revisión. Esta circunstancia se puede dar debido a un cambio en el diseño de una línea, las diferencias que afecten al recuento de material, que este hecho puede conllevar, serán debidamente controladas por el sistema de materiales en Obra y no por el SCC.

2.1.3.4 Importar Vales de Salida de Material

El Sistema deberá cargar los vales de Salida en el mismo formato en que se entregan a los contratistas. Estos vales se entregan en el formato de salida de MATUB.

El formato de estos ficheros es el siguiente:



No se podrán cargar Vales de Salida con material perteneciente a Isométricas que no hayan sido cargadas previamente en el SCC mediante la opción de importar Isométricas.

Cada Vale de Salida puede contener información de una o varias isométricas.

2.1.3.5 Marcar Realizado Taller

El Sistema debe implementar una interfaz adecuada con la maqueta 3D que permita al usuario realizar las operaciones básicas de visualización sobre el 3D como son: diferentes funciones de navegación, selección de elementos, ocultar elementos no seleccionados, zoom sobre objetos, búsqueda de líneas, etc.

Desde esta opción el usuario debe tener acceso a la información de los vales de salida de material entregados al contratista y debe poder visualizar las líneas incluidas en cada vale. Para cada una de las líneas se deberá mostrar el material asociado clarificándolo, según corresponda, de la siguiente forma:

- **Material no marcado:** lista del material necesario que aún no se ha sido marcado en la maqueta 3D por el contratista.
- **Material Entregado:** lista del material necesario que ha sido entregado al contratista en algún vale de salida.
- **Material Prefabricado:** lista del material de prefabricación, entregado en algún vale de salida, que ha sido marcado en la maqueta 3D.
- **Material Montado:** lista del material de montaje, entregado en algún vale de salida, que ha sido marcado en la maqueta 3D.

Se debe implementar una funcionalidad que permita al usuario resaltar sobre la maqueta 3D los elementos prefabricados, los elementos montados y, en general, dar la posibilidad de distinguir los elementos de las líneas que ya han sido construidos (prefabricado o montados) de los elementos que faltan por construir, bien porque aún no se haya entregado el material o bien porque el contratista aún no los haya construido.

En esta opción también se debe recoger la posibilidad de que el usuario pueda marcar elementos especiales que no se representan en la maqueta 3D, como es el caso de pernos y juntas, pero que es necesario tener en cuenta puesto que forman parte de la lista de material necesario para construir la línea y, por tanto, son necesarios para el cálculo del avance de la construcción.

De la misma forma que se debe permitir al usuario marcar elementos, también debe existir una funcionalidad que permita desmarcar elementos que hayan sido marcados por error o que sea necesario desmarcarlos por cualquier otro motivo.

2.1.3.6 Gestionar datos de costes

El sistema debe contener una opción de gestión de datos de coste en una interfaz gráfica de usuario que permita introducir los costes para poder realizar un cálculo del coste del avance de la construcción.

De esta manera el coste será independiente en cada proyecto y se almacenará el tipo: prefabricado o montaje, el diámetro y el coste.

2.1.3.7 Gestionar informes

El sistema debe contener un sistema de gestión de informes para poder controlar el avance de la construcción. Se generará un fichero en formato Excel al que el usuario podrá acceder para modificarlo si lo necesitara.

Habrán dos tipos de informe:

- Informe de avance por línea en kg
- Informe de avance por línea en coste

En cada uno se podrán elegir las unidades del proyecto que formarán parte de dicho informe.

El informe de avance por línea en kg se compondrá de una primera hoja que será un resumen que incluirá: peso necesario, peso construido, diferencia y avance de cada unidad seleccionada y un total de todas las unidades incluidas en el informe. Y en las hojas siguientes se detallará el avance por línea en kg de cada unidad.

En el informe de avance por línea en coste también se incluirá una hoja resumen que incluirá: cantidad necesaria, cantidad prefabricada o montada, diferencia y avance. Y en las hojas siguientes se detallará el avance por línea en coste de cada unidad.

En el Anexo A se adjunta un ejemplo del formato que deben cumplir los informes de avance. El formato será el mismo para los dos tipos de informes variando las unidades en que se muestra la información: kg o euros.

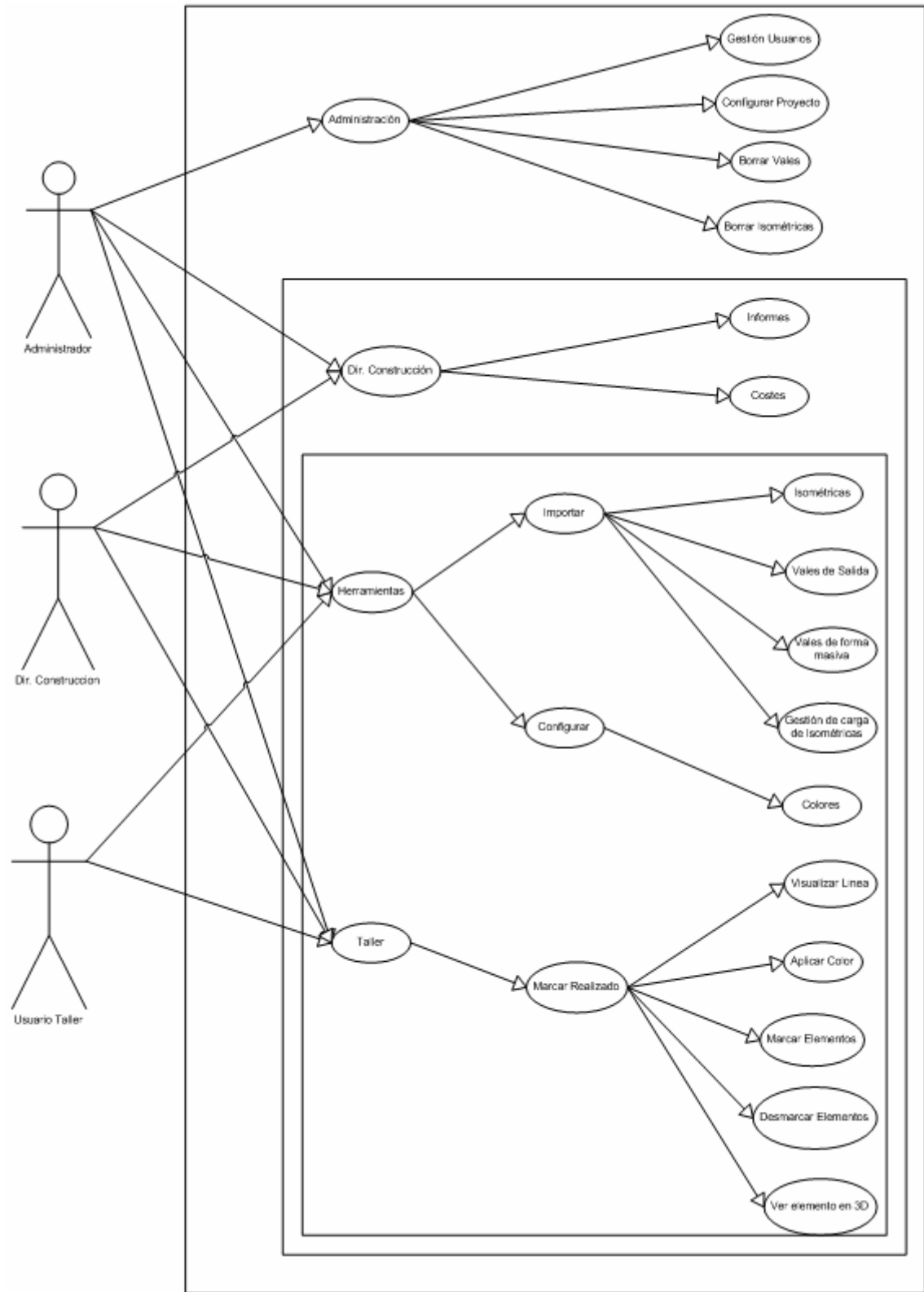
2.1.4 Definición de Casos de uso

El diagrama de Casos de uso muestra los perfiles básicos que han sido definidos y cada perfil determina las opciones a las que tendrá acceso en el sistema, de manera que cada usuario que se cree estará asociado a un perfil.

El administrador se encargará de realizar la configuración de la aplicación definiendo las opciones adecuadas para cada proyecto concreto: gestión de usuarios, configuración del proyecto, carga de la información inicial del proyecto... Y además tendrá acceso a todo el sistema.

El usuario del taller sólo tendrá acceso a las opciones de importar isométricas e importar vales de salida, así como a la interfaz con el visualizador de la maqueta 3D para realizar el marcado de elementos sobre la maqueta.

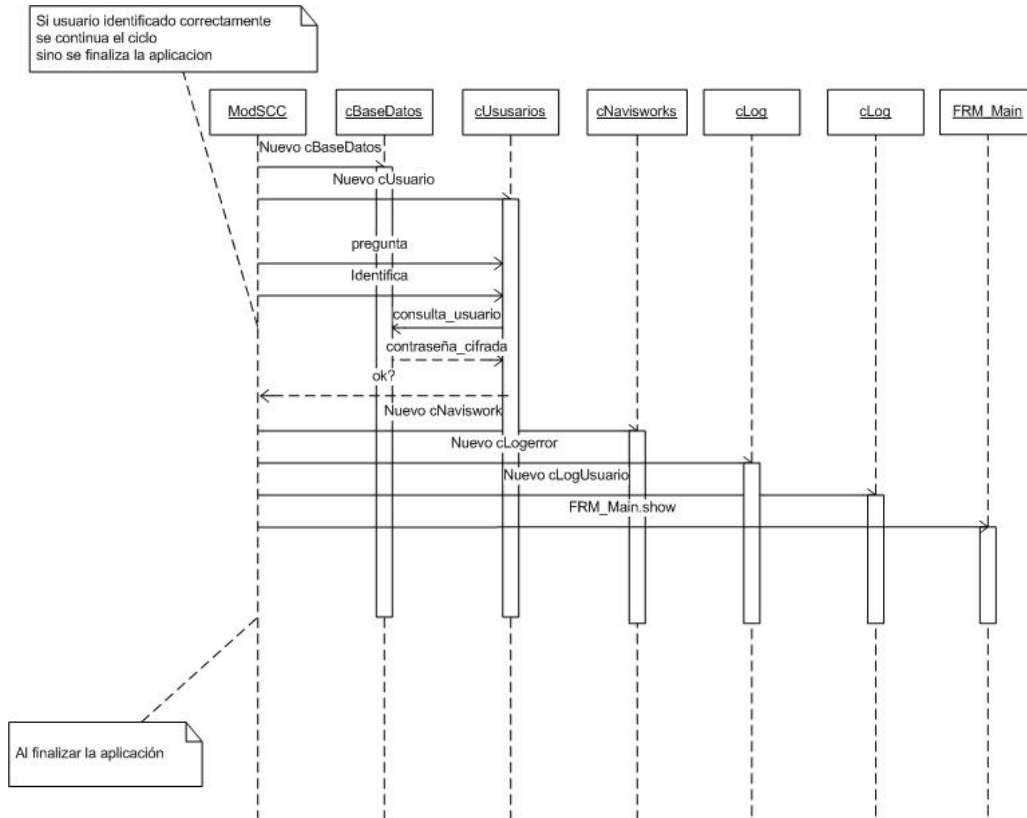
El usuario de Dirección de Construcción se encarga de introducir costes para realizar el cálculo de avance y poder generar los informes de avance que ofrece el sistema y también deberá tener acceso a las mismas opciones que el usuario taller.



2.2 Fase de Construcción del Software

2.2.1 Diagramas de Secuencia del Sistema

2.2.1.1 Inicio de la ejecución



Al inicial el Modulo principal (ModSCC) se realiza la creación de los objetos necesarios para la autenticación del usuario, estos son un objeto de la clase cUsuario y otro de la clase cBaseDatos.

Lo primero que se hace tras crear estos objetos es lanzar un formulario de captura de datos (FRM_AccesoUsuario) en el que se introduce el nombre del usuario y la contraseña.

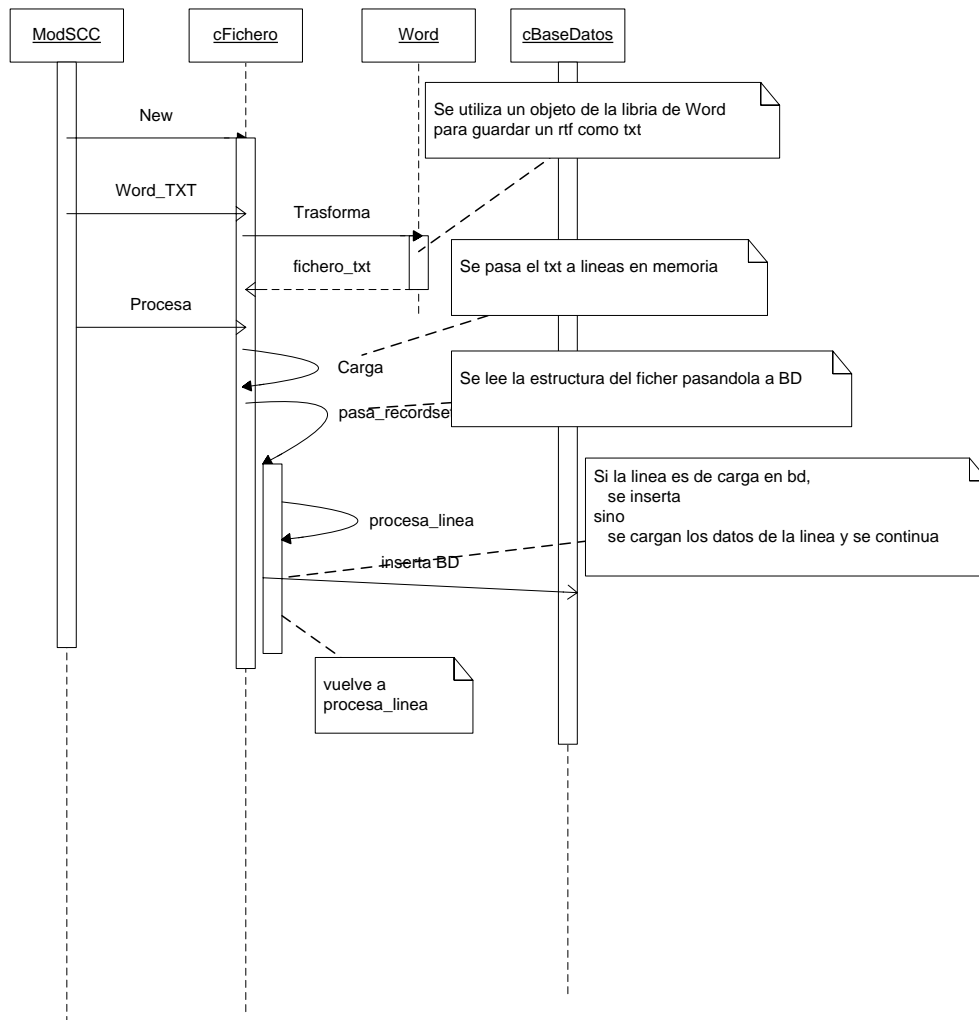
Si el usuario cancela, se detiene la ejecución.

Si el usuario Acepta:

- 1.- Se consulta al objeto base de datos el usuario, devolviendo la contraseña Cifrada, si no se devuelve ninguna contraseña, el usuario no existe y se finaliza la ejecución.
- 2.- Se cifra la contraseña introducida por el usuario y se compara con la contraseña cifrada obtenida de la base de datos. Si no son iguales no finaliza la ejecución.
- 3.- Una vez identificado, el objeto base de datos contiene una lista de opciones disponibles según el perfil cargado de la base de datos. Dichas opciones son utilizadas por el FRM_Principal para presentar solamente las opciones permitidas.

2.2.1.2 Importar

Todos los proceso de Importar como son: Importar Isométricas, Importar Vales e Importar Vales de Forma Masiva realizan la actividad de importación siguiendo el siguiente diagrama de secuencia:



Lo primero que se hace es crear una instancia de la clase cFichero a la que se le pasa un fichero rtf, la clase lo transforma atxt usando librerías de la herramienta Word.

Una vez asado a fichero txt, se realiza su carga en estructuras manejables en memoria y se llama al método privado pasa_recordset.

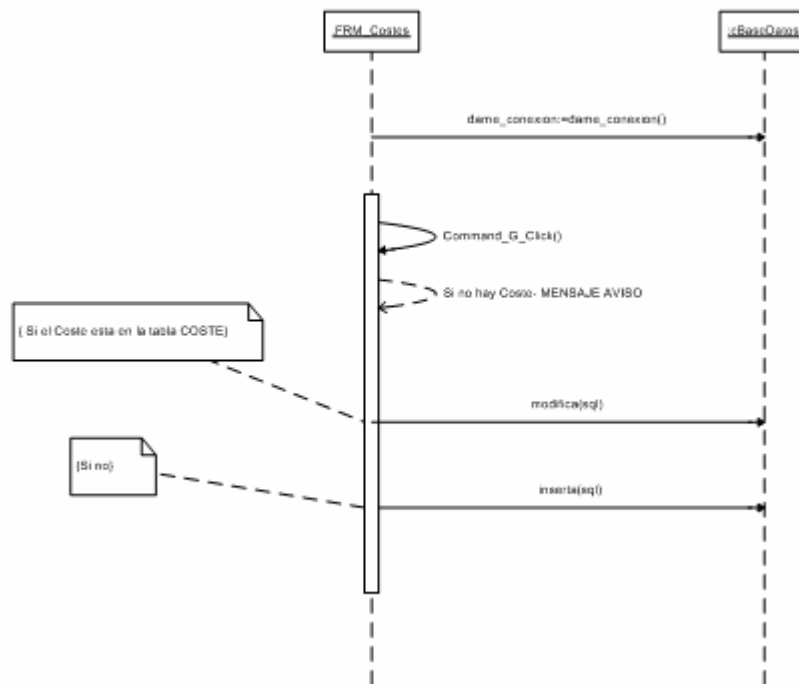
El método pasa a recordset realiza la función abstracta de pasar el fichero de txt a la base de datos, para ello utiliza la función procesa_linea, la cual tras consumir una línea indica que clase de línea es (cabecera, totales, línea a insertar) y en caso de ser una línea con información necesaria almacena en variables los datos leídos.

El método pasa a recordset, una vez analizada la línea, inserta en base de datos si es necesario, almacena temporalmente los datos o no realiza ninguna acción y vuelve a invocar a procesa_linea con la siguiente línea hasta que no quedan más líneas.

2.2.1.3 Guardar/ Modificar Coste

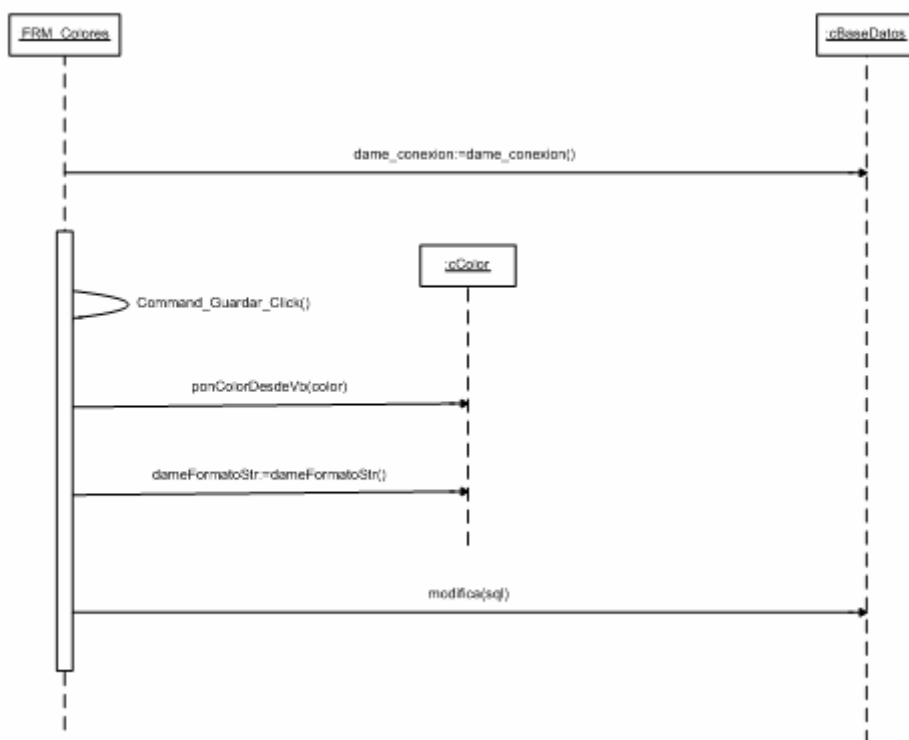
Muestra el diseño de la acción de guardar el coste de prefabricación y montaje. Si el coste no está indicado muestra un mensaje de aviso y si no lo modifica o lo inserta dependiendo de si este coste existe o no en la tabla de COSTES.

La ejecución comienza con la solicitud de la conexión a la base de datos para mostrar los datos de costes que hay registrados en ella y realizar las consultas necesarias de modificación o inserción mediante los métodos correspondientes de la clase cBaseDatos.



