



Guía de desarrollo, Anexo 03.03

Guía de gestión de proyectos en ALM 16.0.1

Autor: Oficina de Pruebas

GERENCIA INFORMÁTICA
JOSEFA VALCÁRCEL, 44
28027-MADRID



Índice General

1	INTRODUCCIÓN	4
1.1	OBJETIVO	4
1.2	AUDIENCIA.....	4
1.3	GLOSARIO.....	4
2	MÓDULOS DE LA APLICACIÓN	4
2.1	MODULO MANAGEMENT	4
2.1.1	<i>Releases</i>	5
2.1.1.1	Objetivo	5
2.1.1.2	Workflow de una Release	6
2.1.1.3	Estructura.....	7
2.1.1.4	Evolutivos	8
2.1.1.5	Subsistemas.....	8
2.1.1.6	Crear ciclos de otro ya existente	10
2.1.1.6.1	¿Cuándo usar esta opción?	13
2.2	MÓDULO REQUIREMENTS	14
2.2.1	<i>Objetivo</i>	14
2.2.2	<i>Estructura General</i>	14
2.2.3	<i>Información común</i>	17
2.2.4	<i>Requisitos</i>	19
2.2.4.1	Requisitos funcionales	19
2.2.4.2	Requisitos no funcionales	22
2.2.4.2.1	Rendimiento	22
2.2.4.3	Requisitos de interfaces	24
2.2.4.3.1	Interfaces de usuario.....	24
2.2.4.3.1.1	RNFI-YY:.....	24
2.2.4.3.2	Interfaces de/con otros sistemas: Ofrecidas.....	24
2.2.4.3.2.1	RNFI-XX:.....	24
2.2.4.3.3	Interfaces de/con otros sistemas: Consumidas.....	24
2.2.4.3.3.1	RNFI-XY:.....	24
2.2.4.3.4	Interfaces de comunicaciones	25
2.2.4.3.4.1	RNFI-ZZ:.....	25
2.2.4.1	RNFs.....	25
2.2.5	<i>Trazabilidad de Requisitos</i>	28
2.2.5.1	Requisitos Funcionales y de Rendimiento	28
2.2.5.1	RNFs.....	33
2.3	MÓDULO TEST PLAN	35
2.3.1	<i>Objetivo</i>	35
2.3.2	<i>Estructura</i>	35
2.3.3	<i>Ubicación ficheros de datos de entrada para auditorías</i>	45
2.3.3.1	Fichero XLS de pruebas funcionales.....	46
2.3.3.2	Ficheros de pruebas de rendimiento.....	46
2.3.3.3	Otros ficheros.....	48
2.3.3.3.1	Pruebas de rendimiento	49
2.3.3.3.2	Pruebas funcionales	49
2.3.4	<i>Casos de prueba</i>	51
2.3.4.1	Normativa de ubicación de casos de prueba.....	51
2.3.4.2	Llamada a casos de prueba (Call to test).....	51
2.3.4.3	Tipos de casos de prueba	53
2.3.4.3.1	Casos de prueba de rendimiento	53
2.3.4.3.1.1	Reglas de Nomenclatura	54
2.3.4.3.1.2	Ejemplo	54
2.3.4.3.1.3	Pasos a realizar	55
2.3.4.3.2	Casos de prueba funcionales (Manuales).....	55
2.3.4.3.2.1	Reglas de Nomenclatura	55



2.3.4.3.2.2	Ejemplo	55
2.3.4.3.2.3	Pasos a realizar	57
2.3.4.3.3	Casos de prueba funcionales (Automáticas)	59
2.3.4.3.3.1	Reglas de Nomenclatura	59
2.3.4.3.3.2	Ejemplo	59
2.3.4.3.3.3	Pasos a realizar	60
2.3.4.3.4	Componentes comunes de casos de prueba funcionales manuales	61
2.3.4.3.4.1	Reglas de Nomenclatura	61
2.3.4.3.4.2	Ejemplo	62
2.3.4.3.4.3	Pasos a realizar	63
2.3.4.4	Documentación de casos de prueba	64
2.3.4.4.1	Detalles de un caso de prueba de rendimiento	64
2.3.4.4.2	Detalles de un caso de prueba funcional	65
2.3.4.4.3	Diseño de pasos	67
2.3.4.4.3.1	Anexar capturas a los pasos	68
2.3.4.4.4	Attachments	69
2.3.4.4.5	Req. Coverage	69
2.3.4.5	Abrir Scripts Vugen desde ALM	71
2.4	MÓDULO TEST LAB	73
2.4.1	Objetivo	73
2.4.2	Creación de una batería de pruebas	74
2.4.3	Ubicación ficheros de datos de entrada para ejecuciones	76
2.4.3.1	Pruebas funcionales Manuales	76
2.4.3.2	Pruebas funcionales Automatizadas	76
2.4.3.3	Pruebas de carga	77
2.4.3.4	Pruebas de sobrecarga	78
2.4.3.5	Pruebas escalabilidad	78
2.4.3.1	Pruebas seguridad	78
2.4.3.2	Fichero .ZIP	78
2.4.3.3	Varias ejecuciones en una misma build	79
2.5	MÓDULO DEFECTS	79
2.5.1	Objetivo	79
2.5.2	Visualización de defectos en una release-ciclo	80
2.5.3	Modificación de defectos	83
2.5.4	Añadir comentarios	85
3	GESTIÓN DE EVOLUTIVOS	87
3.1	DESCRIPCIÓN	87
3.2	EJEMPLO	88
3.2.1	Módulo Management	89
3.2.2	Módulo Requirements	90
3.2.3	Módulo Test Plan	93
3.3	CASO ESPECIAL – EVOLUTIVO SIN CAMBIOS DE REQUISITOS/CASOS DE PRUEBA	95
3.4	COMPROBACIÓN	97



1 Introducción

1.1 Objetivo

El presente Documento es la Guía de Gestión de Proyectos de la Herramienta **Application Lifecycle Management (ALM)** en la versión 16.0.1.

Esta Guía **no es un Manual** de **Application Lifecycle Management (ALM)**. Los Manuales Oficiales los podemos encontrar en el menú ayuda de los diferentes productos de MicroFocus/OpenText.

1.2 Audiencia

Este documento está dirigido a todas las personas que colaboren en labores relacionadas con la gestión, desarrollo, auditoría, implantación y explotación de los sistemas de información de la gerencia de informática de la Dirección General de Tráfico.

1.3 Glosario

Los términos y acrónimos que se utilizan en este documento y en el resto de documentos de la guía se encuentran recogidos por orden alfabético en el Anexo 30. Glosario con el objetivo de facilitar su lectura y comprensión

2 Módulos de la Aplicación

2.1 Modulo Management





2.1.1 Releases



2.1.1.1 Objetivo

Desde el módulo **Management** se realiza la gestión de las Releases del proyecto.

Una Release representa un conjunto de una o varias versiones de la aplicación informática que compartan funcionalidades. Siguiendo el Anexo 38 Normativa de Versionado, una Release estará compuesta por todas las versiones que compartan los dígitos x.y.

Dentro de las Releases están los Ciclos que se corresponden con cada una de las versiones definidas por x.y.z que se pretende distribuir. Por último tenemos las Builds, que son los productos de desarrollo (ejecutable) que serán objeto de las pruebas, que siguen la nomenclatura especificada en el “Anexo 10.0 Guía Técnica de uso, estructura y nombrado en SVN”. Atendiendo al nombre de la Build podremos saber si se trata de una versión de Preproducción (con el sufijo –rc, –beta, o sin sufijo), o de Desarrollo (sufijo –SNAPSHOT).

Durante la creación de la aplicación el equipo de desarrollo irá elaborando de manera paralela la documentación asociada (Requerimientos, Planes de Pruebas, Scripts de Rendimiento) que será revisada por la Oficina de Pruebas previamente a la ejecución de las pruebas solicitadas. Por otro lado los Ciclos no contendrán cambios en la funcionalidad de la aplicación, sino correcciones de errores técnicos o de funcionalidades detectados en la versión, y tendrán siempre una fecha de inicio y de fin. Para poder poner la fecha de inicio y fin, será necesario pulsar botón derecho sobre el ciclo y en el menú contextual, seleccionar la opción “Reschedule”. Tanto la Release como el Ciclo se deben crear en ALM a través del módulo “Management – Releases”. Las Builds las creará la Oficina de Pruebas a petición de desarrollo cuando se vayan a ejecutar pruebas.

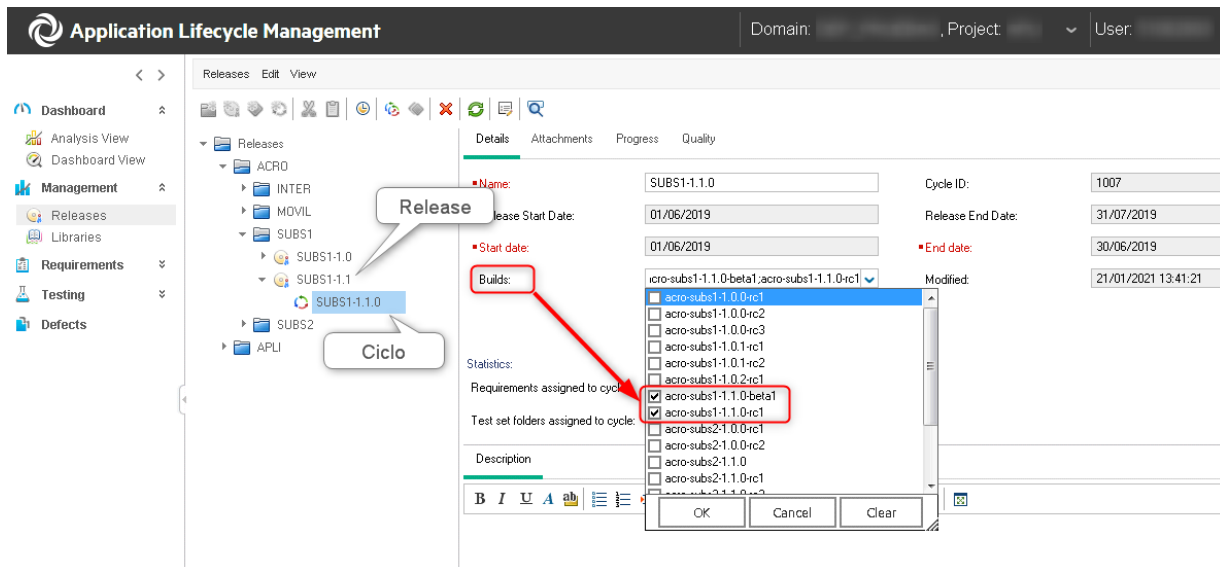
Así por ejemplo, si Desarrollo está trabajando en código en la versión objetivo **acro-sub1-1.1.0**, para lo cual libera varias builds (acro-sub1-1.1.0-rc1 y acro-sub1-1.1.0-beta1), en ALM debería tener las siguientes entidades:



Release: SUBS1-1.1

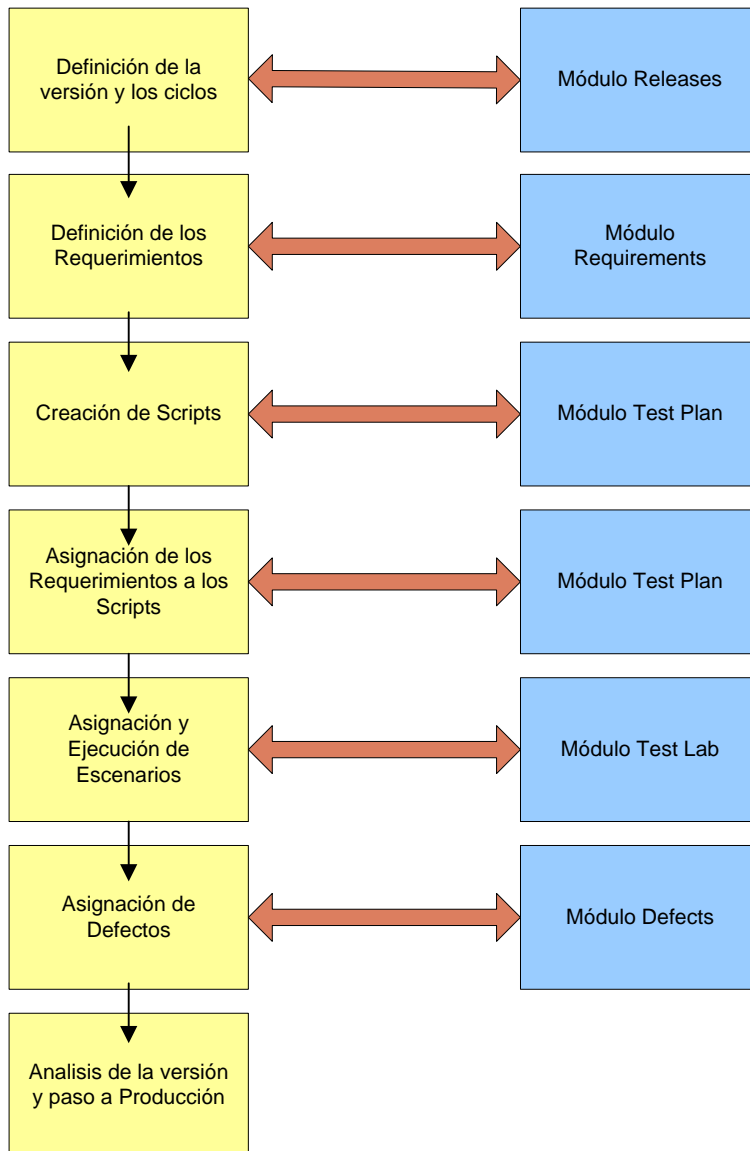
Cycle: SUBS1-1.1.0

Builds: acro-sub1-1.1.0-rc1, acro-sub1-1.1.0-beta1 (nota: las Builds las crea la Oficina de Pruebas al recibir petición de ejecución)



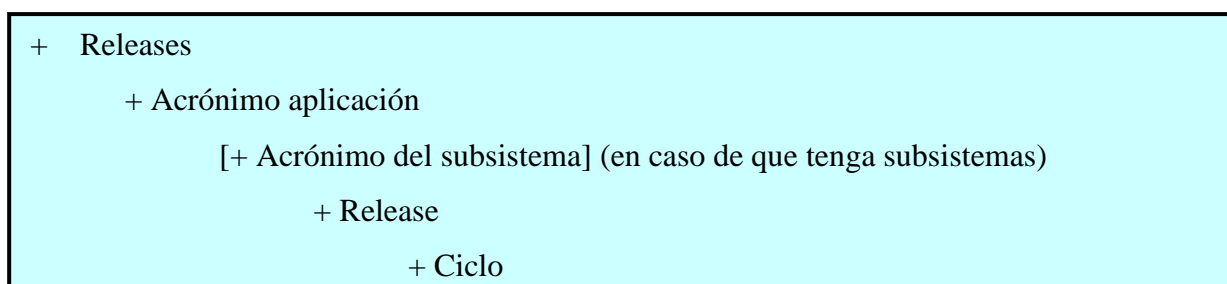
2.1.1.2 Workflow de una Release

Pasos de trabajo a realizar en la liberación de una Release y su correspondencia con los módulos de ALM.



2.1.1.3 Estructura

La estructura general a utilizar en el módulo de Management será la siguiente:



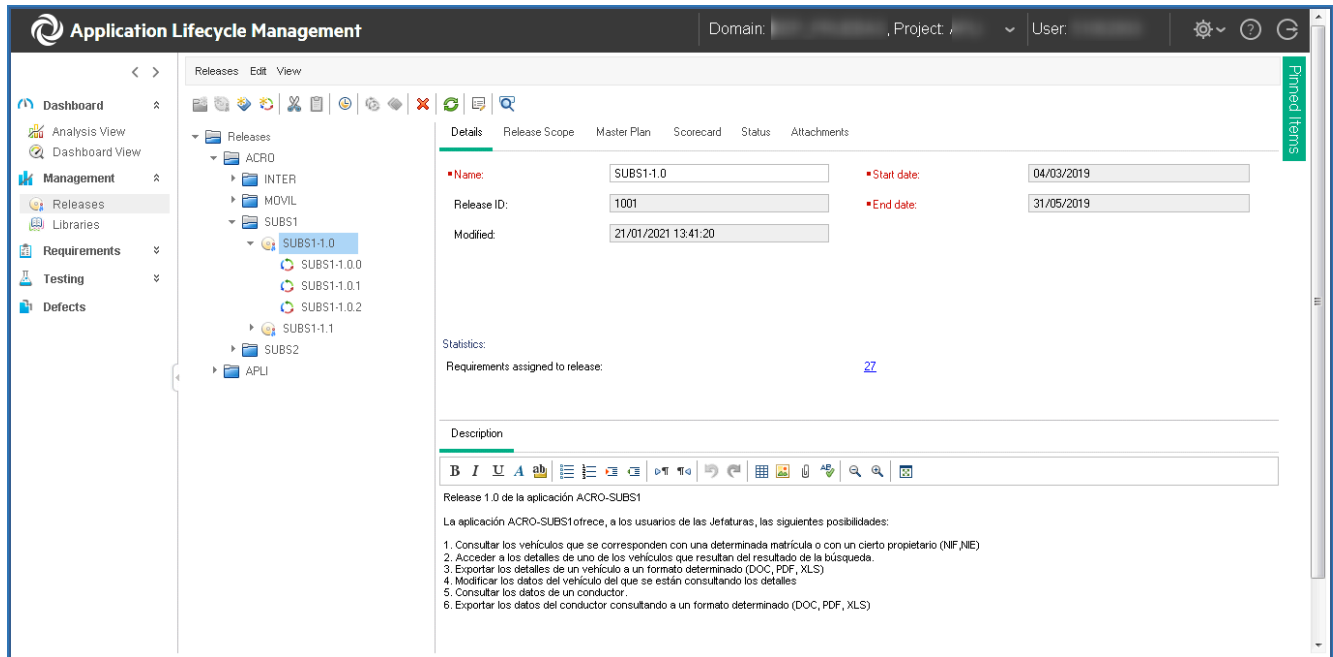


Ilustración 1. Estructura de Releases de proyecto con subsistemas en el módulo Management

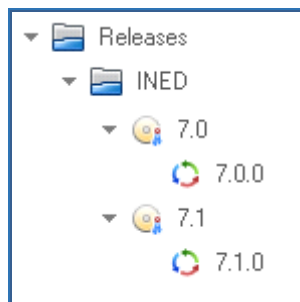


Ilustración 2. Ejemplo de estructura de Releases de proyecto sin subsistemas en el módulo Management

2.1.1.4 Evolutivos

Ver sección “3 Gestión de evolutivos”.

2.1.1.5 Subsistemas

En el caso de que se trate de un proyecto con subsistemas se debe crear una carpeta por cada subsistema existente (esto aplica para todos los módulos de ALM). Las Releases, Ciclos y Builds deben incluir el nombre del subsistema en el nombre.

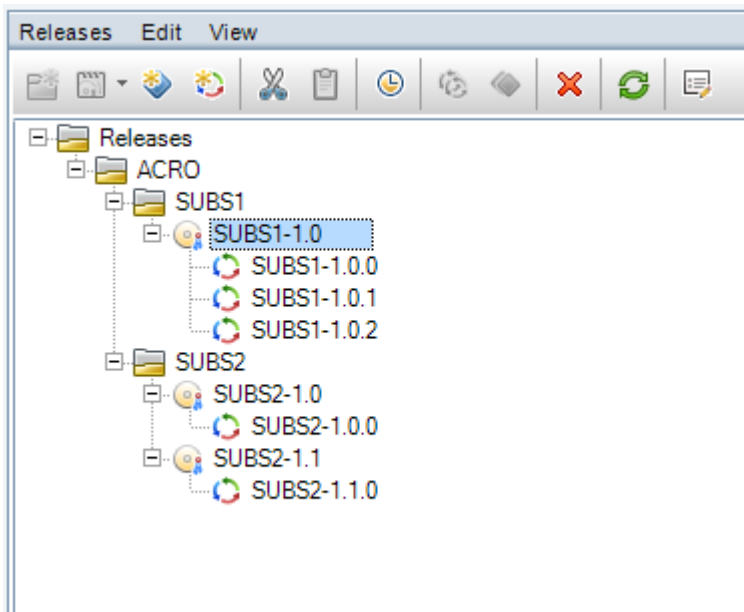


Ilustración 3. Estructura de subsistemas en el módulo Management, con releases menores

En lo que respecta a los requisitos, si existieran requisitos comunes a varios subsistemas deberán duplicarse en ALM y crear un requisito por cada subsistema.

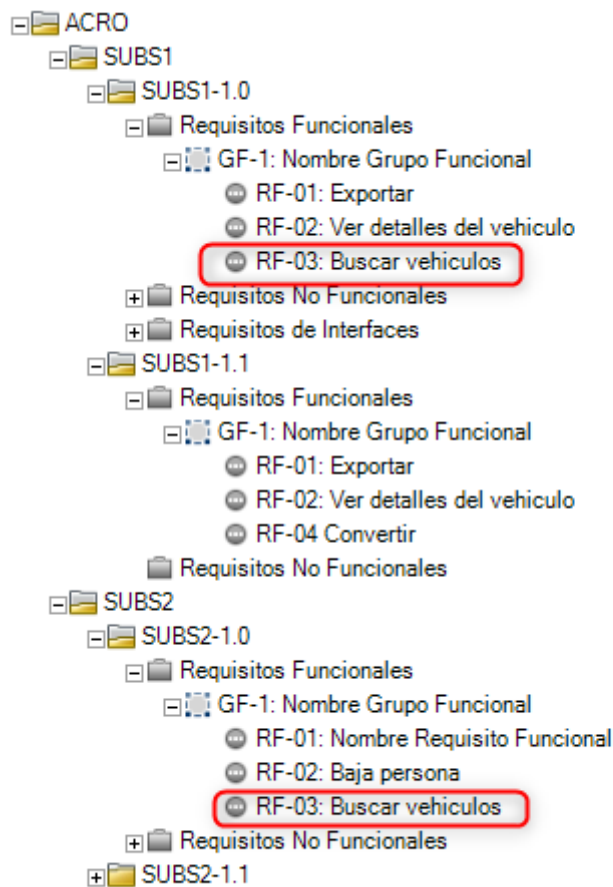


Ilustración 4. Requisitos comunes a subsistemas distintos. Módulo Requirements

2.1.1.6 Crear ciclos de otro ya existente

Desde el módulo **Management** se realiza la gestión de las releases y ciclos de los proyectos. Cuando ya tenemos un ciclo creado ([SUBS-]1.0.0) y queremos crear un nuevo ciclo en el que aparezcan asociados todos los requisitos la mejor opción es:

1. Seleccionar el ciclo último que haya sido validado
2. Pulsar botón derecho y seleccionar la opción “*Duplicate Cycle CTRL+D*”

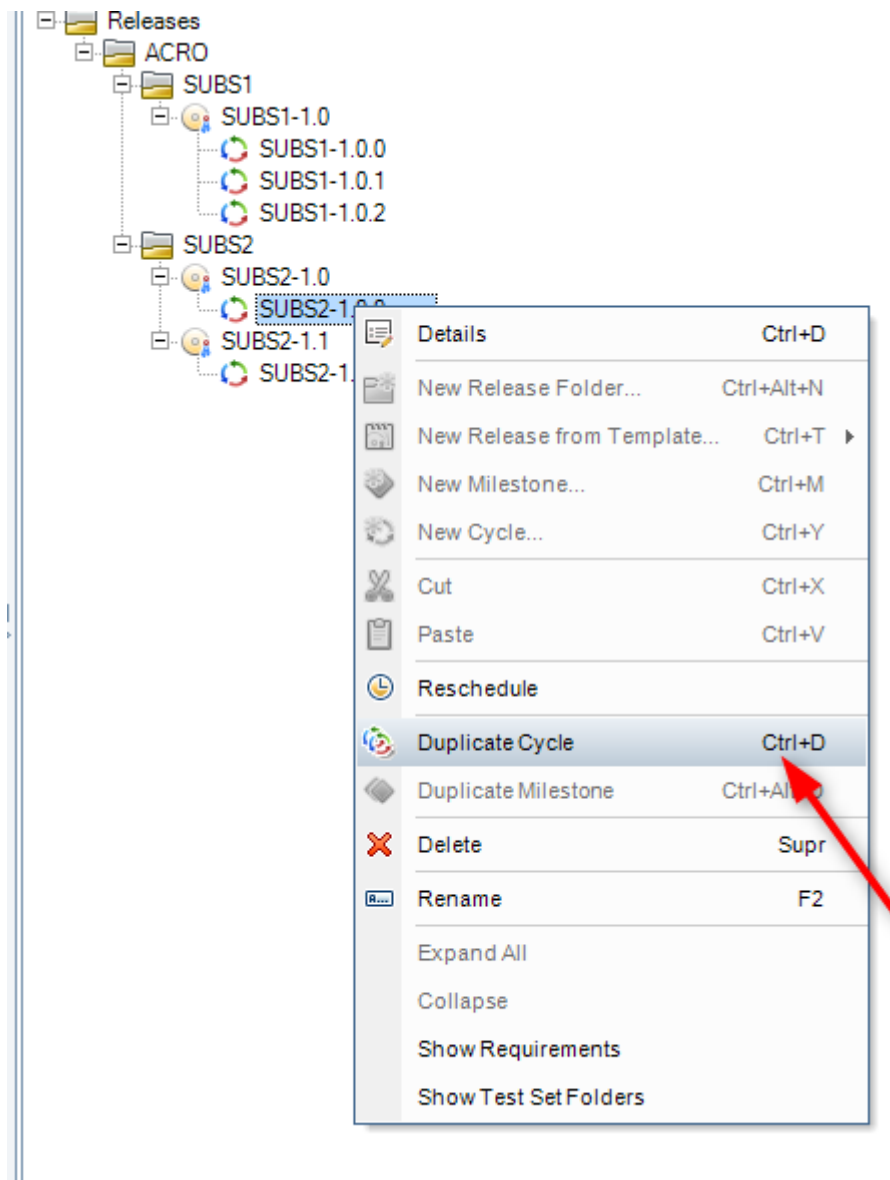


Ilustración 5. Duplicar Ciclo

Se creará un nuevo ciclo con el mismo nombre y finalizado en “_Copy_1”. Ahora se deberá proceder a renombrarlo con el nombre del nuevo ciclo.

