



Eusko Jaurlaritzaren Informatika Elkartea  
Sociedad Informática del Gobierno Vasco

## Enterprise Architect:

Manual de usuario

**Fecha:**

**Referencia:**

EJIE S.A.  
Mediterráneo, 3  
Tel. 945 01 73 00\*  
Fax. 945 01 73 01  
01010 Vitoria-Gasteiz  
Posta-kutxatila / Apartado: 809  
01080 Vitoria-Gasteiz  
[www.ejie.es](http://www.ejie.es)

# Control de documentación

Título de documento: ENTERPRISE ARCHITECT

## Histórico de versiones

Código:

Versión: 1.2

Fecha:

Resumen de cambios: Punto 5.2 . Ejemplo de documentación de OTC.

## Cambios producidos desde la última versión

Primera versión.

## Control de difusión

Responsable: Ander Martínez

Aprobado por:

Firma:

Fecha:

Distribución:

## Referencias de archivo

Autor: Consultoría de áreas de conocimiento

Nombre archivo: EA. Manual de usuario para ArinBide v1.2.docx

Localización:

## Contenido

	Capítulo/sección	Página
1	Introducción	5
2	Conceptos básicos	5
2.1	UML	5
3	Funciones elementales	8
3.1	Crear el proyecto	8
3.2	Conexión al repositorio/proyecto	11
3.3	Creación de Modelos	13
3.3.1.	Ejemplo: Requerimientos del Sistema	14
3.4	Exportación/Importación XMI	23
3.5	Documentación	26
3.5.1.	Generación Documentos Word	26
3.5.2.	Diseño Plantillas	32
3.6	Modelo de seguridad	36
3.7	Trabajo en grupo	39
3.7.1.	Mantenimiento de usuarios	39
3.7.2.	Usuario administrador	42
3.7.3.	Mantenimiento de grupos	42
3.7.4.	Gestión de bloques	45
3.7.5.	Otras funciones	46
3.8	Modelado Datos con EA	47
3.8.1.	Creación Modelo de Datos	47
3.8.2.	Importar desde BD	60
4	Anexo 1 : Diagrama de Caso de Uso	64
4.1	Resolución	64
5	Enterprise Architect VS Arin Bide	82
5.1	Arquitectura Proyecto	82

5.2 Documentación	82
5.2.1. Documentación en EA	83
5.2.2. Documentación Arin Bide	85
5.3 Análisis Sistema Información (ASI)	87
5.3.1. Modelado UML	87
5.3.2. Generación Documentación	105
5.4 Definición de la Arquitectura del Sistema – DS1 1	109
5.4.1. Modelado UML	109
5.4.2. Generación Documentación	117
5.5 Diseño de Casos de Uso Reales – DS1 3	122
5.5.1. Modelado UML	122
5.5.2. Generación Documentación	122
5.6 Diseño de Clases – DS1 4	125
5.6.1. Modelado UML	125
5.6.2. Generación Documentación	132
5.7 Diseño Físico de Datos – DS1 6	134
5.7.1. Modelado UML	134
5.7.2. Generación Documentación	143
5.8 Diseño Físico de Datos – DS1 8	145
5.8.1. Modelado UML	145
5.8.2. Generación Documentación	151
5.9 Diseño de la Migración y carga inicial de datos – DS1 9	152
5.9.1. Modelado UML	152
5.9.2. Generación Documentación	161
5.10 Diseño de la Migración y carga inicial de datos – DS1 9	163
5.10.1. Modelado UML	163
5.10.2. Generación Documentación	176

## 1 Introducción

El presente documento describe cuales son las tareas básicas que se pueden ejecutar en la explotación de la herramienta de modelado Enterprise Architect Corporate Edition.

El contenido del documento integra, tanto los aspectos de uso en el entorno de EJIE como las características elementales de funcionamiento de la aplicación.

## 2 Conceptos básicos

Enterprise Architect Corporate Edition es una herramienta CASE para el diseño y construcción de sistemas software.

Enterprise Architect Corporate Edition soporta la especificación UML 2.0, que describe un lenguaje visual que permite la definición de los modelos de un proyecto.

Se trata de una herramienta progresiva que cubre todos los aspectos del ciclo de un desarrollo, proporcionando una completa trazabilidad desde la fase inicial de diseño hasta el desarrollo y posterior mantenimiento. Así mismo, también proporciona soporte para testing y control de cambios.

Enterprise Architect Corporate Edition, permite la realización de ingeniería directa e inversa, sincronizar los elementos de los modelos con el código fuente de las clases (ActionScript, C++, C#, Delphi, Java, Python, PHP, VB.NET y Visual Basic), diseñar y generar elementos de base de datos y generar documentación de gran calidad fácilmente exportable a formato RTF.

Enterprise Architect Corporate Edition soporta todos los modelos/diagramas de UML 2.0. Permite diseñar desde procesos de negocio, sitios web, interfaces de usuario, configuraciones hardware, hasta estimar el esfuerzo del proyecto en horas

El repositorio esta basado en DBMS proporciona buenos tiempos de respuesta cuando se trabaja con varios usuarios debido a su estructura interna. Además, cualquier problema de conexión que se produzca, debería ser cubierto por las habilidades del servidor DBMS, permitiendo deshacer cualquier transacción interrumpida por problemas externos.

En nuestro caso se ha seleccionado SQL Server 7.0 como repositorio de proyectos, y la licencia Corporate Edition.

Para obtener información adicional sobre el producto acceder a su página web:

<http://www.sparxsystems.com.au/products/ea.html>

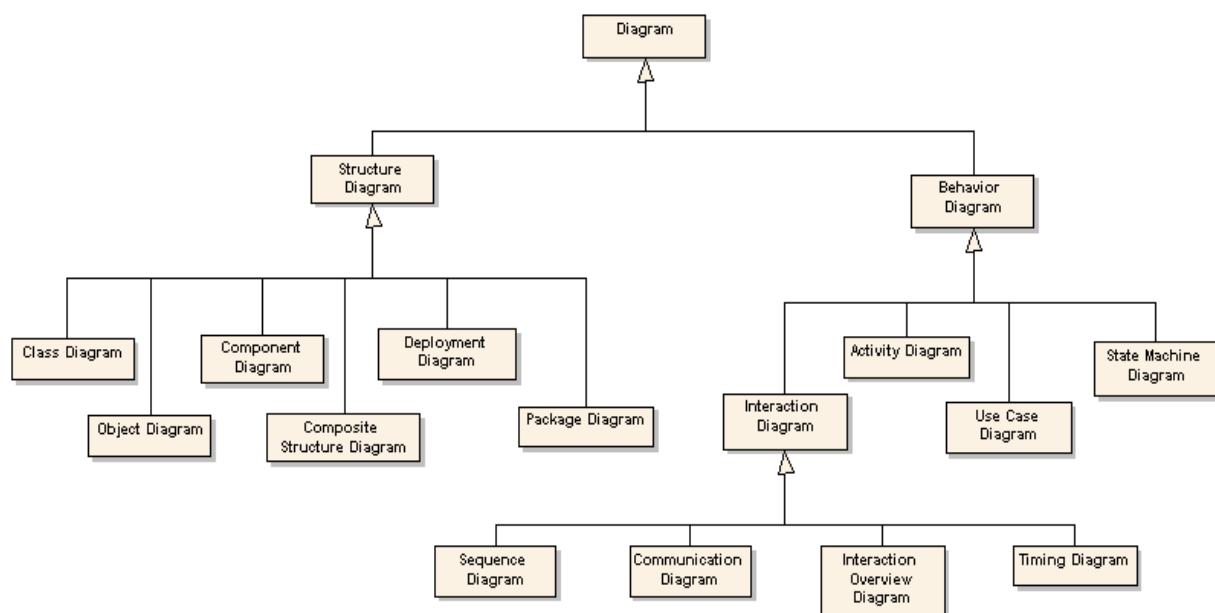
### 2.1 UML

Lenguaje Unificado de Modelado (UML, por sus siglas en inglés, Unified Modeling Language) es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad; aún cuando todavía

no es un estándar oficial, está respaldado por el OMG (Object Management Group). Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema de software. UML ofrece un estándar para describir un "plano" del sistema (modelo), incluyendo aspectos conceptuales tales como procesos de negocios y funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes de software reutilizables.

Es importante resaltar que UML es un "lenguaje" para especificar y no para describir métodos o procesos. Se utiliza para definir un sistema de software, para detallar los artefactos en el sistema y para documentar y construir. En otras palabras, es el lenguaje en el que está descrito el modelo. Se puede aplicar en una gran variedad de formas para dar soporte a una metodología de desarrollo de software -pero no especifica en sí mismo qué metodología o proceso usar.

UML cuenta con varios tipos de diagramas, los cuales muestran diferentes aspectos de las entidades representadas.



Tal y como se puede ver en la imagen existen dos grandes grupos:

### Diagramas de Estructura

Se trata de los diagramas que enfatizan los elementos que deben existir en el sistema modelado.

- **Diagrama de clases:** Describe la estructura de un sistema mostrando sus clases, atributos y las relaciones entre ellos. Los diagramas de clases son utilizados durante el proceso de análisis y diseño de los sistemas informáticos, donde se crea el diseño conceptual de la información que se manejará en el sistema, y los componentes que se encargaran del funcionamiento y la relación entre uno y otro.
- **Diagrama de componentes:** Un diagrama de componentes representa la separación de un sistema de software en componentes físicos (por ejemplo archivos, cabeceras, módulos, paquetes, etc.) y muestra las dependencias entre estos componentes.
- **Diagrama de objetos:** Se puede considerar un caso especial de un diagrama de clases en el que se muestran instancias específicas de clases (objetos) en un momento particular del sistema.

Los diagramas de objetos utilizan un subconjunto de los elementos de un diagrama de clase. Los diagramas de objetos no muestran la multiplicidad ni los roles, aunque su notación es similar a los diagramas de clase.

- **Diagrama de estructura compuesta (UML 2.0):** Se trata de un diagrama que permite contextualizar las partes que componen a una clase, muestra la estructura interna de una clase, incluyendo sus puntos de interacción a otras partes del sistema.
- **Diagrama de despliegue:** Se utiliza para modelar el hardware utilizado en las implementaciones de sistemas y las relaciones entre sus componentes.
- **Diagrama de paquetes:** Muestra como un sistema está dividido en agrupaciones lógicas mostrando las dependencias entre esas agrupaciones. Dado que normalmente un paquete está pensado como un directorio, los diagramas de paquetes suministran una descomposición de la jerarquía lógica de un sistema.

## Diagramas de comportamiento

Se trata de los diagramas que enfatizan lo que debe suceder en el sistema modelado

- **Diagrama de Actividades:** representa los flujos de trabajo paso a paso de negocio y operacionales de los componentes en un sistema. Un Diagrama de Actividades muestra el flujo de control general.
- **Diagrama de Casos de Uso:** Un diagrama de casos de uso es una representación gráfica de parte o el total de los actores y casos de uso del sistema, incluyendo sus interacciones.
- **Diagrama de Estados:** Se trata de diagramas que se usan para representar gráficamente máquinas de estados finitos.
- **Diagramas de Interacción:** Se trata de diagramas que enfatizan el flujo de control y de datos entre los elementos del sistema modelado
  - **Diagrama de Secuencia:** Muestra la interacción de un conjunto de objetos en una aplicación a través del tiempo y se modela para cada caso de uso.
  - **Diagrama de Colaboración:** Un diagrama de colaboración es una forma alternativa al diagrama de secuencia de mostrar un escenario. Este tipo de diagrama muestra las interacciones entre objetos organizadas entorno a los objetos y los enlaces entre ellos.
  - **Diagrama de Tiempos:** Es una gráfica de formas de onda digitales que muestra la relación temporal entre varias señales, y cómo varía cada señal en relación a las demás.
  - **Diagrama de Vistas de Interacción:** Se trata de un diagrama que describe la cooperación entre distintos diagramas de interacción, proporcionando un flujo de control determinado.

Así mismo, EA proporciona los siguientes diagramas extendidos:

- **Diagramas de Análisis:** Se trata de un diagrama de actividades simplificado, que se utiliza para la definición de procesos de negocio de alto nivel.
- **Diagramas a Medida:** Se trata de un diagrama de clases extendido utilizado para capturar requerimientos, interfaces de usuario o modelos de diseño.
- **Diagramas de Requerimientos:** Diagrama que permite describir los requerimientos del sistema y sus características.
- **Diagramas de Mantenimiento:** Se trata de un diagrama utilizado para describir peticiones de cambio dentro de un sistema.
- **Diagramas de interfaz de usuario:** Se trata de diagramas que permite el diseño de las pantallas utilizando formularios, controles y etiquetas.
- **Diagramas de base de datos:** Se trata de un diagrama que permite definir las tablas y las relaciones entre las mismas de una base de datos.

### 3 Funciones elementales

Los pasos mínimos que debe ejecutar el responsable del proyecto se pueden resumir como:

- Solicitar el repositorio del proyecto en BD
- Crear el proyecto
- Activar seguridad
- Establecer usuario(s) administrador(es)
- Crear grupos de usuarios y asignarles privilegios
- Crear usuarios y asignarlos a grupos

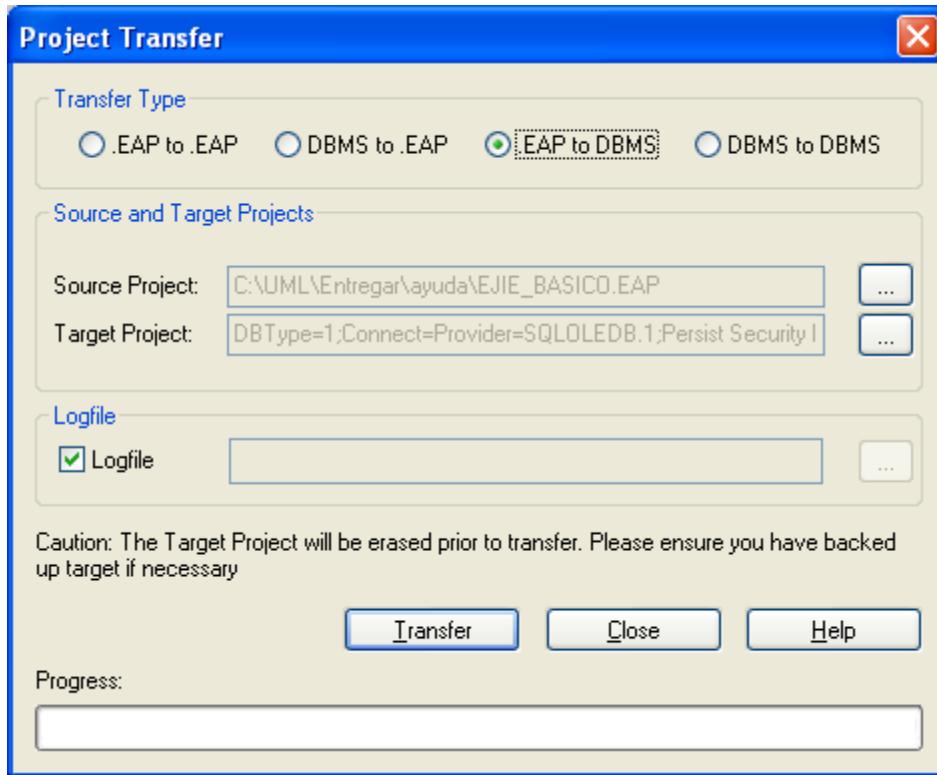
#### 3.1 Crear el proyecto

Previamente (en fase 0) se habrá solicitado al servicio de implantación la creación del repositorio en SQLServer, facilitándonos los siguientes datos:

- Nombre del servidor
- Nombre de la base de datos
- Usuario
- Password

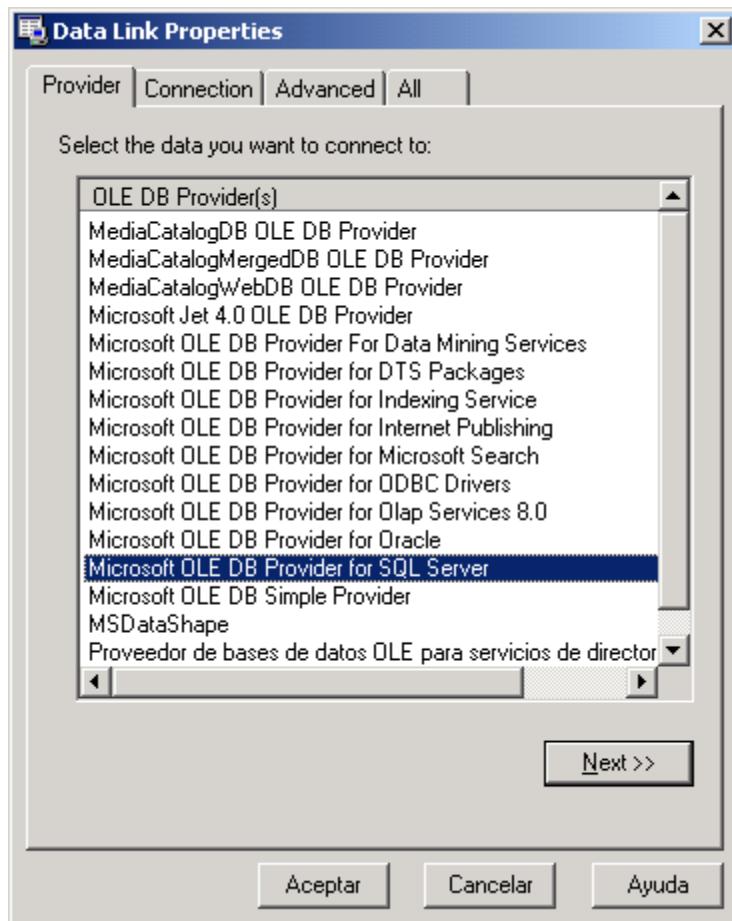
**Importante:** El proceso descrito a continuación lo deberá ejecutar el **responsable del proyecto** una sola vez. Las sucesivas veces que se llevase a cabo supondría perder todos los modelos contenidos en el repositorio remoto, es decir, sería equivalente a un proceso de reinicio del proyecto.

Que implantación confirme la creación del repositorio no significa que ya se pueda empezar a trabajar, tan solo se dispone del espacio necesario para gestionar el proyecto. Por tanto el primer paso a realizar por el responsable es crear el proyecto, ejecutando el menú **Tools→ Data Management→Data Transfer**

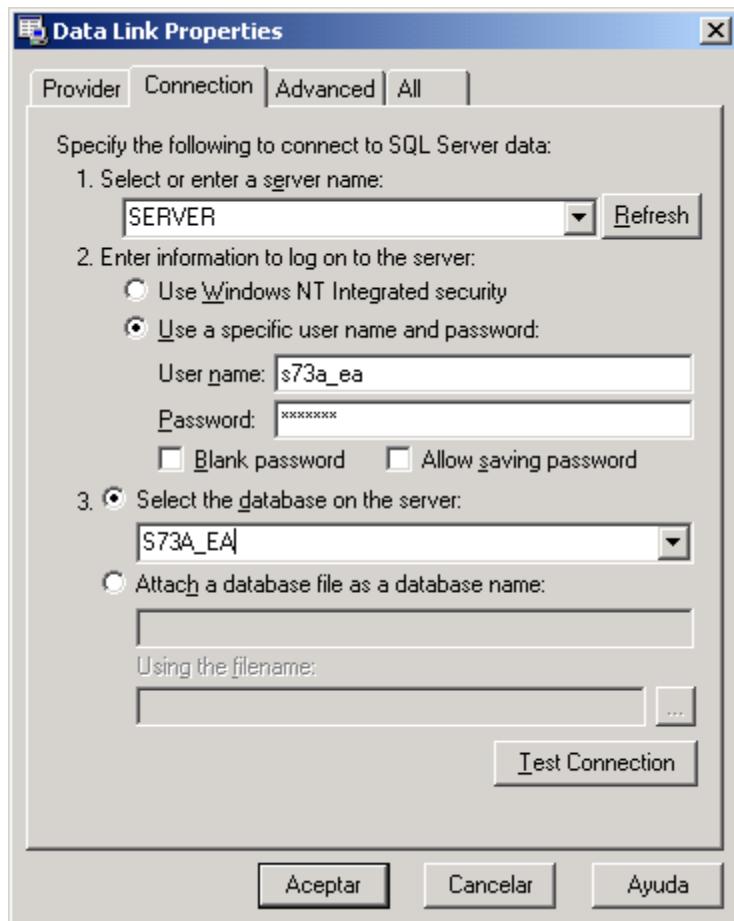


Se completa la pantalla:

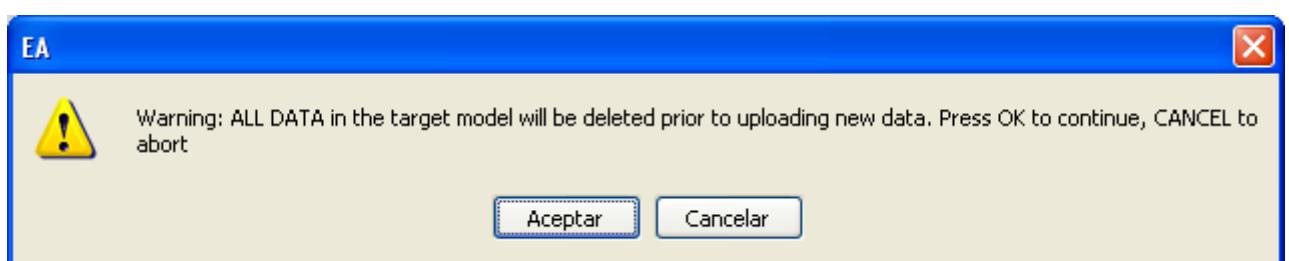
- En el tipo de transferencia a realizar se marca la opción **.EAP to DBMS**
- En el campo de **Source Model** se pincha el botón ... que aparece alineado permitiendo así buscar en nuestro disco un fichero EAP que se va a usar como modelo, **el que se ha de instalar que contiene la estructura de EJIE es el fichero EJIE\_BASICO.EAP**
- Pinchando el botón “...” alineado con **Target Model** se seleccionara como destino la BD SQL Server que nos han generado:



- Una vez se ha seleccionado el driver de conexión a la base de datos (Microsoft OLE DB Provider for SQL Server) se deberá indicar la conexión a la base de datos:



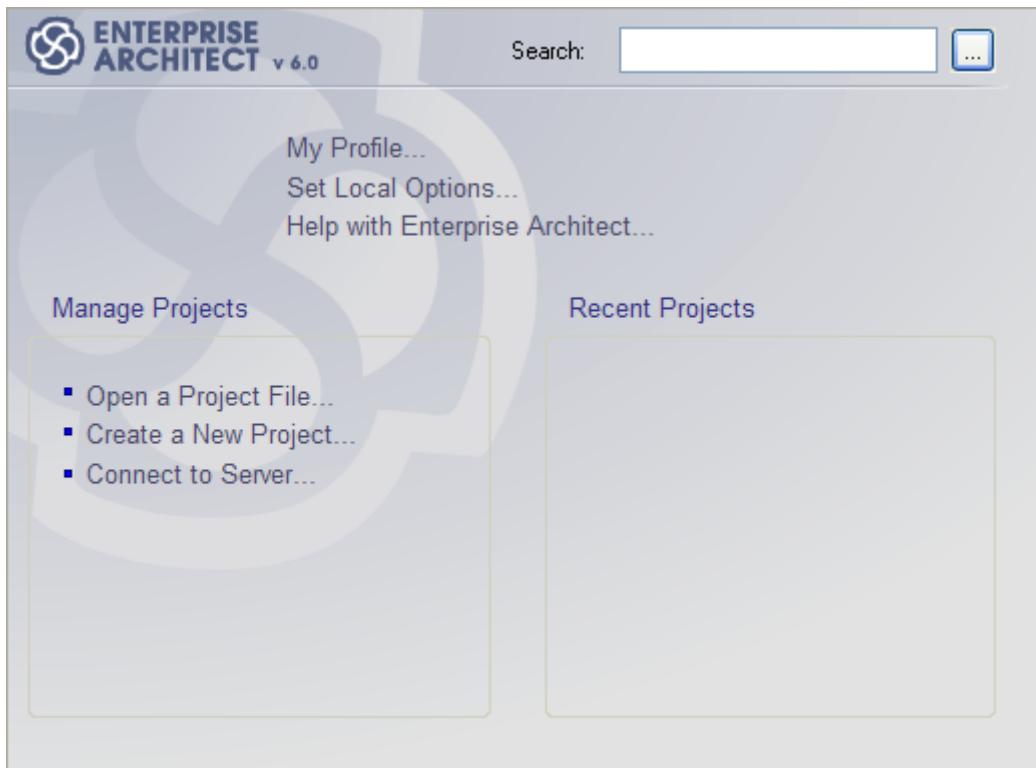
Una vez de vuelta a la pantalla de transferencia de datos se ha pinchar en **Transfer Data** iniciando así el proceso de carga, aunque alertara de que todos los datos que contenga se perderán.



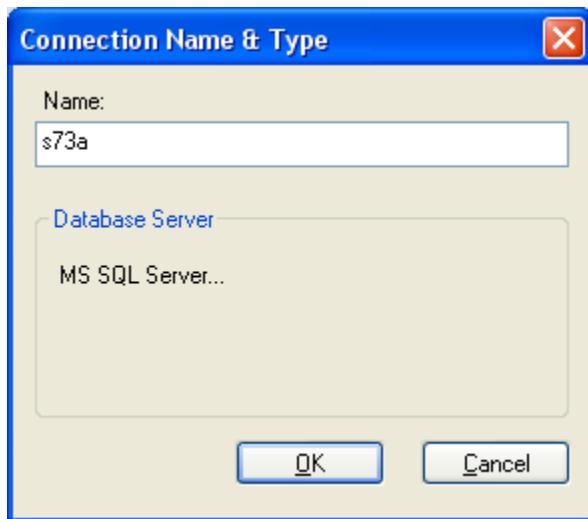
Terminado el proceso se pincha en **Close**.

### 3.2 Conexión al repositorio/proyecto

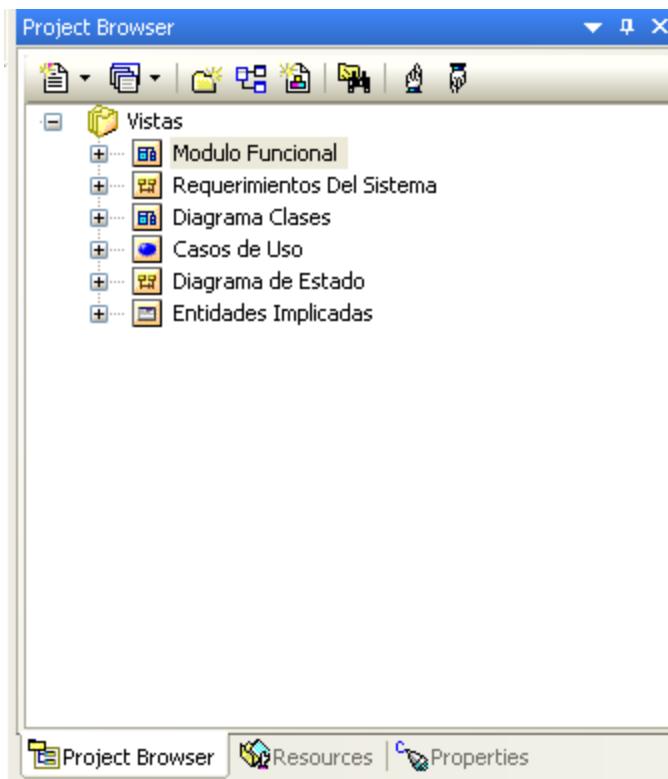
Lo comentado a continuación sirve como guía de conexión al repositorio remoto para cualquier tipo de usuario.



Pinchando sobre **Connect to Server...** se muestra la pantalla de selección de driver de conexión a BD, en la que se ha de seleccionar **Microsoft OLE DB Provider for SQL Server**. Despu  s se indicaran los par  metros de conexi  n a la base de datos. Ahora hay que darle como nombre el c  digo de aplicaci  n:

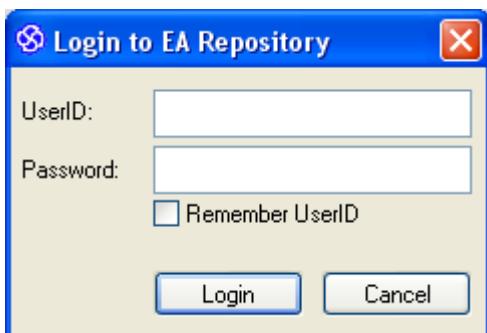


Si la configuraci  n se ha efectuado de modo correcto, se cargará el proyecto EAP residente y existente en la base de datos a la que se ha realizado la conexi  n, pudi  ndose observar en la vista **Project Browser** los diferentes paquetes que componen el proyecto :



En la pantalla principal se ofrece la posibilidad de abrir un proyecto ya existente, crear un nuevo proyecto y/o conectarse a un servidor. Al haber seleccionado como repositorio de documentación la opción de base de datos, el primer paso que deberá realizar cualquier usuario (además del responsable de aplicación) al iniciar la aplicación será conectarse al repositorio a través de la opción **Connect to Server**.

Si se ha activado la seguridad del proyecto y al usuario que intenta acceder se le ha dado de alta sin la posibilidad de utilizar la seguridad Windows (no recomendado), le aparecerá la pantalla de autenticación:



Este punto se explicará más adelante.

### 3.3 Creación de Modelos

Una vez creado el proyecto, el siguiente paso es la modelización de la aplicación a partir de la creación

de los distintos tipos de diagramas que la herramienta proporciona.

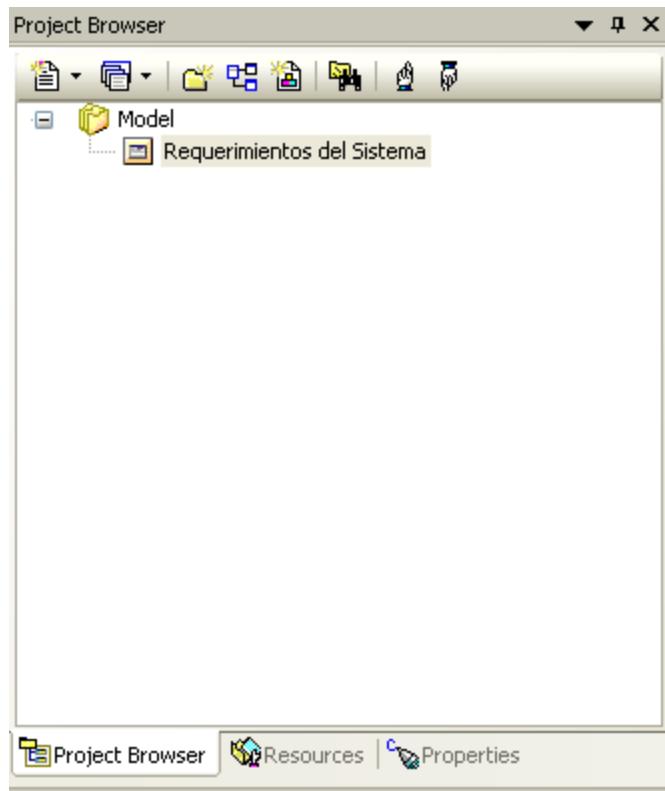
### 3.3.1. **Ejemplo: Requerimientos del Sistema**

Como ejemplo de creación de un diagrama, se va a realizar la creación del diagrama de requerimientos de un sistema, de tal forma que permita recoger tanto los requisitos funcionales como los no funcionales.

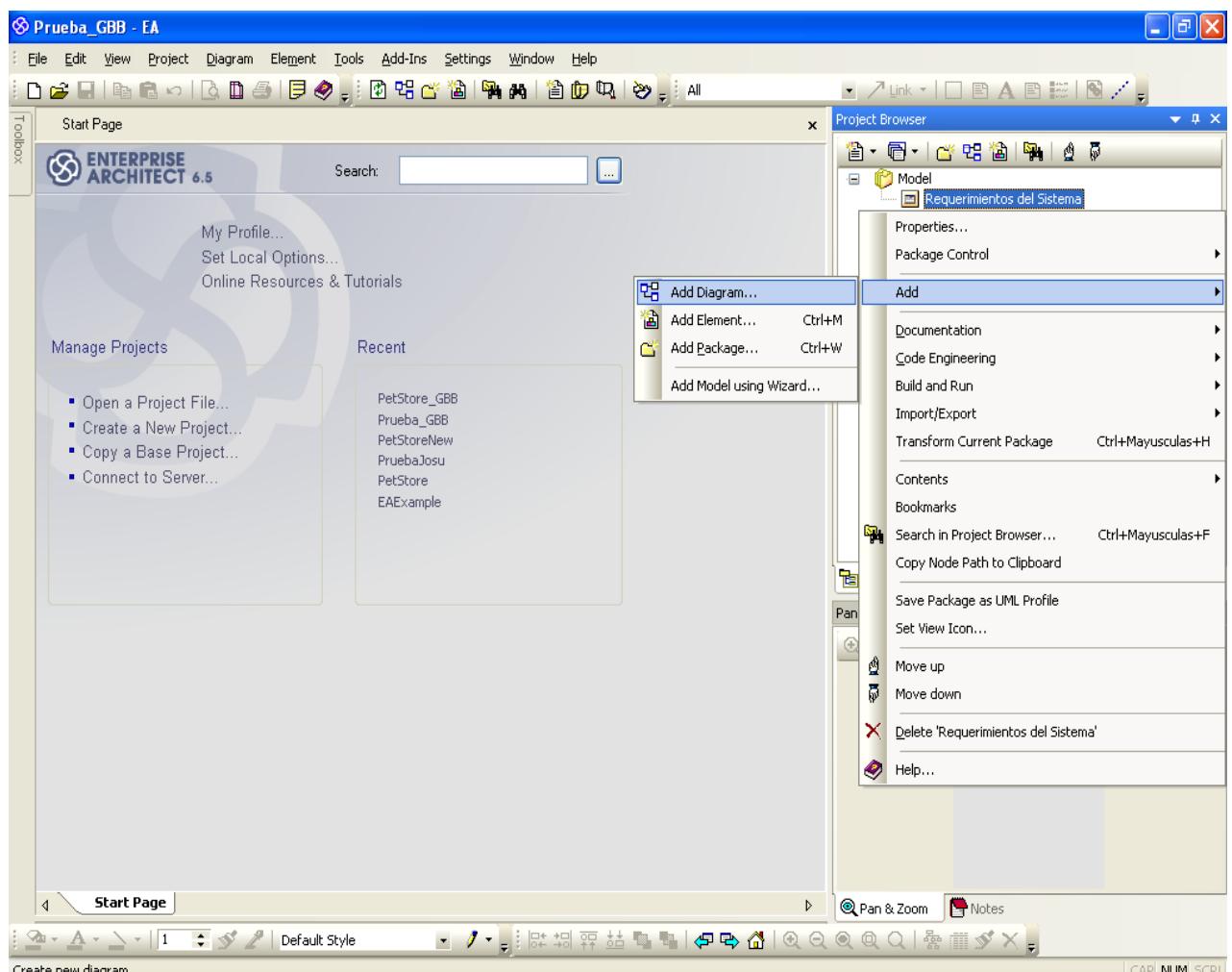
Antes de la creación de los diagramas se ha de crear la estructura del proyecto. Para ello sobre la vista **Project Browser** y sobre el objeto **Model**, pulsar el botón derecho y seleccionar la opción **New View**.



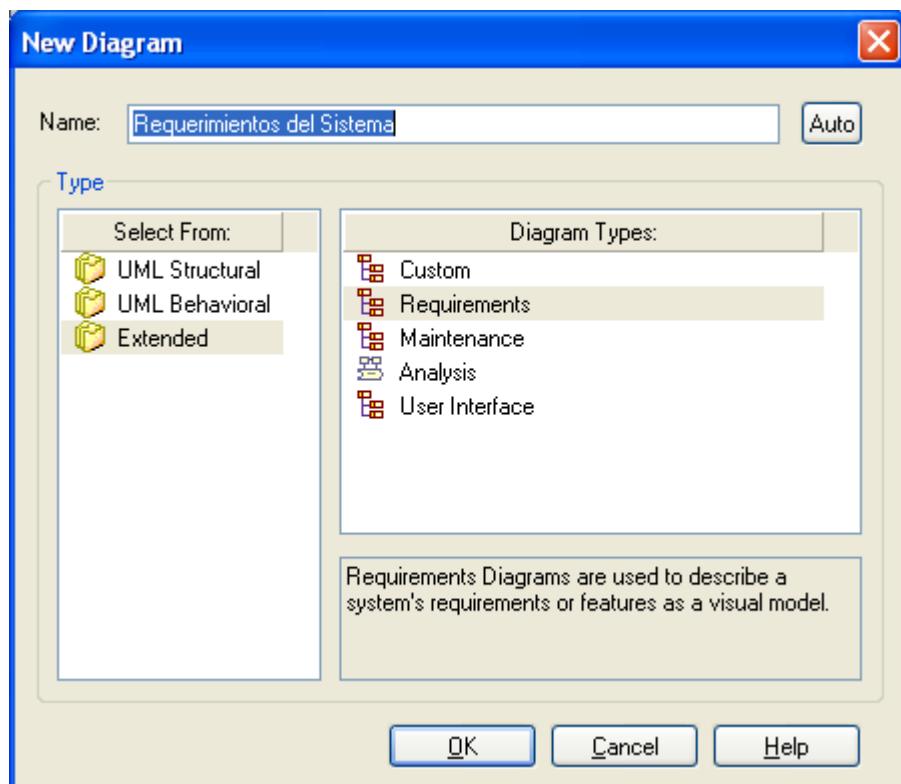
De esta forma se irá generando una estructura de paquetes en la que almacenar los diagramas en base a la funcionalidad y el tipo de diagramas que se vayan generando.



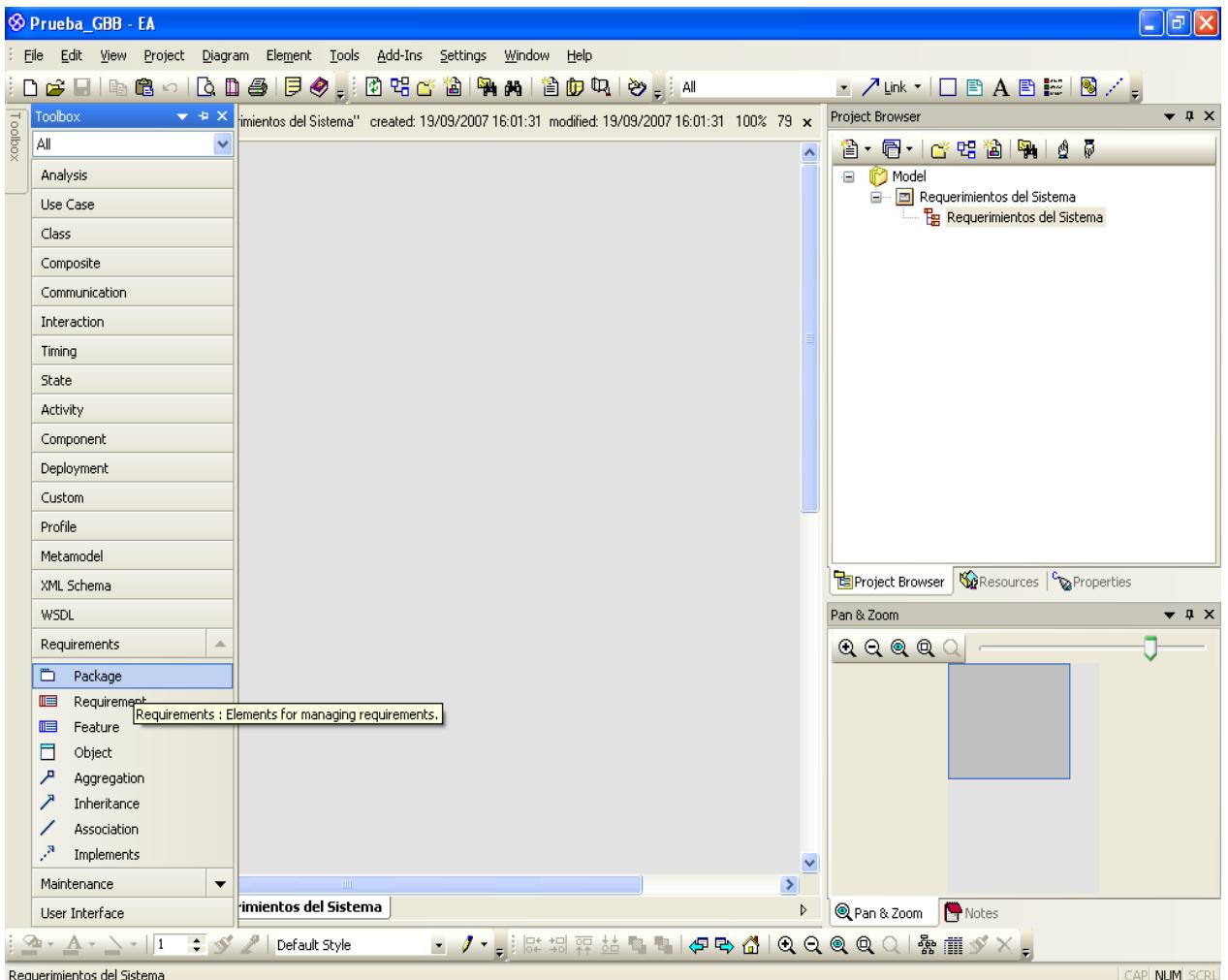
Una vez creado el paquete, se selecciona y en el menú emergente que se muestra a través del botón derecho del ratón, se ha de seleccionar la siguiente opción **Add -> Add Diagram**



A continuación se selecciona la opción que se muestra en la imagen y se pulsa el botón **Ok**

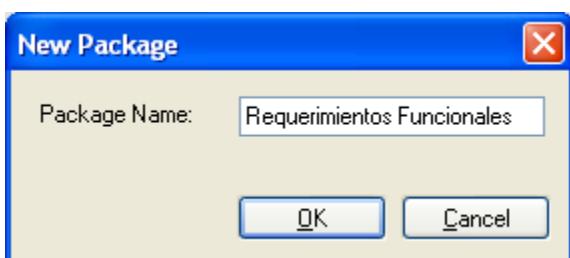


Una vez creado el diagrama se deberá realizar el diseño del mismo. Para ello al acceder a la **ToolBox**, se encontrara activa la pestaña de diseño de **Requirements**

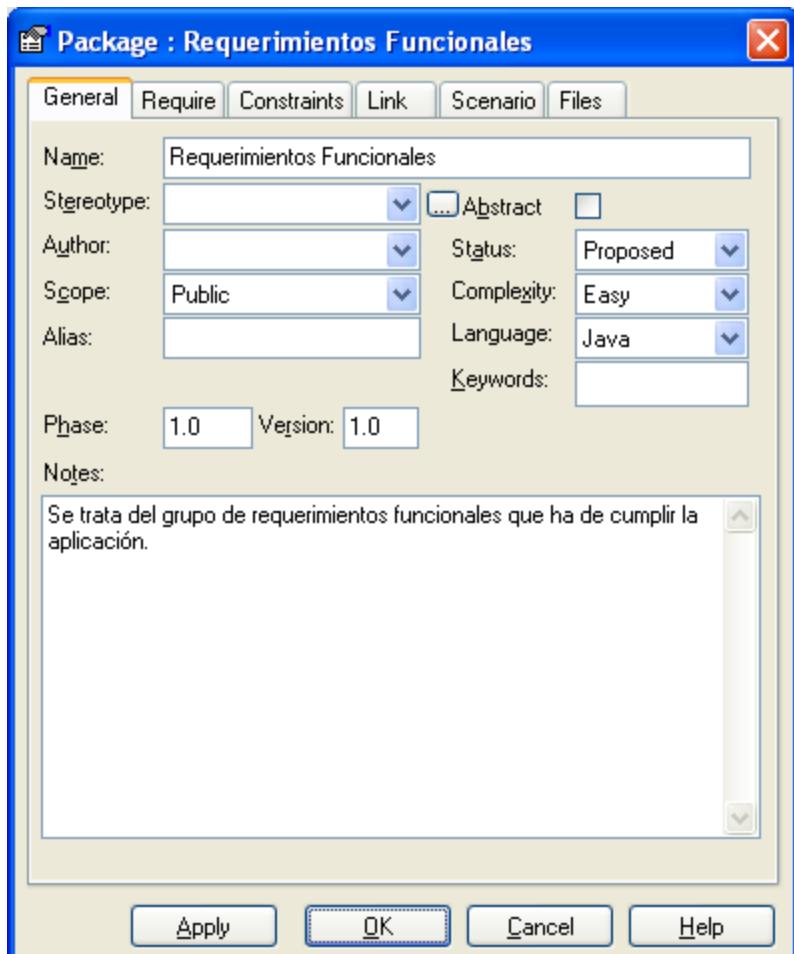


Para la realización del diseño se deberán crear tantos paquetes como grupos de funcionalidades se vayan a recoger en la aplicación. Para ello se deberá arrastrar desde la caja de herramientas el objeto **Package** al diseño.

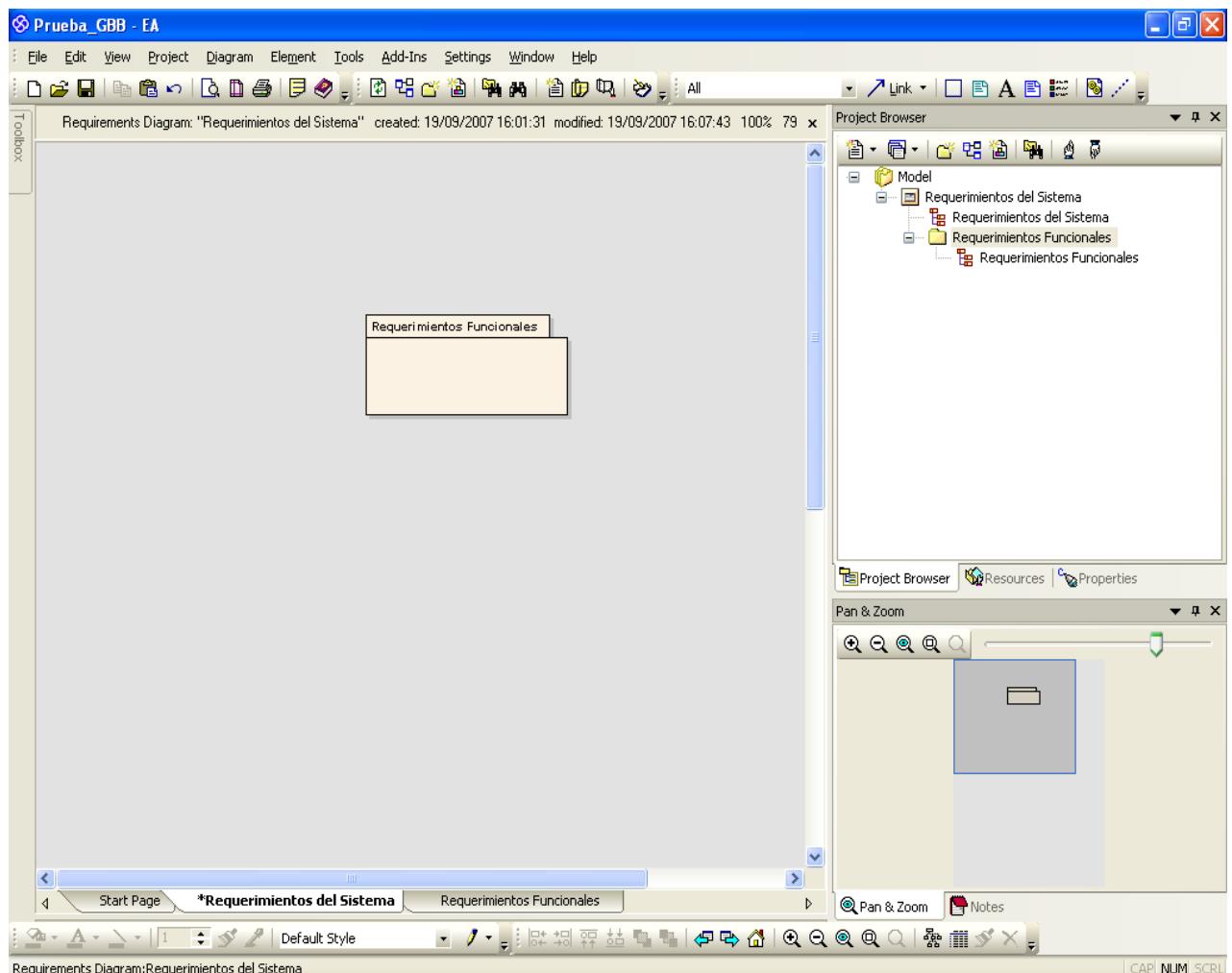
Una vez arrastrado, se solicitará que se introduzca el nombre del paquete



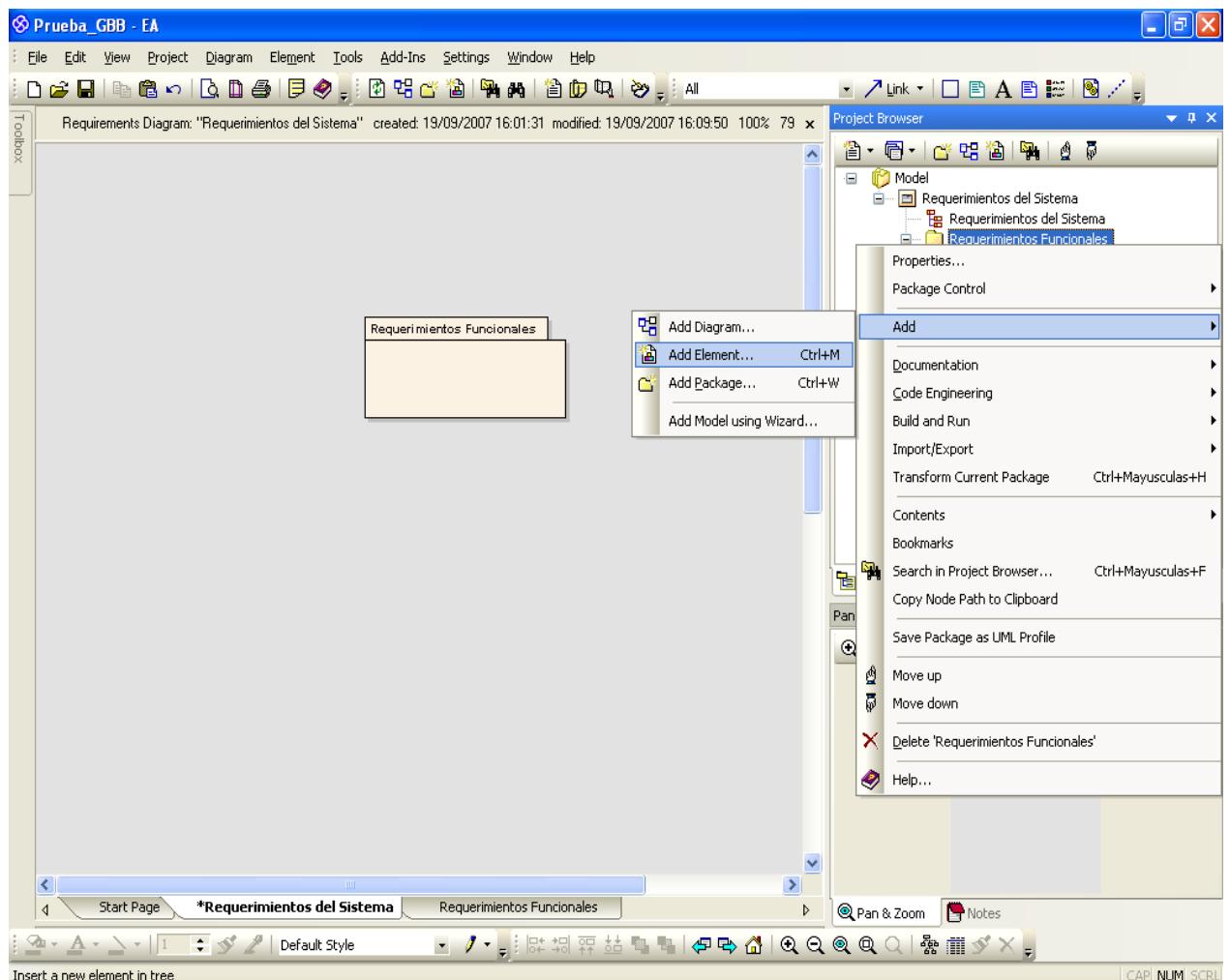
Y se mostrará la ventana de propiedades del mismo:



A continuación el diagrama presentara el siguiente aspecto:

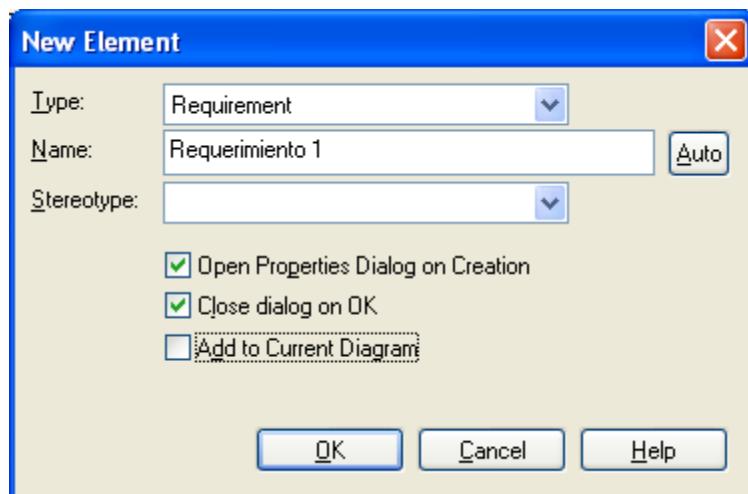


Definidos los distintos tipos de requerimientos del sistema, el siguiente paso es agregar cada uno de los requerimientos en el tipo correspondiente. Para ello se deberá seleccionar el tipo de requerimiento en el que se quiere agregar un nuevo requerimiento y del menú emergente del botón derecho del ratón seleccionar la opción **Add -> Add Element**



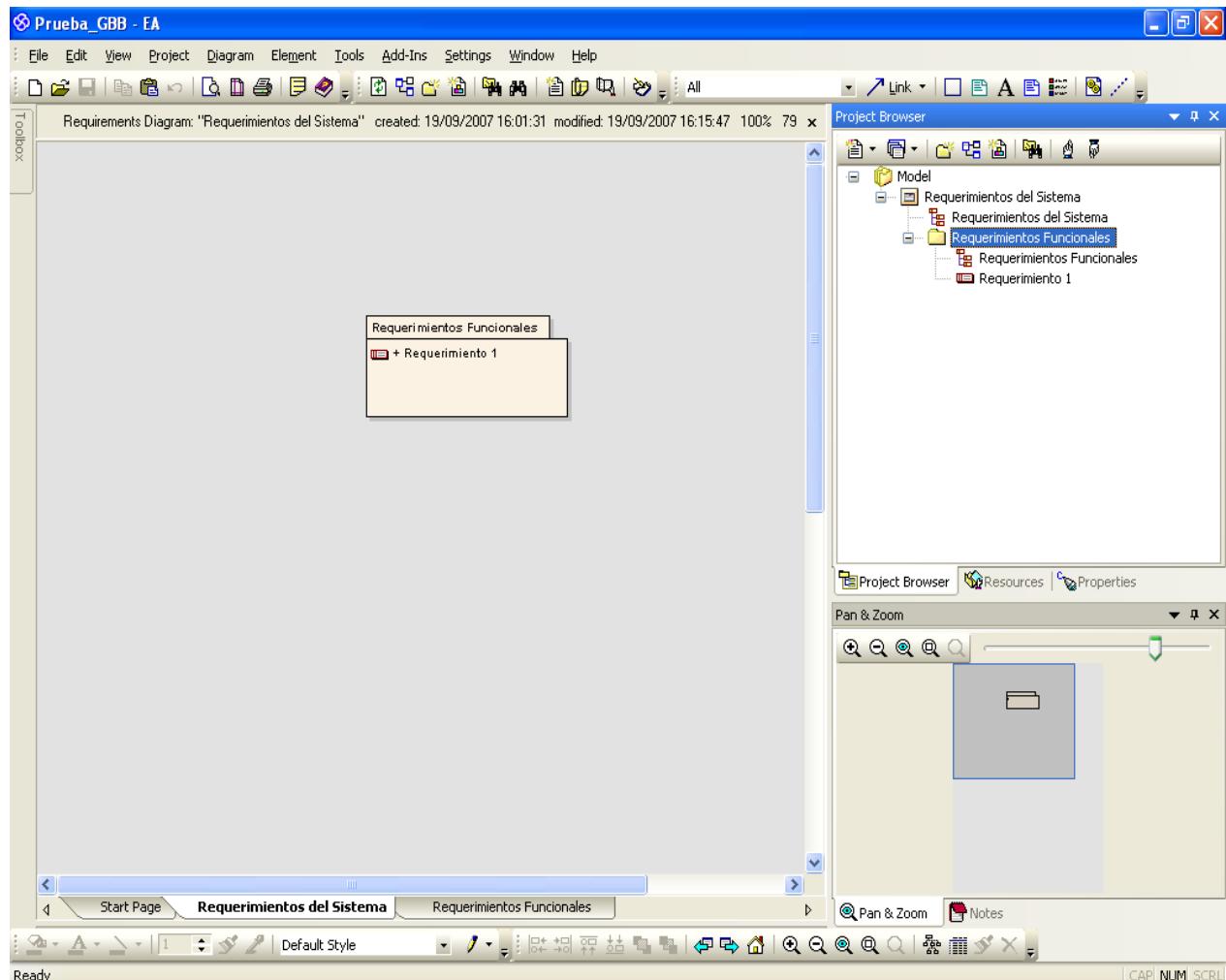
A continuación se mostrara una ventana con las siguientes opciones:

- **Type:** Tipo del elemento a agregar
- **Name:** Nombre del elemento.
- **Stereotype:** Estereotipo del elemento a añadir.



Por defecto, el campo **Stereotype** tiene el valor **Functional**. Se puede introducir un estereotipo particular que la herramienta es capaz de almacenar para la siguiente operación.

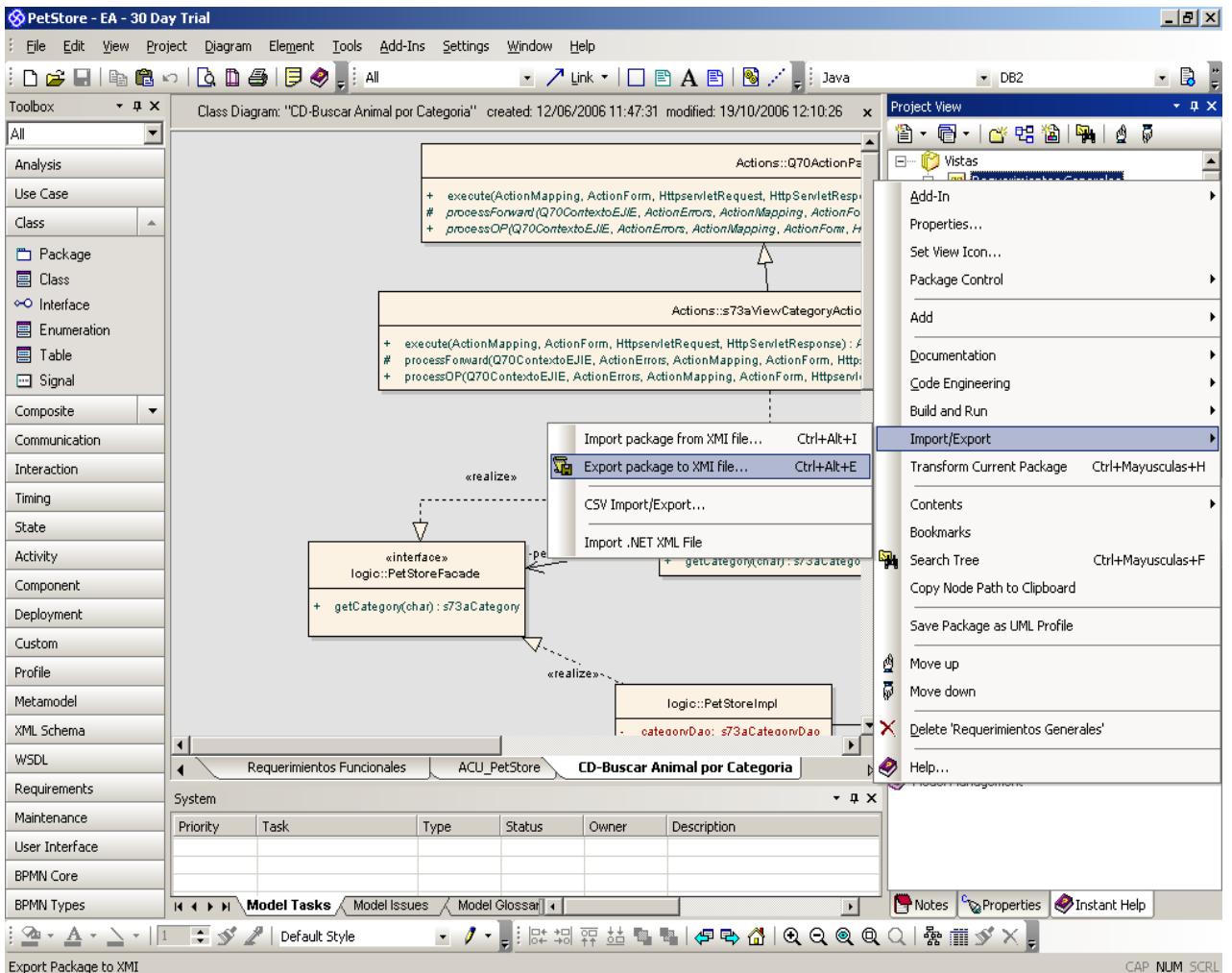
Finalmente el diagrama de requerimientos quedara tal y como se muestra a continuación:



### 3.4 Exportación/Importación XMI

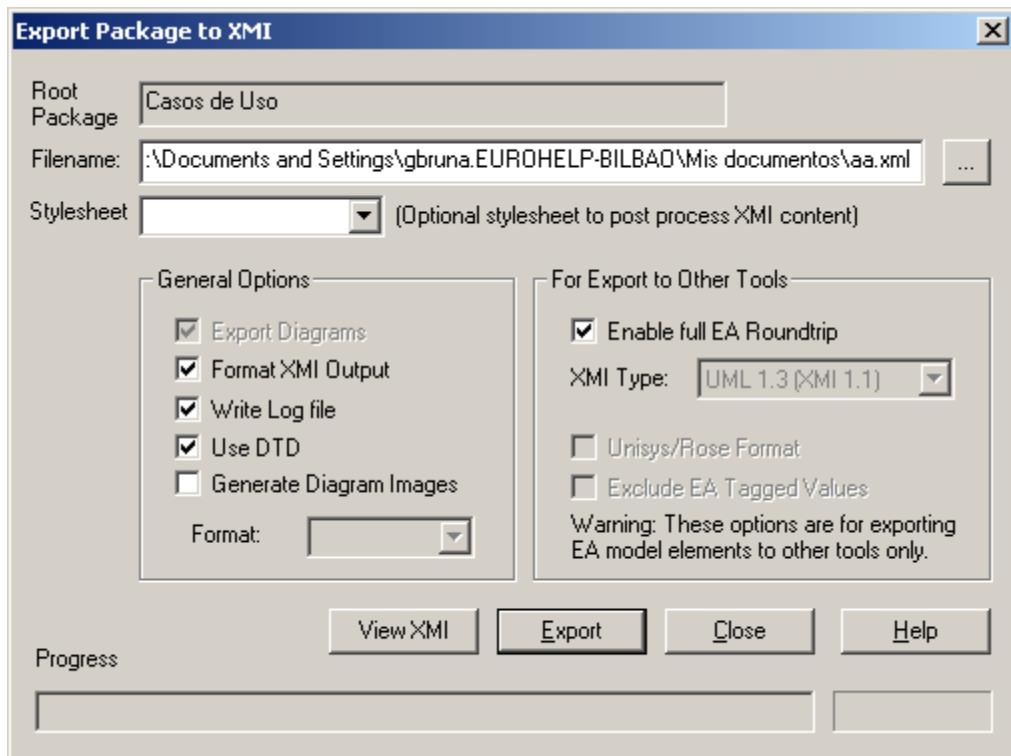
Una de las importantes ventajas que proporciona Enterprise Architect, es la implementación que realiza de la especificación de intercambio de diagramas (XMI) que le permite la importación y exportación de los modelos hacia y desde otras herramientas de modelado.

Para la realización de la exportación se deberá seleccionar de la **Project Browser**, el paquete que se quiera exportar y pulsar el botón derecho del ratón y seleccionar **Import/Export -> Export package to XMI file**



Se mostrará un cuadro de dialogo con las siguientes opciones a configurar:

- **Filename:** Nombre del fichero al que exportar el diagrama
- **Stylesheet:** Plantilla a utilizar en la exportación de los datos.
- **Export diagrams yes/no:** Seleccionar si se exportan los diagramas
- **Use Unisys Rose Format:** Seleccionar si se exporta en formato Rose UML 1.3 o en formato XMI 1.1
- **Format XML output - yes/no:** Seleccionar si se genera un XML de salida legible (Implica una mayor duración del proceso)
- **Write log file:** Seleccionar si el proceso se escribe en un log. El log se guardara en el mismo directorio que el fichero XML.
- **Use DTD:** Seleccionar si se utilizar UML 1.3 DTD. Así se validara el estado correcto del modelo y que no se han producido errores sintácticos.
- **XMI 1.0:** Indicar si se exportara en formato XMI 1.0
- **XMI 1.2:** Indicar si se exportara en formato XMI 1.2
- **XMI 1.1:** Es la opción por defecto, generar el fichero de salida en formato XMI 1.1
- **Exclude EA Tagged Values:** Elimina la información específica de EA en la exportación.

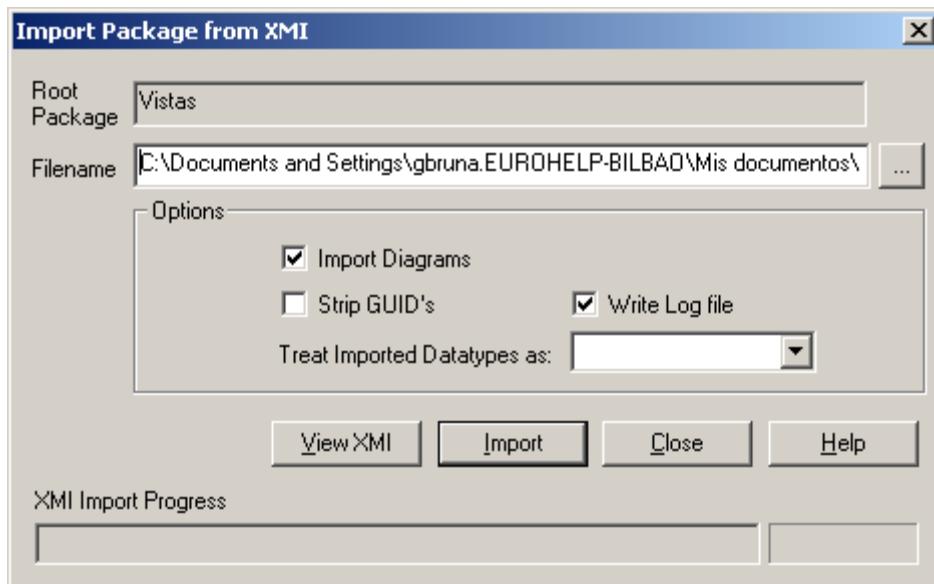


Finalmente pulsando el botón **Export** se realizar la exportación y a través del botón **View XMI** se podrá visualizar el fichero generado.

Es importante indicar que cuando se realiza la exportación con **XMI 1.0** con EA, se produce la perdida de datos debido a las limitaciones de dicha versión.

Para la realización de la importación, del menú anterior se deberá seleccionar **Import Package from XMI**. Se mostrará un cuadro de dialogo con las siguientes opciones a configurar:

- **Filename:** Nombre del fichero que se va a importar.
- **Import diagrams:** Indicar si se van a importar diagramas.
- **Strip GUIDS:** Seleccionar para eliminar la información de los identificadores universales del fichero a importar. Esto permite la importación de un paquete dos veces en el mismo modelo.
- **Write log file:** Seleccionar si el proceso se escribe en un log. El log se guardara en el mismo directorio que el fichero XML.



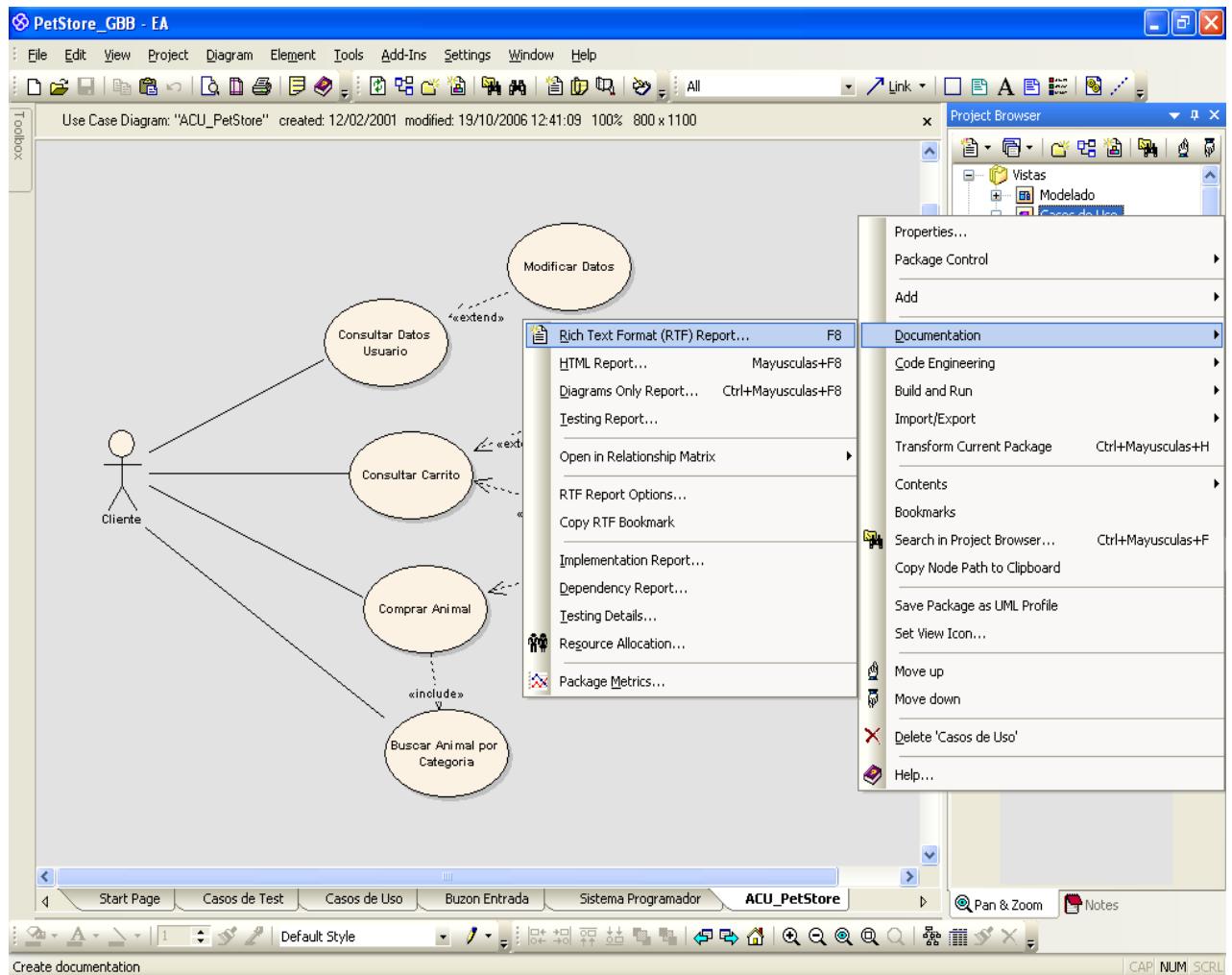
Es importante indicar, que cuando se realiza la importación sobre un paquete, se los pierden los datos existentes al realizar la importación.

### 3.5 Documentación

Una vez creados los modelos, Enterprise Architect proporciona la posibilidad de la generación de documentación a partir de la información introducida en cada uno de los elementos que se han creado en los distintos modelos, para ello posibilita bien la utilización de plantillas prediseñadas o la implementación de nuevas plantillas.