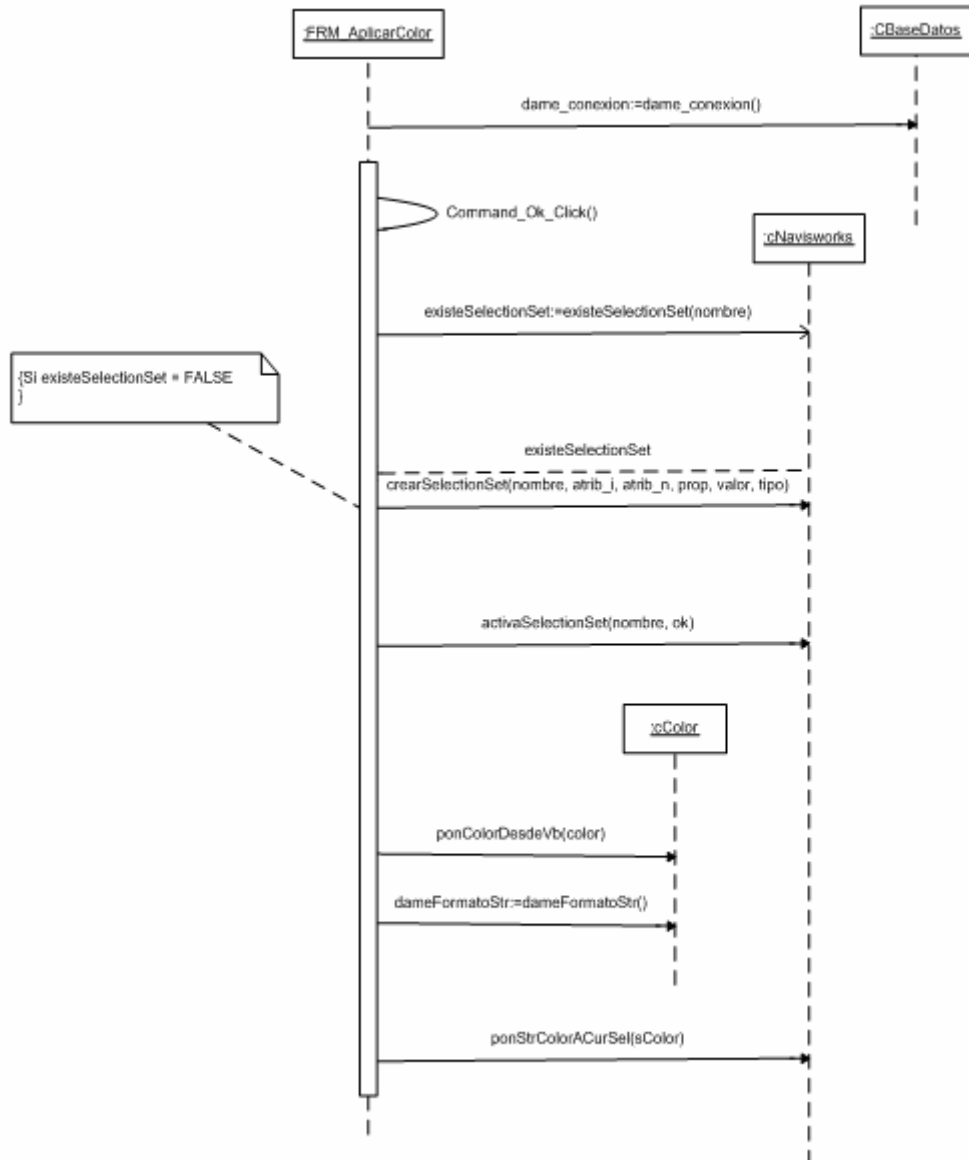
2.2.1.4 Guardar Color

Muestra el diseño de la acción de guardar el color que queremos asignar en la maqueta al material que haya sido marcado como prefabricado o montado. El color se selecciona desde el formulario FRM_Colores. En él se muestra la paleta de colores de Windows. En la base de datos, el color se guarda en formato RGB como una cadena de texto de la siguiente forma: (0..255, 0..255, 0..255). Por este motivo es necesario utilizar los métodos ponColorDesdeVb y dameFormatoStr de la clase cColor, que se encargan de guardar el color seleccionado desde el formulario y luego convertirlo al formato establecido en la base de datos y lo modifica en la tabla _CONFIG_COLORES.



2.2.1.5 Aplicar Color

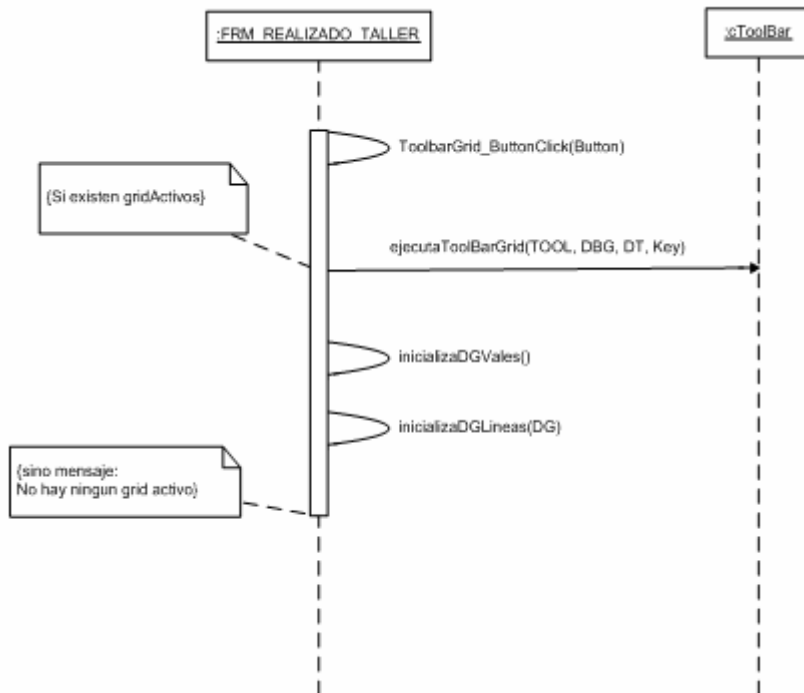
Muestra el diseño de la acción de aplicar color en los materiales que previamente han sido configurados, aplica el color de la tabla `_CONFIG_COLORES` a los materiales ya sean prefabricados o montados en la maqueta.



2.2.1.6 Barra de herramientas de selección de isométricas

Muestra el diseño de la acción de las opciones de la barra de herramientas de selección de isométricas, que para utilizarla, es necesario activar mediante un clic la tabla de datos (grid) sobre la que se desea trabajar (tabla de vales o tabla de líneas).

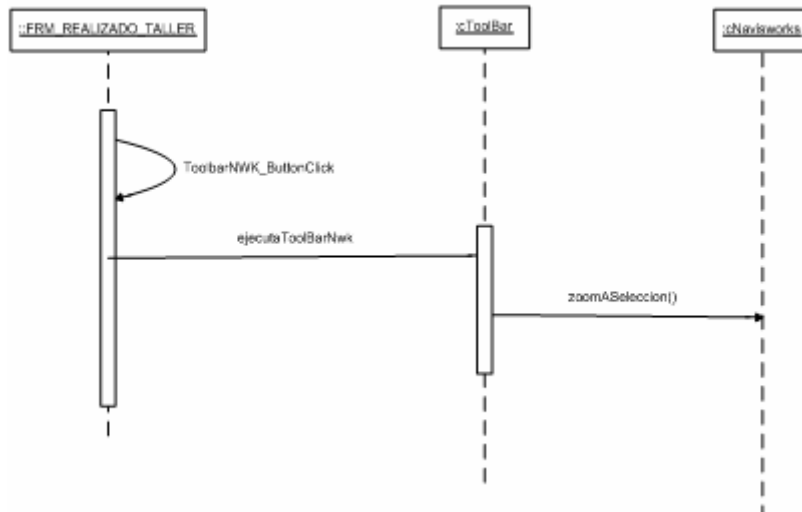
En este caso muestra el clic de la opción de la barra de herramientas Orden Ascendente pero es similar para las demás opciones (Orden Descendente, Buscar, Filtro por Selección y Quitar Filtro).



2.2.1.7 Barra de herramientas de la Interfaz del visualizador 3D

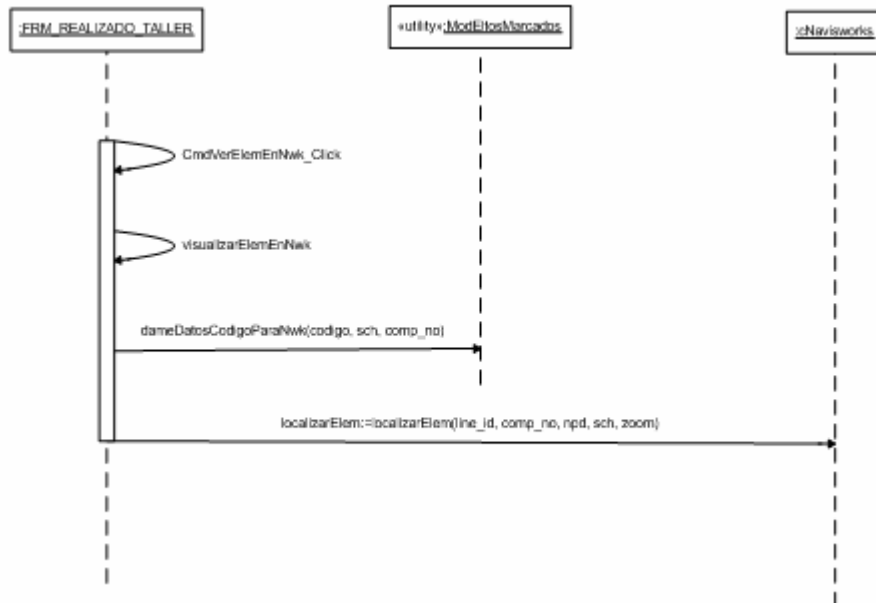
Muestra el diseño de la acción que se desencadena al utilizar alguna de las opciones de la barra de herramientas de la interfaz con el visualizador de la maqueta 3D. El diagrama ilustra la opción de realizar un zoom sobre los elementos seleccionados en la maqueta.

En este caso muestra el clic de la opción de la barra de herramientas Zoom de elementos seleccionados.



2.2.1.8 Ver elemento de la lista en el 3D

Muestra el diseño de la acción de ver elemento de la lista en 3D que permite visualizar sobre la maqueta 3D el elemento seleccionado en la lista de material activa.

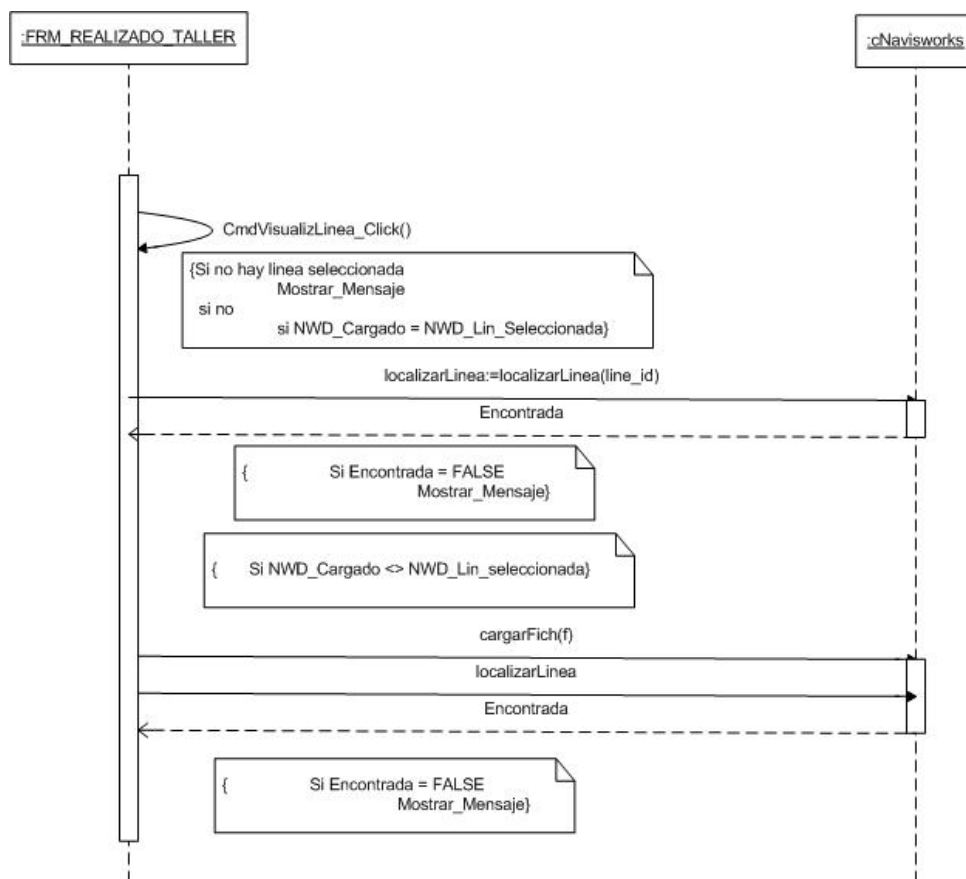


2.2.1.9 Visualizar Línea sobre el 3D

En el siguiente diagrama se ilustra la secuencia de código que debe implementarse para buscar una línea y visualizarla en la maqueta 3D.

La ejecución se inicia en el formulario del Taller, invocando el evento clic del botón correspondiente a la opción de Visualizar Línea. Esta acción desencadena una serie de llamadas a métodos que permiten realizar las comprobaciones necesarias para visualizar la línea seleccionada o mostrar el mensaje de aviso oportuno.

Para poder visualizar una línea se deben cumplir las siguientes condiciones: que exista una línea seleccionada en la tabla, que la línea que se busca se encuentre en el fichero con extensión nwd que está cargado actualmente y, por último, que la línea efectivamente esté en la maqueta 3D.



2.2.1.10 Marcar elementos del 3D

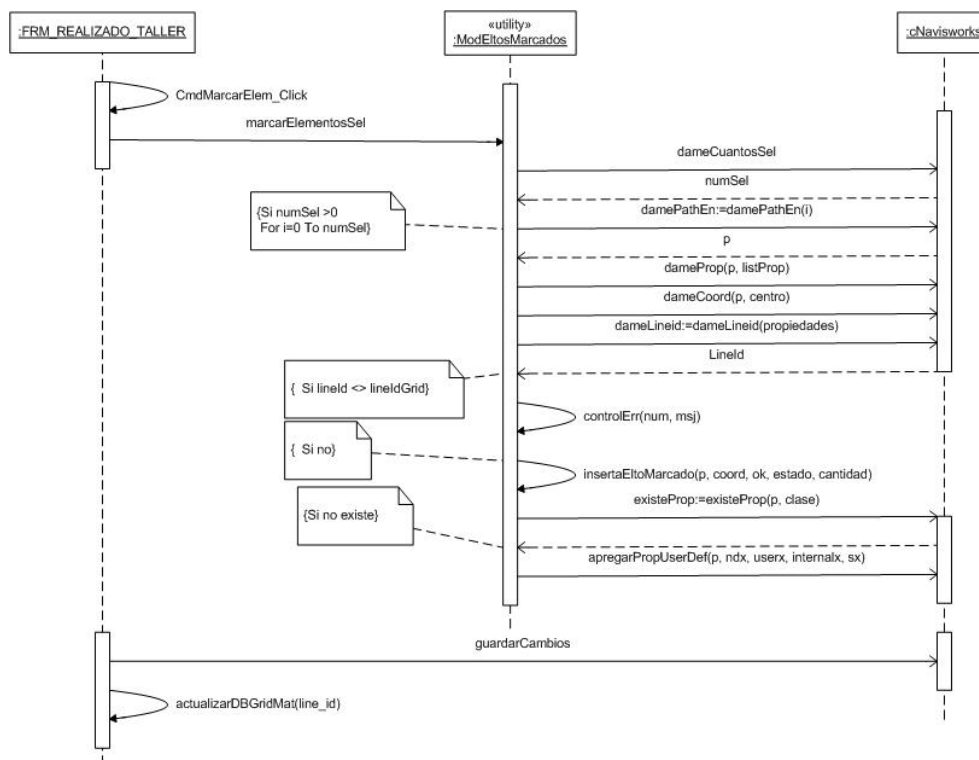
En el siguiente diagrama de secuencia se indica el comportamiento que debe implementar el sistema con el fin de realizar la acción que permite al usuario marcar elementos sobre la maqueta 3D.

En primer lugar, se debe extraer información de la maqueta 3D con lo que se utilizarán los métodos de la clase *cNavisworks* necesarios para obtener los datos siguientes:

- El número de elementos seleccionados, mediante el método *dameCuantosSel*.
- La información necesaria para identificar el elemento que se ha seleccionado. Para ello, es necesario recuperar la ruta en la jerarquía de objetos de la maqueta utilizando el método *damePathEn*. A continuación, se extraerán los datos del elemento seleccionado mediante los métodos: *dameProp*, *dameCoord* y *dameLineid*.

Se validarán los datos extraídos. Si son correctos, se insertará en la base de datos el registro de datos asociado al elemento marcado, mediante el método *insertaEltoMarcado* y se creará, en la maqueta 3D, la propiedad que indica que el elemento ha sido prefabricado o montado, utilizando el método *agregarPropUserDef*. En caso de detectarse algún error, se tratará mediante el procedimiento *controlErr*.

Finalmente, se guardarán los cambios realizados sobre la maqueta 3D, utilizando el método *guardarCambios* de la clase *cNavisworks*, y se actualizará la información del formulario para reflejar el nuevo estado que presentan las diferentes listas de material mediante el método *actualizarDBGridMat* del formulario taller.



2.2.1.11 Desmarcar elementos del 3D

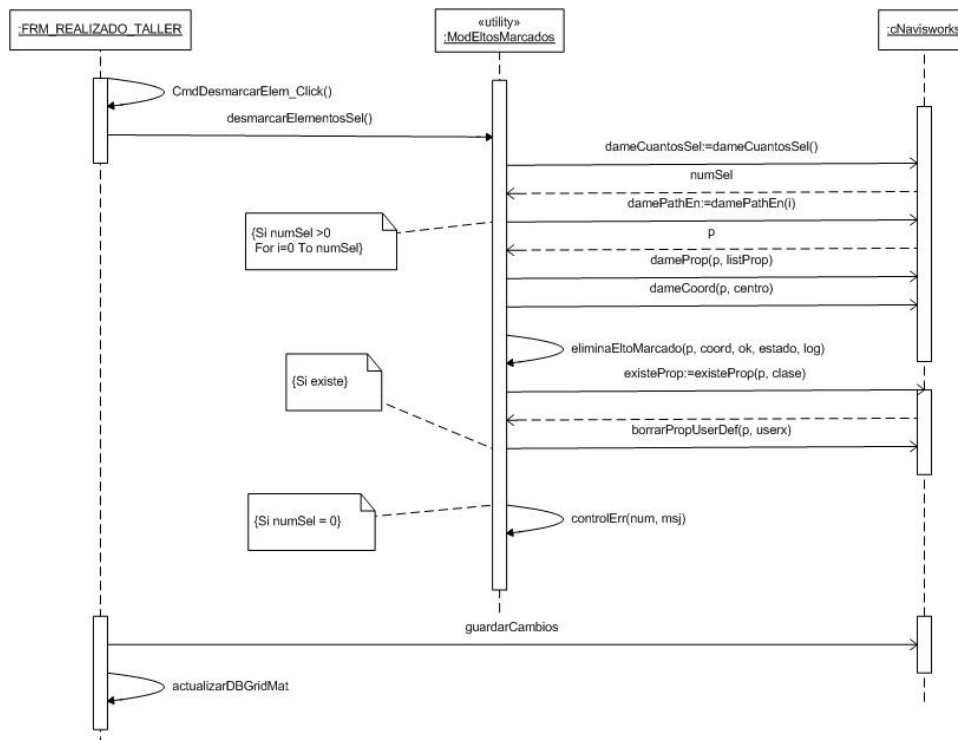
En el siguiente diagrama de secuencia se indica el comportamiento que debe implementar el sistema con el fin de realizar la acción que permite al usuario desmarcar elementos sobre la maqueta 3D.

En primer lugar, se debe extraer información de la maqueta 3D con lo que se utilizarán los métodos de la clase *cNavisworks* necesarios para obtener los datos siguientes:

- El número de elementos seleccionados, mediante el método *dameCuantosSel*.
- La información necesaria para identificar el elemento que se ha seleccionado. Para ello, es necesario recuperar la ruta en la jerarquía de objetos de la maqueta utilizando el método *damePathEn*. A continuación, se extraerán los datos del elemento seleccionado mediante los métodos: *dameProp* y *dameCoord*.

Una vez extraídos los datos se eliminará el registro correspondiente en la base de datos, mediante el método *eliminaEltoMarcado* y se borrará, de la maqueta 3D, la propiedad que indica que el elemento ha sido prefabricado o montado según corresponda, utilizando el método *borrarPropUserDef*. En caso de detectarse algún error, se tratará mediante el procedimiento *controlErr*.

Finalmente, se guardarán los cambios realizados sobre la maqueta 3D, utilizando el método *guardarCambios* de la clase *cNavisworks*, y se actualizará la información del formulario para reflejar el nuevo estado que presentan las diferentes listas de material mediante el método *actualizarDBGridMat* del formulario taller.