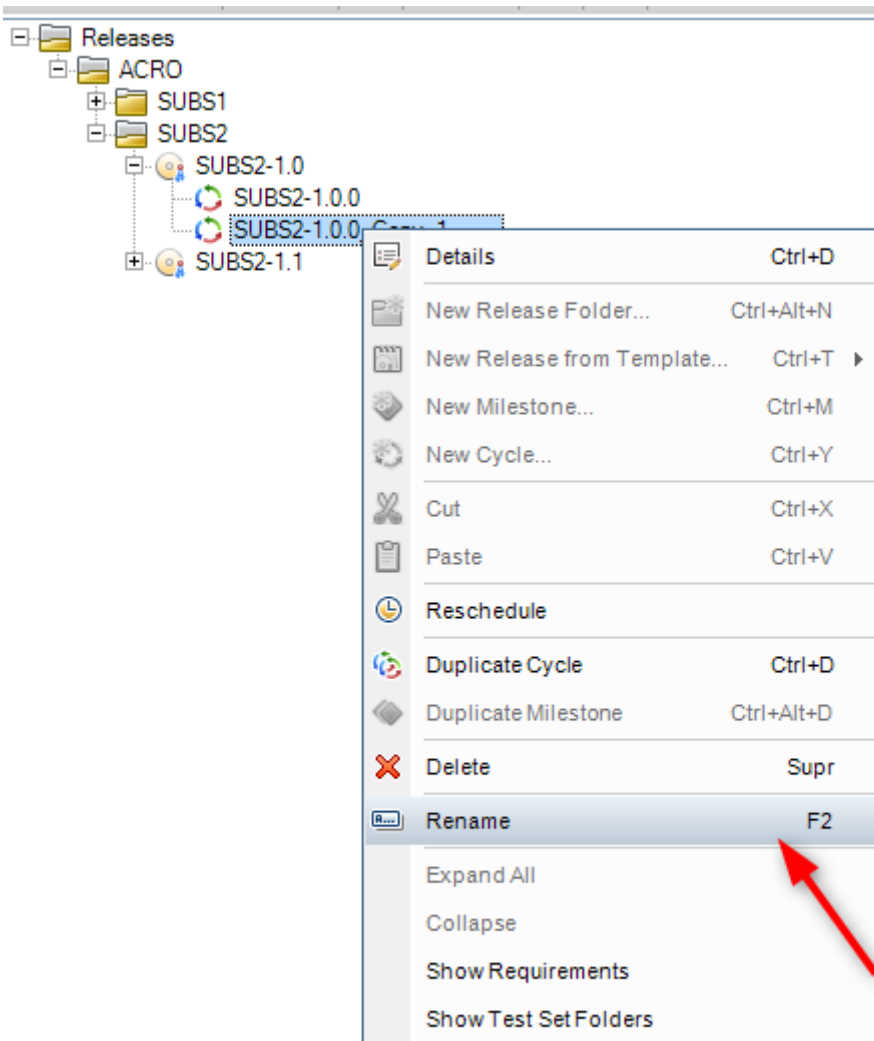
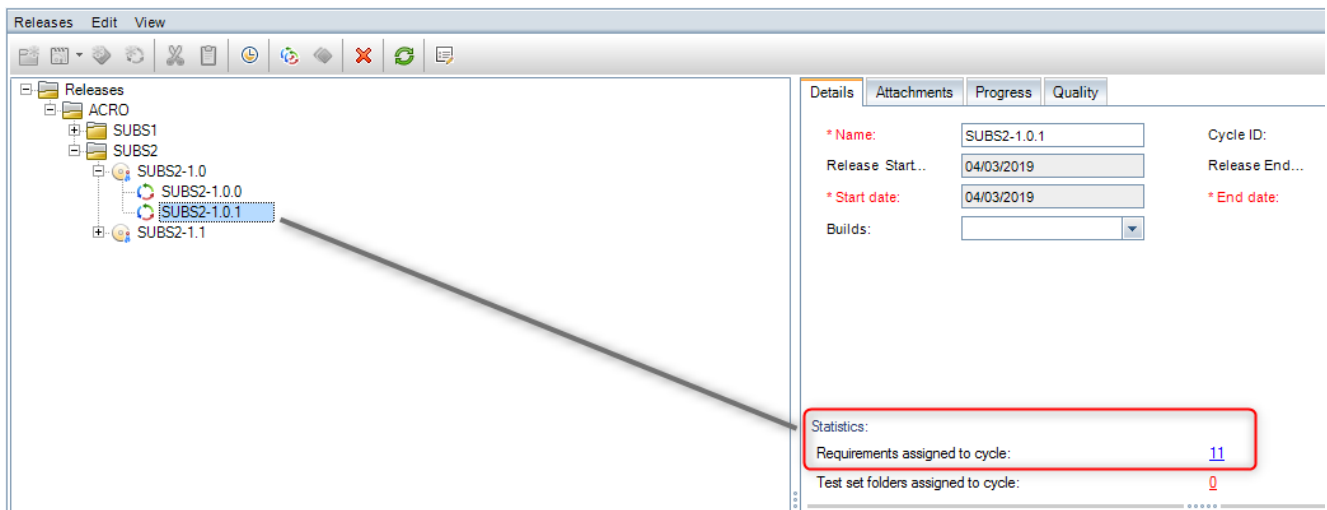
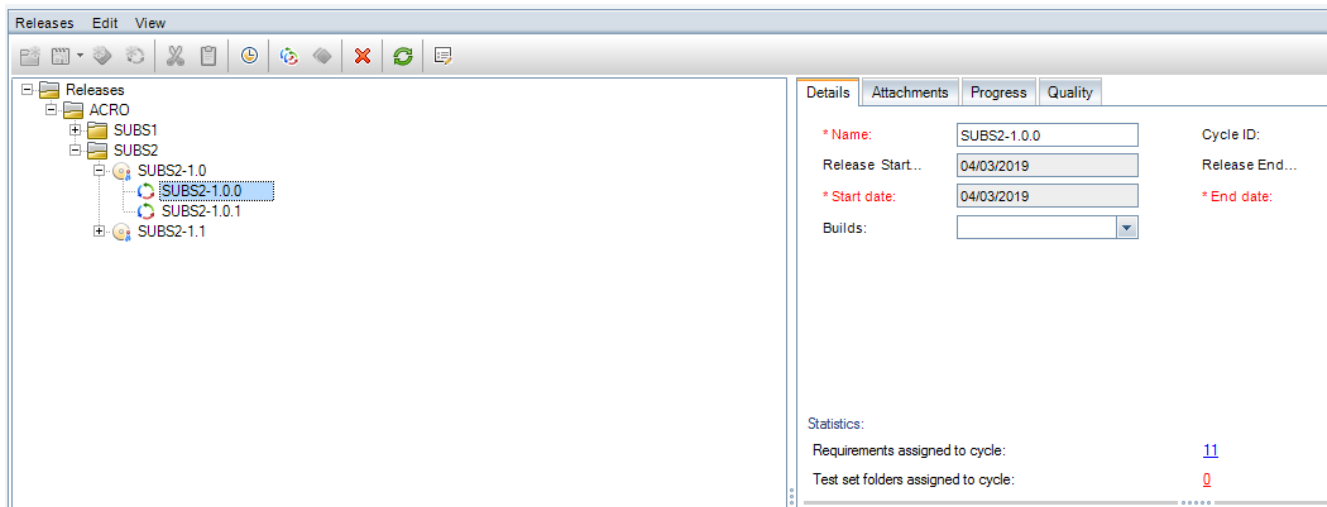


Ilustración 6. Renombrar Ciclo

4. Seleccionar el nuevo ciclo creado y finalizado en “_Copy_1”.
5. Pulsar botón derecho y seleccionar la opción “*Rename F2*”

Verificar que siguen asociados el mismo número de requisitos.



2.1.1.6.1 ¿Cuándo usar esta opción?

En el caso de tener los requisitos y/o plan de pruebas auditados correctamente para un ciclo “[SUBS-]1.0.0” y al ir a ejecutar las pruebas la versión desplegada es “acro-(subs-)-1.0.1-beta.1” entonces se tendría que crear el ciclo “[SUBS-]1.0.1”. De este modo en Test Lab la Oficina de Pruebas puede asociar la ejecución al nuevo ciclo correspondiente a la versión desplegada.

Se entiende que un incremento en la “z” no supone una modificación de requisitos, por lo que no sería necesario volver a auditar los requisitos ni el plan de pruebas.

En el caso de que en el nuevo ciclo (con incremento de “z”) hubiera un cambio de requisitos, lo primero sería plantearse que se debería haber incrementado la “y” y si ya se hubiera desplegado habría



que realizar en ALM las modificaciones de requisitos correspondientes a ese nuevo ciclo y pedir la auditoría de requisitos y plan de pruebas previamente a la ejecución.

2.2 Módulo Requirements

2.2.1 Objetivo



Permite la gestión de Requisitos (Funcionales, de Rendimiento, de Interfaz y RNFs) y su relación con las Pruebas, es decir, **el grado de cobertura de los Requisitos frente a las Pruebas**.

Los requisitos en árbol es una representación gráfica de su especificación, mostrando la relación jerárquica entre las diferentes necesidades. De cada requisito se puede determinar su cobertura y su riesgo.

ALM se ha escogido en DGT como el repositorio principal de todos los requisitos de la aplicación y desde aquí se podrá exportar en un documento Word o PDF con todos aquellos que sean aplicables a las diferentes releases. También se podrán exportar en formato Excel de cara a elaborar la Matriz de Trazabilidad. El procedimiento de exportación se detalla en el Anexo 03.02.

En los siguientes puntos se explica cómo se ha de estructurar la información de requisitos en ALM.

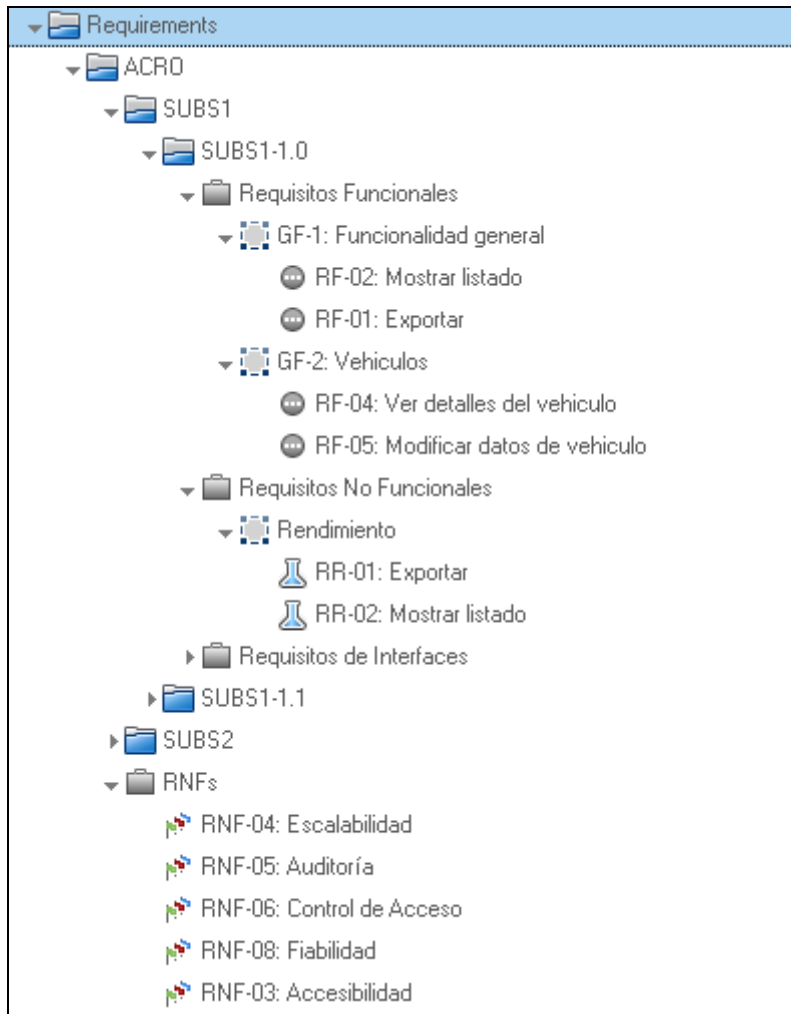
2.2.2 Estructura General


La estructura general a utilizar en el módulo de **Requeriments** será la siguiente:



- + Requirements
 - + Nombre Aplicación
 - + [Subsistema] (en caso de que tenga subsistemas)
 - + Release
 - + Requisitos Funcionales
 - + Grupo Funcional
 - + Requisito Funcional
 - + Requisitos No Funcionales
 - + Rendimiento
 - + Requisito Rendimiento
 - + Requisitos de Interfaces
 - + Interfaces de usuario
 - + Interfaces de/con otros sistemas
 - + Interfaces ofrecidas
 - + Interfaces consumidas
 - + Interfaces de comunicaciones
- + RNFs (resto requisitos no funcionales)

Debe quedar una estructura como la siguiente.



En ALM tenemos el botón “New Requirement”  con el que podremos ir creando la estructura anterior. Los tipos de requisitos permitidos son:

- *Functional*
- *Performance*
- *No Functional*
- *Folder (Requisito de Carpeta)*
- *Business (similar a Carpeta para agrupar los requisitos Funcionales, no Funcionales, de Interfaces y RNFs)*
- *Group (Requisito de Grupo)*



2.2.3 Información común

Situándonos en la estructura anterior al nivel de la “Release” (ej: SUBS1-1.0) (requisito tipo “Folder”) introduciremos toda la información común aplicable a la aplicación (o subsistema) en la release en curso dentro del campo “Description” siguiendo la siguiente estructura (sustituir el texto en azul por la información adecuada de nuestra aplicación):

1. Objeto

El propósito de este documento es presentar una descripción detallada de los requisitos que debe satisfacer el **producto x** que realiza la funcionalidad **explicar qué funcionalidad va a englobar** con la finalidad de saber qué se requiere del sistema y establecer el alcance.

2. Audiencia

La audiencia a la que va dirigido es **especificar la audiencia a la que va dirigido**.

3. Alcance

****Identificación del producto(s) incluyendo su nombre**

Consistencia con definiciones similares de documentos de mayor nivel que puedan existir (ej. Pliego de Descripciones Técnicas, identificación de necesidades etc.).

4. Términos y acrónimos

**** Definición de todos los términos, abreviaturas y acrónimos necesarios para interpretar apropiadamente este documento**

4.1 Términos

En este apartado se incluyen, organizados por orden alfabético, los términos que se utilizan en el propio documento para facilitar su lectura y comprensión.

- Excepción: Cualquier error capturado y manejado por la aplicación durante su ejecución

4.2 Acrónimos

En este apartado se incluyen, organizados por orden alfabético, los acrónimos que se utilizan en el propio documento para facilitar su lectura y comprensión.

- DGT: Dirección General de Tráfico

5. Referencias

****Relación completa de todos los documentos relacionados en la especificación de requisitos de software, identificando de cada documento el título, referencia (si procede), fecha y organización que lo proporciona. Como mínimo debe incluirse en esta lista el pliego de prescripciones técnicas**

6. Descripción general



****Indicar si es un producto independiente o parte de un sistema mayor. En este punto se puede incluir un diagrama que identifique las relaciones con el entorno para facilitar la comprensión.**

6.1 Perspectiva del producto

****** Visión general tanto interna como externa del producto, modos de operación si los hubiera y cualquier condición del entorno o contexto de la aplicación que deba ser tomada en cuenta más adelante en esta especificación etc.

6.2 Necesidades del producto a satisfacer

****** Resumen de las funcionalidades principales que el producto debe realizar, sin entrar en información de detalle.

En ocasiones la información de esta sección puede tomarse de un documento de especificación del sistema de mayor nivel (ej. Requisitos del sistema, Memoria Técnica o del Pliego de Prescripciones Técnicas).

Las funcionalidades deben estar organizadas de manera que el usuario o cualquier interlocutor pueda entenderlo. Para ello se pueden utilizar métodos textuales o gráficos.

Se podrán organizar las necesidades en apartados y se identificarán obligatoriamente con un identificador NEC-XX que coincidirá si se han definido en las matrices de trazabilidad.

6.3 Características de los usuarios

****** Descripción de los usuarios, identificando los roles que interactúan con el sistema.

Se podrán organizar los actores en apartados y se identificarán obligatoriamente con un identificador ACT-XX que coincidirá si se han definido en las matrices de trazabilidad.

6.4 Sistemas externos

****** Indicar si hay que integrar el producto con otros productos de software, sistemas, subsistemas o aplicaciones.

Se podrán organizar los actores de tipo sistema en apartados y se identificarán obligatoriamente con un identificador ACT-XX que coincidirá si se han definido en las matrices de trazabilidad.

7 Diagrama de contexto

****** Incluir en el diagrama los usuarios y sistemas externos con los que interactúa el producto **(Añadir adjunto)**

Añadir a continuación una breve descripción que ilustre lo que se muestra en el diagrama.



Sort By: Name[Ascending]

Name

- Requirements
 - ACRO
 - RNFs
 - SUBS1
 - SUBS1-1.0
 - Requisitos de Interfaces
 - Requisitos Funcionales
 - GF-1: Funcionalidad general
 - RF-01: Exportar
 - RF-02: Mostrar listado
 - GF-2: Vehiculos
 - RF-04: Ver detalles del vehiculo
 - RF-05: Modificar datos de vehiculo
 - Requisitos No Funcionales
 - Rendimiento
 - RR-01: Exportar
 - RR-02: Mostrar listado
 - SUBS1-1.1
 - SUBS2

Details Rich Text Attachments Linked Defects Requirement Traceability Test Coverage Risk Analysis Business M

Name: SUBS1-1.0 **Requirement Type:** Folder

Author: 51062803 **Creation Date:** 04/03/2019

Creation Time: 9:28:11 **Modified:** 15/09/2023 13:40:00

Priority: **Product:**

ReqID: 219 **Reviewed:** Not Reviewed

Target Cycle: **Target Release:** SUBS1-1.0

Description **Comments**

1. Objeto
El propósito de este documento es presentar una descripción detallada de los requisitos que debe satisfacer el producto x que realiza la funcionalidad explicar qué funcionalidad va a englobar con la finalidad de saber qué se requiere del sistema y establecer el alcance.

2. Audiencia
La audiencia a la que va dirigido es especificar la audiencia a la que va dirigido.

3. Alcance
**Identificación del producto(s) incluyendo su nombre
Consistencia con definiciones similares de documentos de mayor nivel que puedan existir (ej. Pliego de Descripciones Técnicas, identificación de necesidades etc.).

4. Términos y acrónimos
** Definición de todos los términos, abreviaturas y acrónimos necesarios para interpretar apropiadamente este documento

4.1 Términos

2.2.4 Requisitos

A continuación se introducirán colgando del nivel “Release” los requisitos funcionales, los no funcionales de Rendimiento y los de interfaces.

Como se explica más abajo (2.2.4.1) también se introducirá una carpeta llamada “RNFs” con todos los RNFs que apliquen colgando de la carpeta de la aplicación.

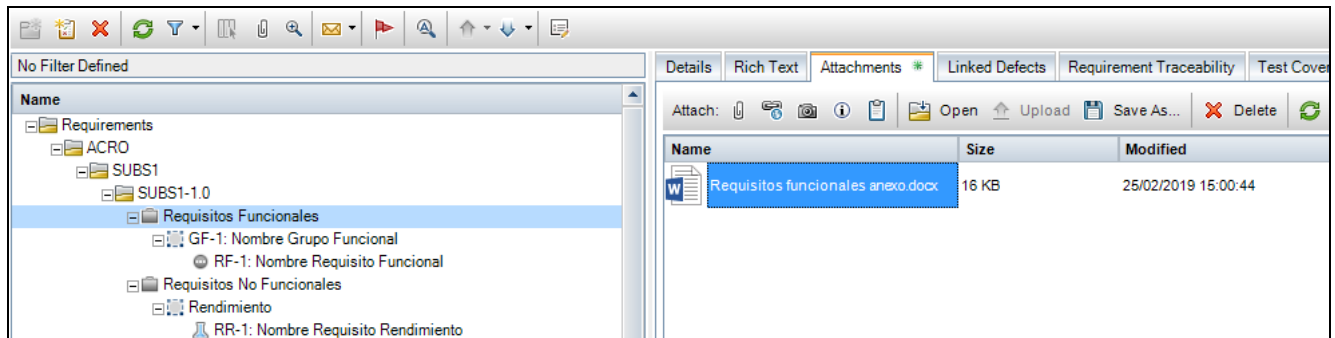
La Oficina de Pruebas solo auditará los requisitos funcionales, los de rendimiento y los RNFs.

2.2.4.1 Requisitos funcionales

A este nivel se introducirá un requisito tipo “**Business**” con el Name = “Requisitos Funcionales” en el cual se introducirá la siguiente información en el campo “Description”:

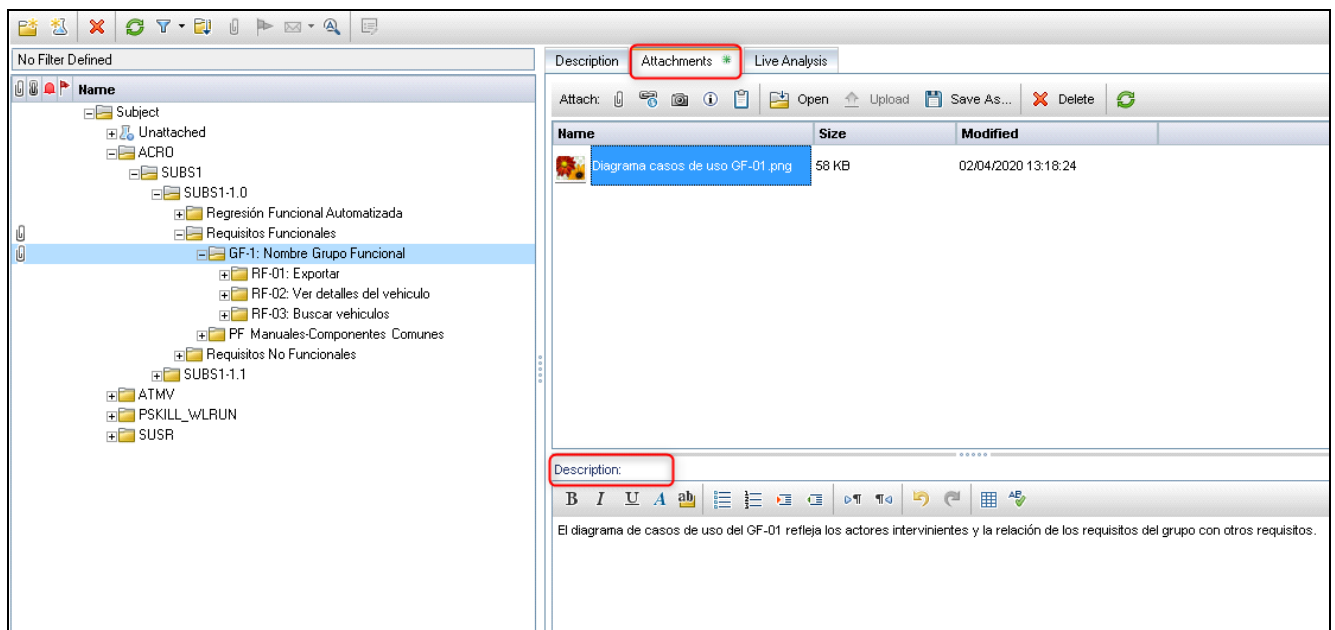
Identificación y descripción de la funcionalidad que realiza el producto.

Si se trata de un producto cerrado y el producto dispone de una especificación de requisitos, se pueden poner referencias directas a esa especificación, así como adjuntar los archivos oportunos en la pestaña “Attachments”.



Se debe organizar este apartado en subapartados por Grupos Funcionales (GF-XX), que se deberán insertar como tipo “**Group**” (Requisito de grupo).

Cada **Grupo Funcional (GF-XX)** debe llevar como adjunto su **diagrama de casos de uso** junto con una breve descripción del mismo como se puede ver en la imagen adjunta.

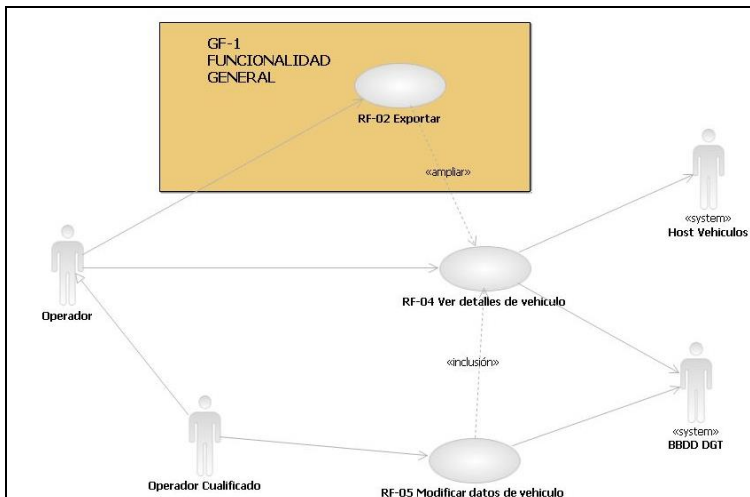


Los diagramas de casos de uso establecen las relaciones entre los diferentes requisitos de un mismo subsistema que luego se reflejarán en la pestaña Requirement Traceability.

Algunas consideraciones importantes sobre los diagramas de casos de uso:

- Debe haber un diagrama por cada GF. En casos de GFs muy extensos se puede admitir que se divida en varios diagramas a fin de facilitar la lectura y comprensión de los mismos. Para ello indicarlo adecuadamente en el Description de los diagramas adjunto en ALM.

- Las líneas de los diagramas no deberían cruzarse ni superponerse ya que eso impide el correcto seguimiento de las trazabilidades. Por ejemplo, en RSA sería usar la opción “Disponer todo”
- En el caso de que un RF esté relacionado con un RF de otro GF, ese RF debe enmarcarse en un rectángulo sombreado en color a fin de indicar que se trata de un RF de otro GF (indicar también a qué GF pertenece). Ejemplo:



- En los diagramas de casos de uso **no** hay que representar la relación entre los RF y los RR, sólo la relación entre RFs.

Por último se introducirán los requisitos funcionales como tipo “**Functional**” y que se identificarán como RF-XX: Nombre del RF.

Los **campos obligatorios** a introducir son:

- Priority (1 a 5). En los requisitos de tipo funcional establece el nivel de detalle del plan de pruebas a realizar. 5 representa la mayor prioridad (en el “Anexo 03.05 Procedimiento de Pruebas Funcionales” se detallan los criterios para la priorización de requisitos). **Al menos un Requisito Funcional de los definidos deberá tener prioridad 5.**
- Target Release
- Target Cycle. Únicamente los requisitos de tipo funcional o rendimiento deben tener asignado el ciclo. Este campo debe estar vacío para los requisitos de carpeta y grupo.
- Description

Dentro del campo “Description” se puede estructurar con la siguiente información:

- Identificación de la funcionalidad o caso de uso



- Descripción de la funcionalidad
- Entradas: información que aporta el actor del caso de uso.
- Salidas: información o mensaje que el sistema le envía al actor del caso de uso al finalizar su ejecución.
- Precondiciones: estado necesario en el que debe encontrarse el sistema para que pueda ejecutarse el caso de uso.
- Postcondiciones: estado alterado en el sistema, tras la ejecución del caso de uso.

Pudiéndose incluir la siguiente información:

- Usuarios y sistemas externos que en él intervienen
- Descripción de los pasos que describen la interacción entre los actores y el sistema.
- Comprobación de validez de las entradas
- Secuencia de operaciones
- Respuesta a situaciones anormales (desbordamientos, comunicaciones, recuperación de errores)
- Relaciones entre entradas y salidas (secuencias de entradas y salidas, fórmulas para la conversión de información)
- Especificación de los requisitos lógicos para la información que será almacenada en base de datos (tipo de información, requerido)

2.2.4.2 Requisitos no funcionales

A este nivel se introducirá un requisito tipo “**Business**” con el Name = “Requisitos No Funcionales”.