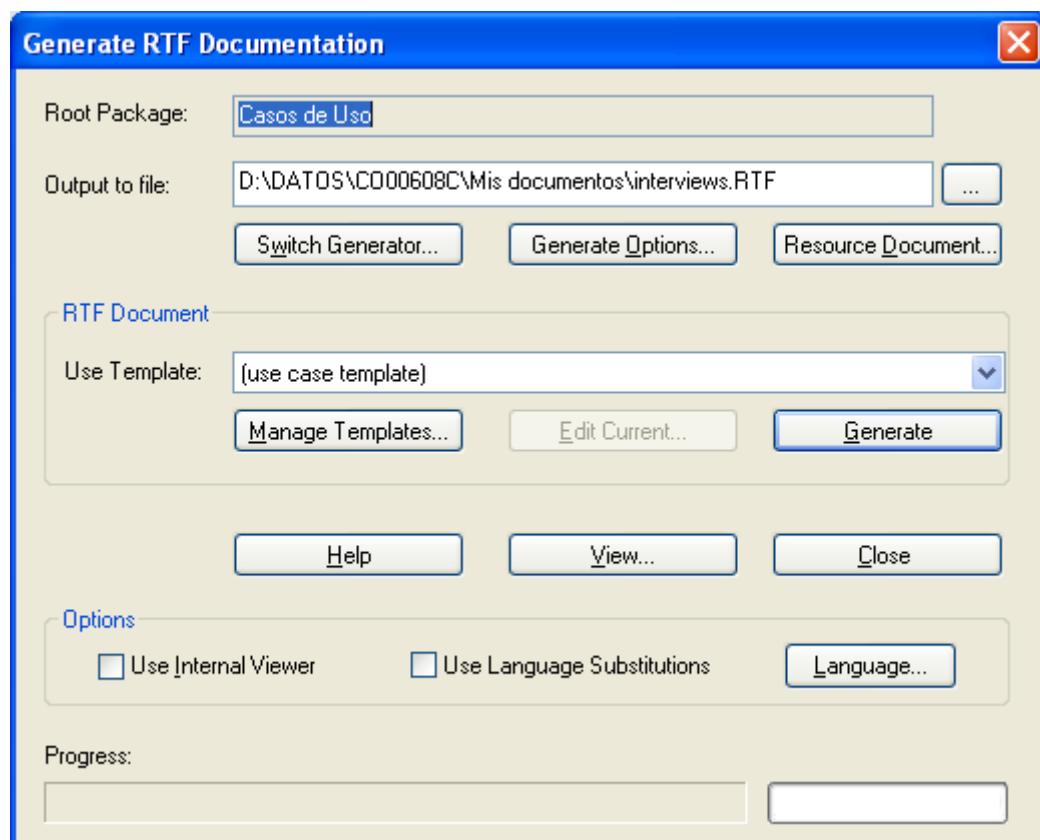
#### 3.5.1. Generación Documentos Word

Para la generación de documentación a partir de las plantillas predefinidas por el sistema, se deberá seleccionar desde la **Project Browser** se quiere exportar a un documento Word, pulsar el botón derecho del ratón y del menú emergente seleccionar la opción **Documentation->Rich Text Format(RTF) Report**



Se mostrará un cuadro de dialogo con las siguientes opciones y botones:

- **Root Package:** El paquete seleccionado para la generación de la documentación
- **Output to file:** Ubicación y nombre del fichero de la documentación generada.
- **Switch Generator:** Permite el acceso a la ventana Legacy RTF Report.
- **Generate Options:** Abre la ventana de opciones del documento.
- **Resource Document:** Permite almacenar la configuración actual como una definición de documento.
- **Use Template:** Permite seleccionar la plantilla a aplicar.
- **Edit Current:** Edita la plantilla seleccionada en el campo anterior. Este campo solo esta activo en caso de que se encuentre seleccionada una plantillada diseñada por el usuario.
- **Generate:** Botón que genera la documentación.
- **Help:** Muestra la ayuda de EA para la generación de documentación.
- **View:** Permite la visualización del documento generado.
- **Close:** Cierra la ventana actual.
- **Use internal Viewer:** Check que implica la visualización de la documentación como una ventana mas del EA.
- **Use Language Substitutions:** Check que activa la sustitución de las palabras configuradas anteriormente.
- **Language:** Botón que abre la ventana de sustitución de palabras.



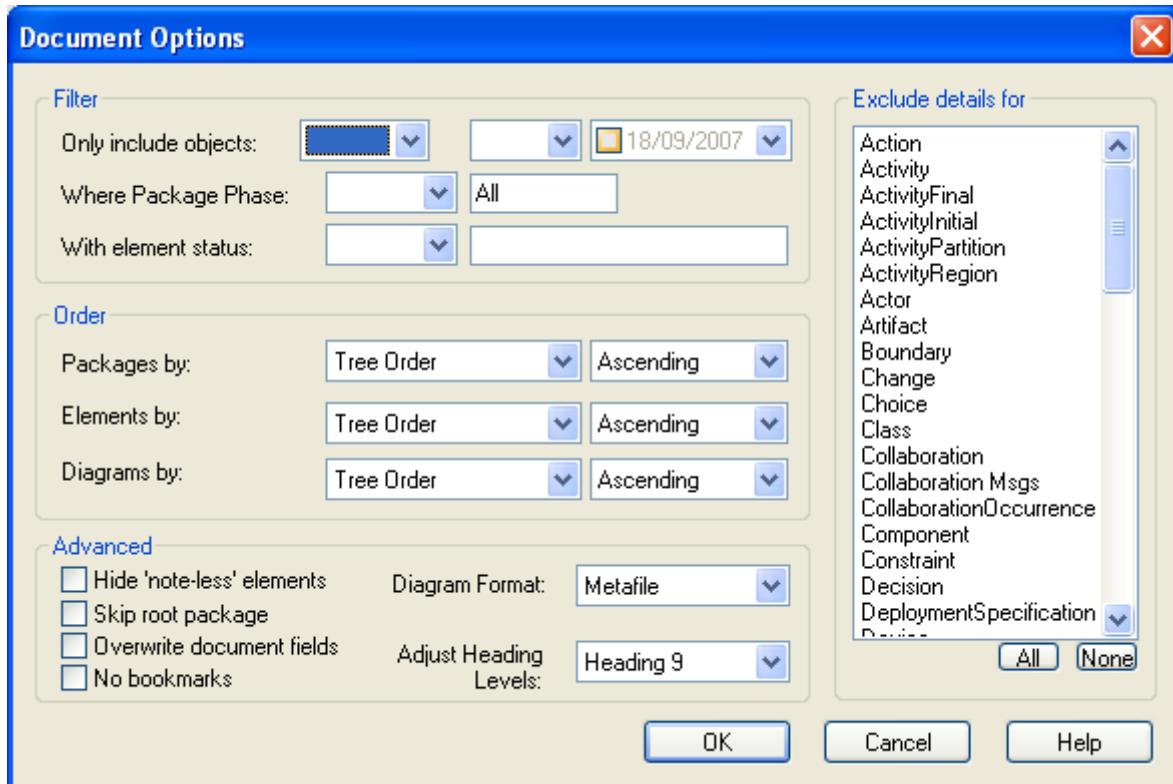
Finalmente con el botón **Generate**, se creara el documento y con el botón **View** se podra visualizarlo.

A continuación se detallaran las distintas opciones que se han presentado en la ventana anterior y que implican acciones mas allá de la selección de un indicador o la introducción de datos:

#### Generate Options

Este botón presenta la ventana que permite gestionar los filtros que se van a aplicar así como el orden en que se van a mostrar los elementos en la generación de la documentación. Se presentan las siguientes opciones de configuración:

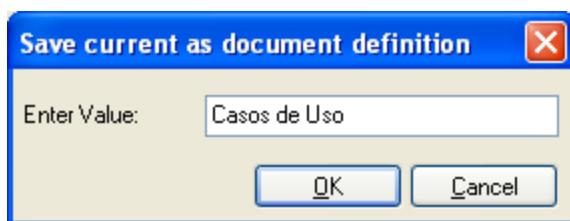
- **Only incluye objects:** Permite filtrar los elementos por la fecha de creación o modificación.
- **Where Package Phase:** Permite filtrar por el valor del campo **Phase** del paquete.
- **With Element Status:** Permite filtrar los elementos en función de su estado.
- **Generate Options:** Abre la ventana de opciones del documento.
- **Packages By:** Ordena los paquetes en orden ascendente o descendente en función del nombre, la fecha de creación o modificación, o en estructura de árbol.
- **Elements By:** Ordena los elementos en orden ascendente o descendente en función del nombre, la fecha de creación o modificación, o en estructura de árbol.
- **Diagrams By:** Ordena los diagramas en orden ascendente o descendente en función del nombre, la fecha de creación o modificación, o en estructura de árbol.
- **Hide 'note-less' elements:** Excluye de la documentación todos los elementos que no tengan notas introducidas.
- **Skip root package:** Excluye el paquete padre de la documentación final y solo muestra los paquetes hijos.
- **Overwrite document fields:** Con la opción deseleccionada se muestran los campos definidos en la plantilla se muestran con sus valores correspondientes. Con la opción seleccionada se sustituirán dichos campos por el texto actual.
- **No bookmarks:** No inserta los marcadores RTF en la documentación.
- **Diagram Format:** Permite indicar el formato de las imágenes de los diagramas a introducir en el documento.
- **Adjust Heading Levels:** Permite gestionar automáticamente al generador los estilos en función de la profundidad del documento.
- **Exclude Details for:** Excluye los elementos seleccionados de la lista del documento final.



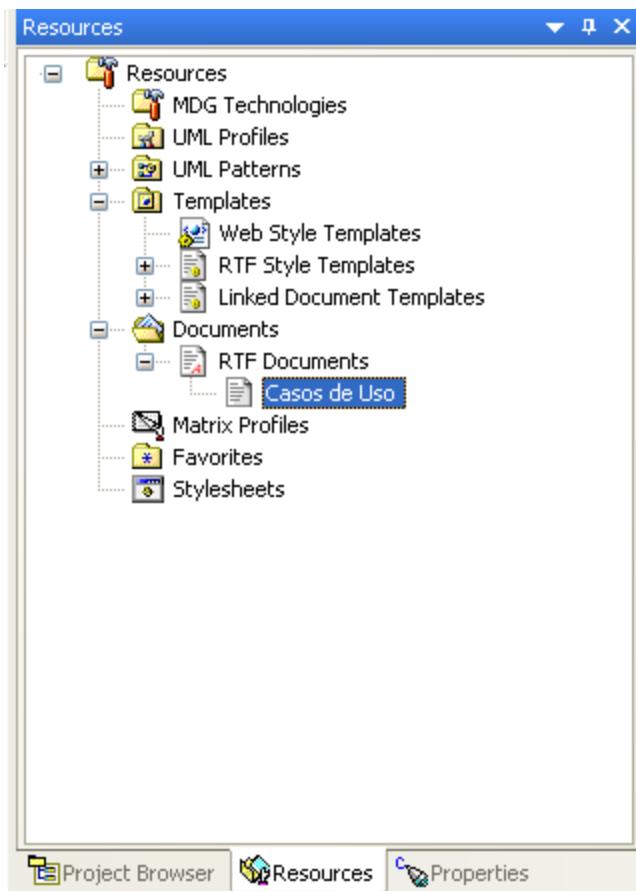
Se ha de indicar que las configuraciones realizadas en la generación de la documentación no serán guardadas, por lo que la próxima vez que se vaya a generar el mismo documento se deberán repetir las especificaciones aquí indicadas.

#### Resource Document

La selección de dicho botón presenta la siguiente ventana



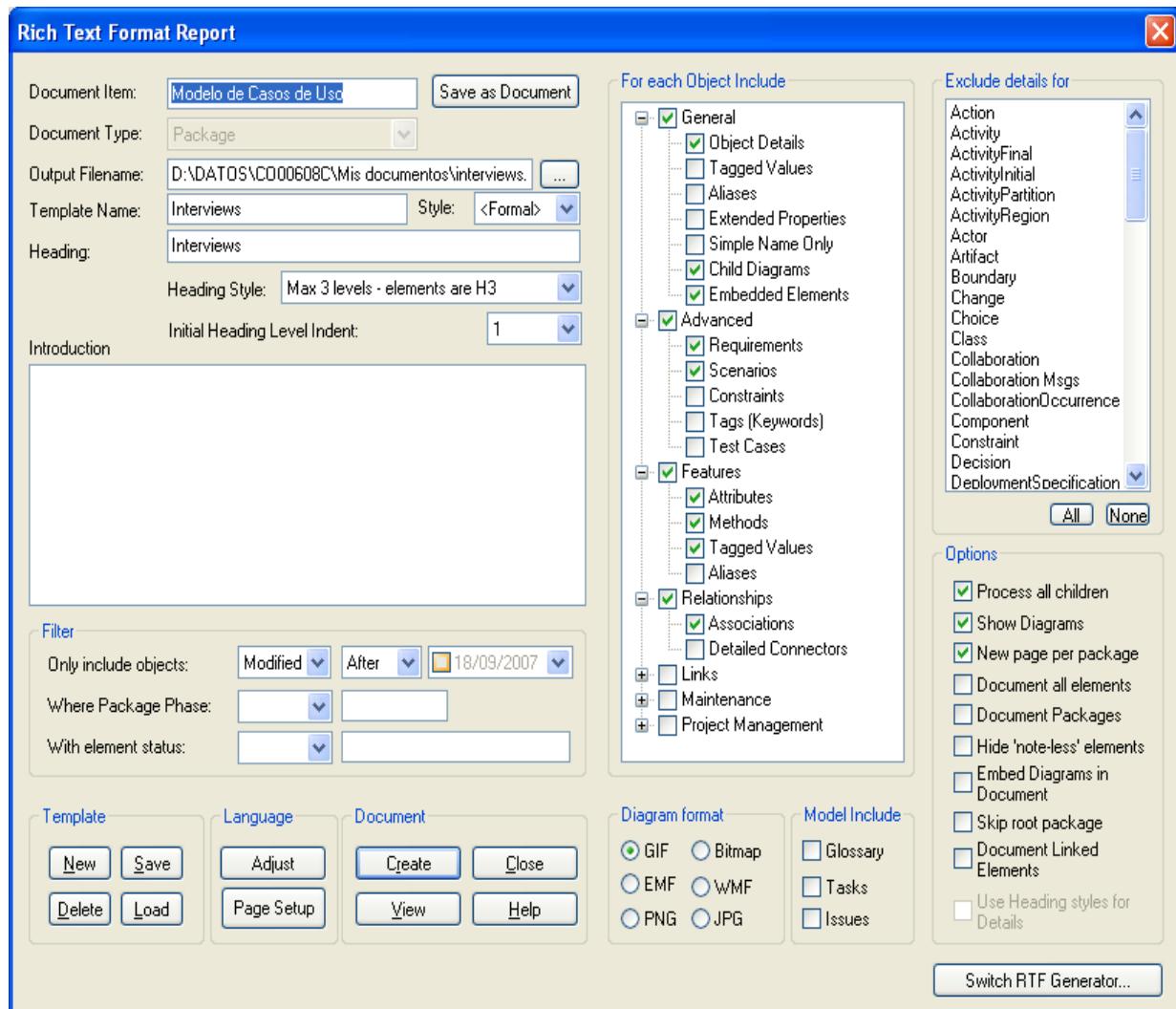
Como se ha comentado anteriormente, permite la grabación de la configuración de los datos introducidos en la ventana **Generate RTF Documentation** y generar de esta manera una plantilla de configuración, que permita su posterior reutilización. Dicha plantilla es accesible desde la vista **Resources -> Documents-> RTF Documents ->**



### Switch Generator

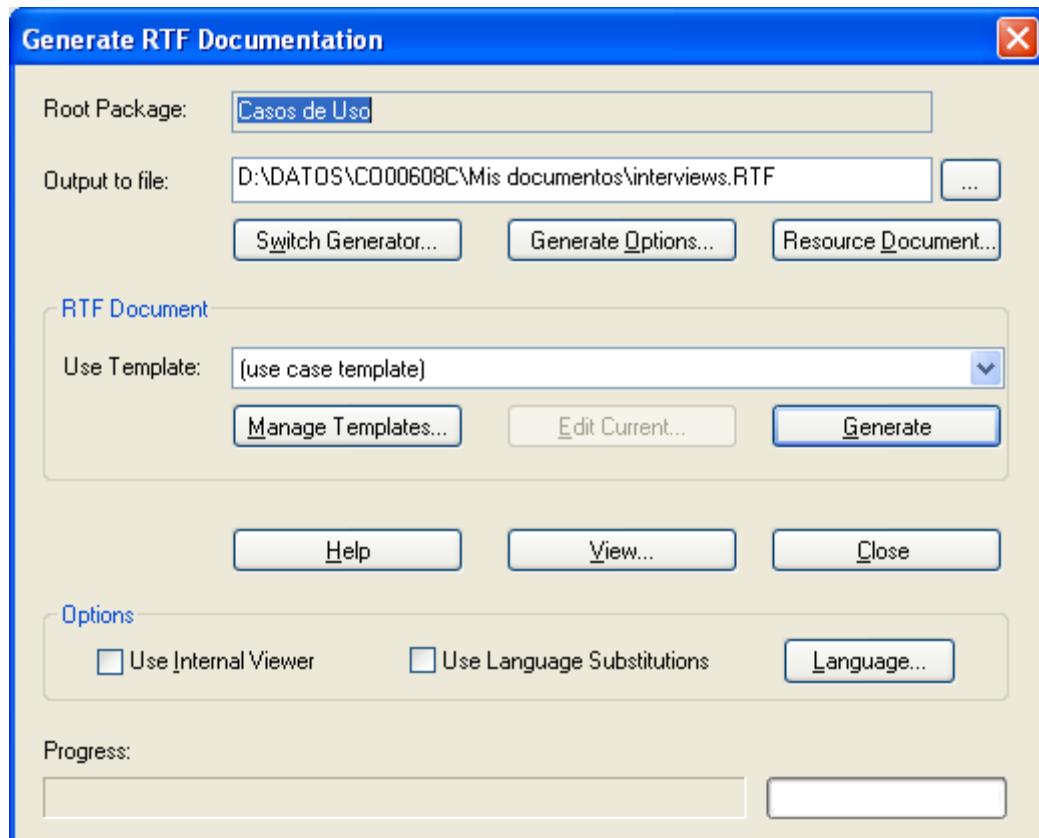
Este botón presenta la ventana RTF Report, que permite la generación de documentación de una forma automática ya que desde este ventana se puede seleccionar el contenido y el formato de los datos a mostrar en el documento final, introducir el nombre del informe, seleccionar la cabecera a utilizar, notas adicionales, la plantilla, etc.

Así mismo, permite crear filtros, detalles a incluir por cada objeto, tipos de elementos a excluir, si se desea procesar los paquetes hijos, si se van a mostrar los diagramas y el formato de los mismos.

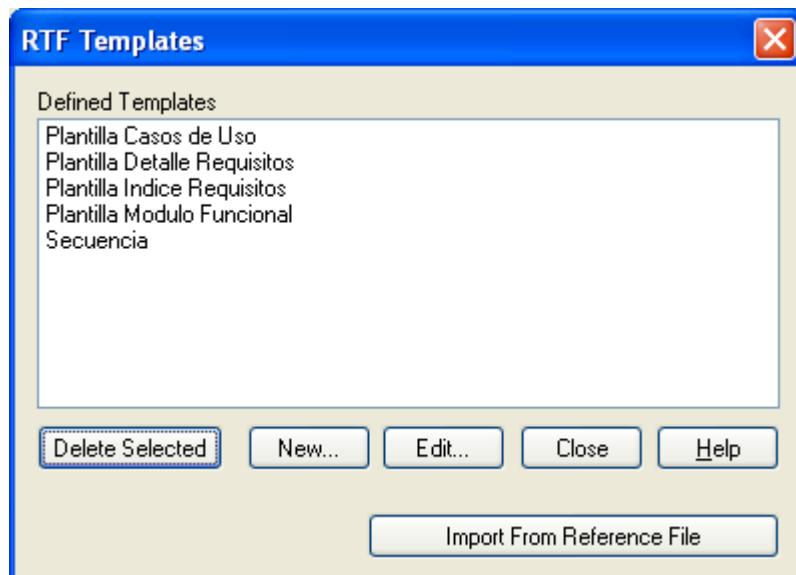


### 3.5.2. Diseño Plantillas

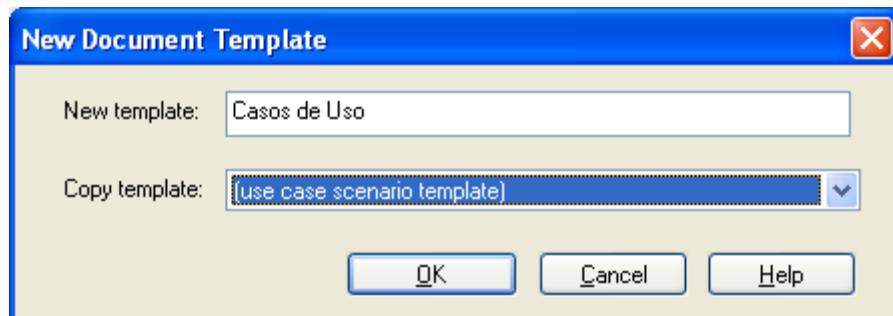
Para realizar el diseño de nuevas plantillas se ha de acceder a la ventana **Generate RTF Documents** tal y como se indica en el punto anterior.



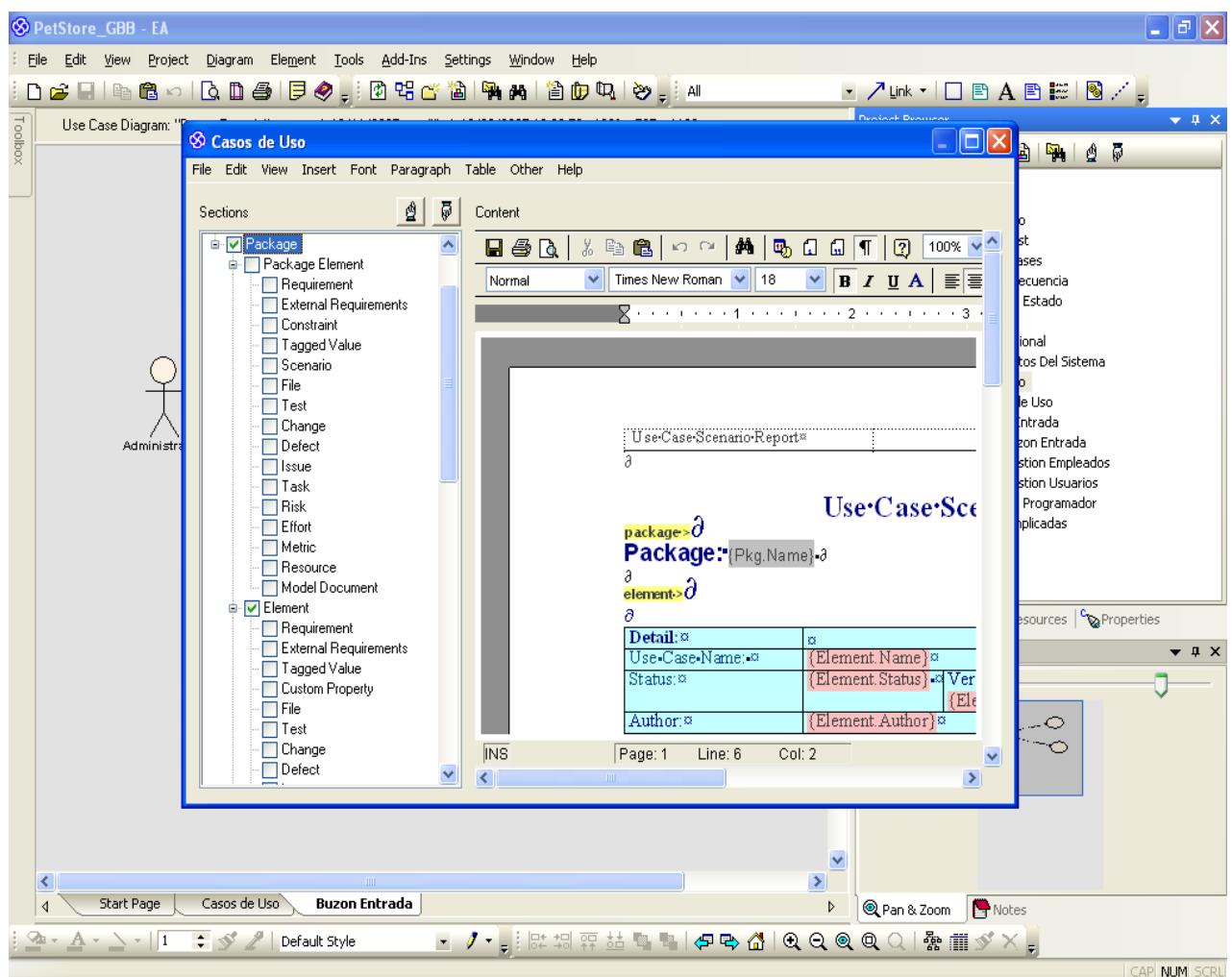
Pulsando el botón **Manage Templates** se accede a la ventana de mantenimiento de las plantillas diseñadas por el usuario.



Para crear una nueva plantilla bastara con pulsar el botón **New** y se mostrara una ventana en la que introducir el nombre de la nueva plantilla y opcionalmente seleccionar una plantilla de la que partir en el nuevo diseño.



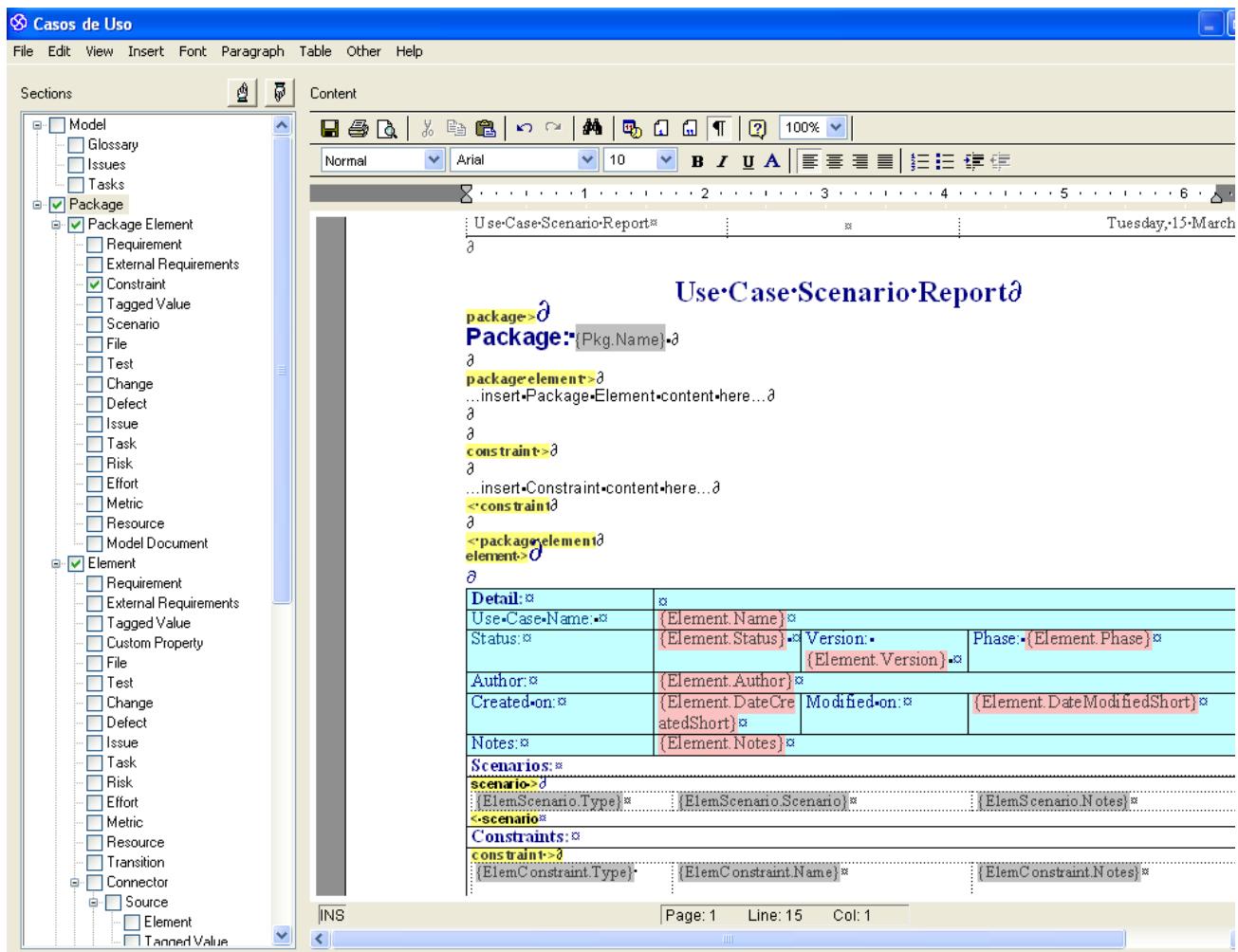
A continuación y pulsando el botón **OK**, se muestra la herramienta de diseño de plantillas



El editor consiste en tres áreas bien diferenciadas:

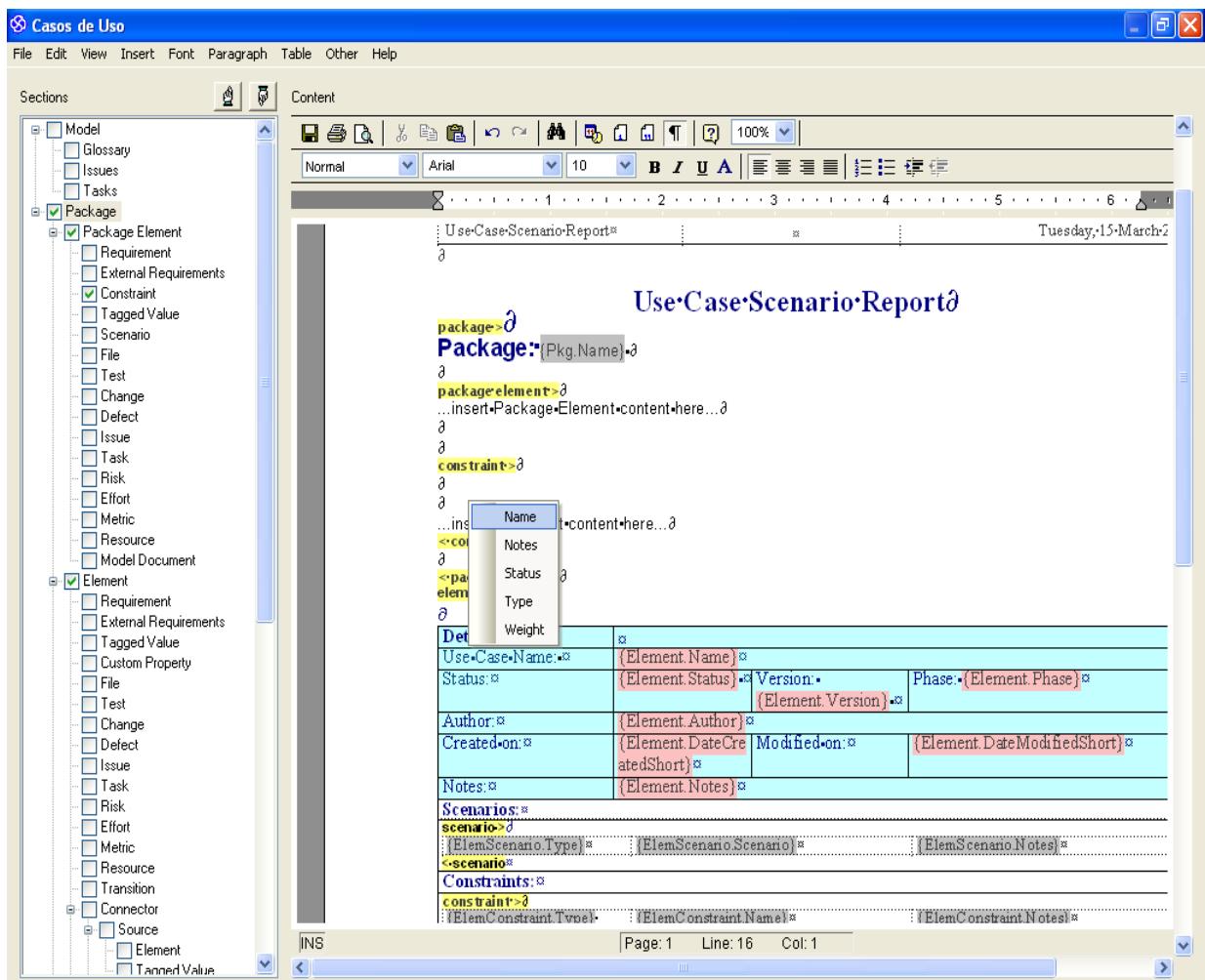
- **Sections:** Se trata de una estructura en árbol que permite seleccionar los distintos objetos que se van a incluir en la documentación final.
- **Content:** Contiene la estructura para la definición de los contenidos del documento. Con el botón derecho y en una sección delimitada por la elección de los elementos del apartado anterior se podrá agregar propiedades de cada objeto seleccionado.
- **Menu:** Proporciona un grupo de funcionalidades para la modificación de textos e inserción de estructuras definidas.

Para agregar una nueva sección en el documento, se deberá seleccionar del apartado **Sections**, aquella que se quiera incluir en el documento. En este caso, se ha seleccionado la sección **Constraint**, del objeto **Package**



Una vez seleccionada se visualizará en el apartado **Content** dicha sección entre tags y en color amarillo, indicativo de que no se puede eliminar manualmente, ya que la única forma de eliminarla es deseleccionarlo del apartado **Sections**.

Una vez seleccionada la sección, ubicando el cursor entre los tags, y a través del botón derecho del ratón se muestran las propiedades del objeto que se pueden integrar en el documento:



Para un mayor conocimiento del manejo del diseñador de plantillas se aconseja la utilización del siguiente manual proporcionado por EA:

[http://www.sparxsystems.com.au/downloads/whitepapers/Working\\_with\\_the\\_RTF\\_generator.pdf](http://www.sparxsystems.com.au/downloads/whitepapers/Working_with_the_RTF_generator.pdf)

### 3.6 Modelo de seguridad

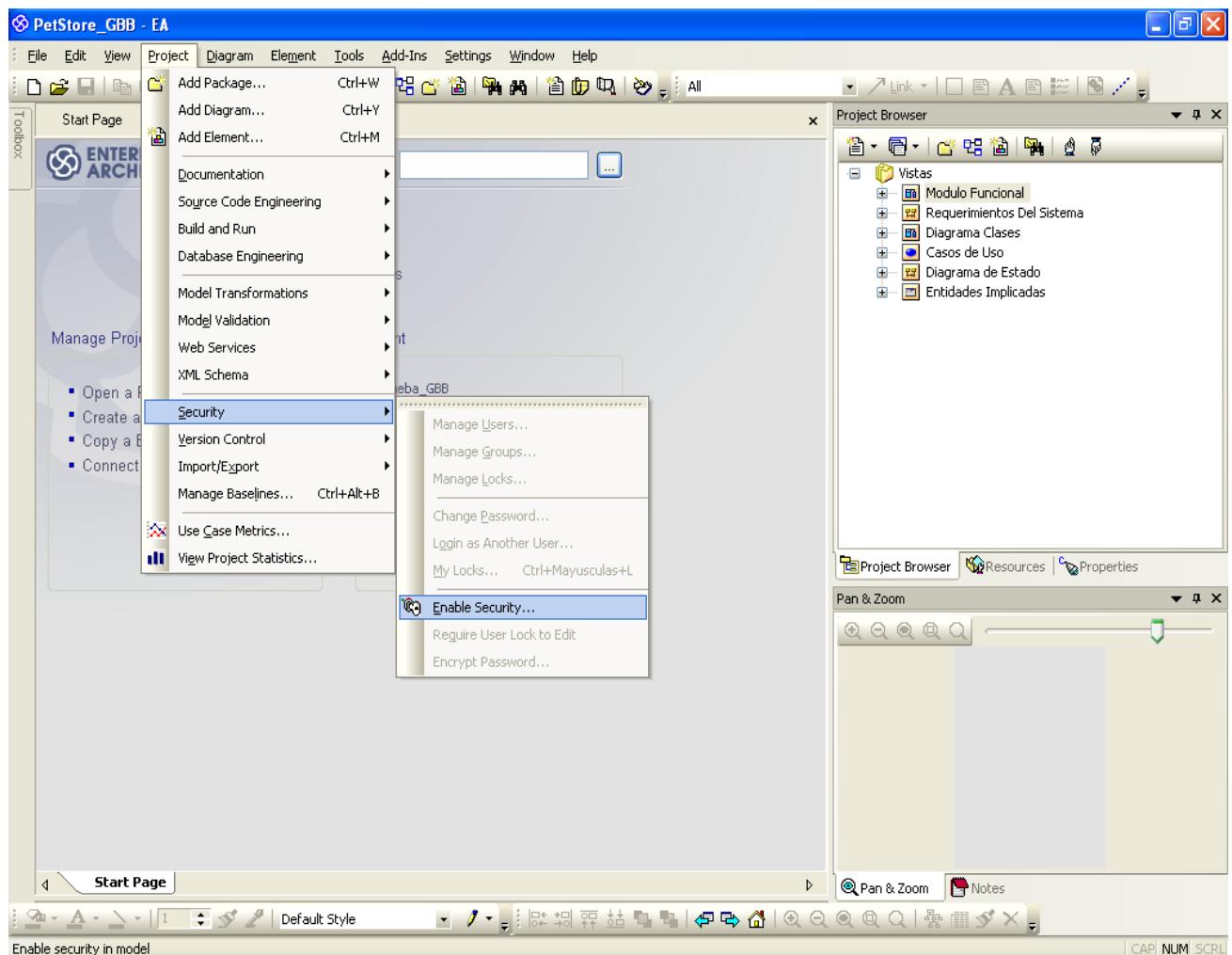
Enterprise Architect Corporate Edition permite aplicar seguridad de acceso al proyecto, además de habilitar el acceso único a cada modelo con objeto de impedir modificaciones paralelas no deseadas.

Aunque en cualquier proyecto, y por defecto en su creación, el modelo de seguridad aplicable a partir del

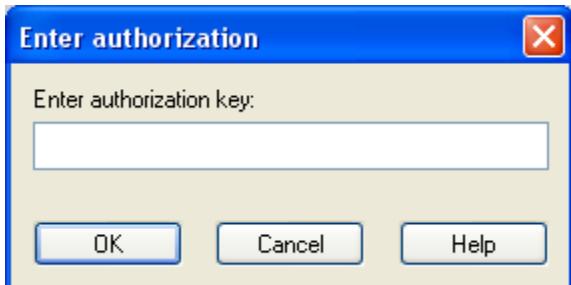
propio Enterprise Architect, está deshabilitado, se recomienda mantenerlo activado. Así, el responsable de proyecto una vez generado (ver el apartado “Crear proyecto” del presente manual) deberá ejecutar los pasos indicados a continuación.

**Importante:** El proceso descrito a continuación solo lo deberá ejecutar el **responsable del proyecto**.

Para poder activar la seguridad se deberá tener activo el propio proyecto. Si éste no dispone de política de seguridades aplicada, se activara accediendo al **Project → Security→ Enable Security**.



Se solicita un código de autorización que se deberá completar.



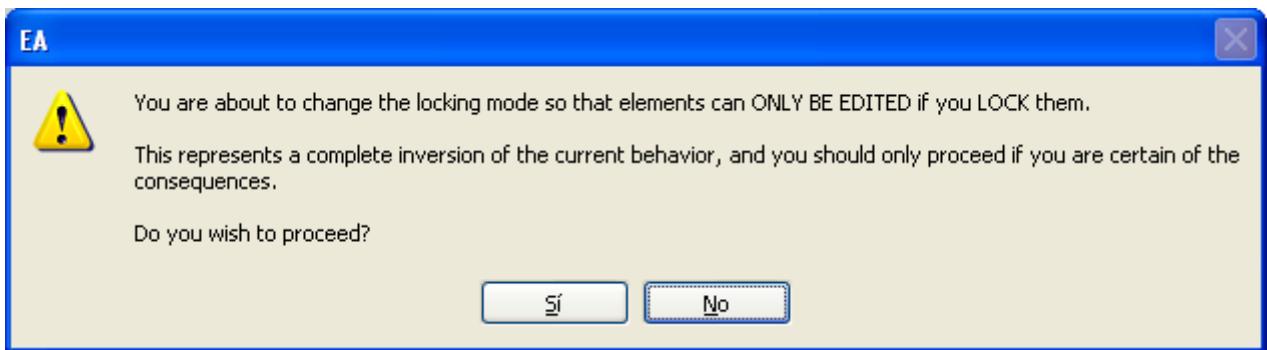
Nota: Esta clave se facilita cuando se hace la compra del producto.

Una vez introducida informa de que además se ha generado un usuario administrador por defecto:



Una vez activada la seguridad se recomienda marcar la opción **Require User Lock to Edit**, lo que garantizará que no habrá actualizaciones indeseables por modificaciones concurrentes de más de un usuario al mismo objeto. Con esta opción el usuario antes de poder realizar cualquier modificación deberá bloquear el objeto, asegurando entonces su uso exclusivo. Así, una vez finalizado el cambio deberá desbloquearlo.

Enterprise Architect pide confirmación antes de efectuar esta operación.



Enterprise Architect también admite otros modelos de bloqueo menos restrictivos, por lo que será decisión del responsable de aplicación adoptar la mejor opción para cada caso.

Una vez se ha accedido al proyecto con las credenciales autorizadas se podrán realizar sobre el proyecto seleccionado las operaciones asociadas a los diferentes perfiles que se hayan creado para el proyecto, y en función del usuario autenticado podrá efectuar aquellas operaciones que le estén permitidas.

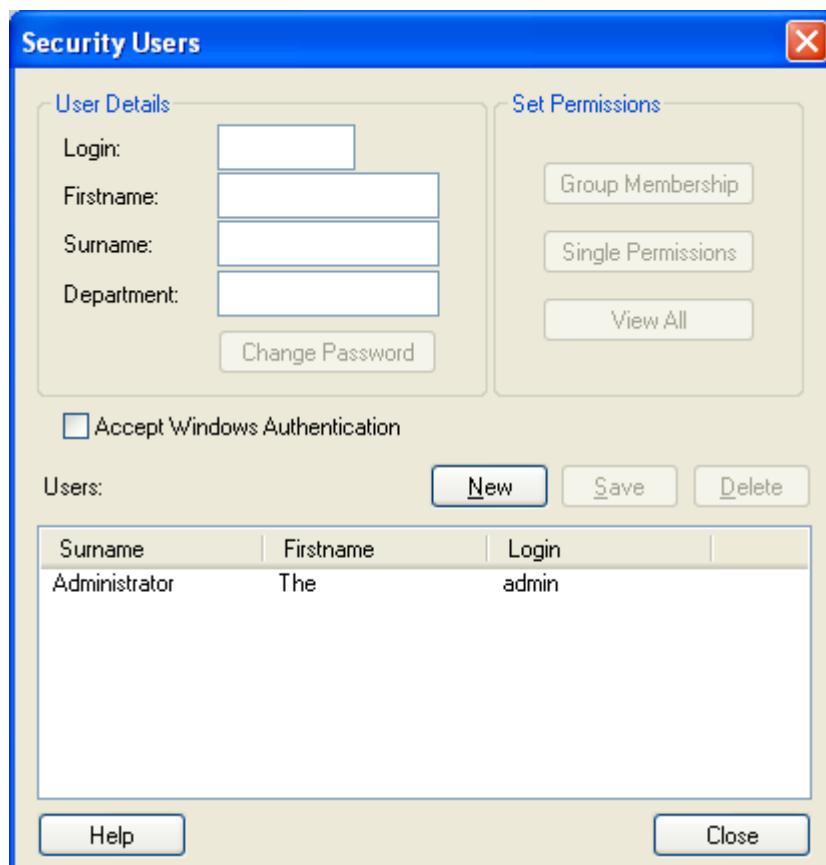
Una vez ha sido habilitada la seguridad, se puede acceder a las otras opciones de administración.

### 3.7 Trabajo en grupo

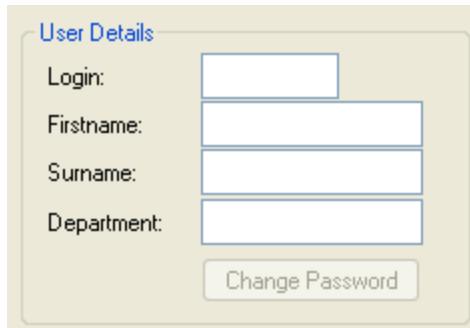
#### 3.7.1. Mantenimiento de usuarios

**Importante:** El proceso descrito a continuación solo lo deberá ejecutar el **responsable del proyecto**.

Para gestionar los usuarios y la asignación de estos a los posibles grupos y permisos identificados para el proyecto acceder al menú **Project→Security→Maintain Users**:



Para efectuar un alta de un usuario en el sistema, se pulsará el botón de **New**, el cual permite acceder al apartado de **User Details** con todos los campos en blanco para proceder a su definición:



The image shows a user interface for 'User Details'. It has a light beige background with a thin gray border. At the top left, the title 'User Details' is displayed in blue. Below the title are four input fields: 'Login' (empty), 'Firstname' (empty), 'Surname' (empty), and 'Department' (empty). Each field is preceded by a label and followed by a small blue rectangular button. At the bottom right of the form is a blue rectangular button labeled 'Change Password'.

Se recomienda utilizar el sistema de credenciales compatible con las propias de acceso al sistema Windows, para ello bastará tan solo con activar la casilla **Accept Windows Authentication**. Una vez introducida la información, se procedera a almacenar la misma pulsando el botón **Save** apareciendo entonces en la lista **Users**.

Para efectuar una actualización de los datos asociados a un usuario en el sistema, se procederá a seleccionar previamente al usuario de la lista, se cumplimentará la información del usuario a modificar existente en el apartado de **User Details** con todos los campos con la información original ya cumplimentada y pulsar **Save**.

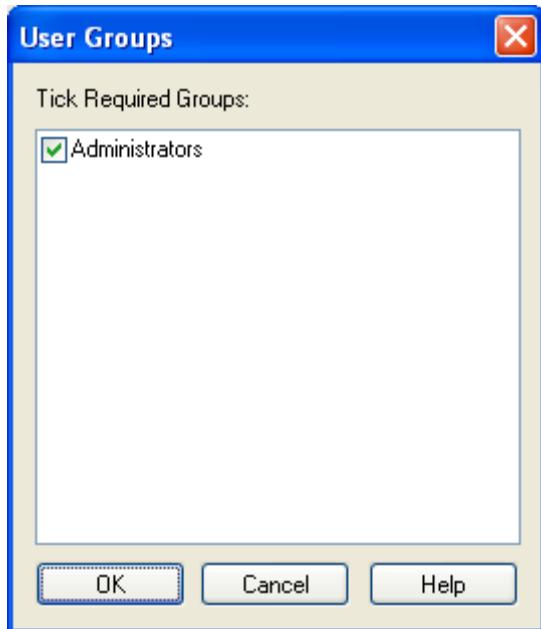
Para efectuar un borrado de un usuario del sistema, se procederá a seleccionar previamente al usuario de la lista, se pulsará sobre el botón **Delete**, el cual realizará el borrado en el sistema. Es de resaltar que el borrado es físico y que el sistema no pide confirmación del mismo, por lo que se deberá estar muy seguro de la operación a efectuar, ya que la misma no es reversible.

Si lo que se desea es asignar o quitar privilegios al usuario, se procederá a seleccionar previamente al usuario de la lista, y en función a los privilegios que se deseen asignar o quitar al usuario, se seleccionará una de las tres opciones posibles:

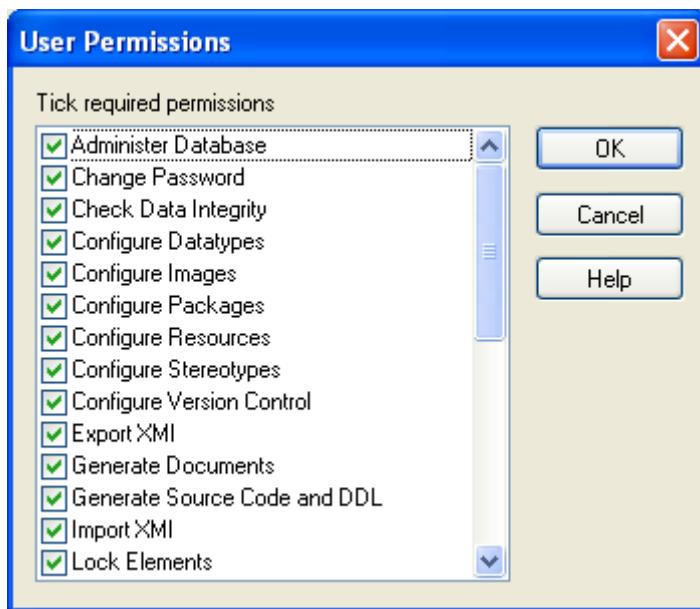


The image shows a user interface for 'Set Permissions'. It has a light beige background with a thin gray border. At the top left, the title 'Set Permissions' is displayed in blue. Below the title are three blue rectangular buttons: 'Group Membership', 'Single Permissions', and 'View All'. The 'Group Membership' button is highlighted with a thicker blue border, indicating it is the selected option.

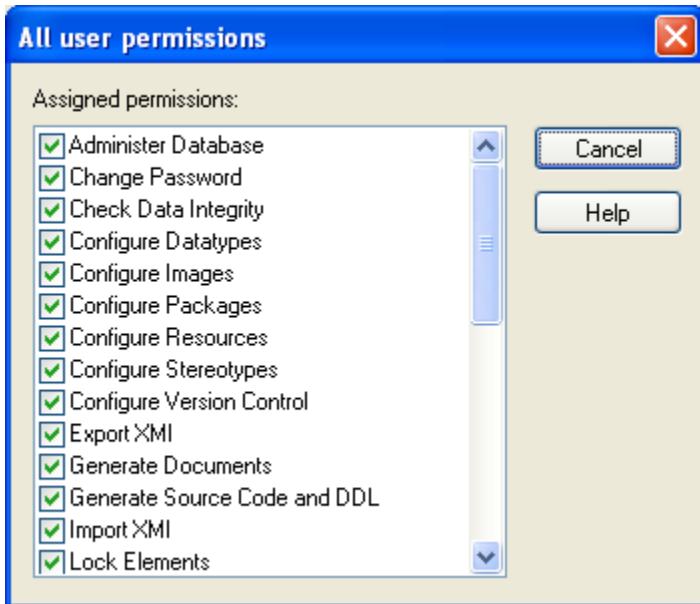
La opción **Group Membership**, permite la asignación/desasignación de un usuario a un grupo. Una vez pulsada esta opción, mediante botones de opción, los que estén marcados generarán la pertenencia a los grupos marcados y a los que no estén marcados, el usuario no dispondrá de pertenencia a ellos:



La opción **Single Permissions**, permite la asignación/desasignación de un usuario a los diferentes permisos existentes en el sistema. Una vez pulsada esta opción, mediante botones de opción, los que estén marcados generarán la pertenencia a los permisos y los que no estén marcados el usuario no dispondrá de pertenencia a ellos. Se recomienda asignar permisos a través de la pertenencia a un grupo al cual se le asignen los permisos deseados:



La opción de **View All** es una combinación de las dos anteriores, por lo que se verán todas las opciones juntas y mezcladas en la misma pantalla de asignación, desasignación.



### 3.7.2. Usuario administrador

**Importante:** El proceso descrito a continuación solo lo deberá ejecutar el **responsable del proyecto**.

Para evitar accesos no permitidos con perfil de administrador es básico actualizar los datos del usuario administrador creado por defecto.

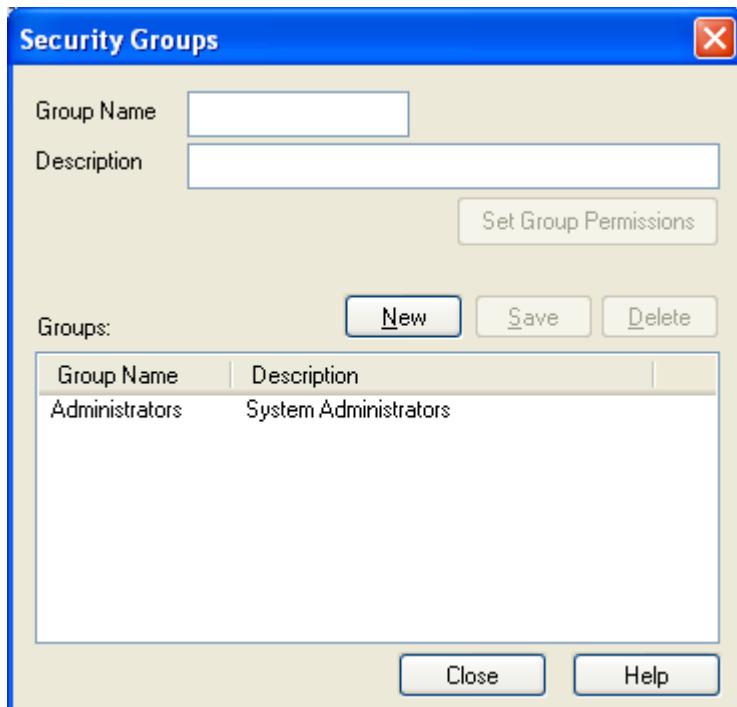
Acceder al menú **Project→Security→Maintain Users**, y seleccionar de la lista **Users**” el referenciado como **Administrador**. Poner como login el código de usuario utilizado para entrar al dominio Windows de EJGVNET y marcar **Accept Windows Authentication**, con lo que no pedirá usuario/password si el usuario se encuentra autenticado en el dominio.

Es recomendable generar otro usuario con permisos de Administrador (perteneciente al grupo “Administrators”).

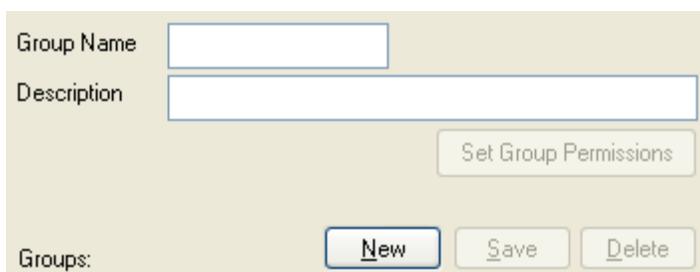
### 3.7.3. Mantenimiento de grupos

**Importante:** El proceso descrito a continuación solo lo deberá ejecutar el **responsable del proyecto**.

Para gestionar los grupos y los permisos o privilegios identificados para el proyecto acceder al menú **Project→Security→Maintain Groups**:



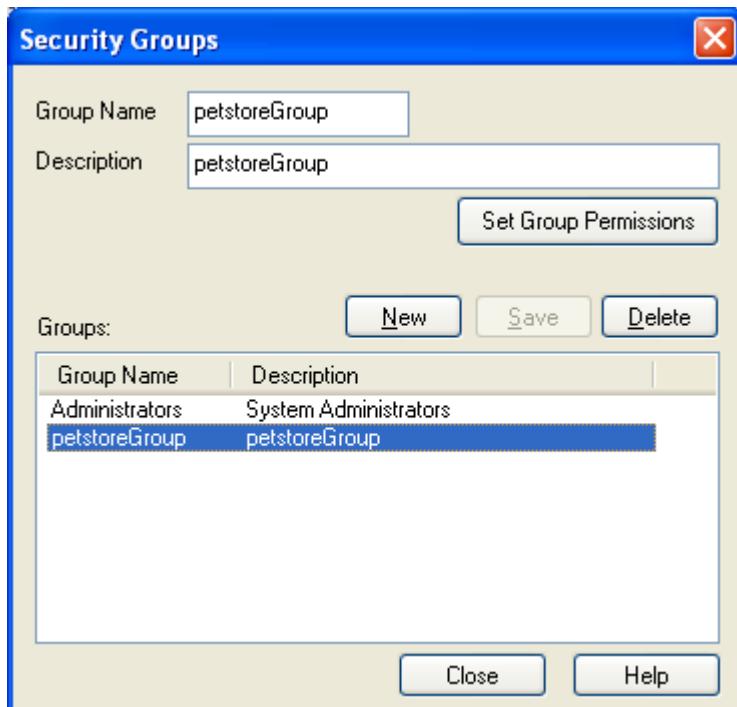
Para efectuar un alta de un grupo en el sistema, se procederá a pulsar el botón de **New**, el cual permite el acceder a la parte superior de la pantalla, con los campos de **Group Name** y **Description** en blanco para proceder a su relleno:



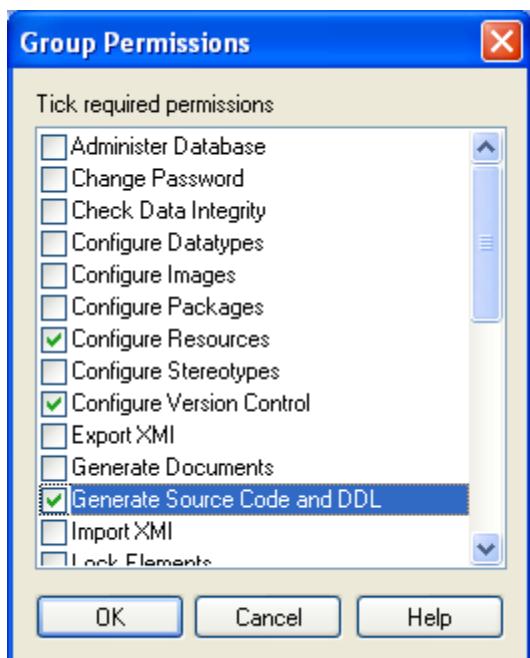
Una vez introducida la información, se procederá a almacenar la misma pulsando el botón **Save**, el cual se habilita en el momento de haber cumplimentado toda la información asociada al grupo.

Para efectuar una actualización de los datos de un grupo en el sistema, se procederá a seleccionar previamente al grupo de la lista, se cumplimentará la información asociada al grupo con todos los campos con la información original ya cumplimentada y una vez introducida la nueva información, se procederá a almacenar la misma pulsando el botón de **Save**.

Para efectuar un borrado de un grupo del sistema, se seleccionará previamente el grupo de la lista, y se pulsará sobre el botón **Delete**, el cual procederá a realizar el borrado del grupo del sistema y a eliminar automáticamente las asignaciones de usuarios pertenecientes a ese grupo. Es muy importante señalar que el borrado es físico y que el sistema no pide confirmación del mismo, por lo que se deberá estar muy seguro de la operación a efectuar, ya que la misma no es reversible.



Si lo que se desea es asignar o quitar privilegios al grupo, se procederá a seleccionar previamente al grupo de la lista, y en función de los privilegios que se deseen asignar o eliminar al grupo, se irán marcando o desmarcando las casillas de opción habilitadas para tal fin:

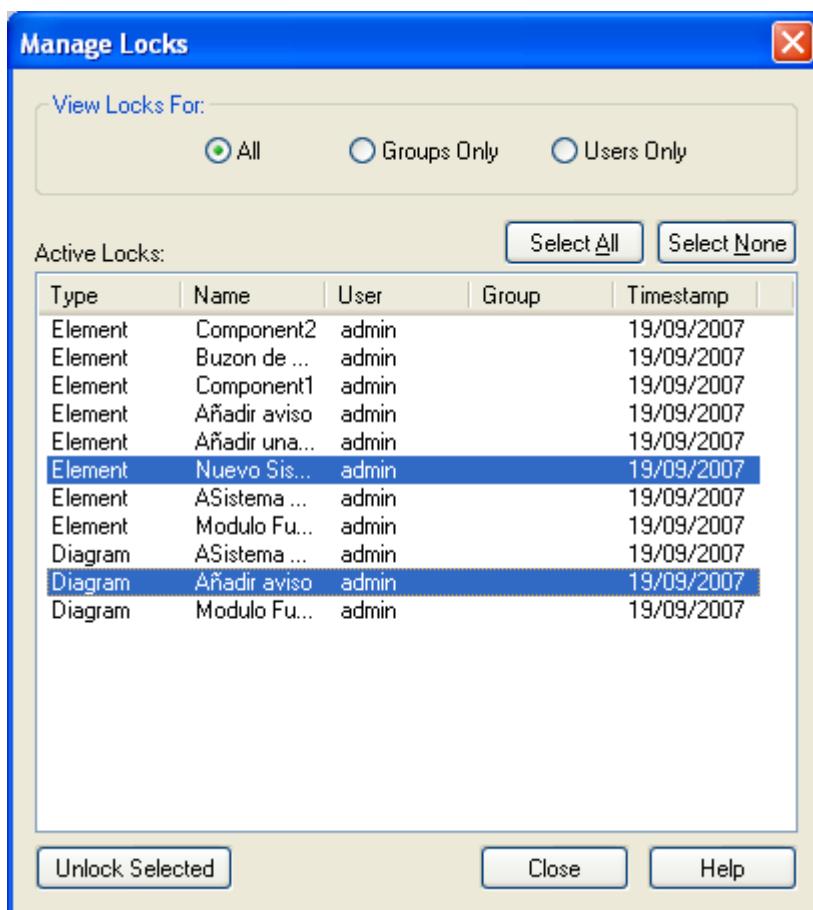


### 3.7.4. Gestión de bloqueos

Si se ha activado la opción **Require User Lock to Edit** (recomendado) cuando un usuario bloquea un objeto, deberá liberarlo explícitamente cuando finalice su tarea de modificación. Esto evidentemente puede generar problemas, bien por olvido, bien porque sea necesario adquirir el objeto en un momento dado y el usuario que lo tiene bloqueado no está disponible, etc.

Es importante recordar que para poder bloquear un elemento el usuario debe tener perfil de administrador o tener asignado el privilegio **Lock Objects**.

Para consultar y gestionar los diferentes bloqueos activos para el proyecto acceder al menú **Project→Security→Manage Locks**:



Es importante recordar que esta opción solo estará activa para usuarios administradores y para aquellos a los que se les asignó el privilegio **Security - Manage Locks**.

Para desbloquear objetos, se puede seleccionar uno por uno de la lista, o pulsando **Select All** se marcarán todos, pinchando después **Unlock Selected** se desbloquerán todos los seleccionados

Si ha activado la seguridad y la opción **Require User Lock to Edit**, para bloquear un elemento, hay que posicionarse en la pestaña **Project Browser** sobre el objeto que se quiere editar, y pulsando botón derecho ejecutar la opción **Apply/Release User Lock...**



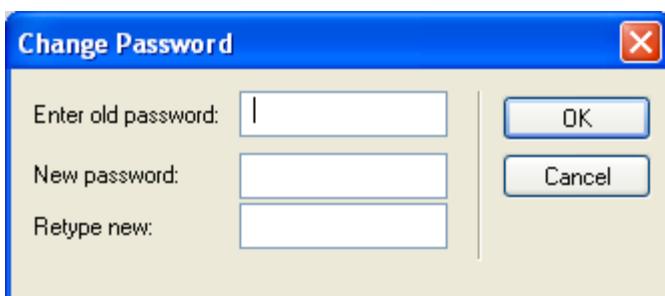
Esta opción de menú también permite el desbloqueo al posicionarse sobre algún objeto bloqueado previamente.



### 3.7.5. Otras funciones

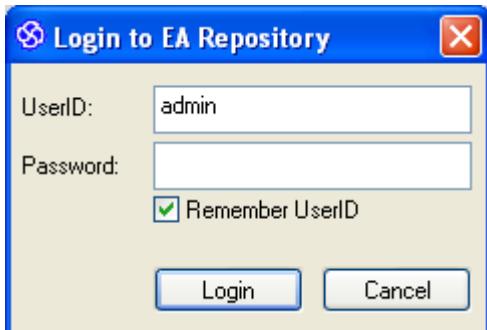
#### Cambio de password

Opción que permite la gestión de la contraseña del usuario activo. Desde este apartado se permite cambiar la contraseña para el usuario que está identificado en el proyecto en ese preciso instante. Para ejecutarlo se debe acceder al menú **Project→Security→Change Password**



#### Login como otro usuario

Desde este apartado se permite cambiar el usuario identificado o autenticado en el proyecto. Para ejecutarlo se debe acceder al menú **Project→Security→Login As Another User**.



Se permite un máximo de tres reintentos en la introducción de un usuario y contraseña válidos para acceder al proyecto. Si por casual se intentara por más de tres ocasiones consecutivas y estas fuesen incorrectas o erróneas, el Enterprise Architect mostrará un mensaje de error, y realizará una salida del acceso al proyecto seleccionado.

### 3.8 Modelado Datos con EA

El modelado y el diseño de base de datos no se encuentran explícitamente cubiertos por la especificación UML, pero si están recogidas en Enterprise Architect en la funcionalidad de Modelado de Datos UML. Dicha funcionalidad permite el mapeo de objetos de base de datos como tablas y sus relaciones, a objetos UML como clases y asociaciones. Así mismo, permite la creación de claves primarias, triggers, constraints, etc.

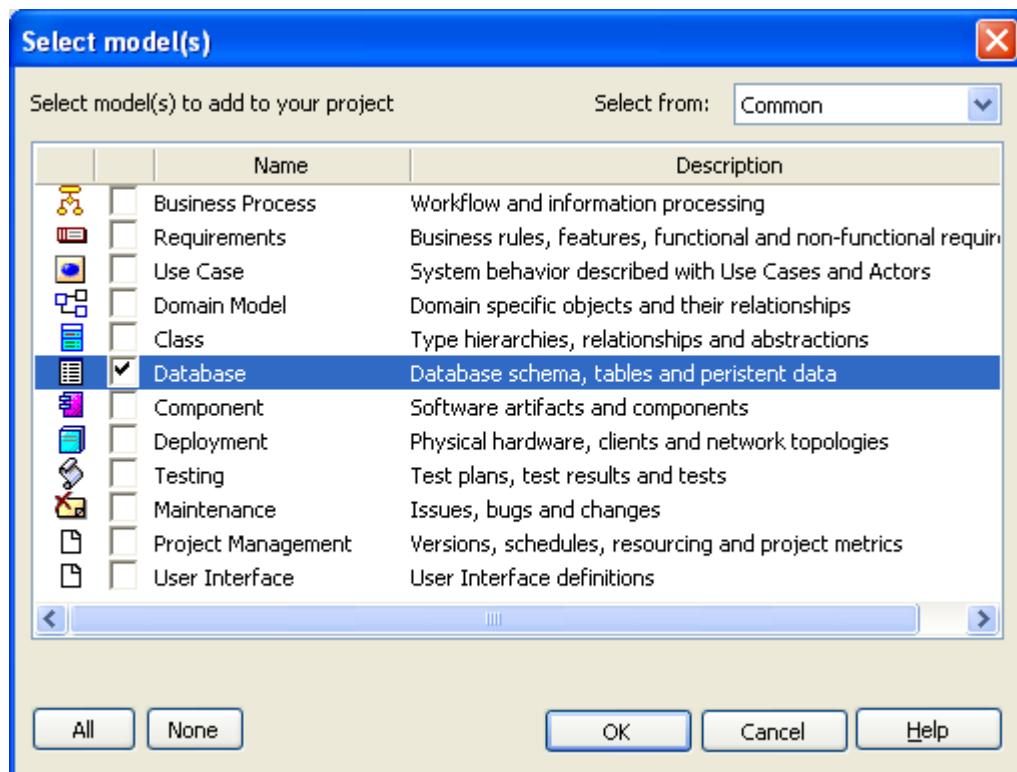
Enterprise Architect soporta la importación de los esquemas de bases de datos de las siguientes bases de datos :

- DB2
- InterBase
- MS Access
- MySQL
- Oracle
- PostgreSQL
- MS SQL Server
- Sybase Adaptive Server Anywhere
- Firebird

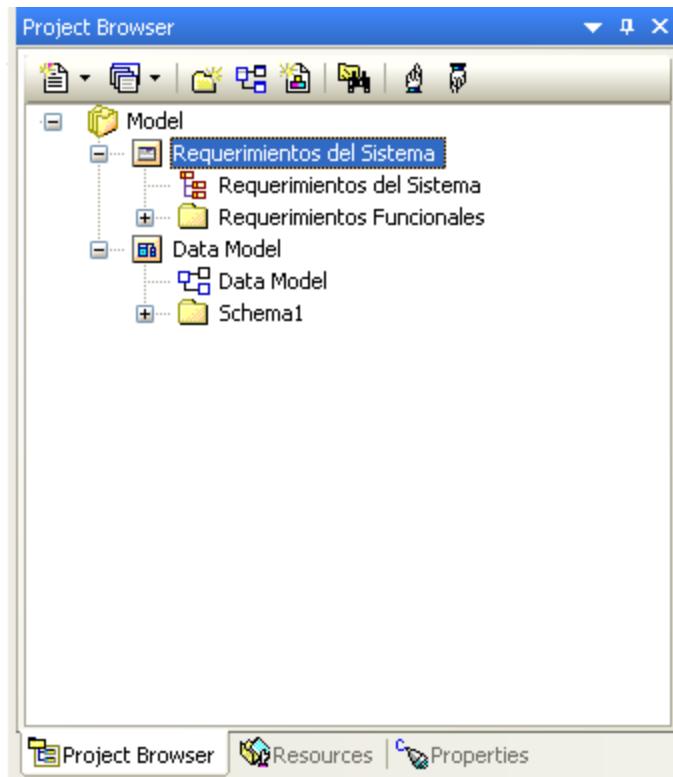
#### 3.8.1. Creación Modelo de Datos

Para crear un modelo de datos, lo primero que se deberá seleccionar es un diagrama en el que ir creando los objetos, para hecho se utilizará como base el modelo que proporciona la herramienta para el diseño de base de datos. Para ello, sobre el elemento root **Model**, pulsar el botón derecho del ratón y del menú emergente seleccionar la opción **Add->Add Model Using Wizard**

De la ventana que se muestra se ha de seleccionar el modelo **Database**.



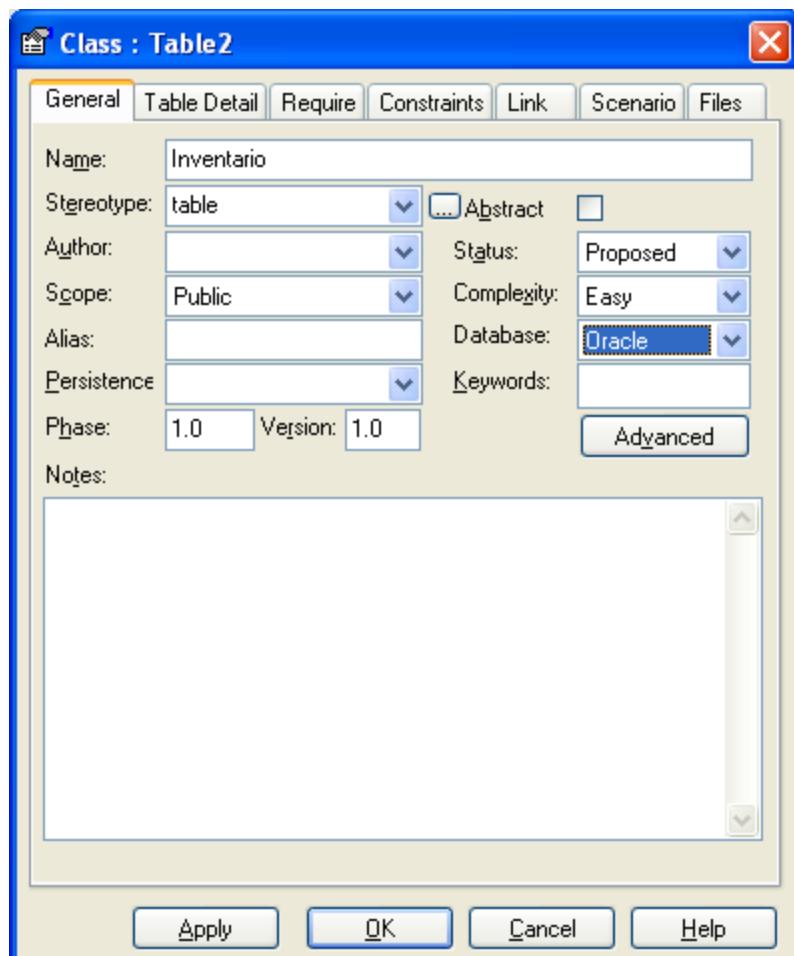
Pulsando el botón **OK** el sistema creara una estructura para el modelado de base de datos, **Data Model**.