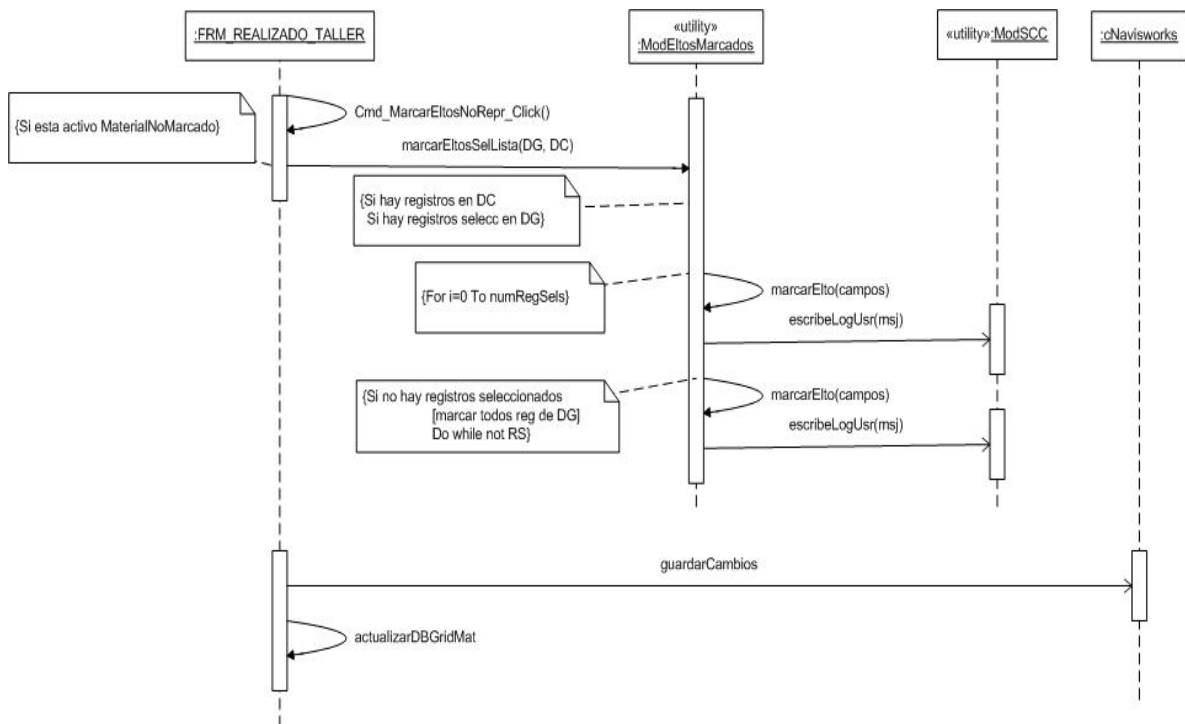
2.2.1.12 Marcar elementos desde lista

En el siguiente diagrama de secuencia se indica el comportamiento que debe implementar el sistema con el fin de realizar la acción que permite al usuario marcar elementos sobre la lista de material no marcado de una isométrica.

La secuencia de código se desencadena desde el formulario del taller mediante el evento clic del botón correspondiente a esta opción. A continuación, se realiza la llamada al método *marcarEltoSelLista* del módulo auxiliar *ModEltosMarcados*. Este método se encarga de comprobar los elementos que están seleccionados en la lista de materiales y de realizar la inserción en la base de datos de los elementos correspondientes mediante el procedimiento *marcarElto*.

Durante la ejecución, se escribirá en el fichero log información sobre las operaciones que se han ido realizando. Esta tarea se llevará a cabo mediante el método *escribeLogUsr* del módulo principal, *ModSCC*.

Finalmente, se guardarán los cambios realizados sobre la maqueta 3D, utilizando el método *guardarCambios* de la clase *cNavisworks*, y se actualizará la información del formulario para reflejar el nuevo estado que presentan las diferentes listas de material mediante el método *actualizarDBGridMat* del formulario taller.

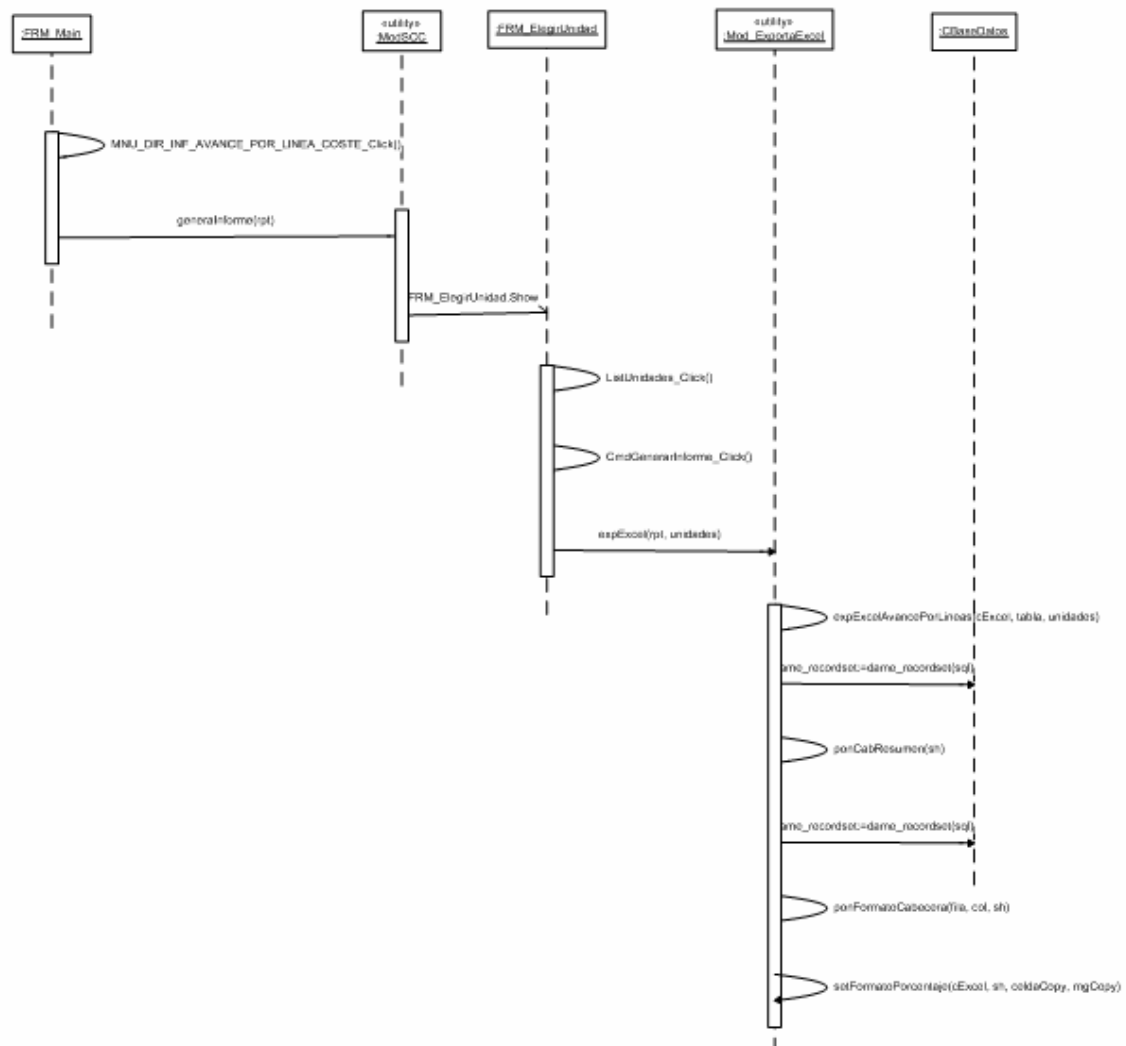


2.2.1.13 Generar Informe

Muestra el diseño de la acción de generar informe. En este caso, se muestra la secuencia para el informe de avance por línea de coste. Análogamente, se realizará para generar el informe de avance por línea en kg.

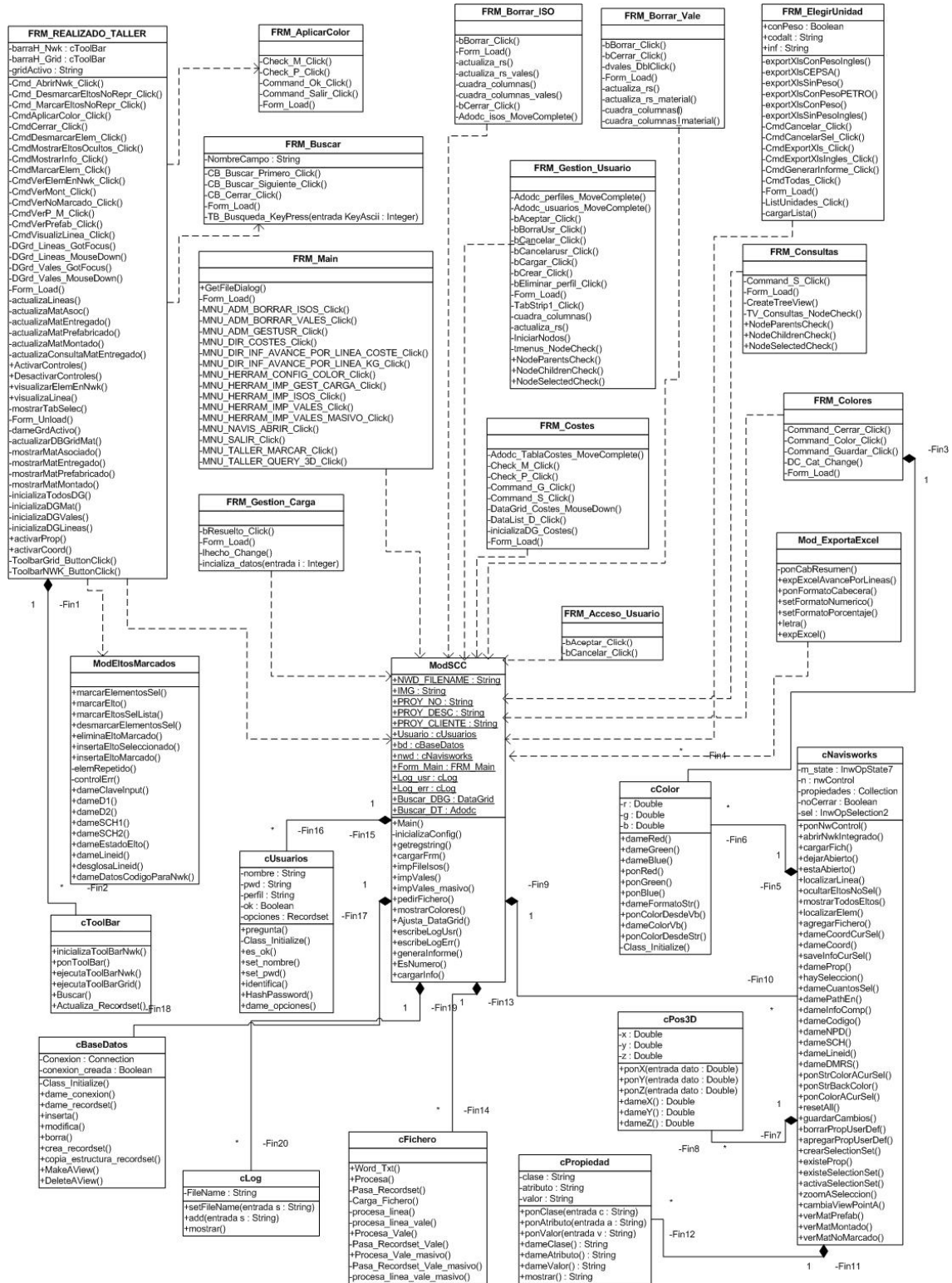
Una vez que se muestra el formulario principal (*FRM_Main*) y se selecciona la opción de *Dir. Construcción* → *Informes* → *Avance por línea en coste*, se invoca el procedimiento *generaInforme("AVANCE_POR_LINEA")*, que se encuentra en el Módulo principal (*ModSCC*). Este procedimiento se encargará de mostrar el formulario *FRM_ElegirUnidad*, que permite elegir la unidad o unidades de las que se quiere generar el informe.

Al generar el informe, se realiza una llamada a *expExcel(rpt, unidades)*, que está en el módulo que se encarga de crear los informes en Excel (*Mod_ExportaExcel*), y éste, a su vez, llama a *expExcelAvancePorLineas(cExcel, tabla, unidades)* que se encarga de ir generando el informe realizando los accesos necesarios a la base de datos y dando el correspondiente formato.



2.2.2 Diagrama de Clases del Sistema

2.2.2.1 Diagrama de clases



2.2.2.2 Clase cColor

- **Descripción:** implementa el comportamiento necesario para tratar el formato de color que maneja Visual Basic y el formato RGB que se almacena en la base de datos.
- **Métodos:**
 - **dameRed() As Double:** accedente que devuelve el valor establecido para el color rojo.
 - Parámetros por valor: Ninguno.
 - Parámetros por referencia: Ninguno.
 - Salida: Devuelve un Double con el valor de r.
 - **dameGreen() As Double:** accedente que devuelve el valor establecido para el color verde.
 - Parámetros por valor: Ninguno.
 - Parámetros por referencia: Ninguno.
 - Salida: Devuelve un Double con el valor de g.
 - **dameBlue() As Double:** accedente que devuelve el valor establecido para el color azul.
 - Parámetros por valor: Ninguno.
 - Parámetros por referencia: Ninguno.
 - Salida: Devuelve un Double con el valor de b.
 - **ponRed(ByVal red As Double):** mutador que establece el color rojo con el valor pasado por parámetro.
 - Parámetros por valor: red de tipo Double.
 - Parámetros por referencia: Ninguno.
 - Salida: Ninguna.
 - **ponGreen(ByVal green As Double):** mutador que establece el color verde con el valor pasado por parámetro.
 - Parámetros por valor: green de tipo Double.
 - Parámetros por referencia: Ninguno.
 - Salida: Ninguna.
 - **ponBlue(ByVal blue As Double):** mutador que establece el color azul con el valor pasado por parámetro.
 - Parámetros por valor: blue de tipo Double.
 - Parámetros por referencia: Ninguno.
 - Salida: Ninguna.
 - **dameFormatoStr() As String:** función que devuelve el valor almacenado en el objeto cColor, en el formato (R, G, B).
 - Parámetros por valor: Ninguno.
 - Parámetros por referencia: Ninguno.
 - Salida: Devuelve un String con el color en formato (R,G,B).
 - **ponColorDesdeVB (ByVal String c):** establece los valores del vector RGB a partir del valor recibido como parámetro en el formato de texto almacenado en la base de datos (R,G,B).

- Parámetros por valor: c de tipo String.
- Parámetros por referencia: Ninguno.
- Salida: Ninguna
- **dameColorVb() As Long**: función que devuelve el valor almacenado en el objeto cColor, en el formato que utiliza Visual Basic.
 - Parámetros por valor: Ninguno.
 - Parámetros por referencia: Ninguno.
 - Salida: Devuelve un Long con el color en formato que utiliza Visual Basic.
- **ponColorDesdeStr (ByVal Long c)**: establece los valores del vector RGB a partir del valor recibido como parámetro en el formato utilizado por Visual Basic.
 - Parámetros por valor: c de tipo Long.
 - Parámetros por referencia: Ninguno.
 - Salida: Ninguna.
- **Class_Initialize()**: procedimiento que inicializa los atributos r, g y b de color a cero.
 - Parámetros por valor: Ninguno.
 - Parámetros por referencia: Ninguno.
 - Salida: Ninguna.
- **Referencias:**

2.2.2.3 Clase cPropiedad

- **Descripción:** implementa el par atributo-valor de una propiedad.
- **Métodos:**
 - **ponClase(ByVal c As String)**: mutador que establece la clase de una propiedad con el valor pasado por parámetro.
 - Parámetros por valor: c de tipo String.
 - Parámetros por referencia: Ninguno.
 - Salida: Ninguna.
 - **ponAtributo(ByVal a As String)**: mutador que establece el atributo de una propiedad con el valor pasado por parámetro.
 - Parámetros por valor: a de tipo String.
 - Parámetros por referencia: Ninguno.
 - Salida: Ninguna.
 - **ponValor(ByVal v As String)**: mutador que establece el valor de una propiedad con el valor pasado por parámetro.
 - Parámetros por valor: v de tipo String.
 - Parámetros por referencia: Ninguno.
 - Salida: Ninguna.
 - **dameClase() As String**: accedente que devuelve el valor establecido para la clase de una propiedad.
 - Parámetros por valor: Ninguno.
 - Parámetros por referencia: Ninguno.

- Salida: devuelve un String.
- **dameAtributo() As String**: accedente que devuelve el valor establecido para el atributo de una propiedad.
 - Parámetros por valor: Ninguno.
 - Parámetros por referencia: Ninguno.
 - Salida: devuelve un String.
- **dameValor() As String**: accedente que devuelve el valor establecido para el valor de una propiedad.
 - Parámetros por valor: Ninguno.
 - Parámetros por referencia: Ninguno.
 - Salida: devuelve un String..
- **mostrar() As String**: función que devuelve un String para mostrar el atributo con su correspondiente propiedad.
 - Parámetros por valor: Ninguno.
 - Parámetros por referencia: Ninguno.
 - Salida: devuelve un String.
- **Referencias:**

2.2.2.4 Clase cLog

- **Descripción:** implementa como generar ficheros .log donde se indique, y se utiliza para informar sobre el resultado de los procesos de carga de datos, como pueden ser: la importación de isométricas, de vales...
- **Métodos:**
 - **setFileName (ByVal String s)**: mutador que establece la ruta y nombre del fichero con el valor pasado por parámetro.
 - Parámetros por valor: s de tipo String.
 - Parámetros por referencia: Ninguno.
 - Salida: Ninguna.
 - **Add (ByVal String s)**: añade al final del fichero indicado el valor pasado por parámetro.
 - Parámetros por valor: s de tipo String.
 - Parámetros por referencia: Ninguno.
 - Salida: Ninguna.
 - **Mostrar ()**: sirve para mostrar posibles errores
 - Parámetros por valor: Ninguno.
 - Parámetros por referencia: Ninguno.
 - Salida: Ninguna.
- **Referencias:**

2.2.2.5 Clase cBaseDatos

- **Descripción:** Esta clase es la encargada de crear, mantener y gestionar tanto los accesos a bases de datos del sistema como la creación de recordsets no vinculados a consultas específicas.
Sólo se debe crear una instancia de esta clase en la aplicación por base de datos que se utilice.
Es el interfaz de la aplicación con el gestor de bases de datos.
- **Métodos:**
 - **cBaseDatos new():** Este método devuelve un objeto de la clase cBaseDatos, crea la conexión a la base de datos “SCC.mdb” que se encuentra en el subdirectorio MDB del directorio de la aplicación.
 - Parámetros por valor: Ninguno
 - Parámetros por referencia: Ninguno
 - Salida: Devuelve una instancia de la clase cBaseDatos.
 - **ADODB.Connection dame_conexion()** accedente que devuelve la conexión de un objeto de clase CBaseDatos.
 - Parámetros por valor: Ninguno
 - Parámetros por referencia: Ninguno
 - Salida: Devuelve una conexión de tipo ADODB.Connection.
 - **ADODB.Recordset dame_recordset(ByVal sql As String):** Crea un recordset y carga en él la consulta especificada en *sql*, en caso que *sql* sea una tabla se carga entera con todos los campos y filas.
 - Parámetros por valor: *sql* de tipo String, especifica una consulta *sql* o el nombre de una tabla
 - Parámetros por referencia: Ninguno.
 - Salida: ADODB.Recordset.
 - **inserta(ByVal sql As String):** Ejecuta la instrucción *insert into* especificada en *sql*.
 - Parámetros por valor: *sql* de tipo String, especifica una instrucción “insert into” para una de las tablas del sistema.
 - Parámetros por referencia: Ninguno.
 - Salida: Ninguna.
 - **modifica(ByVal sql As String):** Ejecuta la instrucción “*update*” especificada en *sql*.
 - Parámetros por valor: *sql* de tipo String, especifica una instrucción “update” para una de las tablas del sistema.
 - Parámetros por referencia: Ninguno.
 - Salida: Ninguna.
 - **borra (By val sql as String)** método que invoca la instrucción *delete* especificada en *sq*
 - Parámetros por valor: *sql* de tipo String que especifica una instrucción SQL de tipo Delete.
 - Parámetros por Referencia: Ninguno.
 - Salida: Ninguna.