Colgando de este nivel se introducirán sólo los Requisitos de Rendimiento ya que estos están asociados a una Release concreta.

2.2.4.2.1 Rendimiento

A este nivel se introducirá un requisito tipo “**Group**” con el Name = “Rendimiento”.

Colgando de este nivel se introducirán cada uno de los requisitos de Rendimiento que se hayan identificado de acuerdo al Anexo 03.07 Procedimiento de Pruebas de Rendimiento. Éstos serán de tipo “**Performance**”, seguirán la nomenclatura “**RR-XX Nombre del Requisito de rendimiento**” y



además de los campos ya comentados de los requisitos funcionales deberán incorporar los siguientes campos:

- En la pestaña Details.
 - Número Máximo de Usuarios
 - Tiempo Medio de Respuesta (sg.) (opcional)
 - Tiempo Máximo de Respuesta (sg.) (opcional)
 - Ritmo transaccional, es decir, el número de veces que se ejecuta el proceso de negocio (script) por hora y por el total de usuarios del RR.
- En la pestaña Description.
 - Información acerca del requisito de rendimiento
 - Tecnología (Web, Webservice, ...)

Requirement Details

ReqID: 217 * Name: RR-01: Rendimiento Pagina de inicio * Requirement Type: Performance

Details

* Priority:	3-High	* Ritmo Transac ...	300
* Target Cycle:	SUBS1-1.0.0;SUBS1	* Target Release:	SUBS1-1.0
* U. Num Máxim...	10	Author:	11836101
Creation Date:	19/01/2009	Creation Time:	11:47:29
Direct Cover Sta...	N/A	Modified:	04/03/2019 15:05:00
Product:		Reviewed:	Not Reviewed
U. Tiempo Máxi...	5	U. Tiempo Medi...	3

Description Comments

Requisito de rendimiento para.....

Tecnología: Web

OK Cancel Help



2.2.4.3 Requisitos de interfaces

A este nivel se introducirá si aplica un requisito tipo “**Business**” con el Name = “Requisitos de Interfaces”.

De este nivel colgarán:

2.2.4.3.1 Interfaces de usuario

Se creará como tipo “Group” y de él colgarán cada uno de los requisitos que apliquen.

2.2.4.3.1.1 RNFI-YY:

Se creará del tipo “No Functional”.

Especificar si aplica los requisitos de interfaz de usuario del producto. Si se establece alguna convención a la hora de representar elementos en pantalla, también se definirá en este apartado

2.2.4.3.2 Interfaces de/con otros sistemas: Ofrecidas

Se creará como tipo “Group” y de él colgarán cada uno de los requisitos que apliquen.

2.2.4.3.2.1 RNFI-XX:

Se creará del tipo “No Functional”.

Para cada producto de software puede especificarse lo siguiente:

- Identificación (nombre y versión) y descripción de la interfaz que se ofrece
- Propósito del interfaz
- Definición del interfaz: contenido y formato

2.2.4.3.3 Interfaces de/con otros sistemas: Consumidas

Se creará como tipo “Group” y de él colgarán cada uno de los requisitos que apliquen.

2.2.4.3.3.1 RNFI-XY:

Se creará del tipo “No Functional”.

Para cada producto de software con el que se interactúa, debe especificarse lo siguiente:



- Identificación (nombre y versión) y descripción del producto software utilizado
- Propósito del interfaz
- Definición del interfaz: contenido y formato


2.2.4.3.4 Interfaces de comunicaciones

Se creará como tipo “Group” y de él colgarán cada uno de los requisitos que apliquen.

2.2.4.3.4.1 RNFI-ZZ:

Describir los requisitos de interfaces de comunicación si hay comunicaciones con otros sistemas o subsistemas y cuáles son los protocolos (ej, web service, restful, etc.) de comunicación -> Maqueta servicios interfaz

2.2.4.1 RNFs

Los requisitos no funcionales, excepto los de Rendimiento, deben incluirse en una carpeta denominada “RNFs” (que será de Tipo “Business” -  RNFs), la cual estará situada bajo la carpeta de la aplicación (ACRO), tal y como se ha visto en una ilustración anterior. En esta carpeta se incluirán todos los requisitos no funcionales que apliquen a los diferentes subsistemas (o aplicación).. Dichos RNFs pueden ser de los indicados en el “Anexo 35 Estándar de Análisis” u otros que Desarrollo considere necesario añadir y a los cuáles se les asignará un número de RNF diferente a los ya asignados según el Anexo 35.

En ALM se ha dejado en cada uno de los proyectos en el módulo “Requirements” una carpeta denominada “_Plantillas-RNFs” que contiene todos los RNFs tal y como están definidos en el Anexo 35 de referencia. De este modo solo es necesario copiar los RNFs que apliquen de dicha carpeta a la carpeta RNFs bajo la carpeta de la aplicación y trazarlos con los subsistemas (o aplicación) a los que aplica como se muestra en el siguiente apartado “Trazabilidad de Requisitos”.



Sort By: Name[Ascending]

Name
▼ Requirements
▶ ACRO
▼ _Plantillas-RNFs
▼ RNFs
RNF-01: Usuario de Control:
RNF-02: Requisitos de interfaz:
RNF-03: Accesibilidad
RNF-04: Escalabilidad
RNF-05: Auditoría
RNF-06: Control de Acceso
RNF-07: Usabilidad
RNF-08: Fiabilidad
RNF-09: Disponibilidad
RNF-10: Multi-idioma
RNF-11: Seguridad
RNF-12: Configuración
RNF-13: Cohesión
RNF-14: Almacén de Datos
RNF-15: Geolocalización
RNF-16: Eficiencia:
RNF-17: Almacenamiento
RNF-18: Portabilidad
RNF-19: Compatibilidad
RNF-20: Interoperabilidad

Se podrá revisar la descripción del RNF ya copiado por si es necesario añadir alguna información adicional relevante para el proyecto.



Ejemplo:

Sort By: Name[Ascending]	Details	Rich Text	Attachments	Linked Defects	Requirement Traceability	Test Coverage	Risk Analysis	Business Models Li
Name								
Requirements								
MIDGT								
Autenticacion								
Backend								
Backoffice								
LectorQR								
Movil								
RNFs								
RNF-03: Accesibilidad								
RNF-04: Escalabilidad								
RNF-05: Auditoría								
RNF-06: Control de Acceso								
RNF-08: Fiabilidad								
RNF-09: Disponibilidad								
RNF-10: Multi-idioma								
RNF-17: Almacenamiento								

■ Name:	RNF-17: Almacenamiento	■ Priority:	4:Very High
■ Requirement Type:	No Functional	Author:	51062803
Creation Date:	20/07/2023	Creation Time:	14:06:46
Direct Cover Status:	Not Covered	Modified:	20/07/2023 14:06:46
Old Type (obsolete):		Product:	
ReqID:	252	Reviewed:	Not Reviewed

Description	Comments
<p>El subsistema MIDGT backoffice hace el uso de un elemento de persistencia para almacenar las noticias en la tabla NoticiasTabla.</p> <p>El repositorio es el DataStorage MidgtDS</p>	

Consideraciones importantes:

- No es necesario asignar prioridad a estos RNFs (en la plantilla están todos con la Prioridad “4-High” por defecto)
- No se deben asociar a Release y Ciclo (estos campos deben quedar vacíos)
- El campo “Description” de la carpeta RNFs se usará para documentar los posibles cambios en los RNFs de una aplicación a lo largo de la vida del proyecto, por ejemplo:

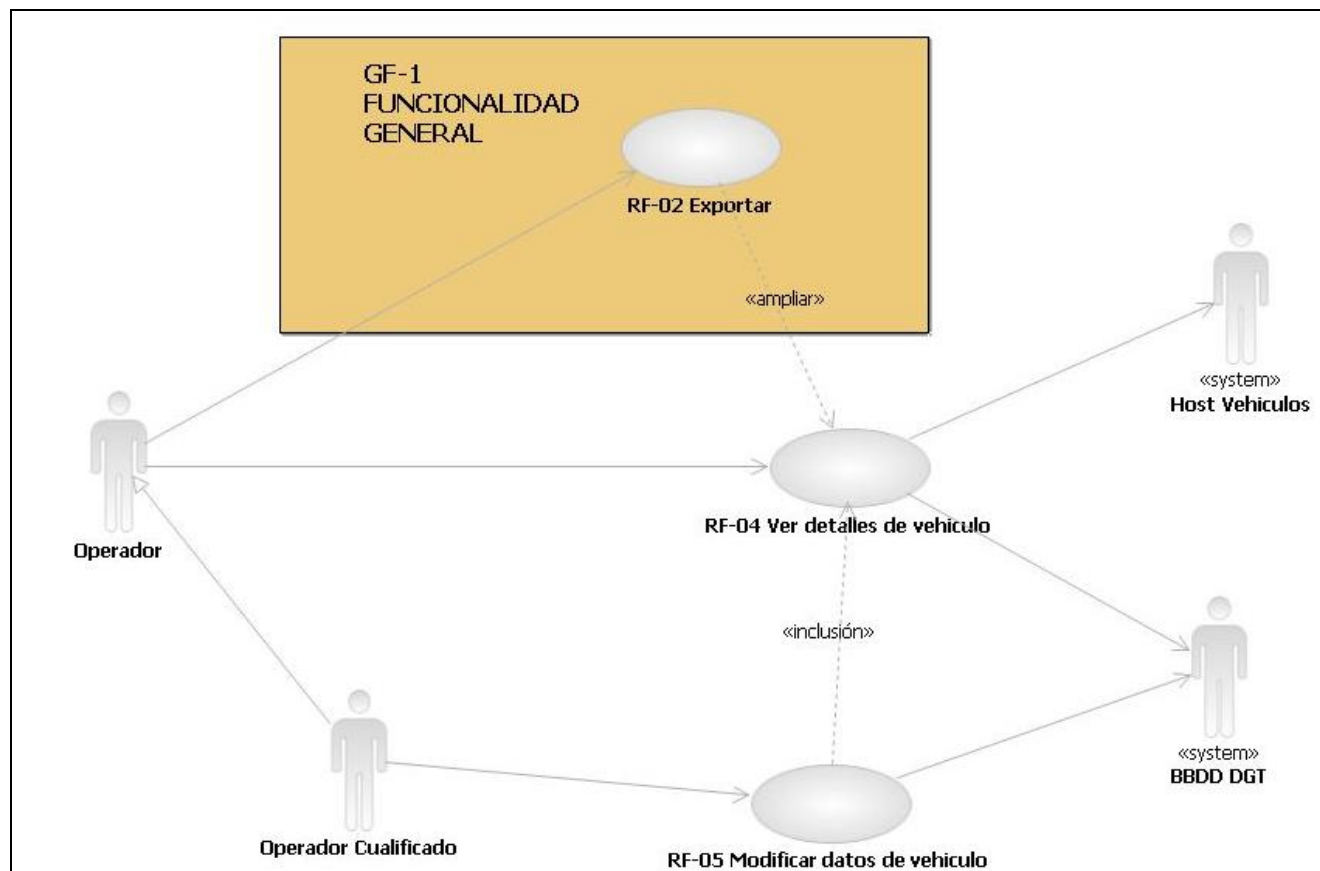
Sort By: Name[Ascending]	Details	Rich Text	Attachments	Linked Defects	Requirement Traceability	Test Coverage	Risk Analysis	Business <>
Name								
Requirements								
ACRO								
RNFs								
RNF-01: Usuario de Control:								
RNF-03: Accesibilidad								
RNF-04: Escalabilidad								
RNF-05: Auditoría								
RNF-06: Control de Acceso								
RNF-08: Fiabilidad								
RNF-09: Disponibilidad								
RNF-10: Multi-idioma								
RNF-17: Almacenamiento								
SUBS1								
SUBS2								

■ Name:	RNFs	■ Requirement Type:	Business
Author:	11836101	Creation Date:	15/09/2023
Creation Time:	13:41:34	Modified:	15/09/2023 20:32:03
Priority:		Product:	
ReqID:	662	Reviewed:	Not Reviewed
Target Cycle:	ReqID:	Target Release:	

Description	Comments
<p>RNFs de la aplicación "ACRO"</p> <p>feb-2023: Inicio del proyecto se definen los RNFs - se adjunta Excel con exportación de RNFs a esta fecha jun-2023: por necesidades del proyecto se decide que a partir de ahora aplique el RNF-09 Disponibilidad a los subsistemas SUBS1 y SUBS2. Se adjunta Excel con la exportación a esta fecha</p>	

Se pueden adjuntar archivos en Attachments si así se considera oportuno:

28



En este caso, cualquier cambio en el Requisito “*RF-04: Ver detalles del vehículo*” afectaría a los requisitos “*RF-02: Exportar*” y “*RF-05: Modificar datos del vehículo*”.

Dicha relación en ALM se establece mediante la opción “**Trace To (requirements affected by)**” que indica **los requisitos que se ven afectados** ante cualquier cambio en el requisito seleccionado.

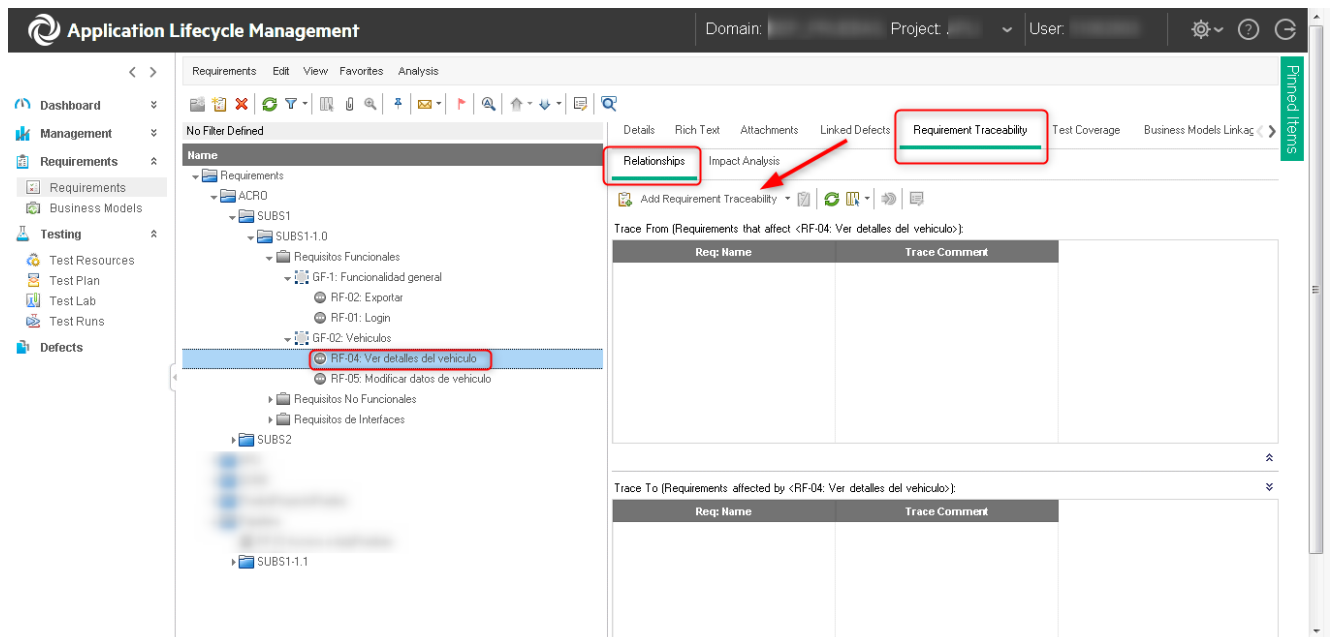
Cuando en ALM se establece esta relación, automáticamente en el requisito afectado se establece la relación “**Trace From (requirements that affect)**” que indica **los requisitos que afectan** a dicho requisito.

Los pasos a realizar desde la vista “**View/Requirement Details**” son:

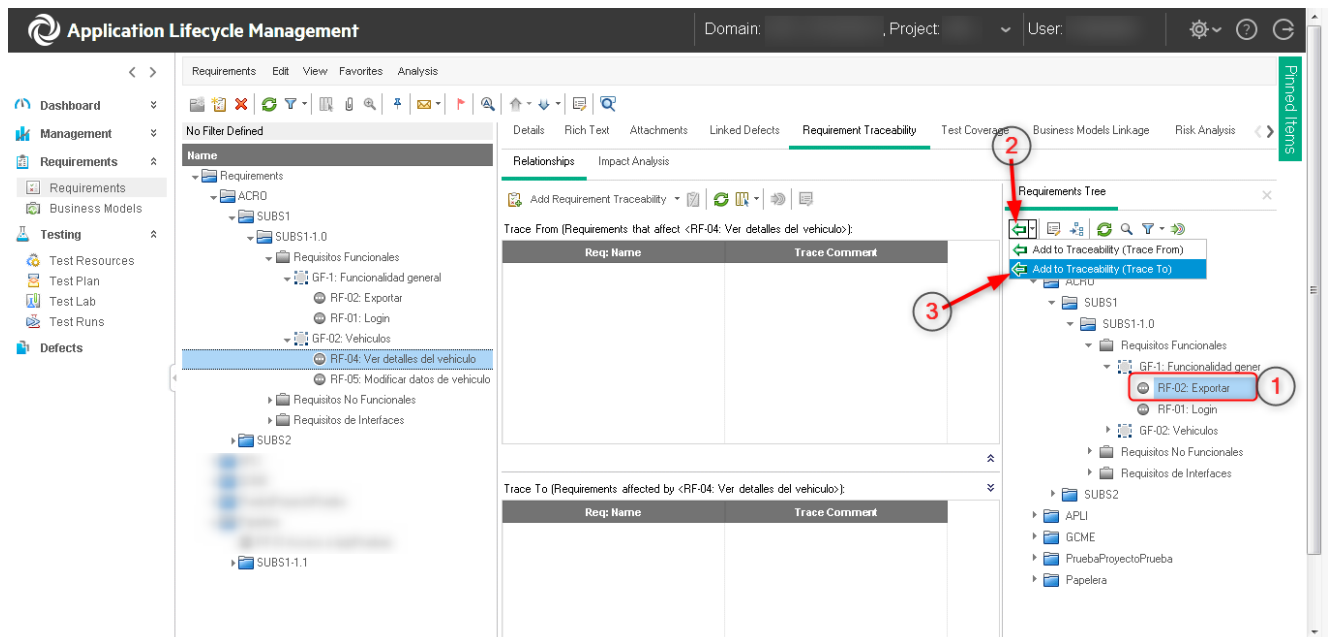
1. Seleccionar el requisito y pulsar en la pestaña “**Requirements Traceability**”



En nuestro ejemplo, seleccionamos el requisito “*RF-04: Ver Detalles del vehículo*”



2. En la parte de “Requirements Tree” seleccionamos el requisito con el que vamos a trazar, en este caso seleccionamos el RF-02: Exportar, pulsamos en la flecha verde y seleccionamos la opción “Add to Traceability (Trace To)”





De este modo se refleja que si cambia el requisito funcional “*RF_04: Ver detalles del Vehículo*” puede verse afectado el requisito “*RF_02: Exportar*”.

No Filter Defined

Name

- Requirements
 - ACRO
 - SUBS1
 - SUBS1-1.0
 - Requisitos Funcionales
 - GF-1: Funcionalidad general
 - RF-02: Exportar
 - RF-01: Login
 - GF-02: Vehiculos
 - RF-04: Ver detalles del vehiculo**
 - RF-05: Modificar datos de vehiculo
 - Requisitos No Funcionales
 - Requisitos de Interfaces
 - SUBS2

Details Rich Text Attachments Linked Defects **Requirement Traceability** Test Coverage

Relationships Impact Analysis

Add Requirement Traceability

Trace From (Requirements that affect <RF-04: Ver detalles del vehiculo>):

Req: Name	Trace Comment
-----------	---------------

Trace To (Requirements affected by <RF-04: Ver detalles del vehiculo>):

Req: Name	Trace Comment
RF-02: Exportar	

Al crear esta relación se crea automáticamente la contraria, es decir, si miramos en el RF-02: Exportar vemos que en el Trace From tiene el RF-04: Ver detalles de vehículo.



The screenshot displays the ALM 16.0.1 interface. On the left, a navigation pane shows a tree structure of requirements: Requirements > ACRO > SUBS1 > SUBS1-1.0 > Requisitos Funcionales > GF-1: Funcionalidad general > RF-02: Exportar. The 'RF-02: Exportar' item is selected and highlighted with a red box. The main area shows the 'Relationships' tab with a table titled 'Trace From (Requirements that affect <RF-02: Exportar>):'. The table has two columns: 'Req: Name' and 'Trace Comment'. A single entry is visible: 'RF-04: Ver detalles del vehiculo'. Below this, there is a section for 'Trace To (Requirements affected by <RF-02: Exportar>):' which is currently empty.

Req: Name	Trace Comment
RF-04: Ver detalles del vehiculo	

Un requisito funcional sólo puede estar trazado con 1 requisito de rendimiento, y un requisito de rendimiento sólo puede estar trazado con 1 requisito funcional (1 a 1). La relación se realizaría de la siguiente forma ya que si cambia el requisito funcional el requisito de rendimiento se ve afectado:

El requisito funcional tendrá una relación de “**Trace To (requirements affected by)**” con el de rendimiento

El requisito de rendimiento tendrá una relación de “**Trace From (requirements that affect)**” con el funcional.



Req: Name	Trace Comment
RF-02: Ver detalles del vehiculo	

2.2.5.1 RNFs


Los RNFs se trazan con los subsistemas a los cuales apliquen, en caso de ser una aplicación sin subsistemas se trazarán con la propia aplicación. La trazabilidad se definirá en el “**Trace To**”
Para ello en ALM para cada RNF que aplique nos vamos a la pestaña “Requirement Traceability” pulsamos en el botón “Add Requirement Traceability”

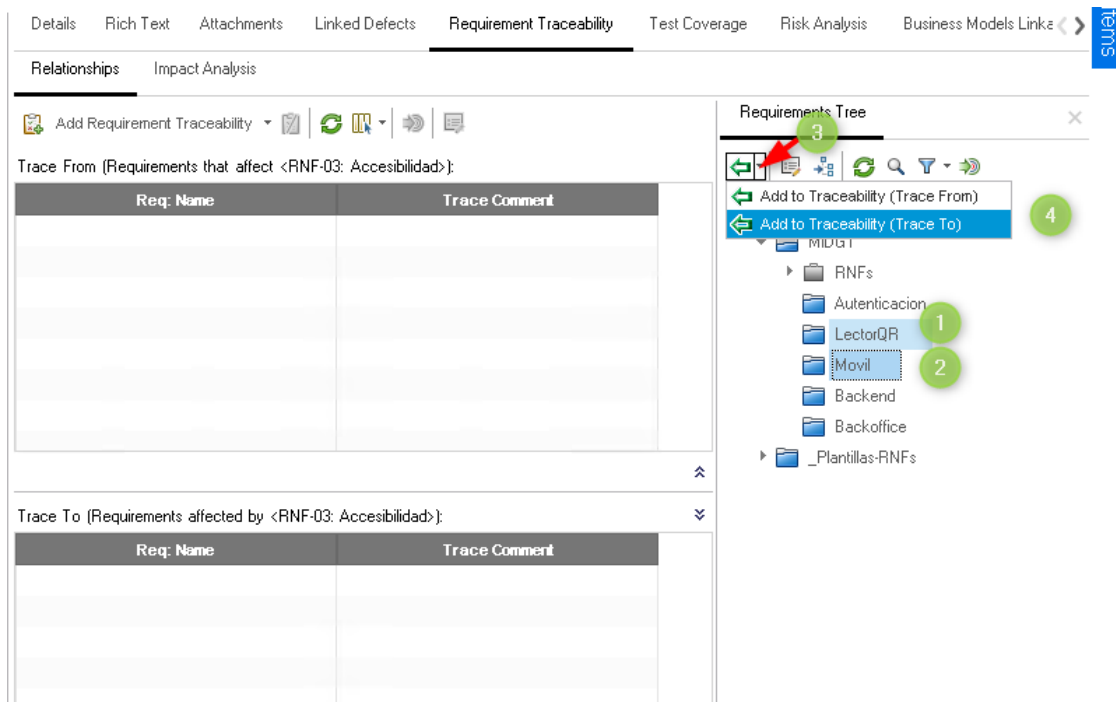
Req: Name	Trace Comment
-----------	---------------

Req: Name	Trace Comment
-----------	---------------

Se nos abre un panel a la derecha “Requirements Tree”, allí desplegamos el árbol de carpetas y seleccionamos las carpetas de los subsistemas a los que aplica dicho RNF (se pueden seleccionar

varias manteniendo pulsada la tecla Ctrl mientras se seleccionan), desplegamos las opciones del botón

de “flecha verde”  y seleccionamos la opción “Add to Traceability (Trace To)”



Relationships Impact Analysis

Add Requirement Traceability

Trace From (Requirements that affect <RNF-03: Accesibilidad>):

Req: Name	Trace Comment
-----------	---------------

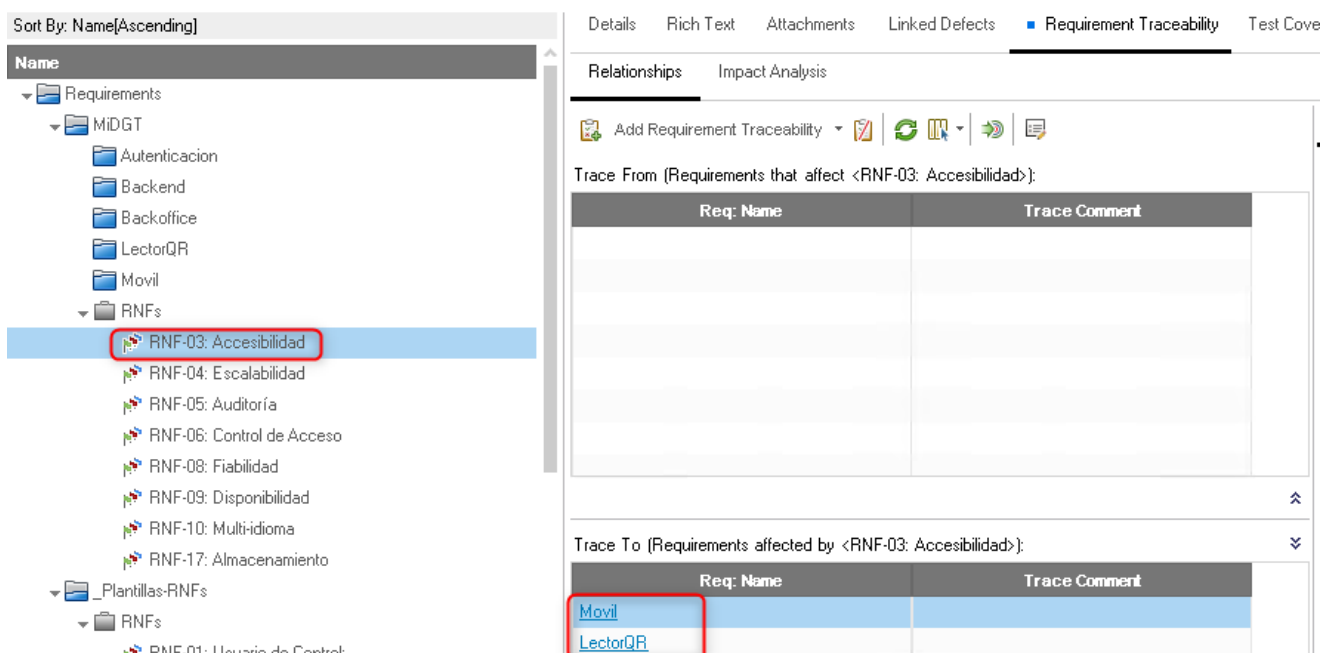
Trace To (Requirements affected by <RNF-03: Accesibilidad>):

Req: Name	Trace Comment
-----------	---------------

Requirements Tree

- MIDGT
 - RNFs
 - Autenticacion
 - LectorQR
 - Movil
 - Backend
 - Backoffice
 - _Plantillas-RNFs

Quedando de este modo:



Sort By: Name[Ascending]

Name

- Requirements
 - MIDGT
 - Autenticacion
 - Backend
 - Backoffice
 - LectorQR
 - Movil
 - RNFs
 - RNF-03: Accesibilidad
 - RNF-04: Escalabilidad
 - RNF-05: Auditoría
 - RNF-06: Control de Acceso
 - RNF-08: Fiabilidad
 - RNF-09: Disponibilidad
 - RNF-10: Multi-idioma
 - RNF-17: Almacenamiento
 - _Plantillas-RNFs
 - RNFs
 - RNF-01: Usuario de Control

Relationships Impact Analysis

Add Requirement Traceability

Trace From (Requirements that affect <RNF-03: Accesibilidad>):

Req: Name	Trace Comment
-----------	---------------

Trace To (Requirements affected by <RNF-03: Accesibilidad>):

Req: Name	Trace Comment
Movil	
LectorQR	



En este ejemplo el “RNF-03: Accesibilidad” aplicaría a los subsistemas “Movil” y “LectorQR” de la aplicación MiDGT.