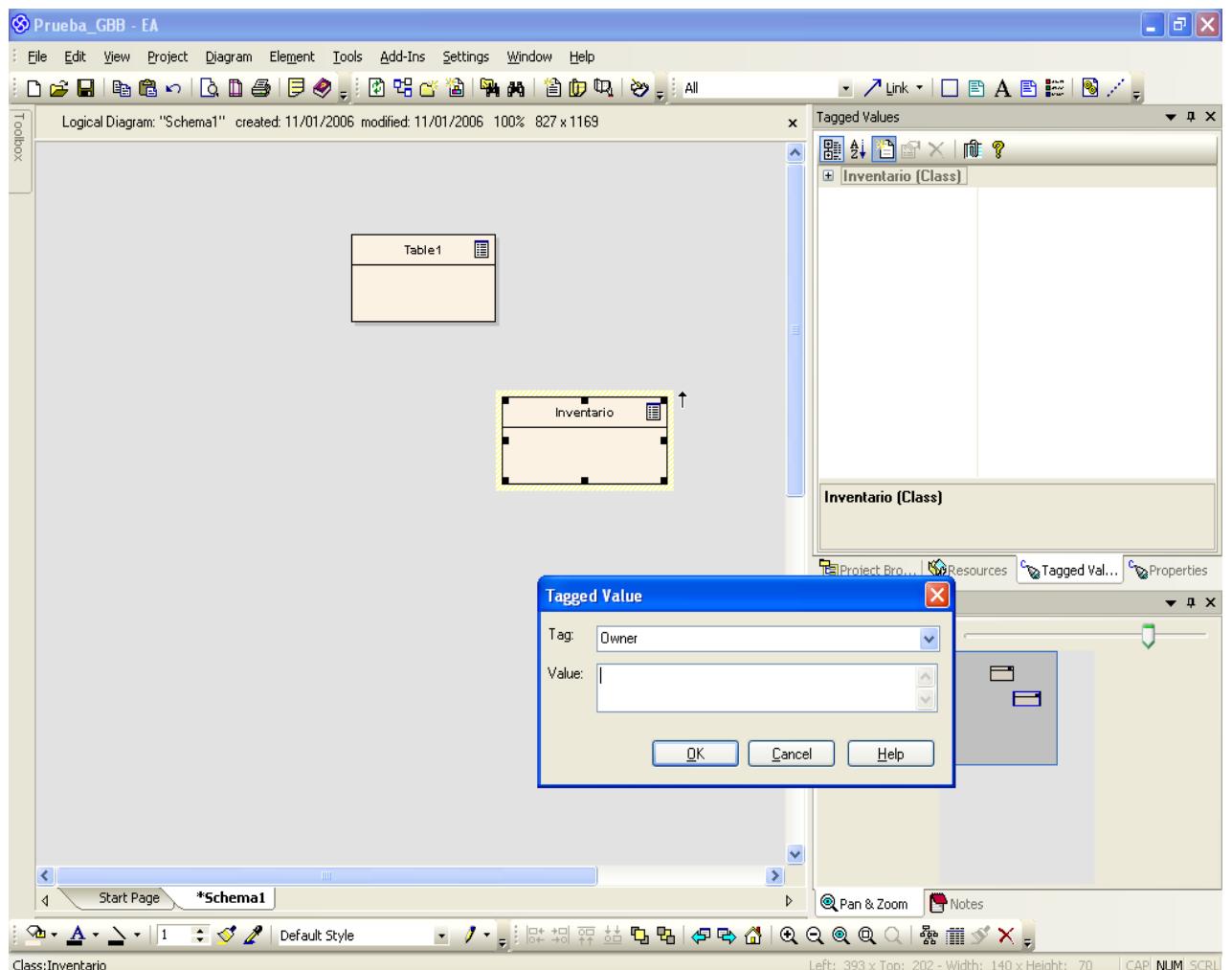
### Crear una tabla

Para crear una tabla en EA, se deberá seleccionar el objeto **Table**, de la sección **Class** de la caja de herramientas y trasladarla al diagrama.

Una vez creada la tabla el siguiente punto es definir a que tipo de Base de Datos va a pertenecer, ya que de esta forma se define la lista de los tipos de datos habilitados para la definición de las columnas.

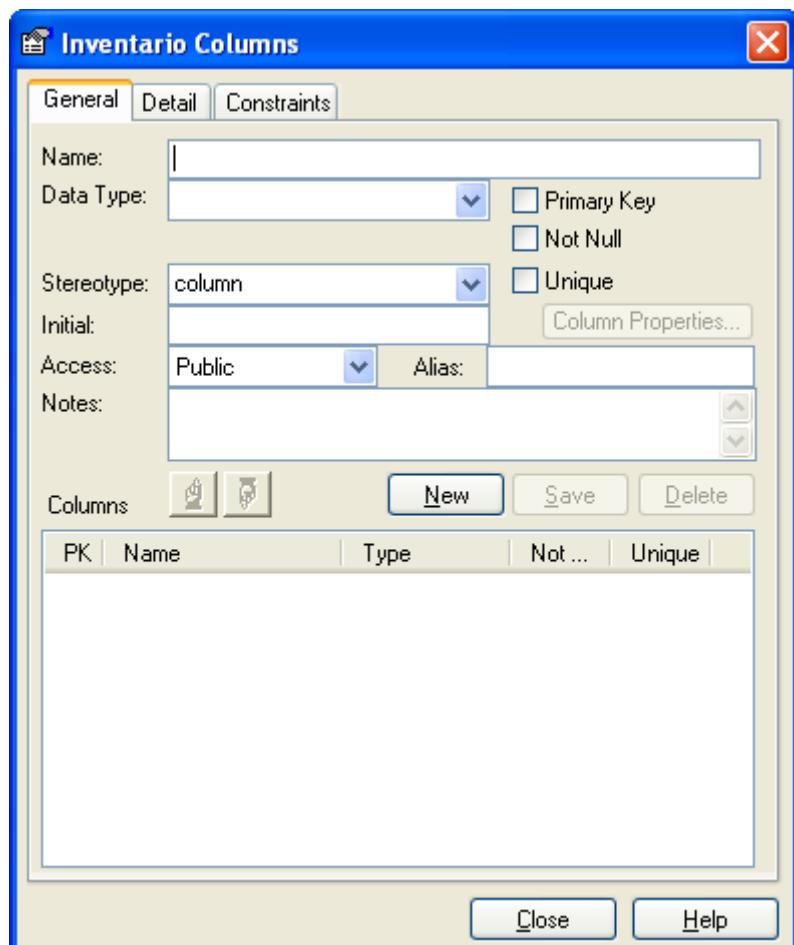


Otra propiedad importante que se ha de asignar a una tabla es la propiedad **Owner**. Esta propiedad se ha de asignar a través de los **Tagged Values**, para ello del menú de la herramienta, se deberá seleccionar **View-> Tagged Values** y seleccionar el objeto al que se le quiere asignar los nuevos valores. A continuación seleccionar el ícono **New Tag** y asignar la propiedad.

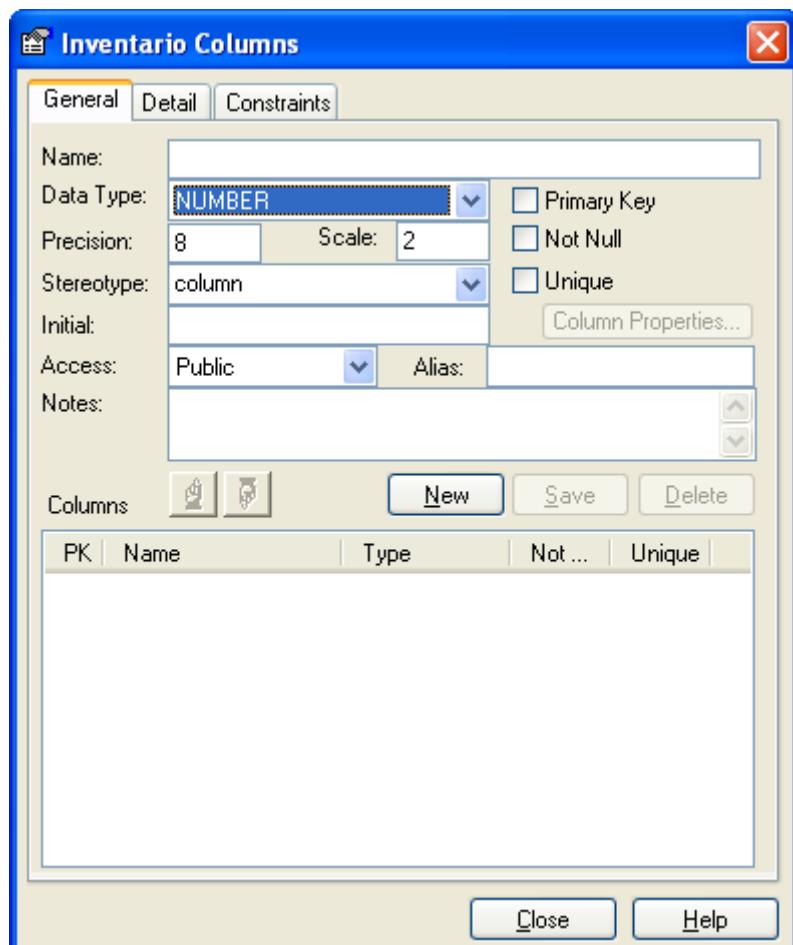


## Crear Columnas

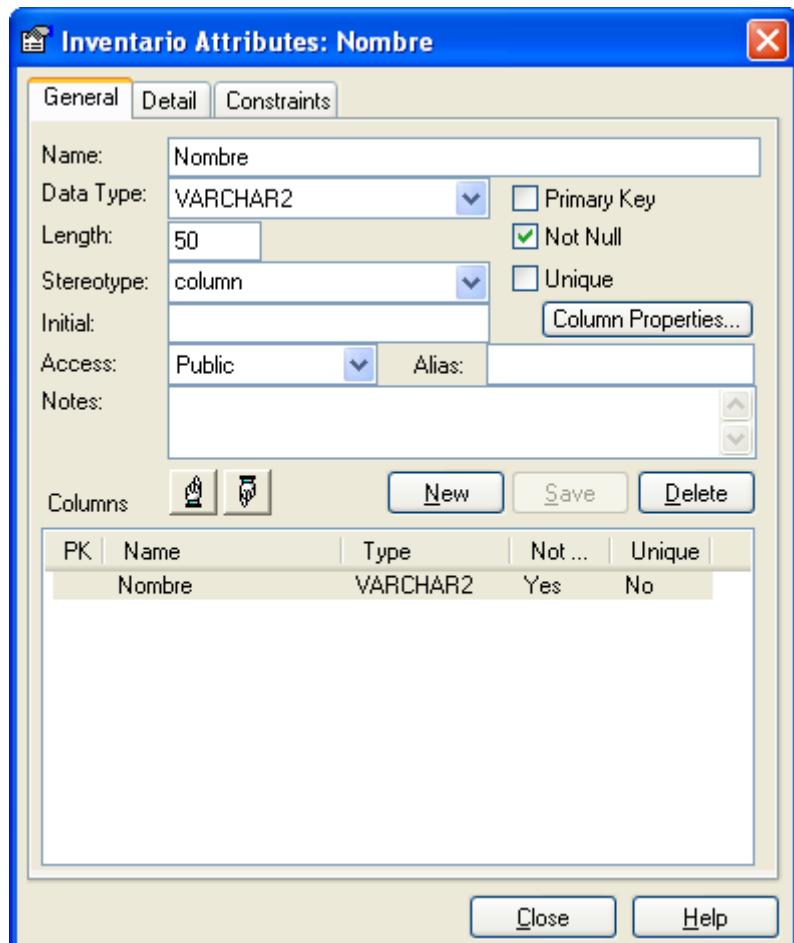
A continuación ya se pueden definir las columnas que van a formar parte de la tabla, para ello se ha de seleccionar la tabla a la que se quieren añadir las columnas y a través del botón derecho del ratón seleccionar la opción **Attributes**.



Si el campo **Data Type** se muestra vacío, quiere decir que no se ha seleccionado en la definición de la tabla qué tipo de base de datos va a pertenecer. Existen algunos tipos de datos, como el **NUMBER** en Oracle, que requieren precisión y escala. Estos campos se muestran por pantalla una vez seleccionado el tipo de dato que lo necesita.



Para crear una nueva columna, bastara con introducir el nombre y el tipo de dato de la misma, y pulsar el botón **Save**



### Clave Primaria

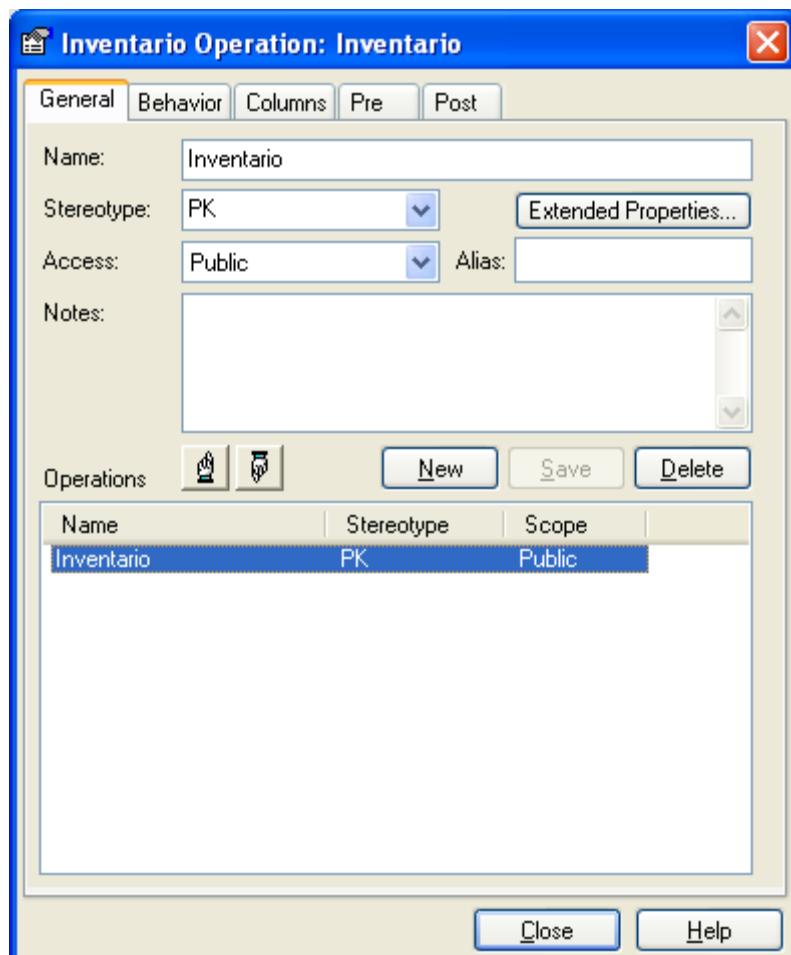
Para la definición de una clave primaria en el caso en que consista en una única columna se ha de marcar la opción **Primary Key**, de la columna correspondiente, en la ventana de gestión de columnas.

Automáticamente, se genera una operación que define la clave primaria para la tabla. Para eliminar la clave primaria definida, bastara con eliminar la operación creada.

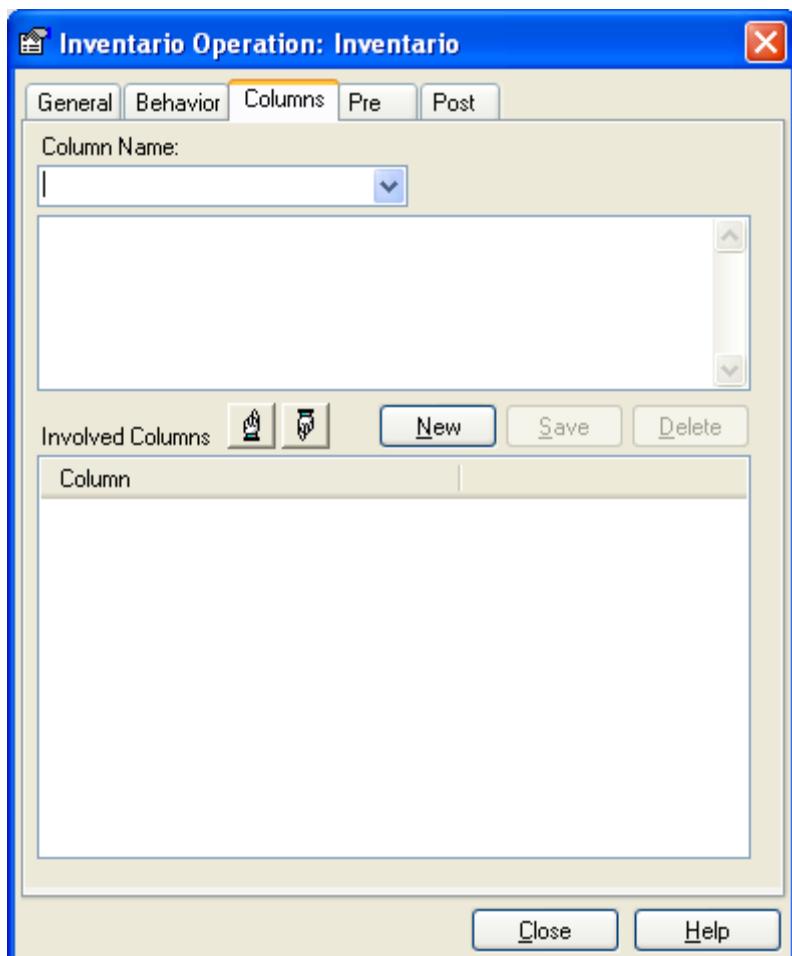


Para definir una clave primaria compleja:

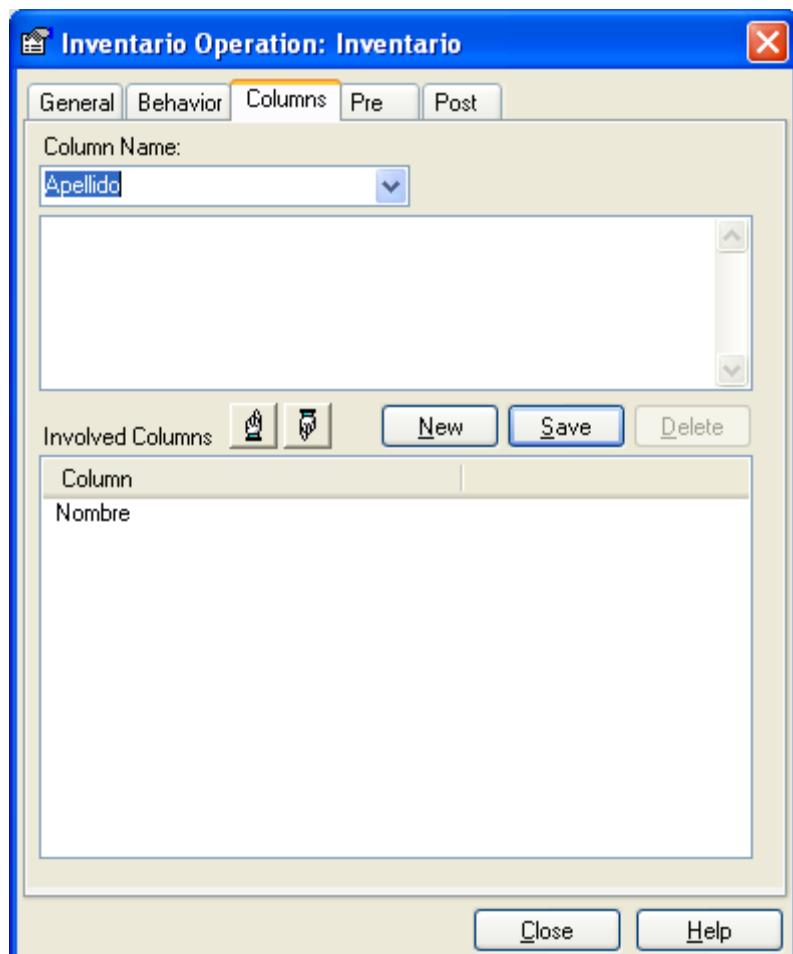
1. Seleccionada la tabla, con el botón derecho del ratón se deberá seleccionar la opción **Operations**.
2. Seleccionar la operación **Primary Key** ("PK\_" en el campo Stereotype) y guardar. Lo que generara una entrada en la lista de operaciones con el nombre introducido en el campo **Name**.



3. Una vez guardado, se ha de seleccionar la operación y activar la pestaña columnas



4. Del campo **Column Name**, se ha de seleccionar cada campo de la tabla que vaya a formar parte de la clave primaria y añadirlo pulsando el botón **Save**.



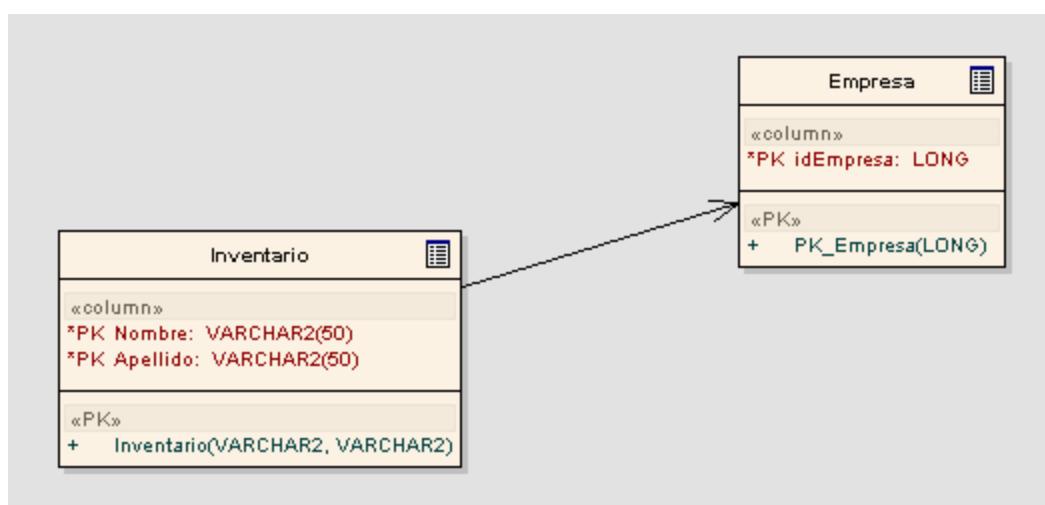
5. Con los botones **Up/Down**, se puede modificar el orden de las columnas en la clave primaria



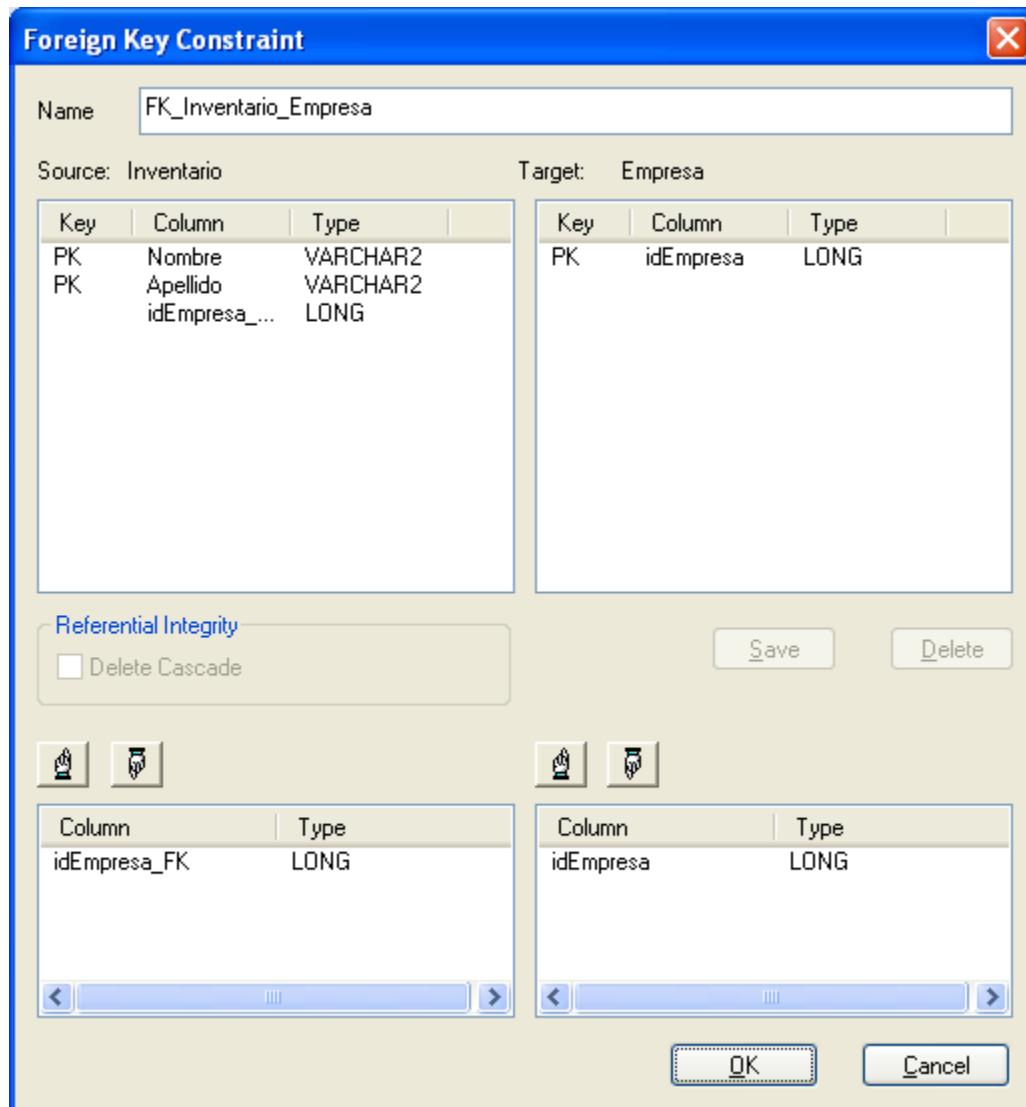
## Claves Externas

Para crear claves externas se han de seguir los siguientes pasos:

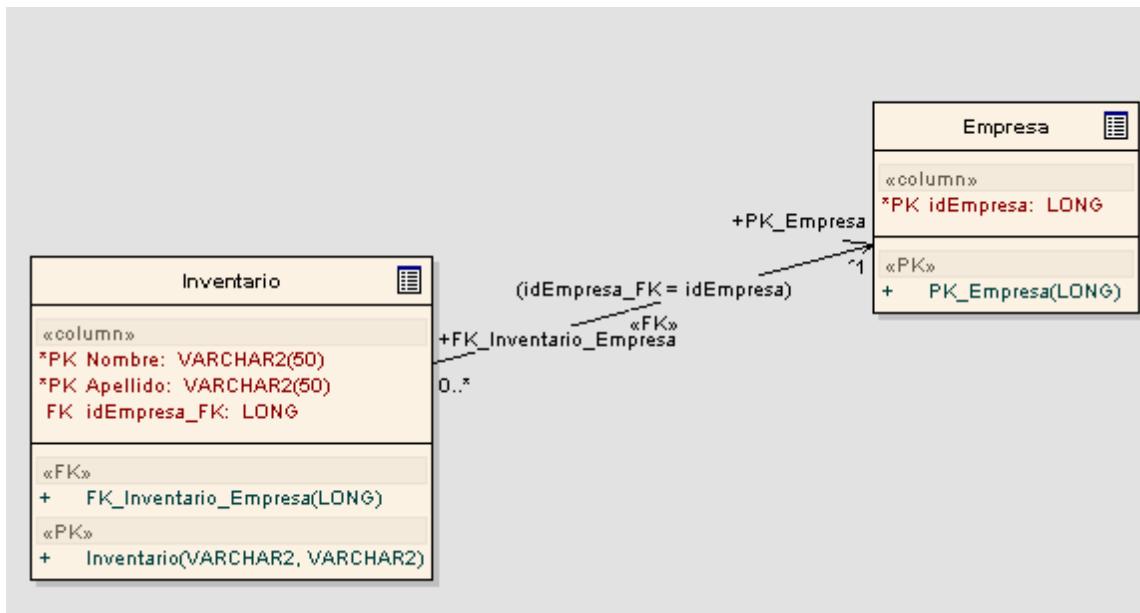
1. La tabla origen deberá tener definidas las columnas que forman la clave primaria de la tabla a la que se va a conectar.
2. De la caja de herramientas seleccionar el conector **Associate**
3. Seleccionar la tabla que contendrá la clave externa y conectarla con la tabla destino.



4. Seleccionar el conector que une ambas tablas y del menú del botón derecho del ratón seleccionar la opción **Foreign Keys**
5. Seleccionar las columnas que formaran parte de la relación en la clave externa y pulsar el botón **Save**. Deberá existir en al tabla en la que se creara la clave externa un campo con el que poder asociar la clave primaria de la otra tabla.

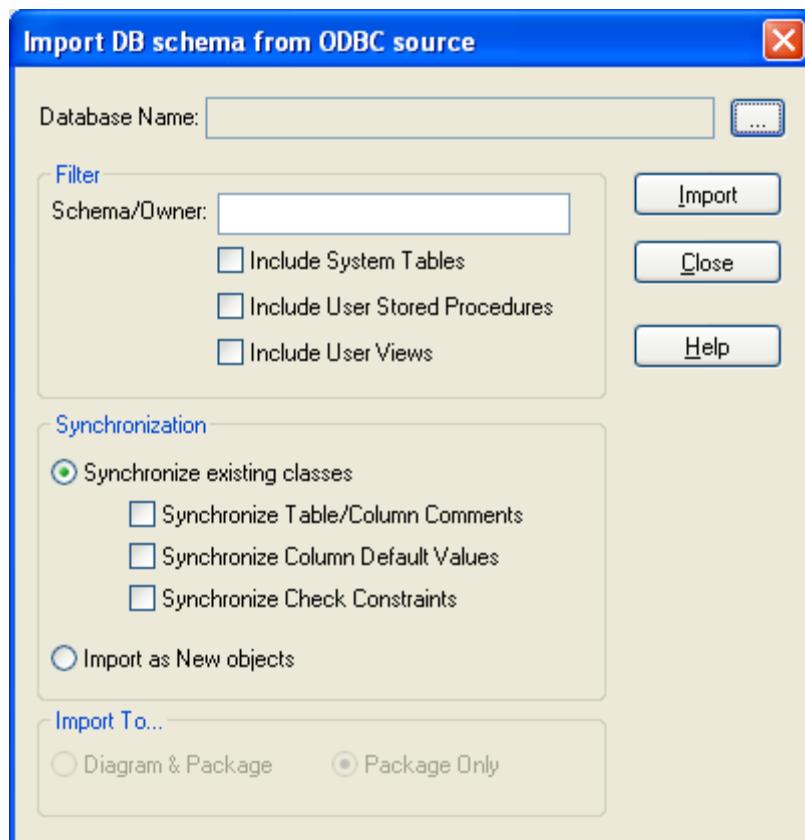


7. Se generaran las entradas de la clave externa y finalmente bastara con pulsar el botón **OK**.

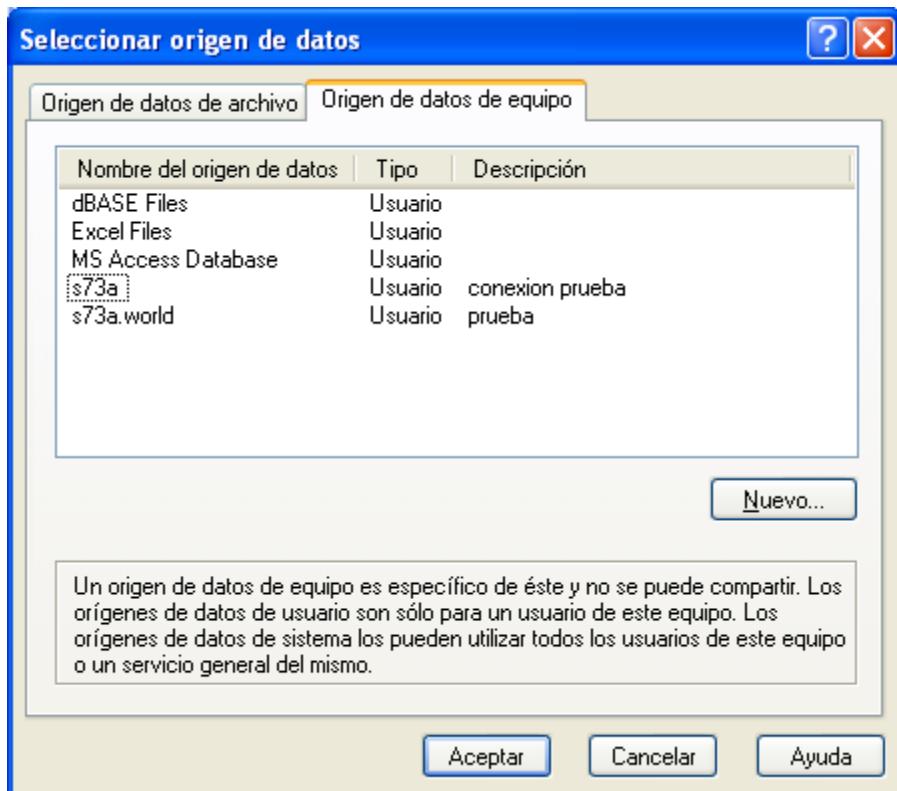


### 3.8.2. Importar desde BD

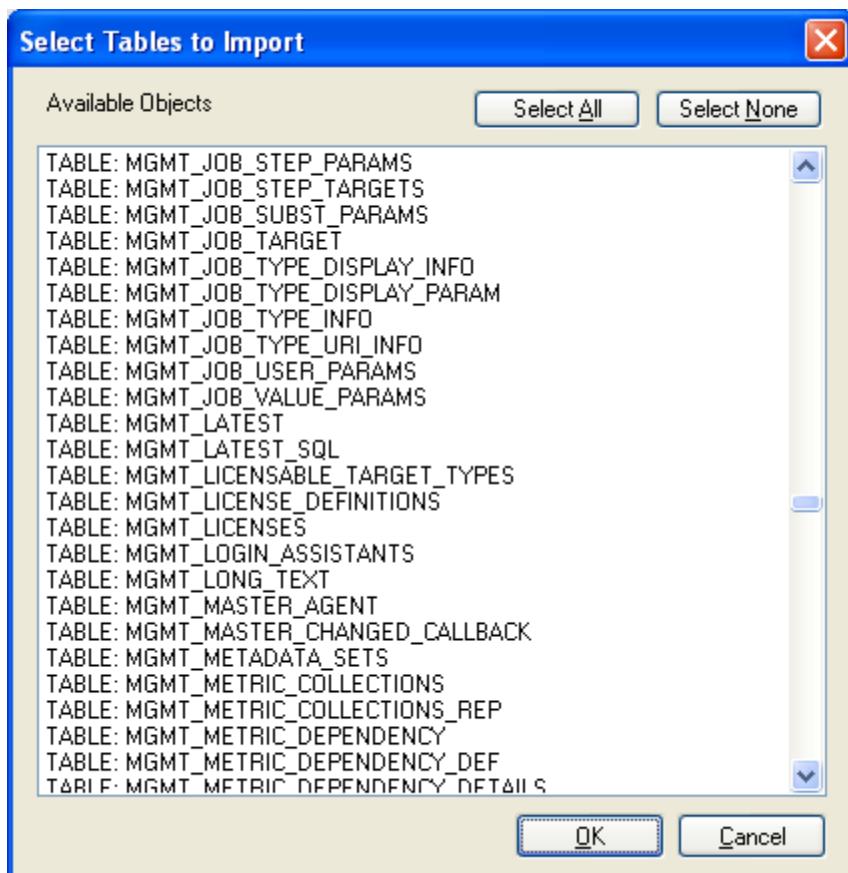
Para la importación de tablas y procedimientos almacenados de base de datos vía ODBC se ha de seleccionar el paquete en el que se realizará la importación. A través del botón derecho del ratón y seleccionando del menú emergente la opción **Code Engineering -> Import DB schema from ODBC**, se mostrará un cuadro de diálogo que nos permitirá establecer la conexión a base de datos.



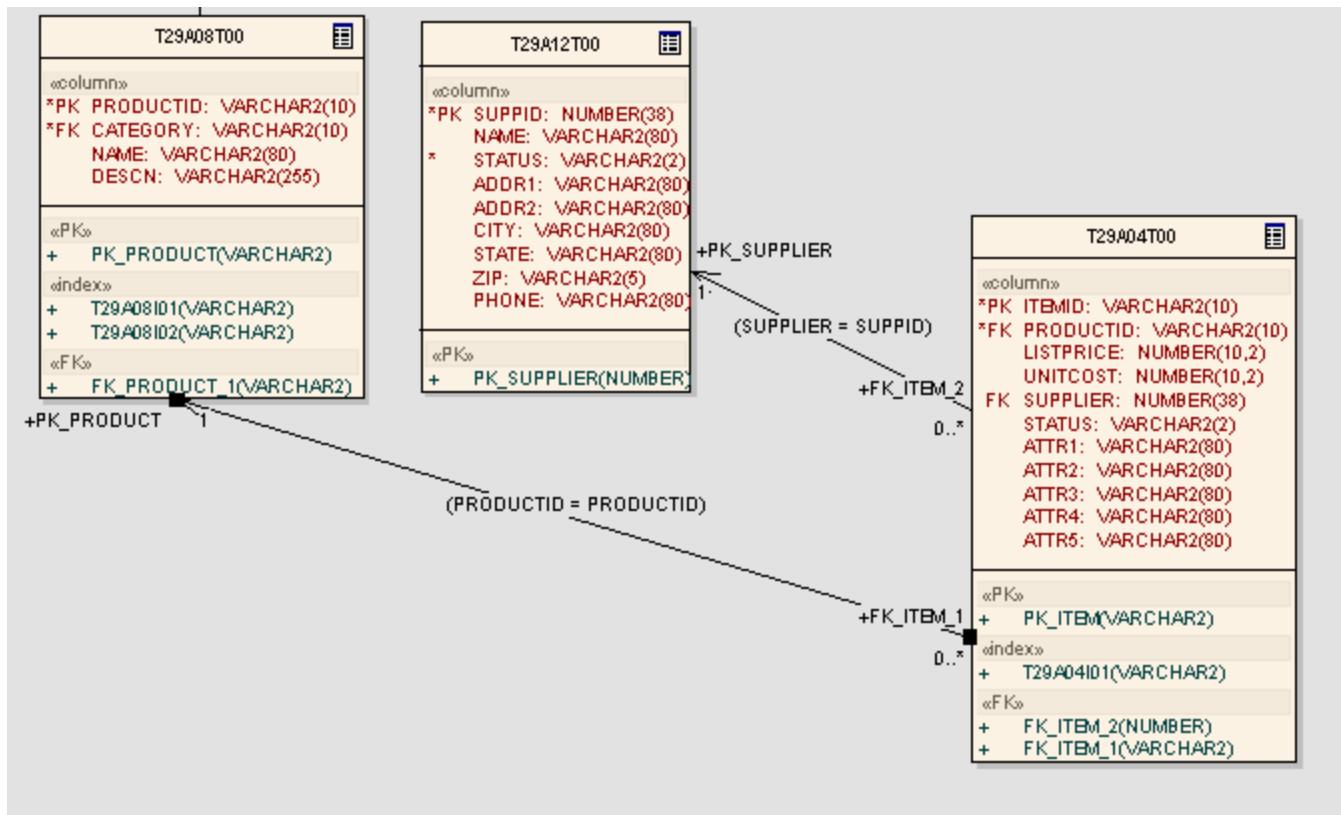
Se deberá seleccionar el **datasource** a utilizar para la importación, por lo que se deberá tener configurado con anterioridad un **datasource** en el sistema.



Una vez seleccionado el **datasource**, a continuación se presentara una relación de tablas y procedimientos almacenados presentes en la base de datos seleccionada. De la lista que se presenta se deberán seleccionar aquellos elementos que se vayan a importar al diagrama.



Una vez seleccionadas las tablas y los procedimientos almacenados a importar en el diagrama, bastara con pulsar el botón **OK**, y Enterprise Architect se encargara de cargar en el diagrama las tablas con sus campos y relaciones correspondientes. En el caso de los procedimientos almacenados se mostrara una clase con el nombre de la base de datos importada y cada operación de la clase se corresponderá con cada procedimiento almacenado importado



## 4 Anexo 1 : Diagrama de Caso de Uso

El presente ejercicio consta de dos partes:

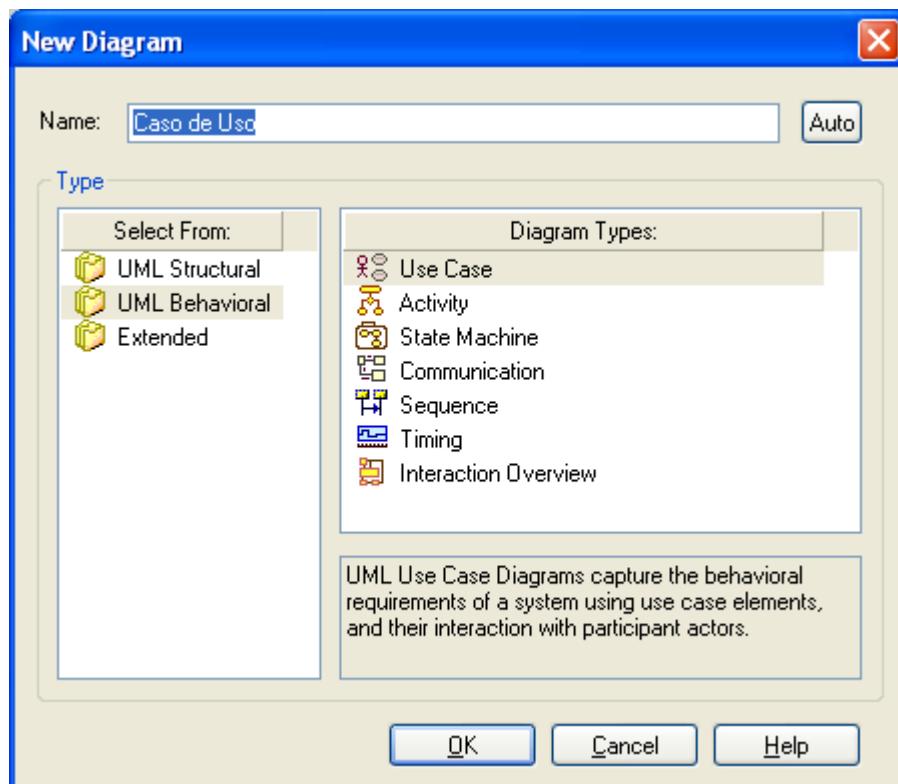
- Creación de un diagrama con los siguientes elementos:
  - Administrador-> Persona encargada del mantenimiento de los usuarios en el sistema.
  - Alta de Usuarios -> Caso de uso que implementa la operación de alta usuarios en el sistema.
  - Baja de Usuarios -> Caso de uso que implementa la operación de baja de usuarios en el sistema.
- Diseño de una plantilla para la generación de la documentación correspondiente de los casos de uso. En el documento final se deberá presentar la siguiente información:
  - El nombre del paquete como título.
  - Cada uno de los elementos que forman parte del diagrama con su correspondiente descripción
  - Imagen del diagrama.

### 4.1 Resolución

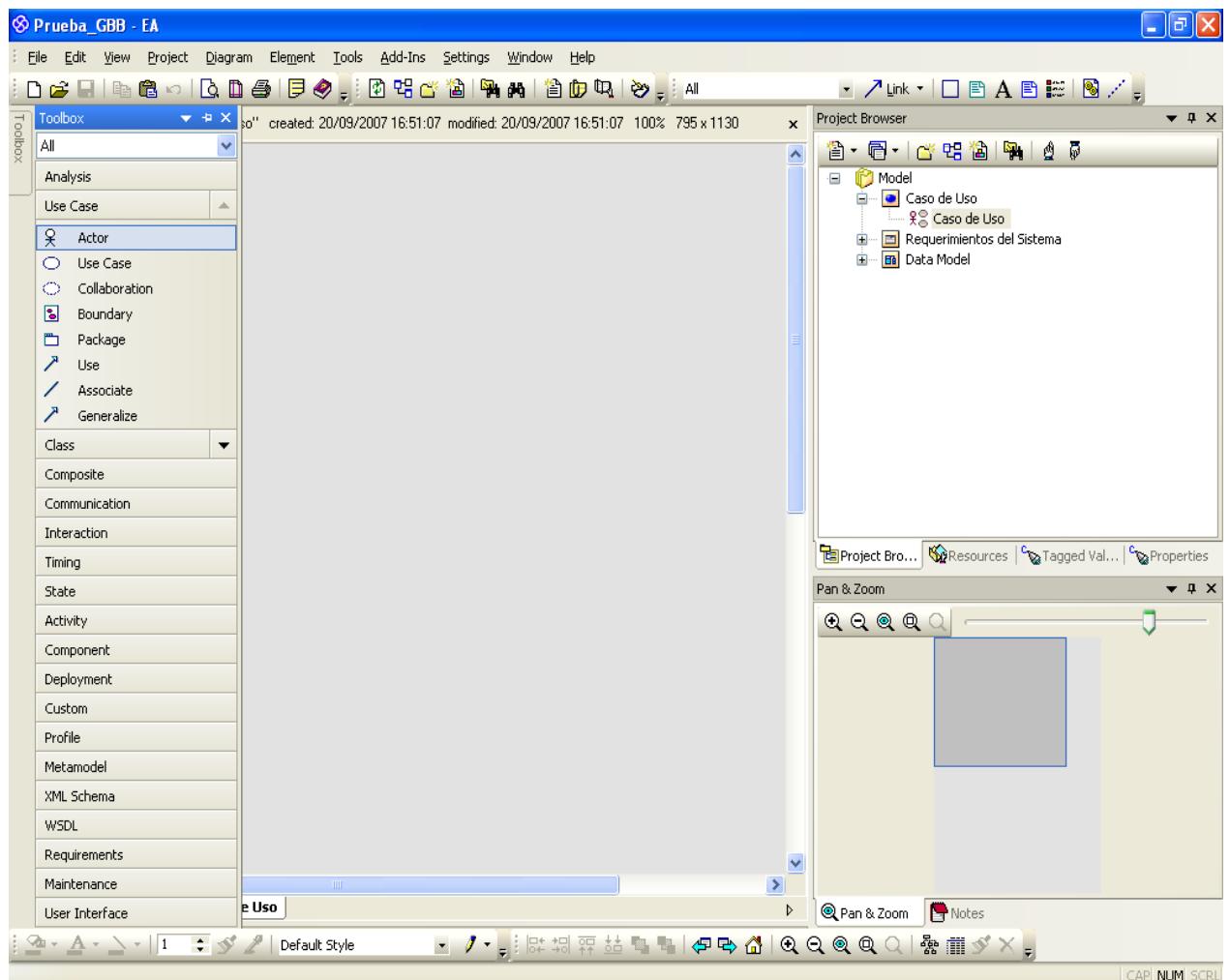
Para la creación de un diagrama de caso de uso, el primer paso consiste en la creación del paquete donde ubicar el diagrama.

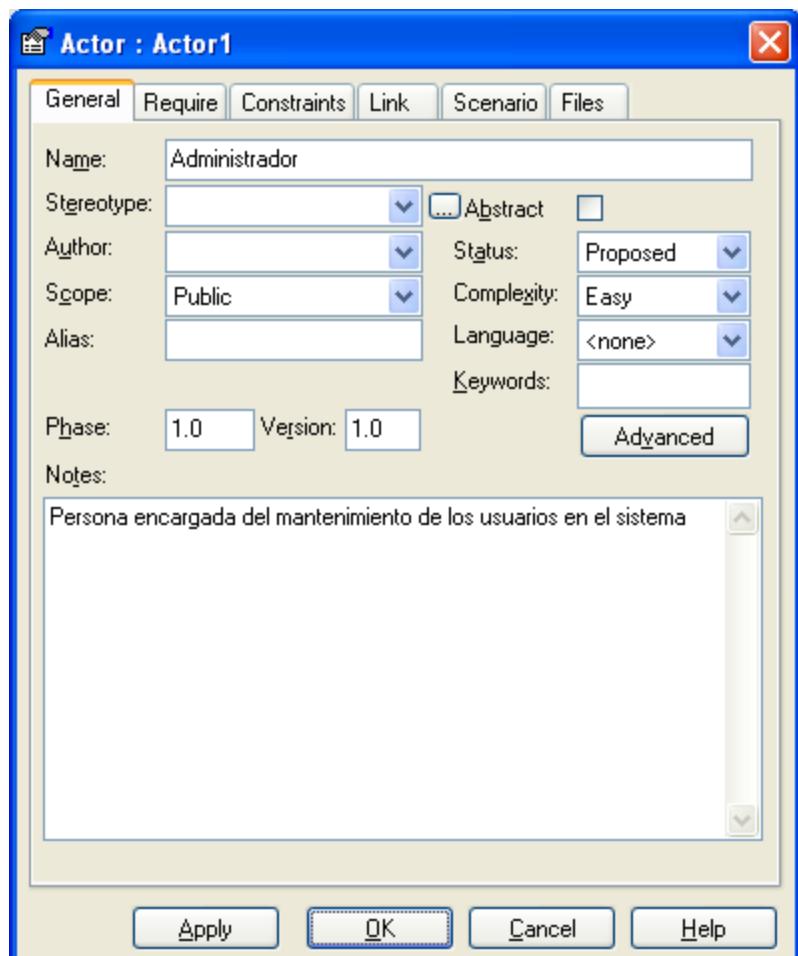


Una vez creado el paquete el siguiente paso es la creación de un diagrama de caso de uso en el que implementar el diseño.

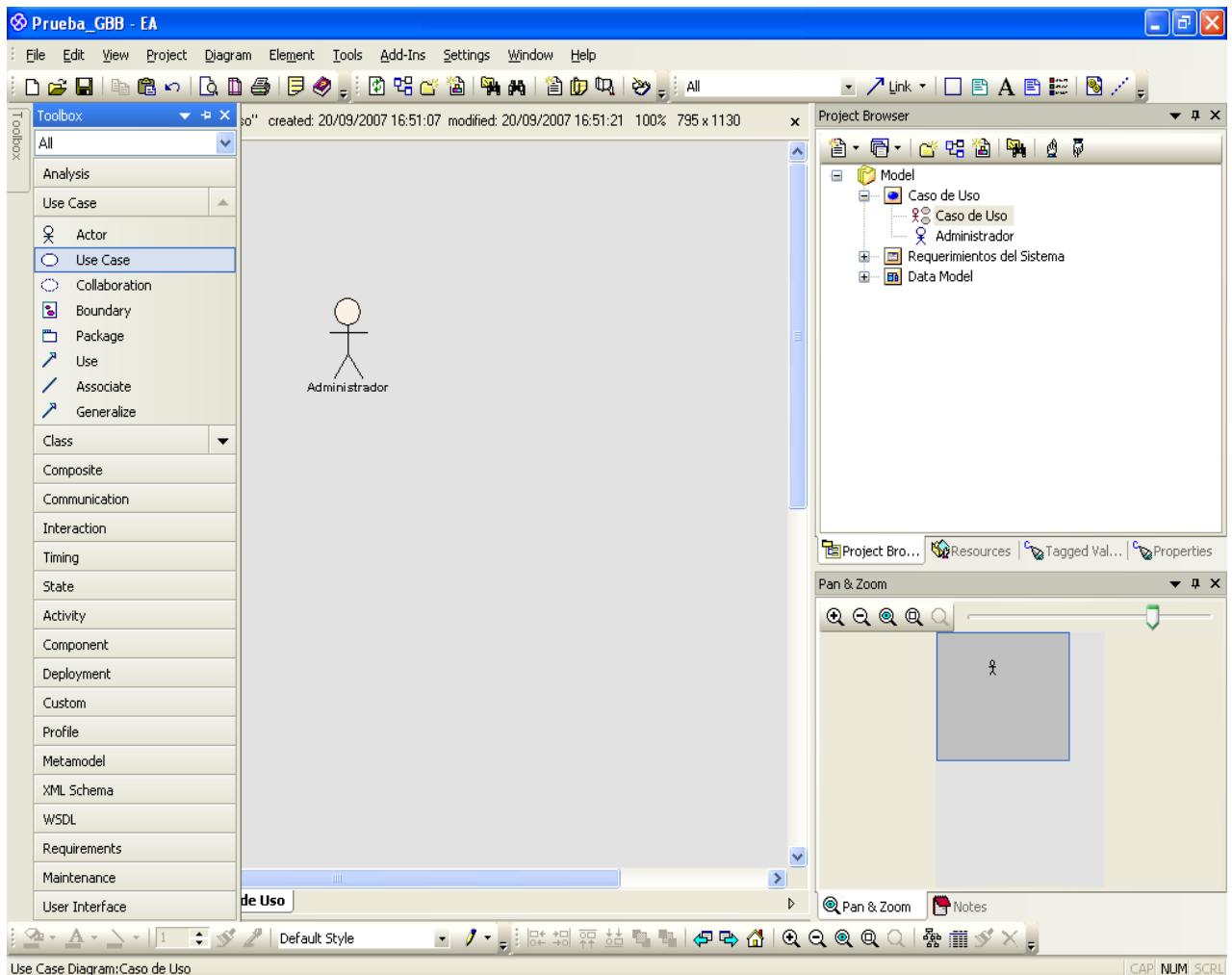


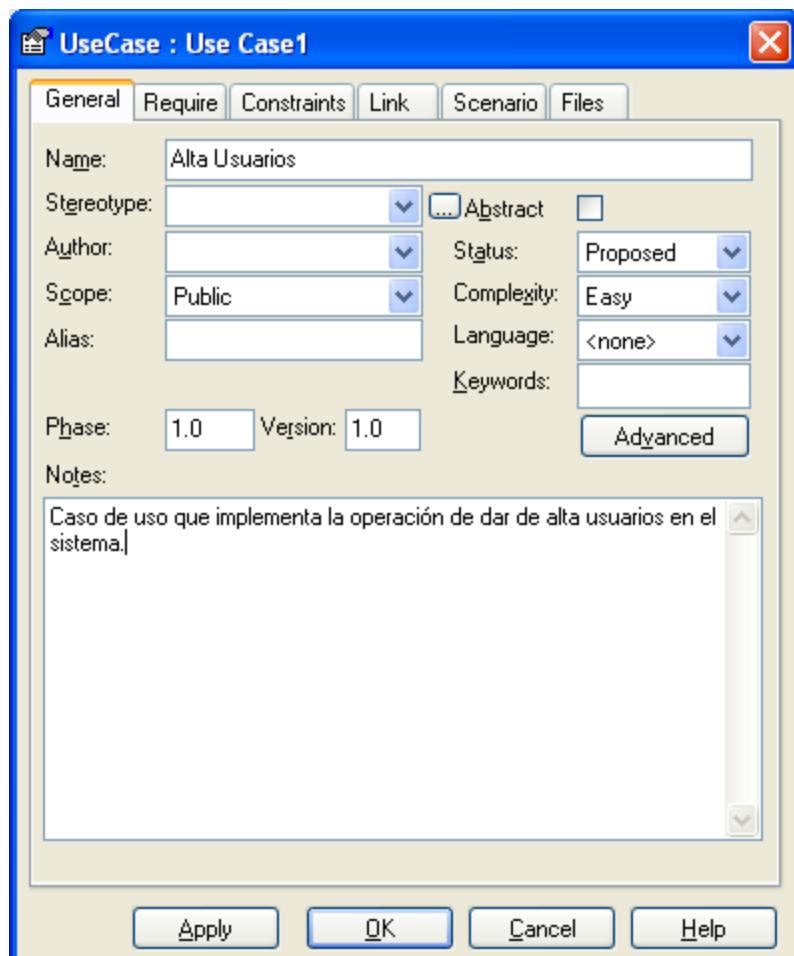
Creado el diagrama, se van incorporando los objetos que intervienen en el mismo, inicialmente se agrega al diagrama el actor.



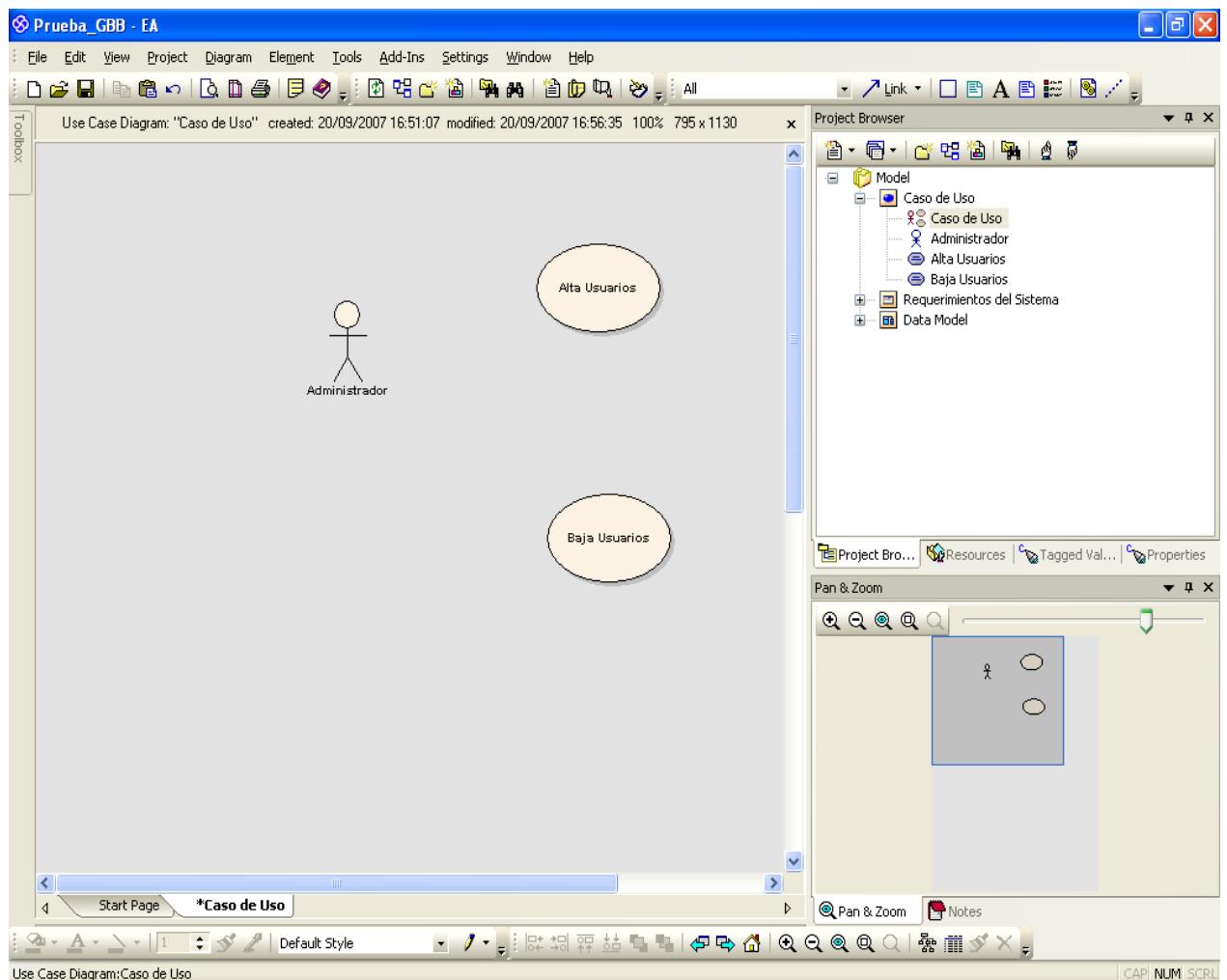


A continuación se agrega el premier caso de uso

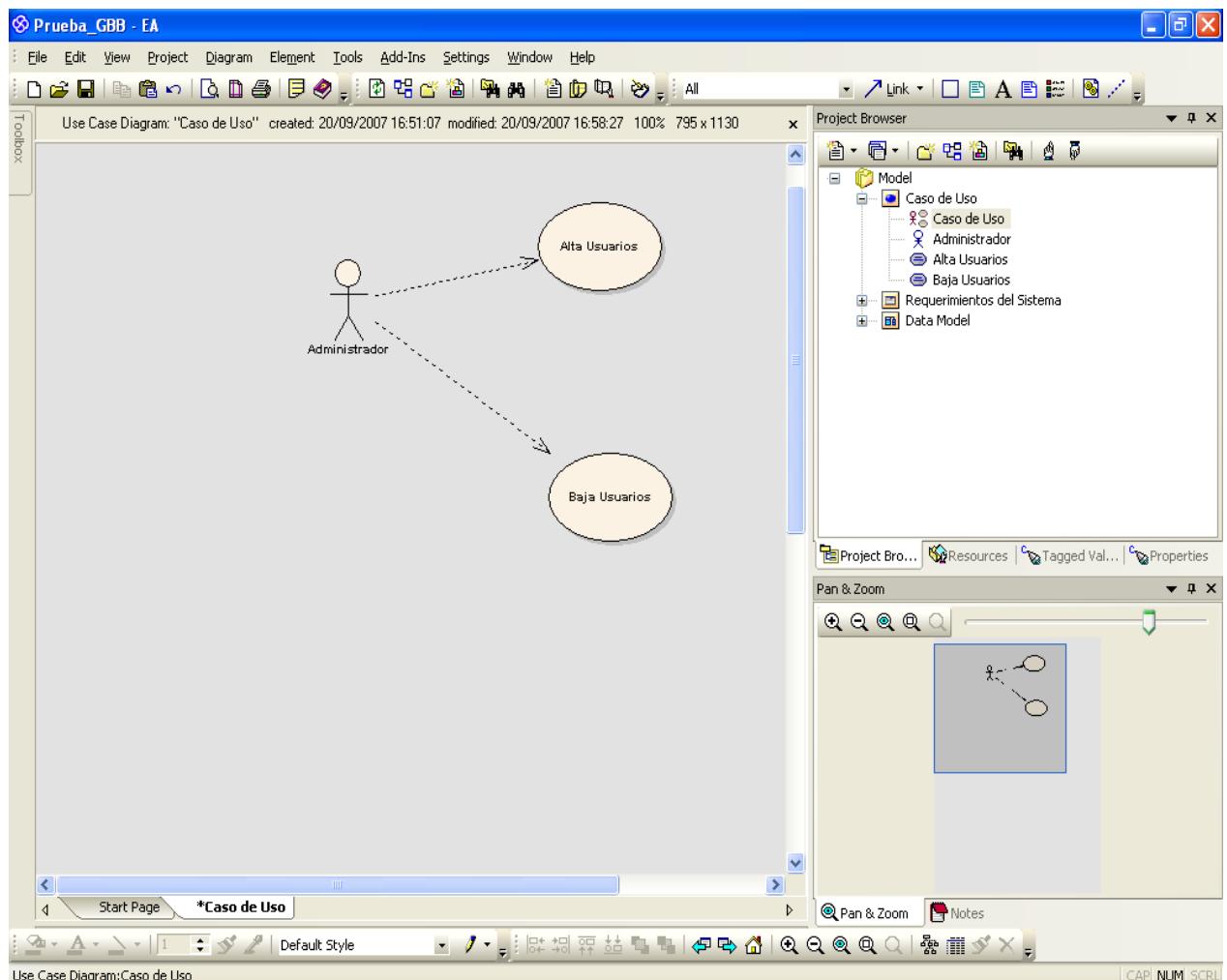




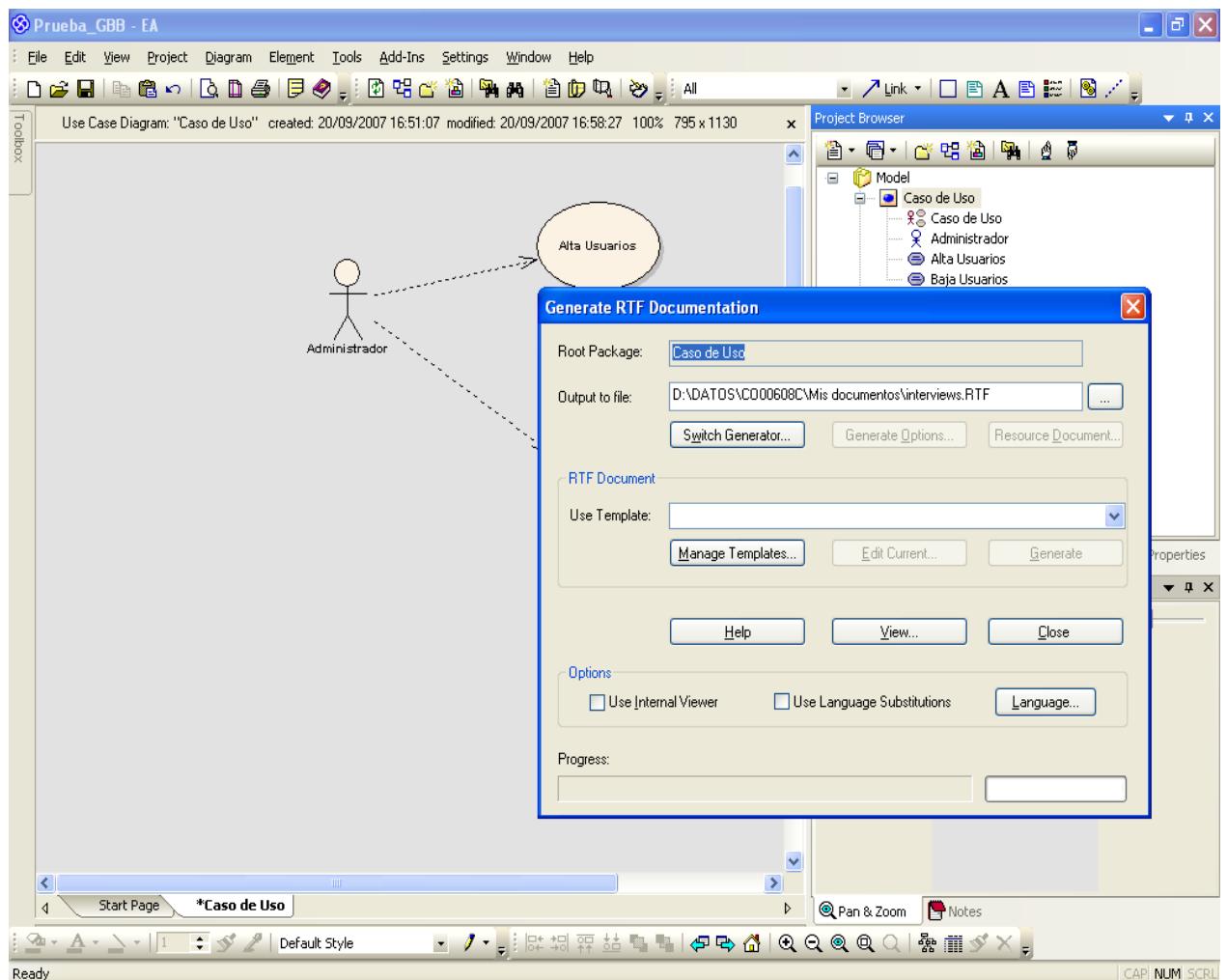
Para el siguiente caso de uso, se ha de repetir el mismo proceso, disponiendo finalmente de un diagrama con el siguiente aspecto



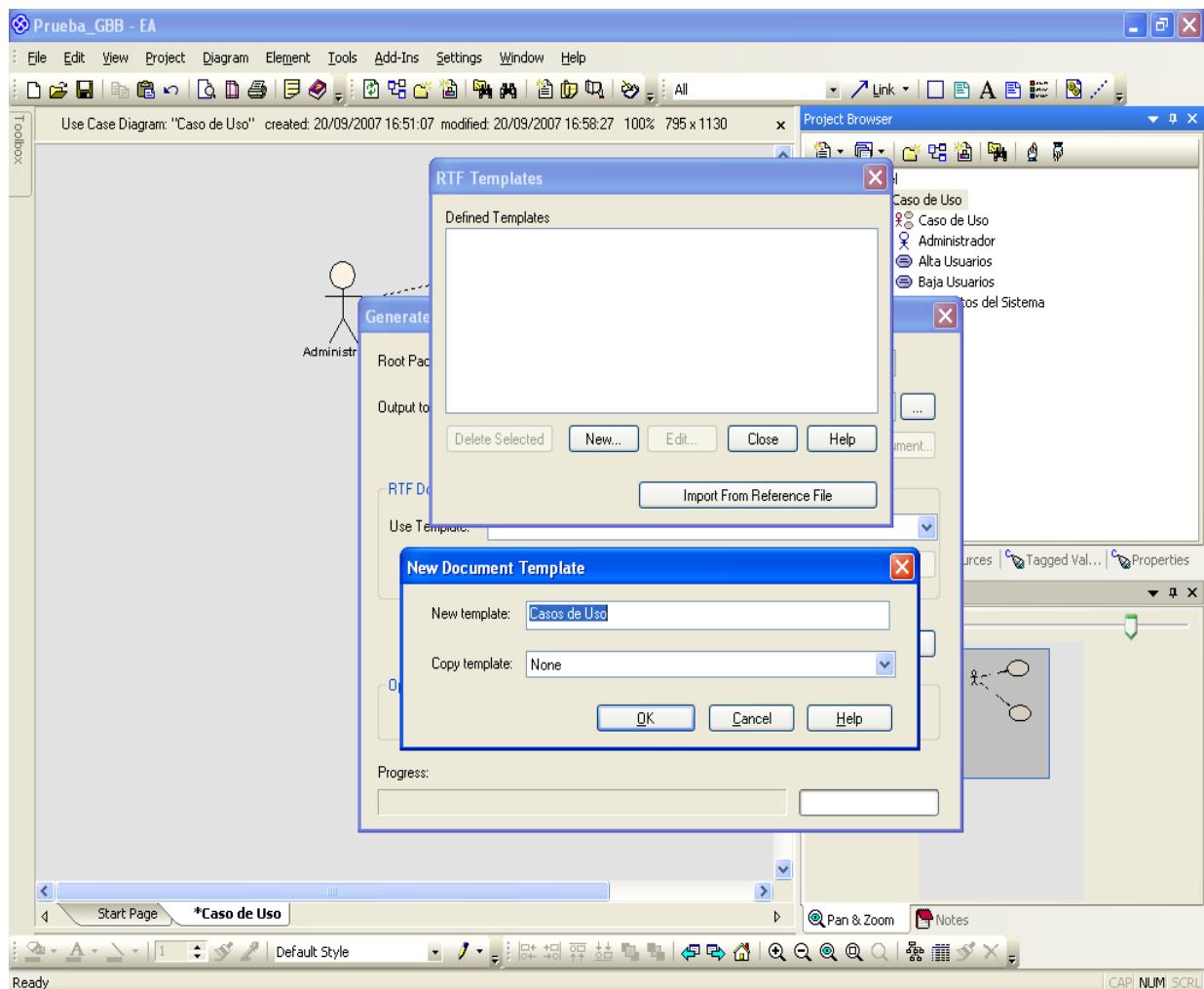
Finalmente queda por asociar los casos de uso al actor