- **crea_recordset(ByVal tabla As String, rs As ADODB.Recordset):** Inicia el recordset pasado como *rs* con la estructura de tabla/consulta especificada en *tabla*. El recordset estará vacío y no vinculado a la tabla en la Base de Datos
 - Parámetros por valor: *tabla* de tipo String, especifica una consulta sql o el nombre de una tabla
 - Parámetros por referencia: *rs*, de tipo ADODB.Recordset, elimina el contenido anterior y se carga la estructura de sql.
 - Salida: Ninguna.
 - **copia_estructura_recordset(ByVal rsorigen As ADODB.Recordset, rsdestino As ADODB.Recordset):** Inicia el recordset pasado como *rsdestino* con la estructura del recordset *rsorigen*. El recordset *rsdestino* estará vacío y no vinculado a ninguna tabla en la Base de Datos
 - Parámetros por valor: *rsorigen* de tipo ADODB.Recordset, contiene el recordset cuya estructura se quiere copiar.
 - Parámetros por referencia: *rsdestino*, de tipo ADODB.Recordset, elimina el contenido anterior y se carga la estructura de *rsorigen*.
 - Salida: Ninguna.
 - **MakeAView(ByVal SPTQueryName As String, ByVal strSql As String):** Crea una vista en la base de datos del Sistema con el nombre especificado en *SPTQueryName*, la vista ejecutará la consulta contenida en *strSql*.
 - Parámetros por valor:
 - SPTQueryName de tipo String, contiene el nombre de la vista.
 - strSql: de tipo String Contiene la consulta a guardar e la base de datos.
 - Parámetros por referencia: Ninguno.
 - Salida: Ninguna.
 - **DeleteAView(ByVal SPTQueryName As String):** Borra la vista de la base de datos del Sistema con el nombre especificado en *SPTQueryName*.
 - Parámetros por valor: SPTQueryName de tipo String, contiene el nombre de la vista.
 - Parámetros por referencia: Ninguno.
 - Salida: Ninguna.
 - **Cierra():** Cierra la conexión existente con la base de datos del sistema.
 - Parámetros por valor: Ninguno.
 - Parámetros por referencia: Ninguno.
 - Salida: Ninguna
- **Referencias:**

2.2.2.6 Clase cFichero

Descripción: Esta clase es la encargada de leer los ficheros generados por el sistema de gestión de materiales y de cargar la información contenida en ellos en la base de datos del sistema mediante la clase CBaseDatos.

Es el interfaz de la aplicación con los ficheros generados por el sistema de gestión de materiales.

▪ **Métodos:**

- **Word_Txt(Path_Doc As String):** Procedimiento que convierte el fichero especificado en *Path_Doc* de formato “.rtf” a formato de texto plano “.txt” y lo almacena en un fichero con el mismo nombre que el original con extensión .txt.

- Parámetros por valor: *Path_Doc*, ruta completa del documento a convertir, el fichero debe ser un fichero rtf válido.
- Parámetros por referencia: Ninguno.
- Salida: Ninguna.

- **Procesa(Path_Doc As String):** Procedimiento principal que realiza la invocación del resto de procedimientos privados que interpretan el fichero, el fichero pasado en *Path_Doc* debe tener el formato de “material por isométrica” especificado.

El resultado de invocar este método es la carga en Base de Datos la información del fichero de texto *Path_Doc*.

- Parámetros por valor: *Path_Doc*, ruta completa del documento a procesar, el fichero debe ser un fichero de texto con el formato indicado.
- Parámetros por referencia: Ninguno.
- Salida: Ninguna.

- **Procesa_Vale(Path_Doc As String):** Procedimiento principal que realiza la invocación del resto de procedimientos privados que interpretan el fichero, el fichero pasado en *Path_Doc* debe tener el formato de “vale” especificado.

El resultado de invocar este método es la carga en Base de Datos la información del fichero de texto *Path_Doc*.

- Parámetros por valor: *Path_Doc*, ruta completa del documento a procesar, el fichero debe ser un fichero de texto con el formato indicado.
- Parámetros por referencia: Ninguno.
- Salida: Ninguna.

- **Procesa_Vale_Masivo(Path_Doc As String):** Procedimiento principal que realiza la invocación del resto de procedimientos privados que interpretan el fichero, el fichero pasado en *Path_Doc* debe tener el formato de “vale_masivo” especificado, el cual es igual al formato de “material por isométrica”. Solo se utilizará en cargas iniciales.

El resultado de invocar este método es la carga en Base de Datos la información del fichero de texto *Path_Doc*.

- Parámetros por valor: *Path_Doc*, ruta completa del documento a procesar, el fichero debe ser un fichero de texto con el formato indicado.
 - Parámetros por referencia: Ninguno.
 - Salida: Ninguna.
- **Referencias:**
- CBaseDatos

2.2.2.7 Clase cUsuarios

Descripción: Clase encargada de realizar el control de acceso a la aplicación. Para ello, hace uso del formulario *FRM_Acceso_Usuario*. En la aplicación debe existir un objeto global de esta clase llamado Usuario.

- **Métodos:**
- **cUsuarios new():** Este método devuelve un objeto de la clase cUsuarios. Se encarga de inicializar los atributos de la clase.
 - Parámetros por valor: Ninguno.
 - Parámetros por referencia: Ninguno.
 - Salida: Devuelve una instancia de la clase cUsuario.
 - **Pregunta():** Procedimiento que muestra el formulario *FRM_Acceso_Usuario*, el cual guarda al Aceptar el usuario y la contraseña en los atributos privados.
 - Parámetros por valor: Ninguno.
 - Parámetros por referencia: Ninguno.
 - Salida: Ninguna.
 - **es_ok():** Función que devuelve si se ha realizado con éxito el proceso de identificación sobre el usuario actual, se debe haber invocado anteriormente el método *identifica()* al menos una vez.
 - Parámetros por valor: Ninguno.
 - Parámetros por referencia: Ninguno.
 - Salida: *True* si se a identificado correctamente, *false* en caso contrario.
 - **set_nombre(ByVal nom as String):** Accedente del atributo nombre.
 - Parámetros por valor: *nom*, nombre del usuario.
 - Parámetros por referencia: Ninguno.
 - Salida: Ninguna.
 - **set_pwd(ByVal password as String):** Accedente del atributo *pwd*.
 - Parámetros por valor: *pwd*, contraseña en claro.
 - Parámetros por referencia: Ninguno.
 - Salida: Ninguna.

- **Identifica():** Metodo que comprueba el usuario con los datos almacenados en los atributos usuario y pwd, previo a la consulta a base de datos se pasa la contraseña por una función Hash unidireccional, luego se compara con el valor almacenado en base de datos. En caso de que la correlación sea correcta, se establece el atributo ok a true y se carga opciones con los establecidos en base de datos.
 - Parámetros por valor: Ninguno.
 - Parámetros por referencia: Ninguno.
 - Salida: Ninguna.
- **dame_opciones():** Accedente al atributo opciones, en caso de que no se haya inicializado se devuelve *nothing*.
 - Parámetros por valor: Ninguno.
 - Parámetros por referencia: Ninguno.
 - Salida: Ninguna.
- **Referencias:**
 - CBaseDatos
 - FRM_Acceso_Usuario

2.2.2.8 Clase cNavisworks

- **Descripción:** se utiliza para encapsular el acceso al API de programación de Navisworks. En esta clase se implementan los métodos necesarios para interactuar con Navisworks desde el proyecto. De esta forma es posible utilizar el navegador sin necesidad de conocer las particularidades de sus librerías.
- **Métodos:**
 - **cargarFich(ByVal f as String):** abre en el control de Navisworks (nwControl) el fichero .nwd pasado como parámetro.
 - Parámetros por valor: nombre del fichero, incluyendo su extensión nwd, que se quiere abrir.
 - Parámetros por referencia: Ninguno
 - Salida: Ninguna.
 - **ponNwControl(nwc As nwControl):** accedente que establece el objeto nwControl con el valor pasado por parámetro.
 - Parámetros por valor: Ninguno.
 - Parámetros por referencia: el objeto nwControl que se va a establecer dentro del objeto Navisworks.
 - Salida: Ninguna.
 - **abrirNwkIntegrado():** establece el objeto InwOpState7 del objeto Navisworks con el valor del objeto nwControl que se utiliza desde el formulario.
 - Parámetros por valor: Ninguno.
 - Parámetros por referencia: Ninguno
 - Salida: Ninguna.

- **localizarLinea (ByVal String line_id) As Boolean:** busca y selecciona sobre la maqueta todos los elementos asociados a la línea cuyo identificador se pasa como parámetro.
 - Parámetros por valor: identificador de la línea que se va a buscar en la maqueta 3D.
 - Parámetros por referencia: Ninguno
 - Salida: *True* si se encuentra
False en otro caso.
- **localizarElem(ByVal line_id As String, ByVal comp_no As String, _
ByVal npd As String, ByVal sch As String, ByVal zoom As Boolean)
As Boolean:** busca y selecciona el componente cuyos datos se indican en los parámetros de entrada.
 - Parámetros por valor:
 - *Line_id*, identificador de la línea.
 - *Comp_no*, valor de la propiedad *Comp no* en la maqueta.
 - *Npd*, valor de la propiedad *NPD* (diámetro nominal).
 - *Sch*, valor de la propiedad *Sch/thk* (Schedule).
 - si el parámetro *zoom* se establece a *True* se hará un zoom sobre el elemento una vez encontrado y si se establece a *False* no se realizará el zoom.
 - Parámetros por referencia: Ninguno
 - Salida: *True* si encuentra el elemento indicado,
False en otro caso.
- **dameInfoComp(listProp As Collection):** método que devuelve en *listProp* la lista de atributos del elemento seleccionado.
 - Parámetros por valor: Ninguno.
 - Parámetros por referencia: colección de la clase *cPropiedad*.
 - Salida: en *listProp* se devolverá la información asociada al componente seleccionado en la maqueta 3D.
- **ponColorACurSel (c As cColor):** establece el color pasado por parámetro, como color para la selección actual en la maqueta.
 - Parámetros por valor: Ninguno.
 - Parámetros por referencia: *c*, objeto color que se va a establecer como color para la selección actual en la maqueta 3D.
 - Salida: Ninguna.
- **ponStrColorACurSel (ByVal c As String):** establece el color pasado como parámetro, en formato vector RGB (255, 255, 255), como color para la selección actual.
 - Parámetros por valor: *c*, valor del color, en formato cadena de caracteres, que se va a establecer como color para la selección actual en la maqueta 3D.
 - Parámetros por referencia: Ninguno
 - Salida: Ninguna.

- **resetAll():** deshace los cambios realizado anteriormente y deja la configuración como estaba al abrir el fichero de Navisworks.
 - Parámetros por valor: Ninguno.
 - Parámetros por referencia: Ninguno
 - Salida: Ninguna.
- **dameCoordCurSel() As cPos3D:** función que devuelve un objeto cPos3D con las coordenadas del elemento seleccionado en la maqueta.
 - Parámetros por valor: Ninguno.
 - Parámetros por referencia: Ninguno
 - Salida: coordenadas X, Y, Z del elemento seleccionado.
- **dameCoord(p As InwOaPath3, centro As cPos3D):** método que devuelve en *centro* las coordenadas del elemento situado en *p*.
 - Parámetros por valor: Ninguno.
 - Parámetros por referencia:
 - *p* indica la ruta dentro de la jerarquía de Navisworks del elemento.
 - *Centro*, devolverá sus coordenadas X, Y, Z.
 - Salida: en el parámetro *centro* se devolverán las coordenadas X, Y, Z del elemento indicado.
- **saveInfoCurSel():** método que almacena en el atributo de clase *propiedades* , los atributos del elemento seleccionado en la maqueta.
 - Parámetros por valor: Ninguno.
 - Parámetros por referencia: Ninguno
 - Salida: Ninguna.
- **dameProp(p As InwOaPath3, listProp As Collection):** método que devuelve en *listProp* , que es una colección de objetos cPropiedad, los atributos correspondientes al elemento situado en *p*.
 - Parámetros por valor: Ninguno.
 - Parámetros por referencia:
 - *p* indica la ruta dentro de la jerarquía de Navisworks del elemento.
 - *listProp*, devolverá la lista de propiedades correspondiente al elemento.
 - Salida: *listProp*, lista de propiedades correspondiente al elemento indicado.
- **haySeleccion() As Boolean:** función que devuelve *True* si hay algún elemento seleccionado en la maqueta y *False*, en otro caso.
 - Parámetros por valor: Ninguno.
 - Parámetros por referencia: Ninguno
 - Salida: *True* si hay algún elemento seleccionado en la maqueta *False* en otro caso.
- **dameCuantosSel() As Integer:** función que devuelve el número de elementos seleccionados en la maqueta.
 - Parámetros por valor: Ninguno.
 - Parámetros por referencia: Ninguno
 - Salida: número de elementos seleccionados en la maqueta 3D.

- **damePathEn(ByVal i As Integer) As InwOaPath3:** función que devuelve el objeto InwOaPath3 seleccionado en i-ésimo lugar.
 - Parámetros por valor: *i*, posición en la que se encuentra el objeto InwOaPath3 al que se quiere acceder.
 - Parámetros por referencia: Ninguno
 - Salida: objeto InwOaPath3 situado en la posición i-ésima.

- **dameCodigo(propiedades As Collection) As String:** devuelve el valor de la propiedad *Comp no* de la colección *propiedades*, pasada como parámetro.
 - Parámetros por valor: Ninguno.
 - Parámetros por referencia: *propiedades*, lista de propiedades.
 - Salida: valor de la propiedad *Comp no* de la lista pasada como parámetro.

- **dameNPD(propiedades As Collection) As String:** devuelve el valor de la propiedad *NPD* de la colección *propiedades*, pasada como parámetro.
 - Parámetros por valor: Ninguno.
 - Parámetros por referencia: *propiedades*, lista de propiedades.
 - Salida: valor de la propiedad *NPD* de la lista pasada como parámetro.

- **dameSCH(propiedades As Collection) As String:** devuelve el valor de la propiedad *Sch/thk* de la colección *propiedades*, pasada como parámetro.
 - Parámetros por valor: Ninguno.
 - Parámetros por referencia: *propiedades*, lista de propiedades.
 - Salida: valor de la propiedad *Sch/thk* de la lista pasada como parámetro.

- **dameLineid(propiedades As Collection) As String:** devuelve el valor de la propiedad *Line_id* de la colección *propiedades*, pasada como parámetro.
 - Parámetros por valor: Ninguno.
 - Parámetros por referencia: *propiedades*, lista de propiedades.
 - Salida: valor de la propiedad *Line_id* de la lista pasada como parámetro.

- **dameDMRS(propiedades As Collection) As String:** devuelve el valor de la propiedad *DMRS* de la colección *propiedades*, pasada como parámetro.
 - Parámetros por valor: Ninguno.
 - Parámetros por referencia: *propiedades*, lista de propiedades.
 - Salida: valor de la propiedad *DMRS* de la lista pasada como parámetro.

- **guardarCambios():** método que guarda en el fichero .nwd abierto, los cambios realizados.
 - Parámetros por valor: Ninguno.
 - Parámetros por referencia: Ninguno
 - Salida: Ninguna.

- **apregarPropUserDef(p As InwOaPath3, ByVal ndx As Long, ByVal userx As String, ByVal internalx As String, ByVal sx As String):** método que añade una propiedad al elemento situado en el path *p*.
 - Parámetros por valor: *ndx*, con valor 0 para indicar que se añade la propiedad en otro caso se modifica.
 - *Userx*, indica el nombre de la propiedad que se añade.
 - *Internalx*, indica el nombre interno de la propiedad que se añade.
 - *Sx*, valor de la propiedad que se añade.
 - Parámetros por referencia: *p* indica la ruta dentro de la jerarquía de Navisworks del elemento.
 - Salida: Ninguna.

- **existeProp(p As InwOaPath3, ByVal clase As String) As Boolean:** función que devuelve true si la clase de propiedades *clase*, existe en elemento situado en el path *p*.
 - Parámetros por valor: *clase*, valor del nombre de la clase de propiedades que quiere comprobarse si existe.
 - Parámetros por referencia: *p* indica el path dentro de la jerarquía de Navisworks del elemento.
 - Salida: *True* si existe la clase de propiedades *clase* en el elemento indicado.
False en otro caso.

- **borrarPropUserDef(p As InwOaPath3, userx As String):** borra la propiedad de usuario *userx* del elemento situado en *p*.
 - Parámetros por valor: *userx*, valor del nombre de la clase de propiedades que se quiere eliminar.
 - Parámetros por referencia: *p* indica el path dentro de la jerarquía de Navisworks del elemento.
 - Salida: Ninguna.

- **crearSelectionSet(ByVal nombre As String, ByVal atrib_i As String, ByVal atrib_n As String, ByVal prop As String, ByVal valor As String, tipo As NWK_TipoCond):** método que crea en el fichero nwd activo un *Selection Set* que permite seleccionar todos los elementos de la maqueta que cumplan una determinada condición de búsqueda.
 - Parámetros por valor: *nombre*, indica el nombre del *Selection Set*.
 - *atrib_i*, nombre interno de la clase de propiedades que se utiliza en el criterio de búsqueda.
 - *atrib_n*, nombre de usuario de la clase de propiedades que se utiliza en el criterio de búsqueda.

- *Prop*, nombre de la propiedad por la que se realizará la búsqueda.
 - *Valor*, valor de la propiedad indicada en *prop*.
 - Parámetros por referencia: *Tipo*, tipo de búsqueda (Tiene Atributo o Contiene).
 - Salida: Ninguna.
- **existeSelectionSet(ByVal nombre As String) As Boolean**: comprueba si en Navisworks existe un *Selection Set* con el nombre pasado por parámetro.
 - Parámetros por valor: *nombre*, indica el nombre del *Selection Set*.
 - Parámetros por referencia: Ninguno
 - Salida: *True* si existe un *Selection Set* con nombre indicado, *False* en otro caso.
- **activaSelectionSet(ByVal nombre As String, ok As Boolean)**: método que selecciona en la maqueta los elementos que cumplen el criterio indicado en el *Selection Set* cuyo nombre se pasa por parámetro.
 - Parámetros por valor: *nombre*, indica el nombre del *Selection Set* que se quiere activar.
 - Parámetros por referencia: *ok*, parámetro de control.
 - Salida: *ok*, devolverá *True* si la operación se ha realizado con éxito, y devolverá *False* en otro caso.
- **mostrarTodosEltos()**: método que permite visualizar todos los elementos de la maqueta, incluso los que estuvieran ocultos por la aplicación de algún *Selection Set*.
 - Parámetros por valor: Ninguno.
 - Parámetros por referencia: Ninguno.
 - Salida: Ninguno.
- **ocultarEltosNoSel()**: método que oculta todos aquellos elementos de la maqueta 3D que no estén seleccionados.
 - Parámetros por valor: Ninguno.
 - Parámetros por referencia: Ninguno.
 - Salida: Ninguno.
- **verMatMontado()**: método que activa el *Selection Set* correspondiente al material “Montado”. Si el *Selection Set* no existe, lo crea.
 - Parámetros por valor: Ninguno.
 - Parámetros por referencia: Ninguno.
 - Salida: Ninguno.
- **verMatNoMarcado(ByVal Negado As Boolean)**: método que permite mostrar/ocultar los elementos que han sido marcados por el contratista como montados o prefabricados.
 - Parámetros por valor: *negado*, si tiene valor *True* se muestran los no marcados, si tiene valor *False*, se muestran los marcados.
 - Parámetros por referencia: Ninguno.
 - Salida: Ninguno.

- **verMatPrefab():** método que activa el *Selection Set* correspondiente al material “Prefabricado”. Si el *Selection Set* no existe, lo crea.
 - Parámetros por valor: Ninguno.
 - Parámetros por referencia: Ninguno.
 - Salida: Ninguno.
- **cambiaViewPointA(ByVal p As String):** método que permite cambiar al usuario la forma de desplazarse por la maqueta.
 - Parámetros por valor: *p*, indica el tipo de desplazamiento que se permite. Puede tomar uno de los siguientes valores: Fly, Examine, Walk, Turntable, Pan, Zoom, ZoomBox, Orbit y Select.
 - Parámetros por referencia: Ninguno.
 - Salida: Ninguno.
- **ponStrBackColor(ByVal sColor As String):** método que establece el color del fondo del visualizador de la maqueta al color pasado por parámetro.
 - Parámetros por valor: *sColor*, color que se quiere establecer como color de fondo en formato RGB.
 - Parámetros por referencia: Ninguno.
 - Salida: Ninguno.
- **zoomASeleccion():** método que realiza un zoom sobre el 3D hacia el elemento o elementos que estén seleccionados en la maqueta.
 - Parámetros por valor: Ninguno.
 - Parámetros por referencia: Ninguno.
 - Salida: Ninguno.
- **Referencias:**
 - cColor
 - cPos3D
 - cPropiedad

2.2.2.9 Clase cPos3D

- **Descripción:** implementa el comportamiento de un vector de 3 dimensiones. Permite acceder a cada una de las componentes por separado y se utilizará para manejar las coordenadas de los objetos de la maqueta 3D.
- **Métodos:**
 - **ponX (ByVal dato As Double):** mutador que establece la coordenada X del vector 3D con el valor pasado por parámetro.
 - Parámetros por valor: valor de la coordenada X que se quiere establecer en el objeto.
 - Parámetros por referencia: Ninguno
 - Salida: Ninguna.