Se procedería de igual modo con el resto de RNFs.

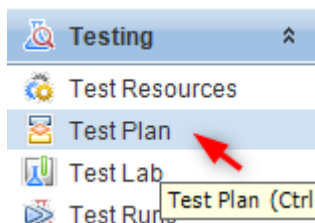
Ejemplo para aplicación sin subsistemas:

Req. Name	Trace Comment
BTVE	

Trace To (Requirements affected by <RNF-01: Usuario de Control>):

2.3 Módulo Test Plan

2.3.1 Objetivo



En el módulo Test Plan de ALM es donde se va a realizar la documentación de los planes de pruebas funcionales (manuales y automatizadas) y de rendimiento de los aplicativos.

2.3.2 Estructura

Una vez introducidos los requisitos en el módulo “**Requirements**” se deben crear su correspondencia en el módulo “**Test Plan**”.



Desde ALM se puede generar automáticamente la estructura de carpetas para que cubran todos los requisitos especificados en el módulo “**Requirements**”.

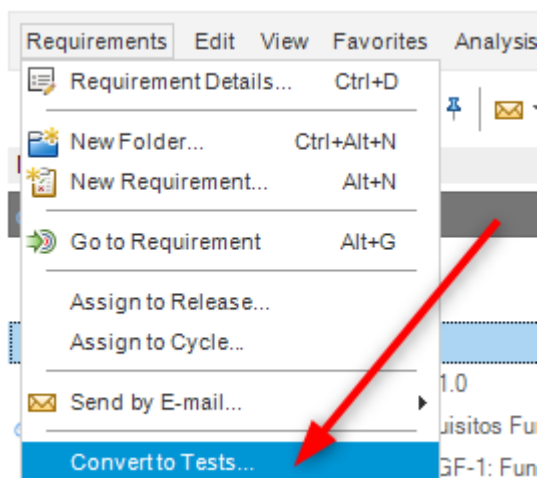
Los pasos a realizar son:

1. Ir al módulo “**Requirements**”
2. Seleccionar la carpeta con el nombre de la aplicación.

Por ejemplo, la carpeta “**SUBS1**”

La imagen muestra la interfaz de usuario de Application Lifecycle Management (ALM). En el panel izquierdo, se encuentra el menú de navegación con opciones como Dashboard, Management, Requirements, Business Models, Testing y Defects. El panel central muestra la estructura de carpetas de requisitos, con 'SUBS1' seleccionada. El panel derecho muestra los detalles de la carpeta seleccionada, incluyendo campos como Name, Author, Creation Time, Priority, ReqID, Target Cycle, Requirement Type, Creation Date, Modified, Product, Reviewed, y Target Release. En la parte inferior, hay una sección para la descripción y comentarios.

3. Ir al menú y ejecutar “Requirements\Convert to Tests...”



4. En el “Step1 de 3”, seleccionar la opción “**Convert all requirements to subjects.**” Pulsar en el botón “**Next**”.



Step 1 of 3: Select Automatic Conversion Method

MICRO FOCUS

Application Lifecycle Management
Convert To Test

Automatic Conversion Method

☐ Convert lowest child requirements to design steps.

☐ Convert lowest child requirements to tests.

☒ Convert all requirements to subjects.

☐ Generate single test.

☐ Add Test to Test Set

☐ Keep current Requirements filter

Req1
Req2
Req3

Req1
Req2
Req3

Req1
Req2
Req3

Req1
Req2
Req3

Cancel < Back Next > Help

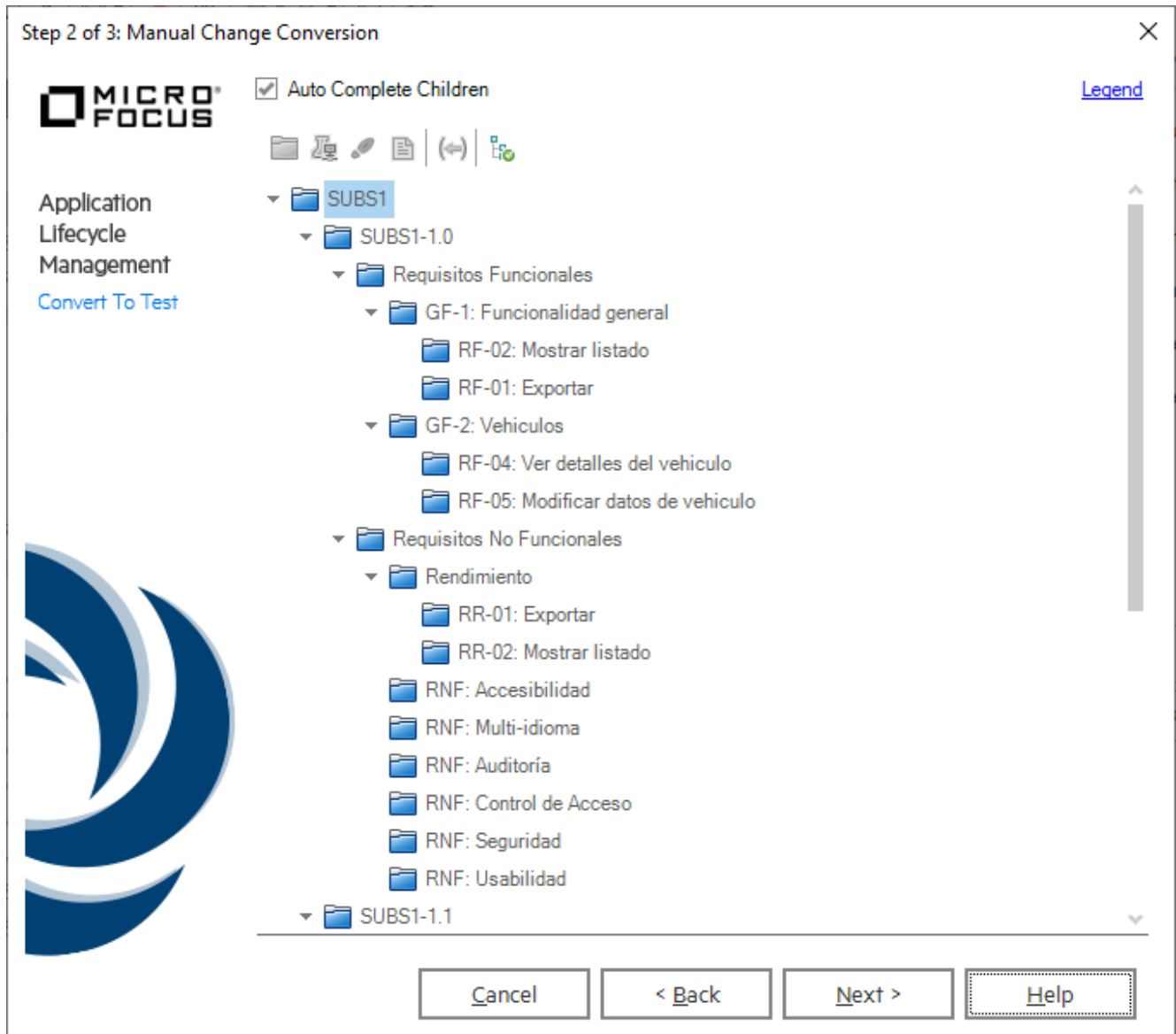
Aparecerá la barra de progreso y el cuadro del “Step2 de 3” con el árbol de requisitos:

Processed

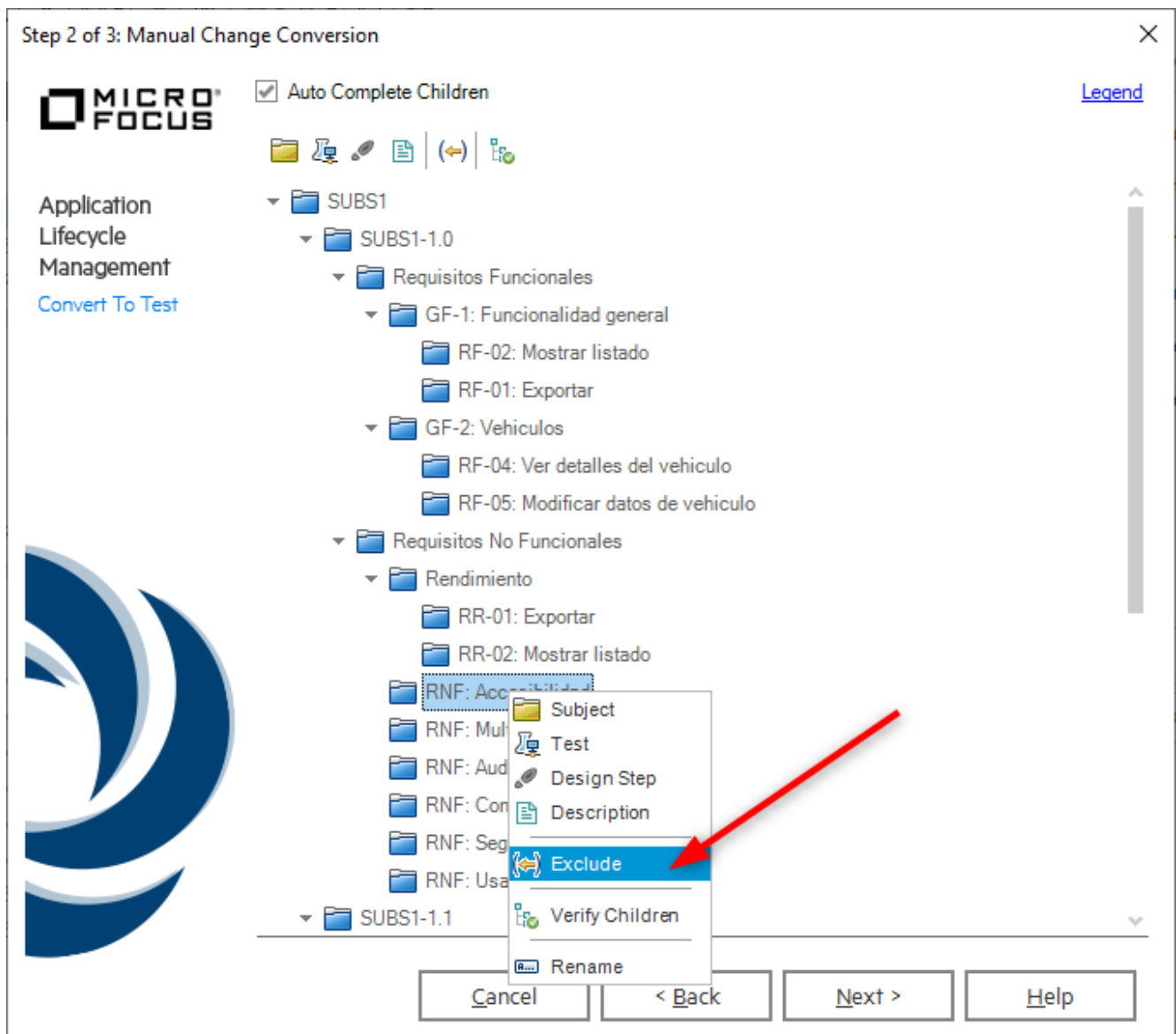
Processed RR-02 Buscar Vehículos

93%

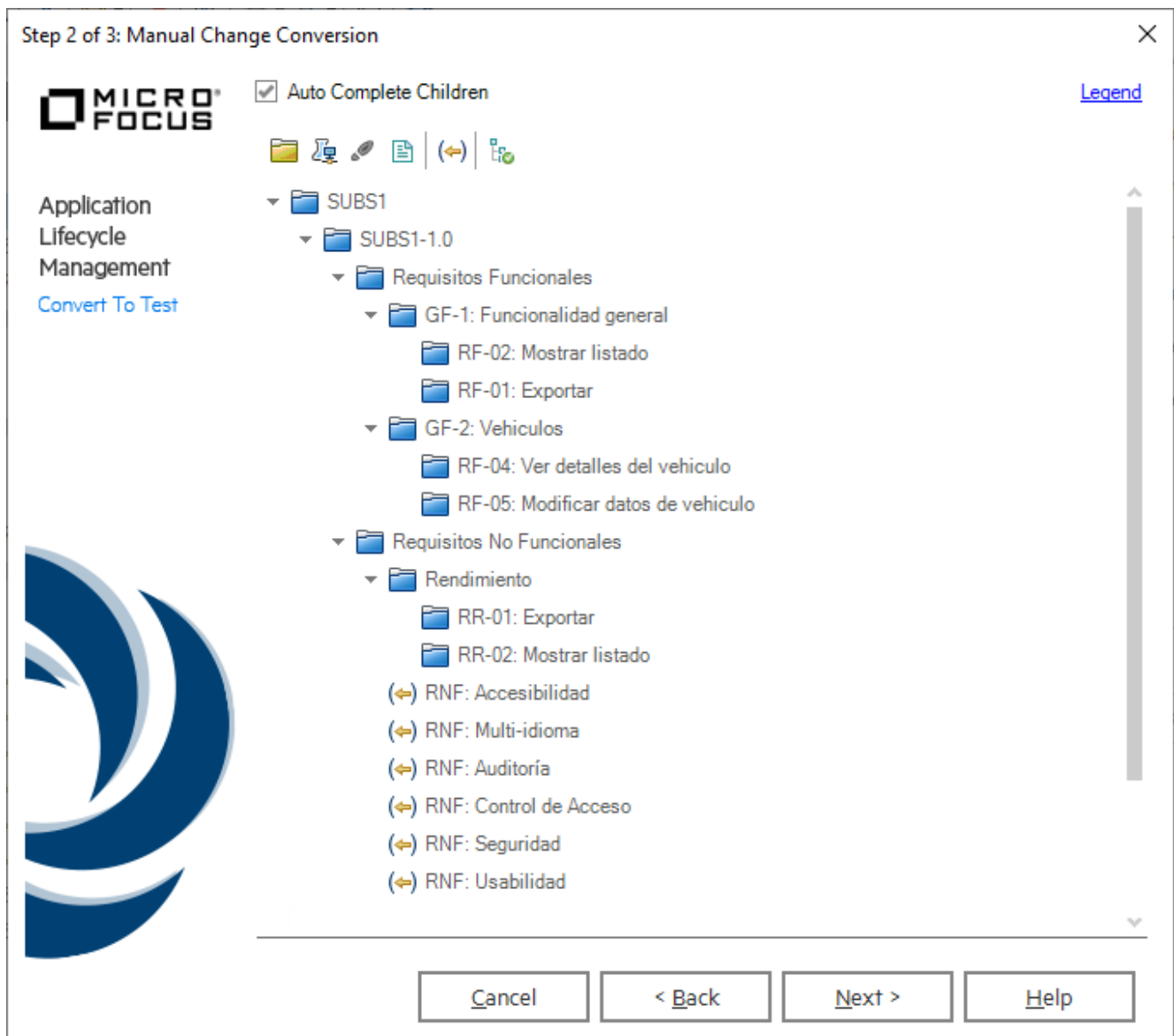
Stop



5. Excluimos los Requisitos No funcionales que no sean de Rendimiento así como los Requisitos de Interfaces. Para ello nos situamos sobre cada uno de ellos, pulsamos botón derecho y seleccionamos **“Exclude”**




Quedando así:




6. Pulsar en el botón “**N**ext” y seleccionar la ruta destino pulsando en el botón




Step 3 of 3: Select Destination Path

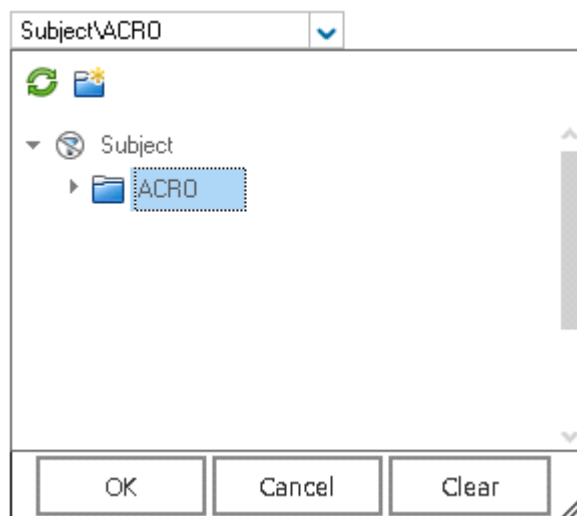
 Destination Subject Path _____

Subject: 

Application
Lifecycle
Management
[Convert To Test](#)



7. Seleccionar la carpeta “**Subject\ACRO**” porque en este ejemplo estamos convirtiendo toda la carpeta **SUBS1** y pulsar en el botón “**OK**”



8. Pulsar en el botón “**Finish**”




Step 3 of 3: Select Destination Path

MICRO FOCUS

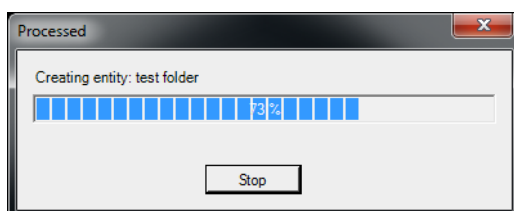
Destination Subject Path _____

Subject: ▼

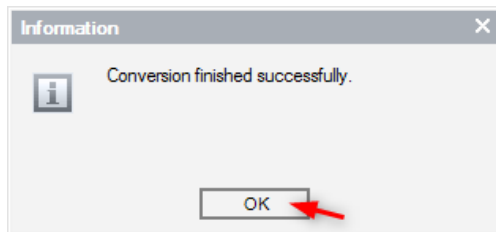
Application
Lifecycle
Management
[Convert To Test](#)



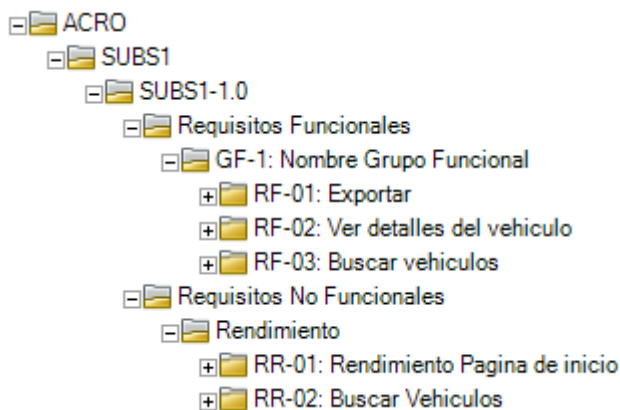
Aparecerá la barra de progreso



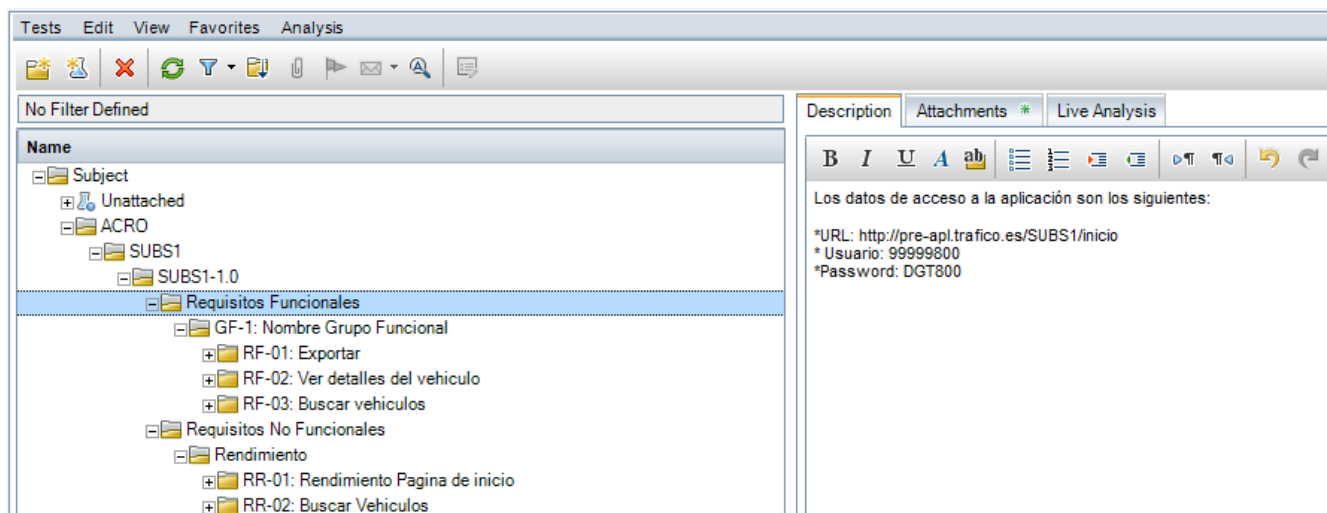
9. Si todo ha ido correctamente, mostrará el siguiente mensaje y se pulsará en el botón “**OK**”



En el módulo “**Test Plan**” se habrá generado la estructura de carpetas correspondiente.



En la pestaña “**Description**” de la carpeta “**Requisitos Funcionales**” debe documentarse la url (o urls) de acceso a la aplicación así como un usuario y password para acceder a la misma.



2.3.3 Ubicación ficheros de datos de entrada para auditorías

Para la auditoría de los planes de prueba la ubicación de los ficheros de datos de entrada deberá estar en la siguiente ubicación:



2.3.3.1 Fichero XLS de pruebas funcionales

El fichero con los datos de prueba para las pruebas funcionales deberá estar en la pestaña “Attachments” de la carpeta “Requisitos funcionales”. El fichero XLS debe cumplir la estructura de pestañas y la nomenclatura especificada en el documento “Anexo 03.01 Guía de datos de entrada de pruebas”.