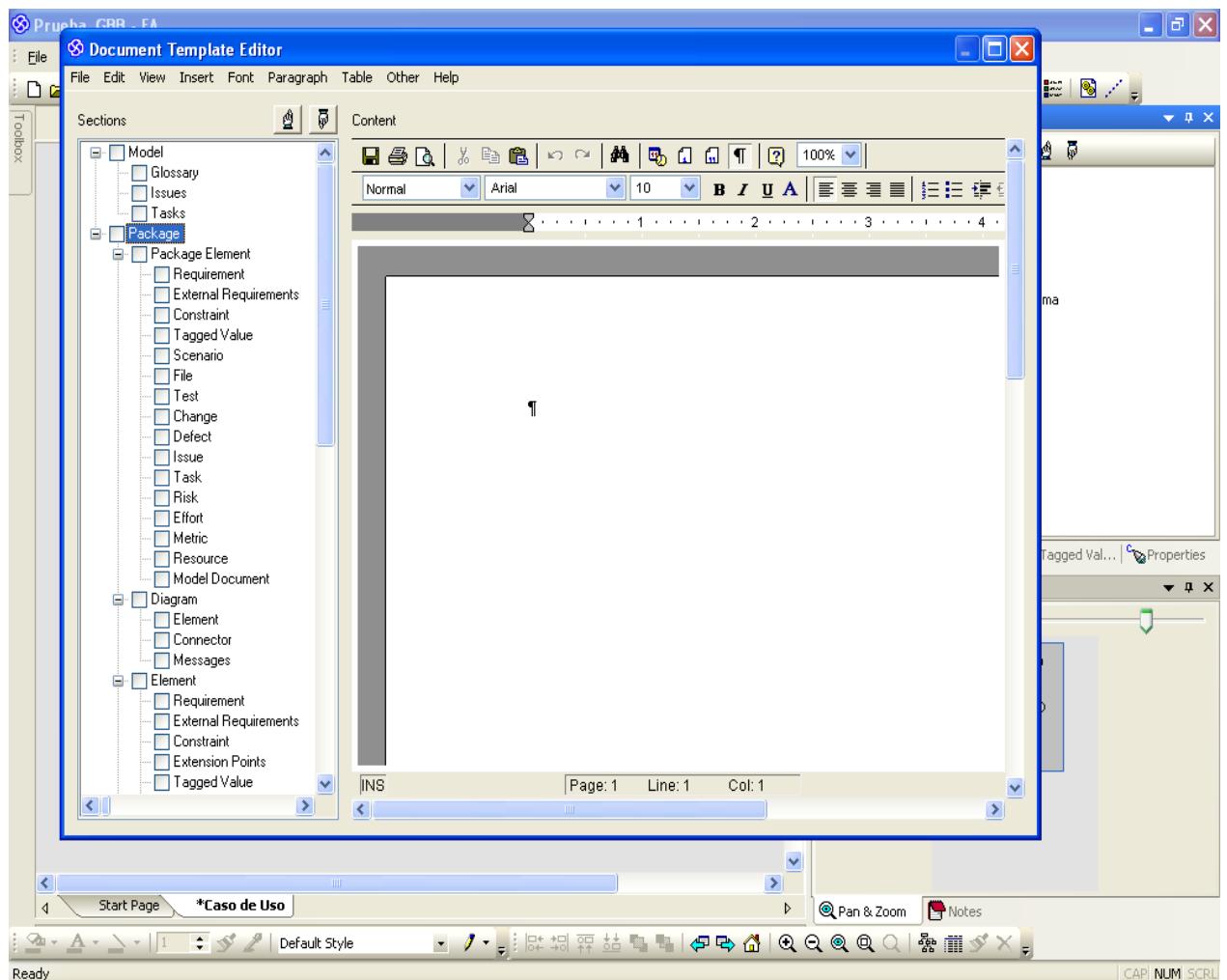
Llegados a este punto, se dispone de un diagrama en el que los elementos se encuentran informados, por lo que se procede al generación de la plantilla para explotar los datos. Para ello se abre el generador de documentación RTF



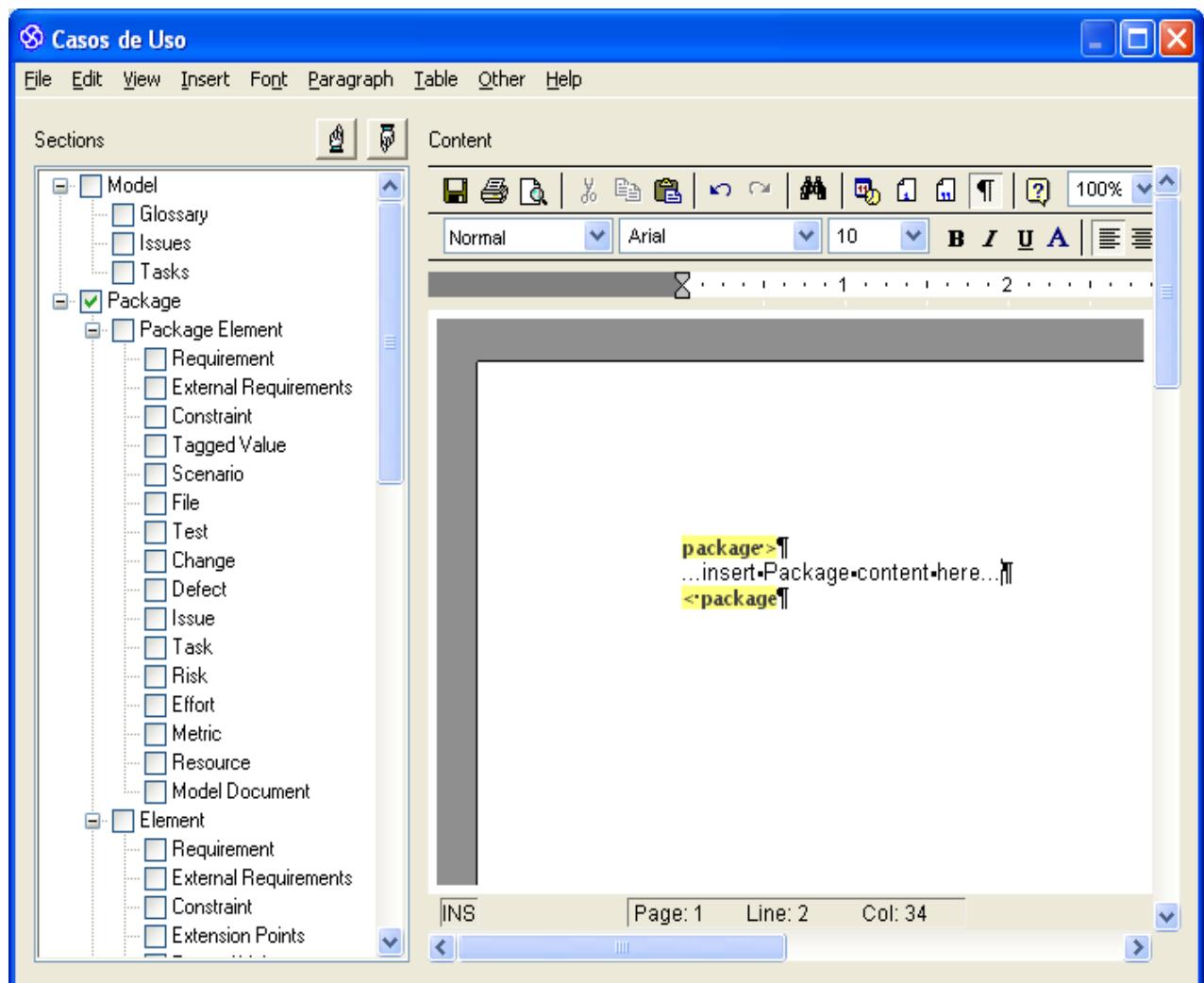
A continuación, se accede al mantenimiento de plantillas y se da de alta una nueva:



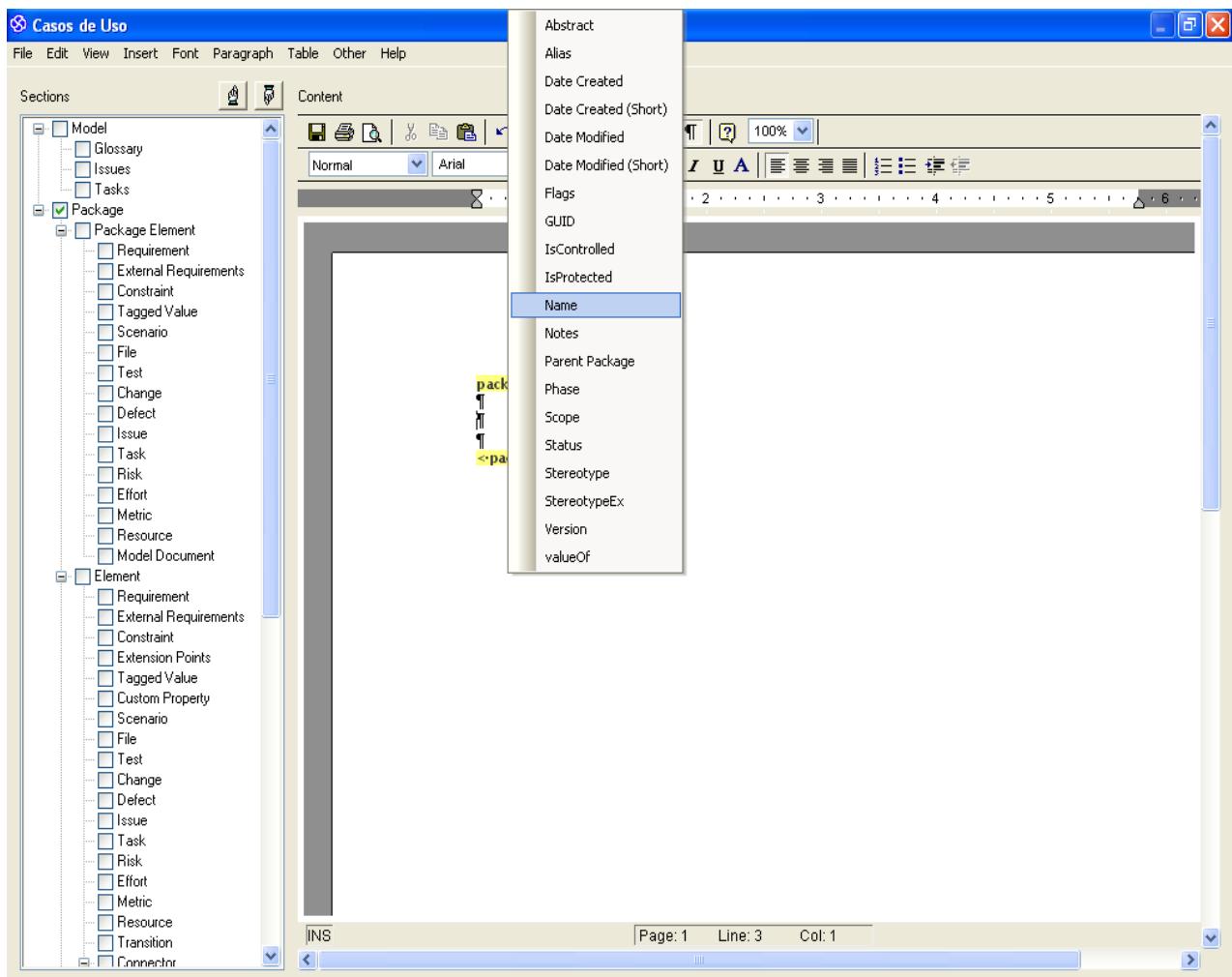
Una vez creada la plantilla, automáticamente se abrirá el editor de plantillas vacío



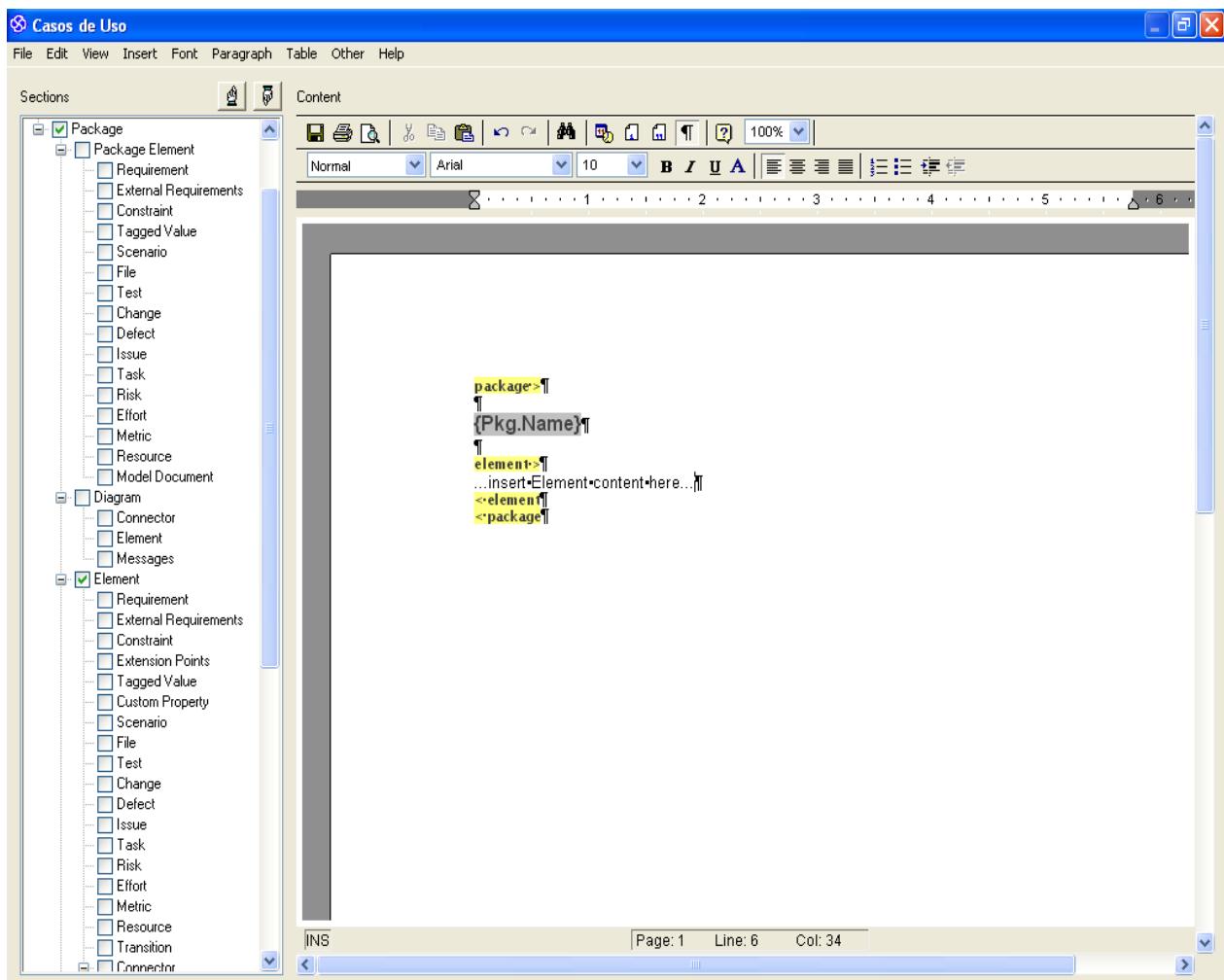
Para agregar el nombre del paquete , se deberá seleccionar del apartado **Sections** el check Package,



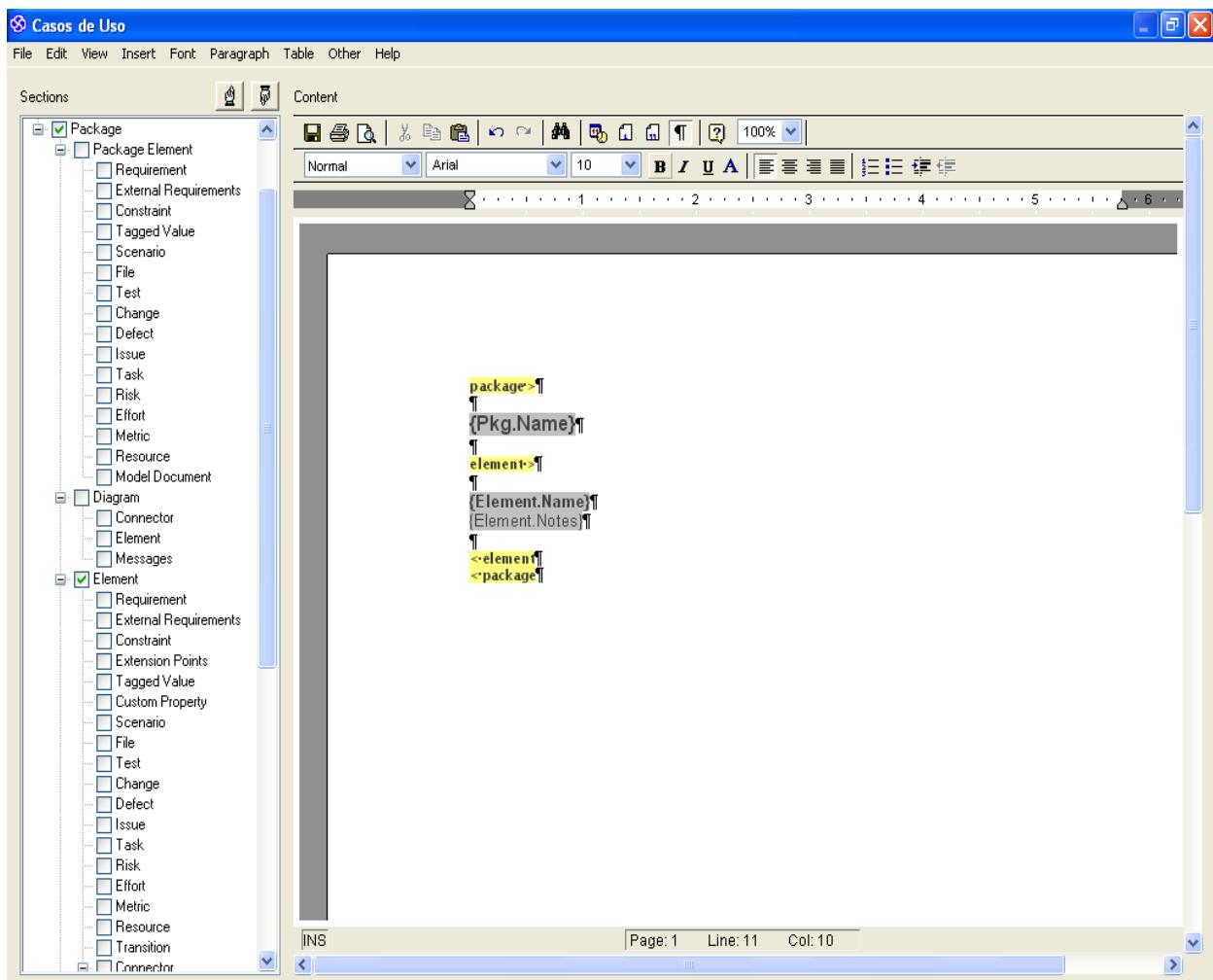
Para obtener el nombre del paquete, en la sección delimitado por los tags seleccionar la propiedad **Name** que se muestra con el botón derecho del ratón:



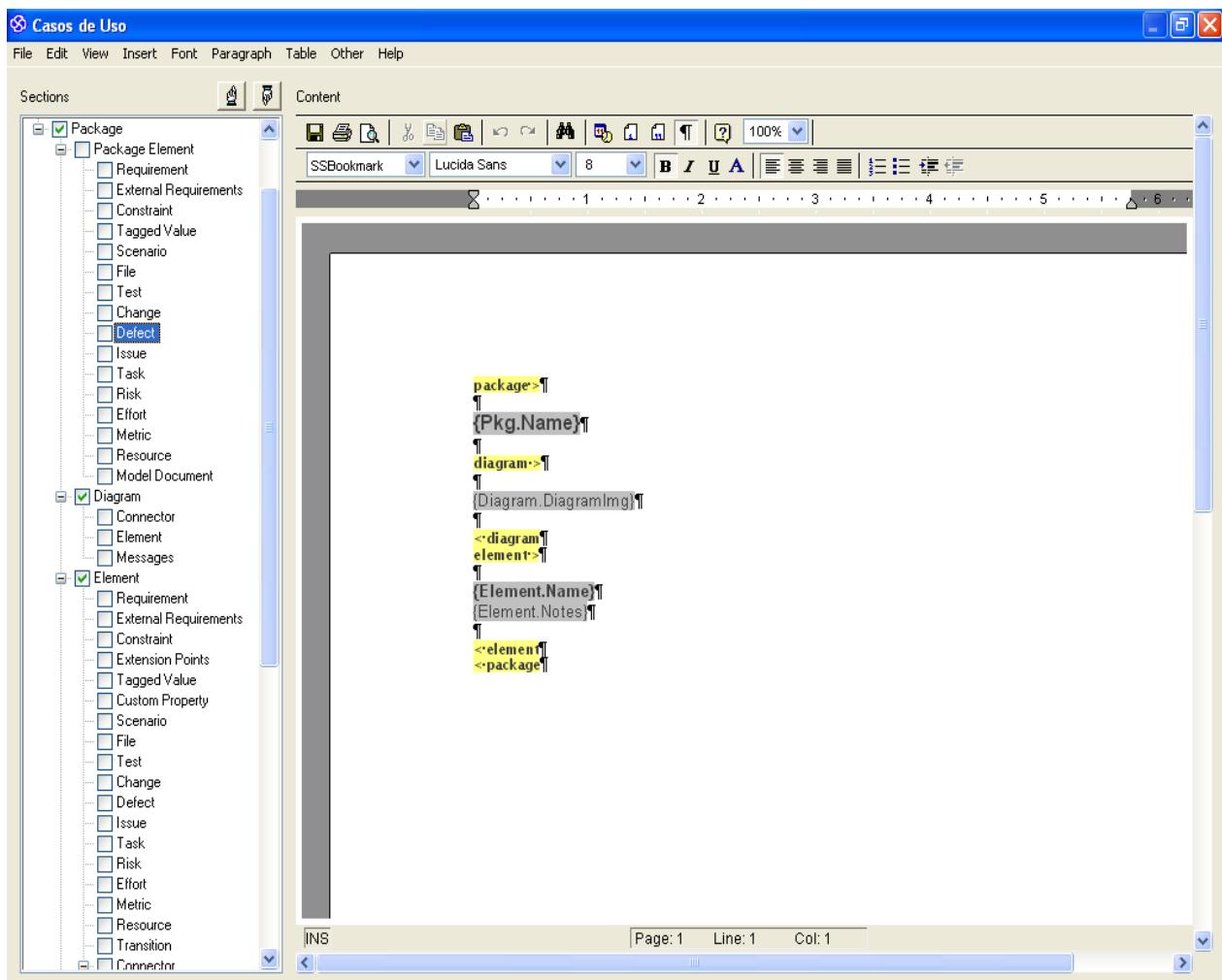
Para agregar los elementos que forman parte del diagrama, se ha de activar el check **Element**



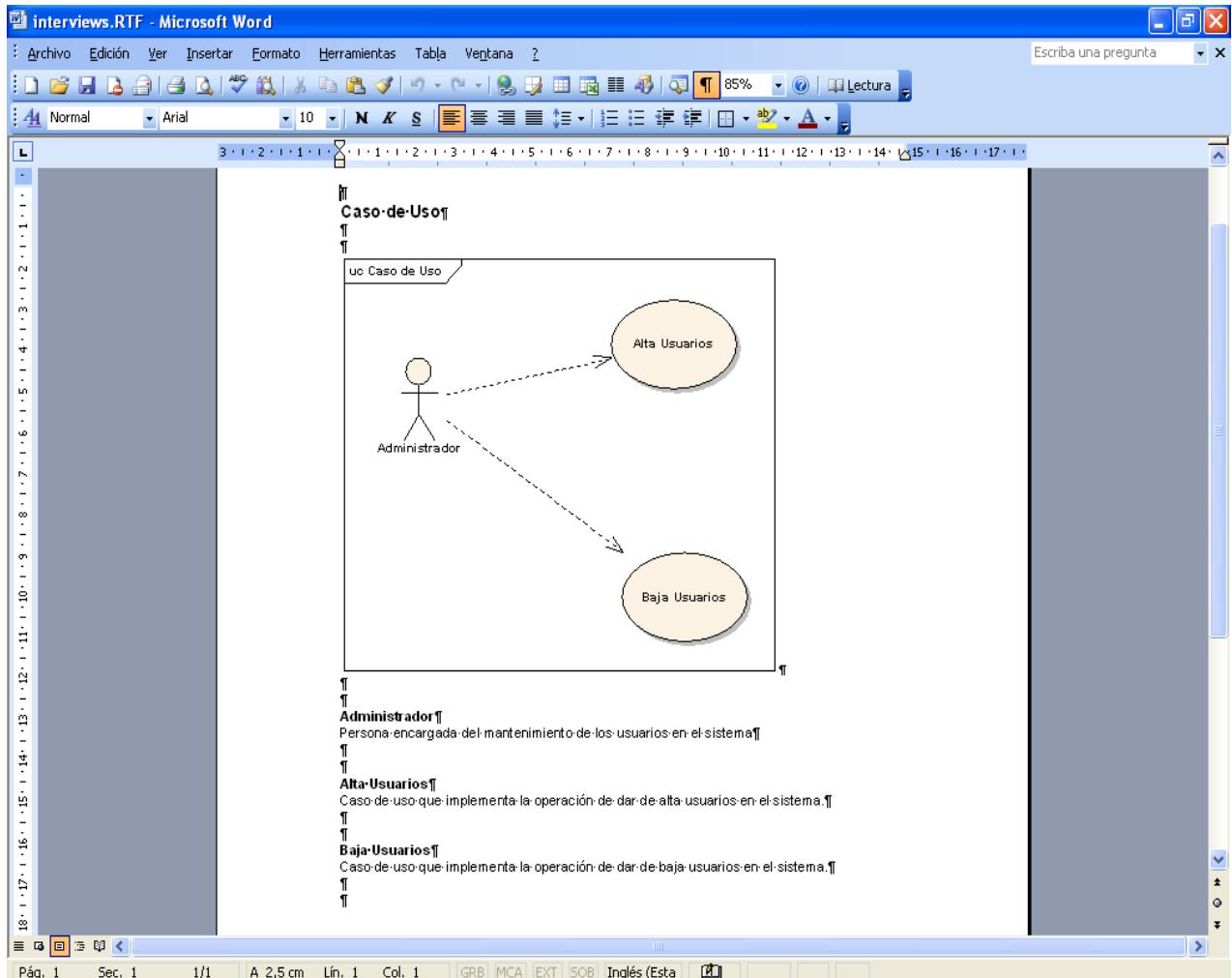
A continuación se han de agregar las propiedades **Name** y **Notes**.



Y finalmente para que se muestre la imagen del diagrama, se deberá activar el check de **Diagram** y agregar la propiedad **DiagramImg**, de tal forma que la plantilla definitiva sea la que se muestra a continuación



El documento final generado presentara el siguiente aspecto:



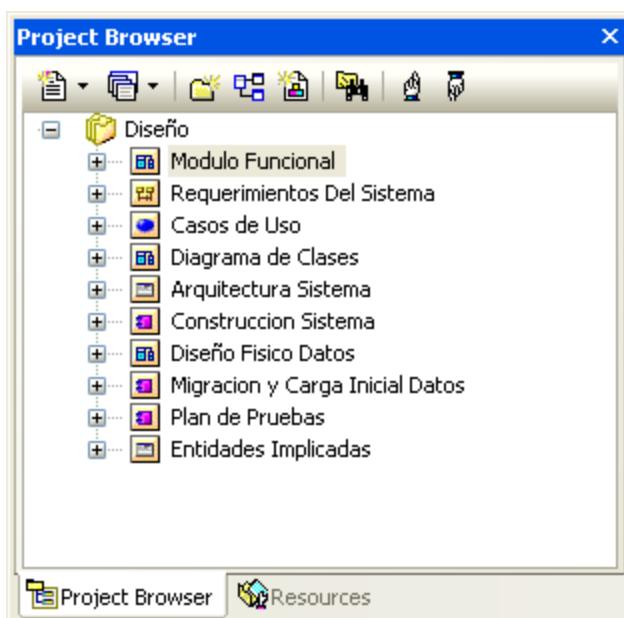
## 5 Enterprise Architect VS Arin Bide

En el siguiente apartado se va a definir el proceso de creación de la documentación correspondiente a Arin Bide a partir de la herramienta Enterprise Architect.

Para ello se proporcionará un esqueleto de proyecto, que se ha de llenar con los datos específicos de cada proyecto, así mismo se proporcionan las plantillas necesarias para la generación de la documentación final.

### 5.1 Arquitectura Proyecto

La arquitectura que todo proyecto de EA presentará en su fase de creación es la siguiente:



Con dicha estructura se cubre la creación de los distintos documentos que completan la documentación de Arin Bide.

**NOTA: La sección Entidades Implicadas recoge todas las clases de la aplicación que intervienen en el desarrollo con sus métodos y atributos.**

### 5.2 Documentación

Dentro del proyecto "Oficina Técnica de Calidad y Pruebas" iniciado por el Área de Asistencia Técnica, y concretamente por el equipo de Consultorías de Áreas del Conocimiento, se han definido un conjunto de herramientas que dan soporte al modelo de calidad de software que ha definido el propio proyecto. Una de estas herramientas es Enterprise Architect (en adelante EA).

Con objeto de facilitar su uso, se ha creado un proyecto de EA de ejemplo, que permite generar la documentación de la metodología ArinBide a medida que se realiza el análisis y diseño del sistema.

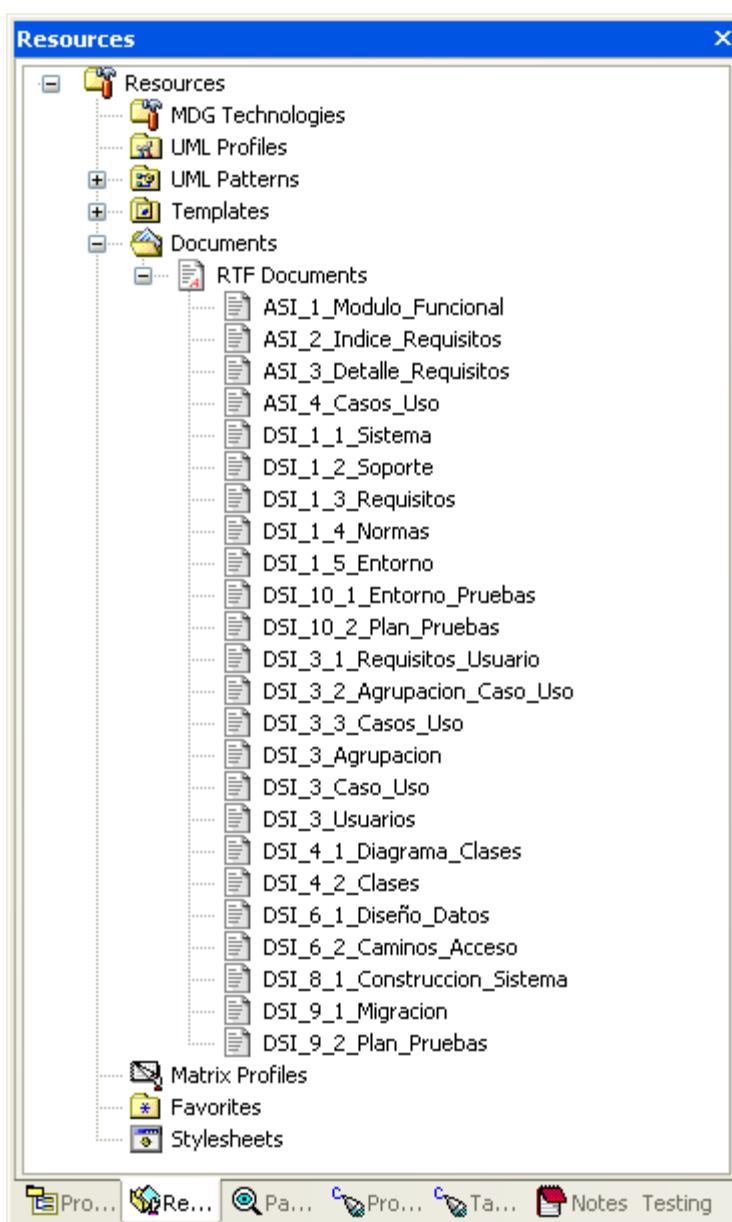
Si se opta por ese ejemplo, puede consultarse el documento **OTC-Herramientas-Guía de uso del ejemplo de**

**EA.doc** Puede descargarse desde <https://ataria.ejie.eus/docs/DocView.aspx?DocumentID={a41be76e-048f-4f58-a85a-d705082f1a65}>

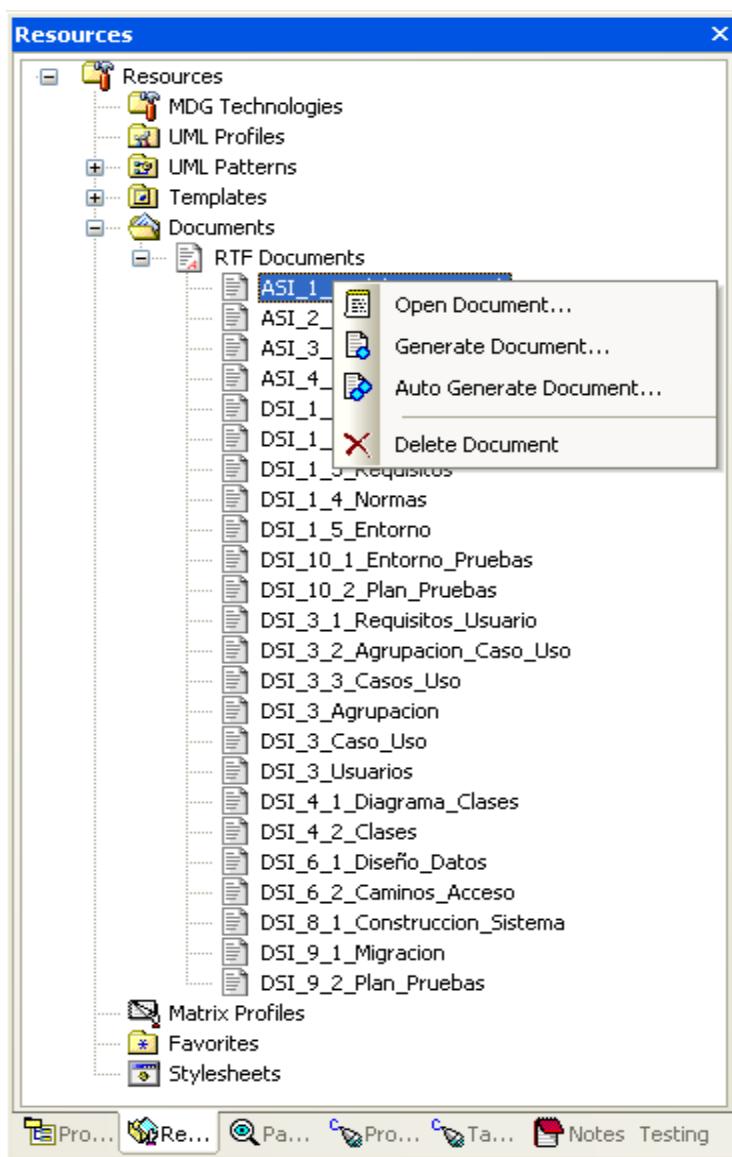
### 5.2.1. Documentación en EA

En la arquitectura estándar que se proporciona, una vez creado el repositorio se presentan las plantillas necesarias para la generación de la documentación.

En la pestaña de **Resources**, en el apartado correspondiente a **Documents -> RTF Documents**, se han creado una serie de accesos directos para la generación automática de la documentación, de tal forma que se asocia la plantilla correspondiente con la información que se desea crear.



Para generar el fichero físico correspondiente a una determinada fase, se deberán seleccionar aquellos accesos directos que formen parte de dicha documentación y ejecutar la opción **Auto Generate Document**, que se muestra al pulsar el botón derecho del ratón



Esta opción ejecutara la generación automática del documento y presentara el resultado del mismo por pantalla.

Para la generación de la documentación se identificará que documento es el que se desea generar y se ejecutarán los distintos accesos preparados para dicha generación. De tal forma que si se desea generar la documentación correspondiente al ASI, se deberán ejecutar todos los accesos que comiencen por ASI.

DOCUMENTO	ACCESOS
ASI-	ASI_1_Modulo_Funcional
	ASI_2_Indice_Requisitos
	ASI_3_Detalle_Requisitos
	ASI_4_Casos_Uso

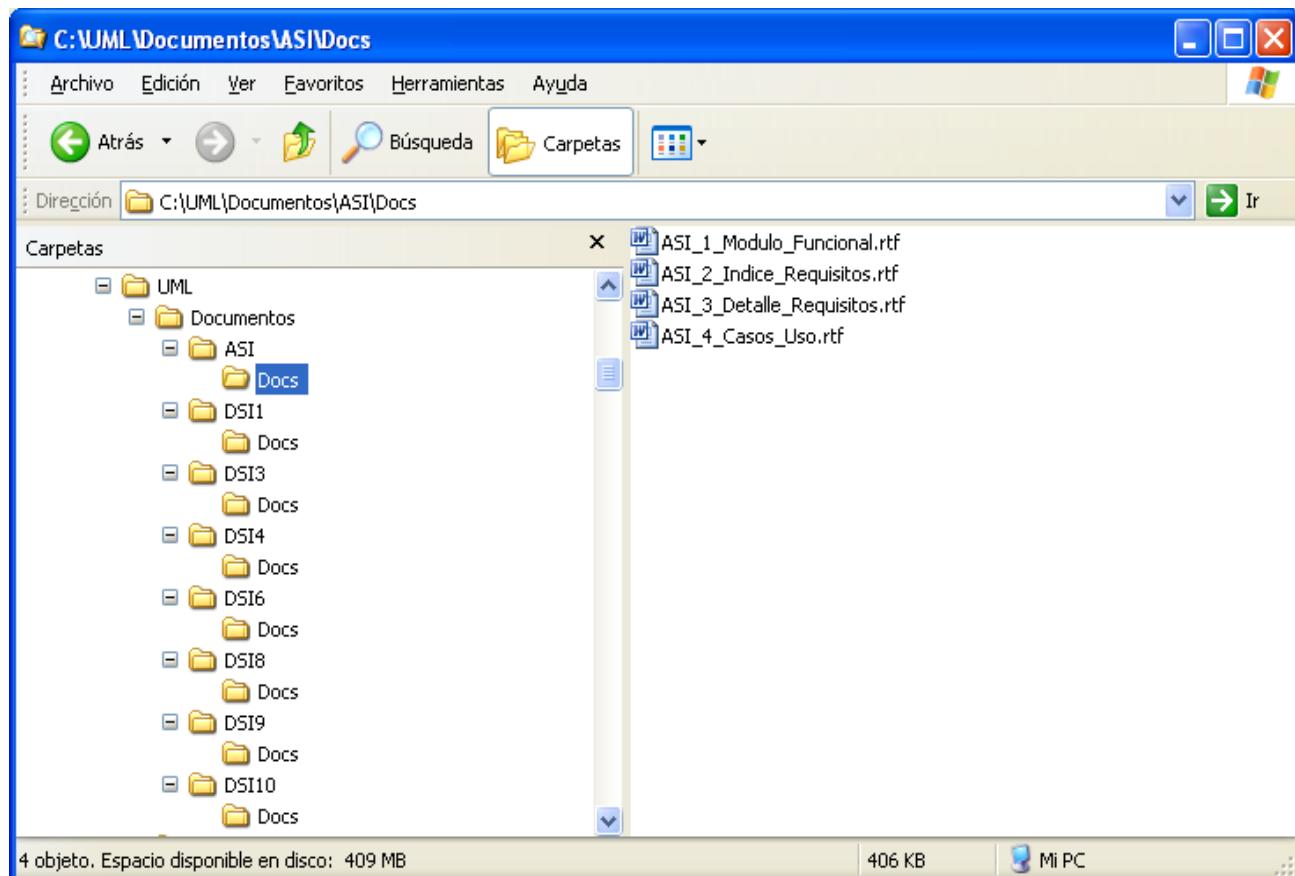
**Nota:** Cabe destacar que el nombre de las plantillas coincide con el nombre de los accesos directos preparados para la generación de la documentación

### 5.2.2. Documentación Arin Bide

Una vez ejecutados los accesos desde la herramienta, la ubicación física de los documentos generados es la siguiente:

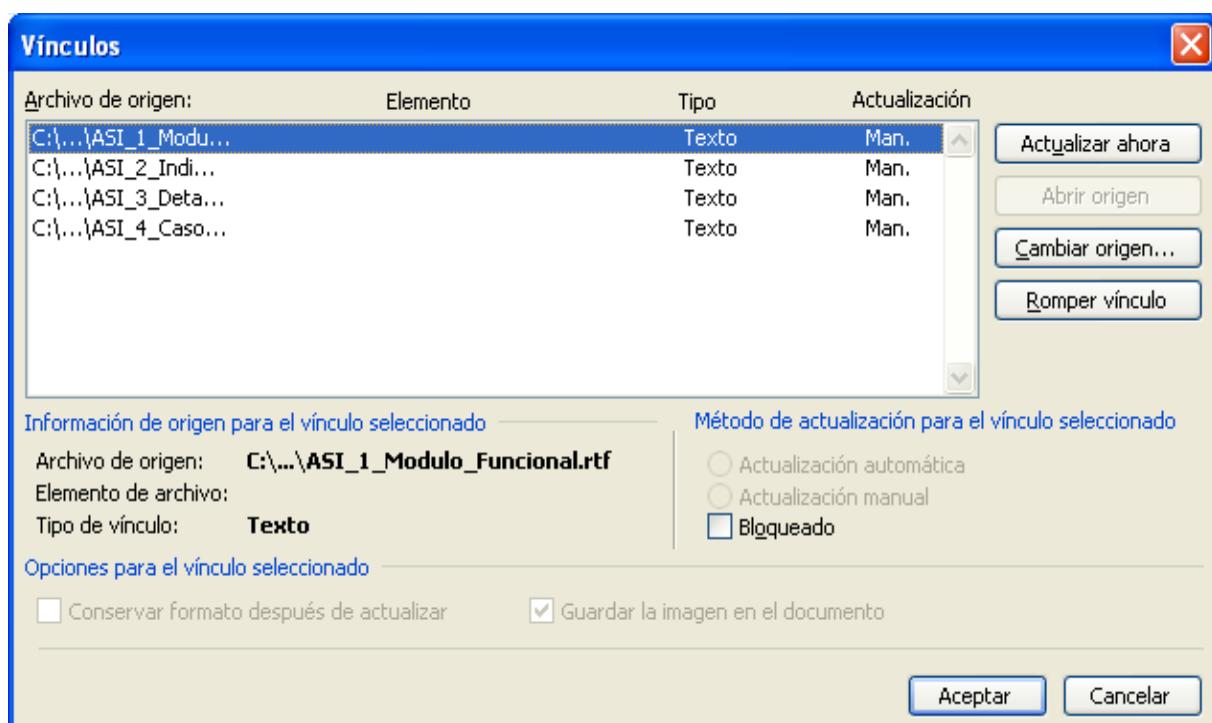
- C:\UML\Documentos\
  - ASI\Docs
  - DSI1\Docs
  - .....

De tal forma que en la carpeta **Docs**, se guardan los documentos intermedios que forma parte del documento definitivo.



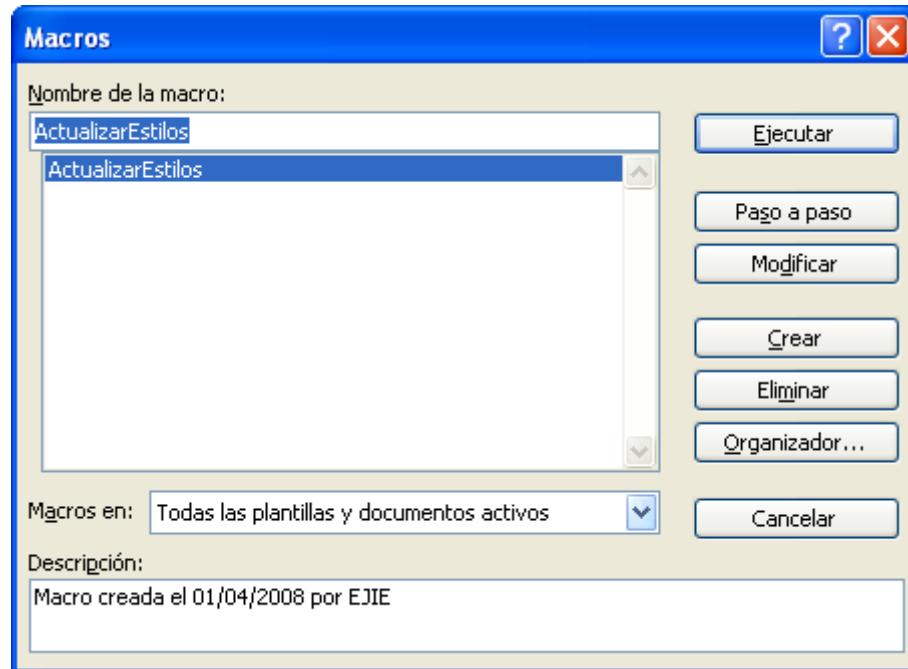
Una vez generados los documentos intermedios, se procederá a su unificación en un documento final, para ello en cada carpeta principal (ASI, DS1...) se proporciona una plantilla preparada para la creación de dicho documento. En dicha plantilla, se han vinculado los distintos documentos que forman parte del documento final, por lo que una vez generados dichos documentos, se deberán refrescar los vínculos del documento principal.

Para ello desde el documento principal, en este caso el ASI.doc, desde el menú **Edición-> Vínculos**, se presentan los distintos documentos vinculados y el orden en el que se encuentran.



Para actualizar el documento, se seleccionara de la lista de **Archivos de Origen**, aquellos documentos intermedios que han sufrido cambios y se pulsara el botón de **Actualizar ahora**.

Una vez actualizado el documento con la información correspondiente, el ultimo paso consiste en asignar los estilos de EJIE al documento, para ello se deberá ejecutar la macro **ActualizarEstilos**, desde el menú **Herramientas -> Macro -> Macros**



Una vez actualizado el documento con los estilos de EJIE, se deberá refrescar el menú del documento.

**NOTA -> Se ha de seguir el mismo proceso para la generación de cualquier documento.**

## 5.3 Análisis Sistema Información (ASI)

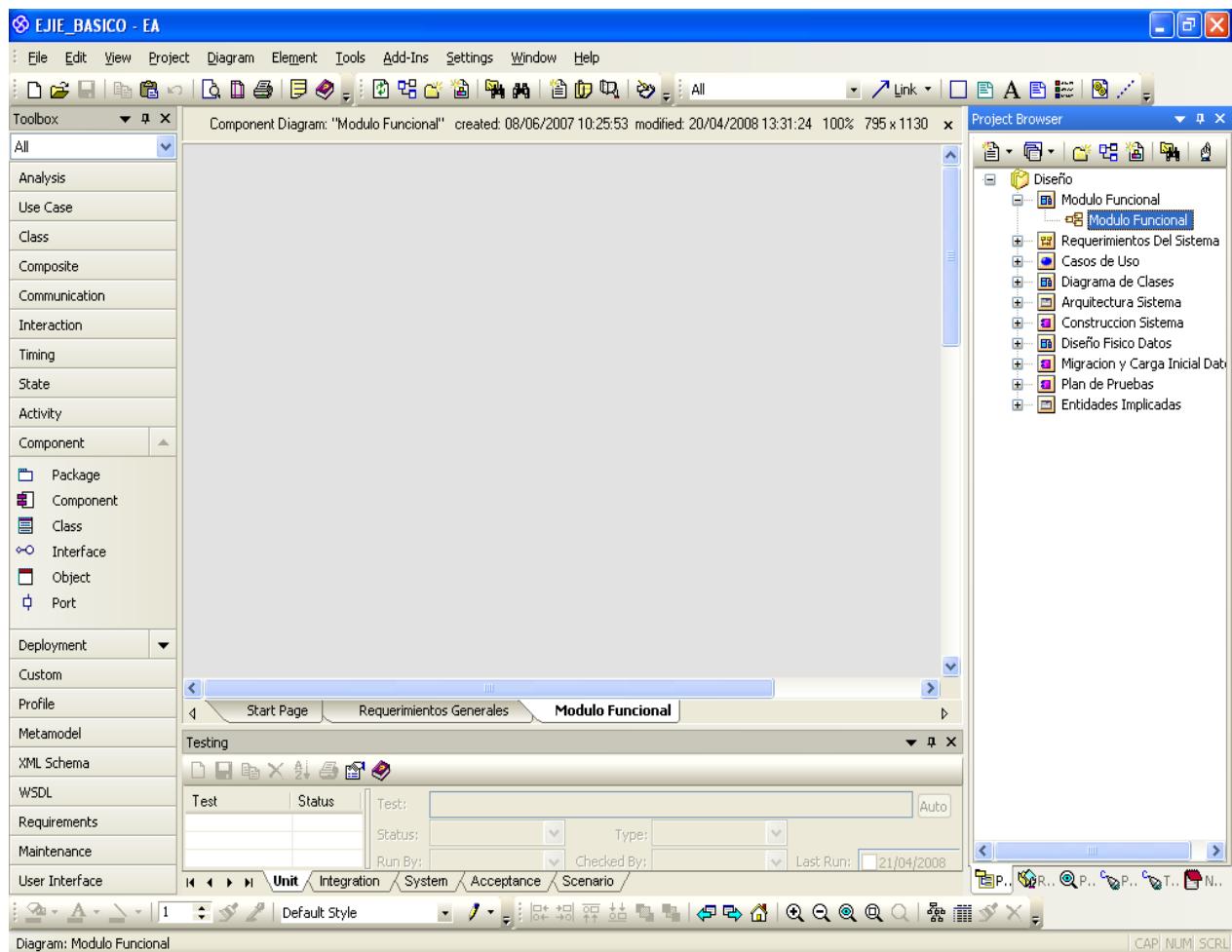
### 5.3.1. Modelado UML

Para la generación del ASI, se han de actualizar de la estructura estándar los siguientes diagramas

- **Modulo Funcional**
- **Requerimientos Sistema**
- **Casos Uso**

### Modulo Funcional

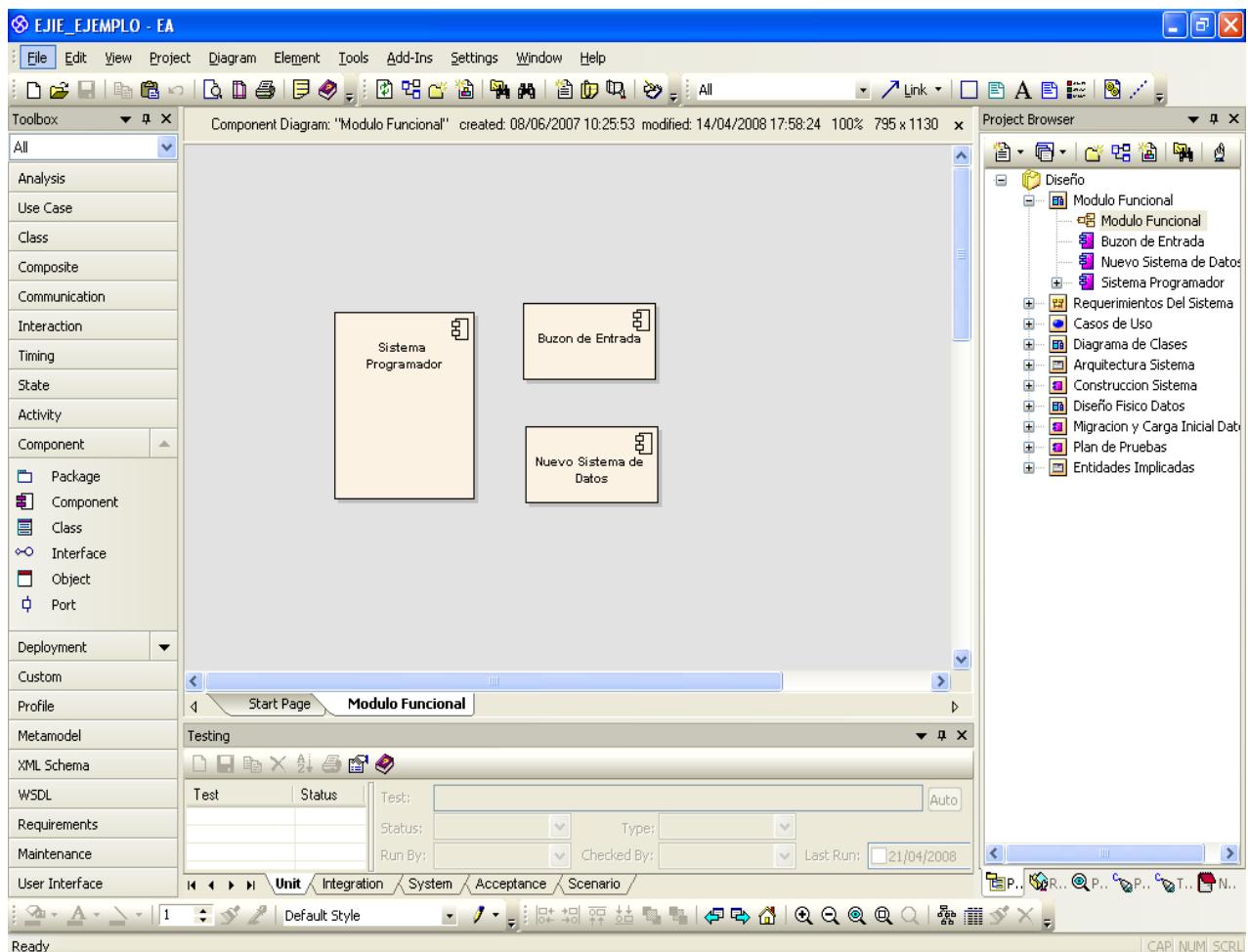
La estructura que se proporciona para la creación del modulo funcional del sistema es la siguiente:



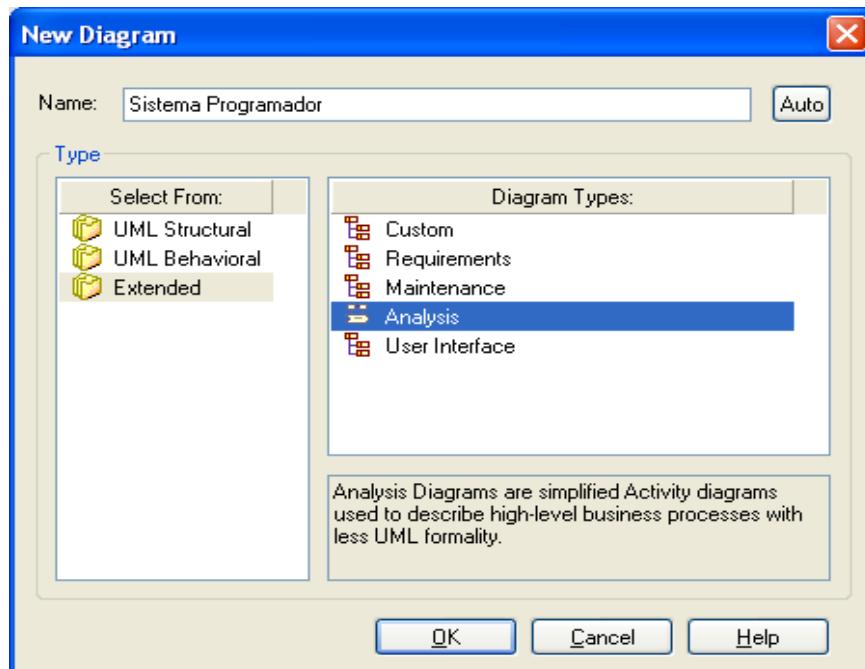
Para creación del modelado en UML, se deberán seguir los siguientes pasos:

- Agregar tantos componentes en el diagrama como módulos funcionales principales existan en la aplicación.

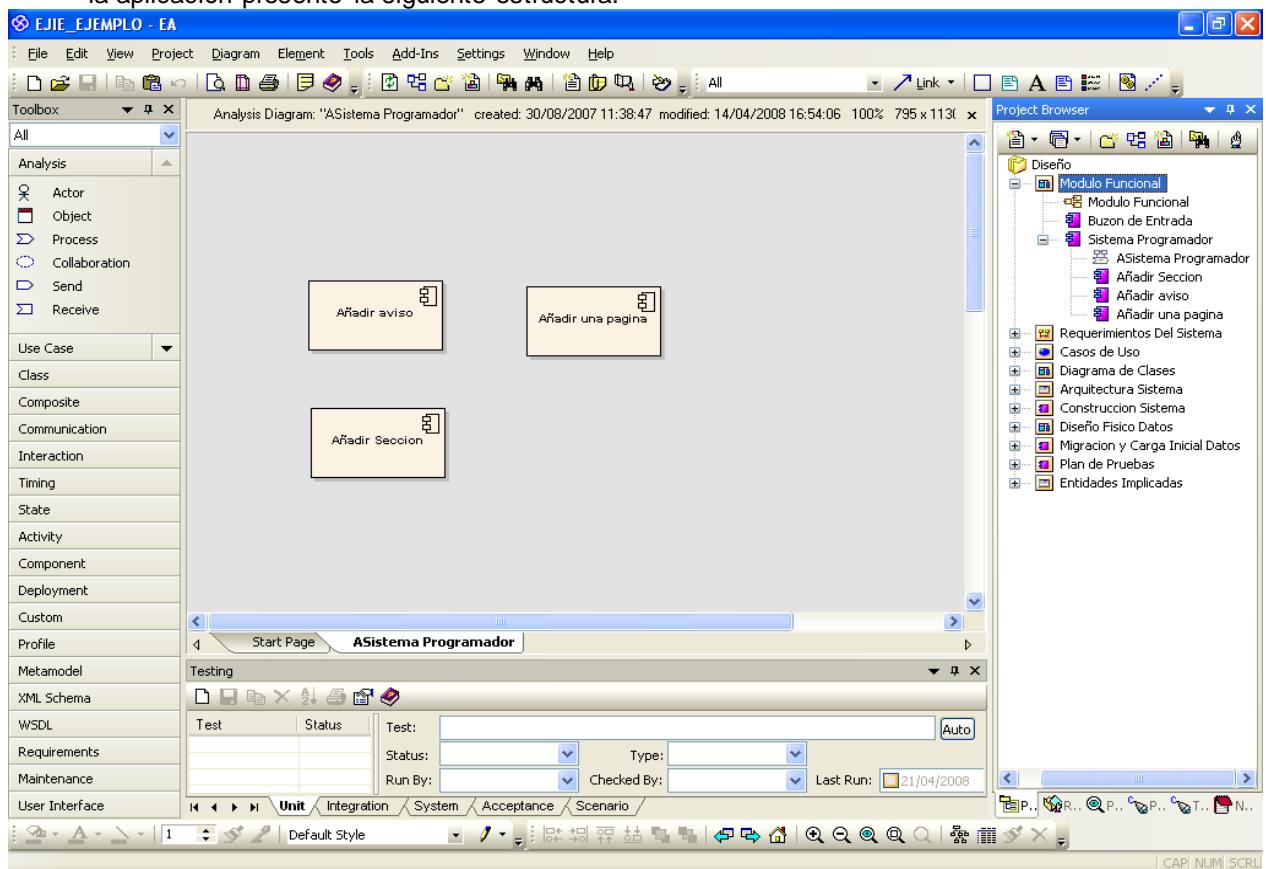
De la ventana de herramientas se deberá arrastrar el objeto **Component** al diagrama



- Por cada **Componente - Modulo Funcional** agregado, se deberán vincular tantos componentes como módulos secundarios formen parte del modulo funcional.  
 Para ello se agrega un diagrama al modulo principal y se agregan tantos componentes como módulos secundarios existan.



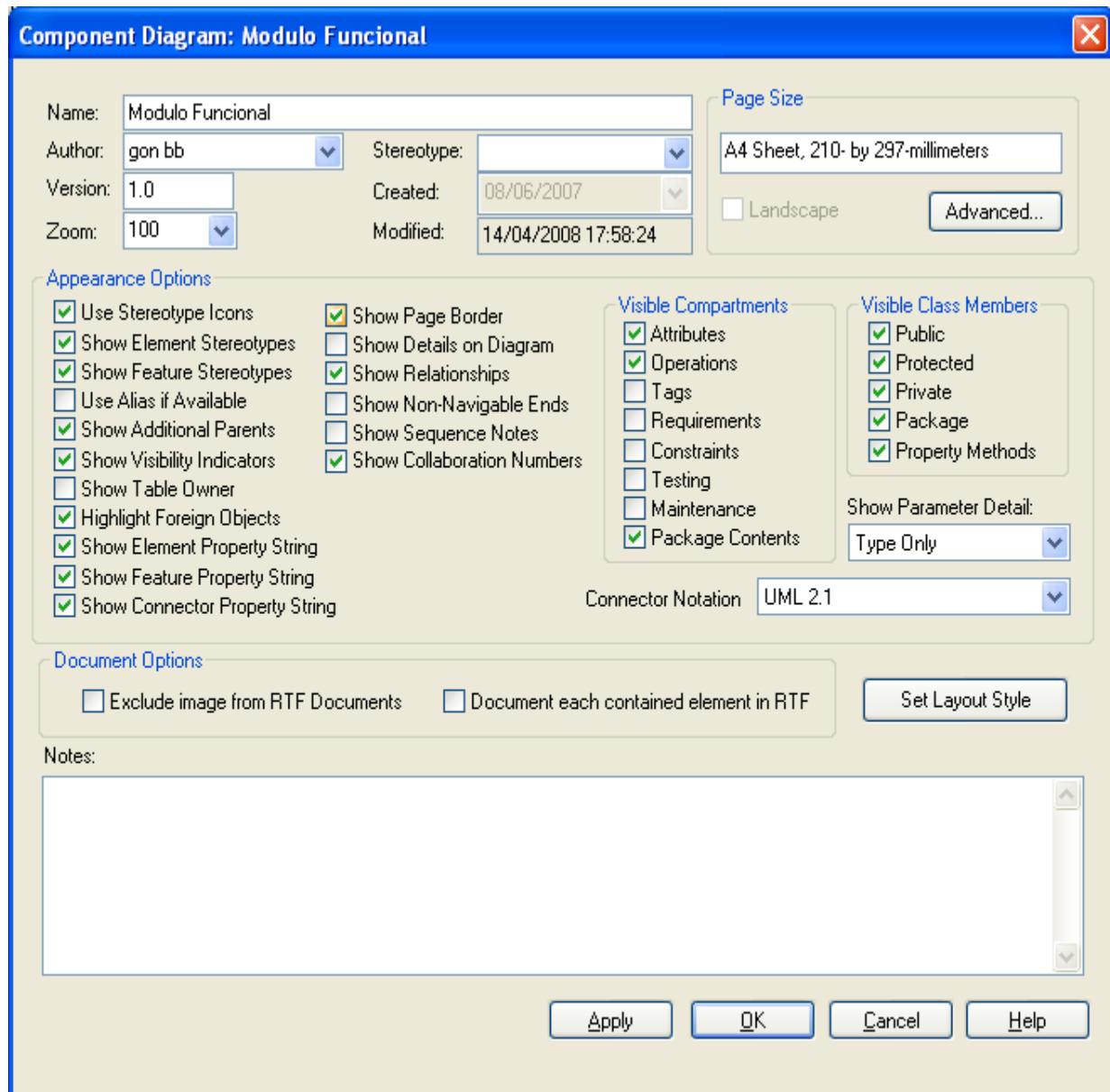
De tal forma que la estructura final, una vez creados los módulos funcionales correspondientes a la aplicación presente la siguiente estructura.



- En cada modulo agregado al diagrama, el campo **Name**, se deberá informar con el nombre del modulo funcional y el campo **Notes**, se deberá informar con la descripción funcional.

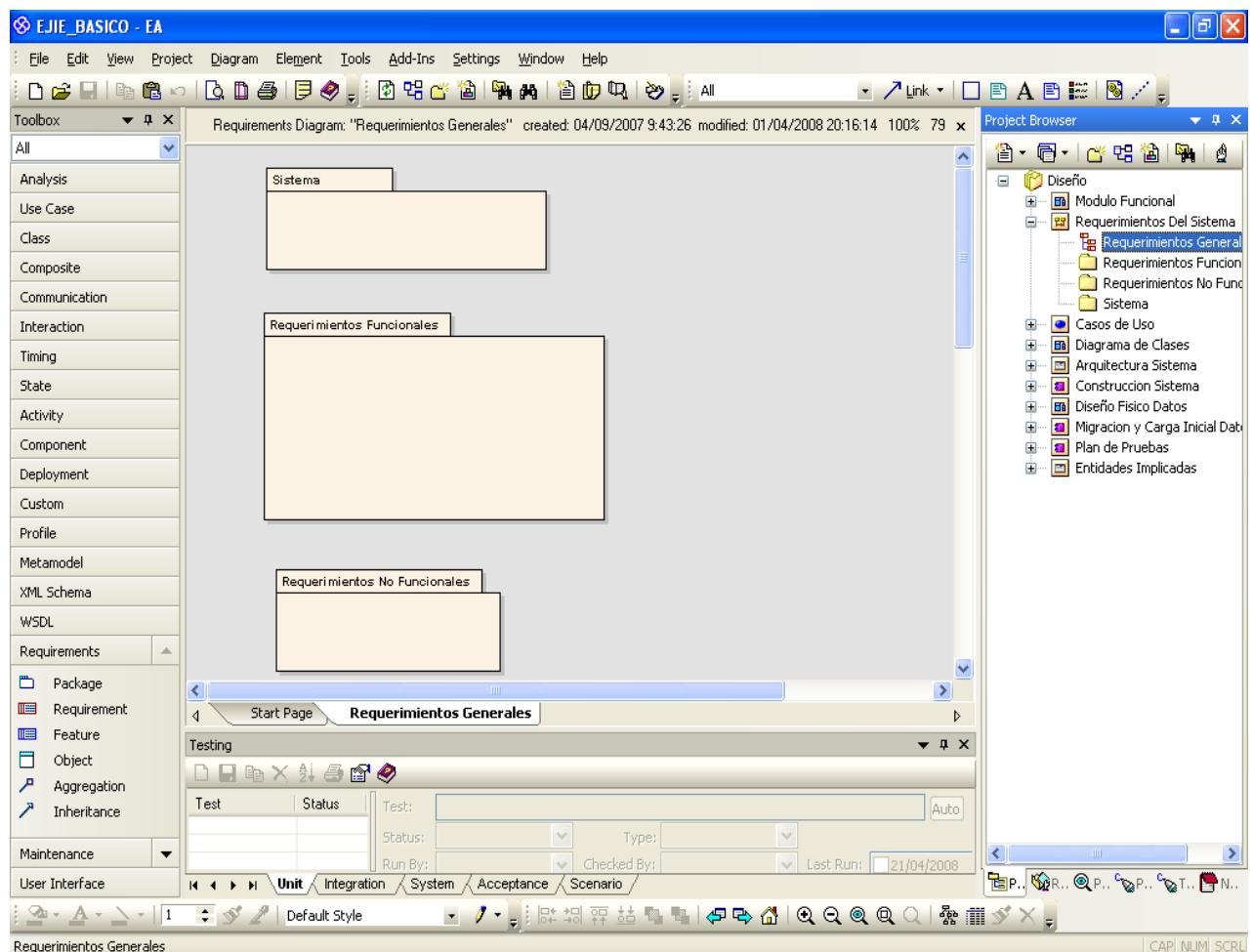


Por ultimo, es importante destacar que todos los diagramas que se creen en esta sección deberán presentar la siguiente estructura.(Seleccionar el diagrama y pulsar la opción **Properties** del menú que aparece al pulsar el botón derecho del ratón)

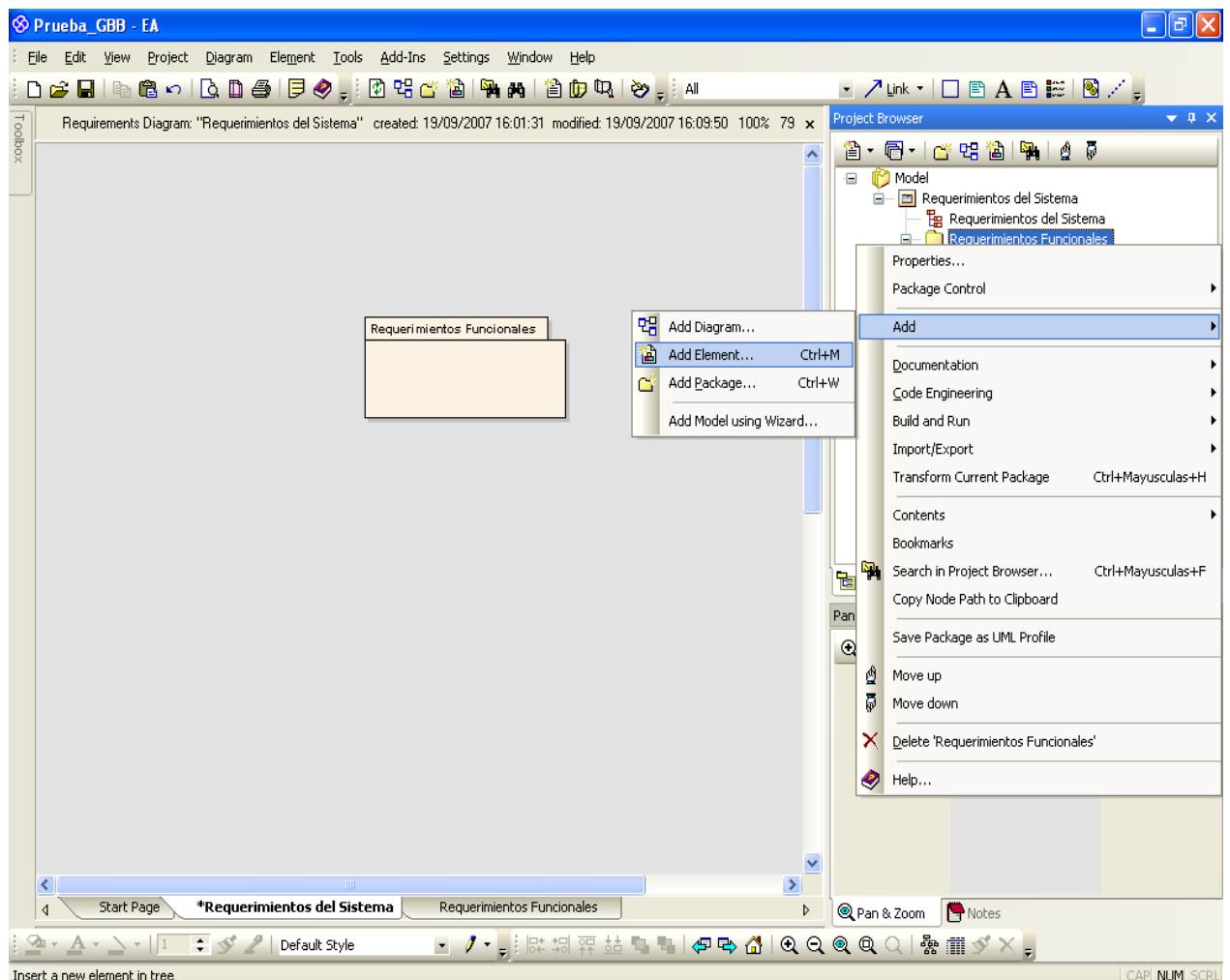


## Requerimientos del Sistema

La estructura que se proporciona para la creación de los requerimientos del sistema es la siguiente:

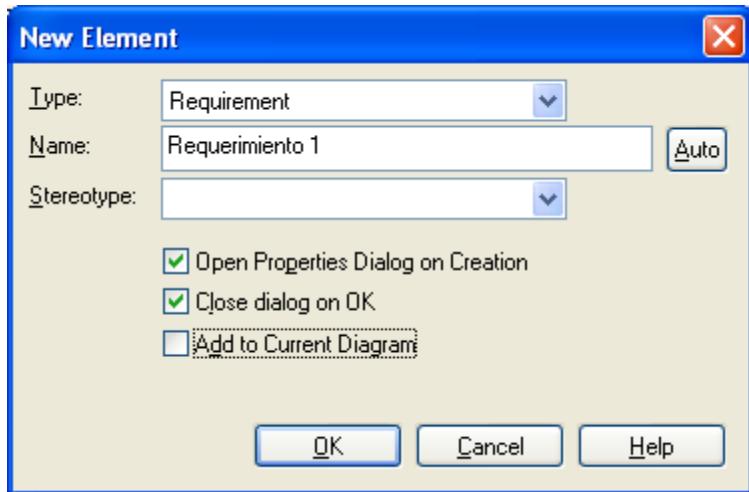


Definidos los distintos tipos de requerimientos del sistema, el siguiente paso es agregar cada uno de los requerimientos en el tipo correspondiente. Para ello se deberá seleccionar el tipo de requerimiento en el que se quiere agregar un nuevo elemento y del menú emergente del botón derecho del ratón seleccionar la opción **Add -> Add Element**



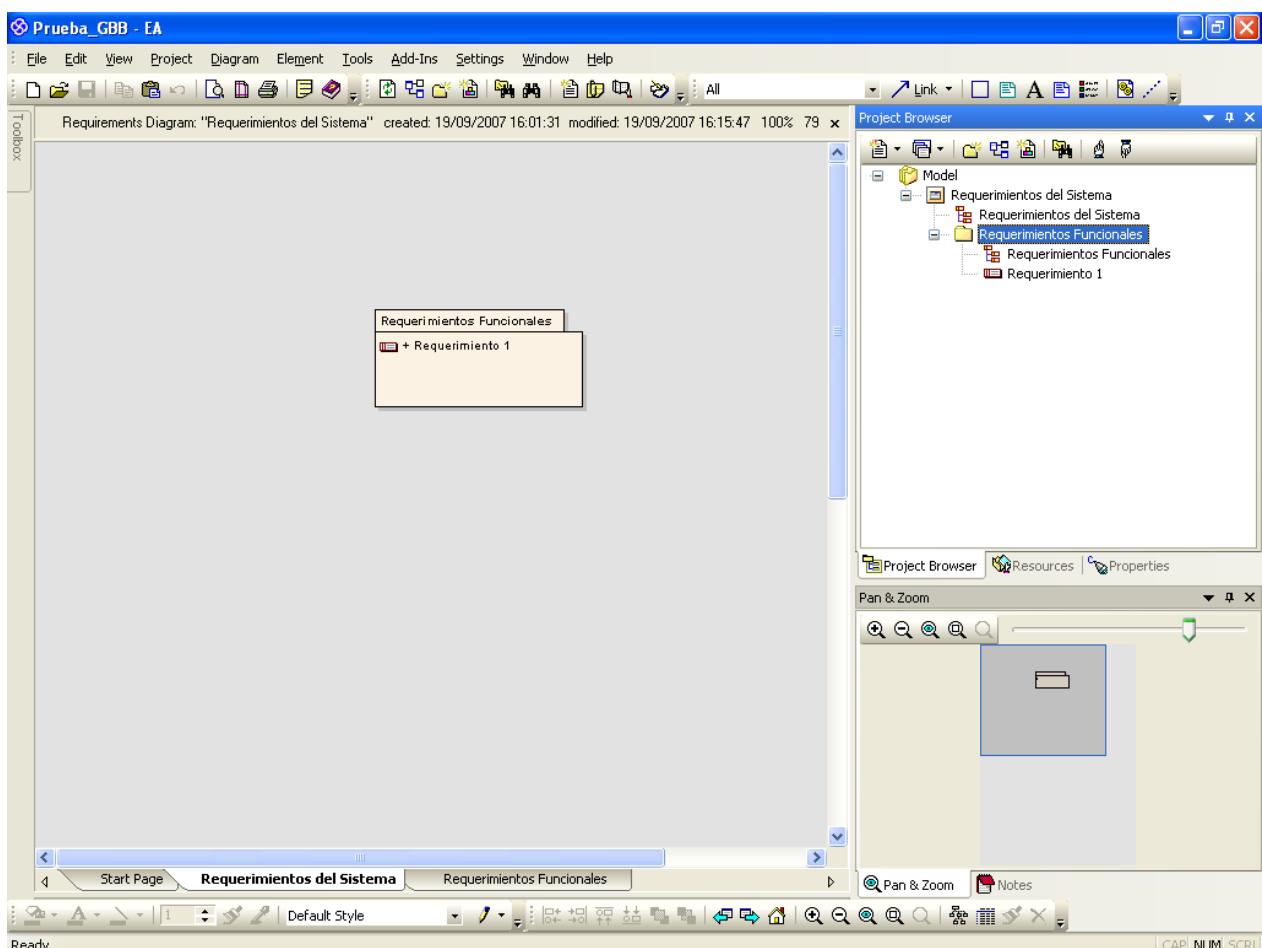
A continuación se mostrara una ventana con las siguientes opciones:

- **Type:** Tipo del elemento a agregar
- **Name:** Nombre del elemento.
- **Stereotype:** Estereotipo del elemento a añadir.