- **ponY (ByVal dato As Double):** mutador que establece la coordenada Y del vector 3D con el valor pasado por parámetro.
 - Parámetros por valor: valor de la coordenada Y que se quiere establecer en el objeto.
 - Parámetros por referencia: Ninguno
 - Salida: Ninguna.
- **ponZ (ByVal dato As Double):** mutador que establece la coordenada Z del vector 3D con el valor pasado por parámetro.
 - Parámetros por valor: valor de la coordenada Z que se quiere establecer en el objeto.
 - Parámetros por referencia: Ninguno
 - Salida: Ninguna.
- **dameX () As Double:** accedente que devuelve el valor de la coordenada X del vector 3D.
 - Parámetros por valor: Ninguno.
 - Parámetros por referencia: Ninguno
 - Salida: valor de la coordenada X.
- **dameY () As Double:** accedente que devuelve el valor de la coordenada Y del vector 3D.
 - Parámetros por valor: Ninguno.
 - Parámetros por referencia: Ninguno
 - Salida: valor de la coordina Y.
- **dameZ () As Double:** accedente que devuelve el valor de la coordenada Z del vector 3D.
 - Parámetros por valor: Ninguno.
 - Parámetros por referencia: Ninguno
 - Salida: valor de la coordenada Z.
- **Referencias:**

2.2.2.10 Clase cToolBar

- **Descripción:** está clase se encarga de manejar las barras de herramientas en el formulario de Taller.
- **Métodos:**
 - **ejecutaToolBarGrid(Tool As Object, DBG As DataGrid, DT As Adodc, ByVal Key As String):** método que ejecuta el comando asociado al botón que se ha pulsado en la barra de herramientas de las tablas de datos.
 - Parámetros por valor: *Key*, indica el valor de la propiedad Key del botón que se ha pulsado en la barra de herramientas.
 - Parámetros por referencia:

- *TOOL*, objeto que representa la barra de herramientas que se va a utilizar.
 - *DBG*, objeto que representa el *DataGrid* sobre el que se realizarán las operaciones.
 - *DT*, objeto *Adodc* asociado al *DataGrid*.
 - Salida: Ninguno.
- **ejecutaToolBarNwk(Boton_actual As MSComctlLib.Button, IView As ListView, IViewCoord As ListView):** método que ejecuta el comando asociado al botón que se ha pulsado en la barra de herramientas para Navisworks.
 - Parámetros por valor: Ninguno.
 - Parámetros por referencia:
 - *boton_actual*, objeto que representa el botón que se ha pulsado en la barra de herramientas.
 - *IView*, objeto que representa el control en el que se visualizarán los atributos de los elementos de la maqueta.
 - *IViewCoord*, objeto que representa el control en el que se visualizarán las coordenadas de los elementos de la maqueta.
 - Salida: Ninguno.
 - **inicializaToolBarNwk():** método que se encarga de realizar las operaciones de inicialización necesarias para la barra de herramientas de Navisworks.
 - Parámetros por valor: Ninguno.
 - Parámetros por referencia: Ninguno.
 - Salida: Ninguno.

2.2.3 Diseño de BBDD

La base de datos del sistema se ha protegido con un sistema de seguridad basada en contraseña, la cual puede ser modificada por un usuario con privilegios de administrador. La contraseña cambiada se guarda encriptada en un fichero dentro del directorio de la BBDD llamado bd.pwd. Para generar este fichero se ha utilizado un algoritmo de tipo RSA de clave pública y privada, que utiliza dos parámetros para generar la contraseña encriptada que son: la clave pública y la longitud, que se puede encontrar en el mismo directorio en el fichero claves.txt.

▪ **Diagrama entidad-relación:**

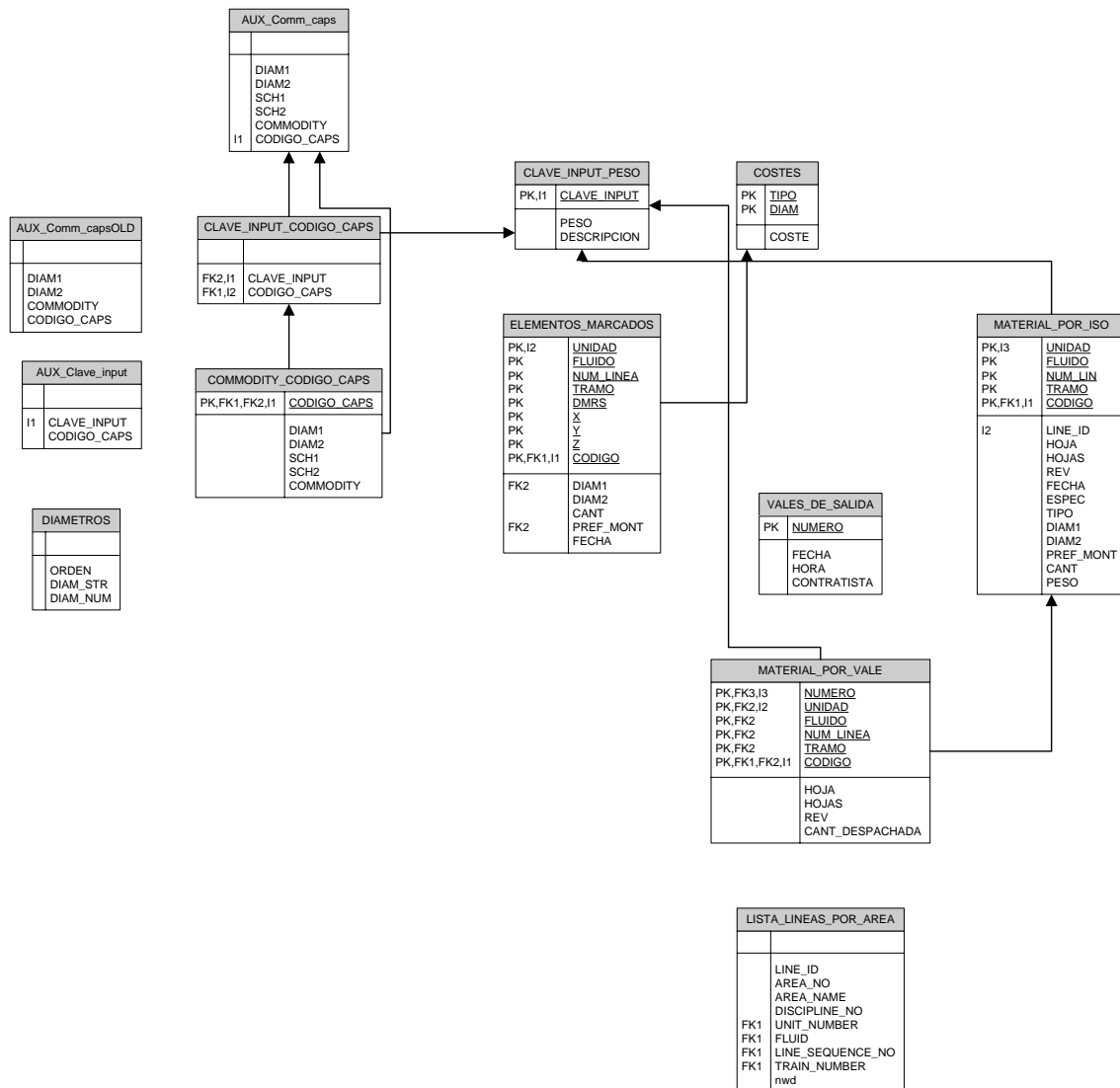
Para el diagrama entidad-relación del Sistema se han clasificado las tablas en dos grupos:

- Las *tablas de Aplicación*: que utiliza el sistema con información propia de la operativa del sistema. Estas tablas almacenan los datos que el sistema debe procesar para lograr los objetivos marcados.
- Las *tablas del Sistema*: que utiliza el sistema para mantenimiento y configuración de diversos aspectos del mismo. Se trata de tablas auxiliares que se serán utilizadas para establecer la personalización del entorno de cada proyecto.

A continuación, se describe el contenido y la función de cada una de estas tablas según su clasificación.

2.2.3.1 Tablas de la aplicación

El diagrama entidad-relación de la parte de aplicación es el siguiente:



Como información y en lugar de establecer cada tipo de datos de los atributos de las tablas, se explicará el significado de cada tabla y las relaciones entre ellas (aún aquellas que se establece por parte de el uso normal de la aplicación).

▪ **MATERIAL_POR_ISO**

Tabla que almacena el conjunto de isométricas que componen el proyecto, y el material que lo constituye. Para cada isométrica son clave los siguientes campos: *Unidad, Fluido, y Num_Lin* (todos de tipo *VarChar*).

Cada isométrica se divide en uno o más tramos, con lo cual, la clave estaría formada por la clave de la isométrica y *Tramo* (También *VarChar*).

Cada tramo está compuesto de uno o más componentes de material diferenciados por el campo *Codigo* (Tipo *VarChar*).

▪ **VALES_SALIDA**

Tabla que almacena los índices de los vales de salida de material que han sido emitidos por el sistema de gestión de materiales y cargados en SCC, en esta tabla se almacena:

- *Numero*, que se establece como clave principal y establece relación implícita en código con la tabla MATERIAL_POR_VALES.
- *Fecha y Hora*, indicadas en el vale origen.
- *Contratista*: Identificador de la empresa a la que se expidió el vale.

▪ **MATERIAL_POR_VALE**

Tabla que establece el material entregado a un contratista en un determinado vale, el material estará identificado con las mismas claves utilizadas en la MATERIAL_POR_ISO y el contenido de la tabla deberá ser coherente con el de la tabla citada.

▪ **ELEMENTOS_MARCADOS**

Esta tabla relaciona los elementos cargados en el sistema por medio de ficheros con los elementos utilizados en la maqueta 3D, por ello necesitamos los campos clave de la tabla MATERIAL_POR_ISO para identificar los elementos a la cual añadimos la parte específica de posición en la maqueta como son los campos X,Y,Z y el campo DMRS. Aparte de lo ya indicado, se recupera los datos referentes a diámetros de la tabla catálogo de diámetros para cada elementoAUX_Comm_caps.

Se establecen los parámetros de *fecha* en el que se almacena la fecha de la modificación y *Pref_Mont* para establecer la categoría del elemento.

▪ **COSTES**

Tabla que establece el coste por metro o por pieza, dependiendo del elemento, relacionado con su diámetro

▪ **COMMODITY_CODIGO_CAPS**

Esta entidad relaciona los códigos de materiales generados por el sistema de materiales con los códigos de materiales utilizados en la maqueta 3D, es una tabla generada externamente y cargada en el sistema íntegramente. El campo *COMMODITY* es clave para el Sistema de de la maqueta 3D; y el campo *CODIGO_CAPS* es clave para el sistema de materiales.

▪ **Aux_Comm_Caps y Aux_Comm_Caps_Old**

Son tablas creadas ex profeso para la relación de materiales y 3D cuando algunos de los 2 sistemas no es o bien NavisWorks o el Sistema de Materiales no es MATUB. Se utilizan campos necesarios para cada uno de los sistemas necesarios.

Estas tablas son tratadas como extensiones de la tabla *COMMODITY_CODIGO_CAPS*.

▪ **LISTA_LINEAS_POR_AREA**

Usada para determinar el fichero de la maqueta 3D en el que se encuentra cada Isométrica o línea. El campo *nwd* contiene el nombre del fichero NavisWorks en el que debe encontrarse.

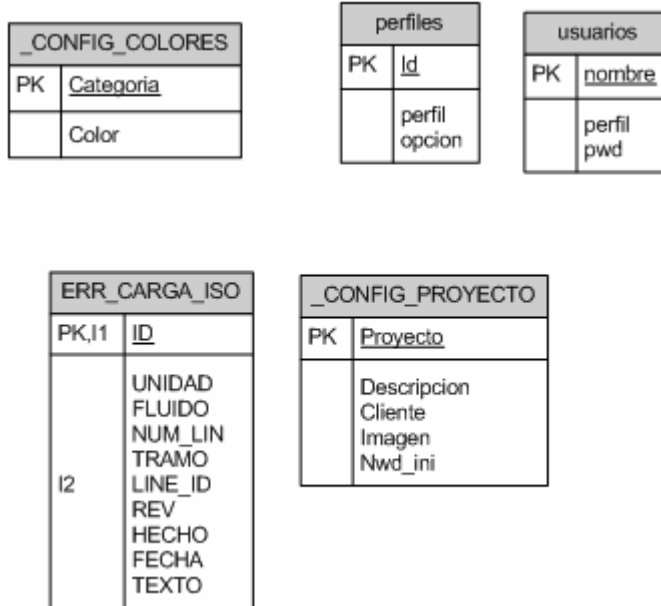
▪ **DIAMETROS**

Se utiliza como catálogo para establecer las relaciones posibles entre los diámetros de los materiales del Sistema. Se utiliza como Clave el campo *Orden*, generado de forma externa a la propia tabla.

Estas tablas son utilizadas por la aplicación para su trabajo normal,

2.2.3.2 Tablas Auxiliares

En este apartado tenemos tablas de gestión del sistema, no relacionadas mediante relaciones SQL pero con las siguientes características:



▪ **_CONFIG_COLORES:** En esta tabla se guarda el color asignado en el formulario de colores a los materiales de cada categoría (Montado o Prefabricado).

▪ **ERR_CARGA_ISO:** Tabla destinada a guardar errores relacionados con procesos de cargas de isométricas o vales, así como de errores en borrados o actualizaciones de los mismos.

En ella se especifica la Isométrica que ha dado el error y un texto descriptivo del evento sucedido.

En el campo *HECHO*, se especifica S/N, indicando S si se ha revisado el error o N si todavía está pendiente de revisión.

▪ **_CONFIG_PROYECTO:** Tabla de Configuración del sistema, en ella se establecen los parámetros generales del proyecto:

- **Descripción:** Se establece el resumen del proyecto.
- **Cliente:** Identificador del cliente del proyecto.
- **Imagen:** Ruta de la imagen a mostrar en la pantalla principal.
- **Nwd_ini:** Ruta de la maqueta 3D en formato legible por Naviswork sobre la que trabajará la aplicación.
- **Perfiles:** Se establecen las opciones disponibles para cada perfil.

- **Usuarios:** Almacena el nombre del usuario, el perfil de la tabla perfiles al que está asignado y el valor hash de la contraseña (contraseña cifrada) para comparar en el inicio de la aplicación.

3 Ampliaciones y Conclusiones

3.1 Sistema Gestor de Base de Datos

El sistema se ha apoyado en una base de datos comercial, de fácil instalación y ampliamente difundida como es Microsoft Access. No obstante, esta base de datos no garantiza la seguridad requerida en un entorno de trabajo industrial y es probable que para un volumen de trabajo real se vea desbordada.

Estos motivos suscitaron la creación de una clase *cBaseDatos*, que gestione la interfaz con el sistema gestor de bases de datos, facilitando la migración a cualquier otro sistema con solo cambiar esta clase.

Los métodos públicos de esta clase, como son: *inserta*, *borra* o *modifica*, en la implementación actual tan sólo ejecutan las sentencias SQL que se les pasa por parámetro, pero debido a la diferenciación y al buen uso de la clase, el cambiar a otro sistema supone los siguientes trabajos:

- Gestión de las conexiones adecuadamente para el Sistema Gestor de Bases de Datos seleccionado.
- En caso de que el Sistema no implemente SQL o tenga diferencias sustanciales con el SQL empleado por Access, se deberá implementar una traducción de las SQL admitidas como parámetros de la clase actual a una implementación SQL del Sistema seleccionado.

Esta separación en una clase aparte de todo el acceso a base de datos, también permite el acceso a bases de datos centralizadas si se tiene previsto conexión de datos de banda ancha con el emplazamiento físico del sistema, modificando la clase actual por otras que incorpore:

- Procedimientos de acceso a bases de datos remotas.
- Nivel de seguridad acorde a la información que se va a utilizar.

3.2 Visual Basic 6

En la elección del lenguaje de programación a utilizar en la realización de SCC, se han evaluado diferentes aspectos, siendo el lenguaje de programación Visual Basic 6 el que más se adecuaba al proyecto en cuestión.

Como características favorables de Visual Basic 6, podemos destacar:

- Facilidad de aprendizaje: En este apartado se debía tener en cuenta que parte del equipo desarrollador no conocía esta tecnología.
- Facilidad en el Manejo de Bases de Datos por medio de Recordsets.
- Integración de API del sistema de visualización de maquetas 3D: El sistema de visualización de maquetas 3D utilizado (JetStream NavisWorks), provee una API propia del programa implementada en Visual Basic 6, con lo que la interacción con ella era requisito indispensable.

En cuanto a desventajas y problemas destacables de este lenguaje de programación, se pueden mencionar los siguientes:

- Aunque el análisis del sistema y el desarrollo se ha realizado siguiendo el paradigma de Orientación a Objetos, este lenguaje de programación no provee las características típicas de estos lenguajes, por lo que se han debido buscar artificios para realizar técnicas típicas como puede ser la implantación de patrones (Vista-Controlador, Singleton...), encapsulación de clases, etc.
- Se ha puesto de manifiesto la problemática de la gestión de errores entre módulos-clases, por ello se ha tenido que implementar un sistema de gestión de errores basado en ficheros logs y lectura de los mismos.

3.3 Visualizador 3D

La elección del visualizador de la maqueta 3D ha estado sujeta a los requerimientos del cliente, que en este caso utiliza mayoritariamente Navisworks. Este visualizador tiene la ventaja de que permite manejar maquetas 3D generadas utilizando como herramienta de diseño PDS (de Intergraph) o PDMS (de Aveva) indistintamente.

Sin embargo, en el futuro, no se descarta la posibilidad de utilizar otro visualizador comercial, denominado Smart Plant Review, que suministra Intergraph y que ofrece mayores posibilidades cuando se trabaja con PDS.

Teniendo en cuenta esta posibilidad, se ha realizado el diseño, de forma que los cambios que se tienen que introducir, en caso de que se quiera optar por esta mejora, están perfectamente definidos y no suponen un impacto en el resto de la aplicación. Estos cambios consisten en:

- Crear una nueva clase, análoga a la clase cNavisworks que existe actualmente, que implemente los métodos necesarios para poder interactuar con la maqueta.
- Añadir las funcionalidades necesarias en el formulario Taller utilizando los métodos de la nueva clase.

De esta manera, resulta relativamente sencillo ampliar el sistema a nuevos visualizadores, o bien, adaptarlo a las nuevas funcionalidades que incorporen versiones futuras del visualizador para el que ha sido implementada la aplicación.

3.4 Integración con herramientas de planificación de proyectos

En el futuro se puede plantear la integración con una herramienta de planificación de tareas, como puede ser Microsoft Project o Primavera. Esta integración permitiría introducir las siguientes mejoras:

- Actualmente los informes reflejan el progreso real de la construcción con respecto al total diseñado. Con la integración de una herramienta de este tipo se obtendría la posibilidad de comparar el progreso real con el planificado. De esta forma, sería posible generar informes referentes tanto al progreso real como al progreso planificado. Y sería posible también obtener comparativas entre ambos.
- Desde el visualizador de la maqueta 3D se podría realizar una simulación del avance de la construcción, en un determinado periodo de tiempo, utilizando como base la planificación realizada en la herramienta correspondiente.
- Realizar una comparación visual sobre la maqueta 3D entre el progreso real de la construcción y el planificado. Al enlazar los objetos del 3D con las tareas planificadas sería posible reflejar gráficamente los retrasos y/o las diferencias en los que se ha incurrido durante la construcción de la instalación industrial.