Ejemplo:

Name	Size	Modified
certificado p12	4 KB	05/05/2015 11:38:00
datos certificado.txt	1 KB	16/03/2020 9:19:02
DENT_ACRO_SUBS1_PRE.xls	26 KB	29/11/2018 9:47:44
FicherosXML.zip	427 KB	21/12/2018 13:35:04
Request.xml	4 KB	12/07/2013 9:38:14
WSDL.xml	34 KB	14/11/2018 12:24:52

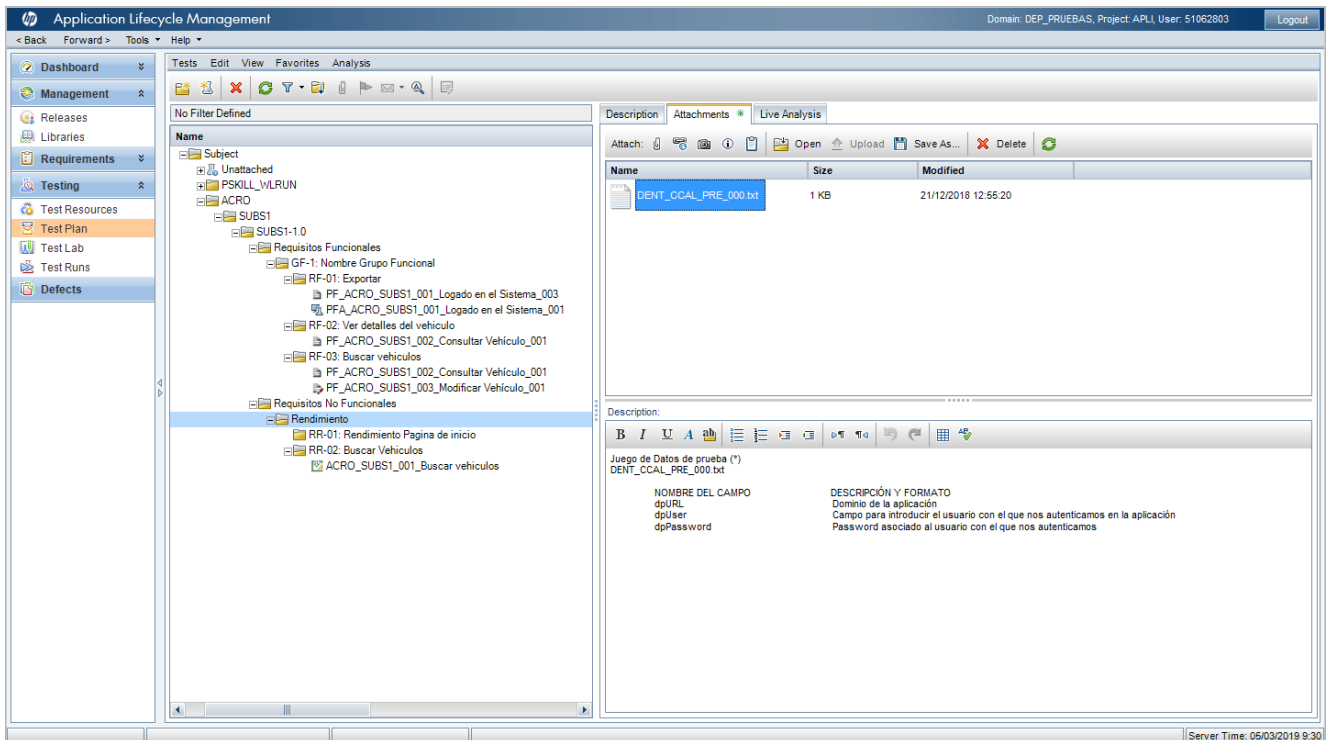
2.3.3.2 Ficheros de pruebas de rendimiento

El fichero con los datos de acceso a la aplicación deberá estar en la pestaña “Attachments” de la carpeta “Rendimiento” y se ha de incluir en la pestaña “Description” del propio fichero la siguiente información:

- Declaración del nombre del fichero
- Declaración de los campos del fichero



Ejemplo del fichero con los datos de acceso en la pestaña “**Attachments**” de la carpeta “**Rendimiento**”:

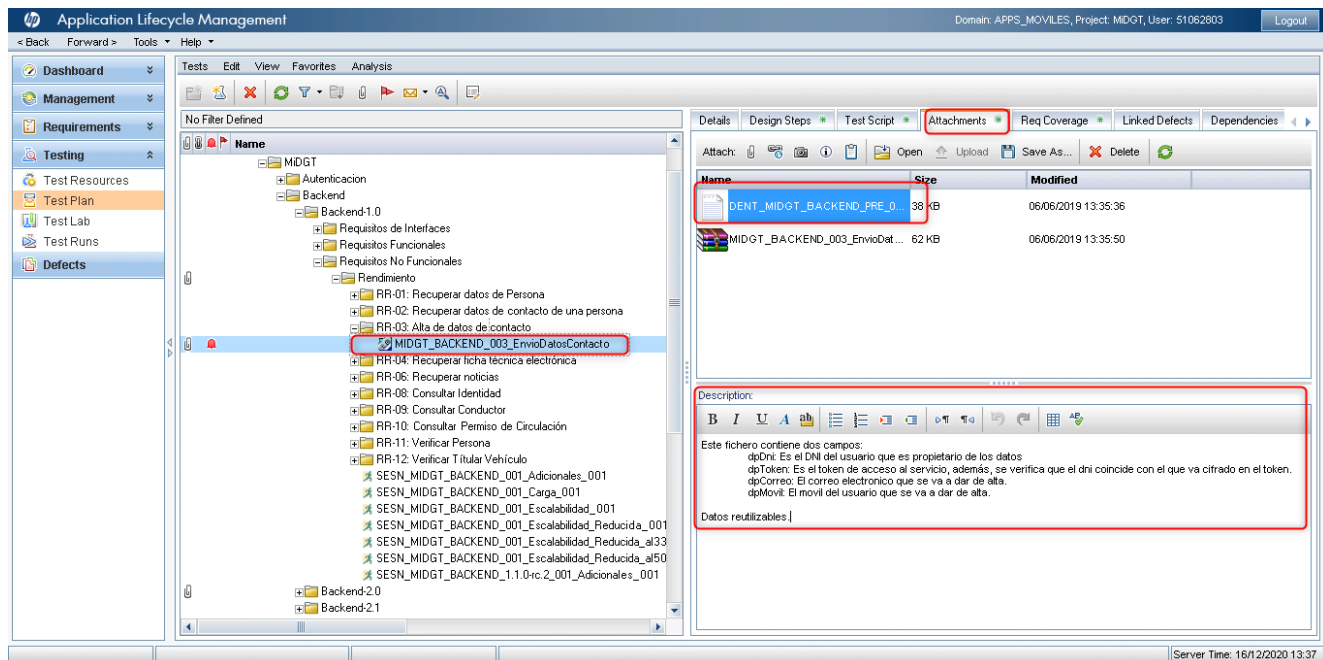


Por otro lado los ficheros con los datos de las pruebas de rendimiento deberán estar en la pestaña “**Attachments**” de los casos de prueba. La declaración de los ficheros y sus campos debe realizarse en el campo “**Description**” de la pestaña “**Attachments**” del fichero tal y como se indica en las siguientes imágenes y cumplir la nomenclatura especificada en el documento “*Anexo 03.01 Guía de datos de entrada de pruebas*”.

En el campo “**Description**” de la pestaña “**Attachments**” del fichero deberá aparecer:

- Declaración del nombre del fichero
- Declaración de los campos del fichero
- Descripción de las dependencias de cada uno de los campos con respecto a los demás parámetros.
- Si los datos son reutilizables o consumibles.

Ejemplo de declaración de fichero de datos de un *script*:



2.3.3.3 Otros ficheros

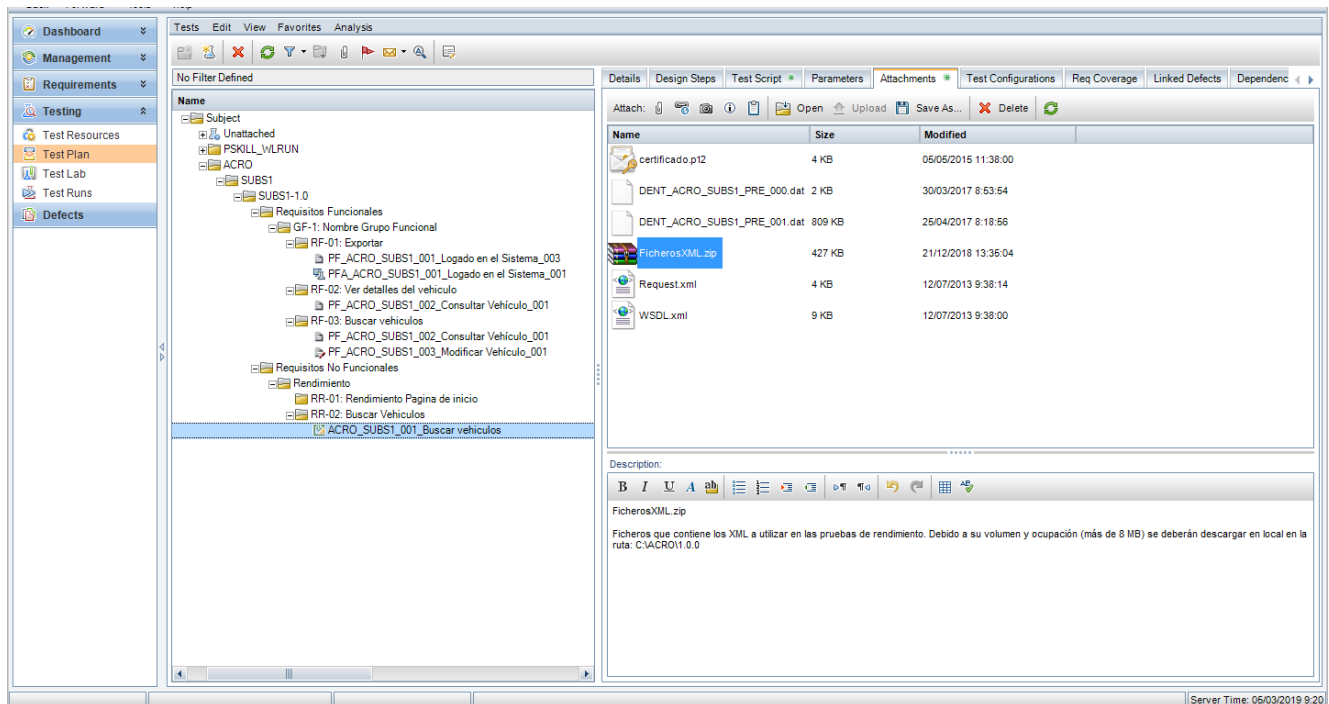
Además de los ficheros con los datos a utilizar en las pruebas en algunas ocasiones necesitamos de otro tipo de ficheros como:

- Certificados y sus datos asociados (Sujeto (Asunto) del certificado, password si es necesaria...). En formato .cer preferiblemente
- Proyecto SoapUI en servicios web
- XML de entrada y salida y llamada SOAP en servicios web
- Fichero YAML o JSON con la definición del API REST (Swagger – Open API 2.0)
- Proyecto Postman para servicios REST
- Documentos explicativos para realizar algún tipo de procedimiento previo (en formato Word o PDF)
- .ZIP (con documentación variada)
- Otros ficheros que puedan ser interesantes



2.3.3.3.1 Pruebas de rendimiento

En estos casos los ficheros y su declaración se deberán anexar al nivel del caso de prueba de rendimiento:



2.3.3.3.2 Pruebas funcionales

Cuando sean necesarios otros ficheros para las **pruebas funcionales** y sean comunes a todos los casos, se colocarán en “Attachments” en la carpeta “**Requisitos funcionales**” y se indicará para ellos su nombre y descripción:

En el caso de que el fichero sea aplicable solo a un caso de prueba, entonces se anexará a nivel de caso de prueba en la pestaña “Attachments” y se incluirá una descripción del contenido del fichero:

50



2.3.4 Casos de prueba

2.3.4.1 Normativa de ubicación de casos de prueba

Al empezar a crear los casos de prueba que cubren los diferentes requisitos, se debe hacer especial hincapié en su ubicación.

Cada requisito debe estar cubierto al menos por un caso de prueba. Y dentro de cada carpeta de requisito funcional o de rendimiento deben existir los casos de prueba que cubren dicho requisito.

Si un caso de prueba cubre varios requisitos funcionales, el caso de prueba deberá estar dentro de la carpeta de uno de los requisitos funcionales que cubre y en la pestaña descripción de la carpeta del otro requisito funcional indicar por qué otros casos de prueba está cubierto.

Ejemplo:

La imagen muestra la interfaz de usuario de 'Application Lifecycle Management'. En el panel izquierdo, se encuentra el menú de navegación con opciones como Dashboard, Management, Requirements, Testing, Test Resources, Test Plan, Test Lab, Test Runs y Defects. El panel central muestra una estructura de árbol de requisitos. Se expande el nodo 'SUBS1', que contiene sub-nodos como 'SUBS1-1.0', 'SUBS1-1.1', 'ATMV', 'PSKILL_WLRUN' y 'SUSR'. El nodo 'SUBS1-1.0' está seleccionado y muestra una lista de requisitos funcionales: 'Regresión Funcional Automatizada', 'Requisitos Funcionales', 'GF-1: Nombre Grupo Funcional', 'RF-01: Exportar', 'RF-02: Ver detalles del vehiculo' y 'RF-03: Buscar vehiculos'. El nodo 'RF-03: Buscar vehiculos' está seleccionado y muestra una lista de casos de prueba: 'PF_ACRO_SUBS1_002_Consultar V...', 'PF_ACRO_SUBS1_003_Modificar Ve...', 'PF Manuales-Componentes Comunes' y 'Requisitos No Funcionales'. El panel derecho muestra la descripción de los casos de prueba, indicando que el requisito funcional está cubierto por los casos de prueba: 'PF_SUBS1_003_Consultar Vehiculo_001' y 'PF_SUBS1_004_Modificar Vehiculo_001'.

2.3.4.2 Llamada a casos de prueba (Call to test)


En algunas situaciones debido a la dependencia funcional que existe entre casos de prueba se puede dar la situación que un caso de prueba realice la llamada a otro caso de prueba. En estos casos

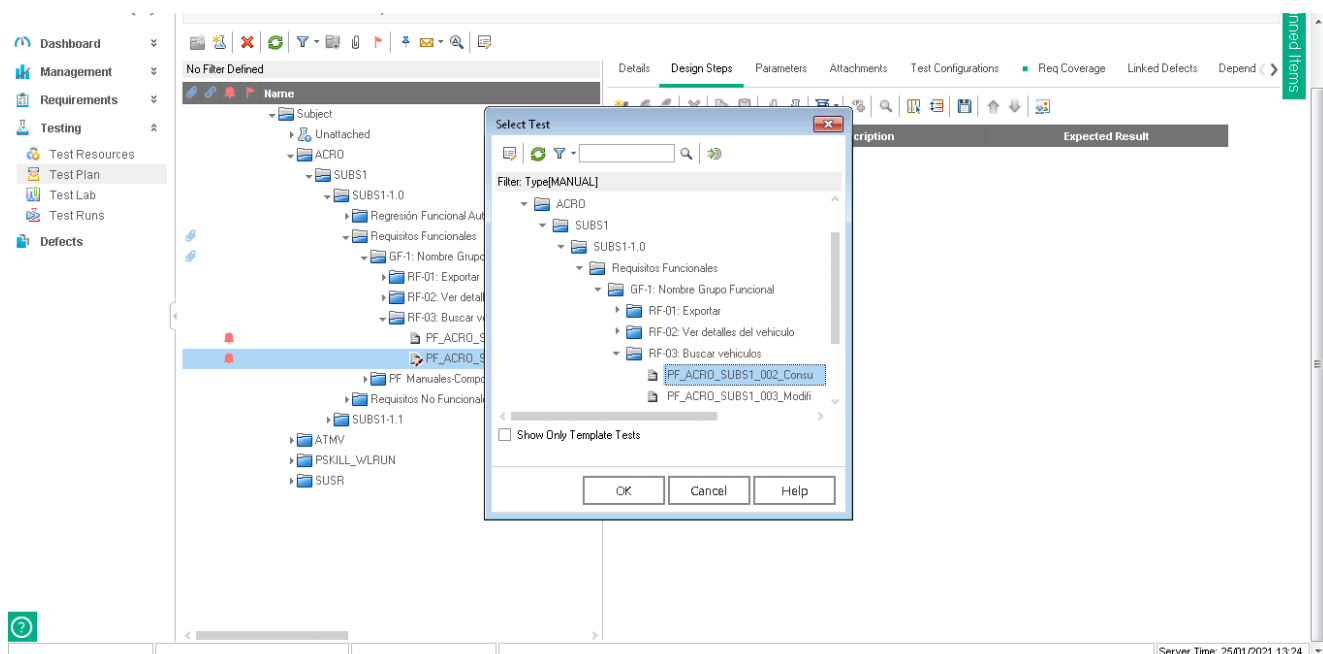
existe siempre una dependencia funcional y no puede existir una dependencia de datos entre ellos.

También puede darse el caso de que existan fragmentos de navegación manual que son comunes a múltiples tests y que por eficiencia sea conveniente definirlos una única vez y hacer llamadas desde todos los tests que lo requieran. A este tipo los llamaremos “Componentes Comunes Manuales” y se explican más adelante en el apartado “2.3.4.3.4 Componentes comunes de casos de prueba funcionales manuales”.

El procedimiento a seguir en ambas situaciones es la de realizar una llamada a un caso de prueba (“*Call to test*”) dentro de la pestaña “Design Steps” del otro caso de prueba.

Los pasos a realizar son:

1. Seleccionar el Step y pulsar en el botón “**Call to Test (Ctrl.+L)**” 
2. En la pestaña “**Select a Test**”, navegar por la estructura y seleccionar el caso de prueba al que queremos llamar.





En la pestaña “**Design Steps**” aparecerá la llamada al caso de prueba y nos quedará como en la siguiente imagen:

Step Name	Description	Expected Result
Step 1	Call <PF_ACRO_SUBS1_002_Consultar Ve...	
Step 2		

La pestaña de datos de entrada que utilice el caso de prueba que hace la llamada (el que tiene el Call to test) contendrá **todos los datos necesarios** para ejecutar ese caso de prueba, incluidos los que se usen en el componente común o el test invocado. Para más información sobre el fichero de datos de prueba ver el apartado “2.3.3.1 Fichero XLS de pruebas funcionales”.

En la “Description” del caso de prueba que hace la llamada se informarán el nombre y descripción de **todos** los campos de esta pestaña. Para más información sobre el campo “Description” de un caso de prueba, ver el apartado “2.3.4.4.2 Detalles de un caso de prueba funcional”.

2.3.4.3 Tipos de casos de prueba

2.3.4.3.1 Casos de prueba de rendimiento



Los casos de prueba de rendimiento no se crean desde ALM, se generan cuando se crea un nuevo script a través de la herramienta LoadRunner-Vugen. De manera que cada caso de prueba de rendimiento de ALM quedará asociado a un script LoadRunner-Vugen.

2.3.4.3.1.1 Reglas de Nomenclatura

2.3.4.3.1.1.1 Sintaxis

<Aplicación_Subistema>_<999>_<Descripción>

Donde:

Aplicación Es el acrónimo que asigna la DGT a cada Aplicación

Subsistema Es el acrónimo del subsistema

999 Es un índice secuencial dentro de la misma Aplicación.
Comienza por **001** y se reinicia por aplicación y por cada subsistema.
Ver ejemplos más abajo.

Descripción Es la descripción de la Transacción de negocio

2.3.4.3.1.2 Ejemplo

Por aplicación:

RR-01 Consulta Vehiculos

ACRO_001_Consulta_Vehiculos

RR-02 Consultas DNI

ACRO_002_Consultas_DNI

RR-03 Consultas Sanciones

ACRO_003_Consultas_Sanciones

Por subsistema:

SUB1: Subsistema 1 dentro de ACRO

RR-01 Consulta Vehiculos

ACRO_SUBS1_001_Consulta_Vehiculos

RR-02 Consultas DOI



ACRO_SUBS1_002_Consulta_DOI

SUB2: Subsistema 2 dentro de ACRO

RR-01 Consulta Sanciones

ACRO_SUBS2_001_Consulta_Sanciones

RR-02 Consulta Pago Sancion

ACRO_SUBS2_002_Consulta_Pago_Sancion

2.3.4.3.1.3 Pasos a realizar

Ver apartado de creación de scripts Vugen en ALM del anexo “Anexo 03.08 Guia de programacion en LoadRunner-Vugen 12.53”.

2.3.4.3.2 Casos de prueba funcionales (Manuales)

2.3.4.3.2.1 Reglas de Nomenclatura

2.3.4.3.2.1.1 Sintaxis

PF_<Aplicación>_<Subsistema>_<999>_<Descripción>_<000>

Donde:

Aplicación Es el acrónimo que asigna la DGT a cada Aplicación

Subsistema Es el acrónimo que asigna la DGT a los subsistemas de la

999 En un índice que se corresponde con una agrupación funcional dentro de cada requisito.
Comienza por **001** y es único por aplicación-subsistema. Ver ejemplos.

Descripción Es el nombre descriptivo del test y con relación al nombre de la agrupación funcional.

000 Es el nº del caso de prueba dentro de cada agrupación funcional que se corresponde con el índice (**999**).
Comienza por **001** y es único por cada agrupación funcional.

2.3.4.3.2.2 Ejemplo



Por aplicación:

GF-01 Consultas de Vehículo:

RF-01 Consulta Matricula

PF_ACRO_001_Consulta_Matricula_001

PF_ACRO_001_Consulta_Matricula_002

PF_ACRO_001_Consulta_MatriculaError_003

PF_ACRO_001_Consulta_MatriculaError_004

RF-02 Consulta DOI

PF_ACRO_002_Consulta_Doi_001

PF_ACRO_002_Consulta_Doi_002

PF_ACRO_002_Consulta_DoiError_003

Por subsistema:

SUBS1: Subsistema 1 dentro de ACRO

GF-01 Consultas de Vehículo

RF-01 Consulta Matricula

PF_ACRO_SUBS1_001_Consulta_Vehiculos_001

PF_ACRO_SUBS1_001_Consulta_Vehiculos_002

RF-02 Consulta DOI

PF_ACRO_SUBS1_002_Consulta_Doi_001

PF_ACRO_SUBS1_002_Consulta_Doi_002

SUBS2: Subsistema 2 dentro de ACRO

GF-01 Consultas de Vehiculos

RF-01 Consulta Sanciones

PF_ACRO_SUBS2_001_Consulta_Sanciones_001

PF_ACRO_SUBS2_001_Consulta_Sanciones_002


RF-02 Consulta Pago Sancion

PF_ACRO_SUBS2_002_Consulta_Pago_Sancion_001



PF_ACRO_SUBS2_002_Consulta_Pago_Sancion _002

2.3.4.3.2.3 Pasos a realizar

1. Ubicarse en la zona del “**TestPlan**” y seleccionar la carpeta–ruta donde vamos a introducir nuestro
2. Pulsar el botón de “**New Test**” .

En la pantalla que aparece introducir los siguientes datos:

Test Type: “**Manual**”

Test Name: Nombre del caso según la nomenclatura dada

Priority: Prioridad del caso de prueba

Technology: Tipo de tecnología sobre la que se realizará la prueba (Web, Webservice, Batch, Otros...)



New Test

✖ AB AB AB Use Default Values Set Default Values Template: <None>

* Test Name: PF_ACRO_SUBS1_001_Logado en el sistema_001 * Type: MANUAL

Details Attachments

* Priority: 3-High * Technology: 1. Web

Creation Date: Designer: 51062803

Status: Design WSDL:

Description Comments

B I U A ab [iconos de formato]

OK Close Help

NOTA: Se repite el proceso por cada prueba **“Manual”** a introducir.



2.3.4.3.3 Casos de prueba funcionales (Automáticas)

(Nota: la Oficina de Pruebas es la encargada de la automatización de casos de pruebas funcionales, se indica la nomenclatura solo a título informativo)

2.3.4.3.3.1 Reglas de Nomenclatura

2.3.4.3.3.1.1 Sintaxis

PFA_<Aplicación>_<Subsistema>_<999>_<Descripción>_<000>

Donde:

Aplicación Es el acrónimo que asigna la DGT a cada Aplicación

Subsistema Es el acrónimo que asigna la DGT a los subsistemas de la

999 En un índice que se corresponde con una agrupación funcional dentro de cada requisito.
Comienza por **001** y es único por aplicación-subsistema. Ver ejemplos.

Descripción Es el nombre descriptivo del test y con relación al nombre de la agrupación funcional.

000 Es el nº del caso de prueba dentro de cada agrupación funcional que se corresponde con el índice (**999**).
Comienza por **001** y es único por cada agrupación funcional.

2.3.4.3.3.2 Ejemplo

Por aplicación:

GF-01 Consultas de Vehículo:

RF-01 Consulta Matricula

PFA_ACRO_001_Consulta_Matricula_001

PFA_ACRO_001_Consulta_Matricula_002

PFA_ACRO_001_Consulta_MatriculaError_003

PFA_ACRO_001_Consulta_MatriculaError_004

RF-02 Consulta DOI

PFA_ACRO_002_Consulta_DOI_001



PFA_ACRO_002_Consulta_Doi_002

PFA_ACRO_002_Consulta_DoiError_003

Por subsistema:

SUBS1: Subsistema 1 dentro de ACRO

GF-01 Consultas de Vehículo:

RF-01 Consulta Matricula

PFA_ACRO_SUBS1_001_Consulta_Vehiculos_001

PFA_ACRO_SUBS1_001_Consulta_Vehiculos_002

RF-02 Consulta DOI

PFA_ACRO_SUBS1_002_Consulta_Doi_001

PFA_ACRO_SUBS1_002_Consulta_Doi_002

SUBS2: Subsistema 2 dentro de ACRO

GF-01 Consultas de Vehículo:

RF-01 Consulta Sanciones

PFA_ACRO_SUBS2_001_Consulta_Sanciones_001


PFA_ACRO_SUBS2_001_Consulta_Sanciones_002

RF-02 Consulta Pago Sanción

PFA_ACRO_SUBS2_002_Consulta_Pago_Sancion_001

PFA_ACRO_SUBS2_002_Consulta_Pago_Sancion_002

2.3.4.3.3 Pasos a realizar

1. Ubicarse en la zona del “**TestPlan**” y seleccionar la carpeta–ruta donde vamos a introducir nuestro
2. Pulsar el botón de “**New Test**” .

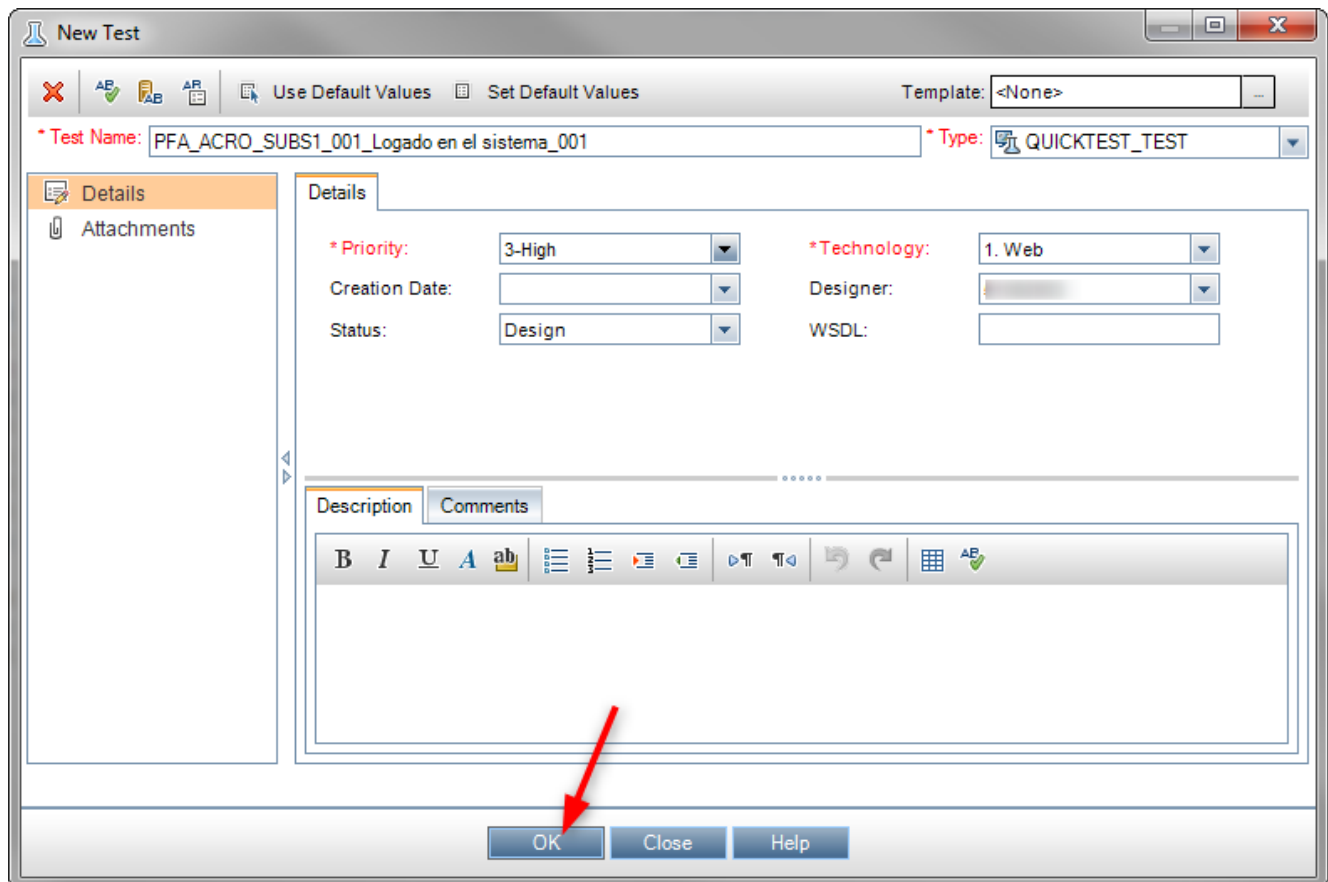
En la pantalla que aparece introducir los siguientes datos:

Test Type: “**QUICKTEST_TEST**”

Test Name: Nombre del script según la nomenclatura dada

Priority: Prioridad del caso de prueba

Technology: Tipo de tecnología sobre la que se realizará la prueba

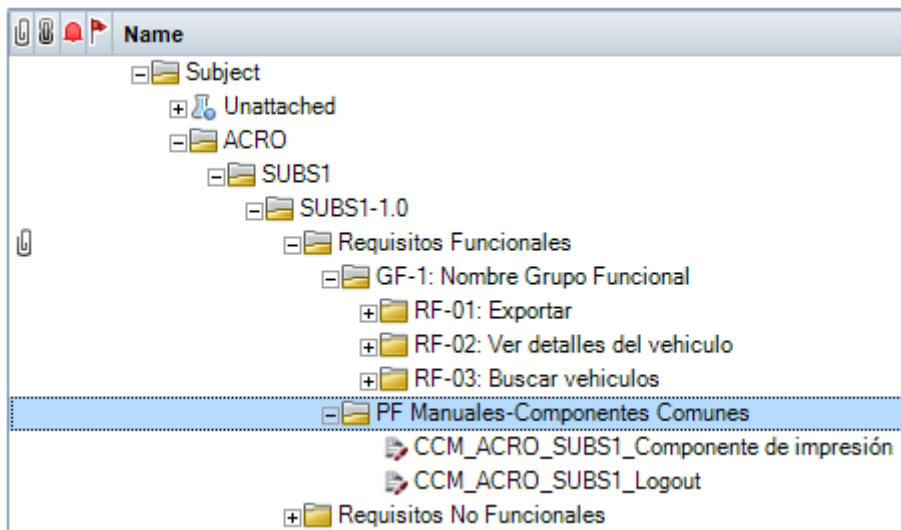


NOTA: Se repite el proceso por cada prueba “Automática” a introducir.