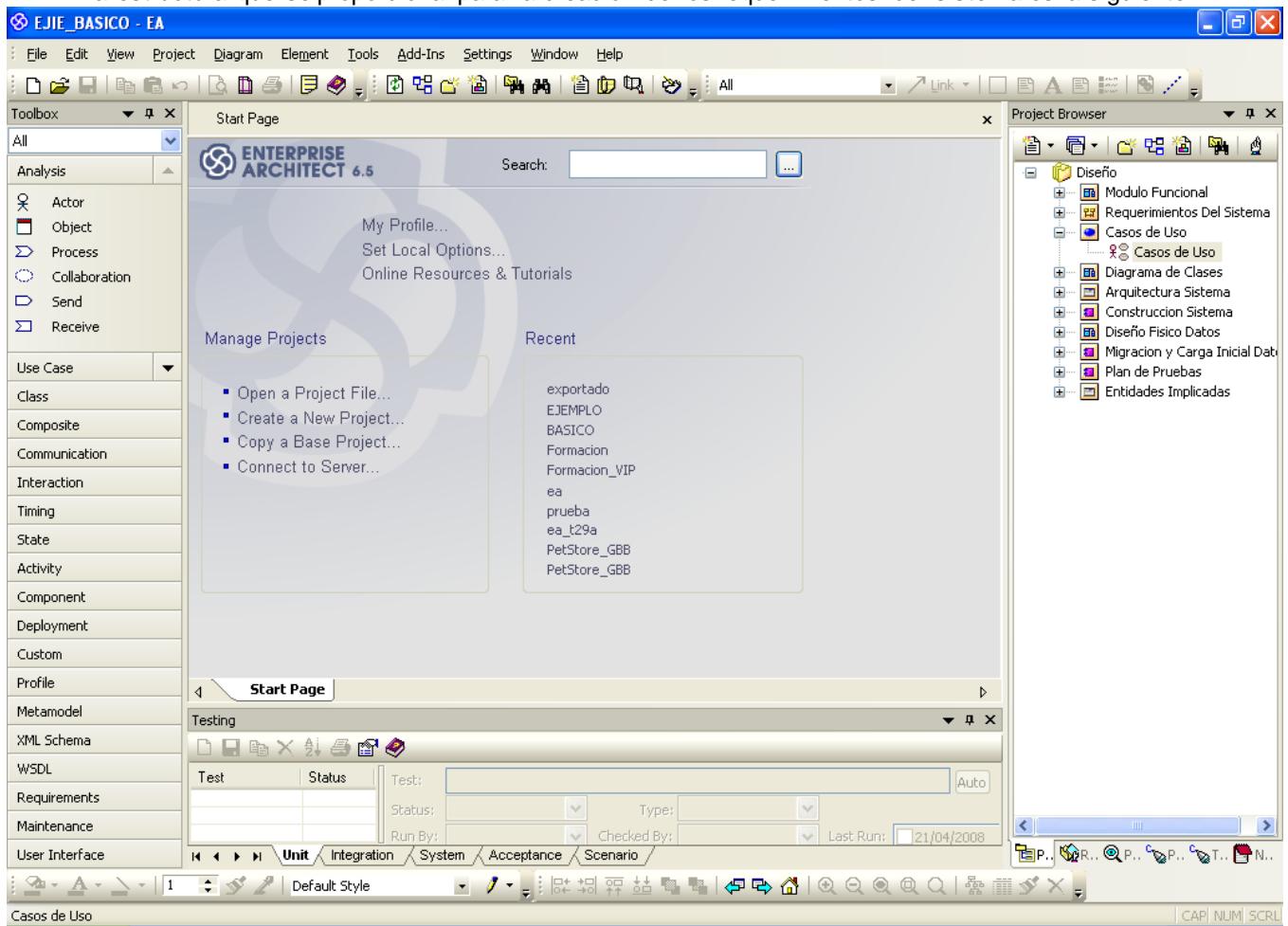
Por defecto, el campo **Stereotype** tiene el valor **Functional**. Se puede introducir un estereotipo particular que la herramienta es capaz de almacenar para la siguiente operación.

Finalmente el diagrama de requerimientos quedara tal y como se muestra a continuación:

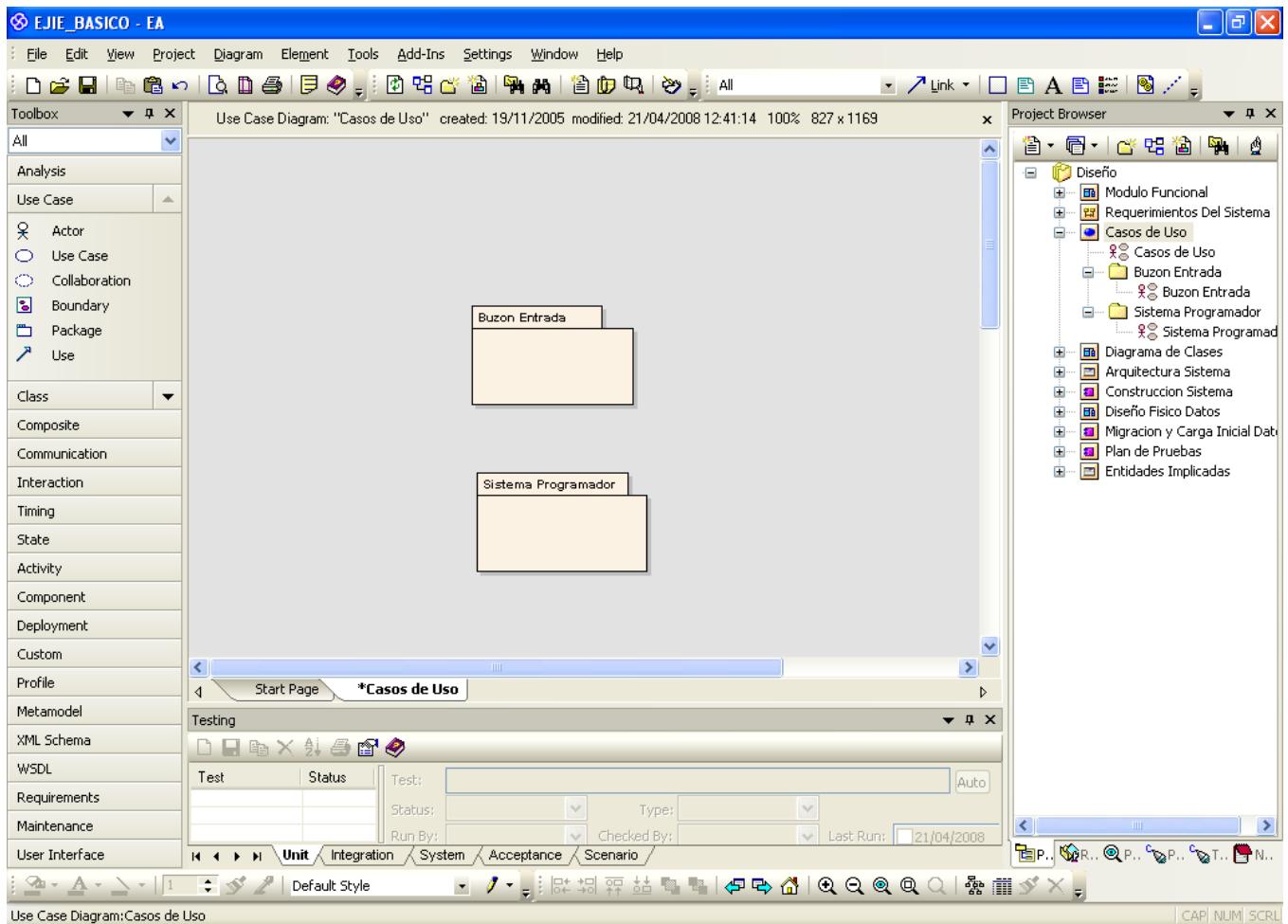


## Casos de Uso

La estructura que se proporciona para la creación de los requerimientos del sistema es la siguiente:

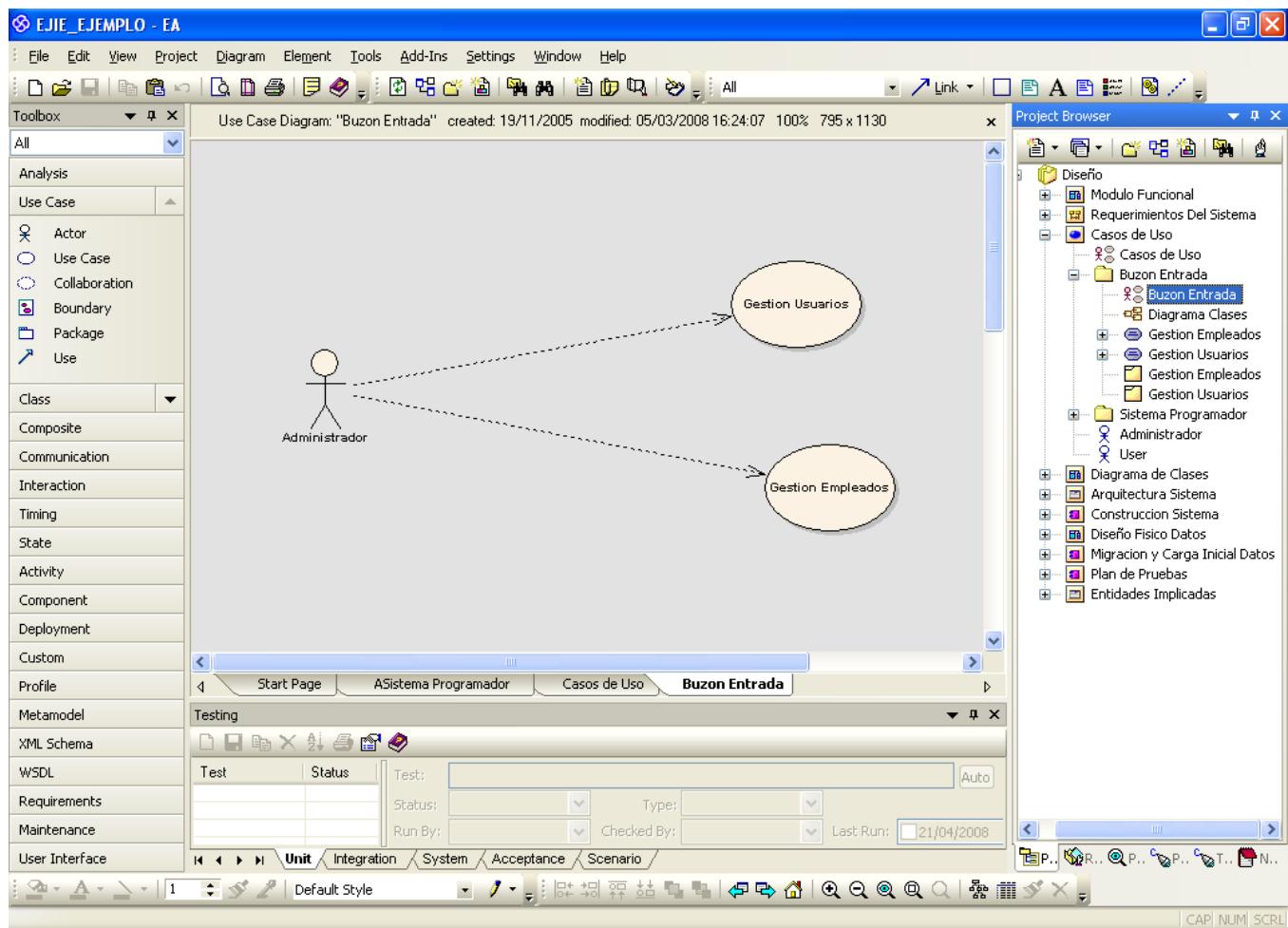


El primer paso que se deberá realizar es crear tantas carpetas como módulos funcionales se vayan a analizar e incorporarlas al diagrama principal. Para ello bastara con arrastrar de la barra de herramientas de la sección de **Analysis** el objeto **Package** sobre el diagrama.

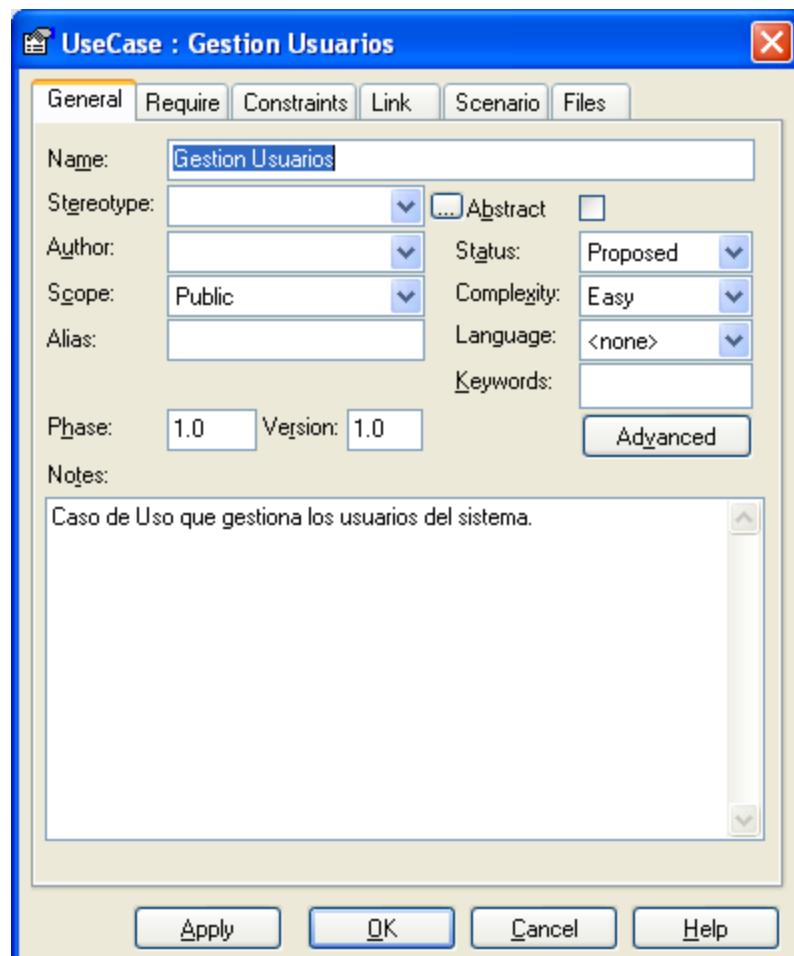


Por cada paquete, se crea un diagrama en el que se deberán modelar los distintos casos de uso que forman parte del modulo funcional que se está modelando.

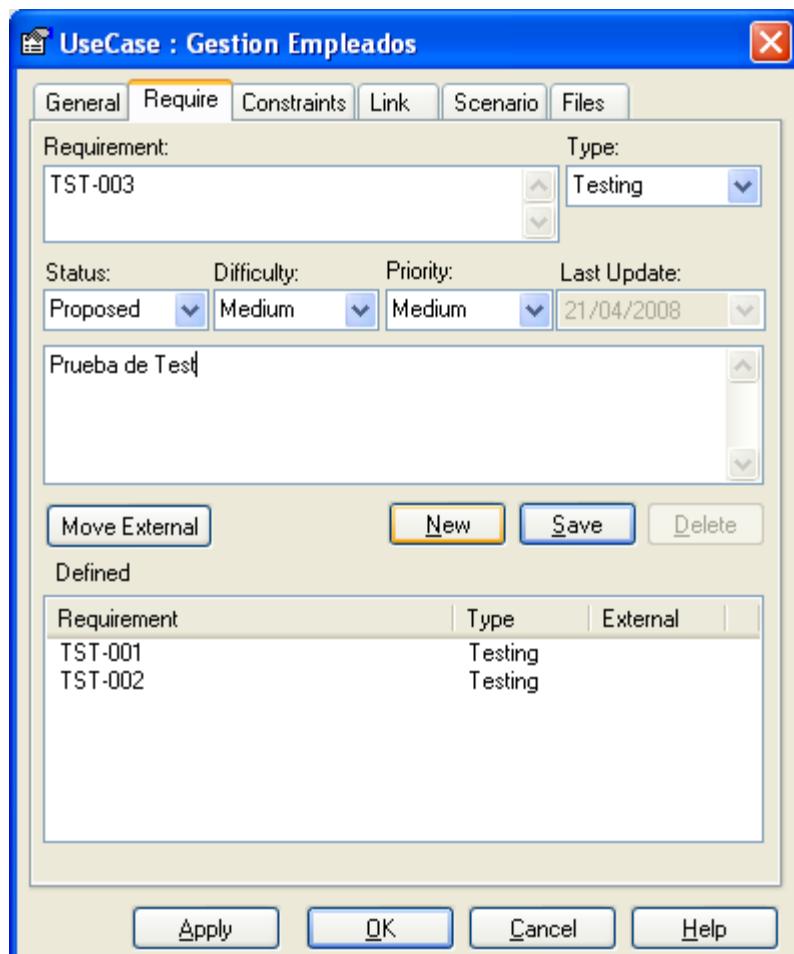
Para ello se deberán identificar los actores y los casos de uso en los que intervienen dichos actores.



En cada caso de uso, se deberá informar el campo **Name**, y el campo **Notes**.

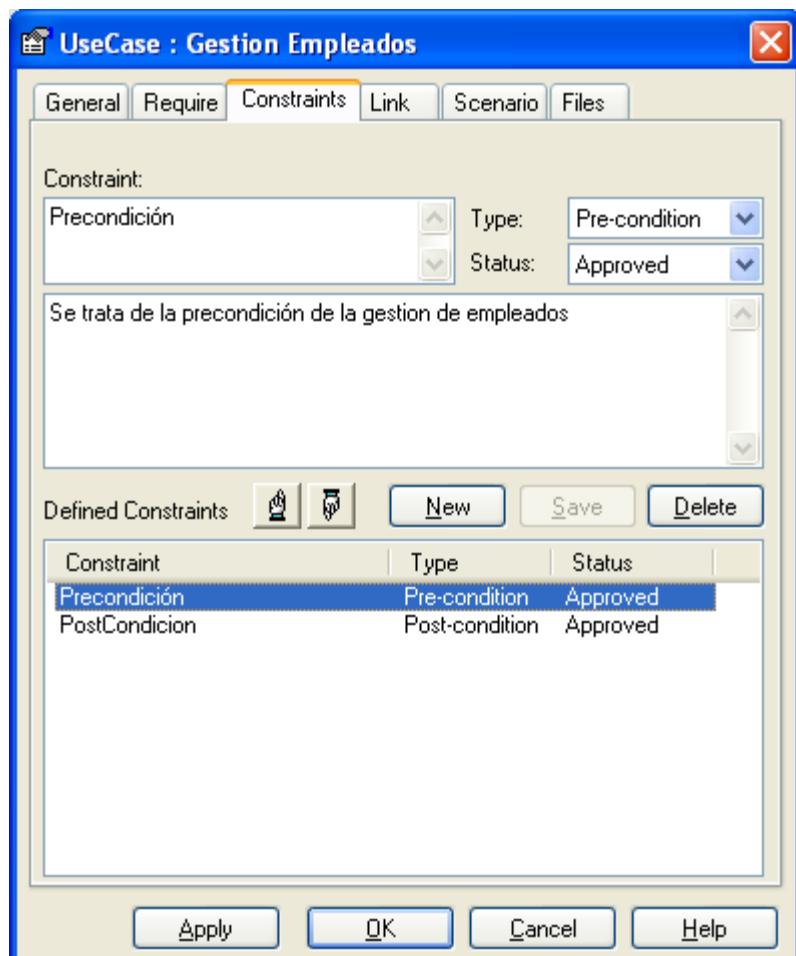


Por cada caso de uso se deberán definir las pruebas que se van a realizar, para ello desde la pestaña **Require** se agregarán tantos test como sean necesarios, tal y como se indica a continuación

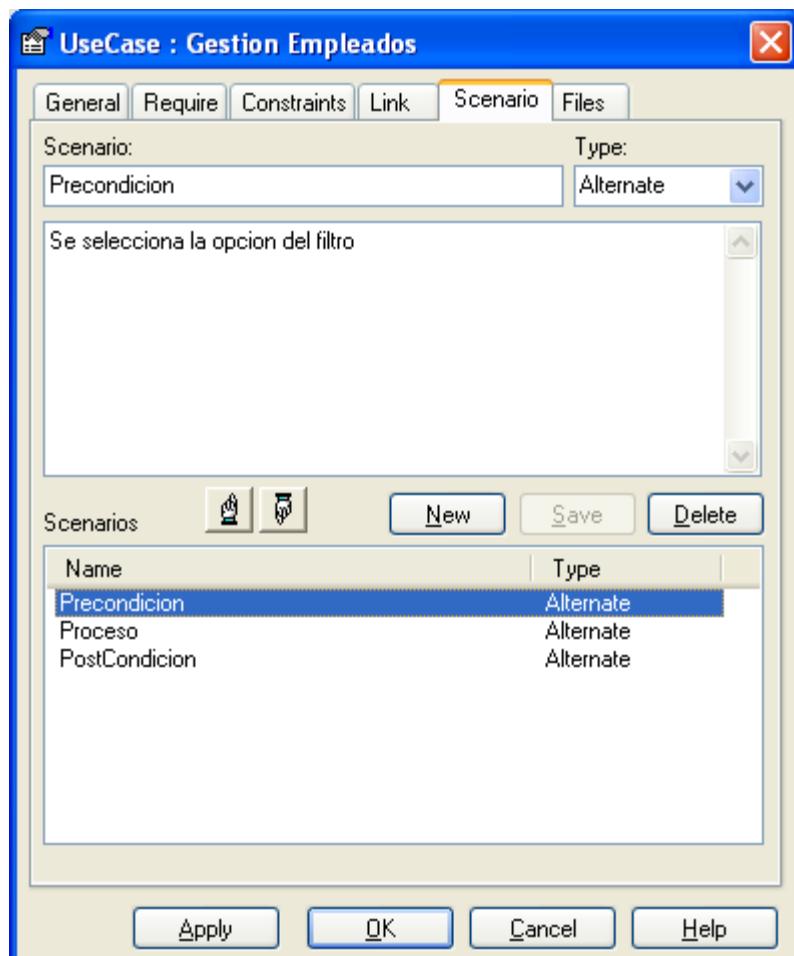


Una vez introducidos los datos se pulsar el botón **Save**.

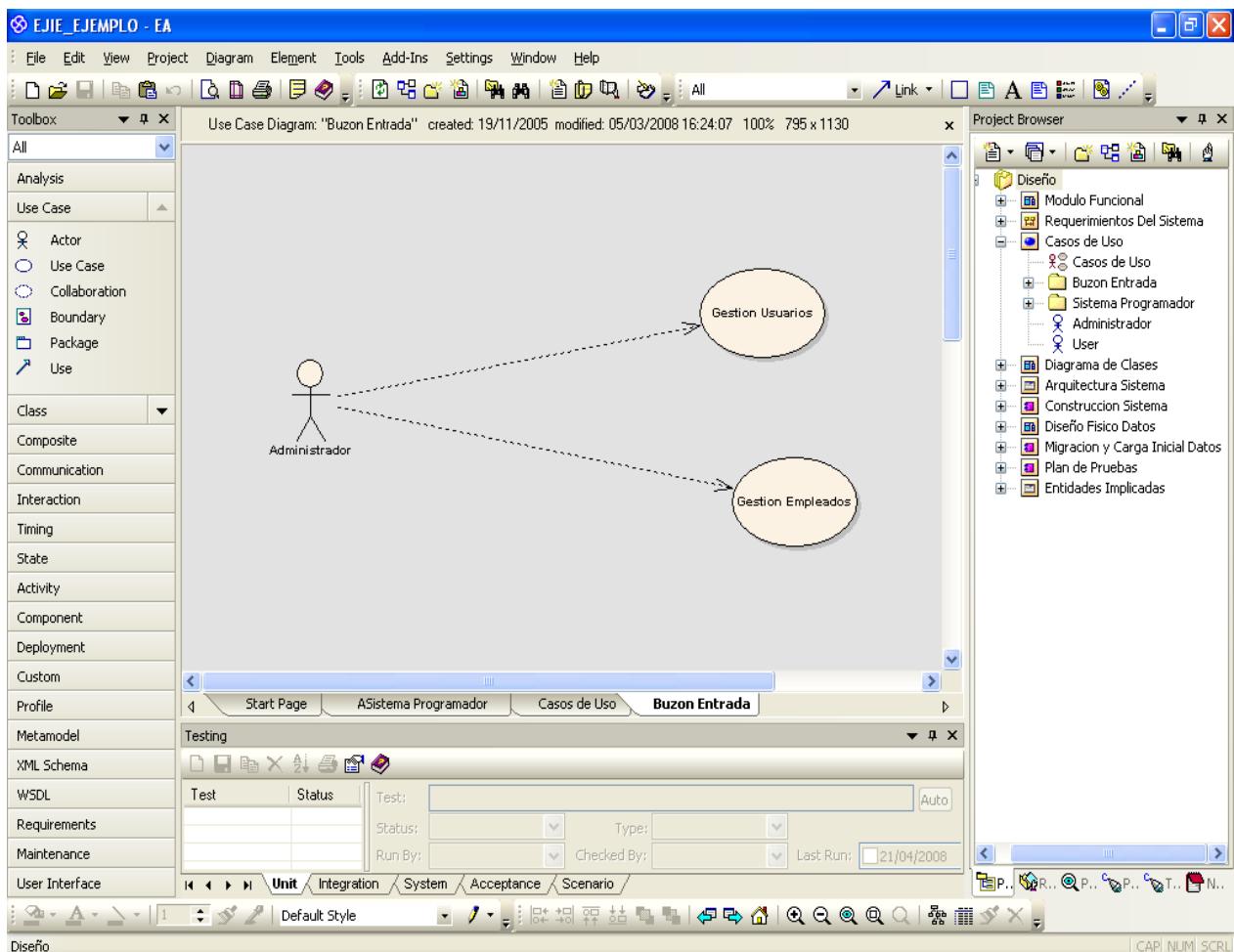
En la pestaña **Constraints**, se podrán definir las condiciones en las que se realizar el caso de prueba



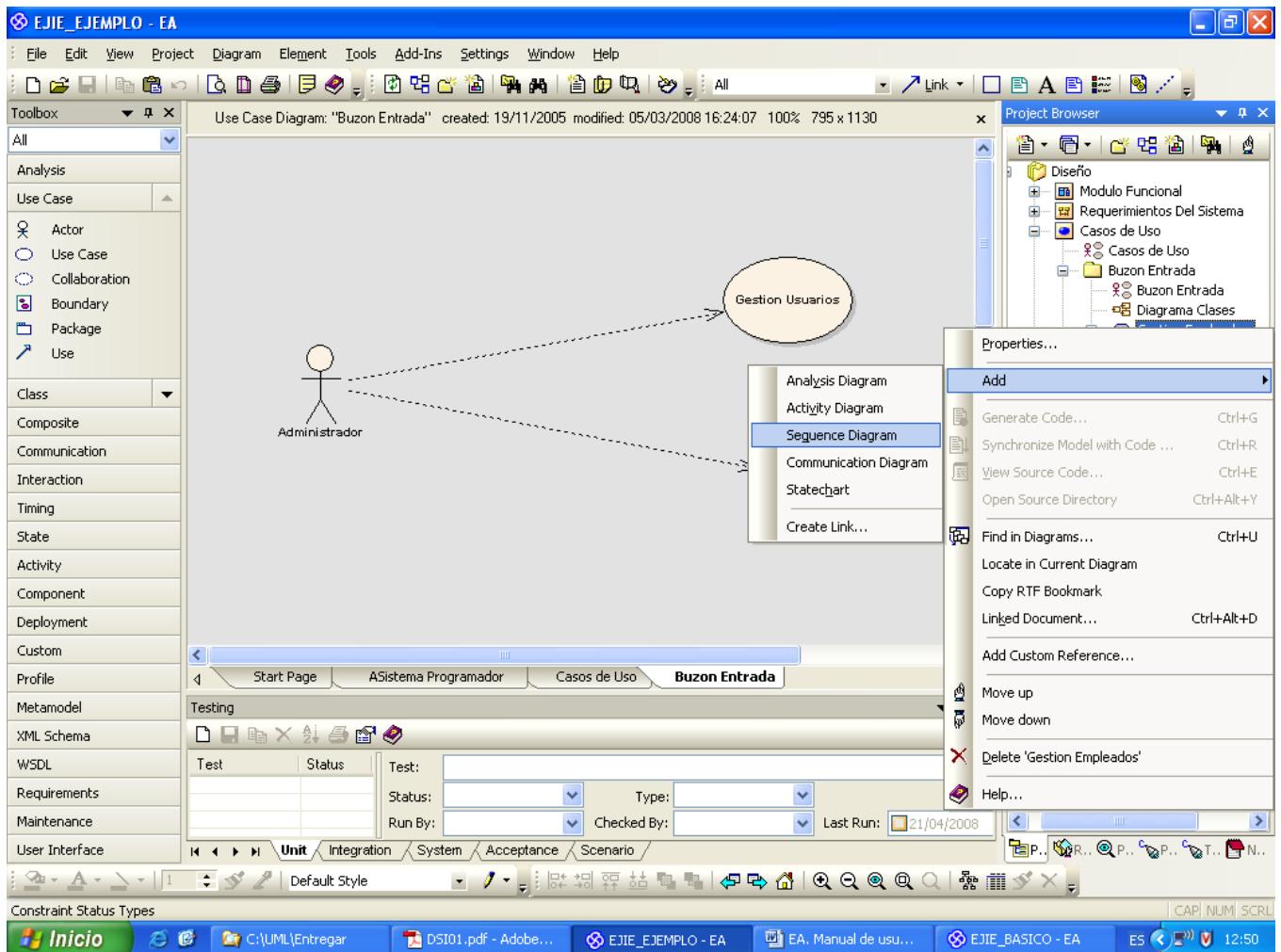
Y en la pestaña **Scenario**, se indicaran los distintos escenarios del caso de uso



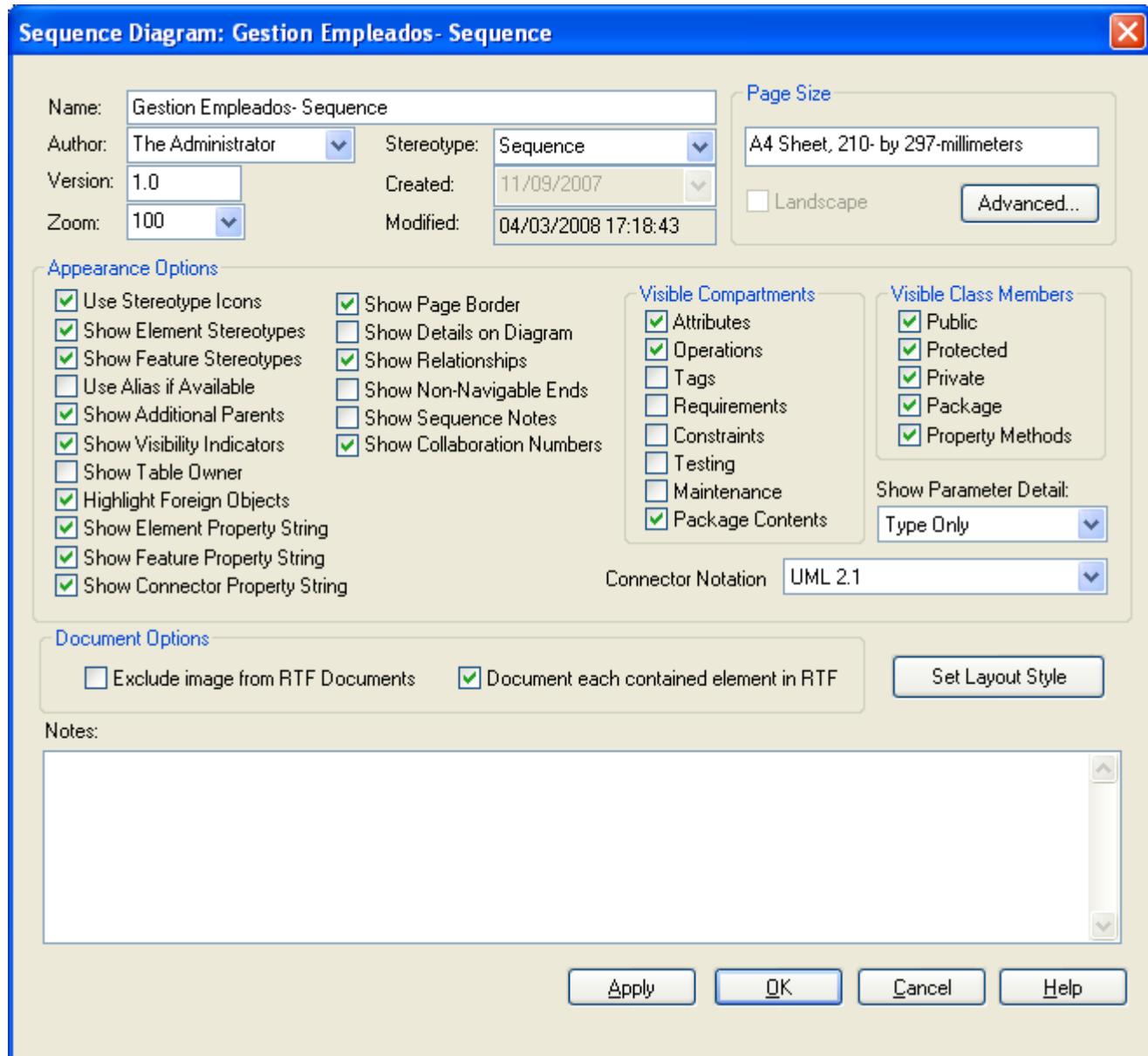
Los actores una vez identificados se deberán ubicar al mismo nivel que los módulos funcionales, de tal forma que se puedan compartir entre los distintos casos de uso



Una vez definidos los casos de uso, por cada uno de ellos vamos a crear su diagrama de secuencia. Para ello se deberá seleccionar el caso de uso y con el botón derecho, seleccionar **Add->Sequence Diagram**



Todo diagrama de secuencia que se cree, deberá presentar la siguiente estructura(Seleccionar diagrama y pulsar **Properties** del menú que se muestra al pulsar el botón derecho)

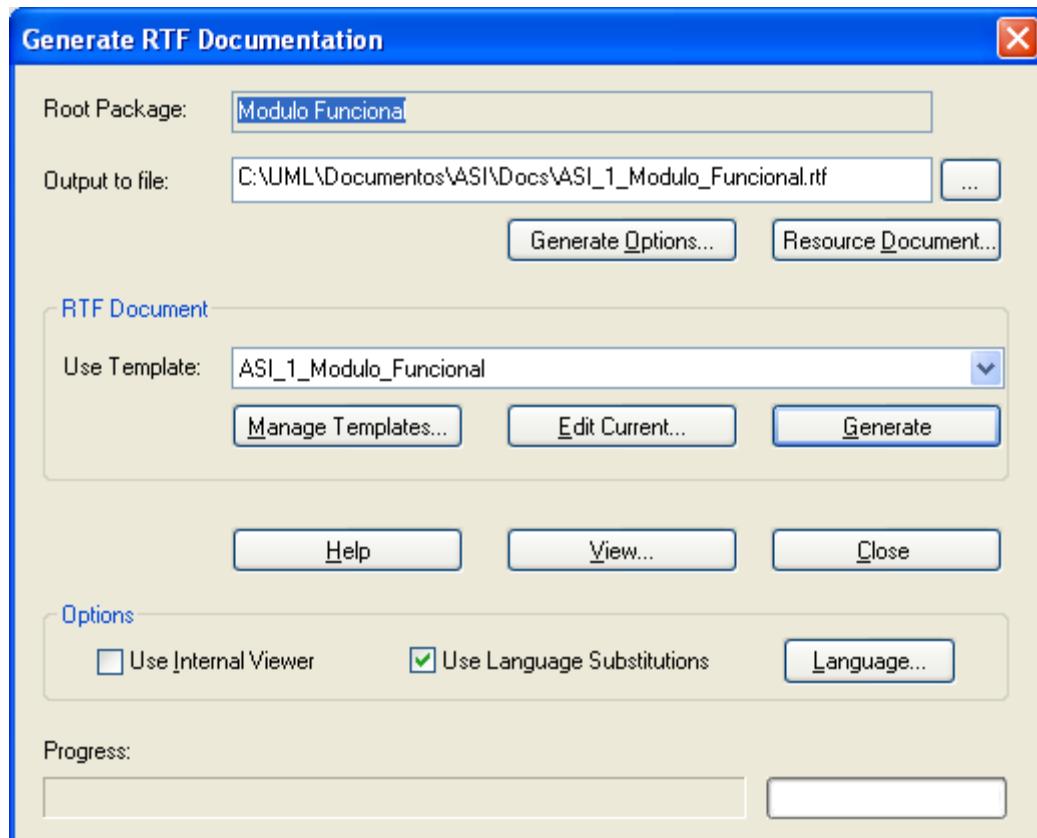


Para el modelado de los casos de uso, se deberán utilizar las clases definidas en la sección **Entidades Implicadas**.

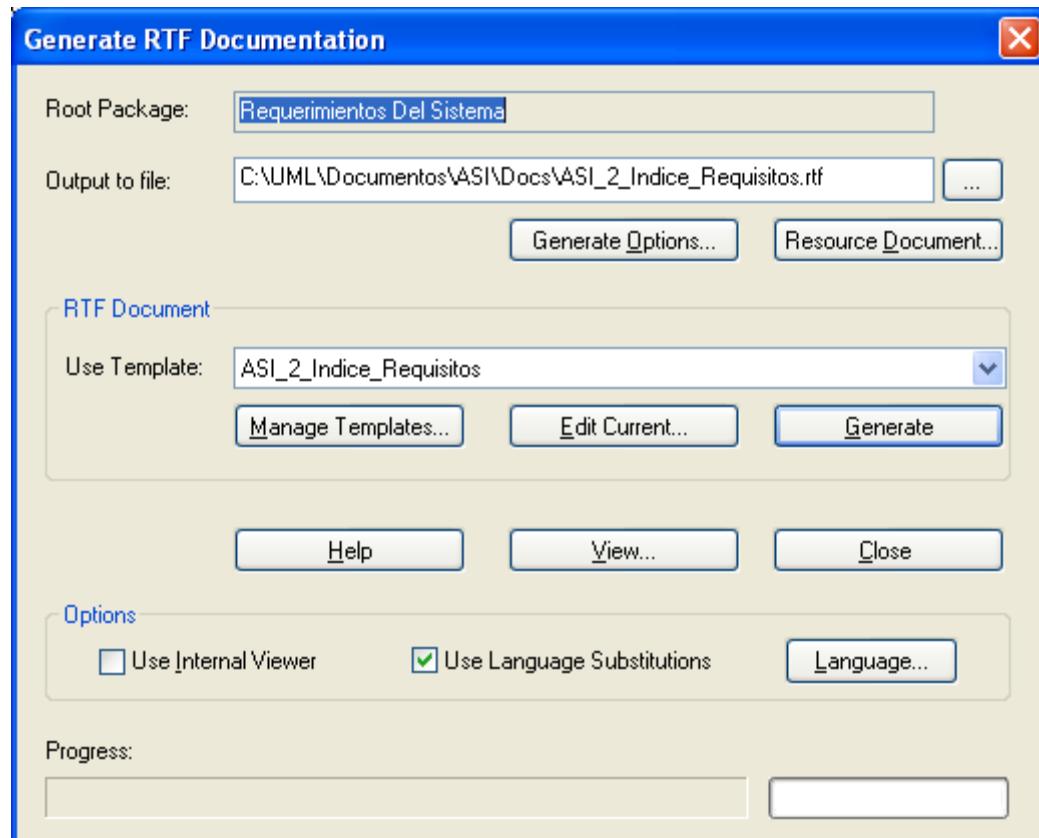
### 5.3.2. Generación Documentación

Los accesos directos que se han de ejecutar para la creación de los documentos intermedios son las siguientes

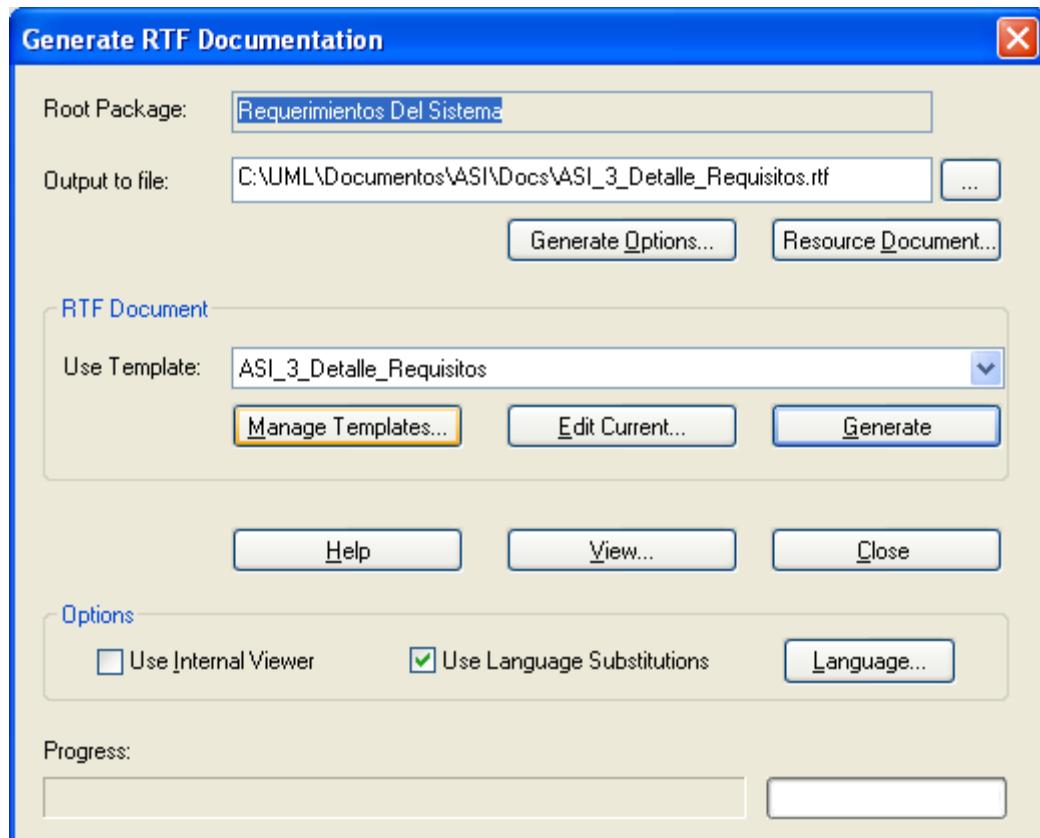
#### **ASI\_1\_Modulo\_Funcional**



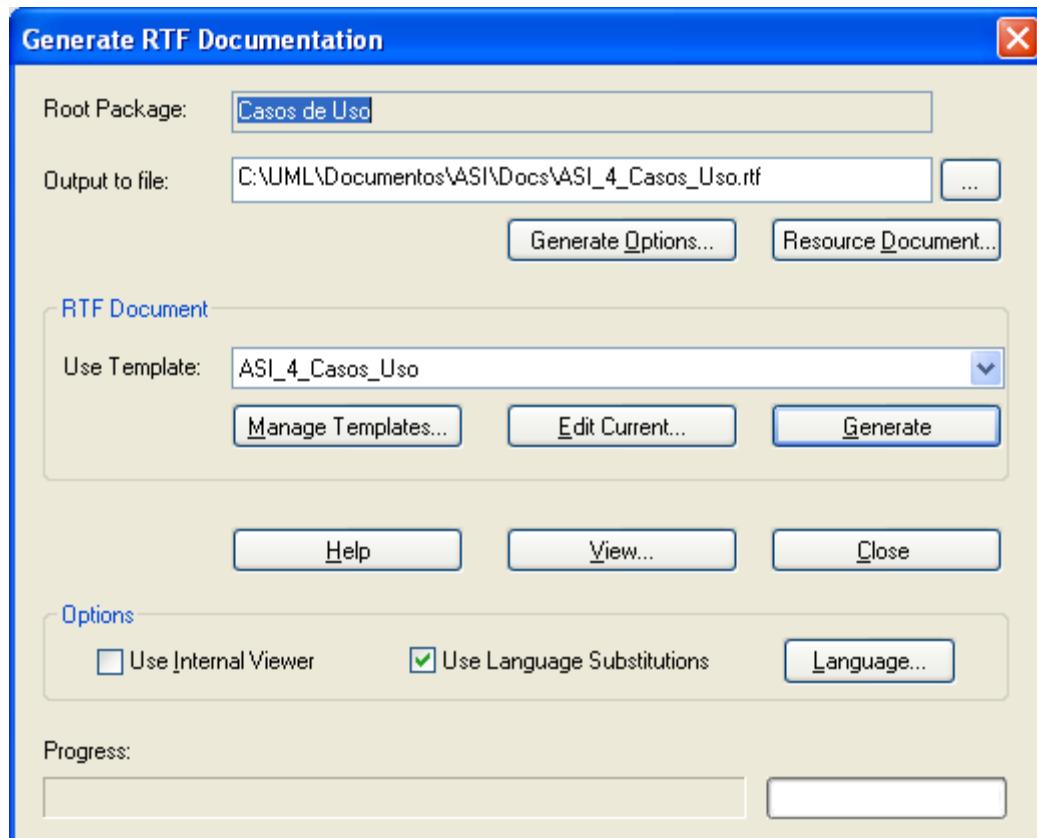
## ASI\_2\_Indice\_Requisitos



### ASI\_3\_Detalle\_Requisitos



### ASI\_4\_Casos\_Uso



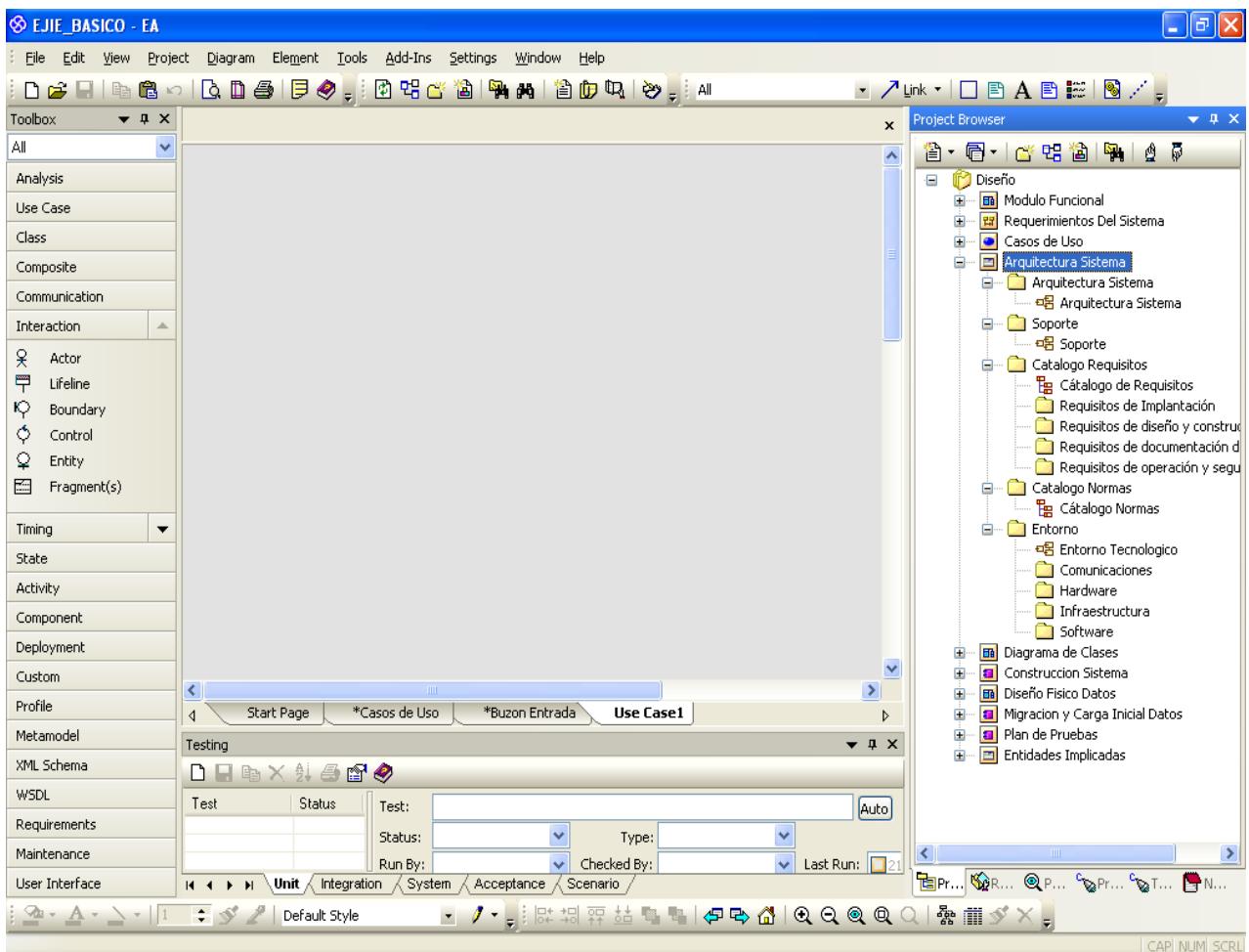
## 5.4 Definición de la Arquitectura del Sistema – DSI 1

### 5.4.1. Modelado UML

Para la generación de la documentación, se han de actualizar de la estructura estándar el siguiente diagrama

- **Arquitectura Sistema**

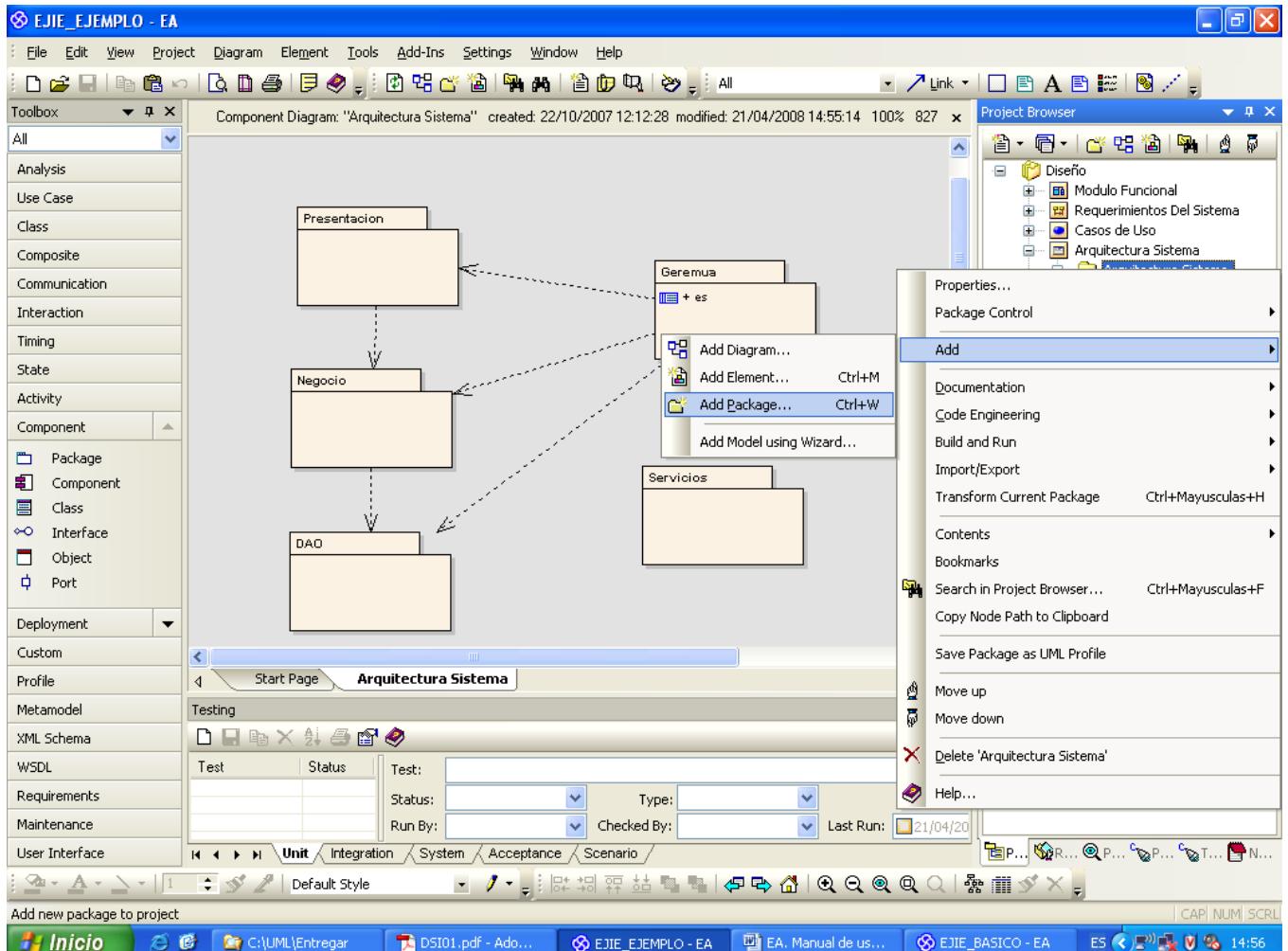
La estructura que se proporciona es la siguiente:



Para creación del modelado en UML, se deberán seguir distintos pasos en función de la carpeta en la se este trabajando:

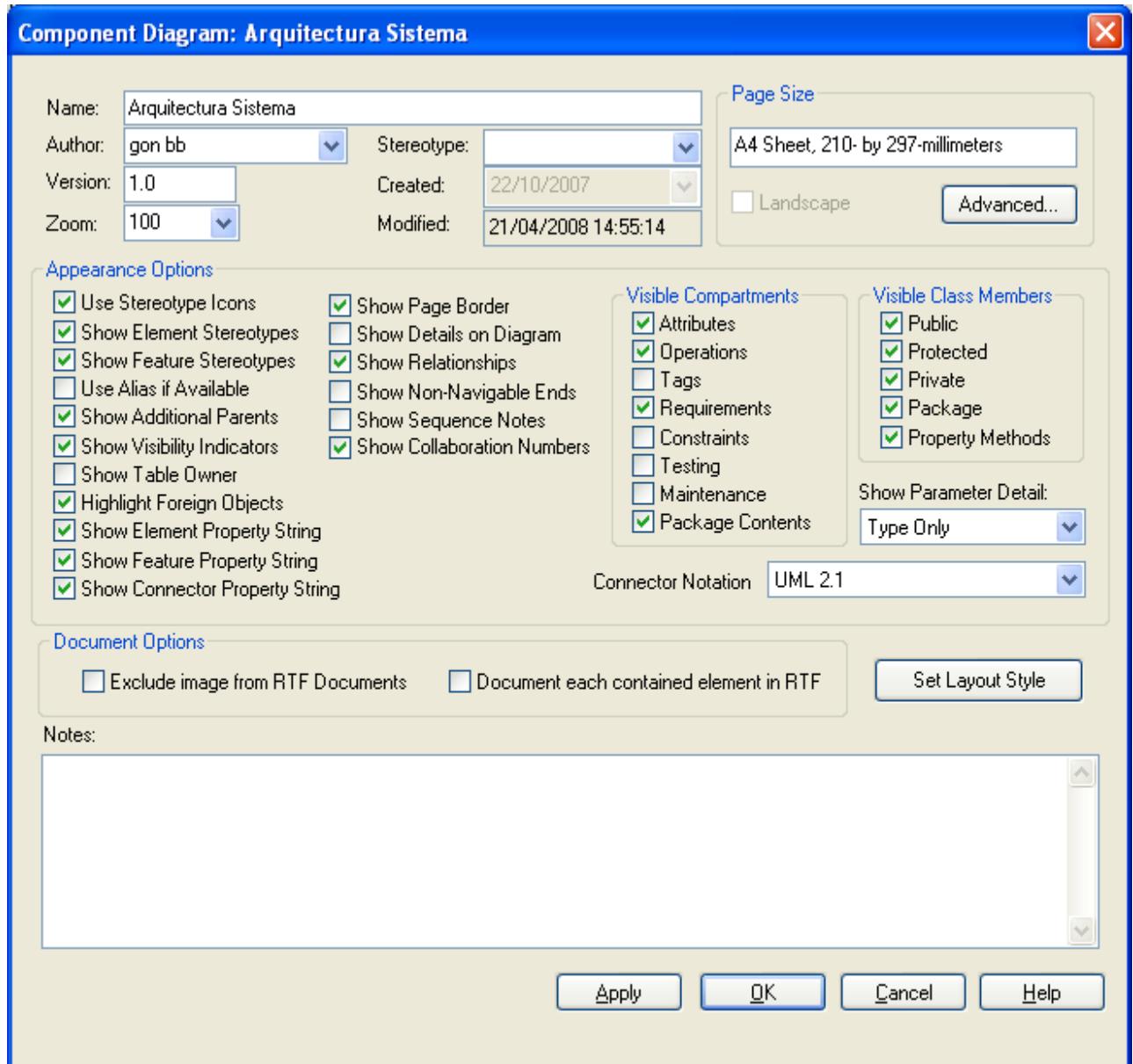
### Arquitectura Sistema

Desde la carpeta **Arquitectura Sistema**, se deberá añadir tantos paquetes como niveles de arquitectura presente la aplicación.



Una vez creados los paquetes se deberán arrastrar al diagrama y establecer sus relaciones. Por cada nivel de arquitectura se podrán agregar tantas características como se consideren necesarios (El proceso es el mismo que cuando se agregan requerimientos, punto 3.3.1)

El diagrama de **Arquitectura de Sistema** deberá presentar las siguientes características:

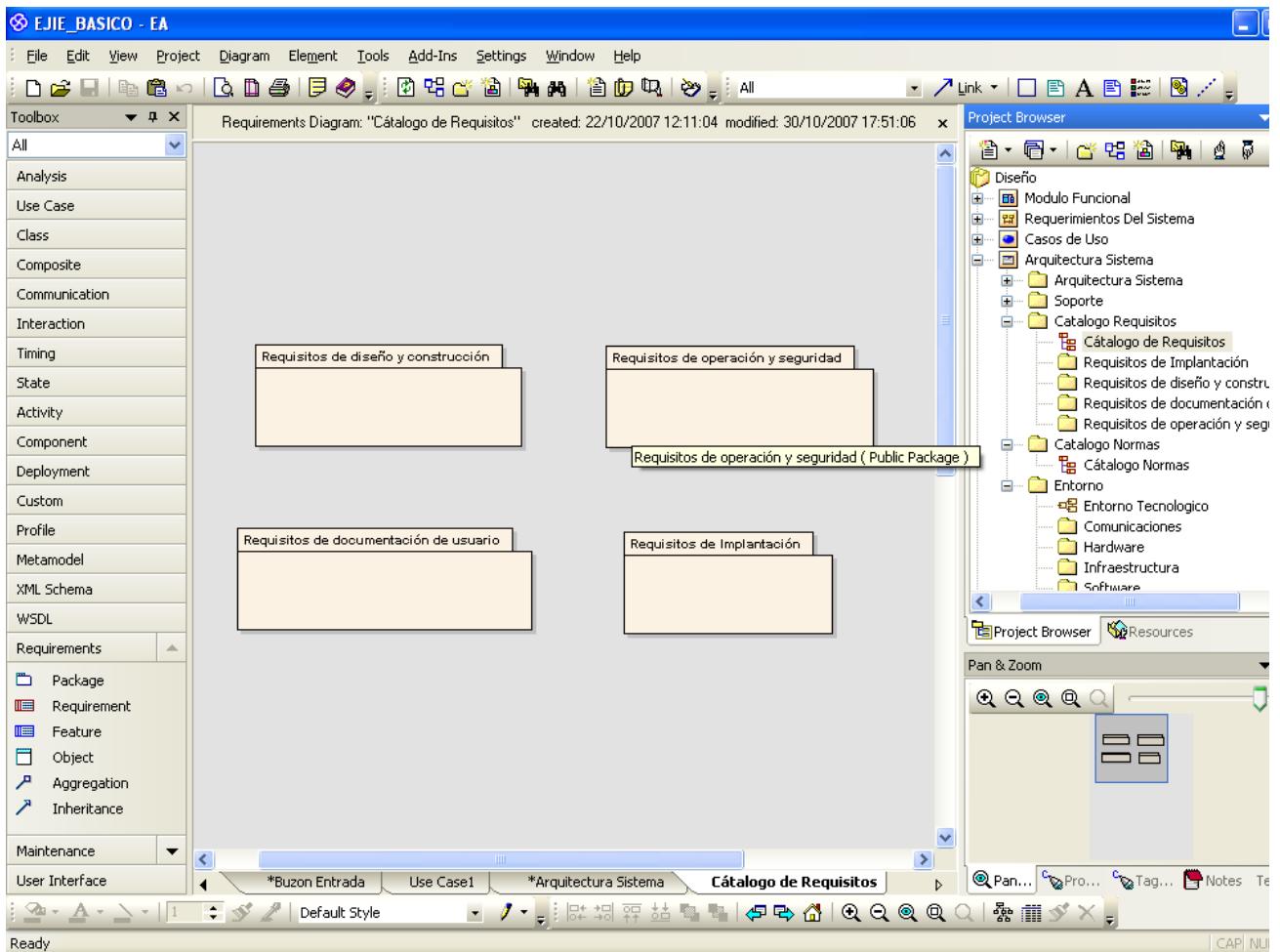


## Arquitectura Soporte

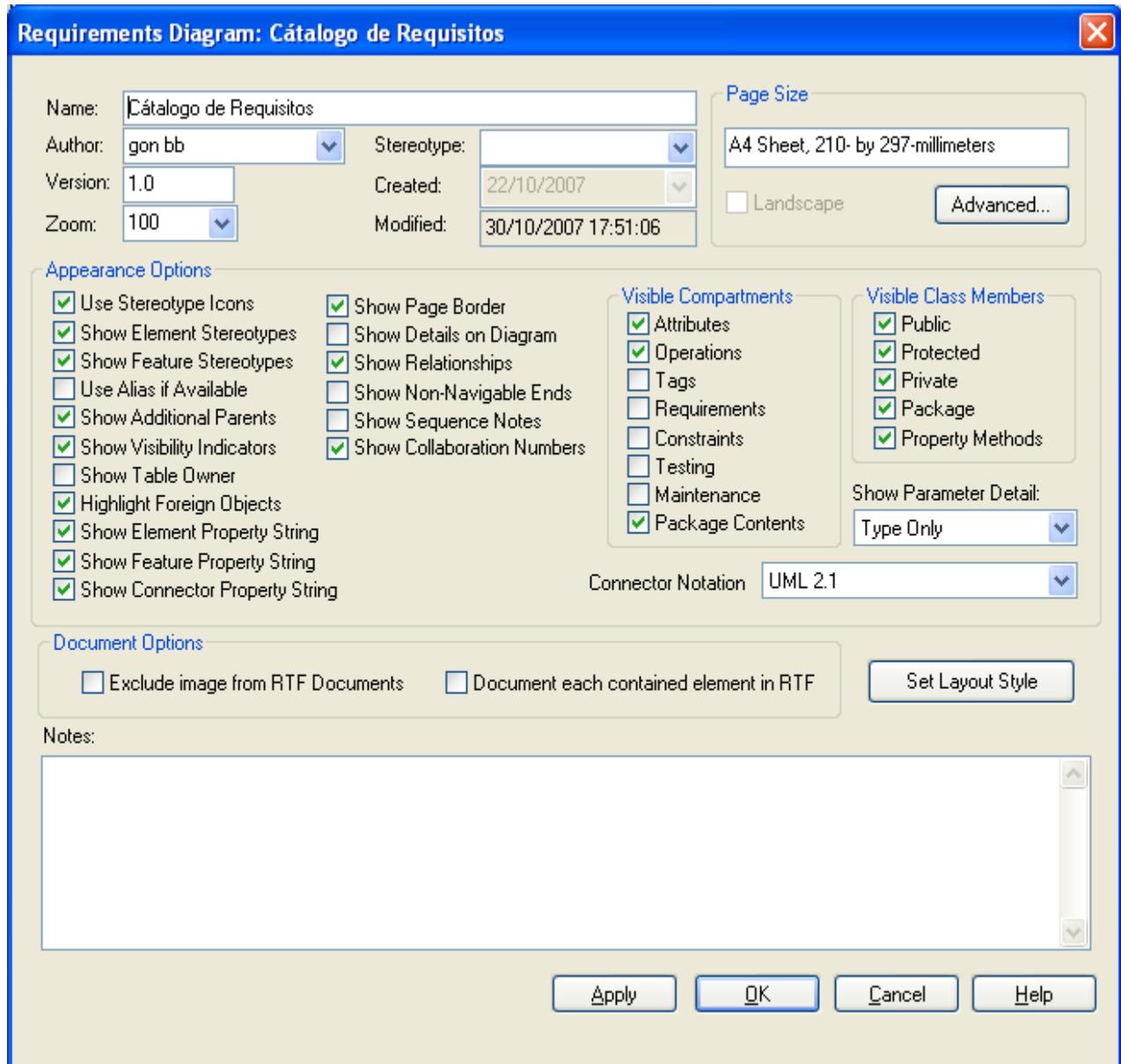
El modelado en esta sección es el mismo que el que se realiza en la **Arquitectura de Sistema**

## Catalogo Requisitos

Desde la carpeta **Catalogo Requisitos** se deberán añadir tantos elementos por paquete como sean necesarios. (El proceso es el mismo que cuando se agregan requerimientos, punto 3.3.1)

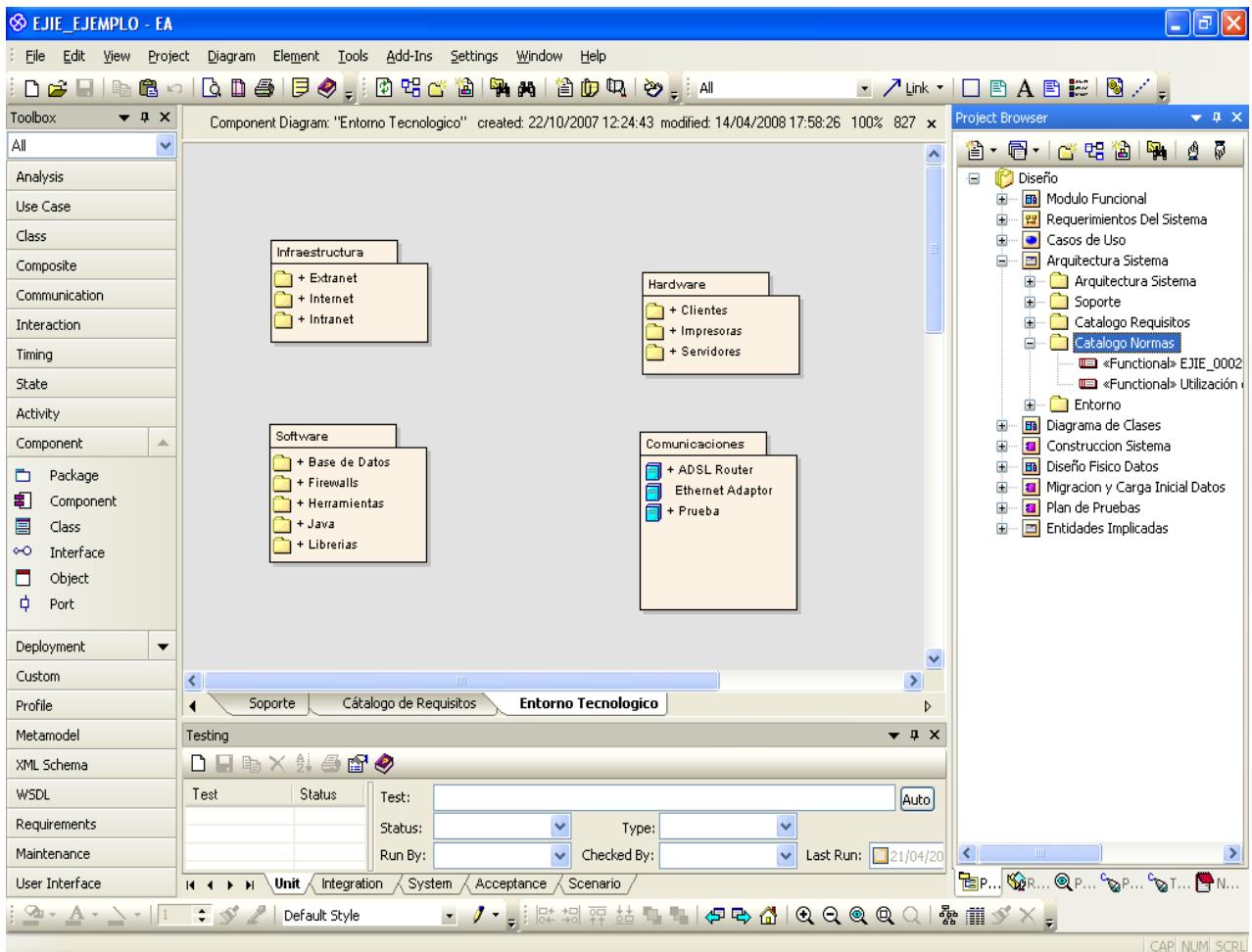


El diagrama deberá presentar la siguientes características:



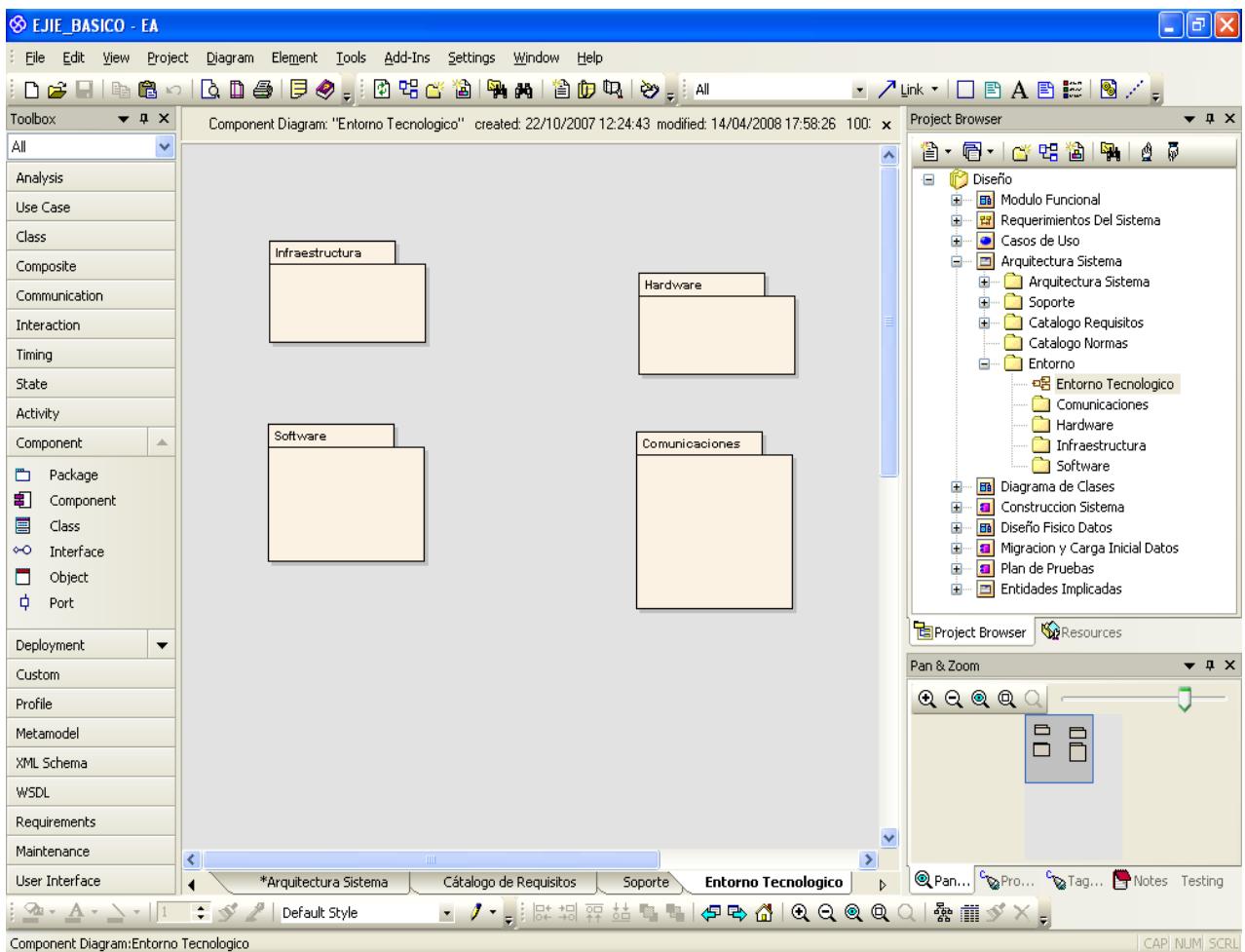
## Catalogo Normas

Desde la carpeta **Catalogo Normas** se deberán añadir tantos elementos como sean necesarios. (El proceso es el mismo que cuando se agregan requerimientos, punto 3.3.1)

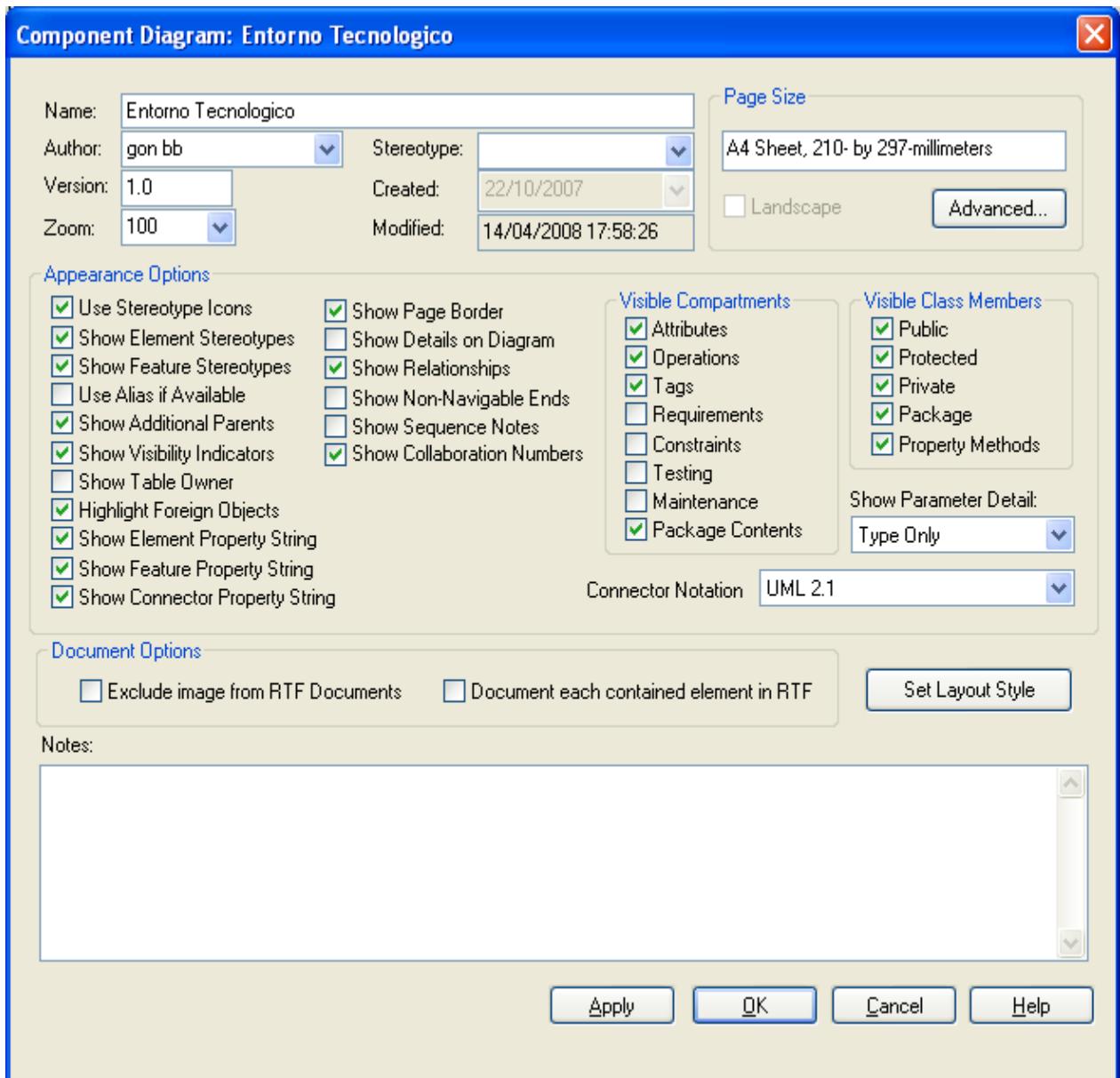


## Entorno

Desde la carpeta **Entorno** se deberán añadir tantos elementos por paquete como sean necesarios.  
(El proceso es el mismo que cuando se agregan requerimientos, punto 3.3.1)



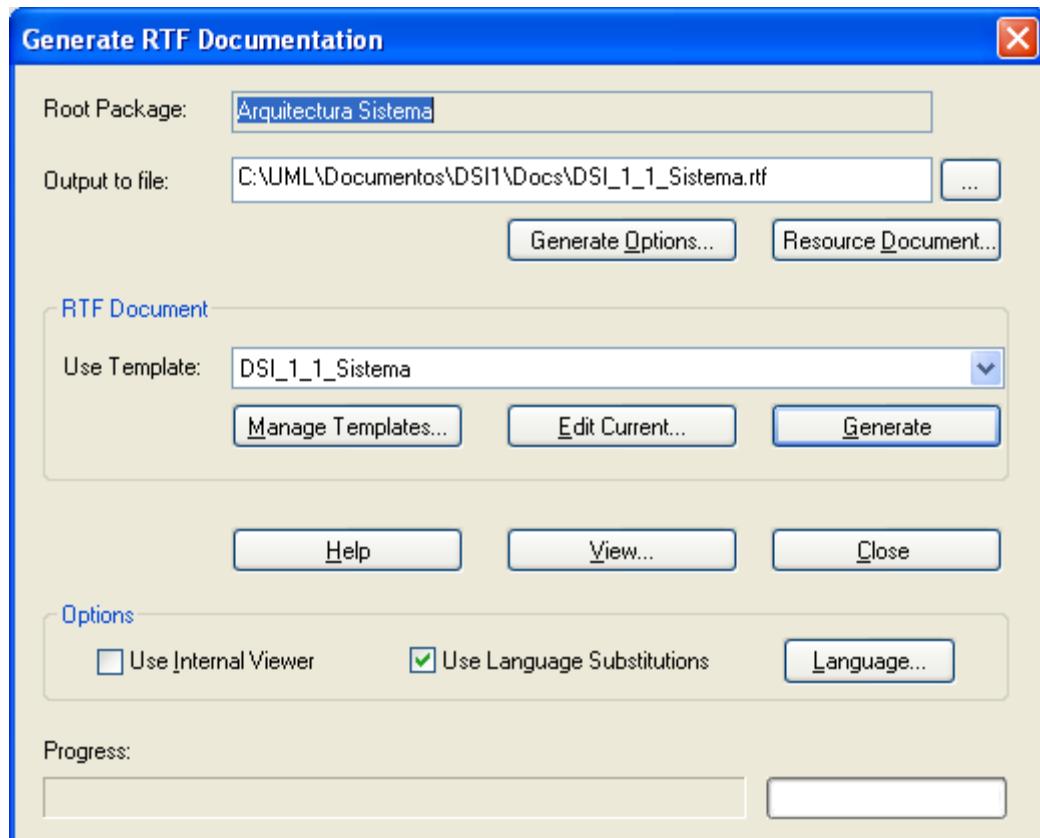
El diagrama presentara las siguientes características:



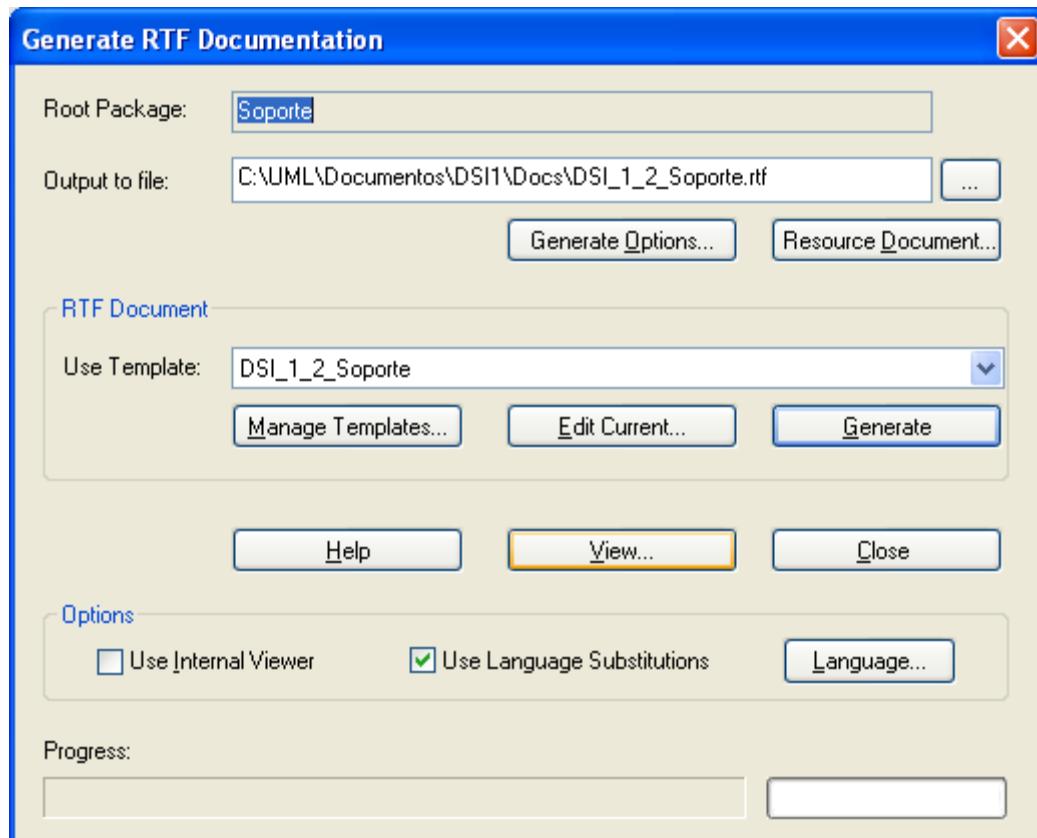
#### 5.4.2. Generación Documentación

Los accesos directos que se han de ejecutar para la creación de los documentos intermedios son las siguientes

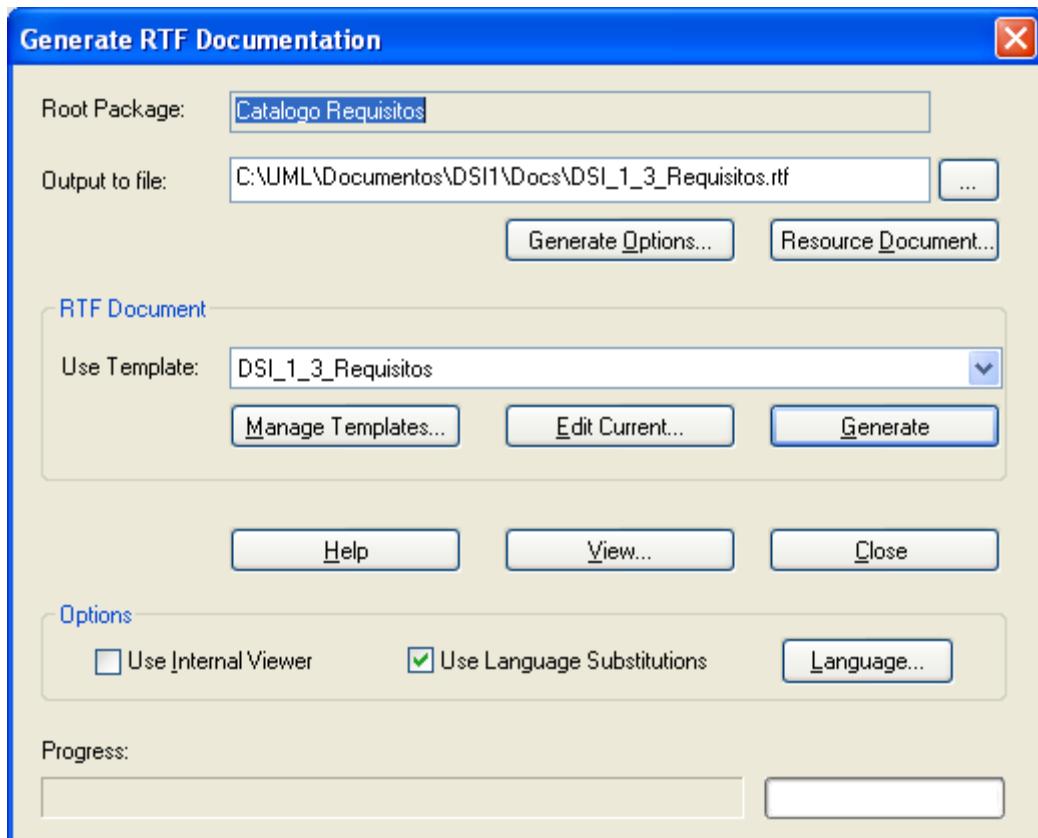
**DSI\_1\_1\_Sistema**



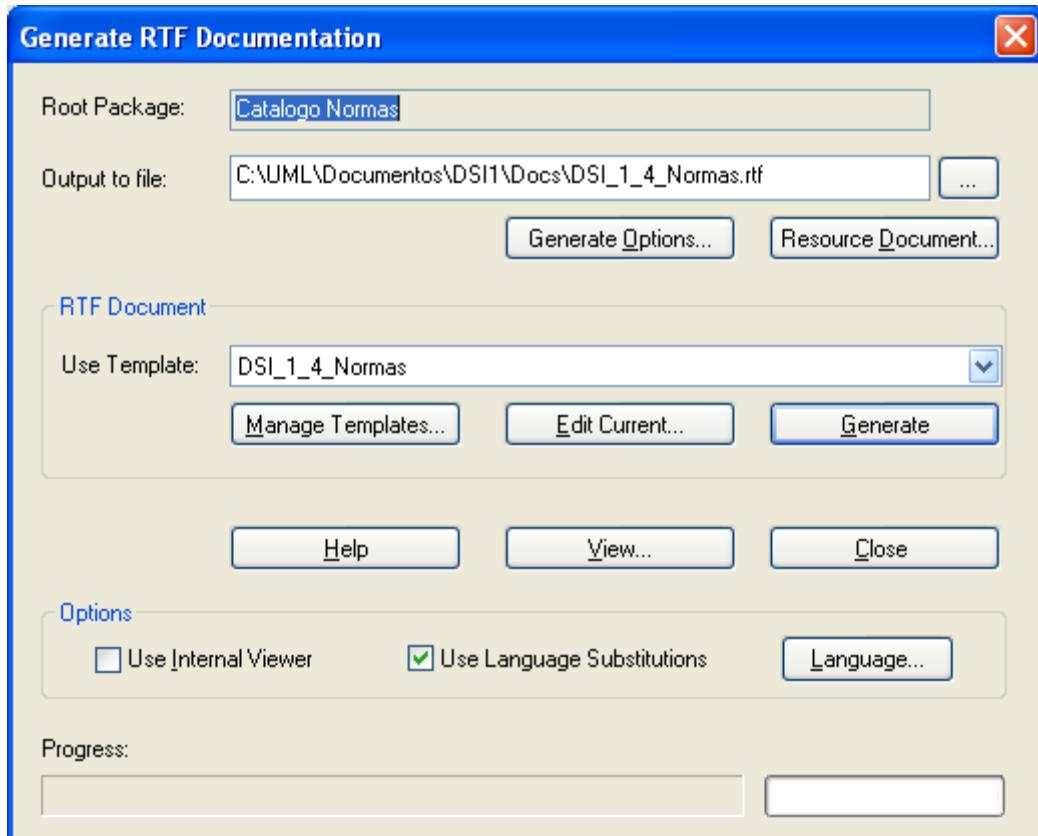
## DSI\_1\_2\_Soporte



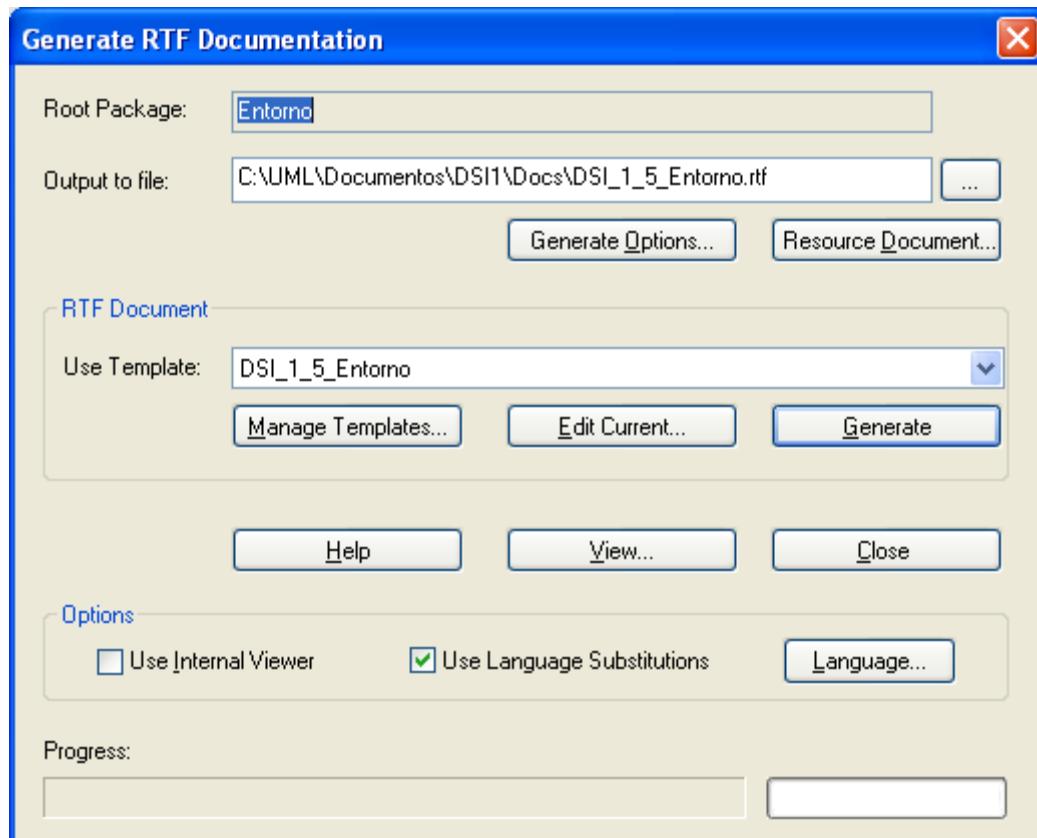
### DSI\_1\_3\_Requisitos



### DSI\_1\_4\_Normas



## DSI\_1\_5\_Entorno



## 5.5 Diseño de Casos de Uso Reales – DSI 3

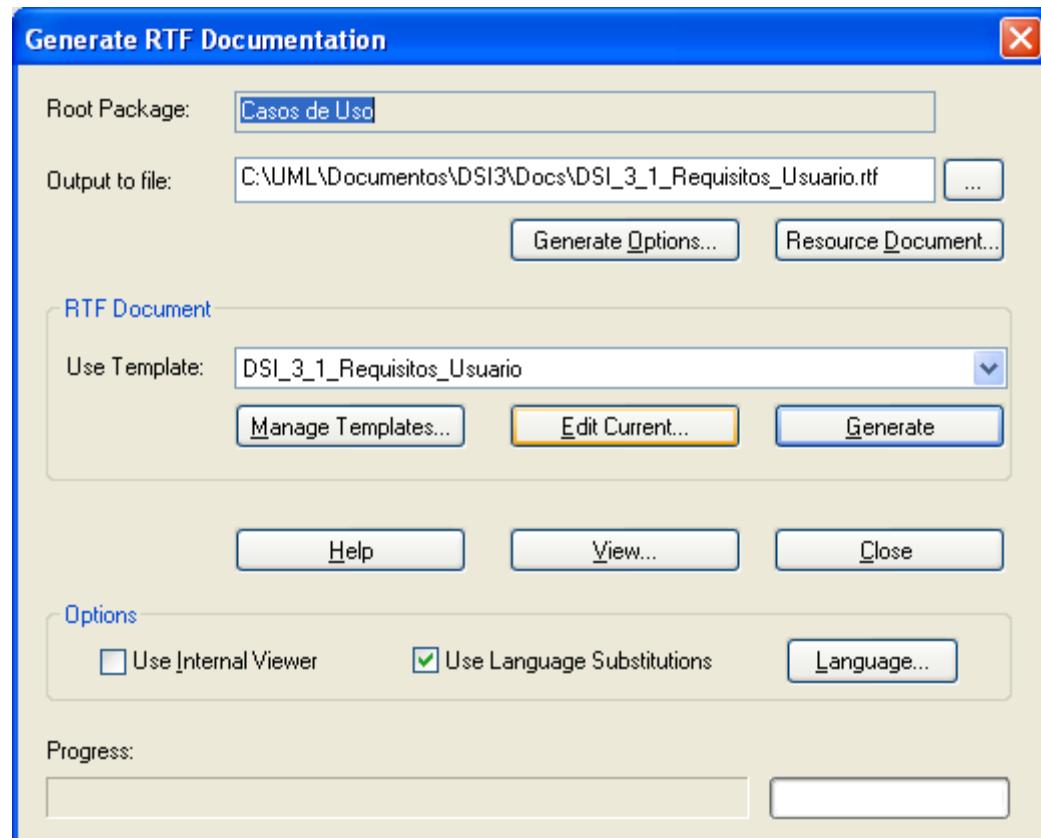
### 5.5.1. Modelado UML

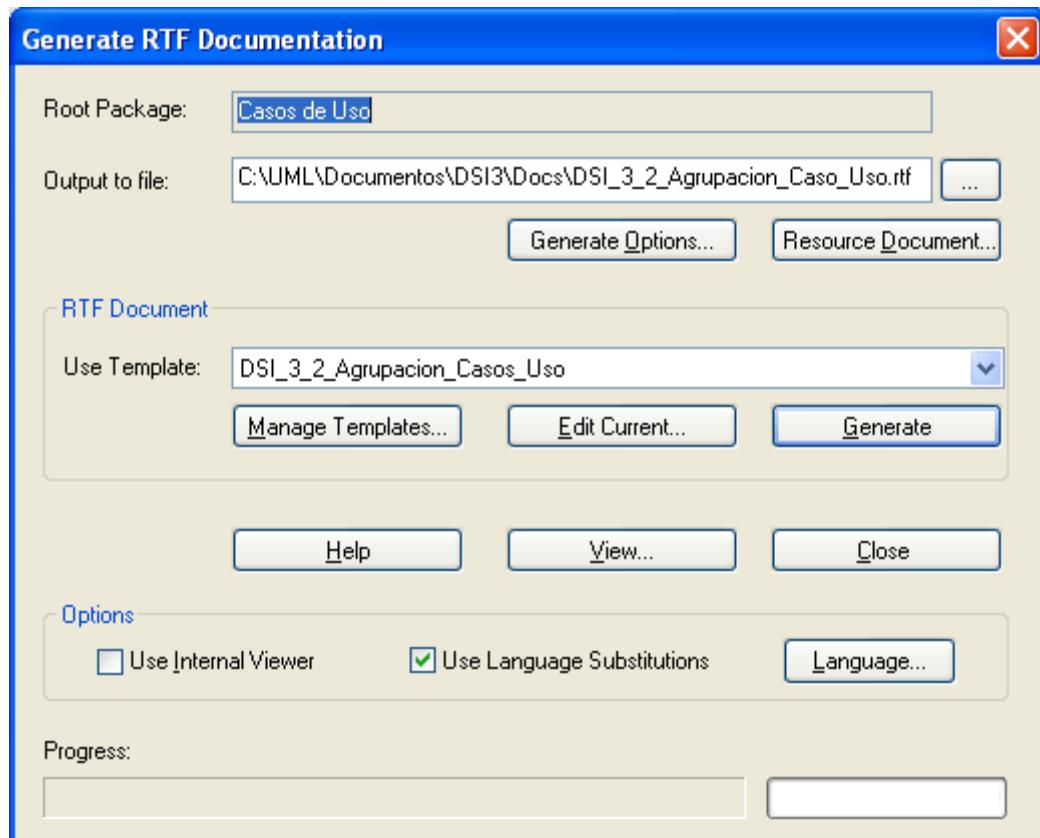
Con el modelado realizado en la fase de **Análisis del Sistema de Información**, el modelado correspondiente a la documentación del Diseño de Caso de Uso, se encuentra realizado, por lo que solo se ha de proceder a la generación de la documentación correspondiente.

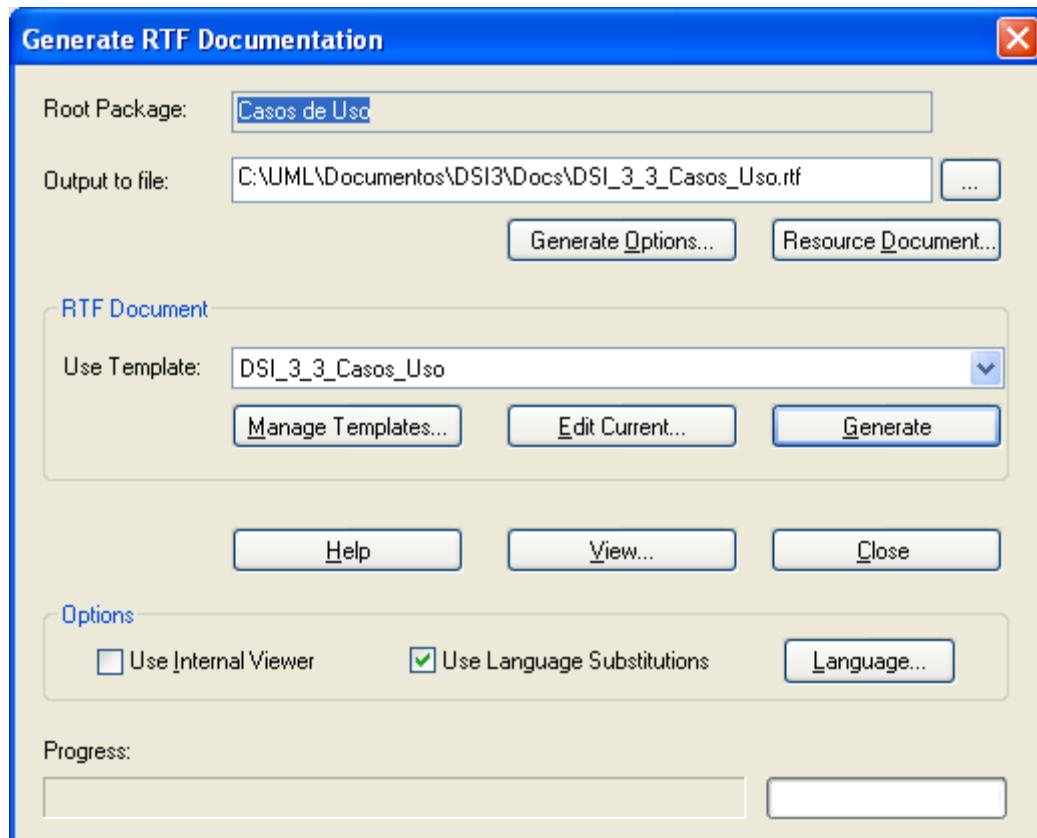
### 5.5.2. Generación Documentación

Los accesos directos que se han de ejecutar para la creación de los documentos intermedios son las siguientes

**DSI\_3\_1\_Requisitos\_Usuario**

**DSI\_3\_1\_Agrupacion\_Caso\_Uso**

**DSI\_3\_3\_Casos\_Uso**



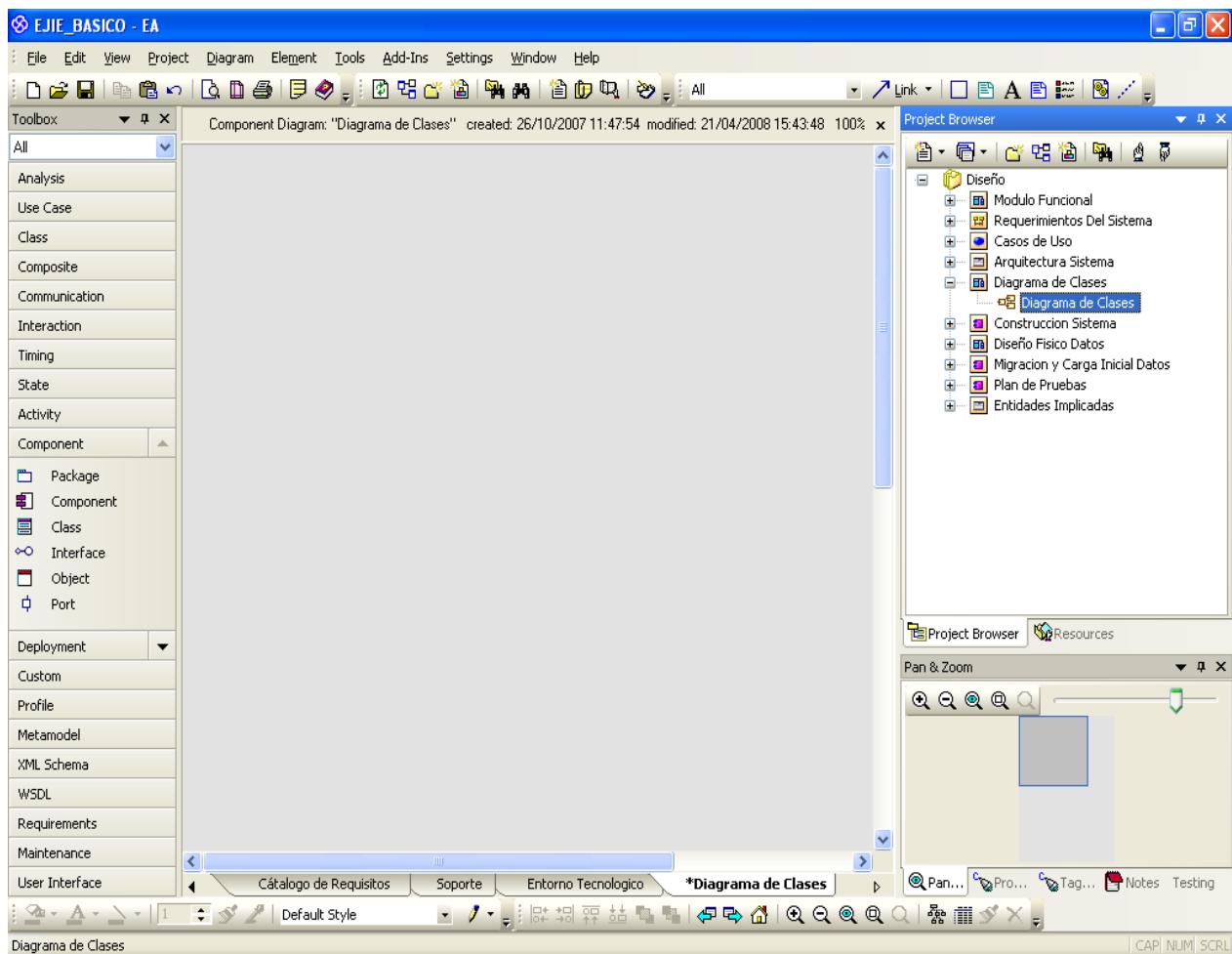
## 5.6 Diseño de Clases – DSI 4

### 5.6.1. Modelado UML

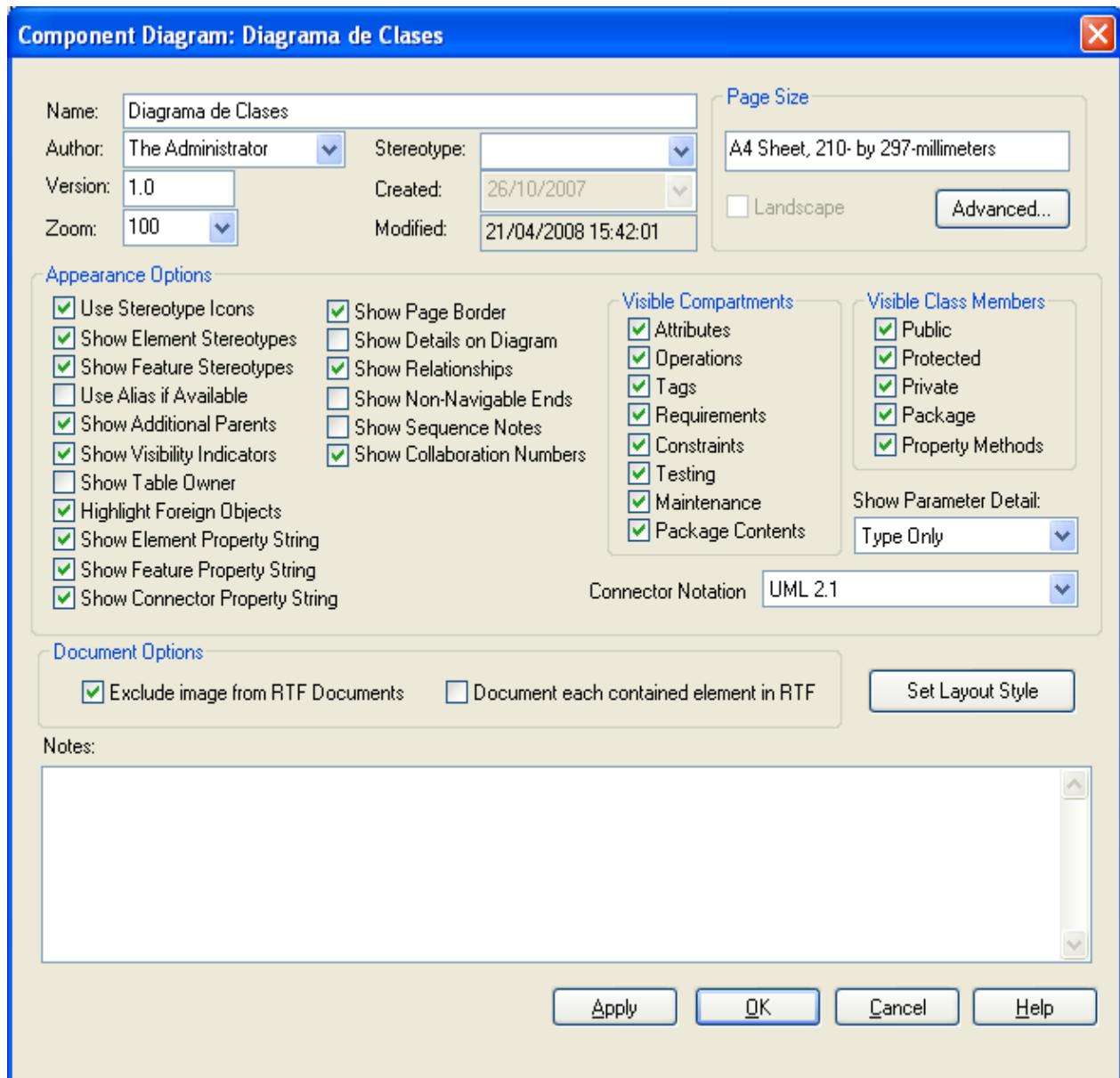
Para la generación de la documentación, se han de actualizar de la estructura estándar el siguiente diagrama

- **Diagrama de Clases**

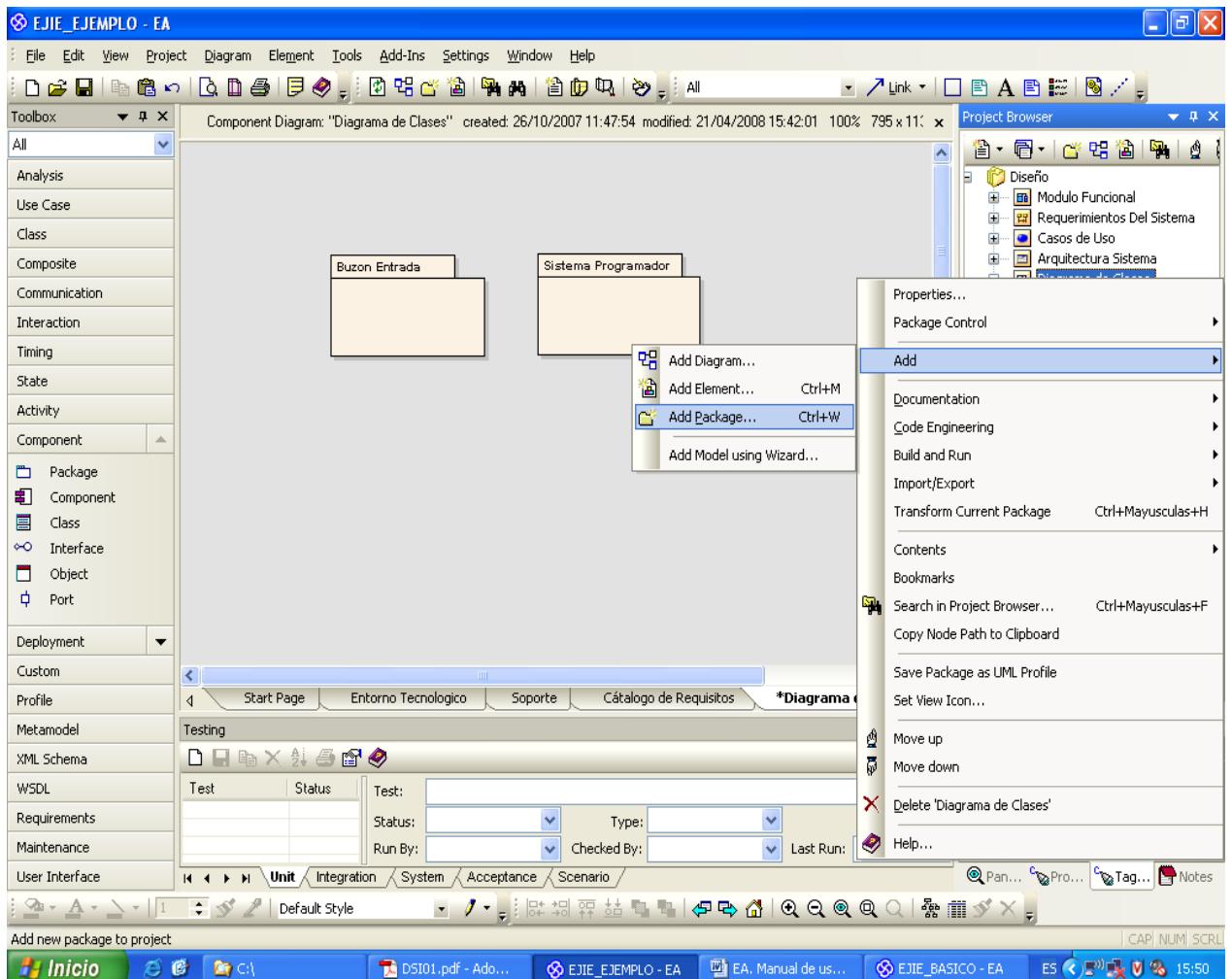
La estructura que se proporciona es la siguiente:



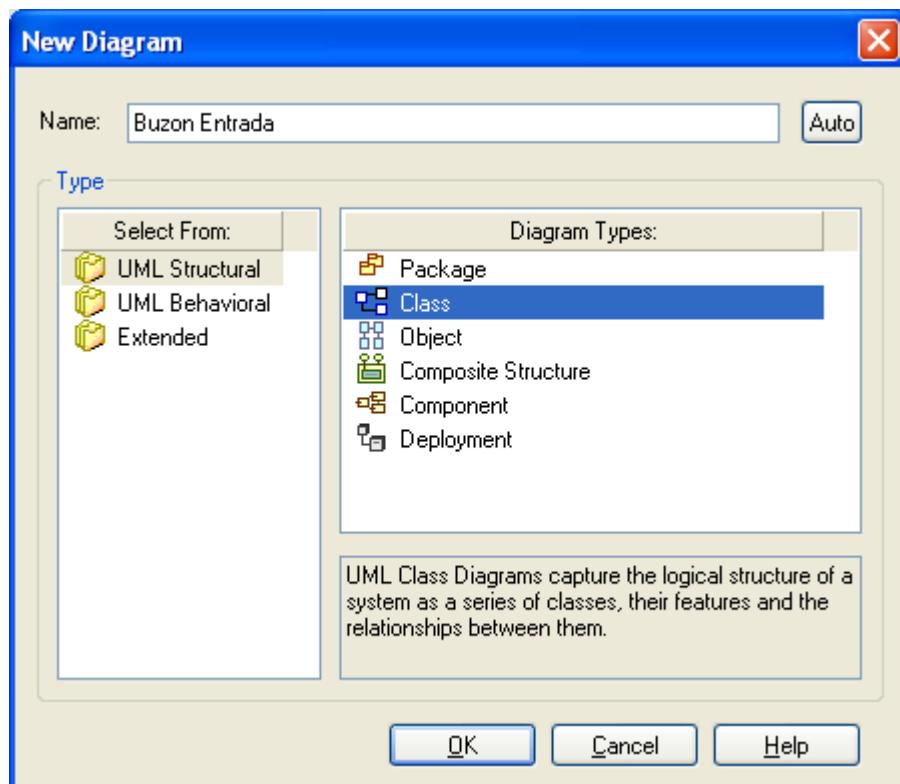
El **Diagrama de Clases** presentara las siguientes características:



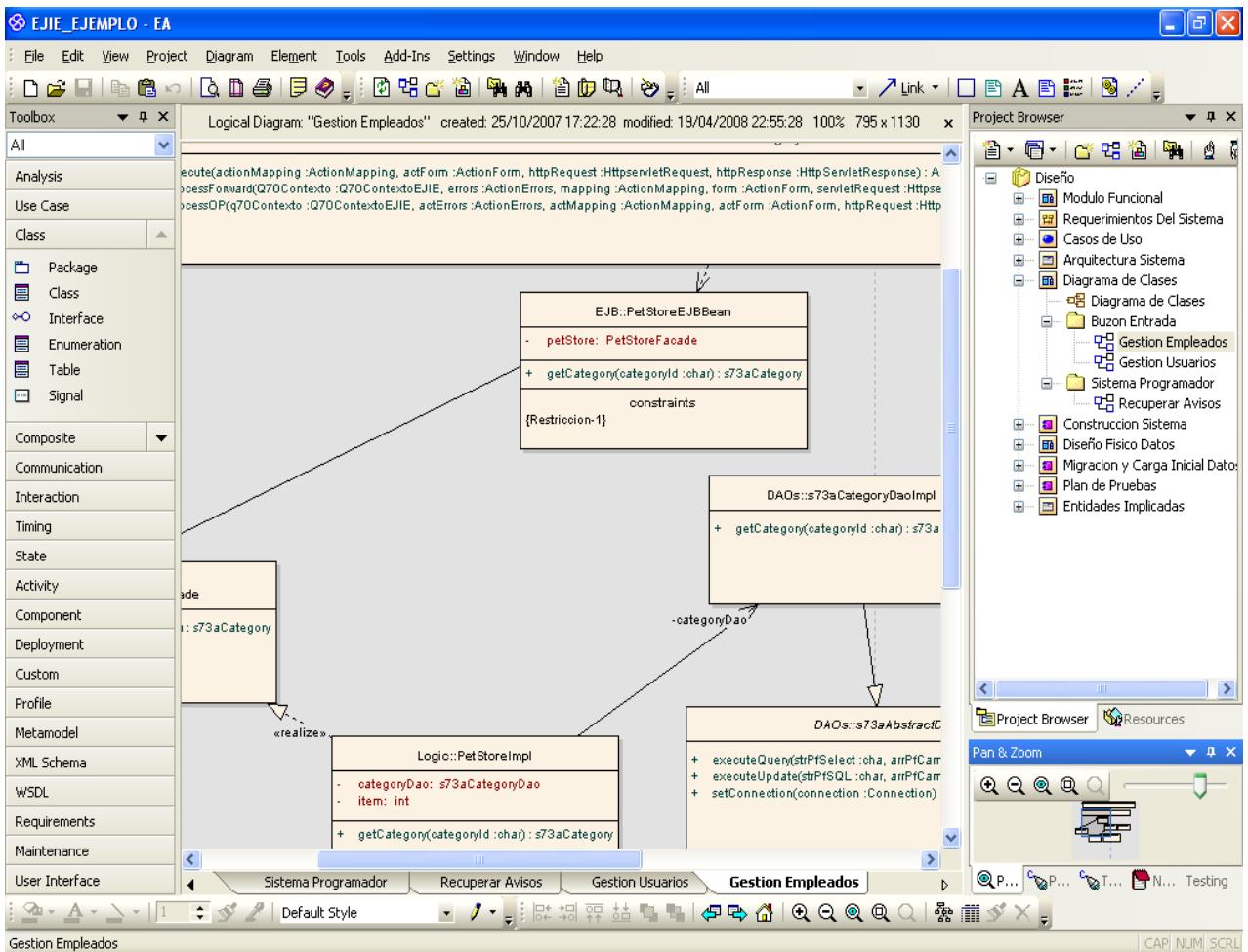
Desde la carpeta **Diagrama de Clases**, se deberá añadir tantos paquetes como módulos funcionales se hayan definido:



Una vez creados los paquetes funcionales se deberán crear tantos diagramas de clases como casos de uso se hayan definido en el modulo **Casos de Uso** por paquete funcional, de tal forma que cada diagrama tenga como nombre, el del caso de uso al que hace referencia. Para ello se seleccionara una de las carpetas y a través del menú emergente que se muestra con el botón derecho se pulsar **Add-> Diagram**

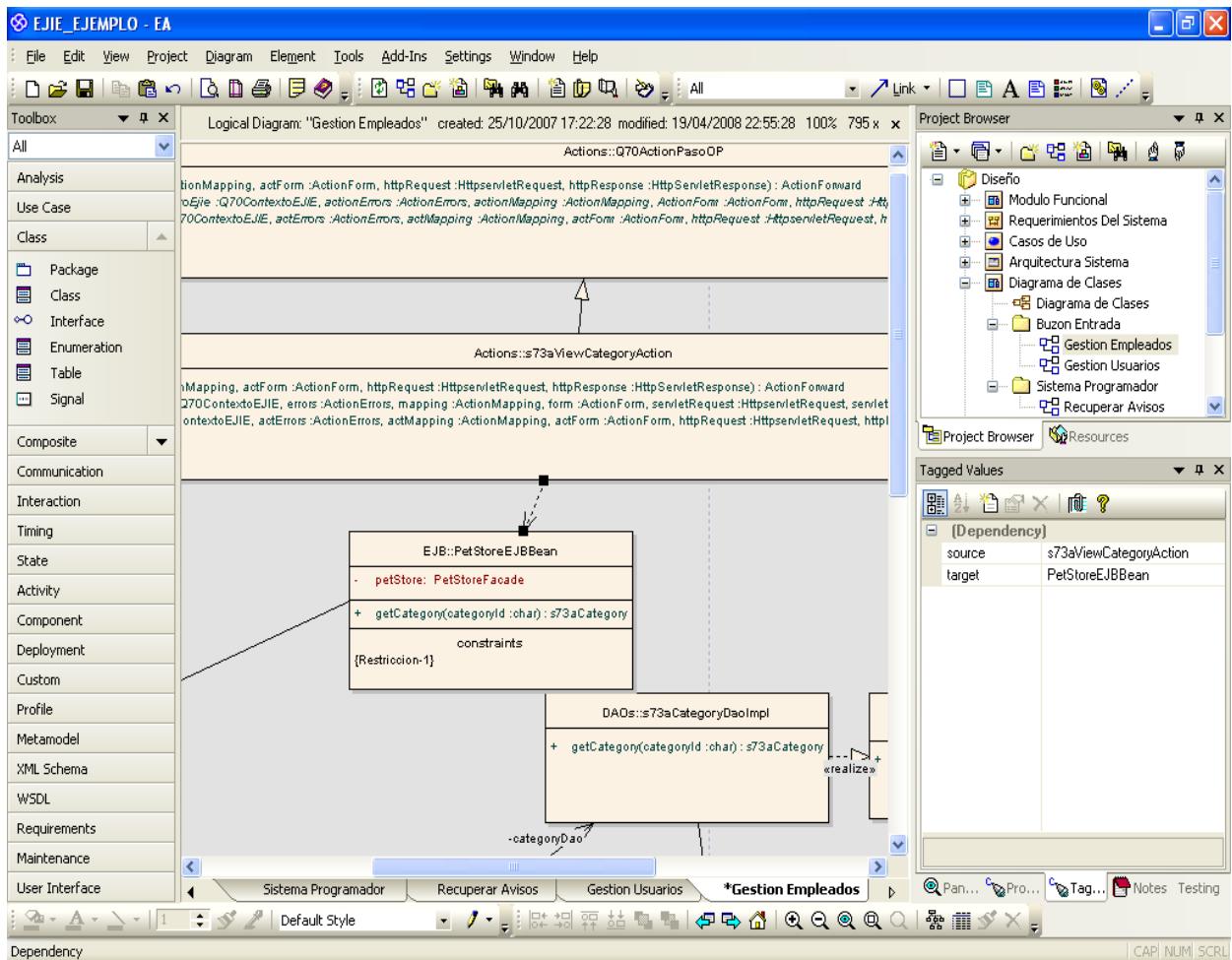


A continuación y una vez creados los diagramas a implementar se procede al diseño del diagrama de clases.

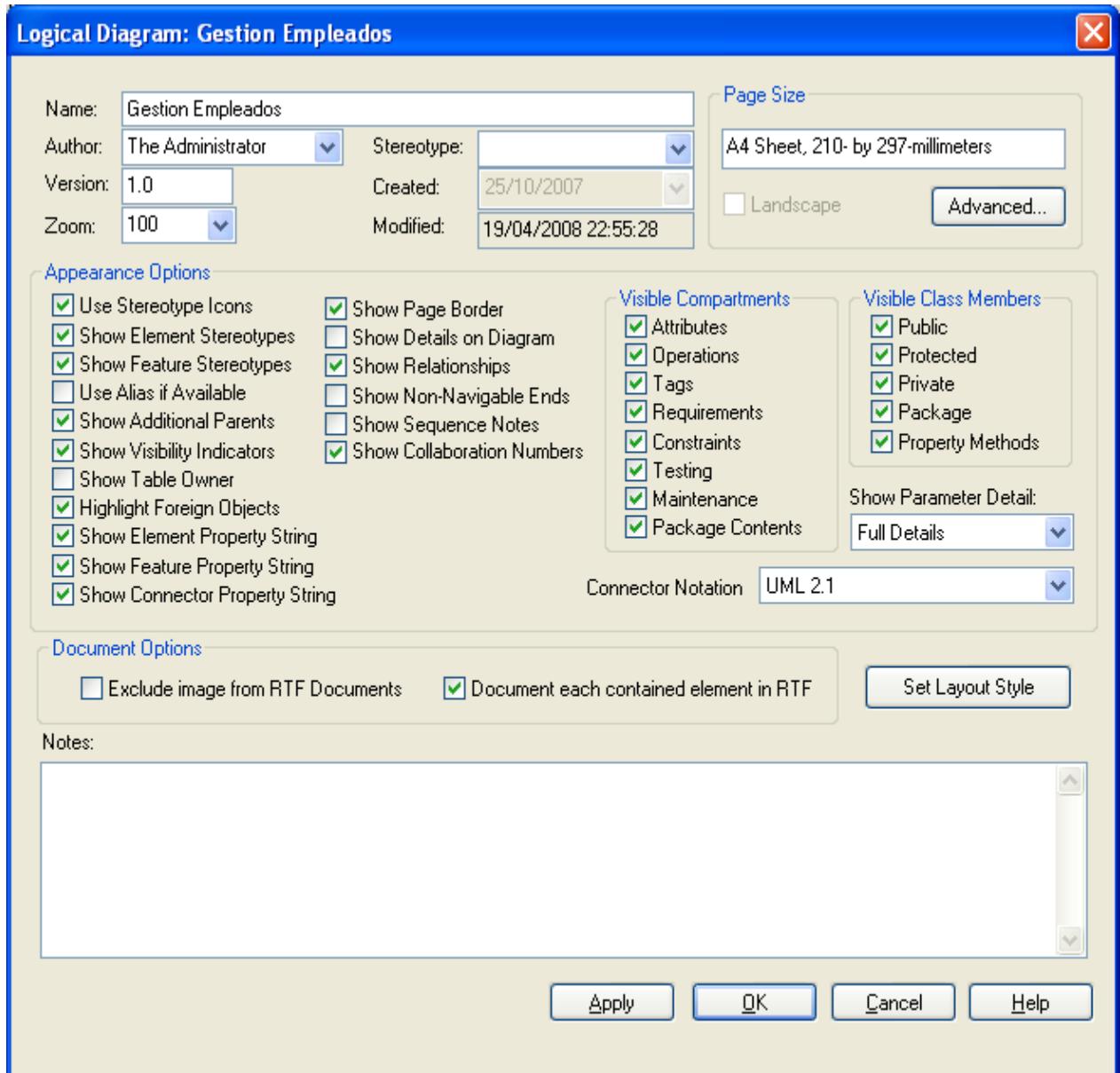


Una vez establecidas las relaciones entre las distintas clases, por cada una de las relaciones establecidas se deberán crear dos **Tagged-Values**,(Punto 3.8.1) :

- **source** -> Nombre de la clase origen de la relación
- **target** -> Nombre de la clase destino de la relación.



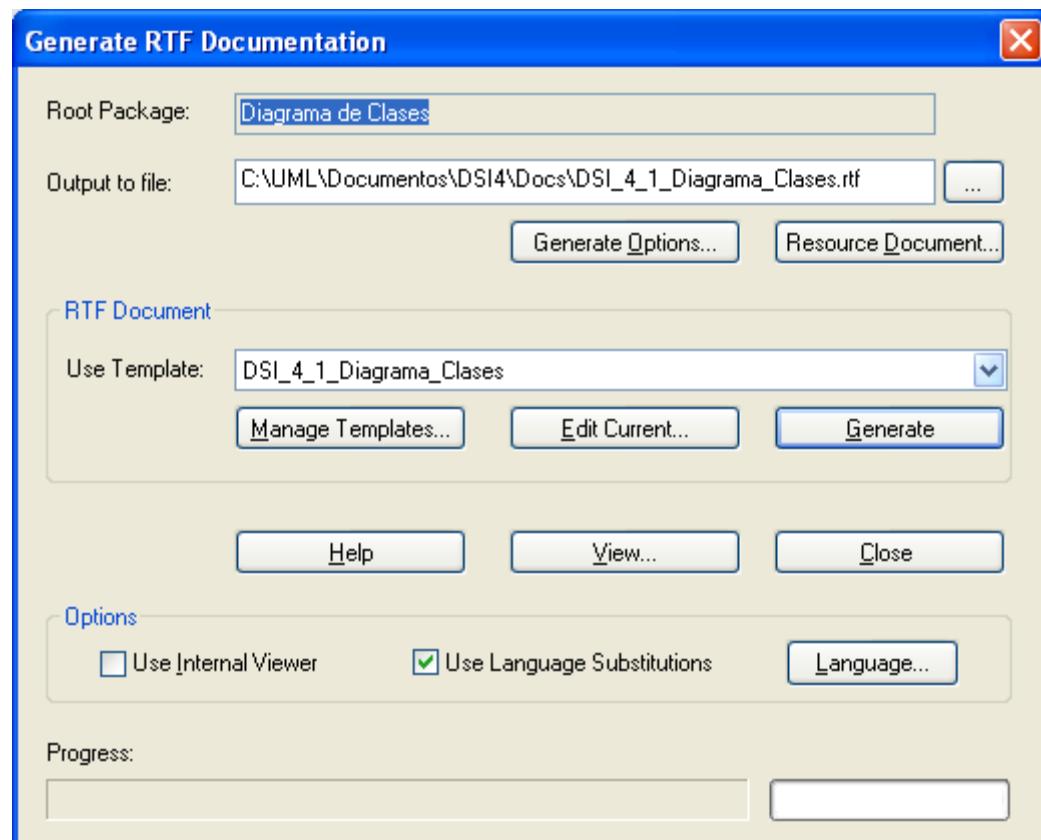
Cada uno de los diagramas creados deberá presentar las siguientes características:



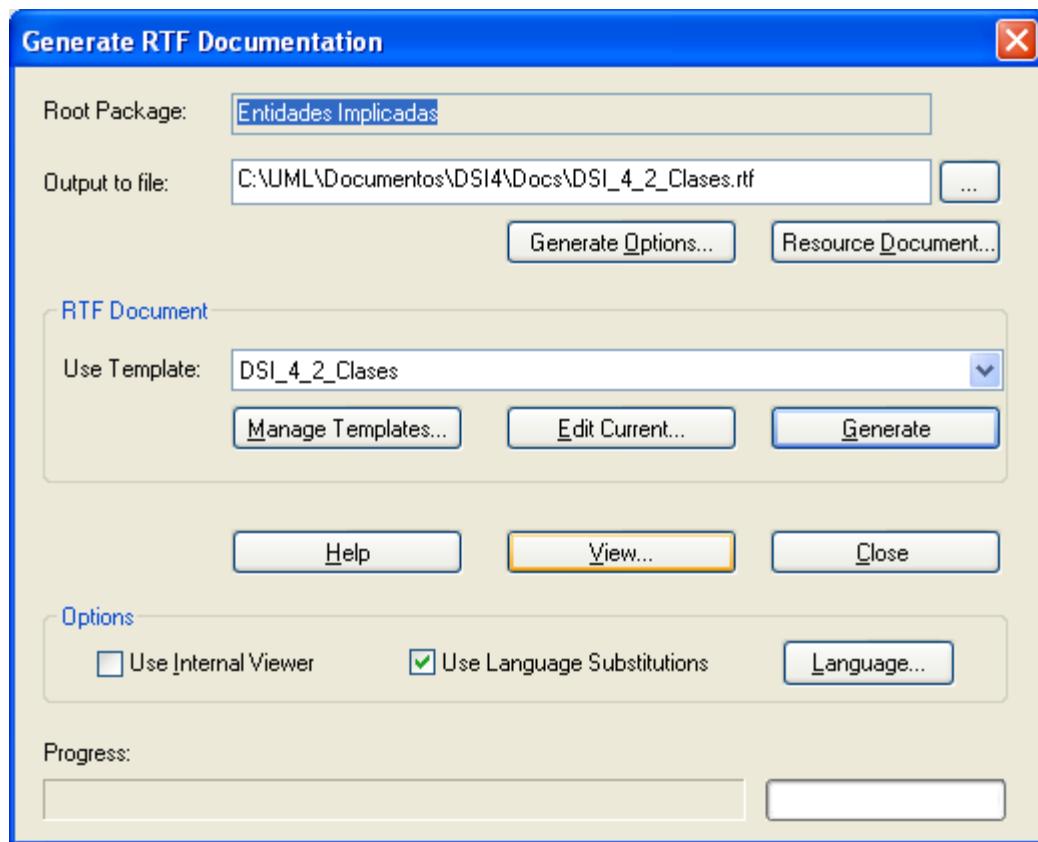
### 5.6.2. Generación Documentación

Los accesos directos que se han de ejecutar para la creación de los documentos intermedios son las siguientes

#### DSI\_4\_1\_Diagrama\_Clases



## DSI\_4\_2\_ Clases



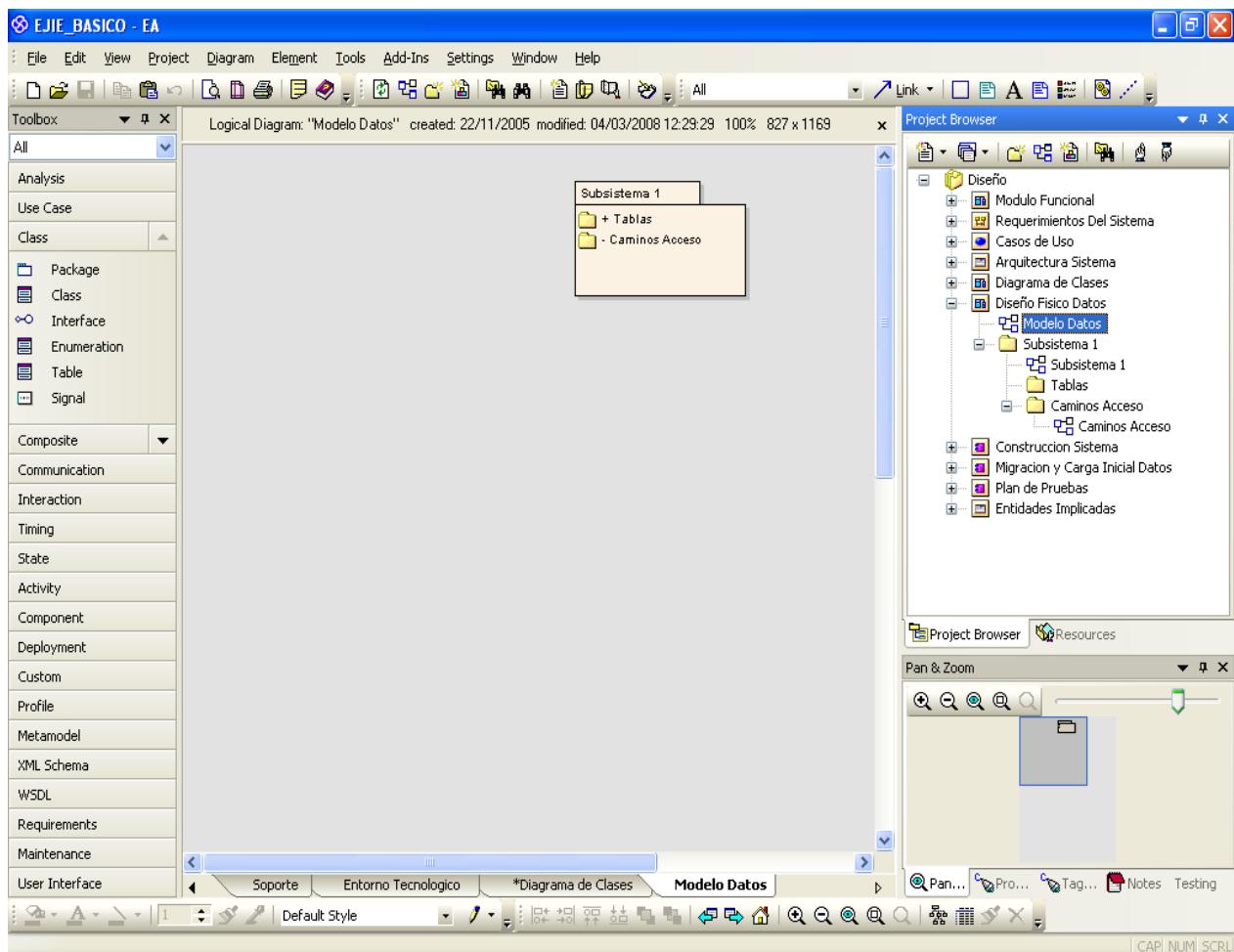
## 5.7 Diseño Físico de Datos – DSI 6

### 5.7.1. Modelado UML

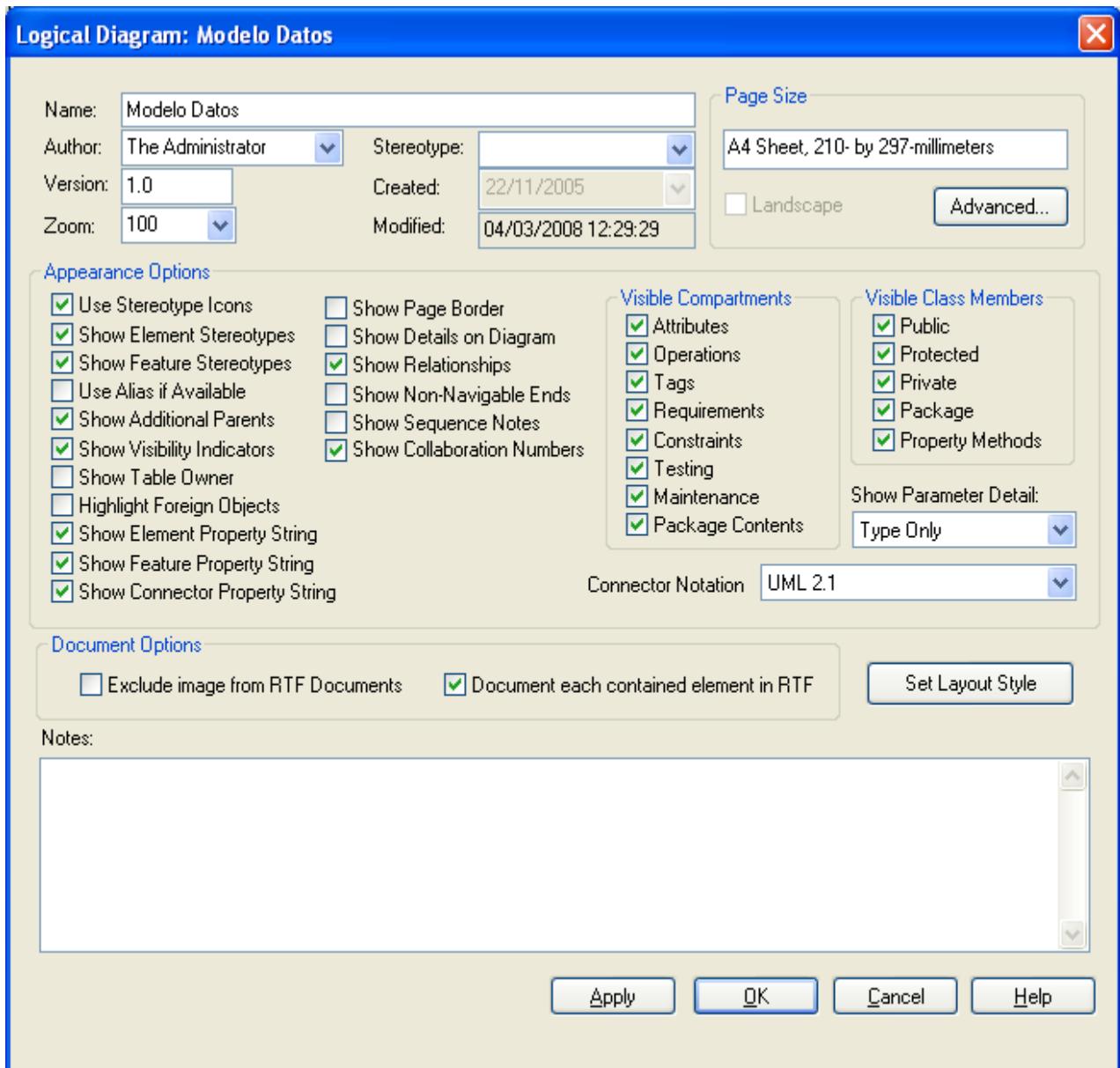
Para la generación de la documentación, se han de actualizar de la estructura estándar el siguiente diagrama

- **Diseño Físico de Datos**

La estructura que se proporciona es la siguiente:

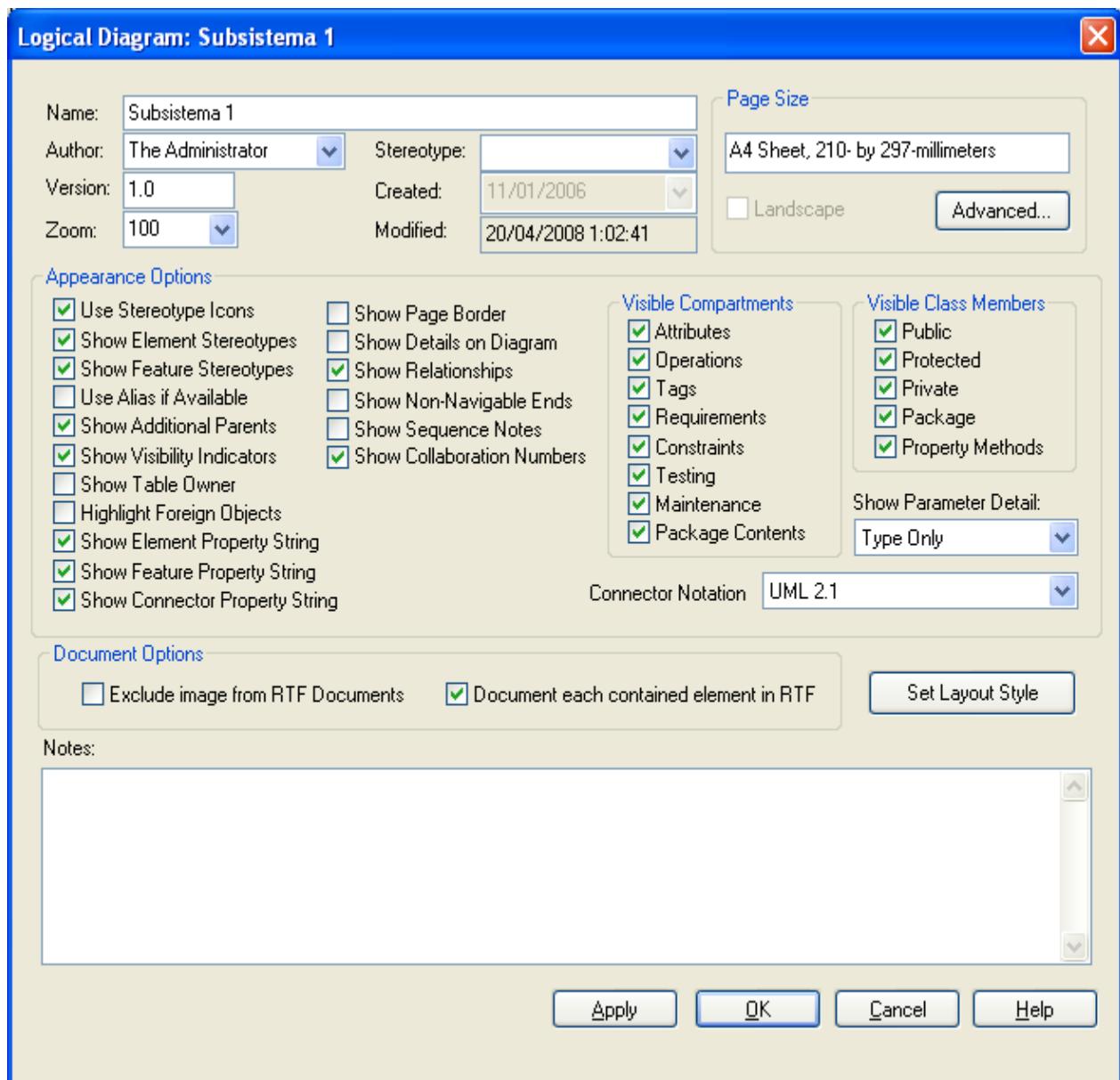


El diagrama **Modelo Datos** presentara las siguientes características:

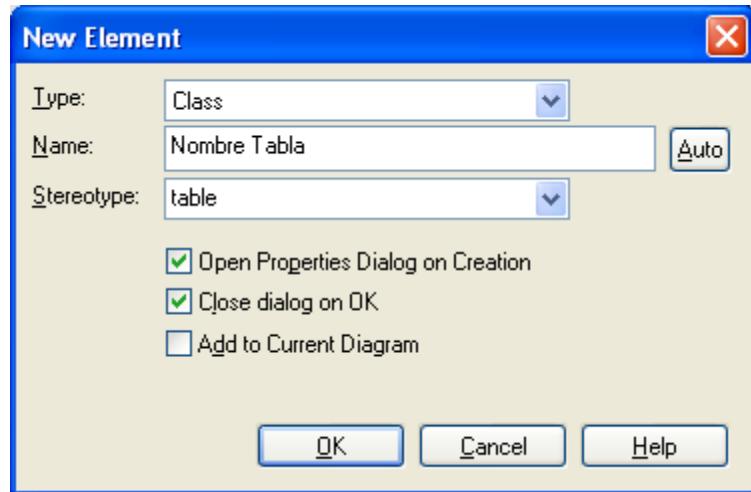


Tanto la carpeta **Subsistema 1** como su diagrama correspondiente, deberá ser modificada y asignarle el nombre del Sistema de Datos de la aplicación. En caso de existir más de un Sistema se deberá duplicar la estructura proporcionada.