4 Bibliografía

Libros

- [Booch99] **El Lenguaje Unificado de Modelado**. G. Booch, J. Rumbaugh, I. Jacobson. Addison Wesley Iberoamericana, 1999.
- [Booch94] **Object-Oriented Analysis and Design**. G. Booch. Benjamin/Cummings, 1994.
- [BJR97] **The UML Specification Document**. G. Booch, I. Jacobson and J. Rumbaugh. Rational Software Corp., 1997.
- [Jacobson92] **Object-Oriented Software Engineering: A Use Case Driven Approach**. I. Jacobson. Addison-Wesley, 1992.
- [Larman99] **UML y Patrones**. C. Larman. Prentice Hall, 1999.
- [Rumbaugh91] **Object-Oriented Modeling and Design**. J. Rumbaugh et al. Prentice-Hall, 1991.
- [Ceballos97] **Enciclopedia de Microsoft Visual Basic 6**. Fco. Javier Ceballos. RA-MA 1997

Páginas Web

- <http://www.vbasic.astalaweb.com>
- <http://www.recursosvisualbasic.com>
- <http://www.caesoft.es>
- <http://www.navisworks.com>
- <http://www.trsa.es>
- <http://msdn.microsoft.com/es-es/vbasic/>
- <https://customer.navisworks.com/slxweb/portal/portalKnown.php>

Anexo A. Ejemplo de Informe de Avance

1 Informe de avance en Kg

- Hoja Resumen

	PESO_NECESARIO	PESO_CONSTR	DIFERENCIA	AVANCE
H46:	911892.669	2225.24464	909667.4244	0%
H48:	2376418.503	187.06	2376231.443	0%
TOTAL	3288311.172	2412.30464	3285898.867	0.1%

- Hojas de detalle totalizado de cada unidad

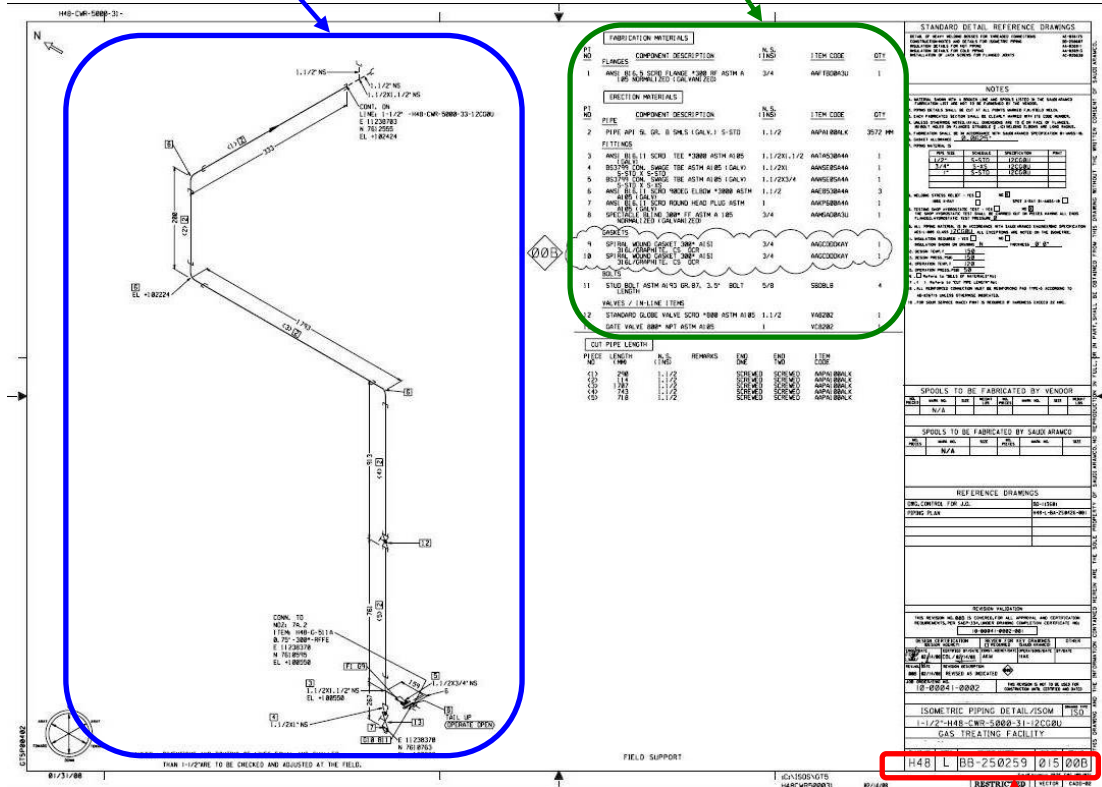
LINE_ID	CODIGO	DESCRIPCION	DIAM1	DIAM2	PREF	MONT	CANT	NECE	CANT	P	M	DIFERENCIA	AVANCE
13	Total H46-A-0001-01						94.7	0	94.7			0%	
22	Total H46-A-0001-02						19.9	0	19.9			0%	
28	Total H46-A-0001-03						6.2	6.2	0			100%	
36	Total H46-A-0001-04						21	0	21			0%	
48	Total H46-A-3650-01						30.9	0	30.9			0%	
52	Total H46-A-3650-02						9	0	9			0%	
56	Total H46-A-3650-03						26	0	26			0%	
63	Total H46-A-3650-04						17.7	0	17.7			0%	
71	Total H46-A-7700-01						34.5	34.5	0			100%	
72	H46-A-7700-E0000027	GATE VAL V3/4			M		2	2	0			100%	
73	H46-A-7700-E00000541	PIPE Sch. S2"			M		59	59	0			100%	
74	H46-A-7700-E00000737	ANSI B16 .111"			M		1	1	0			100%	
75	H46-A-7700-E00000744	ANSI B16 .112"			M		8	8	0			100%	
76	H46-A-7700-E00000755	ANSI B16 .113/4			M		2	2	0			100%	
77	H46-A-7700-E00000765	ANSI B16 .112"	3/4		M		2	2	0			100%	
78	H46-A-7700-E00000766	ANSI B16 .112"	1"		M		1	1	0			100%	
79	H46-A-7700-E00000986	NIPPLE 1 0013/4			M		2	2	0			100%	
80	H46-A-7700-E00001118	BS7399 C Ot2"	1"		M		1	1	0			100%	
81	Total H46-A-7700-81						78	78	0			100%	
82	H46-A-7700-E0000027	GATE VAL V3/4			M		2	0	2			0%	
83	H46-A-7700-E00000044	PISTON C HE1"			M		1	0	1			0%	
84	H46-A-7700-E00000118	BALL VAL VE1"			M		1	0	1			0%	
85	H46-A-7700-E00000538	PIPE Sch. X3/4			M		0.1	0	0.1			0%	
86	H46-A-7700-E00000539	PIPE Sch. S1"			M		2.2	0	2.2			0%	
87	H46-A-7700-E00000747	ANSI B16 .111"			M		1	0	1			0%	
88	H46-A-7700-E00000755	ANSI B16 .113/4			M		1	0	1			0%	
89	H46-A-7700-E00000759	ANSI B16 .111"	3/4		M		2	0	2			0%	
90	H46-A-7700-E00000760	ANSI B16 .111"			M		1	0	1			0%	
91	H46-A-7700-E00000986	NIPPLE 1 0013/4			M		1	0	1			0%	
92	H46-A-7700-E00001115	BS7399 C Ot1"	3/4		M		2	0	2			0%	
93	H46-A-7700-E00001148	ANSI B16 .113/4			M		1	0	1			0%	
94	H46-A-7700-E00003256	HOSE CON 13/4			M		1	0	1			0%	
95	Total H46-A-7700-82						16.3	0	16.3			0%	
99	Total H46-A-7700-83						30	0	30			0%	
113	Total H46-A-7700-84						14.8	0	14.8			0%	
126	Total H46-A-7700-85						19.5	0	19.5			0%	

Anexo B. Ejemplo de Isométrica

En la siguiente figura se indican los componentes de una isométrica, que resultan de interés para el desarrollo del Sistema de Control de Construcción.

Vista Isométrica de la Línea (o tramo) de Tubería

Lista de Materiales



Identificación de la Isométrica

Anexo C. Manual de Usuario

1 Introducción

A continuación se va a describir el funcionamiento del Sistema de Control de Construcción desde el punto de vista del usuario final. Es necesario tener en cuenta que el acceso a la aplicación se realiza de acuerdo a un perfil de usuario que, por defecto, puede ser: *administrador del sistema*, *usuario de taller* y *usuario de Dirección de Construcción*. En función del perfil al que esté asociado el usuario que acceda al sistema, podrá disponer de algunas o de todas las funcionalidades que se detallan en este documento. La definición de los usuarios la realizará el administrador del sistema.

Es importante tener en cuenta que desde la aplicación sólo se podrá acceder a un proyecto. Esta consideración responde a la necesidad de contar con una herramienta flexible que pueda ser configurada y adaptada a las necesidades de cada proyecto, según el cliente que realice la contratación de la instalación industrial. El sistema cuenta con una plantilla que servirá como base para todos los proyectos, incluso será posible utilizar nuevas plantillas a partir de otros proyectos según las necesidades.

El objetivo del sistema consiste en ofrecer, al contratista, una interfaz con la maqueta electrónica 3D que le permita indicar sobre ésta cuál es el avance de la prefabricación y el montaje de las Isométricas que le han sido entregadas. Esta información será tratada por el sistema de tal forma que sea posible, desde la Dirección de Construcción, calcular el porcentaje de avance del trabajo realizado por el contratista con el fin de obtener tanto su coste económico, como el peso del material prefabricado/montado. El peso resulta importante porque es la unidad que se utiliza para medir la dimensión de este tipo de proyectos.

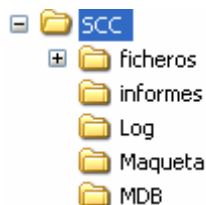
2 Consideraciones previas

Las consideraciones que se van a detallar en este punto, se deben tener en cuenta siempre antes de ejecutar el programa por primera vez, y se deben tener en cuenta para todos los proyectos.

La configuración que se establece en este apartado, debe ser realizada exclusivamente por el administrador del sistema.

2.1 Estructura de carpetas

El sistema está diseñado para trabajar con una base de datos por proyecto. Asociada a cada proyecto, también es necesario mantener una determinada estructura de carpetas que, en principio, debe ser fija. La estructura de carpetas debe ser la siguiente:



Esta estructura de carpetas puede ser creada en un servidor, si se dispone de él, o bien, en la propia máquina donde se ha realizado la instalación del sistema. En cualquiera de los dos casos será necesario disponer de un mecanismo de backups diarios que permita recuperar un estado anterior en caso de ser preciso.

A continuación se explica qué debe contener cada una de las carpetas que forman parte de la estructura de carpetas del sistema SCC.

- **SCC:** es la carpeta raíz de la aplicación, en ella residirá el fichero que lanza la ejecución del sistema, es decir, el **SCC.exe**. El usuario puede establecer el nombre que considere para esta carpeta sin afectar al correcto funcionamiento del sistema.
- **Ficheros:** se creará dentro de la carpeta raíz. Se trata de una carpeta de entrada. En ella se ubicarán los ficheros correspondientes a los informes generados desde MATUB, que pueden ser importados al sistema.
- **Informes:** se creará dentro de la carpeta raíz. Se trata de una carpeta de salida. Esta es la carpeta en la que, por defecto, el sistema genera los informes de avance cuando se elige la opción Informes desde el menú Dir. Construcción. **El nombre de esta carpeta no se debe cambiar.**
- **Log:** se creará dentro de la carpeta raíz. Se trata de una carpeta de salida. En esta carpeta el sistema guardará ficheros .log acerca de los usuarios que han accedido al sistema así como de los posibles errores que se puedan producir durante la ejecución del programa. **El nombre de esta carpeta no se debe cambiar.**

- **Maqueta:** se creará dentro de la carpeta raíz. Se trata de una carpeta de entrada. En esta carpeta se ubicarán los ficheros de la maqueta electrónica 3D (.nwd), correspondientes al proyecto. **El nombre de esta carpeta no se debe cambiar.**
- **MDB:** En esta carpeta se ubicará la base de datos MS Access que utiliza el sistema. **El nombre de esta carpeta no se debe cambiar.**

2.2 Configuración regional

Para el correcto funcionamiento de la aplicación, es necesario contar con la siguiente configuración regional:

- Separador decimal: . (punto).
- Separador de miles: , (coma).

En caso de que el equipo no esté configurado de esta manera, el sistema mostrará el siguiente mensaje de error y no se continuará con la ejecución de la aplicación.

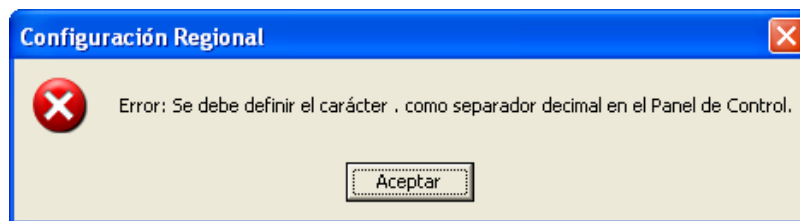


Imagen 1

3 Menú Principal

Al ejecutar el programa con la configuración adecuada, y antes de acceder al formulario principal, aparece el formulario de Acceso Usuario en el que hay que introducir el usuario y contraseña para que el menú principal muestre las opciones a las que tendrá acceso en el sistema un determinado usuario.

En una instalación nueva, el usuario por defecto será Administrador y la contraseña Administrador, se aconseja el cambiar esta contraseña la primera vez que se acceda al sistema.

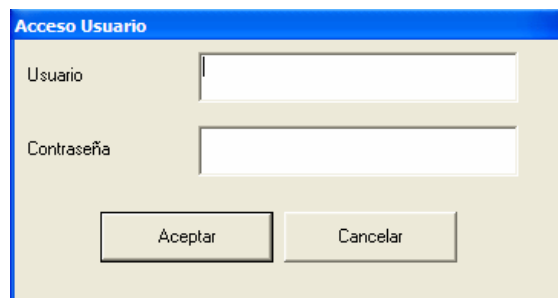


Imagen 2

Después de pulsar el botón de Aceptar aparece el menú principal con las opciones que correspondan al perfil del usuario introducido:



Imagen3

- **Descripción:** Es el formulario principal y se compone del siguiente menú:
 - **Administración:** tiene como submenú:
 - Gestión Usuarios, que se utiliza para gestionar a que opción puede acceder cada usuario y hace una llamada al formulario Gestion_Usuario.
 - Configurar proyecto opción que sirve para configurar el proyecto.
 - Borrar Vales que hace una llamada al formulario Borrar_Vale.
 - Borrar Isométricas que hace una llamada al formulario Borrar_ISO.
 - **Taller:** tiene como submenú:
 - Marcar Realizado opción que se utiliza para hacer una llamada al formulario REALIZADO_TALLER.
 - **Herramientas:** tiene como submenú Importar y Configurar.
 - Importar es la opción que se utiliza para hacer la carga de los ficheros .rtf, teniendo en cuenta que estos pueden ser *Isométricas* o *Vales de salida*, también tiene la opción de cargar *Vales de forma masiva* y hacer *Gestión de carga de Isométricas* que hace una llamada al formulario Geston_Carga.
 - Configurar opción para configurar *Colores* que hace una llamada al formulario Colores.
 - **Dir. Construcción:** Tiene como submenú Coste e Informes.
 - Coste es la opción que sirve para introducir el coste de los materiales y llama al fomulario Costes.
 - Informes que genera informes ya sea por *Avance por línea en kg* y por *Avance por línea en coste* y hace una llamanda al formulario Elegir Unidad.
 - **Salir:** opción para salir del formulario

4 Menú Administración

4.1 Gestión de Usuarios

La gestión de usuarios del sistema se realiza mediante usuarios y perfiles. Dependiendo del perfil al que esté asociado, un usuario tiene acceso o no a diversas acciones.

Se debe tener en cuenta que:

- Un usuario debe estar asociado obligatoriamente a uno y sólo un perfil.
- Un perfil no se puede borrar si tiene usuarios asociados.

Para acceder a la gestión de usuarios se debe acceder desde el menú principal: **Administración -> Gestión de Usuarios**



Imagen 4

A continuación, se abre la pantalla de Gestión de Usuarios:

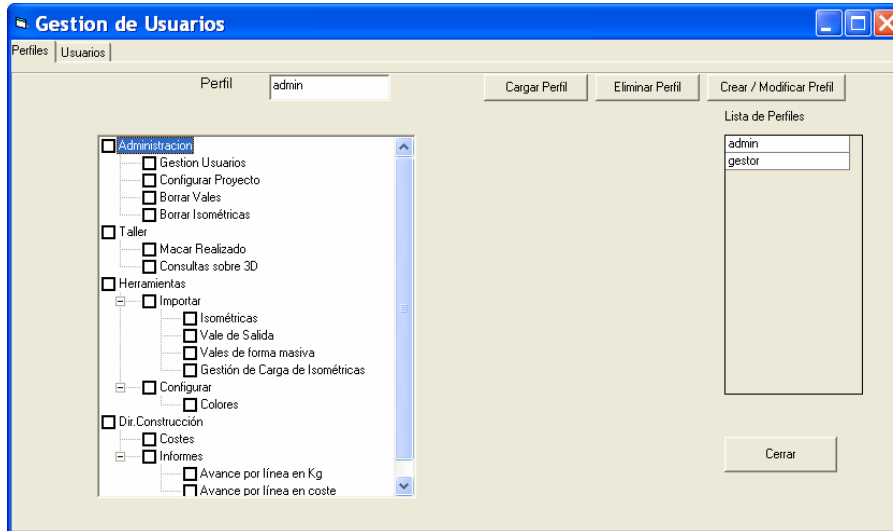


Imagen 5

En esta pantalla se pueden realizar las siguientes operaciones:

- **Cargar Perfil:** Se activan en el árbol las opciones permitidas al perfil seleccionado.
- **Eliminar Perfil:** Se elimina el perfil si no existe ningún usuario vinculado
- **Crear / Modificar Perfil:**
 - Si el nombre de perfil no existe, se genera uno nuevo con las opciones activadas.
 - Si el perfil existe, se modifican las acciones permitidas a las activadas en ese momento.

La Pestaña Usuarios da acceso a las operaciones básicas con usuarios:

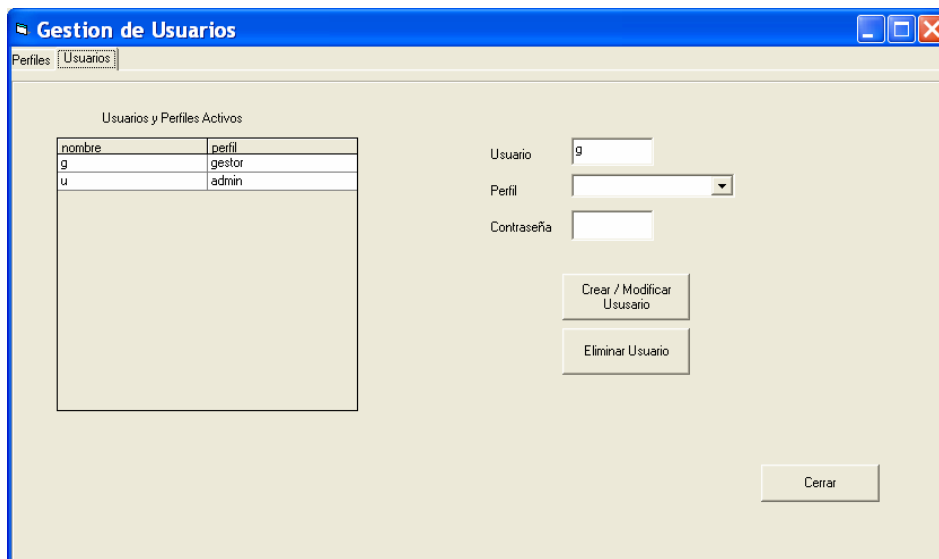


Imagen 6

En esta pantalla, podemos realizar las siguientes operaciones:

- **Crear / Modificar Usuario:**
 - Si el usuario existe se guarda con el perfil seleccionado y con la contraseña introducida (puede ser vacía).
 - Si el usuario no existe, se crea uno nuevo asignado al perfil seleccionado y con la contraseña introducida (puede ser vacía).
- **Eliminar Usuario:** Borra el Usuario del Sistema.

4.2 Configuración del Proyecto

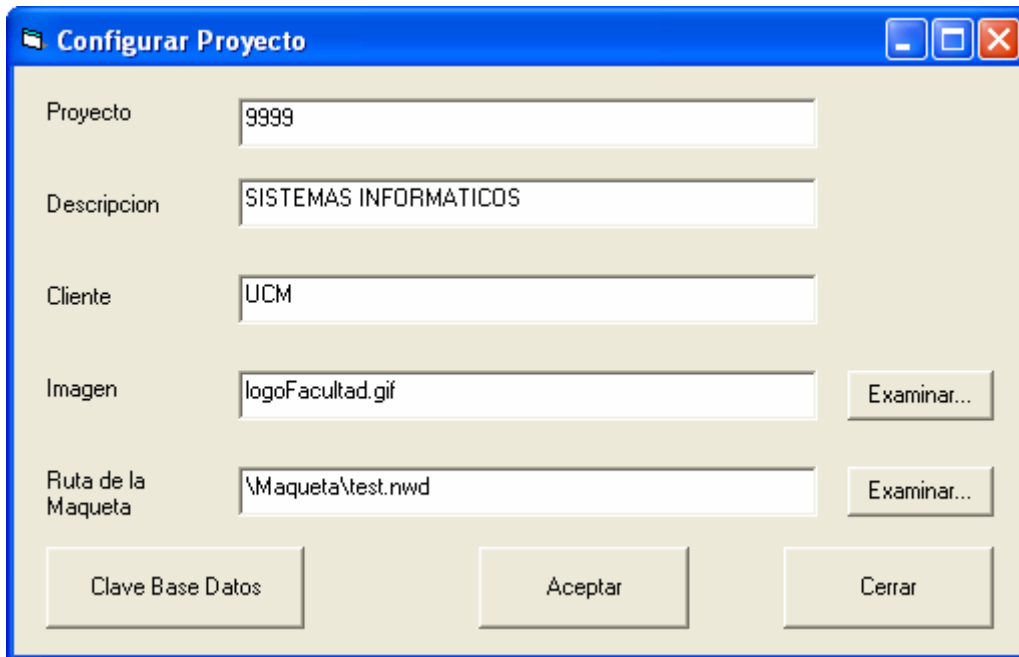


Imagen 7a

En esta opción se puede realizar la gestión de los parámetros del proyecto especificados a continuación:

- Proyecto: Es un texto que representa el código que identifica el proyecto, **solo debe cambiarse antes de comenzar el proyecto.**
- Descripción: En este campo se visualiza una descripción asociada al código de proyecto anterior.
- Cliente: Especifica el cliente del proyecto, **sólo debe cambiarse antes de comenzar el proyecto.**
- Imagen: En este campo se especifica la imagen que aparece en el fondo del formulario principal, al pulsar en Examinar aparece un cuadro de dialogo para facilitar la búsqueda del fichero, se puede filtrar entre jpg, bmp y todos los ficheros. El fichero especificado debe estar dentro del conjunto de subdirectorios del proyecto.
- Ruta de la Maqueta: Se debe especificar la ruta donde se encuentra el fichero clave de la maqueta, este debe ser compatible con la herramienta JetStream NavisWorks. El fichero especificado debe estar dentro del conjunto de subdirectorios del proyecto.
- Gestión de la clave de la Base de Datos.

Si se ha cambiado alguna de las opciones, al pulsar el botón Aceptar aparece un cuadro de diálogo que nos preguntará si queremos guardar la configuración.

Al pulsar el botón *Clave Base Datos* se abre un formulario que genera la clave de conexión a la Base de Datos del Sistema, esta Opción solo debe ser utilizada si se tiene permiso para cambiar la clave de la Base de Datos.