2.3.4.3.4 Componentes comunes de casos de prueba funcionales manuales

2.3.4.3.4.1 Reglas de Nomenclatura

Este tipo de tests se ubicarán en una carpeta llamada “PF Manuales-Componentes Comunes” (PF=Pruebas Funcionales), que estará dentro de la carpeta “Requisitos Funcionales” de la aplicación/Subsistema y fuera de cualquier grupo funcional, porque son fragmentos que pueden ser reutilizados en cualquier test de cualquier Requisito Funcional, por ello estos componentes no llevarán asociados ningún requisito.



2.3.4.3.4.1.1 Sintaxis

La sintaxis de este tipo de componentes es:

CCM_<Aplicación>_<Subsistema>_<Descripción>

Donde:

Aplicación Es el acrónimo que asigna la DGT a cada Aplicación

Subsistema Es el acrónimo que asigna la DGT a los subsistemas de la

Descripción Es la descripción del componente común.

2.3.4.3.4.2 Ejemplo

Por aplicación:

PF Manuales-Componentes Comunes:

CCM_ACRO_Componente de impresión

CCM_ACRO_Logout

CCM_ACRO_Descarga PDF



Por subsistema:

SUBS1: Subsistema 1 dentro de ACRO

CCM_ACRO_SUBS1_Componente de impresión

CCM_ACRO_SUBS1_Logout


CCM_ACRO_SUBS1_Descarga PDF

SUBS2: Subsistema 2 dentro de ACRO

CCM_ACRO_SUBS2_Login

CCM_ACRO_SUBS2_Logout

2.3.4.3.4.3 Pasos a realizar

1. Ubicarse en la zona del “**TestPlan**” y seleccionar la carpeta–ruta donde vamos a introducir nuestro
2. Pulsar el botón de “**New Test**” .

En la pantalla que aparece introducir los siguientes datos:

Test Type: “**Manual**”

Test Name: Nombre del componente según la nomenclatura dada

Priority: Prioridad del componente común

Technology: Tipo de tecnología sobre la que se realizará la prueba



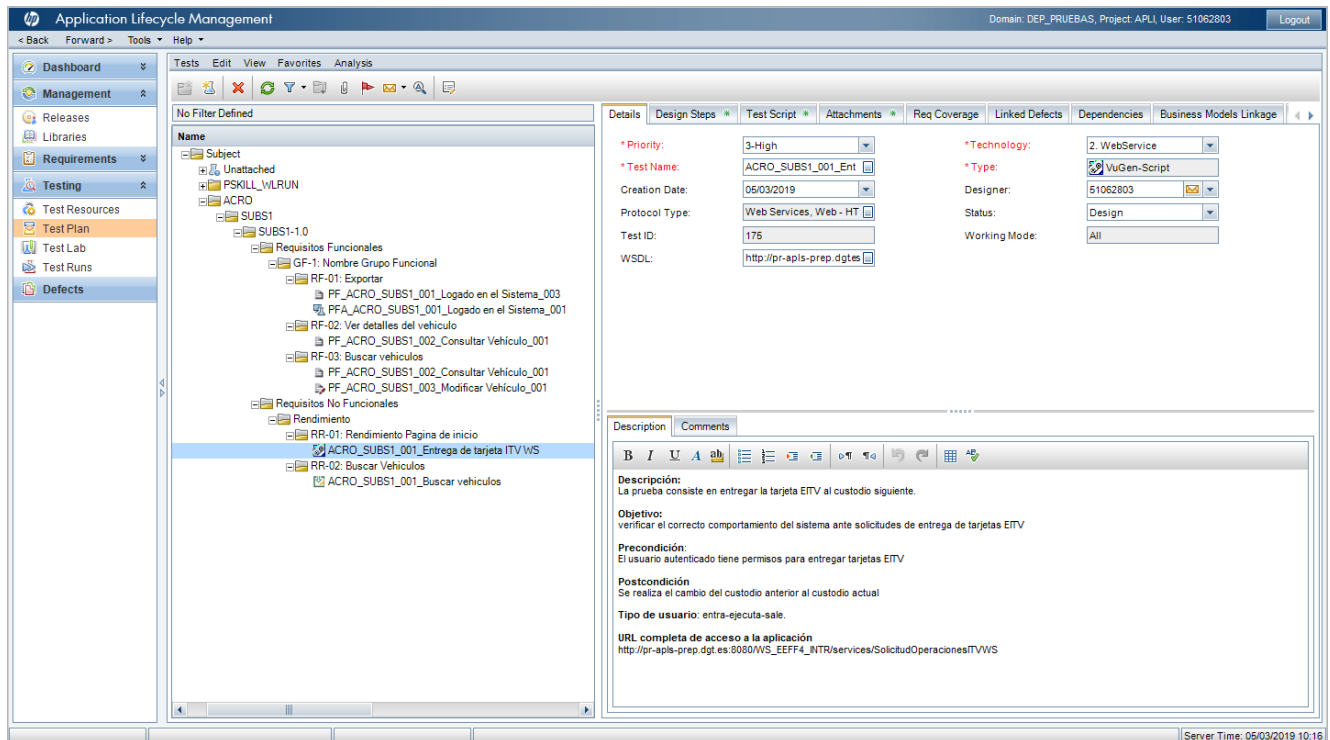
NOTA: Se repite el proceso por cada prueba “*Componente Común Manual*” a introducir.

2.3.4.4 Documentación de casos de prueba

Por cada caso de prueba y componente común manual creado en ALM se deberá incorporar su documentación, para ello habrá que rellenar las pestañas “**Details**” y “**Design Steps**”.

2.3.4.4.1 Detalles de un caso de prueba de rendimiento

En la pestaña “**Details**” de un caso de prueba de rendimiento se deben rellenar los campos “**Priority**”, “**Technology**”, “**WSDL**” si es de tipo **WebService** y la pestaña “**Description**”.



En el campo “**Description**” deberá aparecer toda la información referente al caso de prueba, como por ejemplo:

- Descripción
- Objetivo
- Condiciones necesarias para la ejecución
- URL completa de acceso a la aplicación
- Tipo de usuario, si es persistente en sesión o no, es decir, indicar si el usuario accede en vuser_init o en Action.
- Juego de datos de entrada a utilizar

2.3.4.4.2 Detalles de un caso de prueba funcional

En la pestaña “**Details**” de un caso de prueba funcional tanto manual como automático, y de un componente común manual, se deben rellenar los campos “**Priority**”, “**Technology**” y la pestaña “**Description**”.



The screenshot displays the ALM software interface. On the left is a navigation pane with categories like Dashboard, Management, Requirements, Testing, and Defects. The main area is divided into a tree view on the left and a details panel on the right. The tree view shows a hierarchy of test cases, with 'PF_ACRO_SUBS1_002_Consultar V' selected. The details panel on the right shows fields for Priority (4-Very High), Test Name (PF_ACRO_SUBS1_002_Con), Creation Date (05/03/2019), Status (Design), and Test ID (174). Below these fields is a 'Description' tab, which is active, showing a rich text editor with the following content:

Descripción (*)
La prueba consiste en la búsqueda de los datos de un vehículo que se corresponde con una matrícula introducida por el usuario.

Objetivo(*)
El objetivo es verificar que se muestran correctamente los datos del vehículo buscado.

Condiciones(*)
Solo se podrán consultar los vehículos que tienen el recibo de la ITV pagados y no tienen sanciones.
Tener la aplicación web ACRO-SUBS1 desplegada en el servidor que se vaya a probar y que esté configurada correctamente.
Los componentes a los que acceda la aplicación deben estar funcionando correctamente.
El usuario que realice la prueba debe estar dado de alta correctamente en LDAP.
El usuario debe tener permisos de acceso a la aplicación, bien como operador, o como operador cualificado.

Tipo de Datos(*)
Los datos son reutilizables, se puede consultar varias veces

Juego de datos prueba(*)
DEVT_SUBS1_PRE.xls: Pestaña-->DP002.

NOMBRE DEL CAMPO	DESCRIPCIÓN Y FORMATO
dpCodigo1	Descripción del campo1. Está compuesto por 8 dígitos.
dpCodigo2	Descripción del campo2
dpCodigo3	Descripción del campo3. Depende del valor introducido para dpCodigo2.

En la pestaña “**Description**” deberá aparecer toda la información referente al caso de prueba, como por ejemplo:

- Descripción
- Objetivo
- Condiciones necesarias para la ejecución
- Descripción de las pestañas que contienen los datos utilizados en el caso de prueba: nombre, descripción y formato de los campos, si hay dependencias de cada uno de los campos con respecto a los demás campos, y si los datos son reutilizables o se consumen al ejecutar el caso de prueba. **Nota:** para los Componentes Comunes Manuales (CCM's), es posible que no haya una pestaña en el XLS pero sí lista de campos. Para evitar confusiones se puede indicar en su “Juego de datos de pruebas” este comentario: “*No aplica pestaña independiente para este componente común porque no se puede probar de forma independiente. Los valores de los campos de este CCM dependerán de la pestaña de cada caso de prueba que lo llame. Estos son los campos que se necesitan para este CCM:*” y a continuación, poner la lista de declaración de campos.
- Otros ficheros de entrada a utilizar en la prueba además del xls



- Acciones necesarias a ejecutar (Antes/Después). Para cuando haya que ejecutar alguna sentencia SQL antes (o bien, después) de ejecutar los pasos del test. En esta situación, el fichero mencionado que contenga dicha sentencia SQL se adjuntará en la pestaña “Attachments” del test.


2.3.4.4.3 Diseño de pasos

Pulsar en la pestaña “**Design Steps**” e introducir toda la secuencia de acciones y resultados esperado necesarios para la ejecución del caso de prueba.

Las acciones a realizar para introducir los pasos son:

Pulsar el botón  y rellenar la información relativa a:

- Step Name: nombre del paso.
- Description: descripción detallada de las acciones a realizar sobre el sistema.
- Punto de chequeo (TRAN) (solo para casos de prueba de rendimiento).
- Llamada a sistema externo (solo para casos de prueba de rendimiento): Detalle del sistema externo al cual se llama, por ejemplo, aplicación y proceso (webservices, ejb, etc)
- Expected Result: resultado esperado tras la ejecución del paso, especificando la cadena de texto, imagen,... a chequear, que verifique que el proceso de negocio se ha ejecutado correctamente.

Details	Design Steps *	Test Script *	Parameters	Attachments	Test Configurations	Req Coverage	Linked Defects	Dependenc
								
	Step Name	Description	Expected Result					
	Step 1	Logar en la aplicación, paa ello ejecutar el script "PF_ACRO_SUBS1_001_Logado en el Sistema_001"	Ejecución correcta					
	Step 2	Pulsar en el enlace de la izquierda "Consulta/Modificacion de Vehículos"	Verificar que aparece el texto "Buscador"					
	Step 3	Introducir en el campo "NIF/NIE Propietario", el DNI facilitado como parametro de entrada. Pulsar en el botón "Buscar"	Verificar que aparece el enlace "Ver Detalles" y el DNI del conductor consultado					
	Step 4	Pulsar en el enlace "Ver detalles"	Comprobar que aparece la matricula introducida como parametro de entrada					




En los requisitos de rendimiento donde los tiempos medios y máximos de respuesta excedan de los definidos en la guía de desarrollo es necesario indicar qué paso del caso de prueba está asociado a dicho requisito. El resto de pasos se medirán teniendo en cuenta los tiempos medios y máximos de respuesta estándar definidos en la guía.

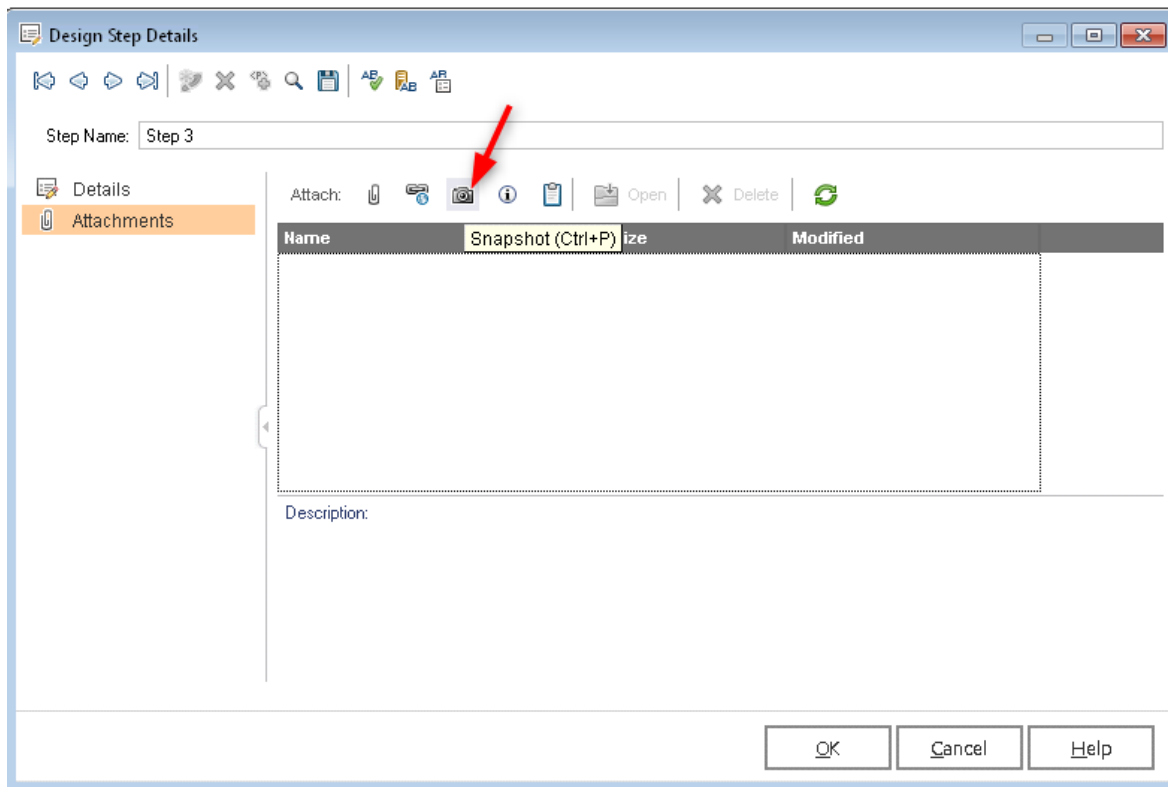
Ejemplo:

“Punto de Chequeo “TRAN_CCAL_002_Pagina_Consulta_Conductores” --> --> Req Relacionado RR-02:Consulta Conductor”

Step Name	Description	Expected Result
Step 1	- Tener la aplicacin web de CCAL desplegada en el servidor que se vaya a ↓	
Step 2	Abrir el navegador e introducir la url de acceso a la aplicación Ir_think_time (8);	Chequear que aparece el contenido "Introduzca sus datos de acceso"
Step 3	Introducir en el campo "Usuario" y "Password" los datos de acceso a la aplicación. Dichos datos están en el fichero adjunto DENT_CCAL_PRE_000.txt Ir_think_time (15); Punto de Chequeo "TRAI_CCAL_000_Logado"	Chequear que aparece el contenido "Está en: Menú de Aplicaciones > CURSO CALIDAD " y aparece el texto "Bienvenido"
Step 4	El usuario selecciona el enlace "Consulta Conductores" . Ir_think_time (7); Punto de Chequeo "TRAN_CCAL_002_Pagina_Consulta_Conductores" --> --> Req Relacionado RR-02:Consulta Conductor	El sistema muestra la pantalla para la selección del criterio de búsqueda de vehículos. Chequear que aparece el contenido "NIF/NIE/CIF conductor "

2.3.4.4.3.1 Anexar capturas a los pasos

Para anexar capturas a los pasos pulsaremos sobre el botón  y se anexarán las capturas de pantallas necesarias para poder ejecutar la prueba.



2.3.4.4.4 Attachments

En la pestaña “Attachments” de cada caso de prueba se anexarán los archivos específicos necesarios para la ejecución del caso de prueba. Por ejemplo:

- Caso de prueba funcional: el XML (o json) de petición y de respuesta en el caso de un WebService
- Caso de prueba de rendimiento: el archivo de datos específico para el script, el XML (o json) de petición y de respuesta en el caso de un WebService

2.3.4.4.5 Req. Coverage

En la pestaña “**Req. Coverage**” se asociará el caso de prueba con el o los requisitos que esté relacionado.

Para ello habrá que seleccionar el caso de prueba y en la pestaña “Req. Coverage” , pulsaremos en

“Select Req”  y seleccionamos el requisito al que se quiere asociar.



Application Lifecycle Management

Domain: DEP_PRUEBAS, Project: APLI User: 51062803

Tests Edit View Favorites Analysis

No Filter Defined

Entity Name Req: Description

Requirement Tree

Requirements

ACRO

SUBS1

SUBS1-1.0

Regresión Funcional Automatizada

Requisitos Funcionales

GF-1: Nombre Grupo Funcional

RF-01: Exportar

PF_ACRO_SUBS1_001_Logado en el Sistema_001

RF-02: Ver detalles del vehiculo

PF_ACRO_SUBS1_002_Consultar Vehiculo_001

RF-03: Buscar vehiculos

PF_ACRO_SUBS1_002_Consultar Vehiculo_001

PF_ACRO_SUBS1_003_Modificar Vehiculo_001

PF Manuales-Componentes Comunes

Requisitos No Funcionales

SUBS1-1.1

ATMV

PSKILL_WLRUN

SUSR

Tras haberlo enlazado:

Application Lifecycle Management

Domain: DEP_PRUEBAS, Project: APLI User: 51062803

Tests Edit View Favorites Analysis

No Filter Defined

Entity Name Req: Description

Requirement Tree

Requirements

ACRO

SUBS1

SUBS1-1.0

Regresión Funcional Automatizada

Requisitos Funcionales

GF-1: Nombre Grupo Funcional

RF-01: Exportar

PF_ACRO_SUBS1_001_Logado en el Sistema_001

RF-02: Ver detalles del vehiculo

PF_ACRO_SUBS1_002_Consultar Vehiculo_001

RF-03: Buscar vehiculos

PF_ACRO_SUBS1_002_Consultar Vehiculo_001

PF_ACRO_SUBS1_003_Modificar Vehiculo_001

PF Manuales-Componentes Comunes

Requisitos No Funcionales

SUBS1-1.1

ATMV

PSKILL_WLRUN

SUSR

NOTA: Repetir el proceso por cada requisito a enlazar con el caso de prueba.

NOTA: los elementos de la carpeta de Componentes Comunes no llevan requerimientos asociados.



2.3.4.5 Abrir Scripts Vugen desde ALM

Para grabar un script de rendimiento ya existente se deberá abrir desde ALM. Los pasos a realizar son:

1. Seleccionar el script de rendimiento a abrir.

The screenshot shows the ALM interface with the 'Test Script' tab selected. The left sidebar displays a tree view of test cases, with 'ACRO_SUBS1_001_Entrega de tarjeta ITV wS' selected. The main panel shows the details of this test script, including its name, creation date, protocol type, and test ID. The 'Description' field contains the following text:

Descripción:
La prueba consiste en entregar la tarjeta ITV al custodio siguiente.

Objetivo:
verificar el correcto comportamiento del sistema ante solicitudes de entrega de tarjetas ITV

Precondición:
El usuario autenticado tiene permisos para entregar tarjetas ITV

Postcondición:
Se realiza el cambio del custodio anterior al custodio actual

Tipo de usuario: entra-ejecuta-sale.

URL completa de acceso a la aplicación
http://pr-apls-prep.dgt.es:8080/WSS_EEFF4_INTR/services/SolicitudOperacionesITVWS

2. Pulsar en el botón “Launch” de la pestaña “Test Script”

The screenshot shows the ALM interface with the 'Test Script' tab selected. The left sidebar displays a tree view of test cases, with 'ACRO_SUBS1_001_Entrega de tarjeta ITV wS' selected. The main panel shows the details of this test script, including its name, creation date, protocol type, and test ID. The 'Description' field contains the following text:

Descripción:
La prueba consiste en entregar la tarjeta ITV al custodio siguiente.

Objetivo:
verificar el correcto comportamiento del sistema ante solicitudes de entrega de tarjetas ITV

Precondición:
El usuario autenticado tiene permisos para entregar tarjetas ITV

Postcondición:
Se realiza el cambio del custodio anterior al custodio actual

Tipo de usuario: entra-ejecuta-sale.

URL completa de acceso a la aplicación
http://pr-apls-prep.dgt.es:8080/WSS_EEFF4_INTR/services/SolicitudOperacionesITVWS

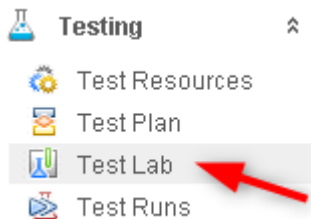
The 'Launch' button is highlighted with a red arrow.

[illegible]



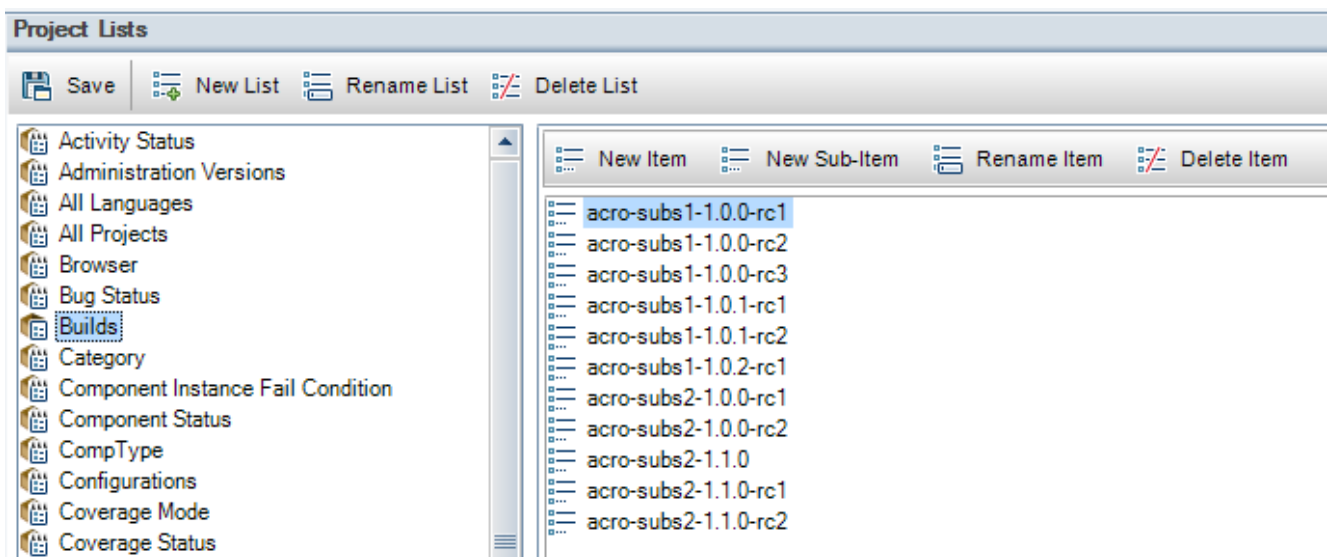
2.4 Módulo Test Lab

2.4.1 Objetivo

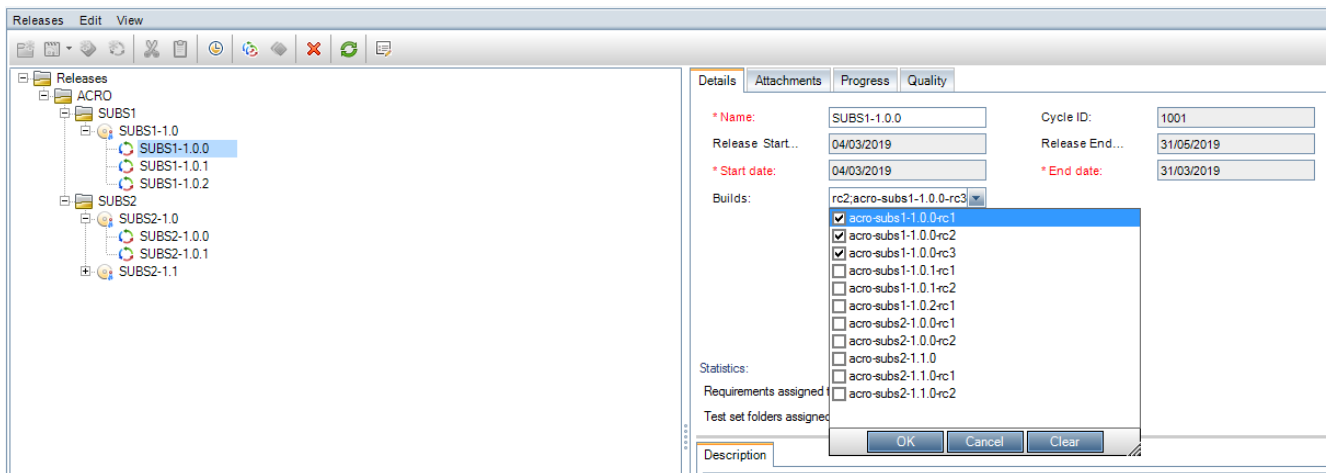


En el módulo Test Lab se crearán las baterías de pruebas y se registrarán las ejecuciones por cada Release, Ciclo y Build.

Para dar de alta las diferentes Builds asociadas a una aplicación sobre las que ejecutar las pruebas se solicitará a la Oficina de Pruebas que se dé de alta dicha Build en ALM dentro del proyecto.



Las Builds se asociarán al ciclo oportuno y se deberá asignar una Build a cada Test Set.