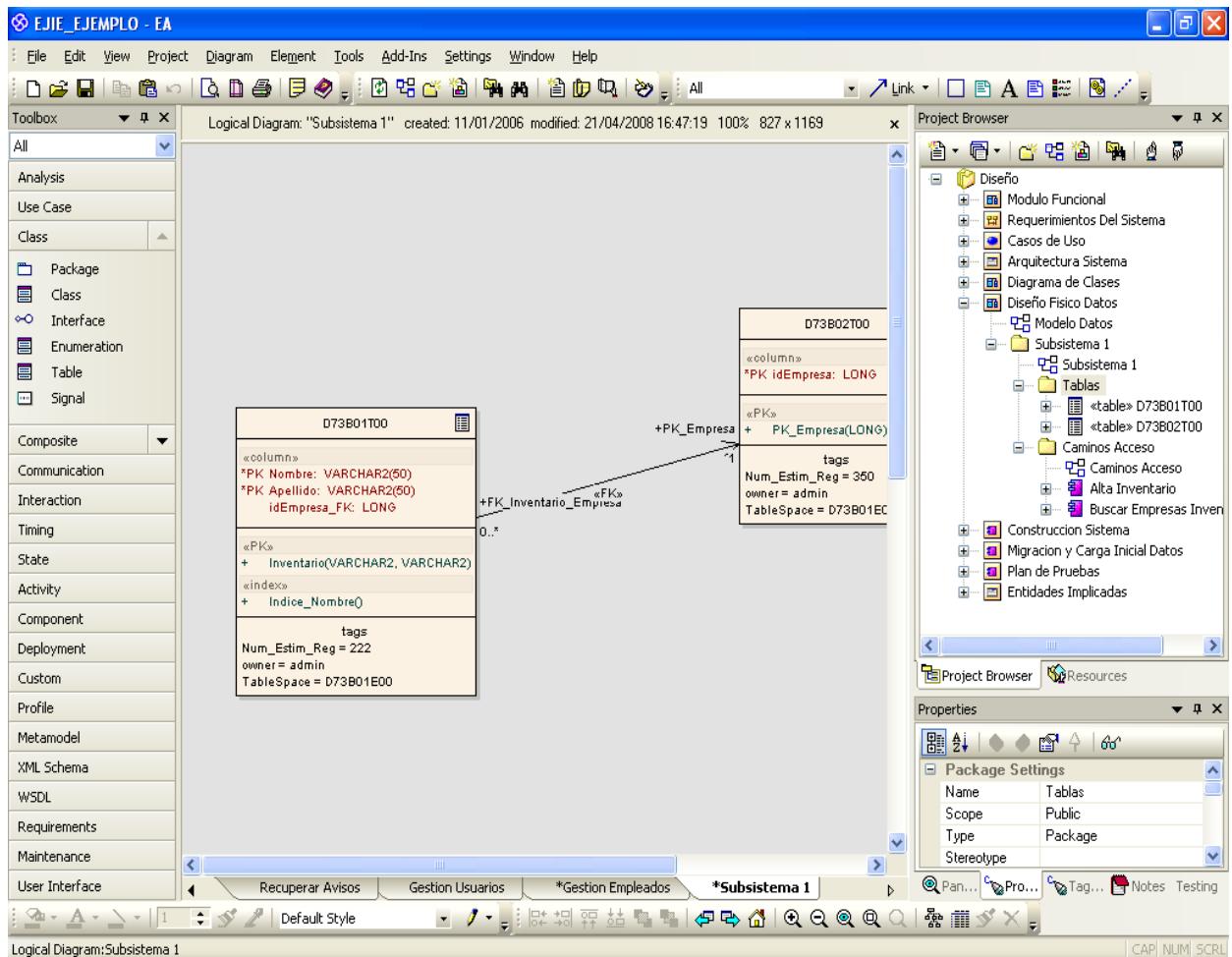
El **Diagrama Subsistema 1** presentara las siguientes características:



Para agregar tablas al diseño el primer punto consiste en añadir a la carpeta **Tablas**, tantos objetos de tipo **Class/Table**, como tablas forman parte del sistema, para ello con el botón derecho del ratón y desde la carpeta tablas se pulsar la opción **Add-> Element**



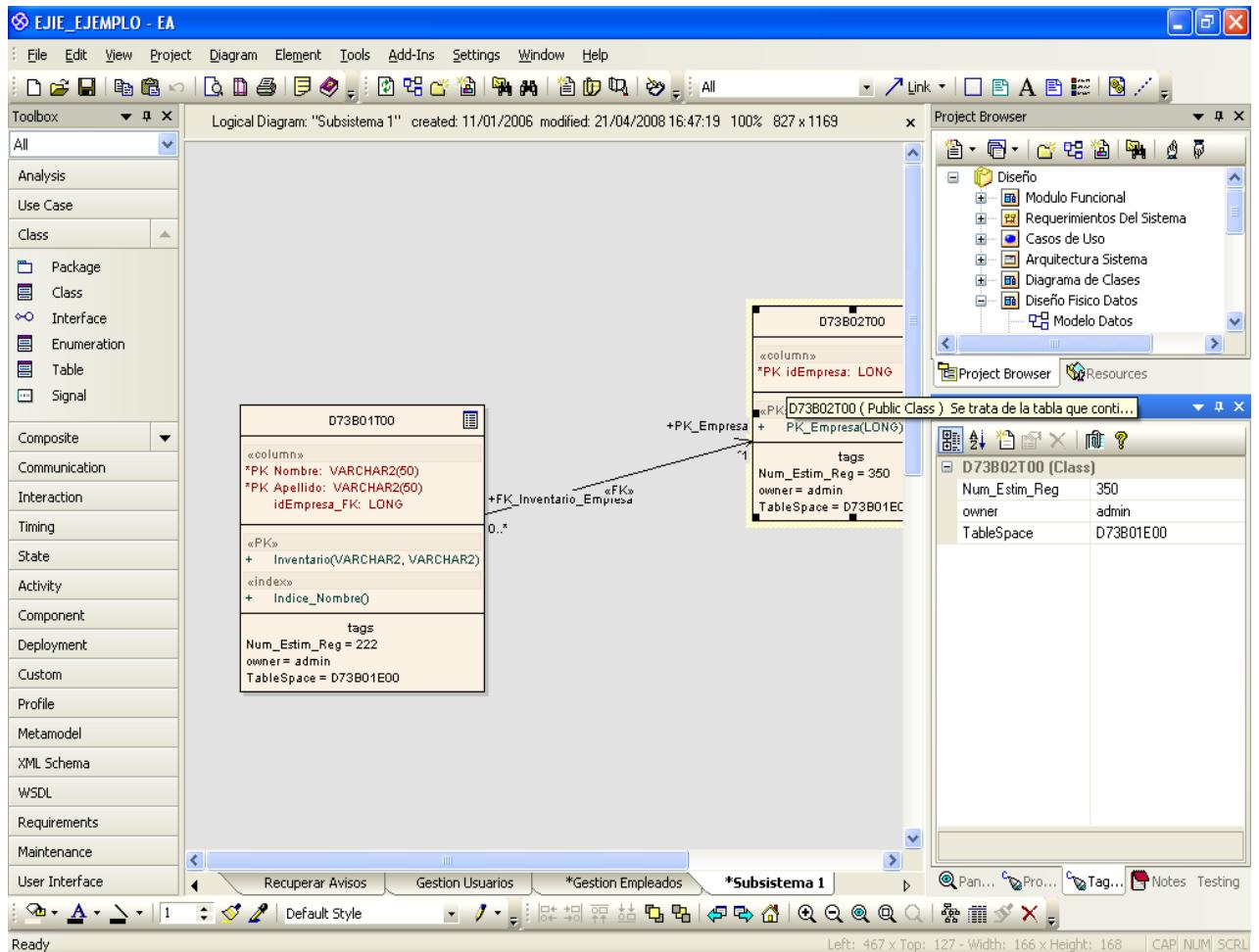
Una vez creadas las tablas y sus atributos y métodos (Punto 3.8), se procederá al modelado del diagrama de entidad relación.



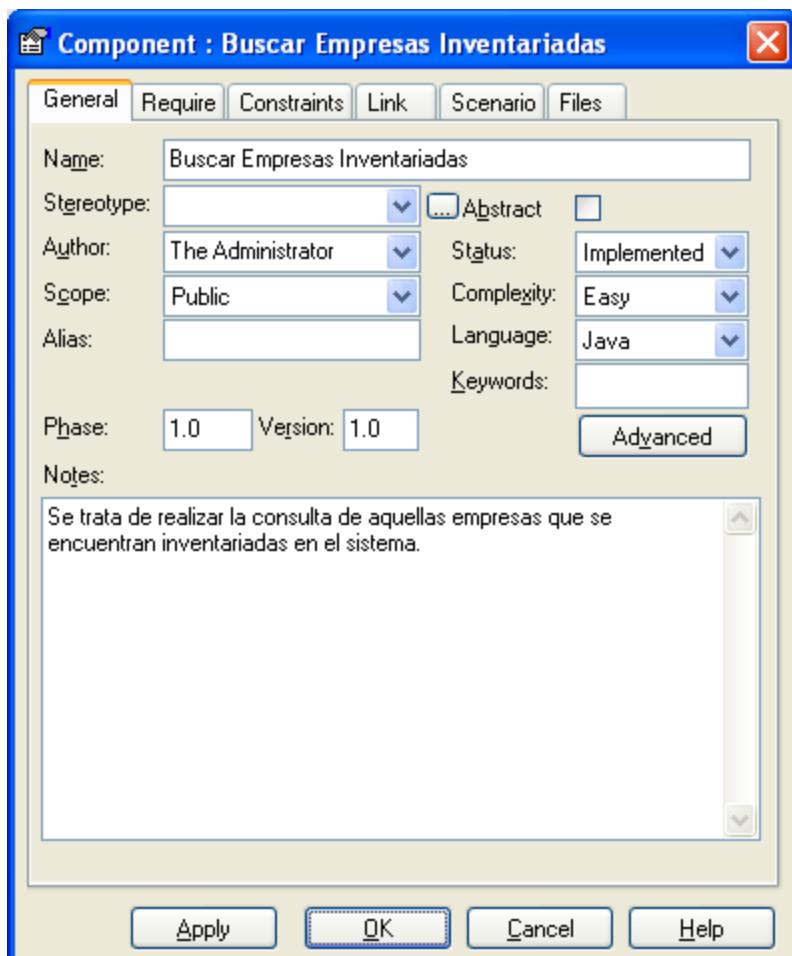
## El proceso de creación

Por cada tabla añadida se deberán crear los siguientes **Tagged-Values**

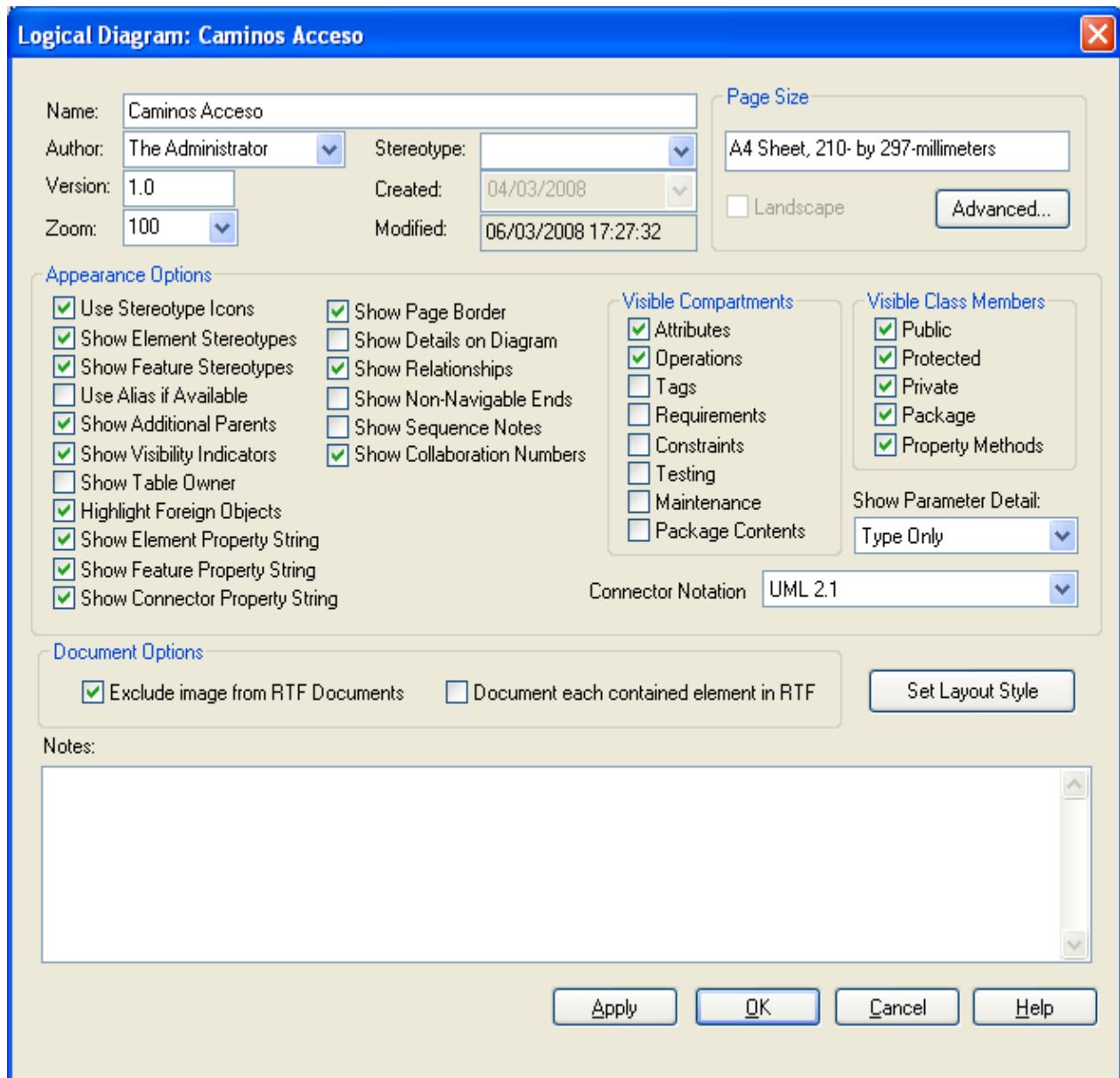
- **Num\_Estim\_Reg** -> Número de registros estimados
- **Owner** -> Propietario de la tabla
- **TableSpace** -> Nombre del tablespace



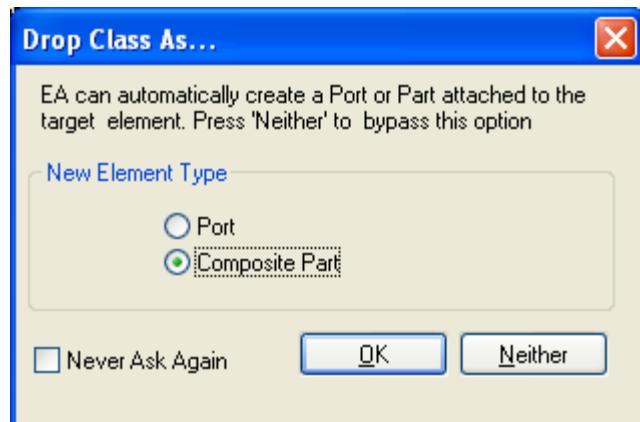
Una vez definidas las tablas, se procederá si es necesario a la definición de los distintos caminos de acceso. Para ello en el diagrama **Caminos de Acceso**, se agregarán tantos componentes como caminos se vayan a definir. En el campo **Notes**, de cada componente creado, se deberá introducir la descripción del camino de acceso que representa:



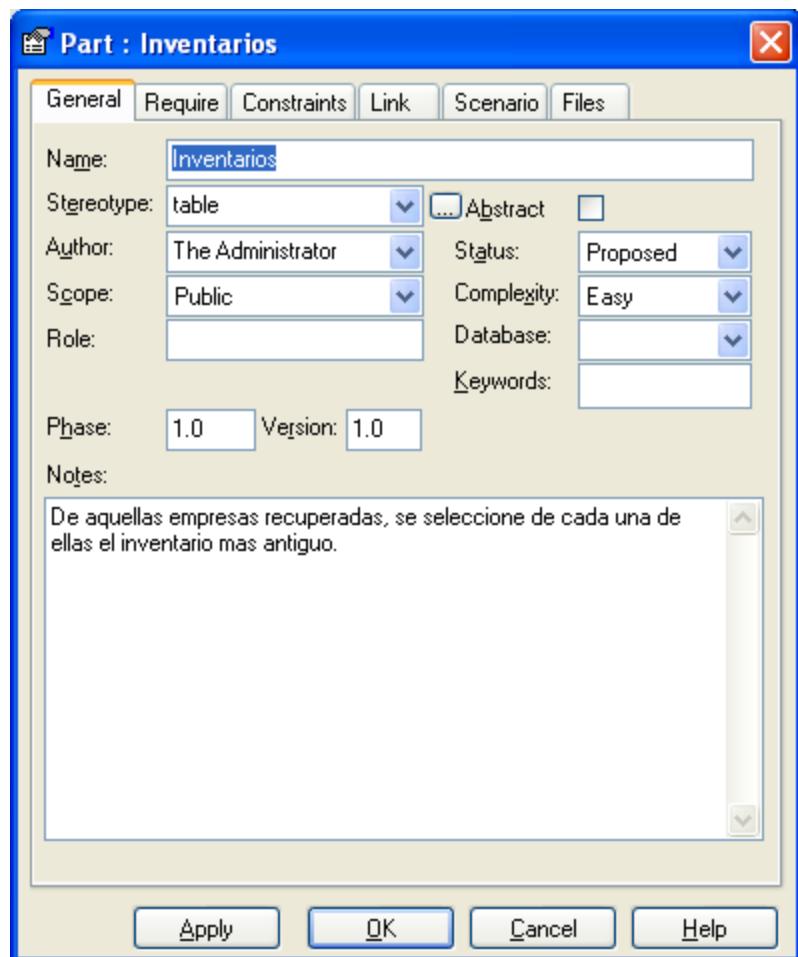
El **Diagrama Caminos Acceso** presentara las siguientes características:



Una vez definidos los distintos caminos de acceso, se deberán agregar aquellas tablas que forman parte de dicho camino de acceso, para ello se deberán arrastrar las tablas que forman parte del camino al objeto presente en el diagrama, de tal forma que al arrastrarlo se presente la siguiente pantalla

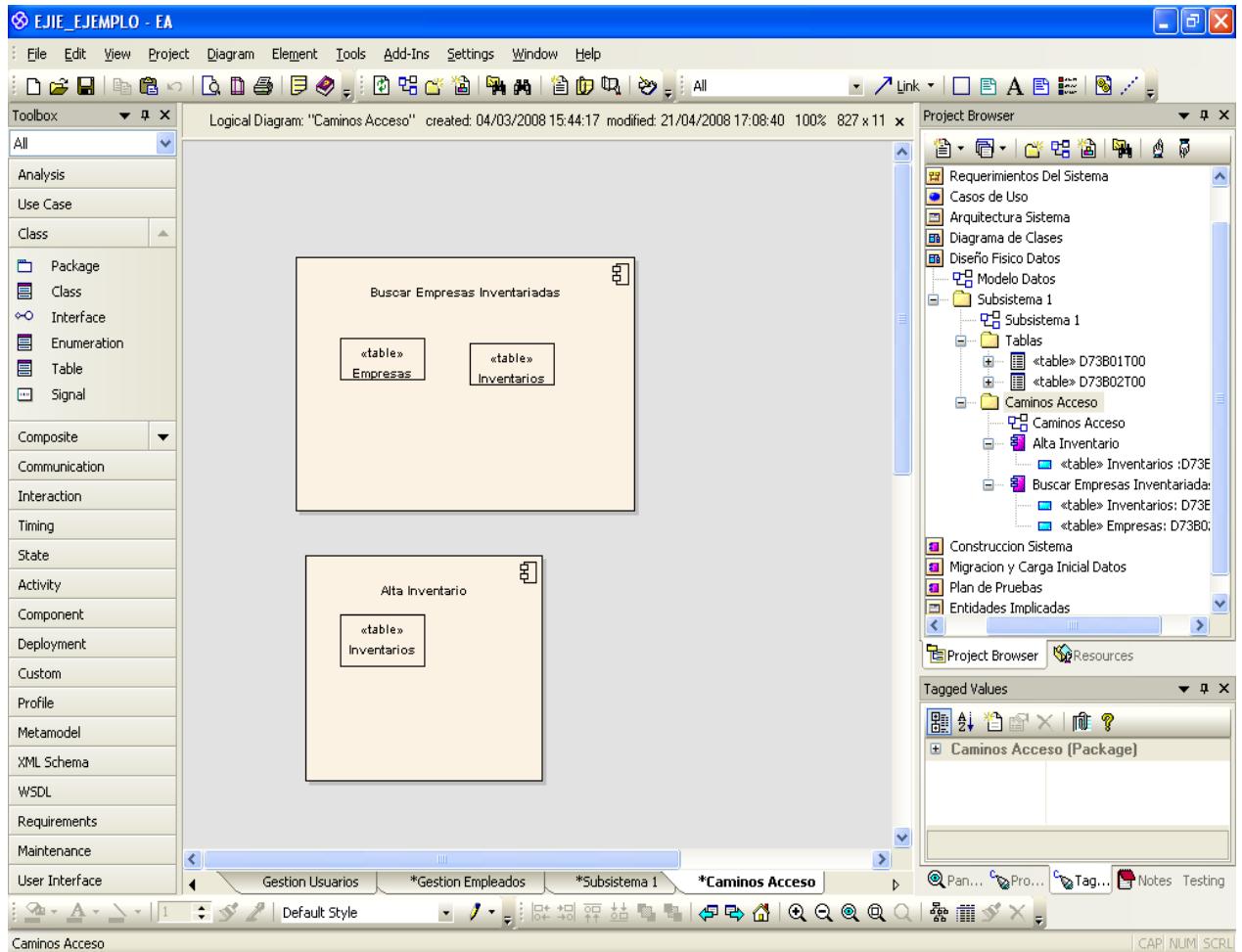


Se pulsa el botón OK, y de la pantalla que se muestra, en el campo **Name** se ha de informar el nombre de la tabla que se esta arrastrando y en el campo **Notes**, la funcionalidad que se realiza sobre dicha tabla en la definición del camino de acceso que se esta realizando.



Una vez agregadas las tablas a los componentes, es importante definir en la vista del explorador el orden en el que se ejecutan los accesos a las tablas, de tal forma que se ubique en primer lugar, la tabla que es accedida inicialmente.

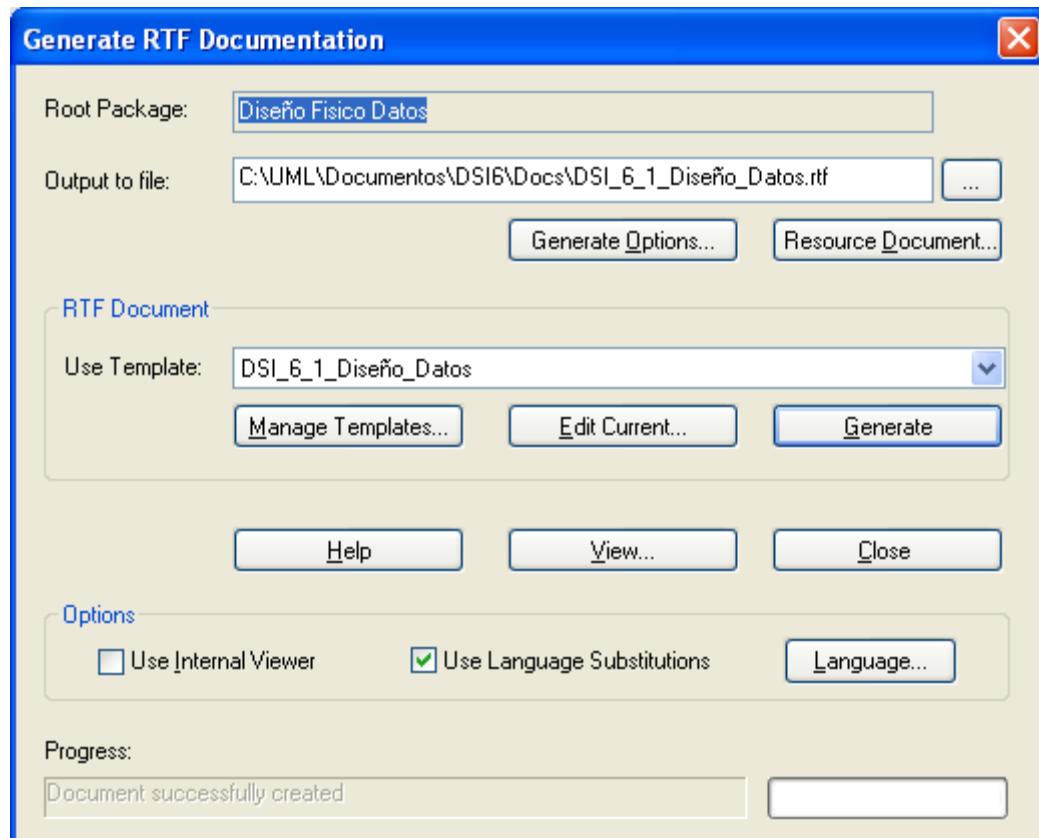
Dicha ordenación se realiza a través de la selección de las tablas definidas en los componentes y haciendo uso de los iconos **Move Element Up** (mano arriba) y **Move Element Down** (mano abajo).



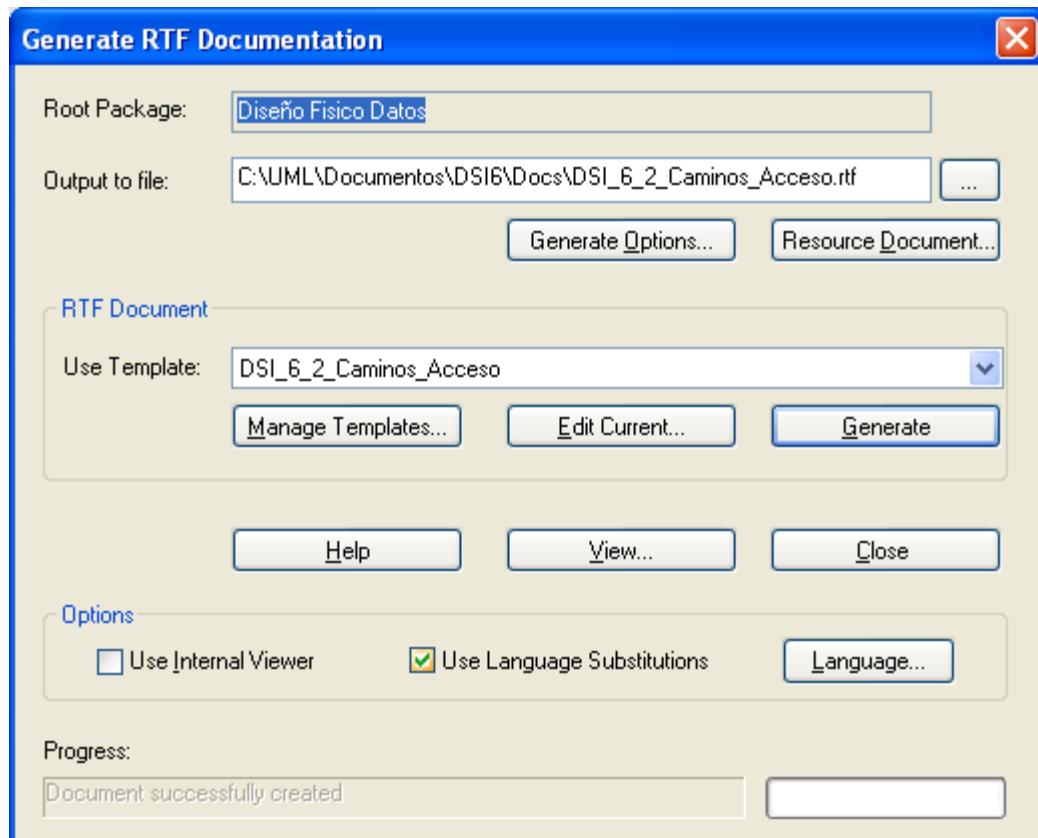
### 5.7.2. Generación Documentación

Los accesos directos que se han de ejecutar para la creación de los documentos intermedios son las siguientes

#### DSI\_6\_1\_Diseño\_Datos



## DSI\_6\_2\_ Caminos\_Acceso



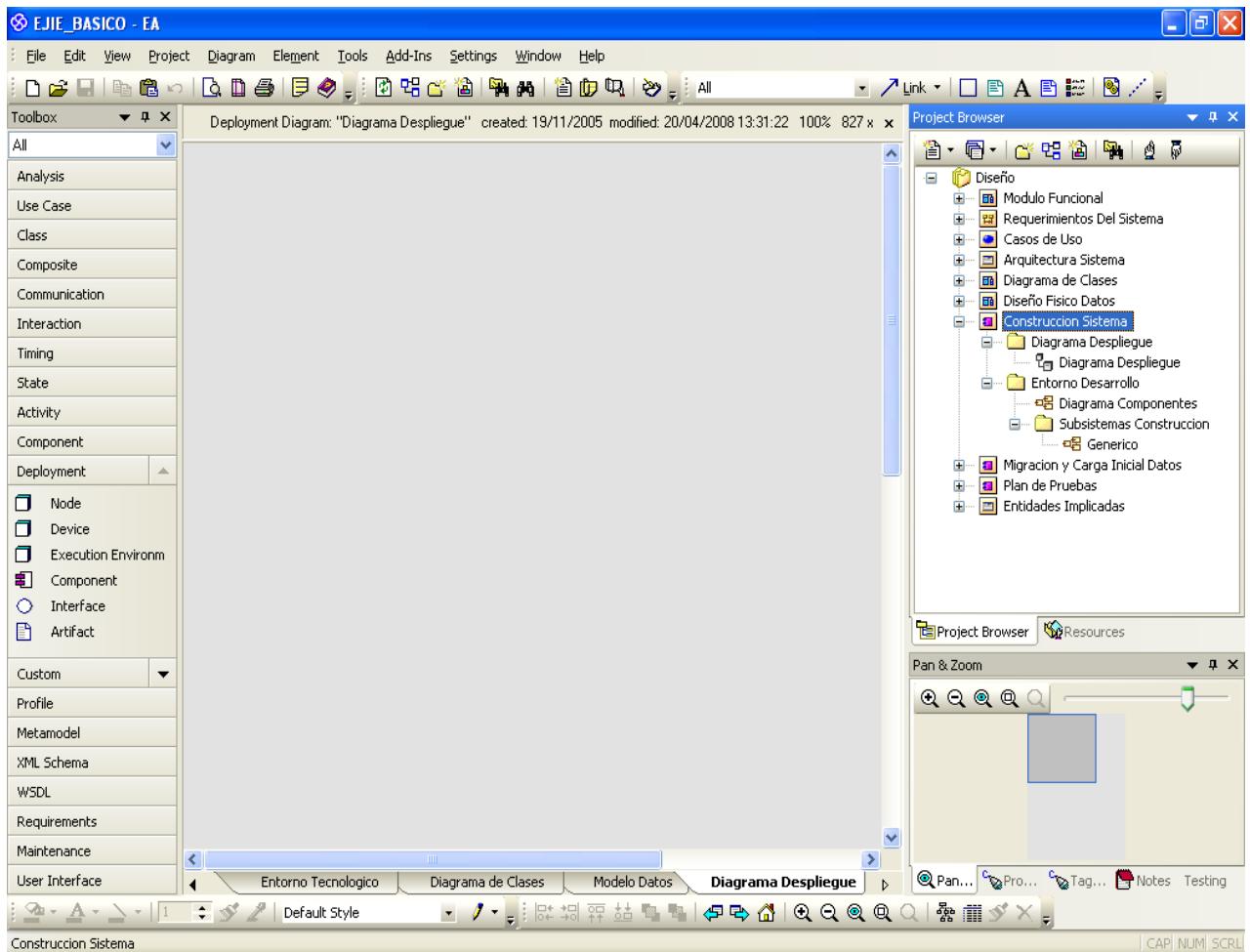
## 5.8 Diseño Físico de Datos – DSI 8

### 5.8.1. Modelado UML

Para la generación de la documentación, se han de actualizar de la estructura estándar el siguiente diagrama

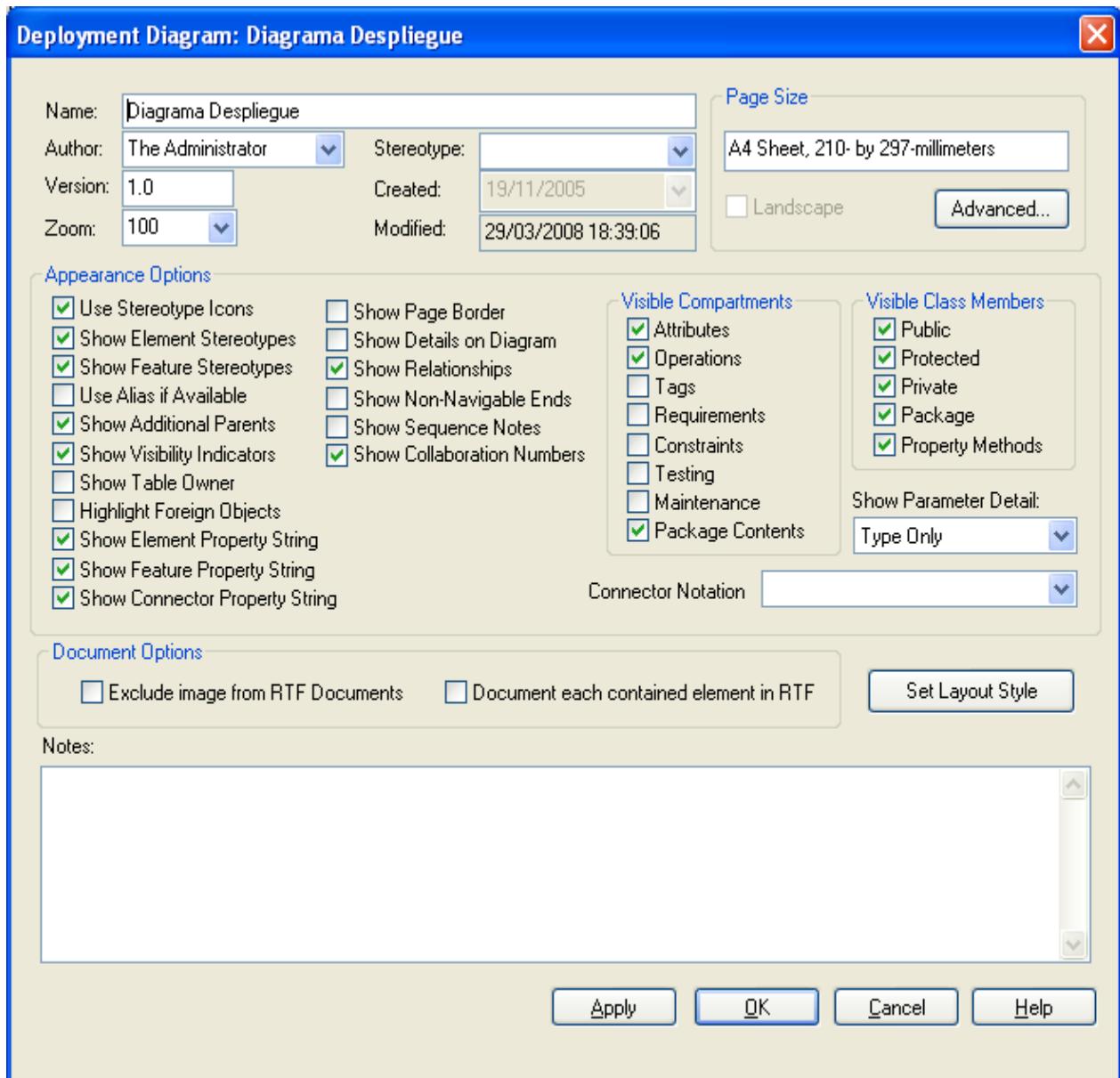
- **Construcción Sistema**

La estructura que se proporciona es la siguiente:



El **Diagrama de Despliegue** permite definir la disposición de las particiones físicas del sistema de información y la asignación de los componentes software a estas particiones.

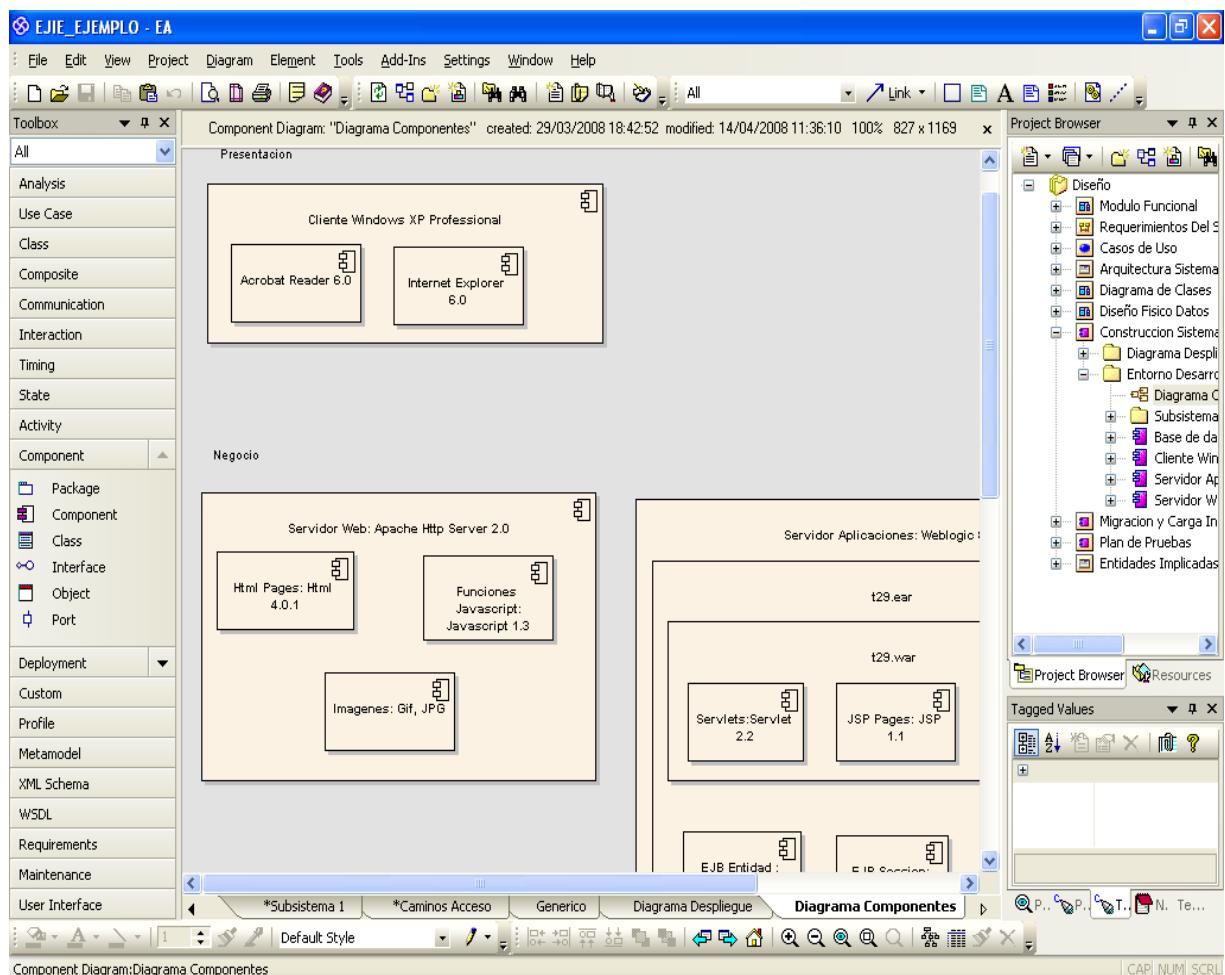
El **Diagrama de Despliegue** presenta las siguientes características:



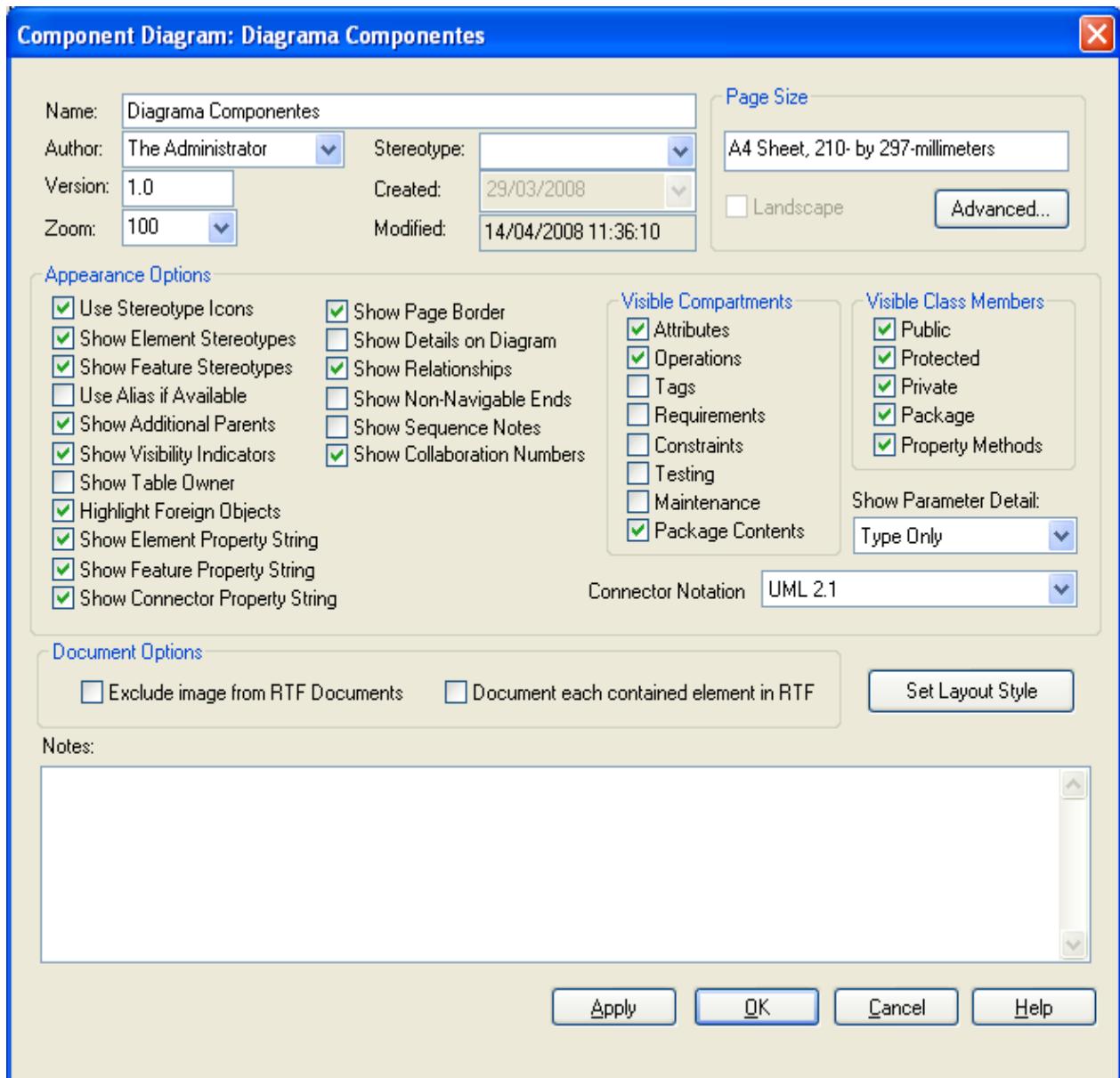
El diagrama de componentes proporciona una visión física de la construcción del sistema de información. Muestra la organización de los componentes software, sus interfaces y las dependencias entre ellos.

La creación del diagrama se dividirá inicialmente en función del número de capas que formen la aplicación y en cada capa se crearán tantos componentes, como elementos software formen parte de dicha capa.

Definidos los componentes principales, se podrán vincular los componentes secundarios.

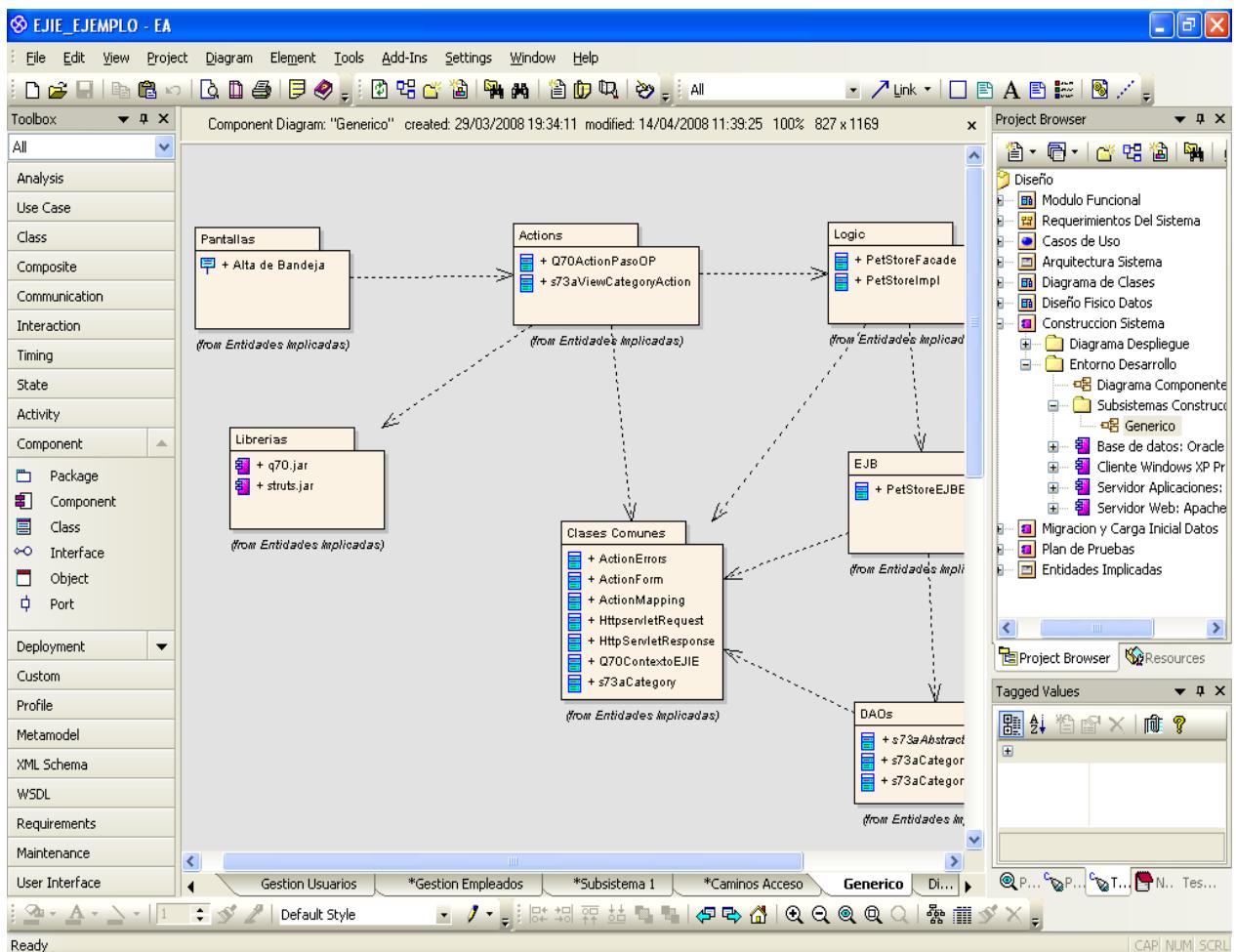


El **Diagrama de Componentes** presentara las siguientes características:

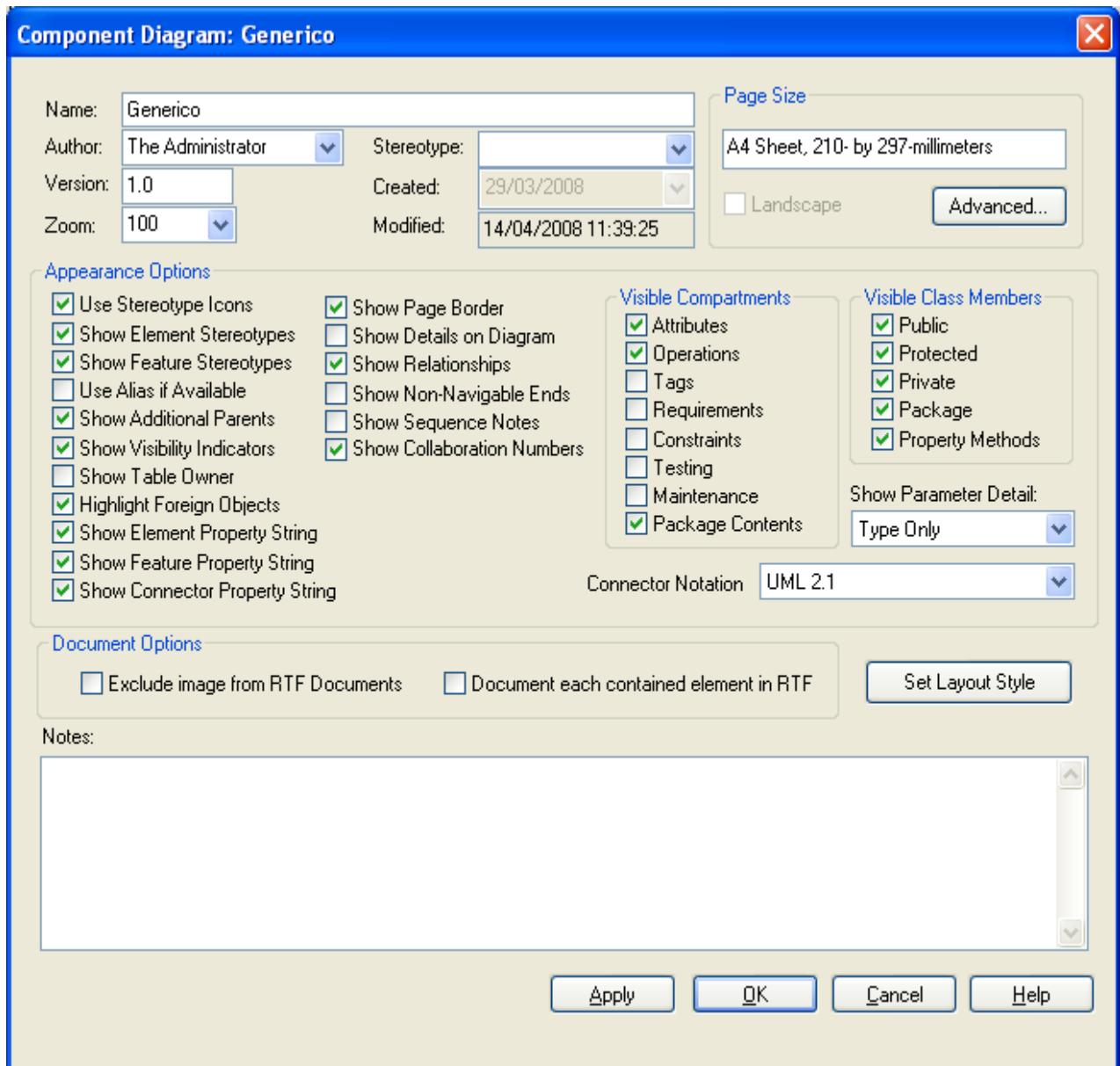


Definido el diagrama de componentes, se pueden llegar a definir los subsistemas de construcción de la aplicación en la carpeta **Subsistemas Construcción**

En dicha carpeta se crearán aquellos subsistemas que sean los suficientemente importantes para ser definidos y la forma de implementación se realizara arrastrando los objetos ya definidos y que se encuentran disponibles en los distintos diagramas ya creados.



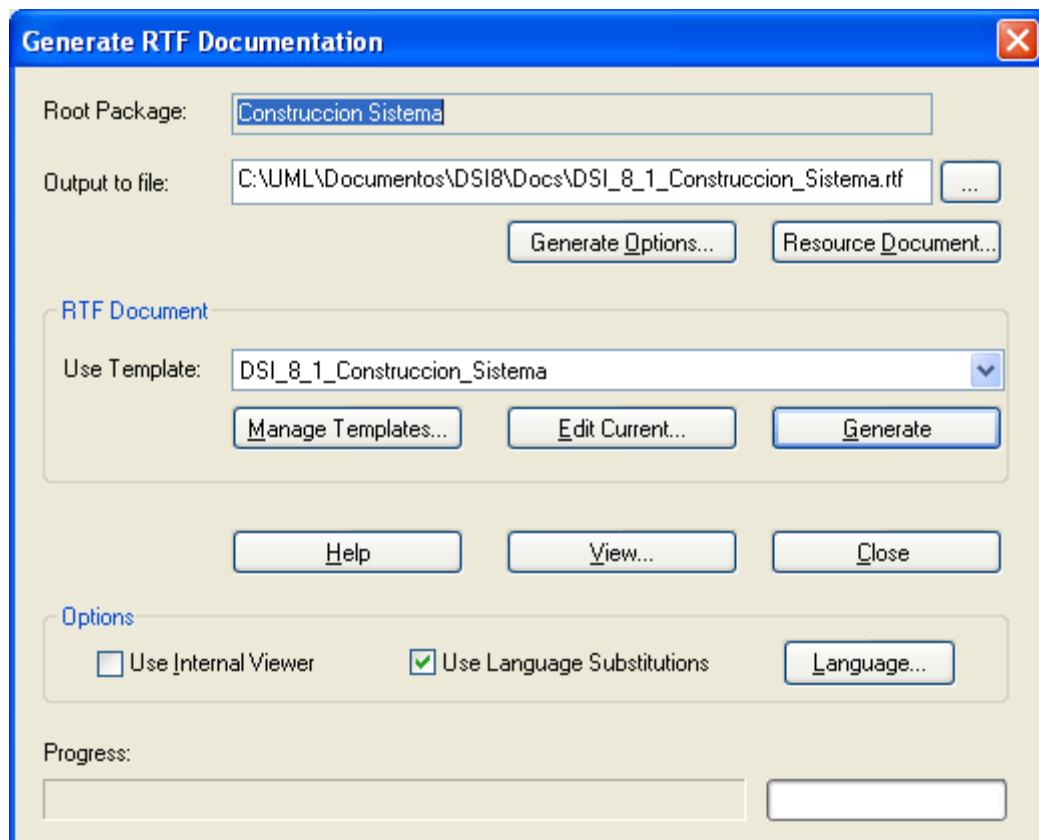
Los diagramas de los subsistemas de Construcción presentaran las siguientes características:



### 5.8.2. Generación Documentación

Los accesos directos que se han de ejecutar para la creación de los documentos intermedios son las siguientes

#### DSI\_8\_1\_Construcción\_Sistema



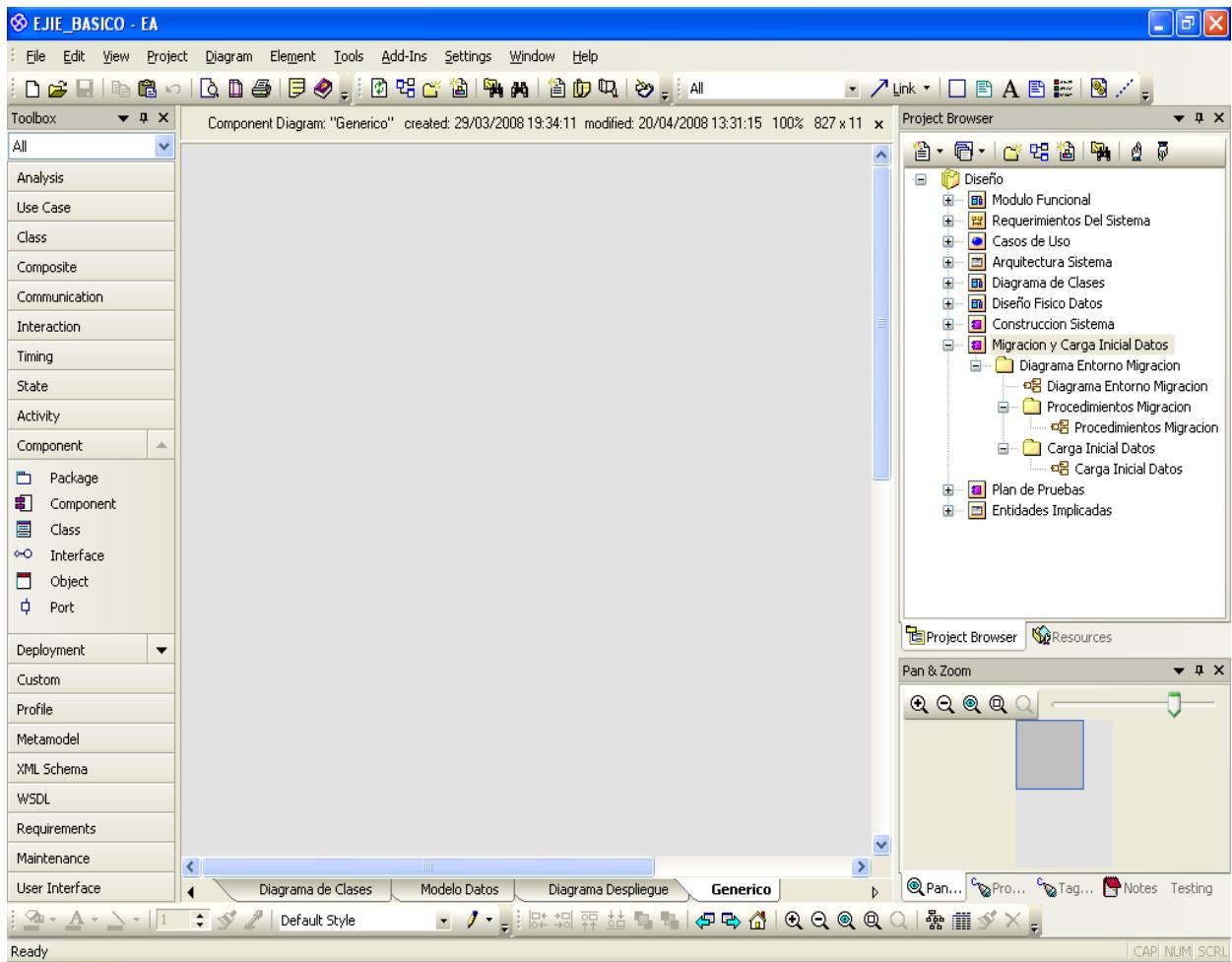
## 5.9 Diseño de la Migración y carga inicial de datos – DSI 9

### 5.9.1. Modelado UML

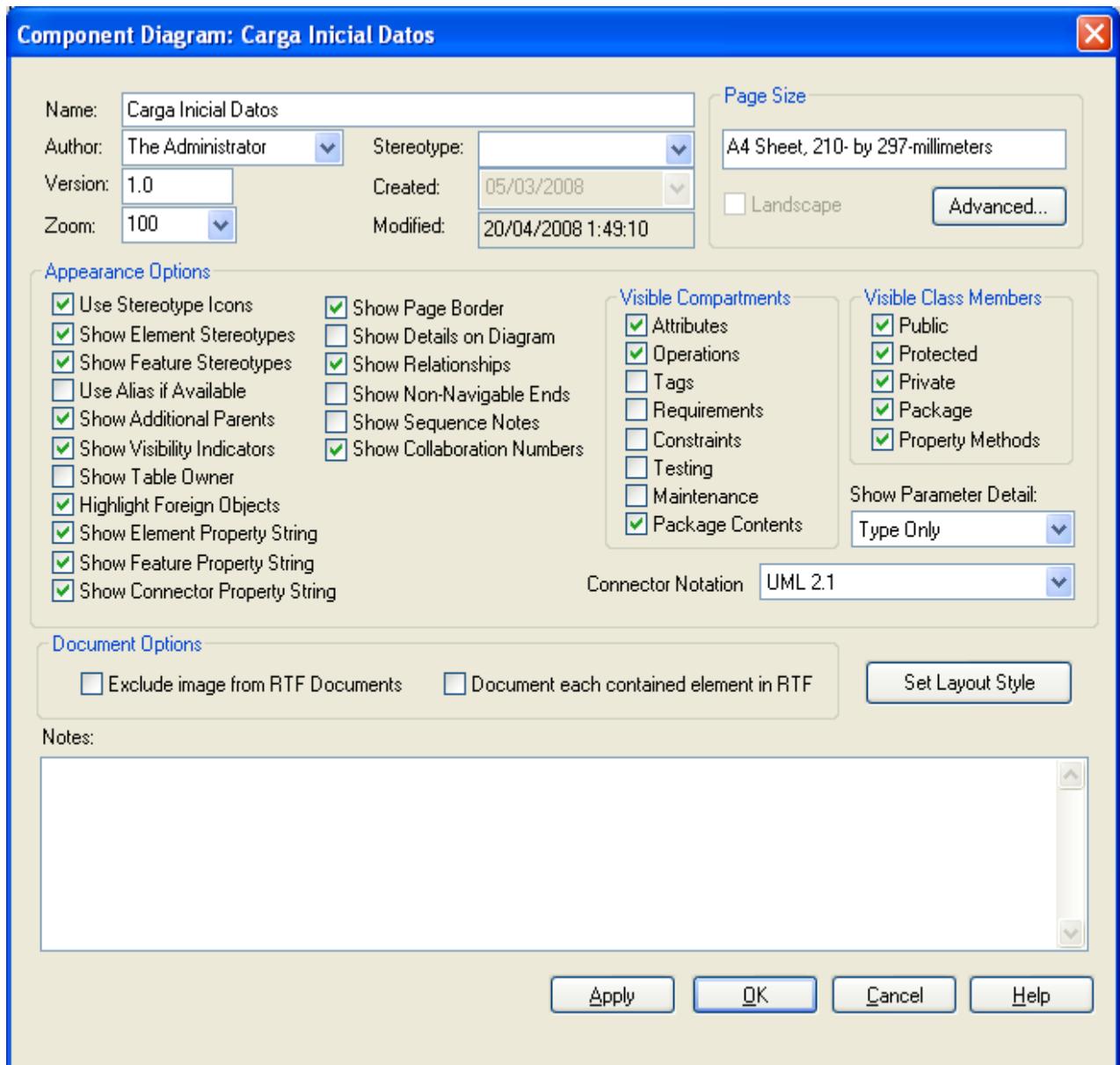
Para la generación de la documentación, se han de actualizar de la estructura estándar el siguiente diagrama

- **Migración y Carga Inicial Datos**

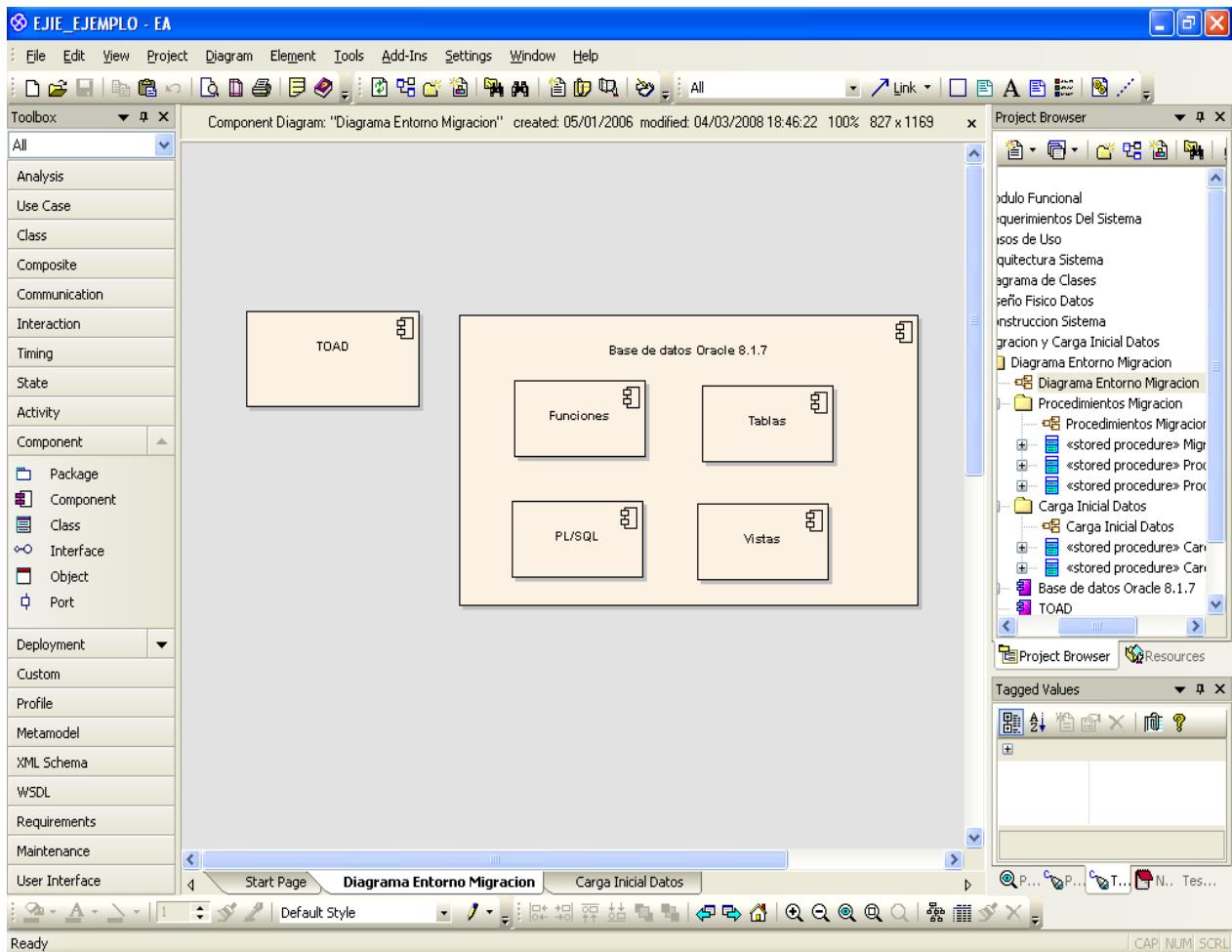
La estructura que se proporciona es la siguiente:



Los diagramas que se crean en este apartado presentan todos la siguiente estructura:

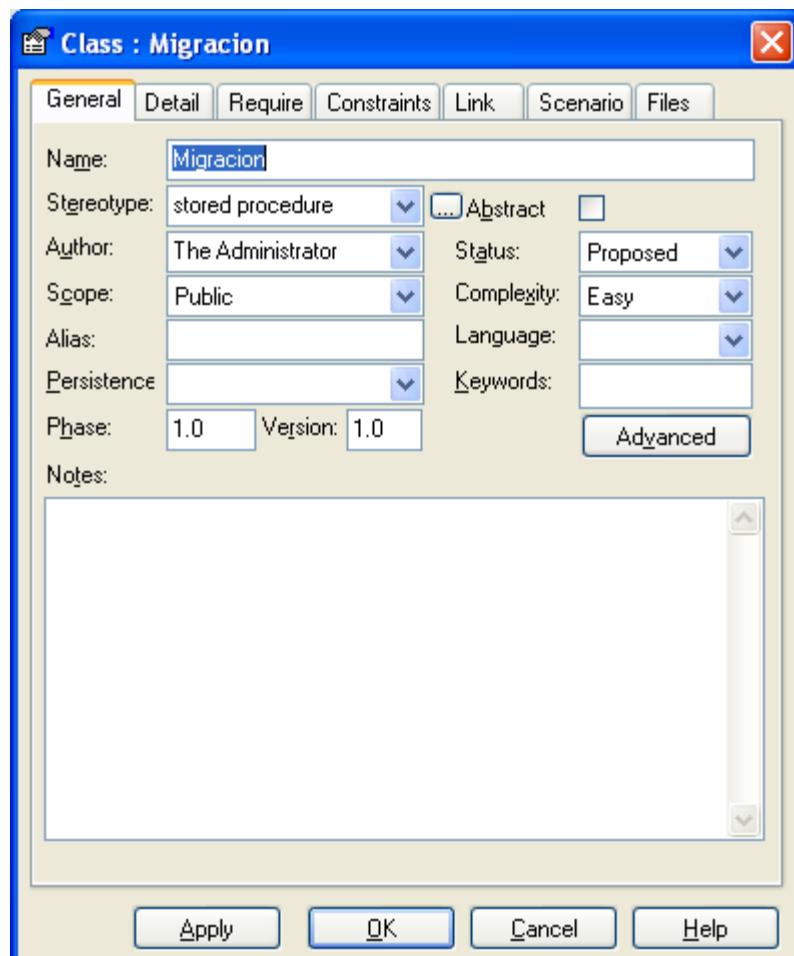


En el **Diagrama de Entorno de Migración** se ha de realizar un diseño del entorno de migración, haciendo uso de objetos de tipo **Component**, tal y como se ha indicado en puntos anteriores, de tal forma que se tengan en cuenta las herramientas o utilidades software específicas

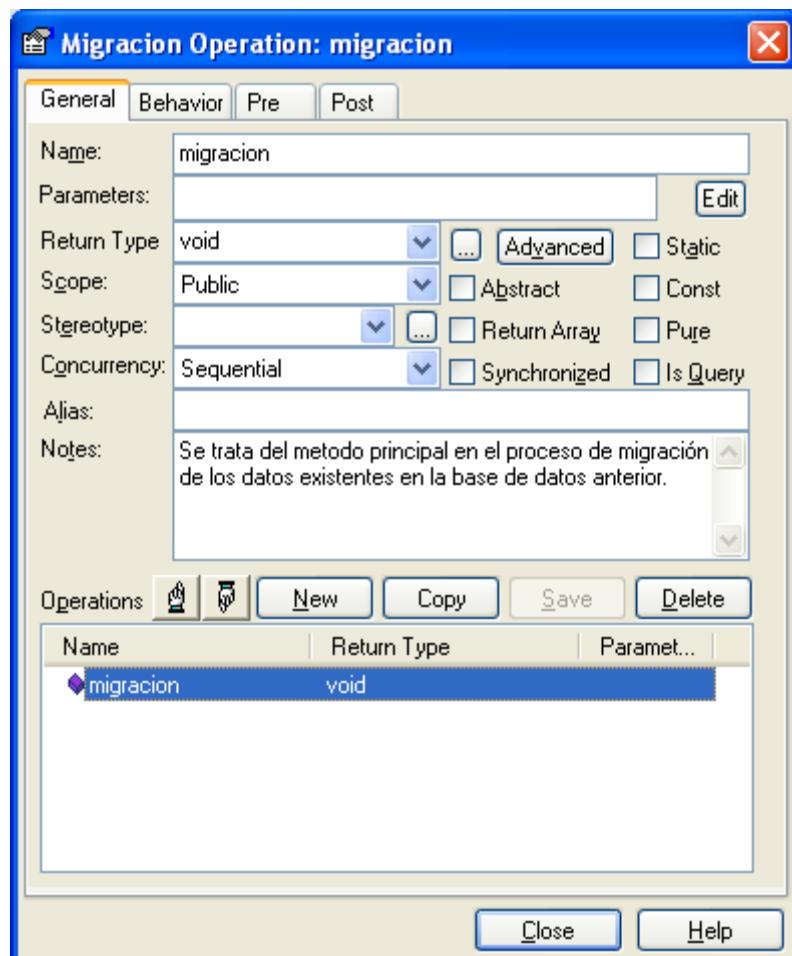


En el **Diagrama Procedimientos de Migración** se definirá el proceso de migración que se va a realizar, de tal forma que se pueden identificar los procesos que forman parte de la migración y sus relaciones (quien invoca a quien, etc).