- **Esta Opción debe ser ejecutada antes de realizar ninguna otra operación con el sistema, en caso de haber realizado alguna operación, será necesario reiniciar la Aplicación antes de continuar.**
- **Para realizar esta operación no debe estar abierta ni accedida la Base de datos por ningún otro programa.**
- **Se aconseja hacer una copia de seguridad del fichero bd.pwd antes de realizar este proceso.**

El proceso de cambio de clave es el siguiente:

- Al pulsar el botón se abre el siguiente formulario:

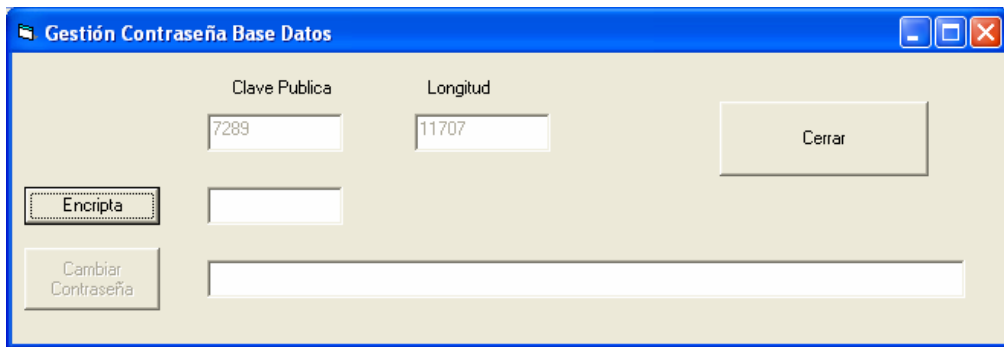


Imagen 7b

- Este formulario indica la clave pública que se debe utilizar para generar la contraseña (en caso de generarse externamente al Sistema) y la longitud de la misma.
- Se debe introducir la clave nueva en el cuadro de texto y pulsar el botón Encripta, lo cual generará una clave cifrada en el cuadro de texto inferior y habilitará el botón Cambiar Contraseña.
- Una vez generada la clave cifrada se debe pulsar Cambiar Contraseña, lo cual generará el fichero bd.pwd en el directorio MDB de la aplicación y modificará la contraseña de la base de datos.
- **NO SE DEBE MODIFICAR ESTE FICHERO GENERADO.**
- En caso de surgir algún error, se deberá sobrescribir el fichero bd.pwd con la copia de seguridad salvada anteriormente.

Al pulsar el botón Cerrar se cierra la ventana y se vuelve al menú anterior.

4.3 Borrar Vales

En esta opción se puede realizar el borrado de vales cargados mediante los procesos de carga normal y de carga masiva.

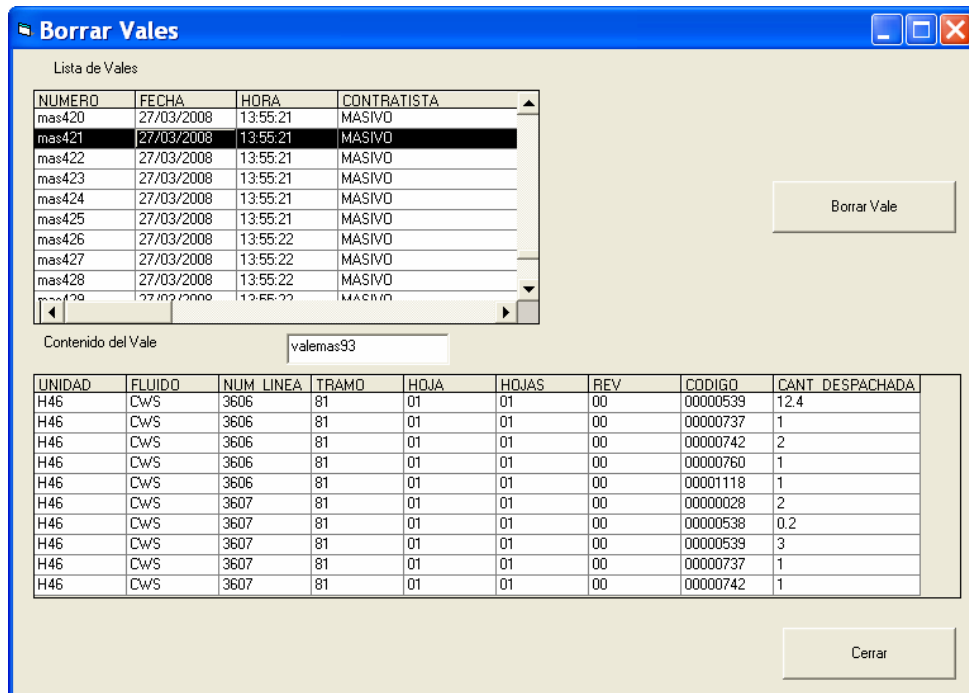


Imagen 8

En la ventana abierta se visualizan todos los vales cargados en el sistema en la Lista de Vales. Del vale marcado en negro se visualiza, en la parte inferior de la pantalla, el contenido del vale. Esta información es solamente informativa ya que no se puede modificar.

Si presionamos el botón Borrar Vale, se realizará el borrado del vale y de todo el material asociado a dicho vale que se encuentre en la base de datos del sistema. No se borra el material asociado a la Isométrica.

4.4 Borrar Isométrica

Esa opción abre la ventana de Borrar Isométricas, desde la cual se puede borrar una isométrica cargada en el sistema.

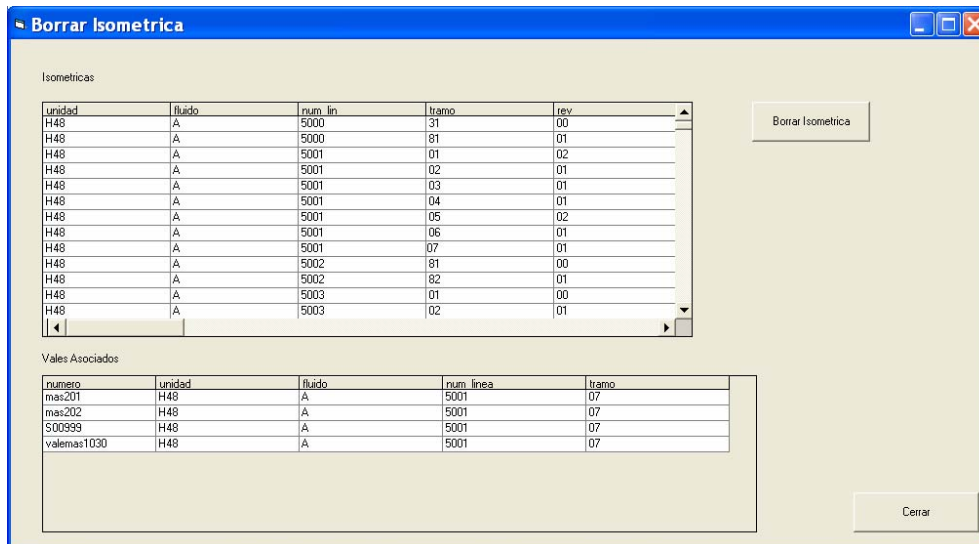


Imagen 9

Al borrar la isométrica, se borrará también todo el material cargado en el sistema asociado a esa isométrica. Al realizar esta acción, el sistema buscará si se han cargado vales sobre este material y, en caso afirmativo, se genera una línea de advertencia indicando la isométrica y el vale en el que se ha generado la inconsistencia. Esta línea se podrá consultar en la herramienta de Gestión de Cargas del menú Herramientas, Importar.

Al seleccionar una isométrica de la parte superior de la pantalla, se cargarán los vales asociados a esa isométrica en la parte inferior de la pantalla. En caso de borrar la isométrica seleccionada, se generarán las líneas de advertencia para los vales abajo indicados.

5 Menú Taller

5.1 Marcar Realizado



Imagen 10

A esta opción tienen acceso todos los perfiles de usuario definidos por defecto. Sin embargo, los usuarios destinados a utilizarla son los usuarios asignados al perfil denominado *usuario del Taller*.

Al seleccionar esta opción desde el menú principal, se accede al siguiente formulario:

5.1.1 Elementos del formulario

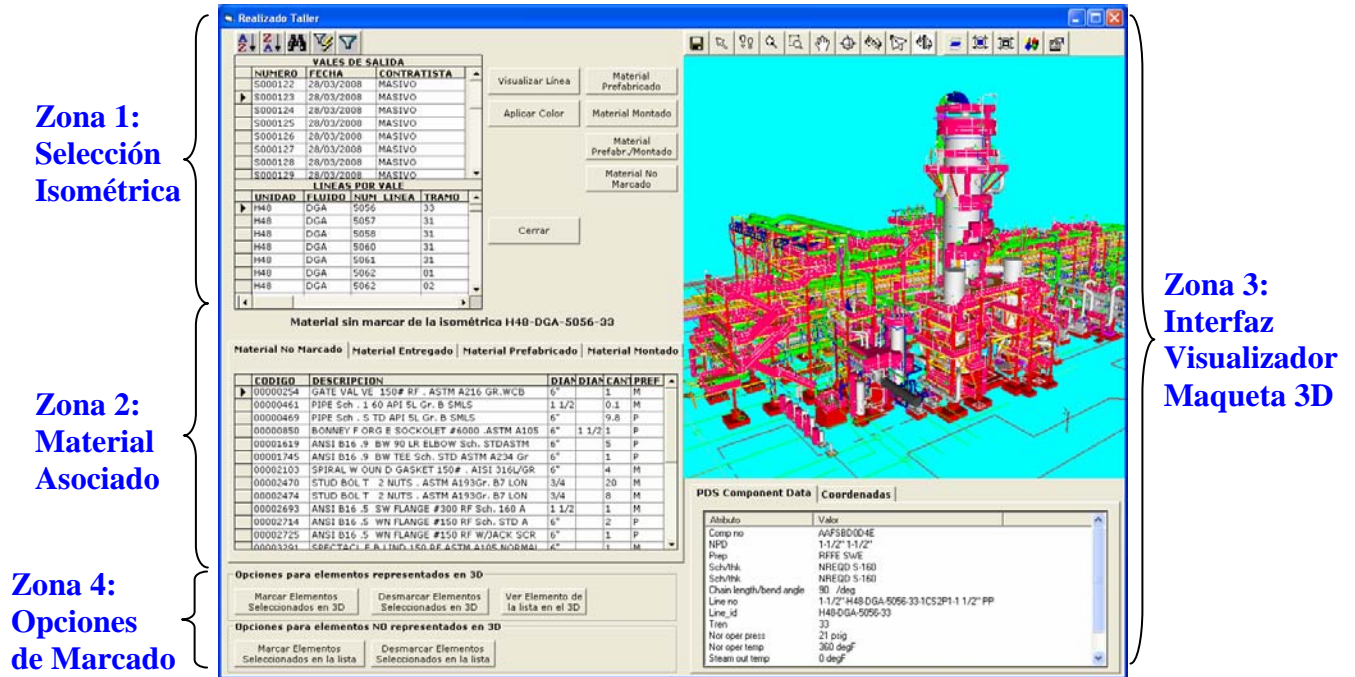


Imagen 11

En el formulario asociado a la opción *Marcar Realizado* se pueden distinguir las zonas indicadas en la imagen, que han sido agrupadas según su funcionalidad. A continuación, se describe cada una de ellas:

- Zona 1, Selección de la Isométrica:** en esta zona se encuentran las herramientas que permiten al usuario seleccionar la Isométrica sobre la que desea trabajar. Consta de los siguientes elementos:
 - Barra de herramientas:** para utilizarla, es necesario activar mediante un clic la tabla de datos sobre la que se desea trabajar (tabla de vales o tabla de líneas). En caso de que no se encuentre activa ninguna tabla aparecerá el siguiente mensaje:

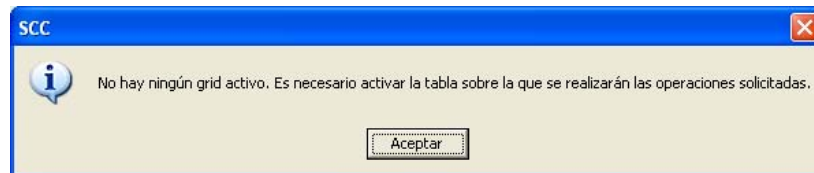





Imagen 12

Las opciones de esta barra de herramientas son las siguientes:

1.  **Orden Ascendente:** Permite ordenar ascendentemente la columna de datos seleccionada.
2.  **Orden Descendente:** Permite ordenar descendientemente la columna de datos seleccionada.
3.  **Buscar:** Permite realizar una búsqueda sobre la columna de datos activa. En el siguiente formulario, se deberá especificar el criterio de búsqueda. Es posible utilizar comodines como se indica:

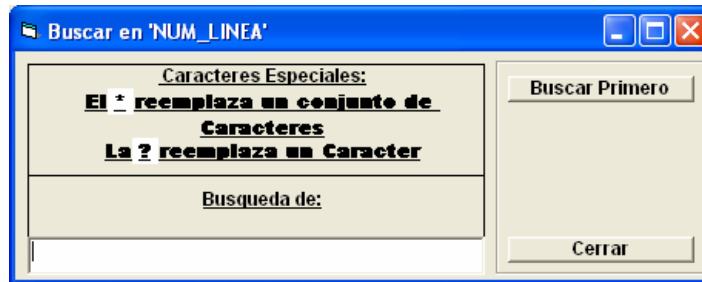




Imagen 13

Una vez introducido el valor buscado, se debe pulsar el botón *Buscar Primero*. Si el valor es encontrado, el selector de registro de la tabla activa se situará sobre el primer registro que cumple el criterio.

4.  **Filtro por Selección:** Permite realizar un filtro por selección. El filtro por selección consiste en seleccionar uno o varios caracteres del valor de un campo y, pulsando esta opción, utilizarlos como criterio de búsqueda automáticamente. El filtro tiene efecto sobre la tabla activa de forma que sólo mostrará los registros que cumplan el criterio seleccionado.
5.  **Quitar Filtro:** Permite desactivar un filtro por selección previamente aplicado. De esta forma, se visualizarán de nuevo todos los registros en la tabla activa.

- **Tabla de Vales de Salida:** contiene cada uno de los vales de salida que han sido cargados en el sistema. Para cada uno de ellos se mostrarán, en la tabla de líneas, las isométricas que tiene asociado. Así, el usuario podrá localizar la isométrica sobre la que debe realizar el marcado.

VALES DE SALIDA			
	NUMERO	FECHA	CONTRATISTA
	S000103	28/03/2008	MASIVO
	S000104	28/03/2008	MASIVO
	S000105	28/03/2008	MASIVO
	S000106	28/03/2008	MASIVO
▶	S000107	28/03/2008	MASIVO
	S000108	28/03/2008	MASIVO
	S000109	28/03/2008	MASIVO
	S000110	28/03/2008	MASIVO

Imagen 14

- **Tabla de Líneas por Vale:** contiene las isométricas asociadas al vale seleccionado en la tabla de vales de salida. Al seleccionar un registro, automáticamente, se mostrará su lista de material en la zona habilitada para ello que se explica a continuación.

LINEAS POR VALE				
	UNIDAD	FLUIDO	NUM LINEA	TRAMO
▶	H48	DGA	5022	31
	H48	DGA	5022	32
	H48	DGA	5022	81
	H48	DGA	5022	82
	H48	DGA	5022	83
	H48	DGA	5022	84
	H48	DGA	5022	85

Imagen 15

- **Zona 2, Material Asociado,** se trata de una zona de visualización de datos en formato tabular. En ella se carga la información relativa al material correspondiente a la línea (o Isométrica) seleccionada en la tabla de Líneas por Vale. El material se ha clasificado en pestañas de la siguiente forma:

Material No Marcado		Material Entregado		Material Prefabricado		Material Montado	
	CODIGO	DESCRIPCION	DIAN	DIAN	PREF	CANT	
▶	00000322	GATE VAL VE 150# RF . ASTM A216 Gr.WCB (3"		M	1	
	00000491	PIPE Sch . X S API 5L Gr.B SMLS(NACE)	24"		P	13.1	
	00000497	PIPE Sch . X XS API 5L Gr.B SMLS (NACE)	2"		P	0.1	
	00000881	BONNEY F ORG ED WELDOLET Sch. XS x XXS AS	24"	2"	P	2	
	00000883	BONNEY F ORG ED WELDOLET Sch. XS x 160 AS	24"	3"	P	1	
	00001809	ANSI B16 .9 BW 45 ELBOW Sch. XS ASTM A23	24"		P	1	
	00001846	ANSI B16 .9 BW ECC REDUCER Sch. XS ASTM	24"	16"	P	1	
	00001890	ANSI B16 .9 BW TEE Sch. XS ASTM A234 Gr.	24"		P	1	
	00002095	SPIRAL W OUN D GASKET 150# . AISI 316L/GR	24"		M	2	
	00002100	SPIRAL W OUN D GASKET 150# . AISI 316L/GR	2"		M	1	
	00002101	SPIRAL W OUN D GASKET 150# . AISI 316L/GR	3"		M	1	
	00002186	FLANGE I SOL ATION KIT 300# RF . GLASS RE	2"		M	1	
	00002582	STUD BOL T 2 NUTS . ASTM A193Gr. B7M LO	5/8		M	12	

Imagen 16

- **Material No Marcado:** muestra la lista del material, de la línea seleccionada, que habiendo sido entregado al contratista en algún vale de salida, aún no ha sido marcado como prefabricado o montado según se indique.
- **Material Entregado:** muestra la lista del material, de la línea seleccionada, que ha sido entregado en algún vale de salida. En esta lista no se tiene en cuenta si ha sido marcado o no.
- **Material Prefabricado:** muestra la lista del material, de la línea seleccionada, que ya ha sido marcado y está clasificado como material de prefabricación (P).
- **Material Montado:** muestra la lista del material, de la línea seleccionada, que ya ha sido marcado y está clasificado como material de montaje (M).

- **Zona 2, Interfaz con el Visualizador 3D**, en esta zona se encuentran los comandos que el usuario tiene a su disposición para visualizar la isométrica sobre la maqueta 3D y navegar sobre los elementos de la maqueta en la ventana del Visualizador.

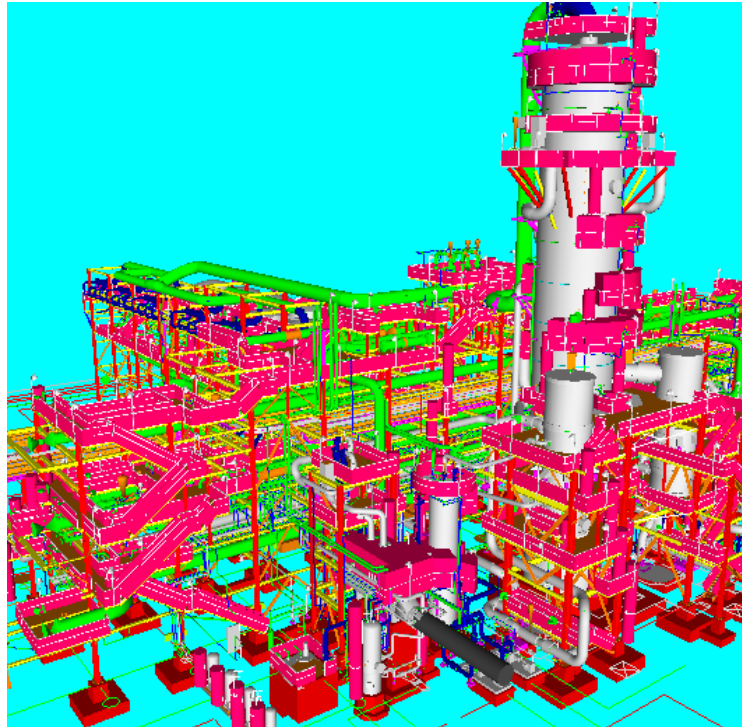















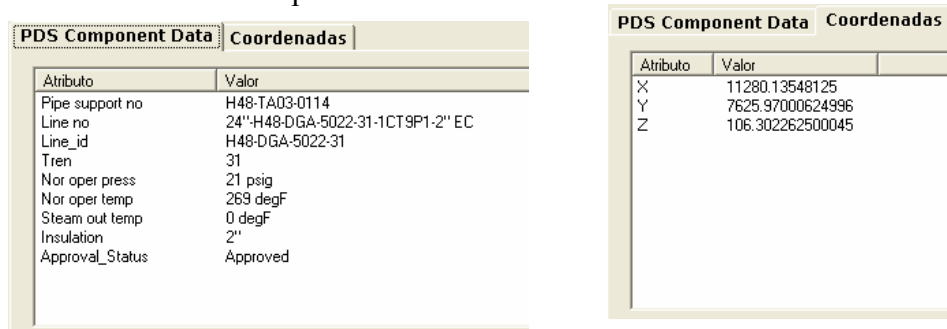


Imagen 17

Consta de las siguientes utilidades:

- **Barra de herramientas:** permite al usuario realizar operaciones básicas de navegación y visualización de elementos sobre la maqueta. Se trata de funcionalidades que están disponibles en Navisworks.
 1.  **Guardar:** guarda el estado actual de la maqueta, en cuanto a colores, punto de vista, etc., en el fichero .nwd asociado.
 2.  **Select:** activa el modo de selección de elementos sobre la maqueta 3D.
 3.  **Walk:** activa el modo de navegación correspondiente en Navisworks.
 4.  **Zoom:** activa el modo de navegación correspondiente en Navisworks.
 5.  **Zoom Box:** activa el modo de navegación correspondiente en Navisworks.
 6.  **Pan:** activa el modo de navegación correspondiente en Navisworks.