2.4.2 Creación de una batería de pruebas

Accedemos al módulo de “**Test Lab**” y estructuramos la batería de pruebas de la siguiente manera.

- + Nombre Aplicación
 - + [Sistema] (en caso de que tenga subsistemas)
 - + Release
 - + Ciclo
 - + Build
 - + Pruebas Funcionales
 - + Test Set Funcionales Manuales
 - + Test Set Funcionales Automatizadas
 - + Pruebas Rendimiento
 - + Test Set Carga
 - + Test Set Sobrecarga
 - + Test Set Escalabilidad
 - + Pruebas Seguridad
 - + Test Set Seguridad



HP Application Lifecycle Management

Domain: DEP_PRUEBAS, Project: APLI, User: 51062803

Test Sets Edit View Tests Favorites Analysis

No Filter Defined

Root

Unattached

ACRO

SUBS1

SUBS1-1.0.0

acro-sub1-1.0.0-rc1

Pruebas Funcionales

TestSet Funcionales Manuales

Details Attachments Live Analysis

Assigned to cycle: SUBS1-1.0.0

Cycle start date: 04/03/2019

Cycle end date: 31/03/2019

Remaining days in cycle: 26

Statistics:

Description

Server Time: 05/03/2019 10:44

A cada Test Set deberemos asignarle la Build correspondiente dentro de las disponibles en el ciclo.

HP Application Lifecycle Management

Domain: DEP_PRUEBAS, Project: APLI, User: 51062803

Test Sets Edit View Tests Favorites Analysis

No Filter Defined

Root

Unattached

ACRO

SUBS1

SUBS1-1.0.0

acro-sub1-1.0.0-rc1

Pruebas Funcionales

TestSet Funcionales Manuales

Select Tests Run Run Test Set

Details Execution Grid Execution Flow Automation Attachments Linked Defects History

* Build to test: acro-sub1-1.0.0-rc1

* Name: TestSet Funcionales Ma

Baseline:

Modified: 05/03/2019 10:43:17

Status: Open

Test Set Folder: Pruebas Funcionales

Type: Default

Close Date:

Open Date: 05/03/2019

Target Cycle: SUBS1-1.0.0

Test Set ID: 1

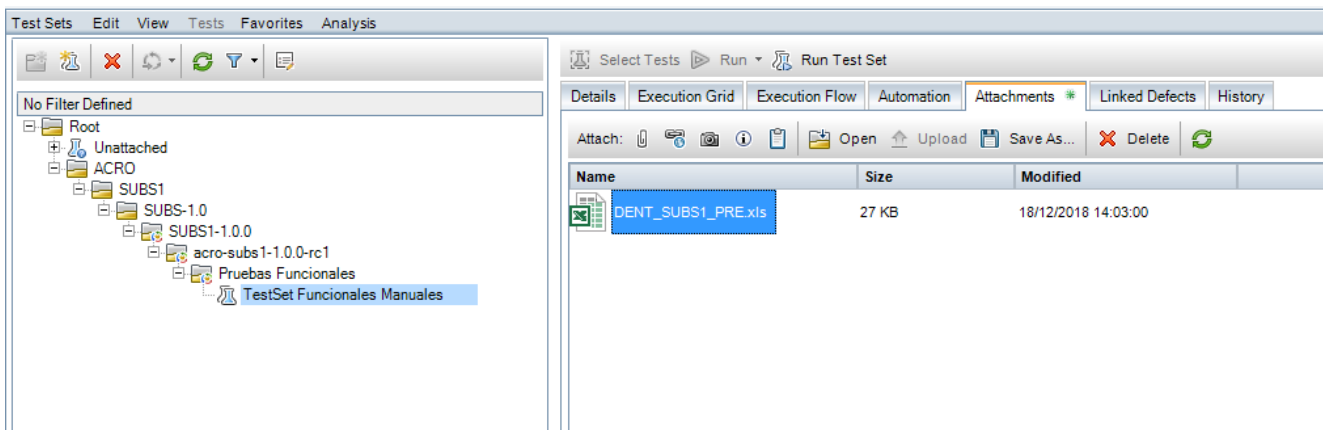
Description

Server Time: 05/03/2019 10:46

2.4.3 Ubicación ficheros de datos de entrada para ejecuciones

2.4.3.1 Pruebas funcionales Manuales

La ubicación de los ficheros con los datos de prueba para la ejecución de pruebas funcionales deberá estar en la pestaña “Attachments” del “Test Set Funcionales Manuales” y será una copia del fichero XLS que Desarrollo habrá adjuntado con los datos actualizados, en la pestaña “Attachments” de la carpeta “Requisitos Funcionales” del Test Plan, en la carpeta de la Release a probar.



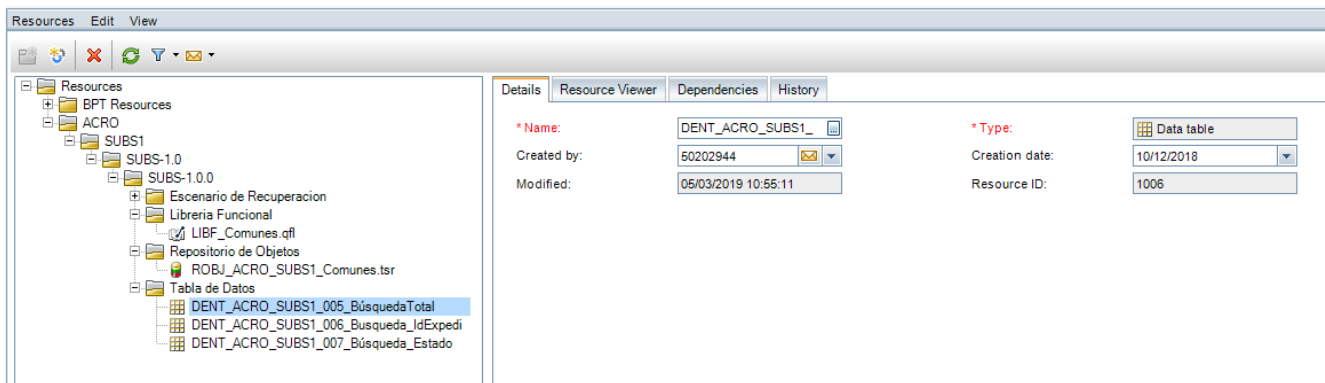
2.4.3.2 Pruebas funcionales Automatizadas

La ubicación de los ficheros con los datos de prueba para la ejecución de pruebas funcionales deberá estar en el menú “Test Resources” dentro del módulo “Testing”. Tendremos la siguiente estructura de carpetas:

- + Nombre Aplicación
 - + [Sistema] (en caso de que tenga subsistemas)
 - + Release
 - + Ciclo
 - + Escenario de Recuperación
 - + Librería Funcional
 - + Repositorio de Objetos
 - + Tabla de Datos

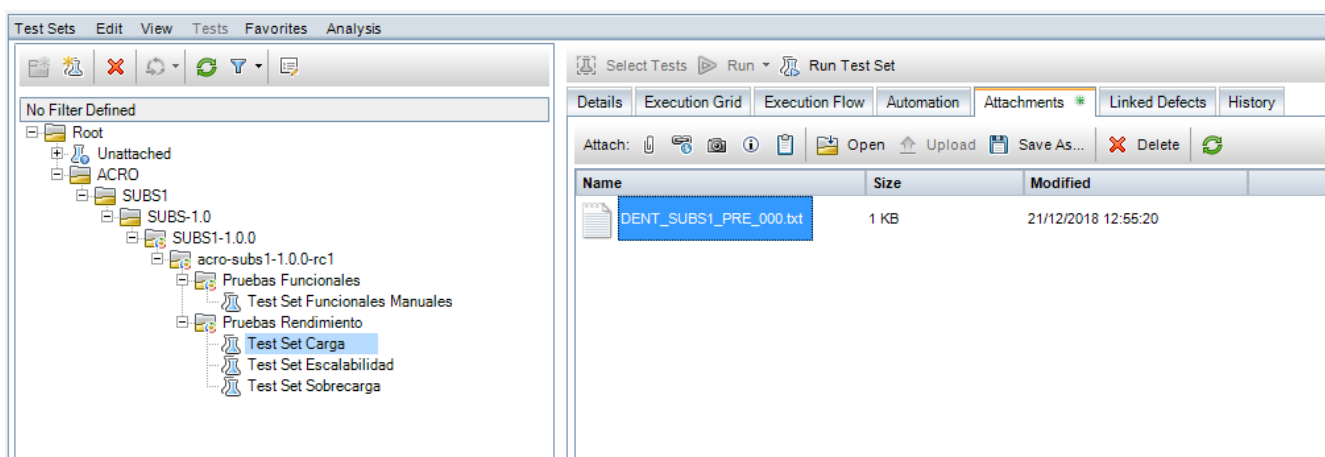


En cada una de las carpetas se incluirán los distintos ficheros necesarios para la ejecución de cada uno de los casos de pruebas a ejecutar en el Test Set creado



2.4.3.3 Pruebas de carga

La ubicación de los ficheros con los datos de prueba para la ejecución de pruebas de carga deberá estar en la pestaña “Attachments” del “Test Set Carga”.





2.4.3.4 Pruebas de sobrecarga

La ubicación de los ficheros con los datos de prueba para la ejecución de pruebas de sobrecarga deberá estar en la pestaña “Attachments” del “Test Set Sobrecarga”.

2.4.3.5 Pruebas escalabilidad

La ubicación de los ficheros con los datos de prueba para la ejecución de pruebas de escalabilidad deberá estar en la pestaña “Attachments” del “Test Set Escalabilidad”.

2.4.3.1 Pruebas seguridad

La ubicación de los ficheros con los datos de prueba para la ejecución de pruebas de seguridad deberá estar en la pestaña “Attachments” de la carpeta “Pruebas Seguridad”.

Name	Size	Modified
ACRO-SUBS1-INF-PSEG-1.0.0.doc	119 KB	25/02/2019 9:46:48
DENT_ACRO_SUBS1_PRE_000.txt	21 KB	05/02/2019 12:12:44

Description:

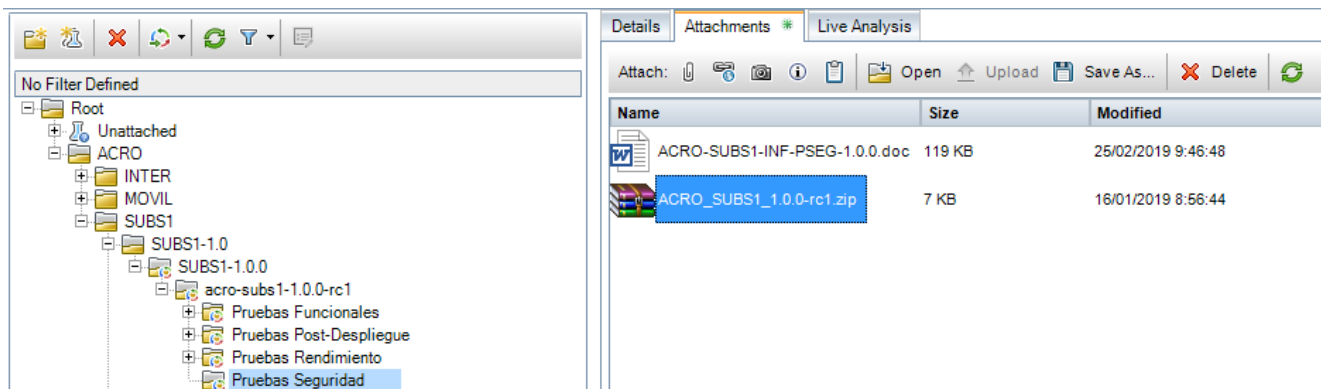
Fichero con datos de entrada para realizar pruebas de Seguridad

2.4.3.2 Fichero .ZIP

Si hubiera varios ficheros a anexar se podría incluir en un ZIP con el nombre de la aplicación y de la build a ejecutar.

Ejemplo:

ACRO_SUBS1_1.0.0-rc1.zip



2.4.3.3 Varias ejecuciones en una misma build

Si hubiera varias ejecuciones en una misma build la ubicación sería la misma pero se agruparía cada lanzamiento en un zip incluyendo al final un índice.

Ejemplo:

ACRO_SUBS1_1.0.0-rc1_1.zip

ACRO_SUBS1_1.0.0-rc1_2.zip

...

2.5 Módulo Defects

2.5.1 Objetivo




Este módulo permite el seguimiento de defectos durante todo el ciclo de vida de los mismos.

Los defectos en ejecución se definen como los errores encontrados por la Oficina de Pruebas durante la ejecución de los diferentes tipos de pruebas en el entorno de preproducción o producción.



2.5.2 Visualización de defectos en una release-ciclo

Para la visualización en el grid de la entidad Defecto en Ejecución se deben realizar los siguientes pasos:

1. Ir al módulo “**Defects**”.
2. Pulsar en el botón “**Set Filter/Sort**” . Aparece la ventana “**Filter**”.
3. Seleccionar los siguientes campos:
 - a. Tipo. Seleccionar el valor “**Defecto en Ejecución**”.
 - b. Detected in Release. Seleccionar la release sobre la que se quieren obtener los defectos dados de alta.
 - c. Detected in Cycle. Seleccionar el ciclo sobre el que se quieren obtener los defectos dados de alta.
 - d. Detected in Build. Seleccionar la Build concreta sobre la que se quieren obtener los defectos dados de alta



Filter defects

Filter: Tipo["Defecto en Ejecución"]; Detected in Build[acro-sub1-1.0.0-rc1]

Filter Cross Filter View Order Group

Field Name	Filter Condition
Detected in Build	acro-sub1-1.0.0-rc1
Detected in Cycle	
Detected in Release	
Detected in Version	
Detected on Date	
Documentación Asociada	
Entorno de Ejecución	
Estimated Fix Time	
Horas / hombre	
Modified	
Origen del Cambio	
Planned Closing Version	
Priority	
Project	
Recursos totales	
Reproducible	
Severity	
Solved in Build	
Status	
Subject	
Summary	
Target Cycle	
Target Release	
Tipo	"Defecto en Ejecución"
Tipo de Cambio	
Tipo de defecto	

OK Cancel Help

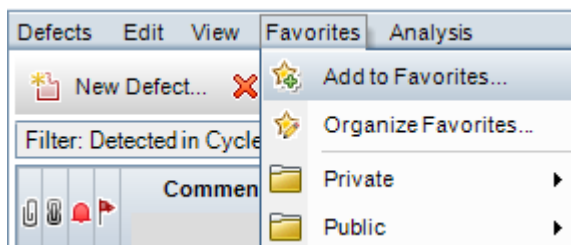
4. Pulsar el botón **OK**. En el grid aparece la lista de defectos en ejecución detectados en una release-ciclo-Build concreto.



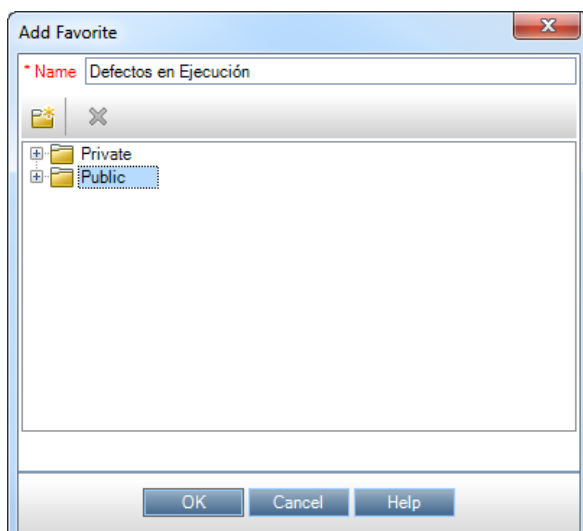
Filter: Detected in Cycle["Releases\TEPV\V00_v00\R001"];Detected in Release["Releases\TEPV\V00_v00"]											
Comments	Tipo	Defect ID	Description	Detected By	Detected on...	Modified	Status	Summary	Detected in...	Detected in...	Subject
	Defecto en Ejecu...	1	Defecto en ejecu...	50202944	27/12/2018	27/12/2018 11:04	New	Defecto en ejecu...	V00_v00_R001	V00_v00	RF-01 Autenticati...

Si se quiere almacenar el filtro como una vista para un acceso más rápido a la información, se deben realizar las siguientes acciones:

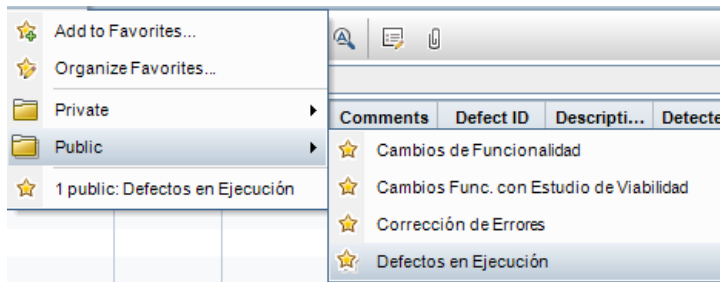
1. Ir a la barra de **Menú / Favorites** y seleccionar **“Add to Favorites”**



2. Aparece la ventana **Add Favorite** donde se introduce en el campo **Name** un nombre representativo de la búsqueda que se quiere almacenar. Seleccionar **Public** o **Private** dependiendo del ámbito de visualización de la vista.



3. Pulsar el botón **OK**.
4. Para seleccionar el filtro guardado ir al campo **Favorite** y en la lista seleccionar el nombre de la vista creada.



2.5.3 Modificación de defectos

Los pasos para la modificación de un defecto en ejecución son los siguientes:

1. Ir al módulo “**Defects**”.
2. Seleccionar en el grid el defecto en ejecución que se quiere modificar.

Defects Edit View Favorites Analysis							
New Defect... [Iconos]							
Filter: Detected in Build[acro-sub1-1.0.0-rc1]; Tipo[Defecto en Ejecución]							
	Defect ID	Subject	Tipo	Priority	Detected on...	Detected in...	Description
1			Defecto en Ejecución		05/03/2019	SUBS1-1.0.0	Defecto en ejecución A

3. Hacer doble clic sobre el registro del defecto en ejecución para editarlo. Se abre la ventana **Defect Details**.



4. Se permite la modificación de los siguientes campos:

- Summary.
- Description.
- Target Release.
- Target Cycle.
- Comments.
- **Status.** Según el perfil del usuario el cambio de estado permitido es distinto:
 - Jefe de proyecto. De **In Progress** a **Rejected** u **Open**.
 - Desarrollo. De New a In Progress, y de Rejected by Pruebas a In Progress.

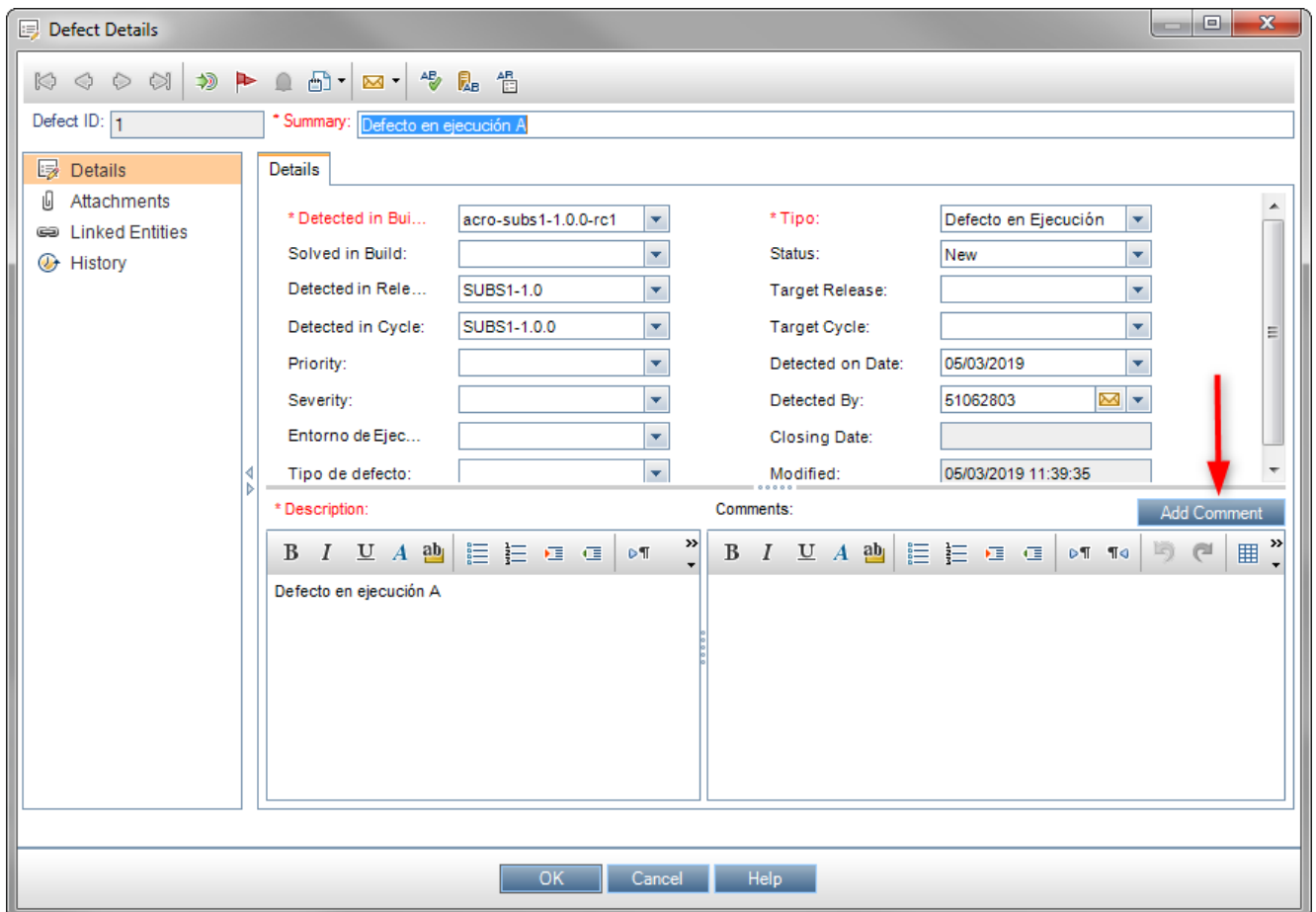
5. Una vez realiza la modificación de los campos se pulsa el botón **OK**. La ventana **Defect Details** se cierra y se retorna al grid del módulo de defecto.

2.5.4 Añadir comentarios

Todos los usuarios involucrados en los defectos deben introducir cualquier comentario que crean pueda servir de interés para la resolución del defecto en ejecución. Este campo también sirve para la comunicación entre los usuarios.

Los pasos para la introducción de un comentario en el campo **Comments** son los siguientes:

1. En la ventana **Defect Details** de un defecto en ejecución, pulsar el botón **Add Comment**.



2. En el campo **Comments** se introduce automáticamente el siguiente literal: **Nombre usuario QC <User QC>, Fecha Actual:**



Defect Details

Defect ID: 1 * Summary: Defecto en ejecución A

Details

* Detected in Bui... acro-subst1-1.0.0-rc1

Solved in Build:

Detected in Rele... SUBS1-1.0

Detected in Cycle: SUBS1-1.0.0

Priority:

Severity:

Entorno de Ejec...

Tipo de defecto:

* Tipo: Defecto en Ejecución

Status: New

Target Release:

Target Cycle:

Detected on Date: 05/03/2019

Detected By: 51062803

Closing Date:

Modified: 05/03/2019 11:39:35

* Description: Defecto en ejecución A

Comments: Irene del Río Gimeno (803) <51062803>, 05/03/2019:

OK Cancel Help

3. Escribir el comentario que se quiere realizar a continuación del literal automático introducido.



4. Pulsar el botón **OK** de la ventana **Defect Details**. La ventana **Defect Details** se cierra y se retorna al grid del módulo de defecto.

3 Gestión de evolutivos

Una vez que ya se conoce cómo se gestionan los proyectos se describe la manera de proceder en ALM cuando arranca un evolutivo.

3.1 Descripción

Antes de comenzar hay que indicar que **es muy importante no modificar lo ya validado y aprobado en fases anteriores**. Cualquier cambio o modificación que se desee realizar en la



información existente en ALM deberá ser notificada por correo electrónico para su evaluación a la Oficina de Pruebas.

Las opciones posibles en un evolutivo en **ALM** (para más detalle sobre evolutivos ver anexo de gestión de la configuración) son:

- Inserción de nuevos requisitos
- Eliminación de requisitos
- Modificación de requisitos

En este apartado vamos a explicar en detalle como enlazar requisitos y casos de pruebas de las diferentes versiones manteniendo su trazabilidad.

Nuestro ámbito estará acotado a las tres posibles situaciones indicadas anteriormente y su correspondencia en ALM será la siguiente:

- Duplicar en ALM los requisitos que cambien
- Duplicar en ALM los requisitos en los que los casos de prueba asociados cambien.
- Duplicar en ALM los requisitos que están directamente relacionados (relación de primer nivel) con requisitos que cambian (trace to y trace from).
- Asociar en ALM los requisitos que no cambien, que no cambien su plan de pruebas y que no cambien los requisitos que están relacionados con la nueva versión.

Con estas pautas en ALM y ayudándonos de los filtros mantendremos la trazabilidad entre las diferentes versiones.

3.2 Ejemplo

Para detallar los pasos a realizar en los diferentes módulos nos basaremos en un proyecto de ejemplo didáctico que utilizaremos durante toda la guía.

Ejemplo:

Proyecto con dos versiones [SUBS1-]1.0 y [SUBS1-]1.1, siendo [SUBS1-]1.1 un evolutivo que añade funcionalidad (RF-04 Convertir) y modifica casos de prueba de un requisito que no cambia funcionalidad (RF-02 Ver detalles del vehículo).

La estructura del proyecto en la versión [SUBS-]1.0 (inicial) la mostramos en la siguiente figura:

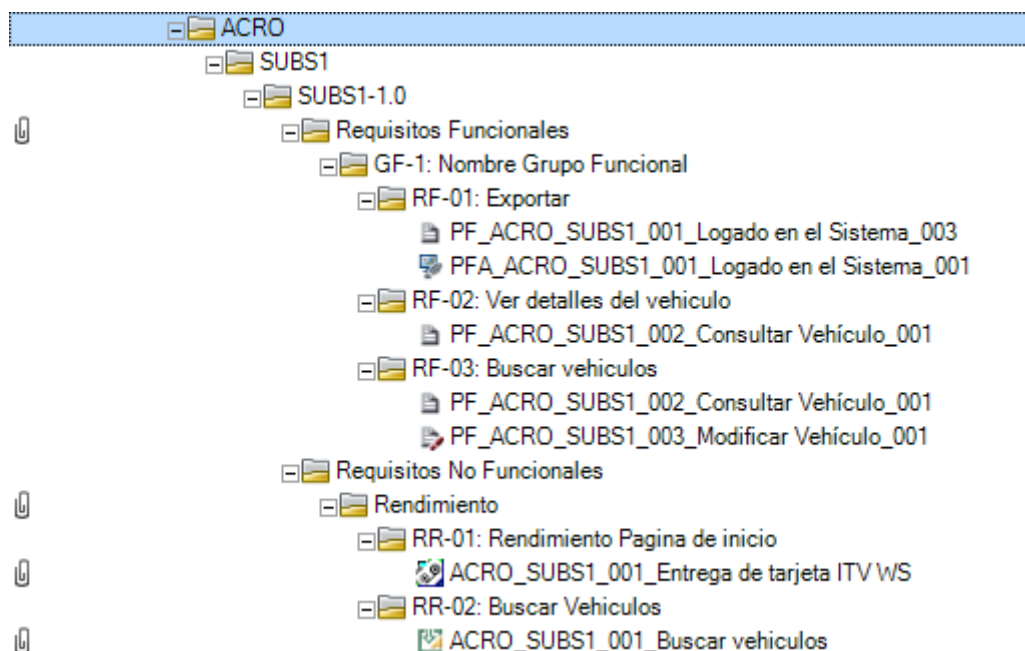


Ilustración 7. Estructura Test Plan SUBS1-1.0

3.2.1 Módulo Management

Dentro del módulo **Management** deberemos crear la nueva versión y ciclo. En este ejemplo, será la versión [SUBS-]1.1 y la revisión [SUBS-]1.1.0.