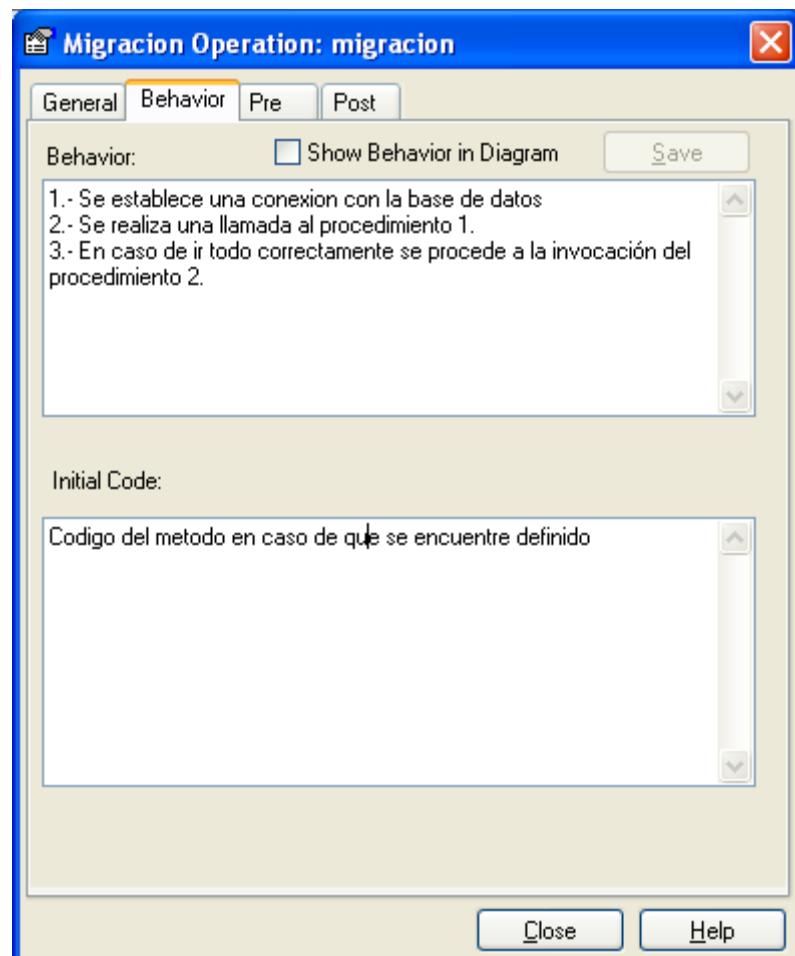
Para ello se deberán agregar tantas clases al diagrama, como procedimientos existan en la migración y cada una de las clases presentara la siguiente estructura:



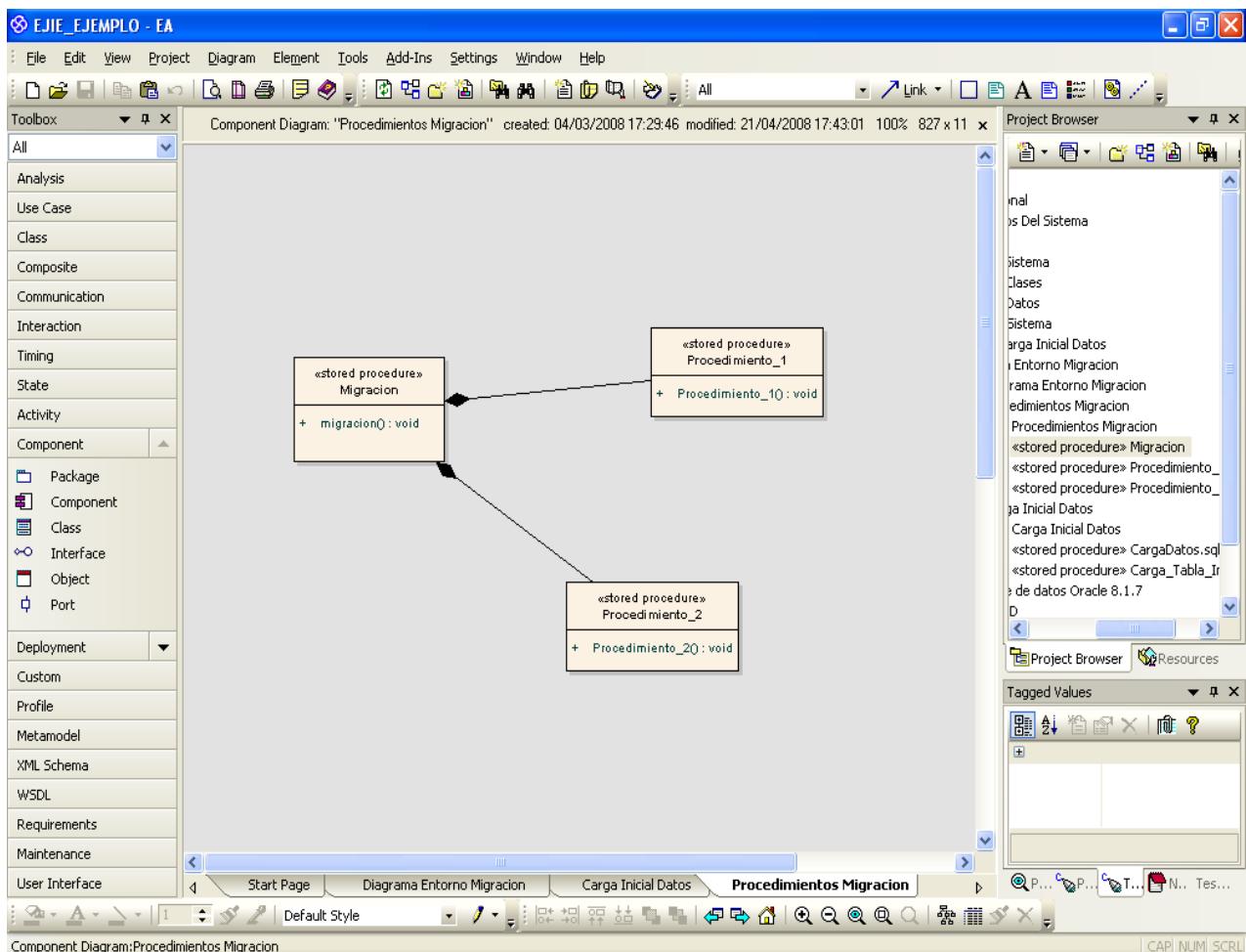
Por cada procedimiento almacenado identificada, se crearan tantos Operations, como funcionalidades proporcione dicho procedimiento:



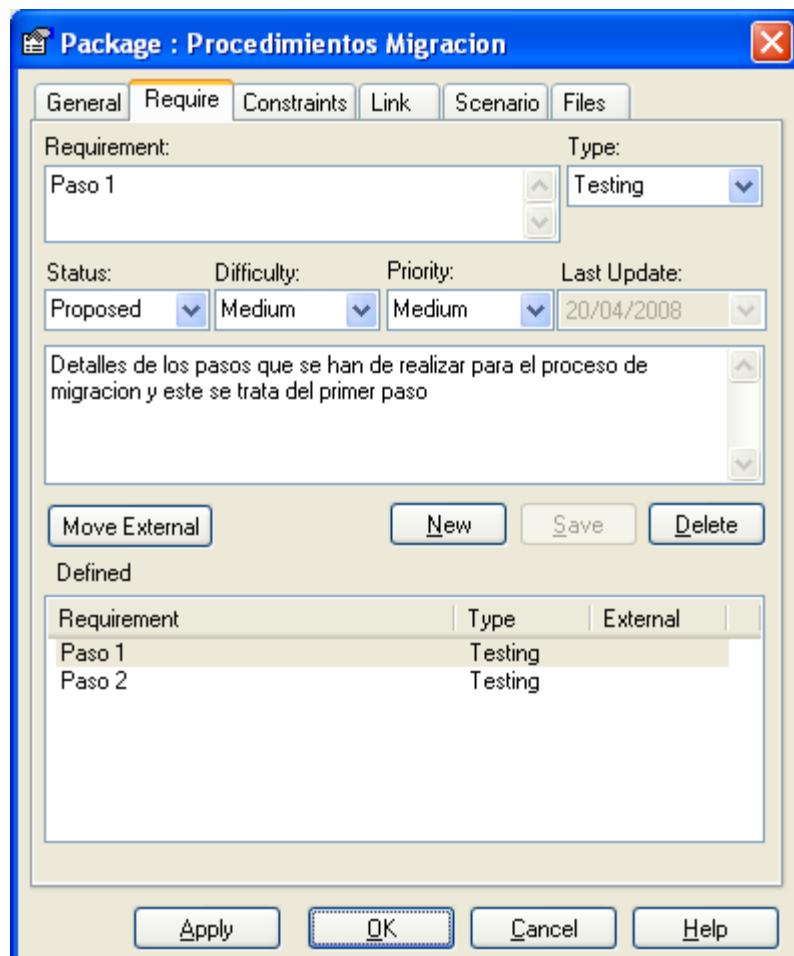
En la pestaña de **Behavior**, se podrá definir tanto el proceso que realiza el procedimiento a nivel funcional (sección **Behavior**) , como el código o pseudo código del procedimiento (sección **Initial Code**)



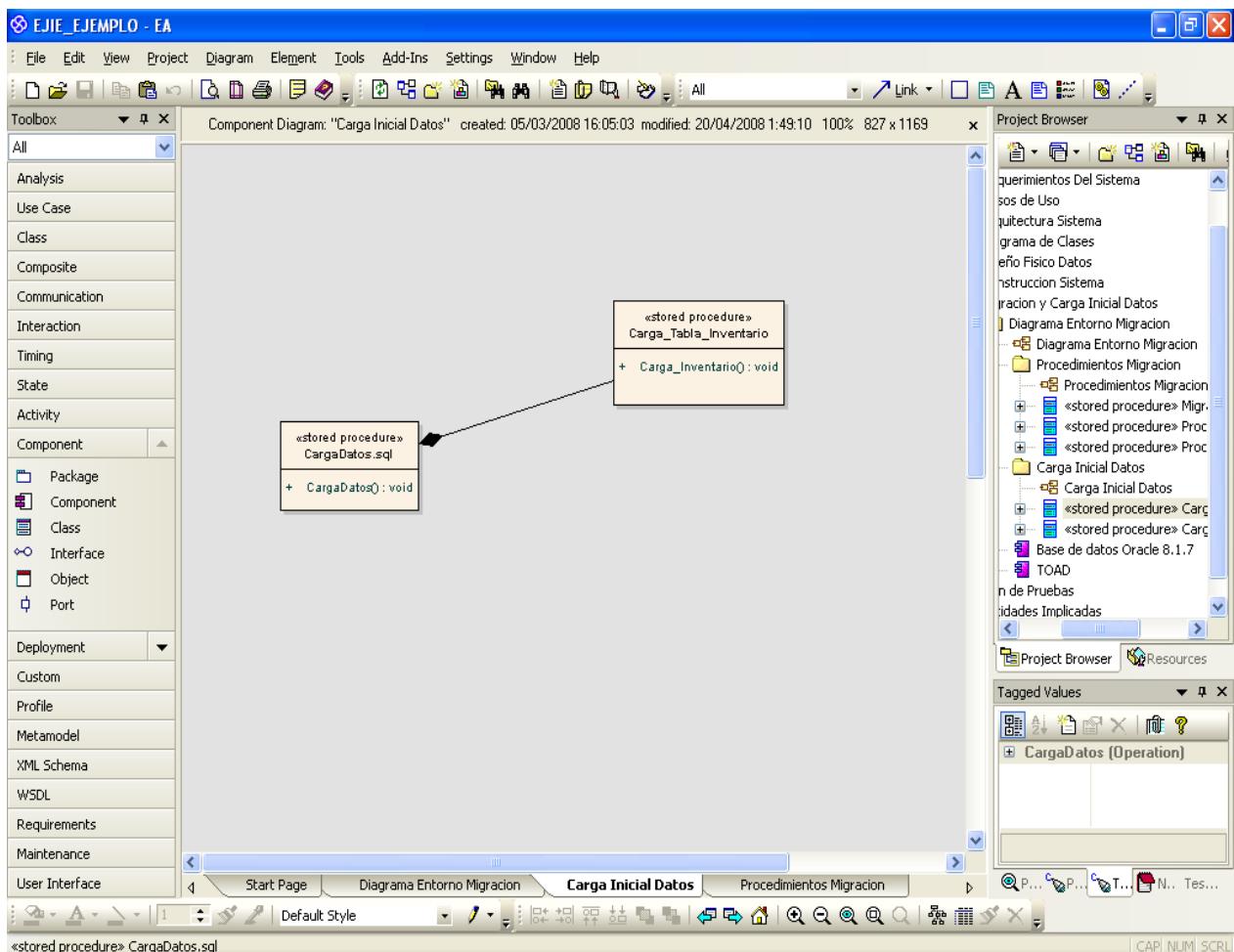
Creados todos los procedimientos, se incorporaran al diagrama y se establecerán sus relaciones:



Definidos los procedimientos que forman parte de la migración se ha de definir el plan de pruebas para la misma, para ello se ha de seleccionar la carpeta **Procedimientos Migración** y acceder a sus propiedades. Desde la pestaña de **Require**, se han de crear tantas entradas de tipo **Testing**, como pruebas de test se vayan a realizar para validar el proceso migración.



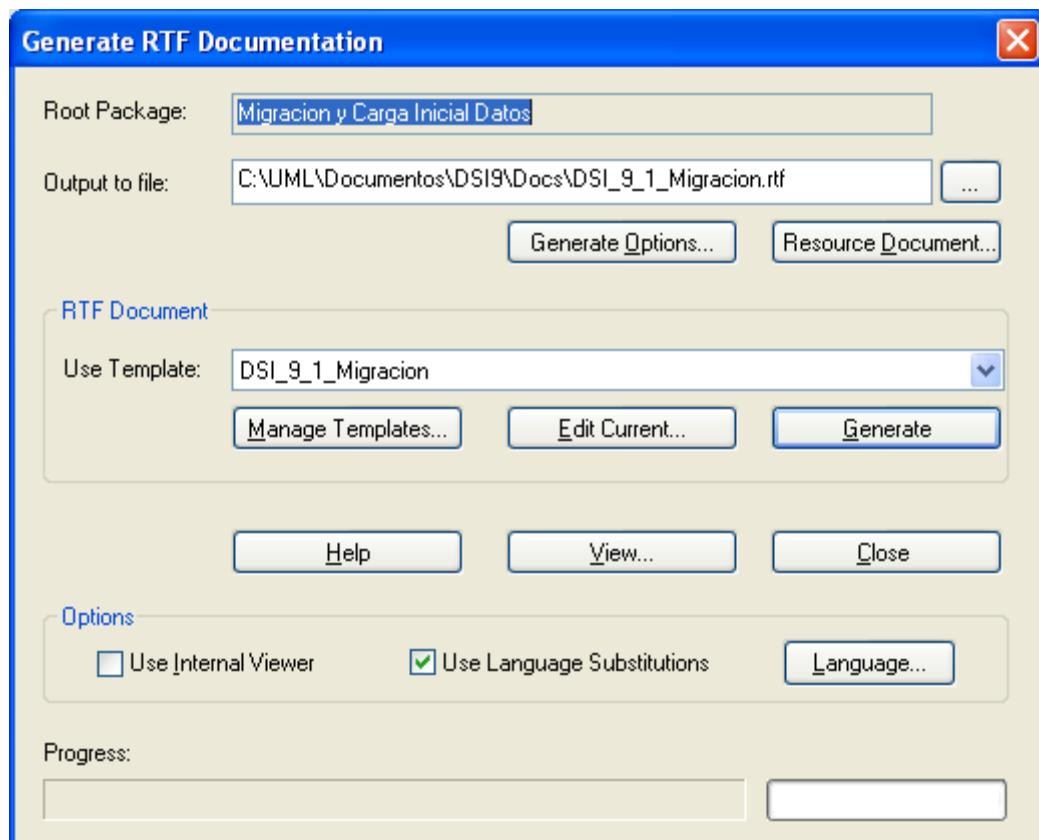
El proceso de modelado de **Carga Inicial** será el mismo que para los **Procedimientos de Migración**



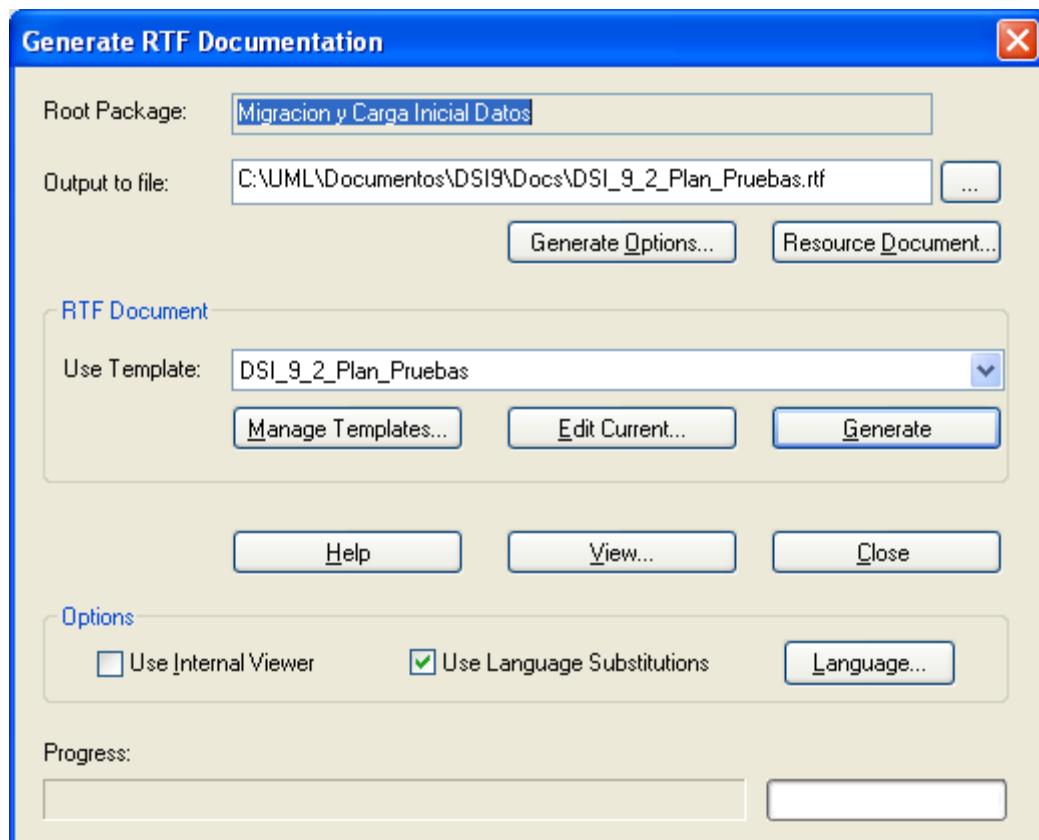
### **5.9.2. Generación Documentación**

Los accesos directos que se han de ejecutar para la creación de los documentos intermedios son las siguientes

#### **DSI\_9\_1\_Migracion**



## DSI\_9\_2\_Plan\_Pruebas



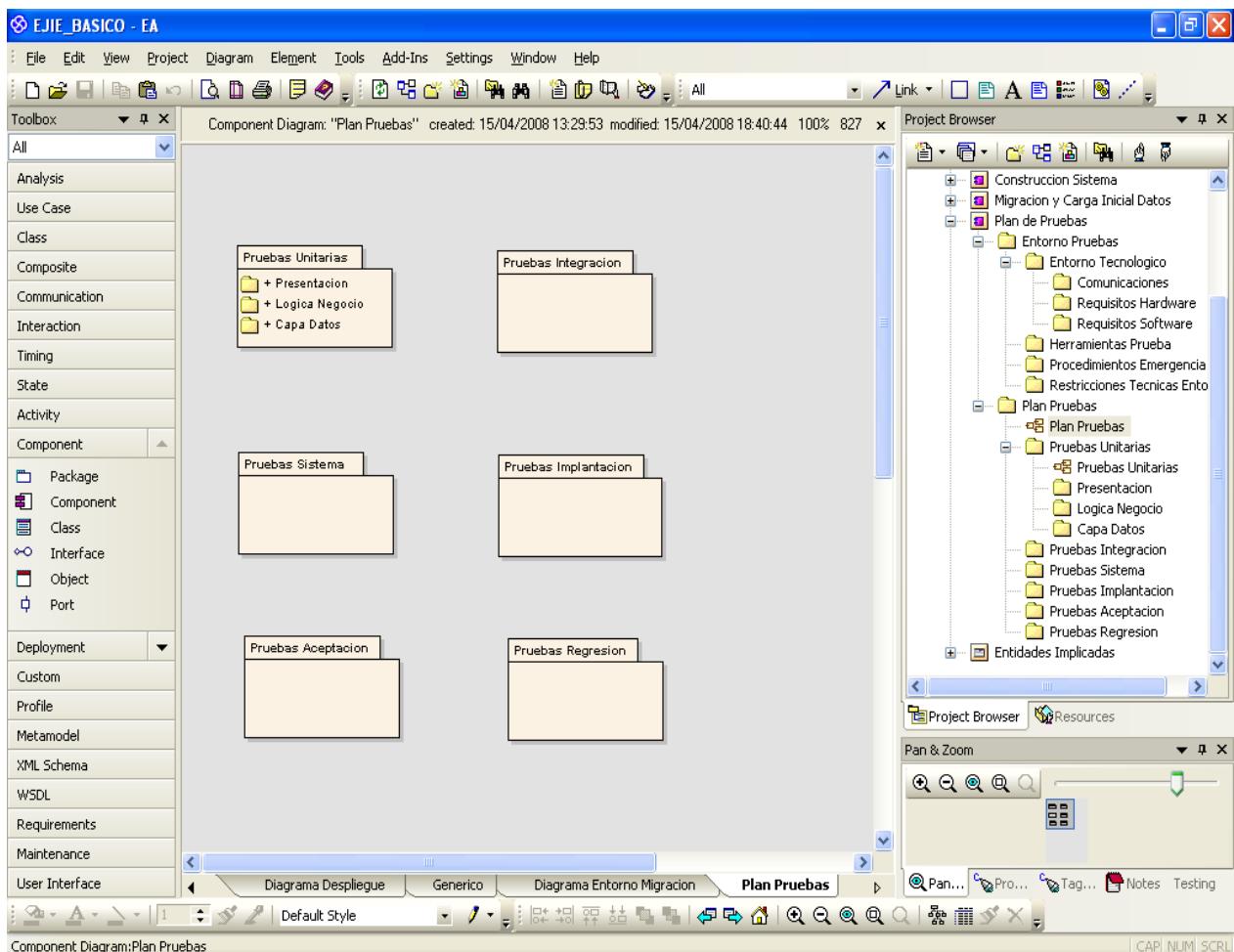
## 5.10 Diseño de la Migración y carga inicial de datos – DSI 9

### 5.10.1. Modelado UML

Para la generación de la documentación, se han de actualizar de la estructura estándar el siguiente diagrama

- **Plan de Pruebas**

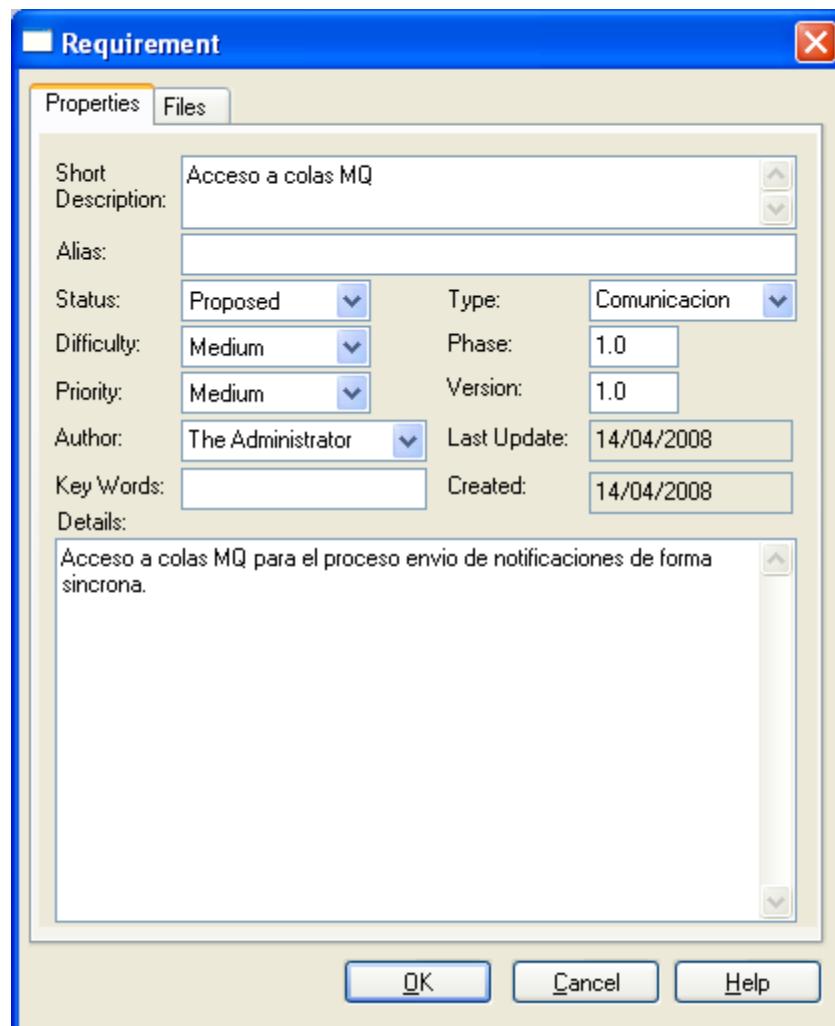
La estructura que se proporciona es la siguiente:

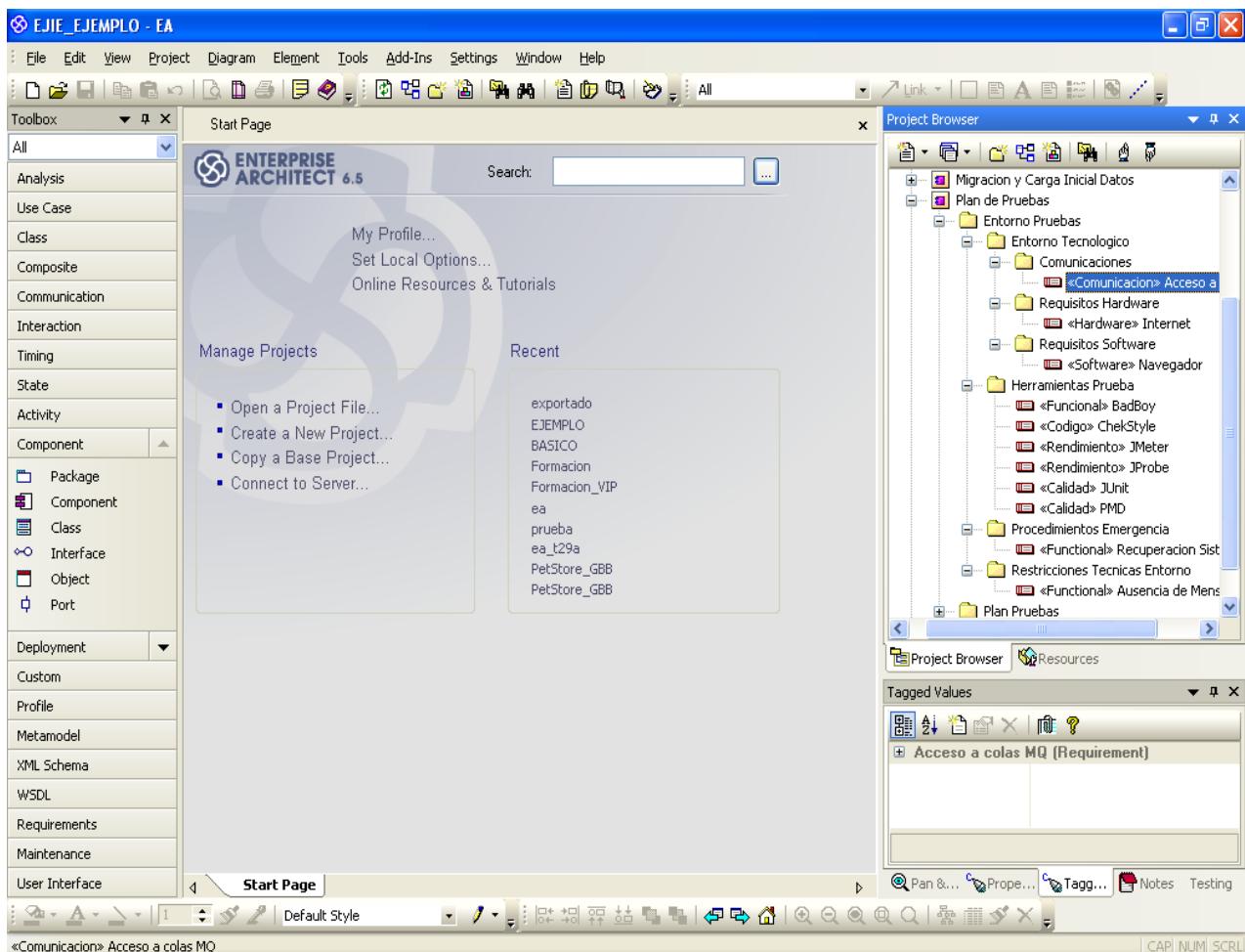


Para la definición del **Entorno de Pruebas** se deberán definir los siguientes aspectos:

- Entorno Tecnológico
  - Comunicaciones
  - Requisitos Hardware
  - Requisitos Software
- Herramientas Prueba
- Procedimientos Emergencia
- Restricciones Técnicas Entorno

Para cada uno de los elementos identificados anteriormente se deberán agregar tantos elementos como sean necesarios(Punto 3.3.1), de tal forma que cada uno de ellos quede bien definido

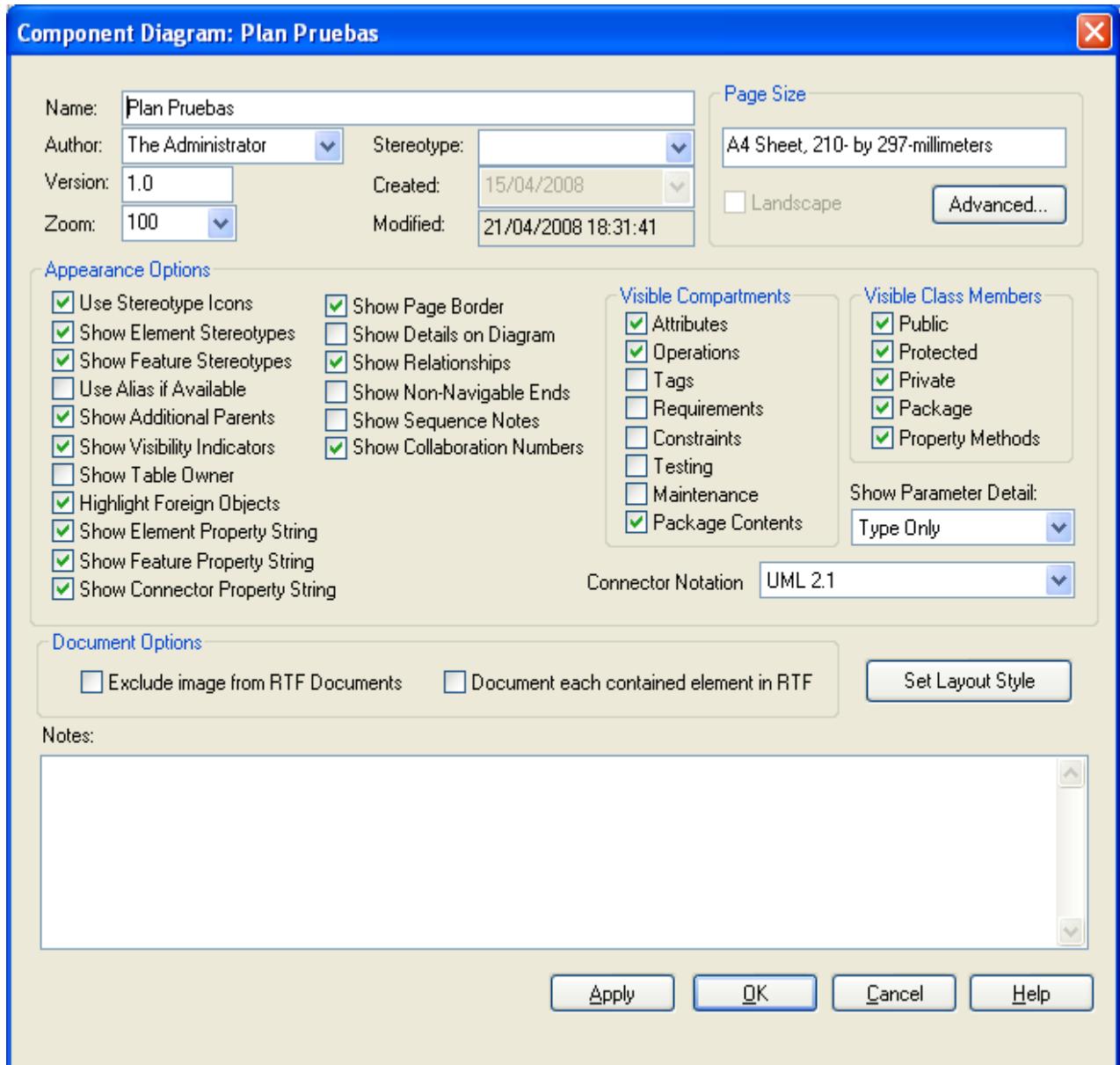




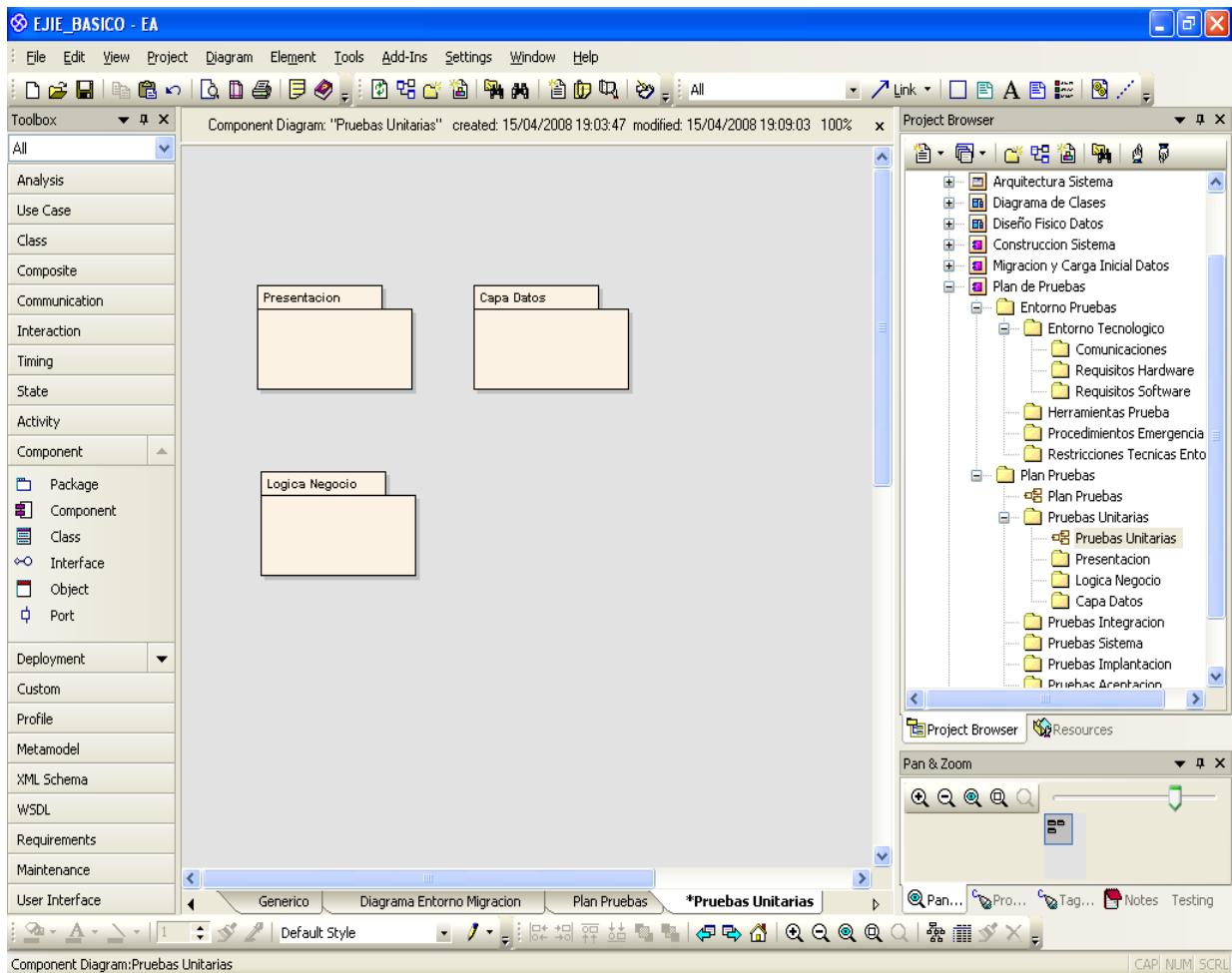
Una vez definido el **Entorno de Pruebas**, se deberá definir el **Plan de Pruebas**, el plan de pruebas presenta la siguientes estructura:

- Pruebas Unitarias
  - Presentación
  - Lógica Negocio
  - Capa Datos
- Pruebas Integración
- Pruebas Sistema
- Pruebas Implementación
- Pruebas Aceptación

Los diagramas correspondientes al **Plan de Pruebas** presentan las siguientes características:



Las pruebas unitarias presentan la siguiente estructura

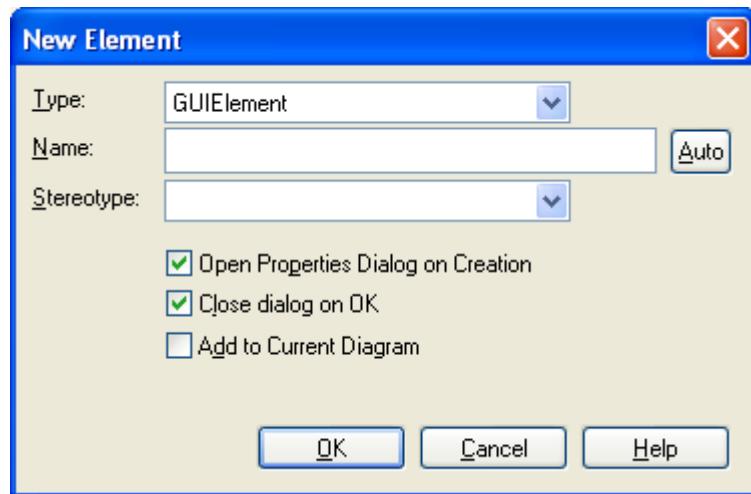


Por cada capa definida en las pruebas unitarias se deberán agregar tantos elementos, como pruebas unitarias se deberán realizar.

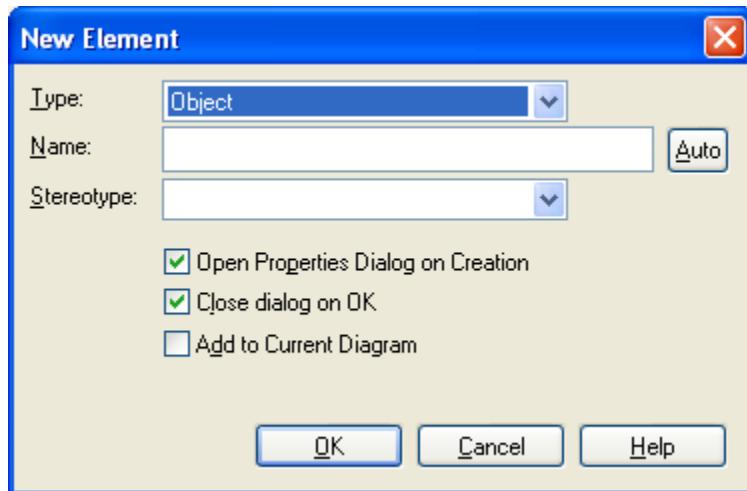
La definición de las pruebas unitarias se realizará de manera genérica, es decir, identificando componentes genéricos, como por ejemplo, definición de pruebas para campos Date, definición de pruebas para campos numéricos, etc.

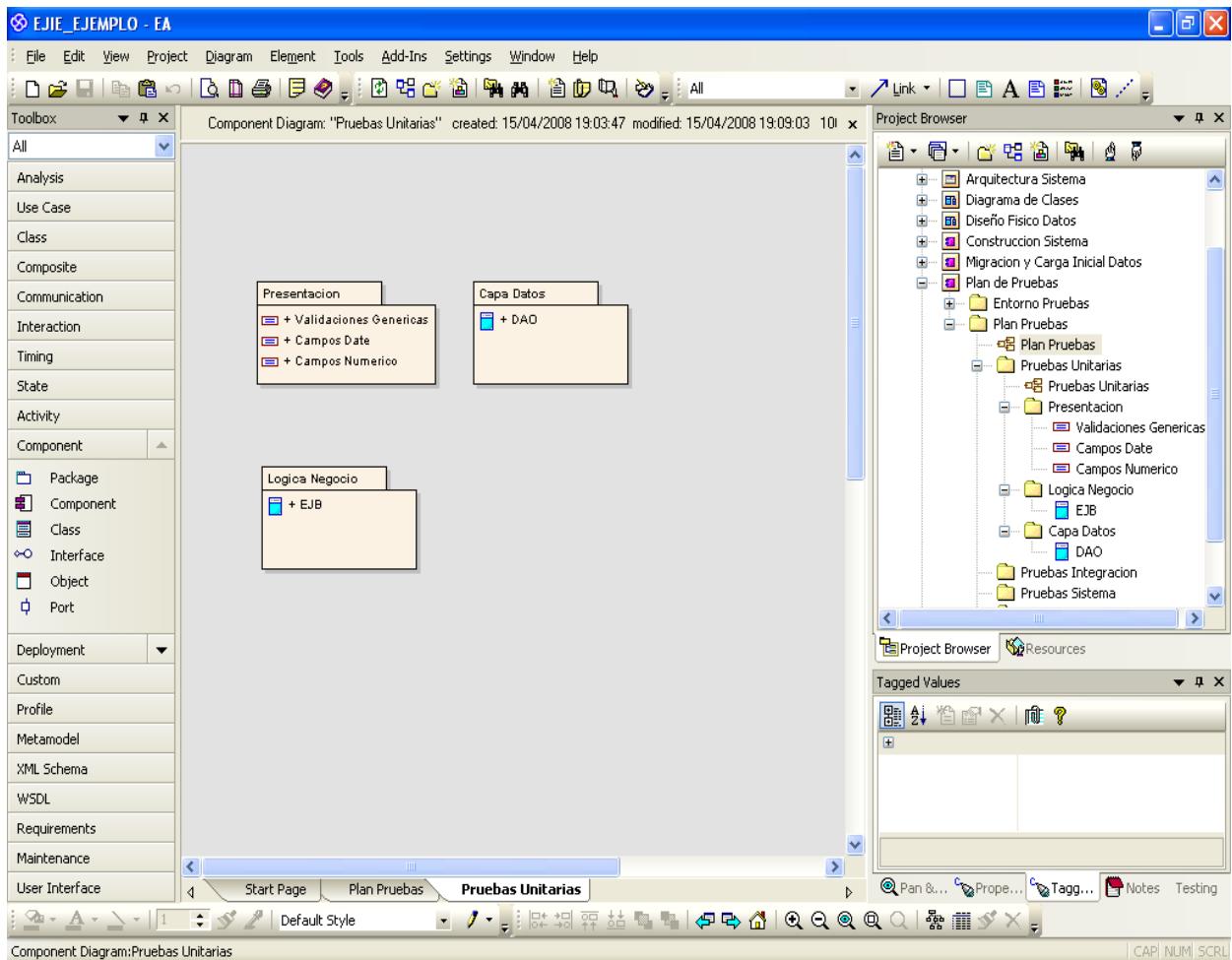
En función de la capa en la que se estén agregando las pruebas, los tipos de elementos a agregar serán diferentes:

- **Presentación**



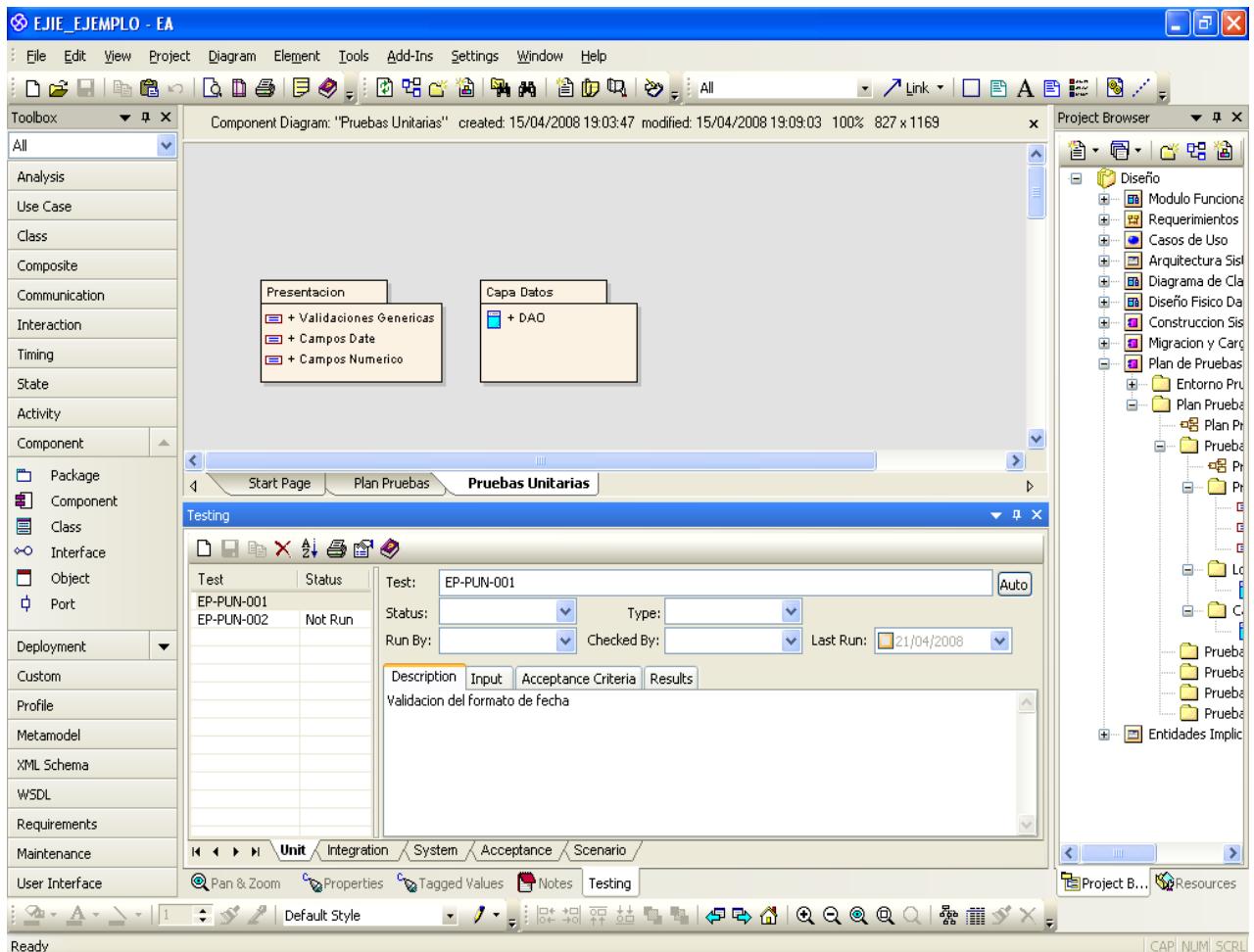
- Negocio y Datos





Una vez definidos los elementos que serán sometidos a pruebas, se deberán definir las mismas, para ello se ha de hacer visible la ventana de **Testing**, accediendo al menú **View->Testing**

A continuación, se deberá seleccionar del explorador aquel elemento al que se vayan a vincular las pruebas, y se visualizara la siguiente pantalla:



Para la creación de las **Pruebas Unitarias**, se seleccionará la pestaña de Unit, y pulsando el botón **New**, se podrán definir las distintas pruebas a realizar.

Para ello se informaran los siguientes campos:

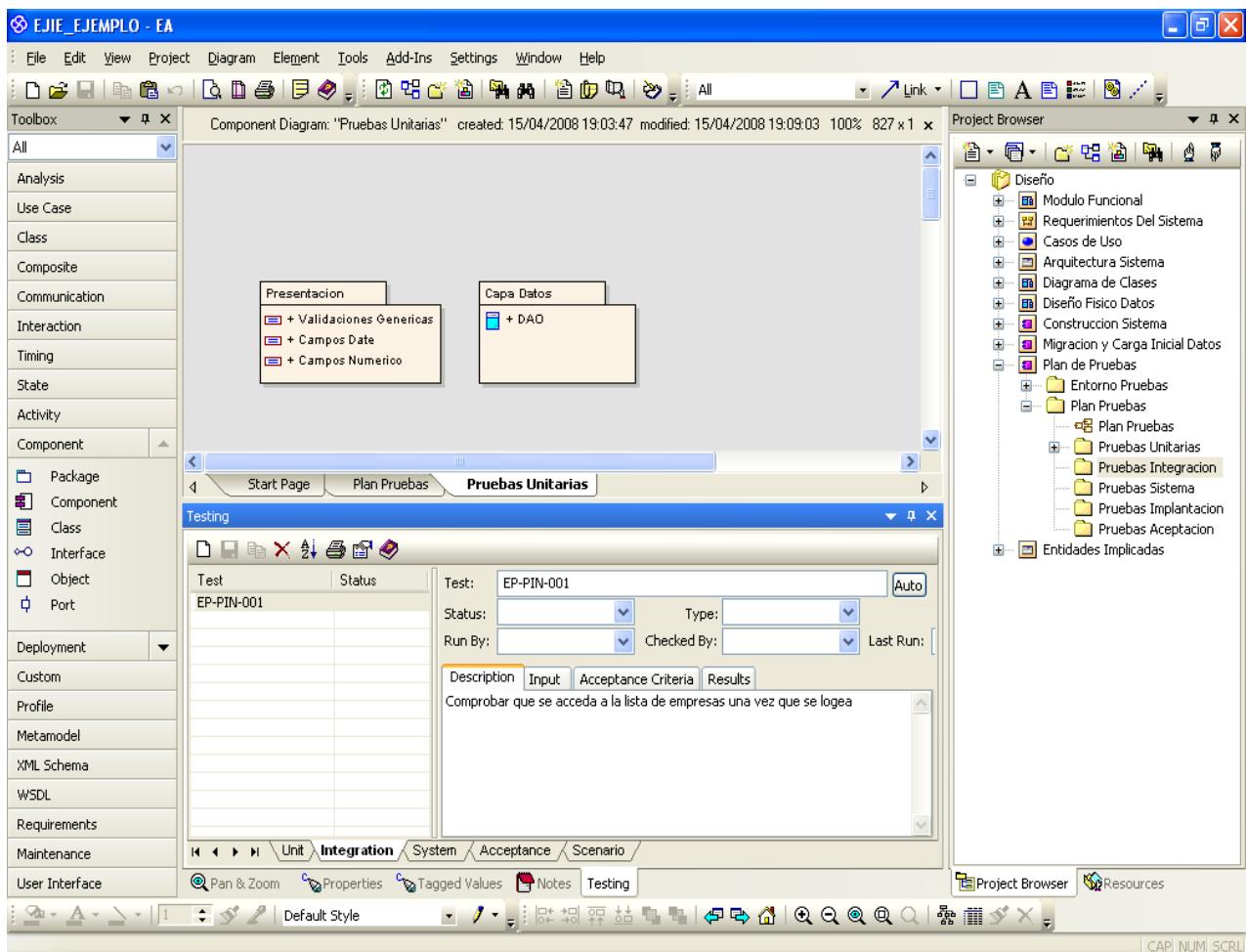
- **Test** -> Nombre del test
- **Description** -> Descripción funcional de la prueba.
- **Input** -> Juego de datos de entrada
- **Results** -> Resultados esperados

Informados dichos campos, para guardar los datos se pulsara el botón **Save**.

Para el resto de pruebas, la definición de los Tests, se realizara sobre cada una de las carpetas creadas al efecto.

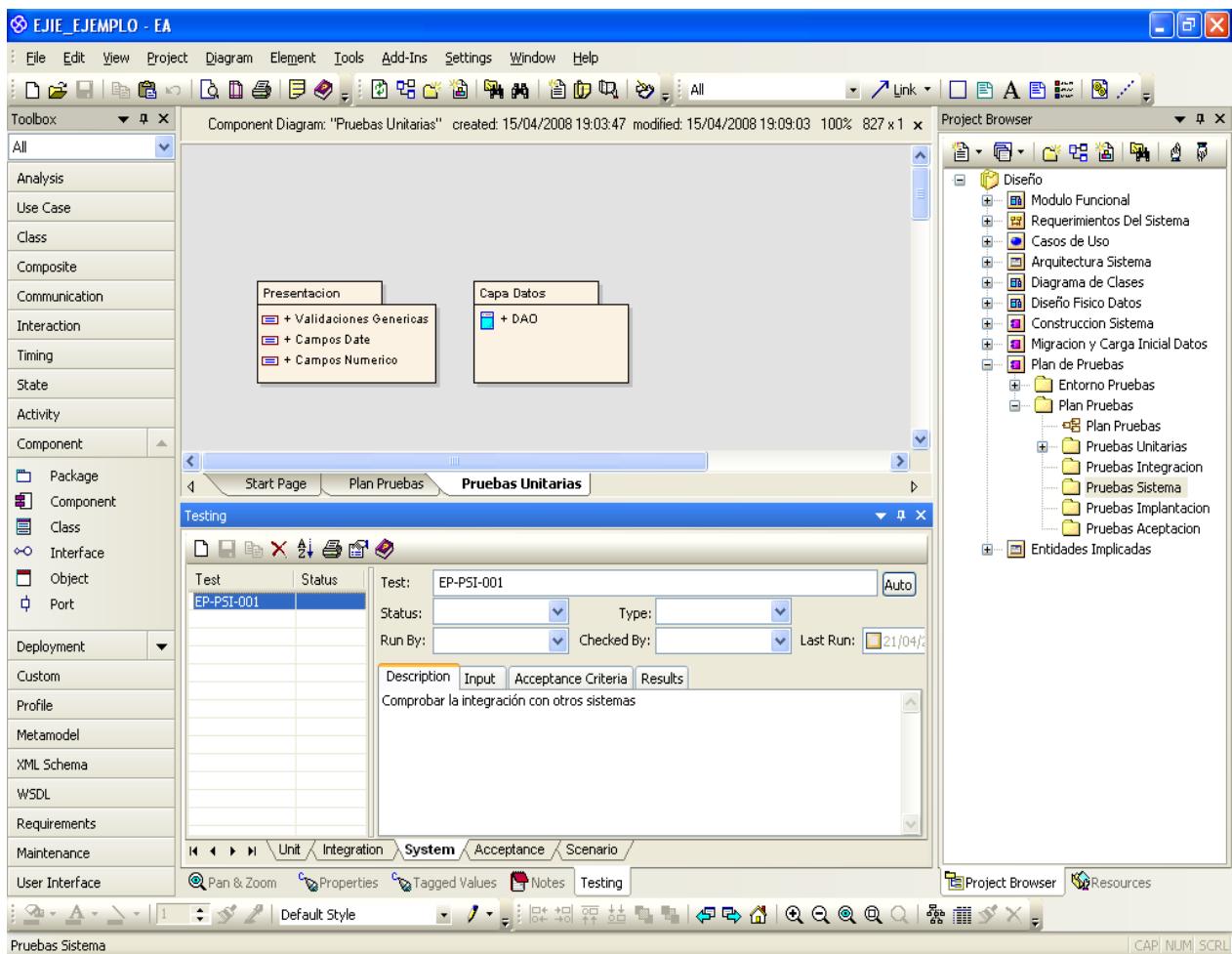
### Pruebas de Integración

Los tests se crearán en la pestaña de Integration



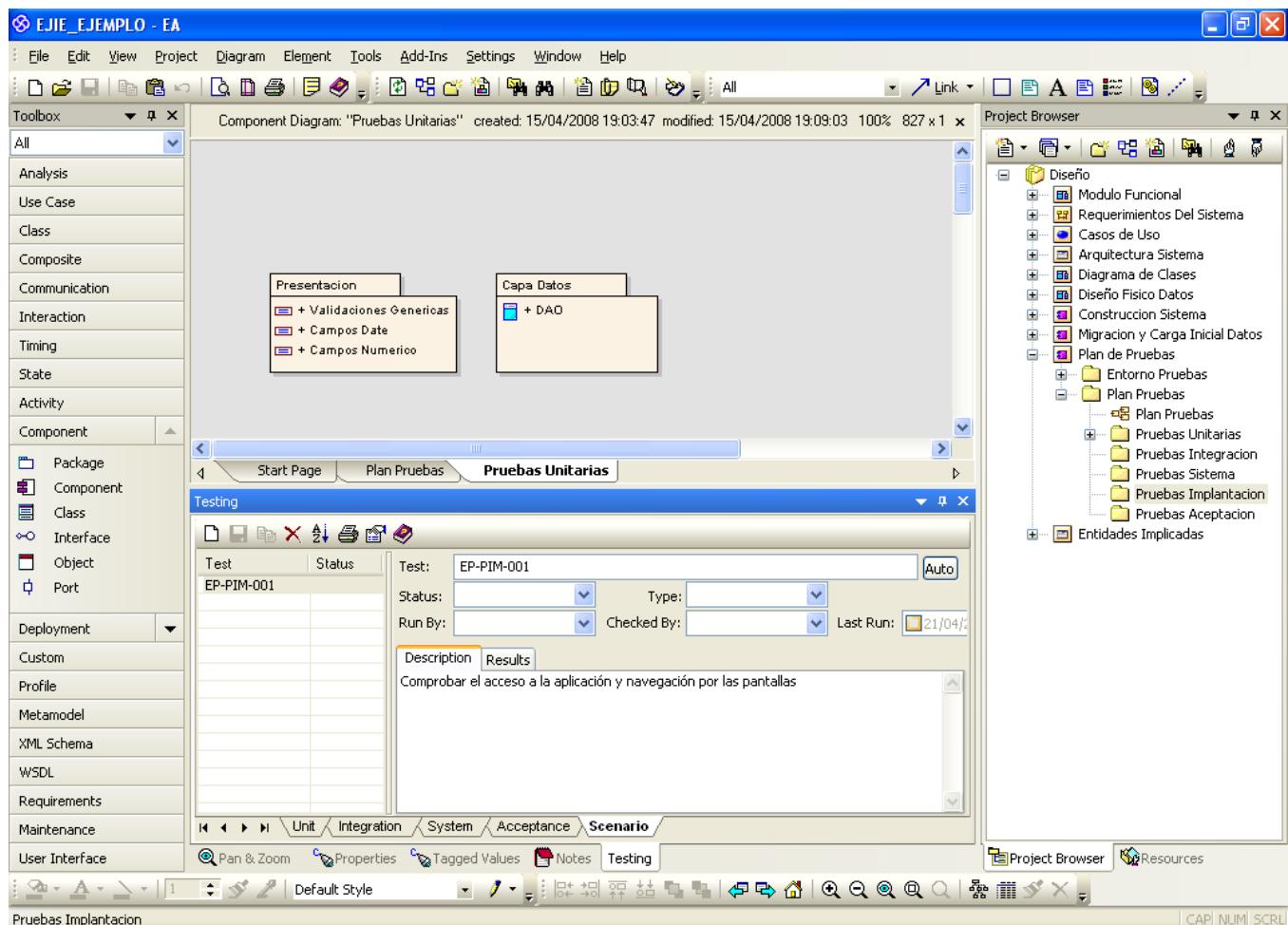
## Pruebas de Sistema

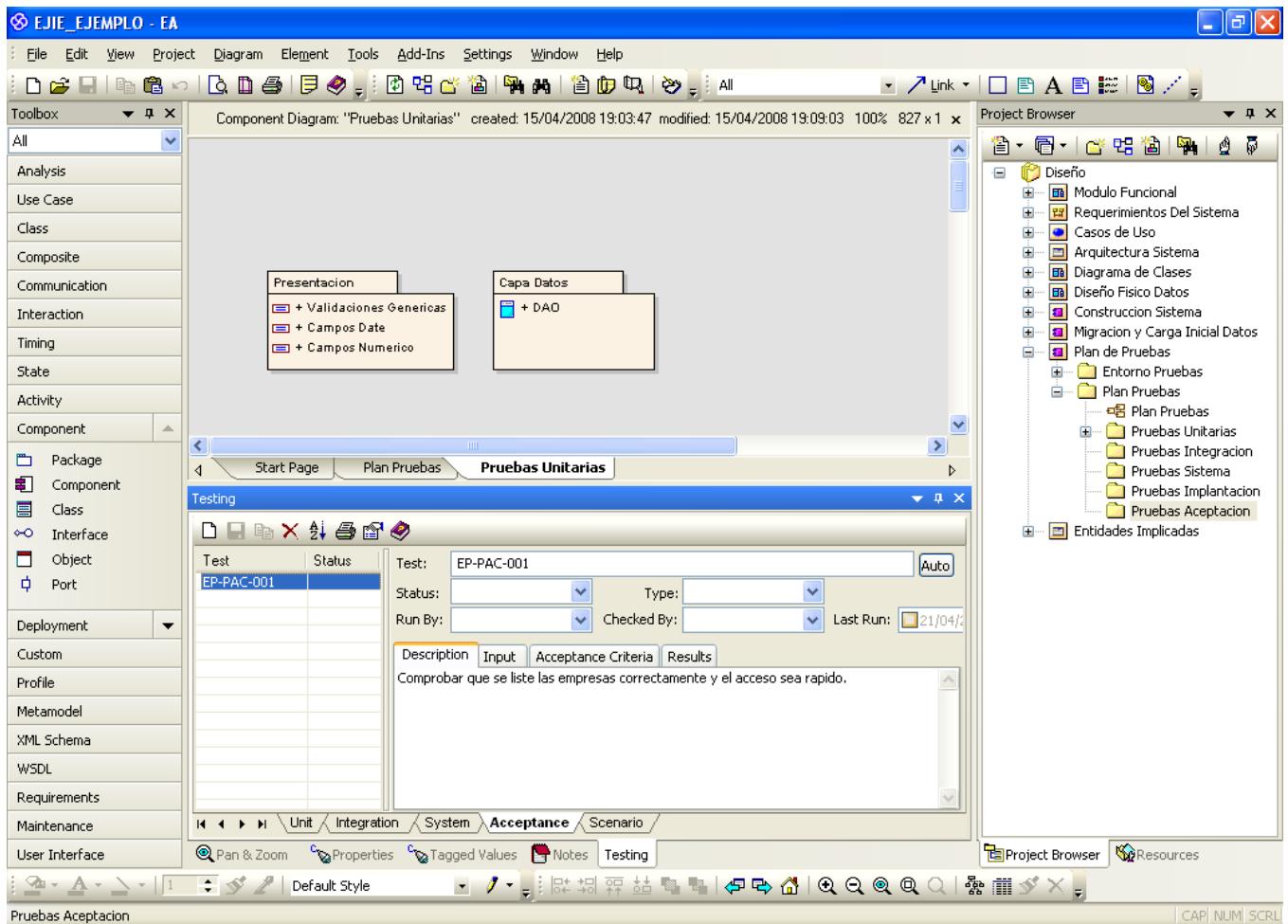
Los tests se crearán en la pestaña de System



## Pruebas de Implementación

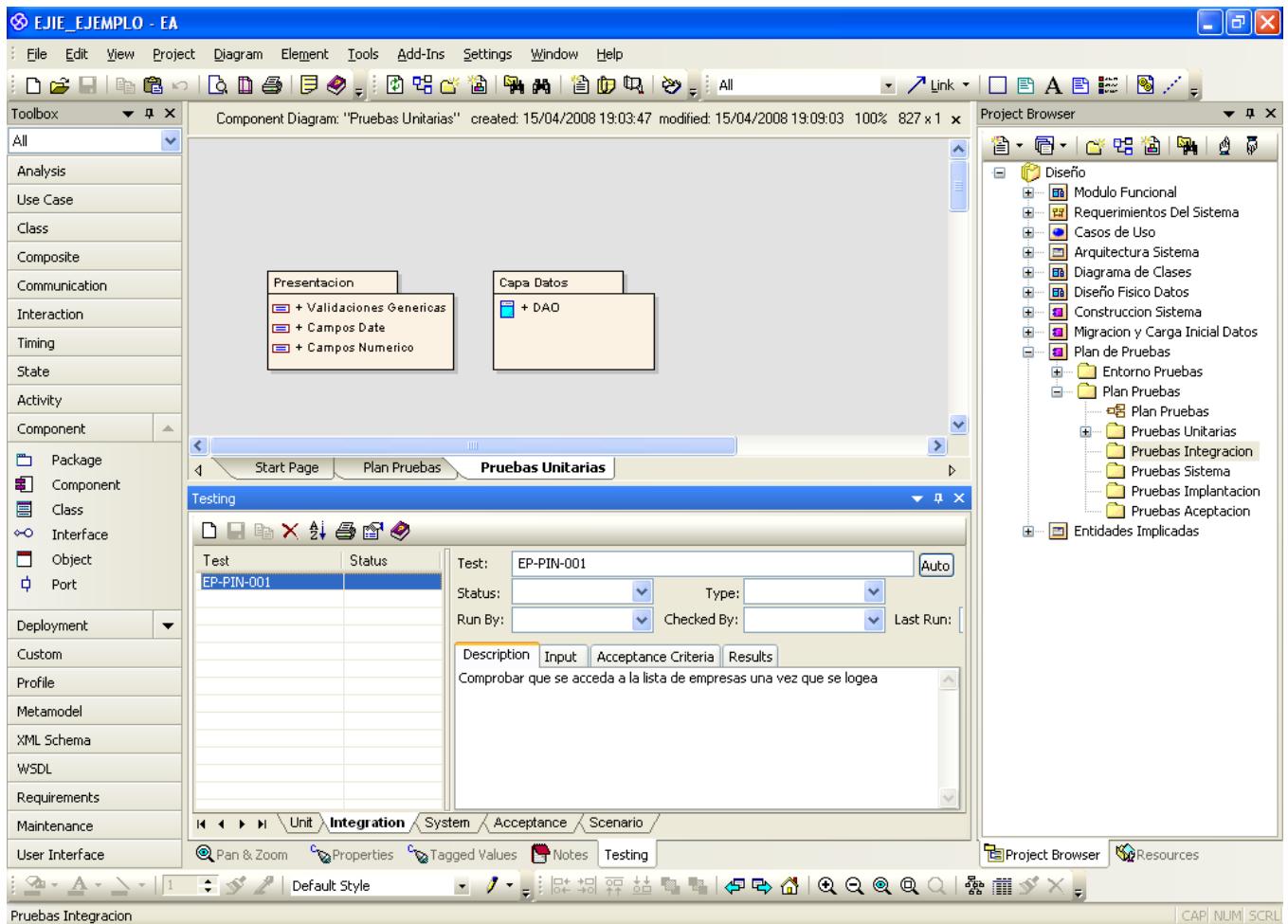
Los tests se crearán en la pestaña de Scenario





## Pruebas de Aceptación

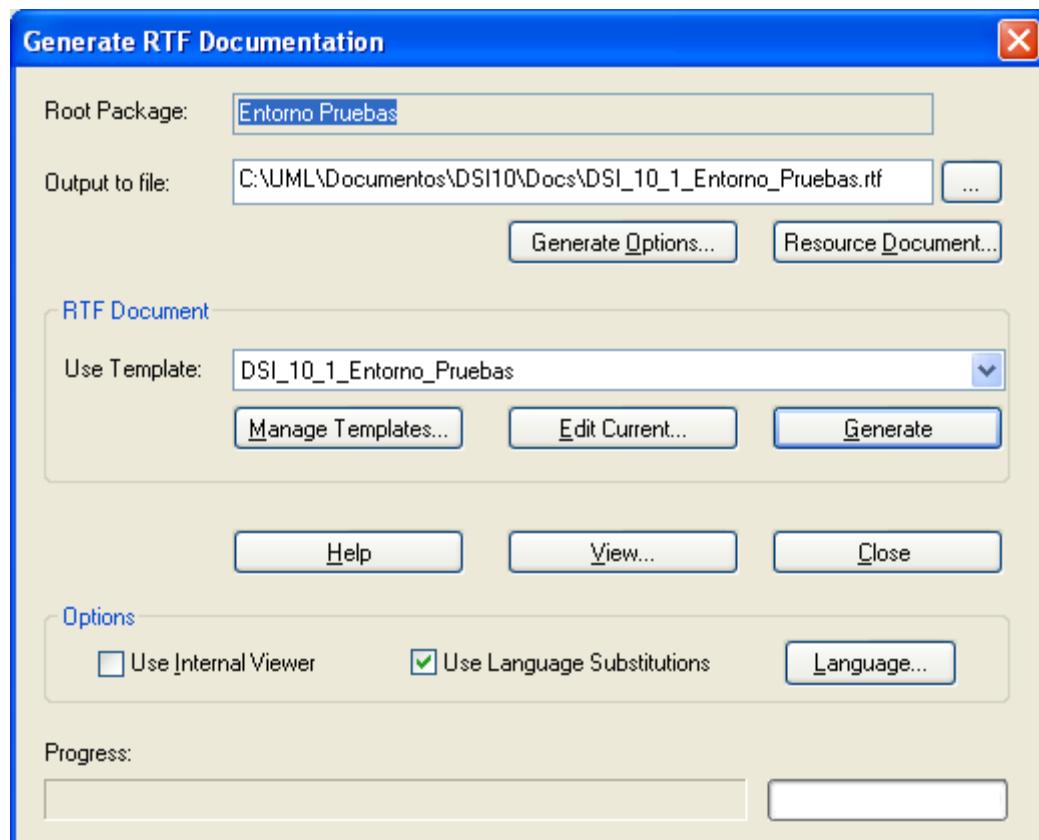
Los tests se crearán en la pestaña de Acceptance



### 5.10.2. Generación Documentación

Los accesos directos que se han de ejecutar para la creación de los documentos intermedios son las siguientes

#### DSI\_10\_1\_Entorno\_Pruebas

**DSI\_10\_2\_Plan\_Pruebas**