7.  **Orbit:** activa el modo de navegación correspondiente en Navisworks.
8.  **Examine:** activa el modo de navegación correspondiente en Navisworks.
9.  **Fly:** activa el modo de navegación correspondiente en Navisworks.
10.  **Turntable:** activa el modo de navegación correspondiente en Navisworks.
11.  **Unselected Hidden:** oculta todos los objetos de la maqueta, excepto aquellos que están seleccionados.
12.  **View Selected:** realiza un zoom sobre los elementos de la maqueta que están seleccionados.
13.  **View All:** muestra todos los elementos de maqueta, incluso aquellos que se habían ocultado previamente.
14.  **BackColor:** permite al usuario cambiar el color de fondo de la ventana de visualización de la maqueta 3D.
15.  **Ver propiedades:** mediante esta opción es posible visualizar los *atributos de diseño* y las *coordenadas* del elemento seleccionado en la maqueta. Si hay varios elementos seleccionados se mostrará la información del primero de ellos.



PDS Component Data		Coordenadas	
Atributo	Valor	Atributo	Valor
Pipe support no	H48-TA03-0114	X	11280.13548125
Line no	24"-H48-DGA-5022-31-1CT9P1-2" EC	Y	7625.9700624996
Line_id	H48-DGA-5022-31	Z	106.302262500045
Tren	31		
Nor oper press	21 psig		
Nor oper temp	269 degF		
Steam out temp	0 degF		
Insulation	2"		
Approval_Status	Approved		

Imagen 18

- **Otros comandos:** en este apartado se va a explicar el funcionamiento de otras opciones de visualización sobre la maqueta, que no están disponibles desde Navisworks. Se trata de comandos que han sido especialmente implementados para esta aplicación:

1. **Visualizar línea:** selecciona y realiza un zoom, en la ventana del visualizador, sobre todos los elementos correspondientes a la línea activa en la tabla, tal y como muestra la imagen 19.

Línea activa seleccionada en azul.

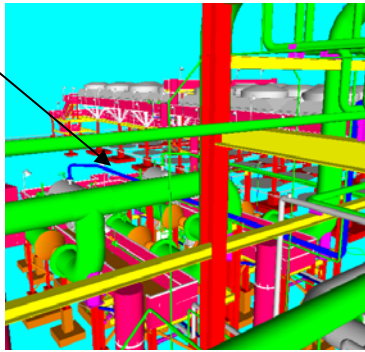
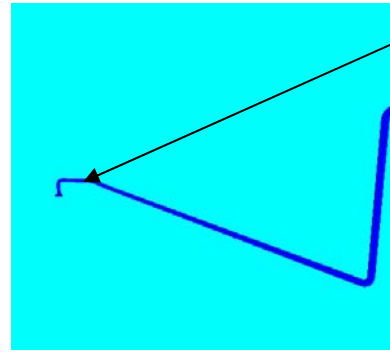



Imagen 19



Línea activa en azul, resto de elementos ocultos.

Imagen 20

Si después de utilizar este comando se utiliza la opción de la barra de herramientas  el resultado sería el que muestra la imagen 20.

2. **Material Prefabricado:** selecciona en la ventana de la maqueta 3D todos aquellos elementos de material que han sido marcados por el usuario y están clasificados como de Prefabricación.
3. **Material Montado:** selecciona en la ventana de la maqueta 3D todos aquellos elementos de material que han sido marcados por el usuario y están clasificados como de Montaje.
4. **Material Prefabricado/Montado:** selecciona en la ventana de la maqueta 3D todos aquellos elementos de material que han sido marcados por el usuario, ya sean de Prefabricación o de Montaje.
5. **Material No Marcado:** selecciona en la ventana de la maqueta 3D todos aquellos elementos de material que aún no han sido marcados por el usuario.
6. **Aplicar Color:** permite definir el color que se aplicará en la maqueta a los materiales de Prefabricados y a los materiales de Montaje. Al elegir esta opción se mostrará el siguiente formulario:

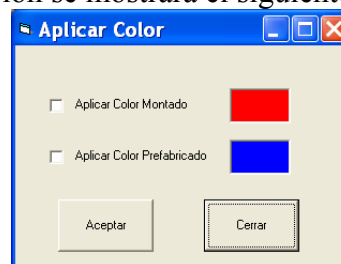


Imagen 21

El *Botón Aceptar* permite aplicar el color a los elementos prefabricados o montados según esté marcado en el formulario. El *Botón Cerrar* permite salir del formulario.

- **Zona 4, Opciones de marcado**, este grupo de opciones permite marcar como contruidos (ya sean prefabricados o montados) los materiales de una línea. Los comandos disponibles se han clasificado en dos grupos dependiendo del lugar sobre el que se realice el marcado: la tabla de material no marcado o la maqueta 3D.
 - **Opciones para elementos representados en el 3D:** se compone de los siguientes comandos:
 1. **Marcar elementos:** marca el elemento o elementos seleccionados en la maqueta 3D como prefabricado o montado según corresponda. Automáticamente, se actualizarán las listas de material no marcado y material prefabricado o montado según la clasificación del elemento o elementos marcados.
 2. **Desmarcar elementos:** elimina el elemento o elementos seleccionados en la maqueta 3D de la lista de elementos marcados. Automáticamente, se actualizarán las listas de material no marcado y material prefabricado o montado según la clasificación del elemento o elementos marcados.
 3. **Ver elemento:** permite seleccionar y visualizar sobre la maqueta 3D el elemento seleccionado en la lista de material activa.
 - **Opciones para elementos NO representados en el 3D:** las opciones de este apartado se aplicarán a aquellos materiales que no se representan en la maqueta 3D, como puede ser el caso de juntas y pernos. Los comandos son los siguientes:
 1. **Marcar elementos:** permite marcar como contruidos, todos aquellos elementos que están seleccionados en la lista de material no marcado. Si no está activa la pestaña correspondiente a la lista de material no marcado aparecerá el siguiente mensaje:

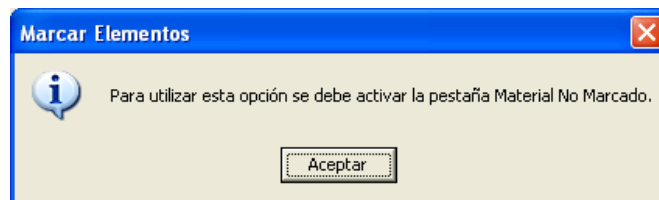


Imagen 22

2. **Desmarcar elementos:** permite eliminar de la lista de contruidos, todos aquellos elementos que están seleccionados en la lista de material prefabricado o montado. Si no está activa alguna de estas pestañas, aparecerá un mensaje advirtiendo esta situación.

6 Menú Herramientas

En el menú herramientas podemos encontrar las opciones que se pueden ver en la imagen inferior.



Imagen 23

Las opciones agrupadas bajo esta herramienta hacen referencia a los procesos de importación de ficheros generados por el sistema MATUB, la gestión de errores de carga y de borrado de isométricas y la configuración de los colores aplicados a las distintas categorías de materiales.

Todos los procesos de Importación son muy lentos y pueden consumir un alto grado de recursos.

6.1 Importar Isométricas

Esta opción se sitúa en el menú herramientas y dentro de este en el submenú Importar.

Permite realizar la importación del informe denominado Listado de Elementos que se genera desde el **menú de Obra** (opción 18) de MATUB siguiendo los siguientes pasos:

Opción 5 Tuberías

Opción 2 Menú Análisis

Opción 8 Listado Elementos

Importante: generar el informe hacia el correo electrónico utilizando la opción de impresión 701.

Al pinchar en ella se abre la venta de importación de Isométricas, desde la cual se selecciona el fichero a cargar generado por el sistema MATUB conforme al formato indicado en la imagen siguiente.

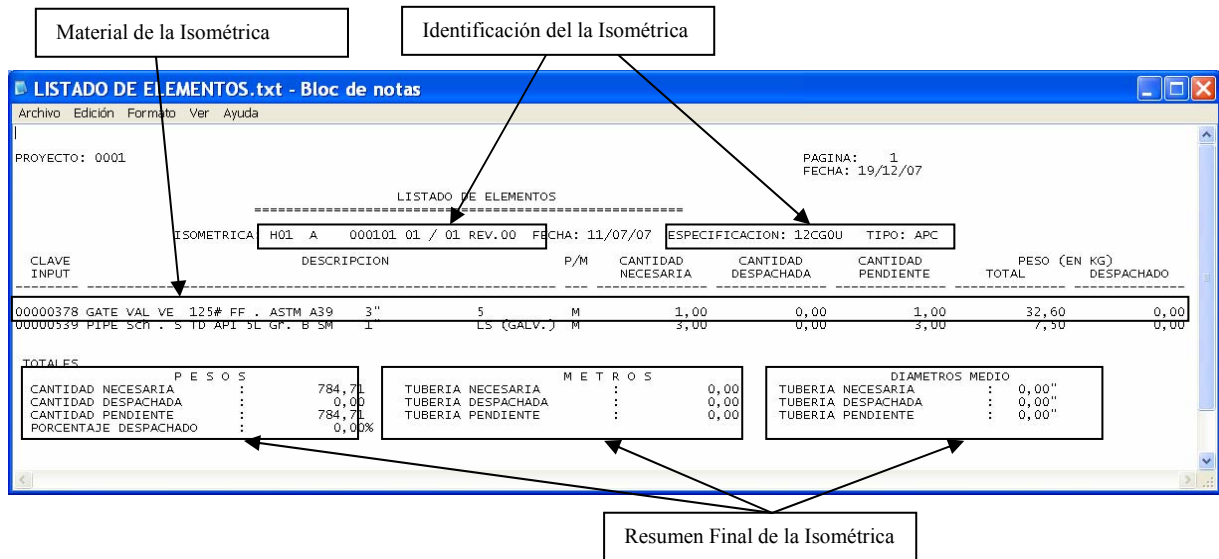


Imagen 24

En este fichero se especifica la Isométrica, el material recontado de la misma y un resumen final. También aparecen identificados la fecha y el proyecto al que pertenece la isométrica.

Al finalizar el proceso, si ha completado el proceso de forma correcta se notificará y se habrá cargado en la base de datos la información referenciada en el fichero, pudiendo ser una isométrica entera, parte de una isométrica o un conjunto de isométricas.

En caso de no aparecer ningún mensaje tras la finalización, se deberá revisar el “log” de error generado en el directorio log de la aplicación para comprobar el motivo del error.

Si se ha cargado nueva versión de una isométrica de la que había generado un vale, se genera una línea de error tratable desde la herramienta de gestión de carga del menú herramientas.

6.2 Importar Vale de Salida

Esta opción se sitúa en el menú herramientas y dentro de este en el submenú Importar.

Al pinchar en ella se abre la venta de importación de Vales de Salida, desde la cual se selecciona el fichero a cargar generado por el sistema MATUB conforme al formato indicado en la imagen siguiente

Permite realizar la importación de vales de salida generados desde el **menú de Obra** (opción 18) de MATUB según una de las dos opciones siguientes:

- 1) Manual, contra una isométrica:

Opción 5 Tuberías

Opción 4 Construcción parcial de una isométrica

2) Automático, contra un análisis:

Opción 5 Tuberías

Opción 2 Menú Análisis

Opción 7 Orden de fabricación

Importante: generar el informe hacia el correo electrónico utilizando la opción de impresión 701.

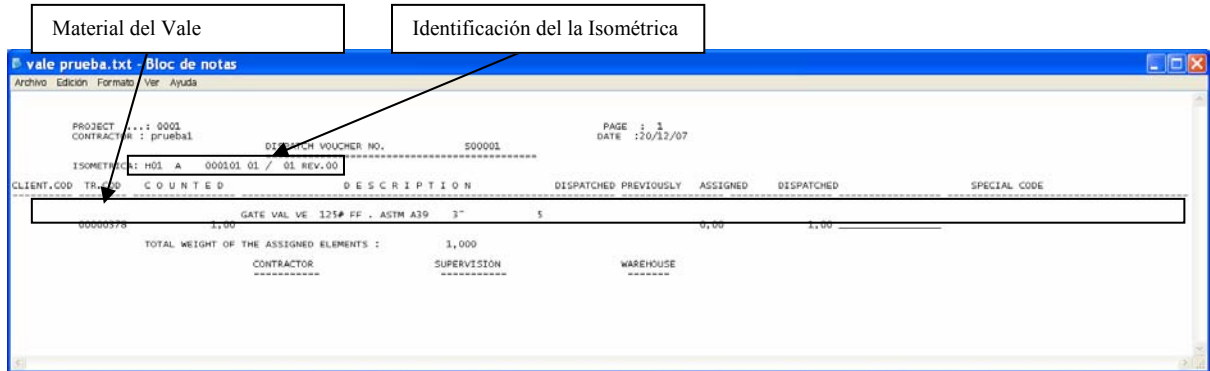


Imagen 25

En este fichero se hace referencia a la isométrica a la que pertenece el material listado en el vale y el número de vale, esta es la información más relevante del fichero.

Al igual que en el caso de la carga de las isométricas, una vez cargadas todas las líneas del vale se muestra un mensaje en pantalla indicando que la operación ha terminado correctamente.

En caso de no aparecer ningún mensaje tras la finalización, se deberá revisar el "log" de error generado en el directorio log de la aplicación para comprobar el motivo del error.

6.3 Importar Vales de Forma Masiva

En casos de carga inicial o de recarga del proyecto, la carga se realizará por esta vía. Al seleccionar esta acción en el menú Herramientas → Importar, se abrirá una ventana en la que deberemos seleccionar el vale masivo generado por MATUB, la diferencia entre esta forma de importar los vales y la carga normal, es que el fichero a cargar debe tener el mismo formato que un fichero de importación de Isométricas.

El proceso de importación masiva, el sistema pedirá un nombre para asociar todo el material cargado en este proceso.

6.4 Gestión de cargas de Isométricas

Desde esta opción del menú se despliega la ventana de Gestión de Cargas, pantalla en la que se visualiza una lista con los errores que se han producido en los procesos de carga de isométricas y de vales.

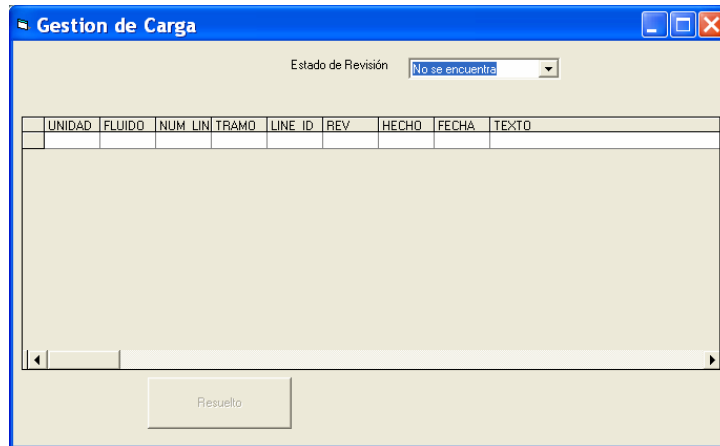


Imagen 26

Estos errores pueden ser debidos a las siguientes razones:

- Al cargar una nueva revisión de una isométrica que tenga un vale en el sistema.
- Al borrar una isométrica con vales en el sistema.

Una vez tratada la alerta de la forma que se considere, se puede cambiar el estado de las alertas seleccionándola y pulsando el botón Resuelto.

Al realizar esta acción, se añadirá la fecha en la que se resolvió la alerta.

6.5 Configurar Colores

Desde la opción Configurar Colores dentro del menú Herramientas y del submenú Configurar, podemos abrir la ventana en la que se configura la representación de las categorías en la maqueta, estas categorías pueden ser Montado, Entregado y Prefabricado; para cada una de ellas se puede definir un color pulsando el botón de búsqueda.

Una vez configurado podemos pulsar el botón Guardar para salvar el resultado en base de datos o bien pulsar Cerrar para no guardarlo.

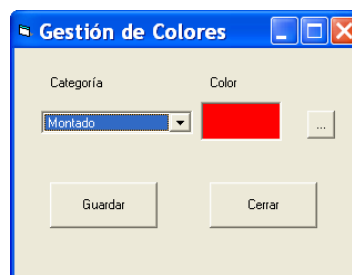


Imagen 27

7 Menú Dir. Construcción

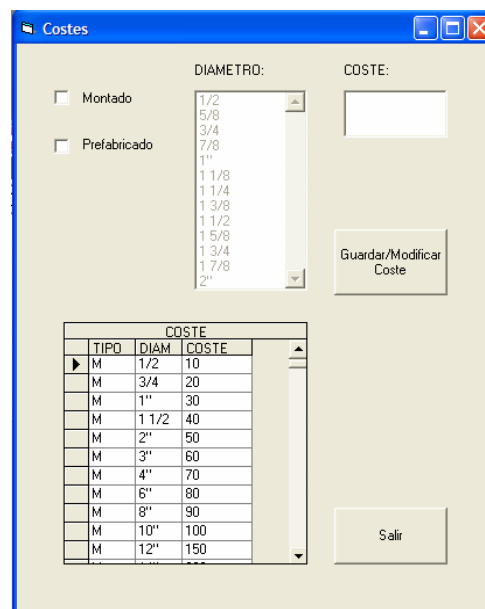
7.1 Costes

En el formulario principal, esta opción permite introducir los costes de prefabricación y montaje correspondientes al proyecto que permitirán realizar el cálculo del avance del proyecto en coste.



Imagen 28

A continuación, se abre el formulario de Costes:

A screenshot of the 'Costes' form. It features two checkboxes: 'Montado' and 'Prefabricado'. To the right, there is a 'DIAMETRO' dropdown menu with options ranging from 1/2 to 2 inches, and a 'COSTE' input field. Below these is a 'Guardar/Modificar Coste' button. At the bottom, there is a table with columns 'TIPO', 'DIAM', and 'COSTE'. The table contains data for diameters from 1/2 to 12 inches, with corresponding costs from 10 to 150. A 'Salir' button is located at the bottom right.

COSTE		
TIPO	DIAM	COSTE
M	1/2	10
M	3/4	20
M	1"	30
M	1 1/2	40
M	2"	50
M	3"	60
M	4"	70
M	6"	80
M	8"	90
M	10"	100
M	12"	150

Imagen 29

- **Descripción:** Formulario que se utiliza para introducir o mostrar el coste de prefabricación y montaje de materiales, según su diámetro nominal.
 - **Tabla Coste:** Muestra el contenido de la tabla COSTES
 - **Botón Guardar/ Modificar Coste:** Guarda o modifica el coste que se refleja en el formulario.
 - **Botón Salir:** opción para salir del formulario.

7.2 Informes

En el formulario principal, la opción para generar los informes de avance que ofrece el sistema es la siguiente:



Imagen 30

A continuación, se abre el formulario Elegir Unidad:

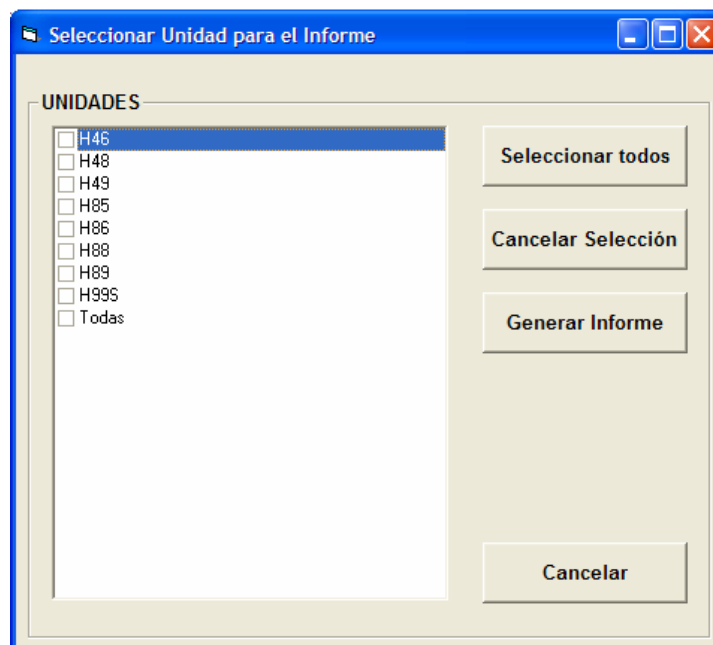


Imagen 31

- **Descripción:** Formulario que se utiliza para seleccionar la unidad o unidades de la que se desea generar informe.
 - **UNIDADES:** Muestra todas las unidades del proyecto y permite seleccionar aquellas que se quieren incluir en el informe.
 - **Seleccionar todos:** Selecciona todas las unidades.
 - **Cancelar Selección:** Elimina la selección de todas las unidades que estén seleccionadas.
 - **Generar Informe:** Genera el informe de las unidades seleccionadas.
 - **Cancelar:** opción para salir del formulario.

En el Anexo A se puede consultar un ejemplo del formato de informes que se generan desde esta opción.

Autorización

Los abajo firmantes: María del Mar Gutiérrez Fernández, César Marqués Collado y Raquel Rodríguez Casero, autorizan a la Universidad Complutense de Madrid a difundir y utilizar con fines académicos, no comerciales, y, mencionando expresamente a los autores, tanto la presente memoria, como el código, la documentación, y/o el prototipo desarrollado.

M^a del Mar Gutiérrez Fernández

César Marqués Collado

Raquel Rodríguez Casero