La nueva versión [SUBS-]1.1 deberá estar asociada obligatoriamente a un ciclo [SUBS-]1.1.0.

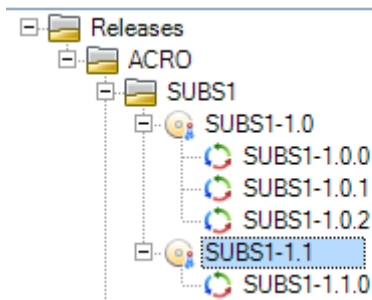


Ilustración 8. Estructura Releases [SUBS-]1.1

3.2.2 Módulo Requirements

En el módulo **Requirements** deberemos crear la correspondiente estructura con la nueva versión [SUBS-]1.1, tal y como se indica en la figura siguiente.

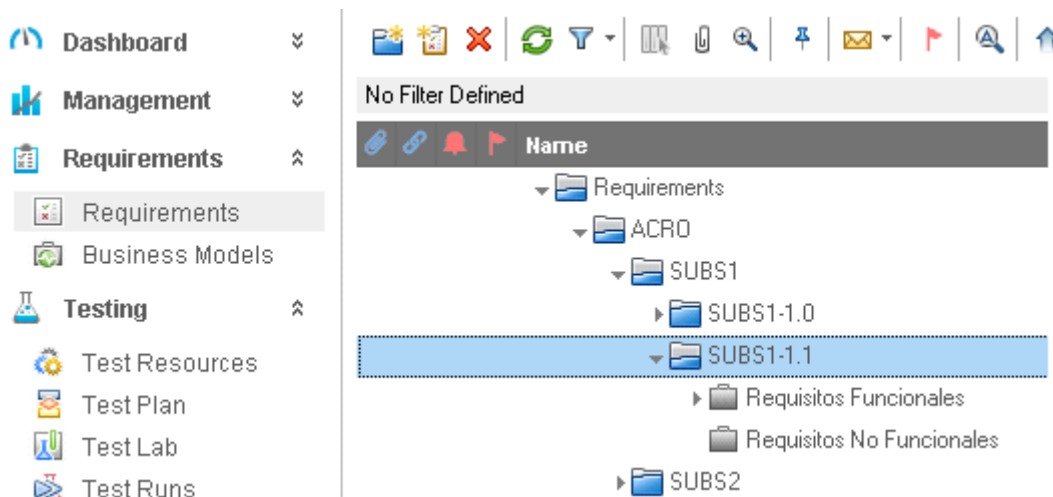
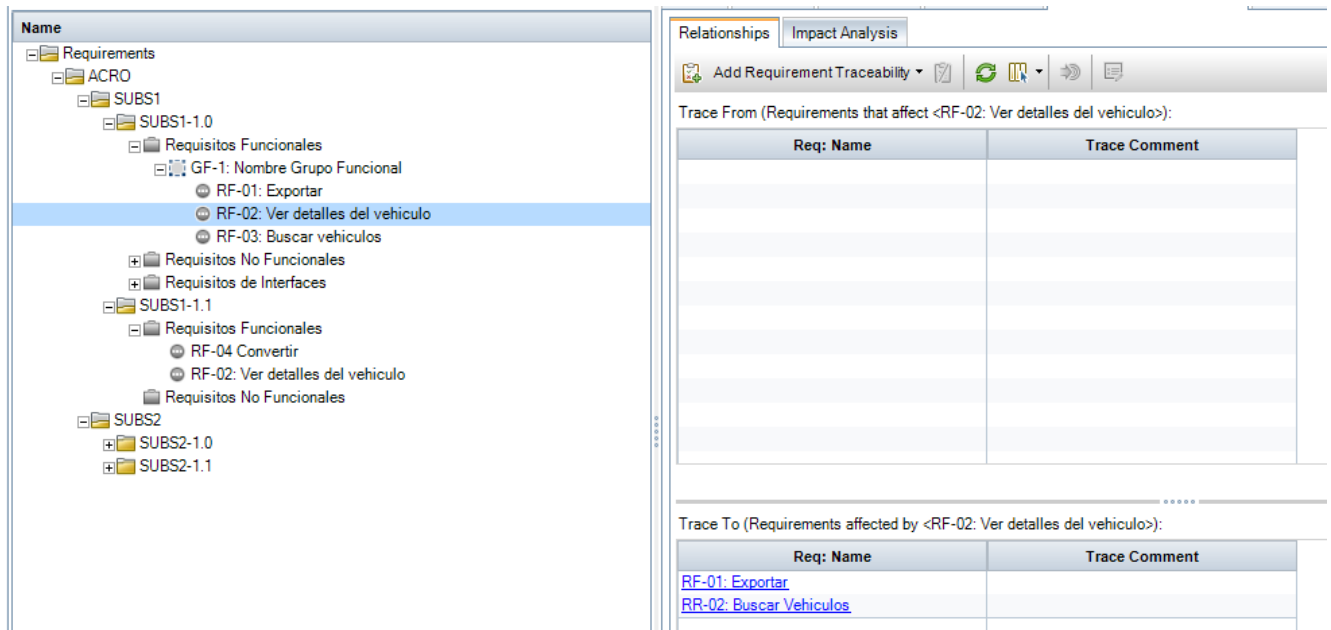


Ilustración 9. Estructura Requirements [SUBS-]1.1

En nuestro ejemplo deberemos crear:

- Un nuevo requisito funcional (RF-04 Convertir) en la versión [SUBS-]1.1
- El requisito (RF-02 Exportar) tanto si es para el caso en que cambie dicho requisito funcionalmente, o porque los casos de prueba que cubren dicho requisito en la versión [SUBS-]1.1 cambien respecto a los que tenían en la [SUBS-]1.0

- El requisito funcional (RF-01 Exportar) por estar relacionado con el requisito funcional (RF-02 Ver detalles del vehículo).



The screenshot shows the Requirements tool interface. On the left, the 'Name' pane displays a hierarchical tree of requirements. The 'Requirements' folder is expanded, showing 'ACRO' and 'SUBS1'. Under 'SUBS1', 'SUBS1-1.0' is expanded, showing 'Requisitos Funcionales' with 'GF-1: Nombre Grupo Funcional'. Under 'GF-1', 'RF-01: Exportar' and 'RF-02: Ver detalles del vehículo' are listed. 'RF-02' is selected. Below 'GF-1', 'Requisitos No Funcionales' and 'Requisitos de Interfaces' are shown. 'SUBS1-1.1' is also expanded, showing 'Requisitos Funcionales' with 'RF-04 Convertir' and 'RF-02: Ver detalles del vehículo', and 'Requisitos No Funcionales'. 'SUBS2' is expanded, showing 'SUBS2-1.0' and 'SUBS2-1.1'. On the right, the 'Impact Analysis' tab is active. It shows 'Trace From (Requirements that affect <RF-02: Ver detalles del vehículo>):' and a table with columns 'Req: Name' and 'Trace Comment'. Below this, 'Trace To (Requirements affected by <RF-02: Ver detalles del vehículo>):' is shown with a table containing 'RF-01: Exportar' and 'RR-02: Buscar Vehiculos'.

Req: Name	Trace Comment
RF-01: Exportar	
RR-02: Buscar Vehiculos	

Ilustración 10. Estructura Requirements [SUBS-]1.0

Por lo tanto, la estructura nos quedaría de la siguiente manera:

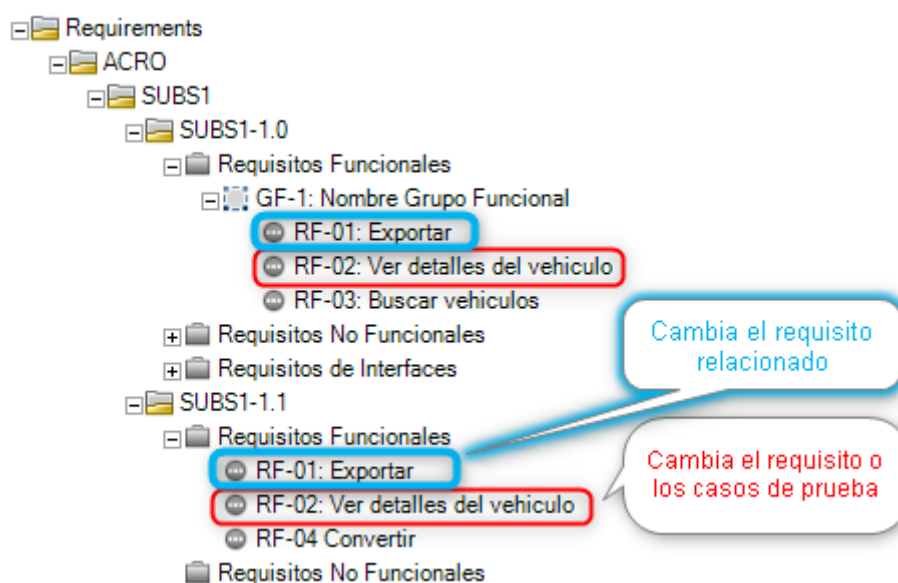
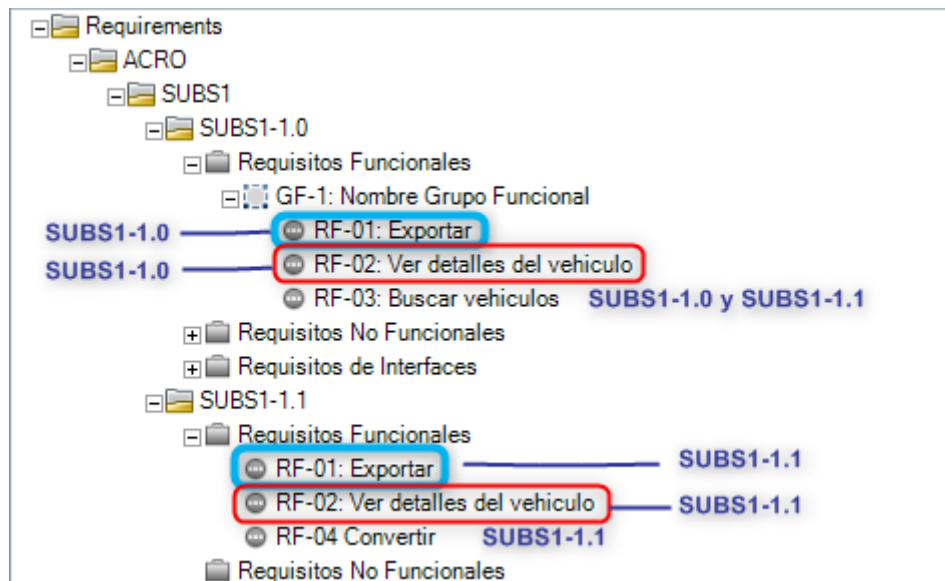


Ilustración 11. Estructura Requirements [SUBS-]1.1

Y cada requisito se debería asociar con su release correspondiente.

**Ilustración 12. Relación de requisitos [SUBS-]1.0 y [SUBS-]1.1**

Cuando un requisito cambia porque está relacionado con otro requisito que cambia en la pestaña “Description” de ALM se debe indicar en una nota en la que se haga mención a dicha circunstancia.

Ejemplo:



No Filter Defined

Name

- Requirements
 - ACRO
 - SUBS1
 - SUBS1-1.0
 - Requisitos Funcionales
 - GF-1: Nombre Grupo Funcional
 - RF-01: Exportar
 - RF-02: Ver detalles del vehiculo
 - RF-03: Buscar vehiculos
 - Requisitos No Funcionales
 - Requisitos de Interfaces
 - SUBS1-1.1
 - Requisitos Funcionales
 - GF-1: Nombre Grupo Funcional
 - RF-01: Exportar
 - RF-02: Ver detalles del vehiculo
 - RF-04: Convertir
 - Requisitos No Funcionales
 - SUBS2
 - SUBS2-1.0
 - SUBS2-1.1

Details

* Name: RF-01: Exportar

* Requirement Type: Functional

* Target Release: SUBS1-1.1

Creation Date: 05/03/2019

Direct Cover Status: Not Covered

Old Type (obsolete):

ReqID: 259

* Priority: 3-High

* Target Cycle: SUBS1-1.1.0

Author: 51062803

Creation Time: 12:09:43

Modified: 05/03/2019 12:26:17

Product:

Reviewed: Not Reviewed

Description

Comments

Requisito Exportar

NOTA: El requisito se duplica por estar relacionado con el requisito RF-02 Ver detalles del vehiculo que ha cambiado (cambio de funcionalidad o del plan de pruebas)

Ilustración 13. Descripción de requisitos que cambian

3.2.3 Módulo Test Plan

En el módulo Test Plan en la versión [SUBS-]1.0 tendríamos la siguiente estructura.

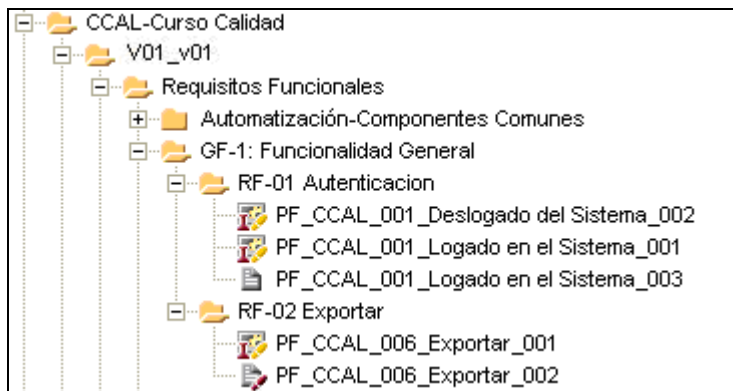


Ilustración 14. Estructura Test Plan [SUBS-]1.0

Generaríamos la estructura de carpetas para la [SUBS-]1.1 y añadiríamos la estructura de carpetas de los requisitos afectados y los casos de prueba nuevos y los que cambian.

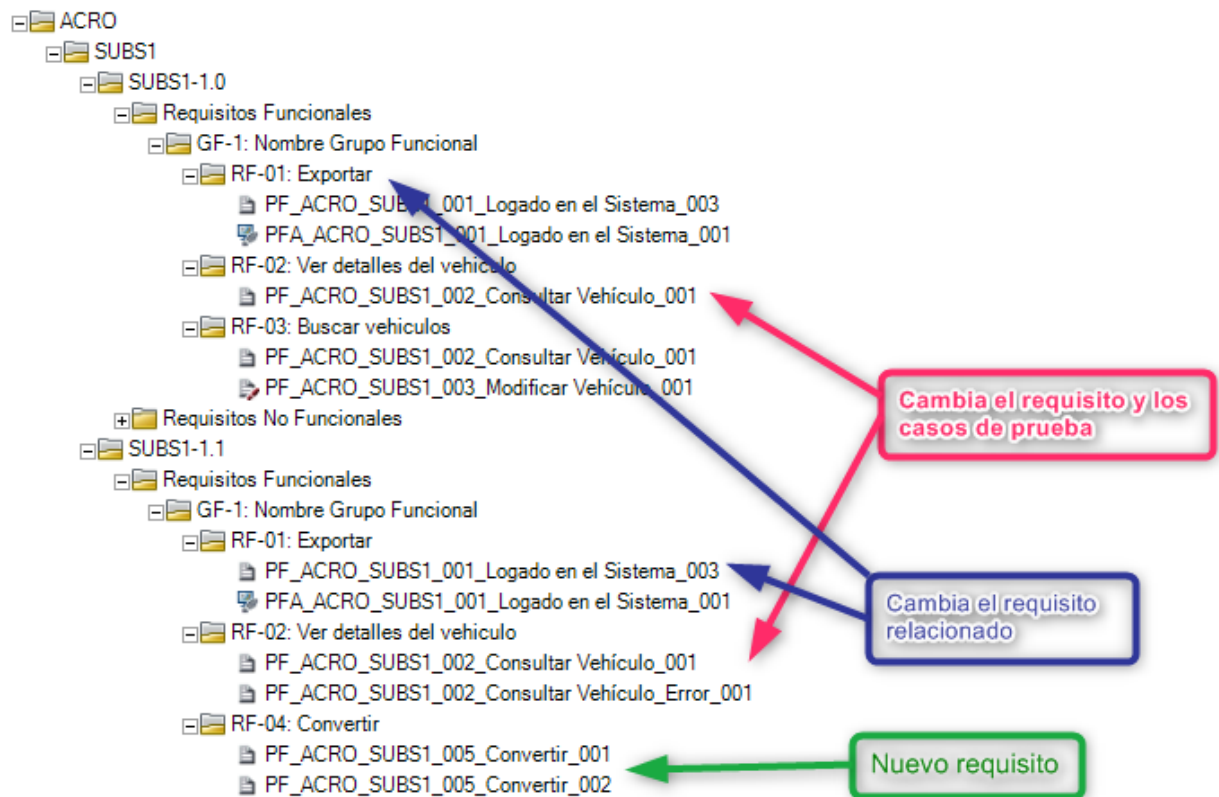


Ilustración 15. Estructura Test Plan V01_v02

Ahora debemos enlazar cada caso de prueba con su requisito correspondiente teniendo en cuenta si es de la versión anterior [SUBS-]1.0 o de la nueva [SUBS-]1.1.

NOTA: Poner atención a los **componentes comunes manuales** (CCM's) que cambian en la nueva versión:

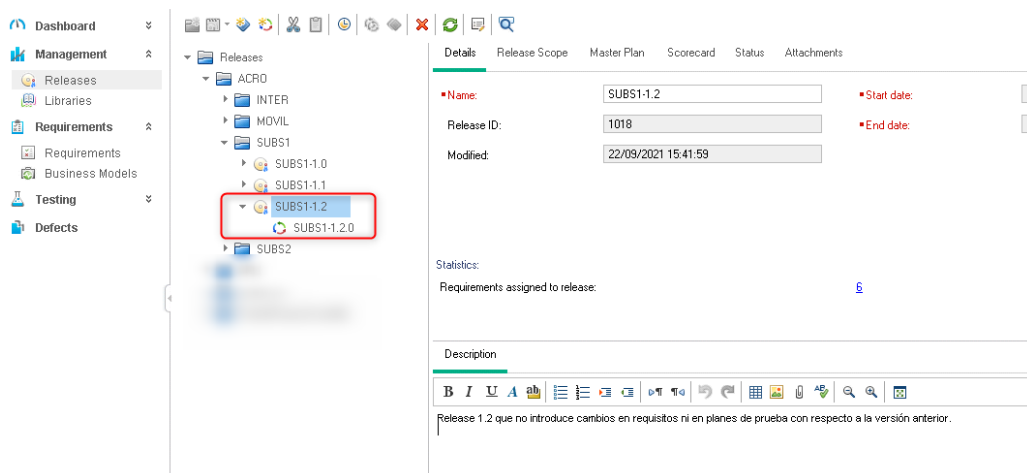
- Los CCM's que cambian se duplicarán en la carpeta "PF Manuales-Componentes Comunes" de la nueva [SUBS-]1.1. Allí serán actualizados.
- Los casos de prueba que invocan con "Call" a un componente común que ha cambiado, se duplicarán en la carpeta de la nueva [SUBS-]1.1, y borrarán el Step con el "Call" al CCM antiguo y añadirán el "Call" al CCM de la nueva [SUBS-]1.1.

- Los requisitos asociados a los casos de prueba afectados, también deberán ser duplicados en el módulo “Requirements”, con todo lo que ello implica, tal y como se ha explicado en el apartado anterior.

3.3 Caso especial – Evolutivo sin cambios de requisitos/casos de prueba

En alguna ocasión puede darse la circunstancia de que haya una nueva release (y ciclo), pero que no haya nuevos/modificación/eliminación de requisitos ni casos de prueba. En este caso también hay que hacer algunos cambios en ALM para reflejar este hecho y que las posibles ejecuciones sobre esa release queden perfectamente trazadas. Los pasos a realizar serían:

- Módulo Management: crear la release y ciclo como se ha explicado en el punto 2.1.1. En la descripción se hará notar que no hay modificaciones en requisitos/casos de prueba con respecto a la anterior versión.



La imagen muestra la interfaz de usuario de ALM. En el panel izquierdo, se expande el menú 'Management' y se selecciona 'Releases'. En el árbol de carpetas, se crea una nueva release llamada 'SUBS1-1.2' (destacada con un recuadro rojo). El panel principal muestra los detalles de esta release:

- Name:** SUBS1-1.2
- Release ID:** 1018
- Modified:** 22/09/2021 15:41:59
- Start date:** 01
- End date:** 30
- Statistics:** Requirements assigned to release: 6
- Description:** Release 1.2 que no introduce cambios en requisitos ni en planes de prueba con respecto a la versión anterior.

- Módulo Requirements: crear la carpeta correspondiente a la Release, con la información similar a la de versiones anteriores y una nota indicando que no hay cambios sobre la versión anterior. Se crearán asimismo las carpetas de “Requisitos funcionales” y “Requisitos no funcionales”.



Name: SUBS1-1.2

Requirement Type: Folder

Author: 51062803

Creation Date: 22/09/2021

Creation Time: 15:37:23

Modified: 22/09/2021 15:38:47

Priority:

Product:

ReqID: 546

Reviewed: Not Reviewed

Target Cycle:

Target Release:

Description:

Requisitos de la aplicación ACRO-SUBS1-1.2

Nota: esta versión no presenta cambios sobre la versión anterior

1. Objeto
El propósito de este documento es definir los requisitos que debe cumplir el sistema de Escuelas Particulares de Conductores (GATO), a fin de conocer los requerimientos y el alcance de este sistema y poder verificar que cumple con las necesidades de la Dirección General de Tráfico.

2. Alcance
El alcance de este documento se centra en la definición de las funcionalidades y requisitos que debe cumplir el sistema.

3. Términos y acrónimos
En este apartado se incluyen, organizados por orden alfabético, los acrónimos y términos específicos de la aplicación, que se utilizan en el propio

- Módulo Test Plan: crear la carpeta correspondiente a la Release, con la información similar a la de versiones anteriores y una nota indicando que no hay cambios sobre la versión anterior. Se crearán asimismo las carpetas de “Requisitos funcionales” y “Requisitos no funcionales”.

Name: SUBS1-1.2

Subject: Unattached

ACRO: SUBS1

Regresión Funcional Automatizada: SUBS1-1.0

SUBS1-1.1: SUBS1-1.2

Requisitos Funcionales:

Requisitos No Funcionales:

ZZ_Puebas_Automatizadas:

Attachments:

Los datos de acceso a la aplicación son los siguientes:

*URL: http://pre-apl.trafico.es/SUBS1/Inicio

* Usuario: 99999800

* Password: DGT800

En la pestaña “Attachments” de la carpeta “Requisitos Funcionales” del Test Plan, se adjuntará el fichero de datos XLS con los valores de los datos de prueba revisados y actualizados para ejecutar las pruebas sobre la nueva Release.

3.4 Comprobación

Para comprobar que nuestro plan de pruebas está correctamente generado y enlazado con los casos de prueba que les aplica podemos hacer la siguiente comprobación.

Realizar un filtro a nivel del campo “**Target Release**” y “**Target Cycle**”.

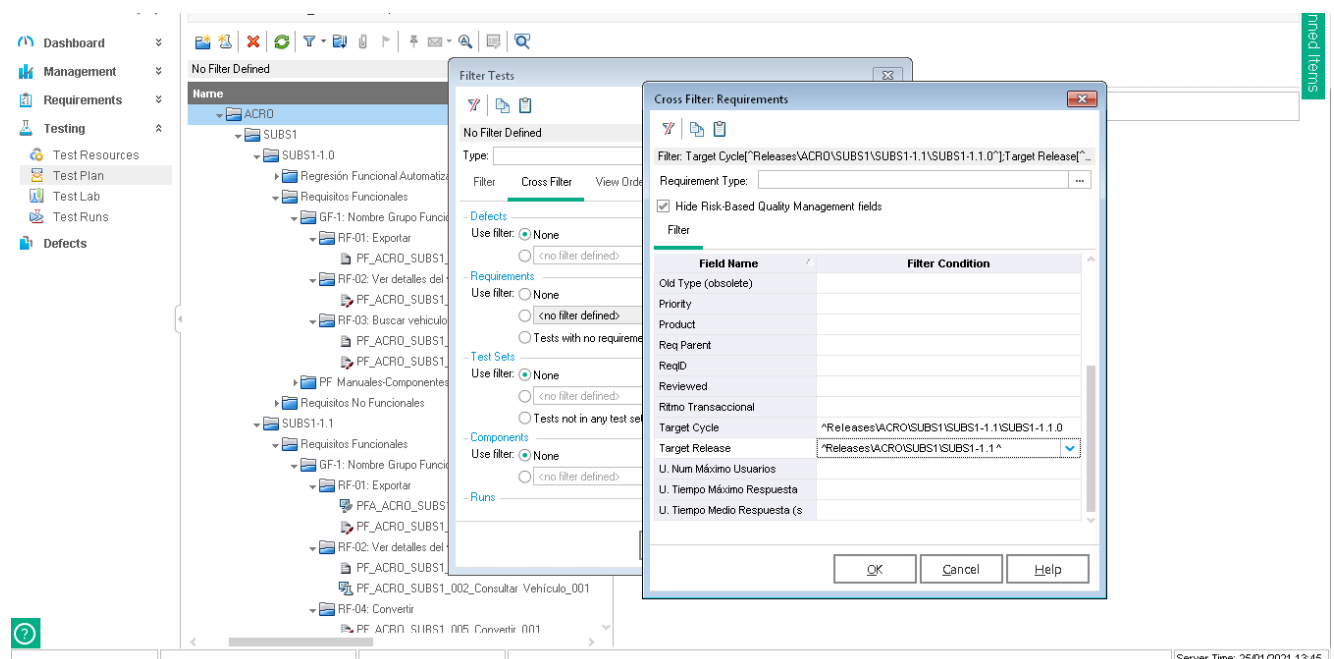


Ilustración 16. Filtro Test Plan

Al aplicar el filtro, nos deberían quedar sólo los casos de prueba que aplican a la release [SUBS-]1.1.