

Universidad de Valladolid

Escuela de Ingeniería Informática

TRABAJO FIN DE GRADO

Grado en Ingeniería Informática

(Mención Ingeniería de Software)

ServiceNow: gestión de proyectos y recursos a través
de ITBM

Autor:

D. Alejandro Muñoz García

Tutora:

D^a M. Mercedes Martínez González

Resumen

Se ha realizado un desarrollo mediante la herramienta de *ServiceNow* la cual es una Plataforma como Servicio (PaaS), una categoría de servicios basados en la nube que provee de la infraestructura necesaria para desarrollar, ejecutar y administrar aplicaciones. Esta plataforma nos permite realizar desarrollos para cualquier departamento de una empresa, aunque habitualmente se la incluye como una plataforma de tipo ITSM (*Informatic Technology Service Management*).

En este Trabajo de Fin de Grado se han desarrollado dos aplicaciones que se incluyen dentro de la herramienta para la gestión de la empresa. Por un lado, se ha desarrollado un sistema para crear y gestionar proyectos de la empresa en todas sus etapas, así como demandas, imputaciones de tiempo trabajado y otros recursos propios de la plataforma cuyo objetivo es permitir a la empresa medir la rentabilidad de los proyectos que realiza y poder mejorar la asignación de recursos que realiza en el futuro.

Esto permite controlar tanto los proyectos como todas las subtareas asociadas a los mismos, calculando los gastos y las horas empleadas en el mismo, todo ello mediante planes de recursos, *Time Cards* y la asignación recursos entre los distintos trabajadores. También se pueden asociar pruebas a los proyectos creados.

Por otro lado, se ha desarrollado otra parte denominada como *Test Management*, que permite subir a la aplicación y gestionar los distintos casos de pruebas y *Testeos* que se llevan a cabo dentro de un proyecto para comprobar que las implementaciones requeridas por un cliente están incluidas. Se pueden planear los distintos *Test* que se pretenden realizar agrupándolos en conjuntos, y en cada *Test* se registran tanto los tiempos de ejecución como los resultados de todos los pasos que se realizan para completar el *Test*. Ofrece un sistema intuitivo y flexible que permite adaptar los *Testeos* que se realizan con un sistema de versiones en caso de que el proyecto vaya evolucionando.

Índice General

Resumen.....	2
Índice General	3
Indice de Figuras	6
Índice de Tablas.....	9
1 Introducción.....	13
1.1 Motivación.....	13
1.2 Objetivos.....	13
1.3 Contexto	14
1.4 Procedimiento	15
1.5 Tecnologías utilizadas.....	15
1.6 Estructura del documento.....	15
2 Metodología de trabajo	18
2.1 Scrum.....	18
2.2 Comunicado de requisitos iniciales	19
2.3 Desarrollos de requisitos de cliente	19
2.4 Desarrollos de elementos técnicos	23
2.5 Roles	23
2.6 Desarrollo de historias de usuario	23
2.7 Casos de uso	31
2.8 Implementación de los elementos definidos.....	50
2.9 Diagramas de flujo.....	51
2.10 Retrospectiva.....	56
2.11 Despliegue y mantenimiento del desarrollo	56
2.12 Línea temporal de trabajo.....	56
3 ¿Por qué se necesitan estos desarrollos?.....	59
3.1 Desarrollo Gestión de Proyectos.....	59
3.2 Desarrollo <i>Test Management</i>	60
4 Elaboración de los ciclos de vida de las entidades del proceso	63
4.1 Ciclos de vida de Gestión de Proyectos.....	63
4.2 Ciclos de vida de <i>Test Management</i>	65
5 Elementos técnicos de la aplicación	68
5.1 Entidades del proceso de proyecto.....	70
5.2 Entidades del proceso de <i>Test Management</i> :.....	80
6 Implementación de entidades de funcionalidad.....	97

6.1	<i>Business Rules:</i>	97
6.2	<i>Client Script</i>	99
6.3	<i>Script Includes</i>	100
7	Plan de pruebas	103
7.1	Plan de Prueba Gestión de Proyectos	103
7.2	Mejoras para Gestión de Proyectos	116
7.3	Datos de Prueba <i>Test Management</i>	117
7.4	Mejoras para <i>Test Management</i>	128
8	Conclusiones	131
9	Bibliografía	133
10	Anexos	136
10.1	Anexos Gestión de proyectos	136
10.2	Anexos <i>Test Management</i>	149

Indice de Figuras

Figura 2.1.1 Metodología Scrum	18
Figura 2.12.1 Diagrama de tiempo del desarrollo	57
Figura 4.1.1 Ciclo de Vida Time Card	64
Figura 4.1.2 Ciclo de Vida Plan de Recursos	64
Figura 4.2.1 Ciclo de Vida de Versión de Test	65
Figura 5.1.1 Diagrama de clases de Gestión de proyectos	71
Figura 10.1.1 Menu SilverStorm Implementation	143
Figura 10.1.2 Menu Project y Project Task	143
Figura 10.1.3 Tabla de proyectos.....	144
Figura 10.1.4 Campos Coste y Esfuerzo de Proyecto	144
Figura 10.1.5 Pestaña Cost Price en Proyecto	144
Figura 10.1.6 Formulario Tarea de proyecto	144
Figura 10.1.7 Costes por Rol de Tarea de Proyecto.....	145
Figura 10.1.8 Esfuerzo por Rol de Tareas de Proyecto.....	145
Figura 10.1.9 Related list de Time Card en Tareas de Proyecto.....	145
Figura 10.1.10 Related list de Planes de Prueba en Tareas de Proyecto	145
Figura 10.1.11 Formulario de creación de Plan de Recursos	146
Figura 10.1.12 Formulario de Plan de Recursos creado desde Demanda.....	146
Figura 10.1.13 Formulario de Plan de Recursos	146
Figura 10.1.14 Related list de Asignación de Recursos en Plan de Recursos	146
Figura 10.1.15 Formulario de Asignación de Recursos	147
Figura 10.1.16 Formulario de Time Card	147
Figura 10.1.17 Lista de Hojas de Tarifa de Gasto	147
Figura 10.1.18 Formulario de Hoja de Tarifa de Gasto	148
Figura 10.1.19 Lista de Recursos de Rol	148
Figura 10.2.1 Menú Test Management 2.0	158
Figura 10.2.2 Dashboard Test Management pestaña 1	158
Figura 10.2.3 Dashboard Test Management pestaña 2	159
Figura 10.2.4 Dashboard Test Management pestaña 3	159
Figura 10.2.5 Lista de Conjunto de Test	159
Figura 10.2.6 Formulario Conjunto de Test y Related list a Test.....	160
Figura 10.2.7 UI Page Add Test to Test Set.....	160
Figura 10.2.8 UI Page Update Test Code con mensaje de error.....	160
Figura 10.2.9 Lista de Test	160
Figura 10.2.10 Formulario de Test con related list Versiones de Test	161
Figura 10.2.11 Formulario de Version de Test y related list Paso	161
Figura 10.2.12 Formulario de Paso	161
Figura 10.2.13 Formulario Resultado de Test.....	162
Figura 10.2.14 Formulario Test Run con related list Resultados de Test	162
Figura 10.2.15 Formulario Plan de Test con related list Ciclo de Test	162
Figura 10.2.16 Formulario Ciclo de Test con related list Software de Ejecución de Test	163
Figura 10.2.17 Formulario Software de Ejecución de Test con related list Asignación Ejecución de Test	163

Figura 10.2.18 Tabla Asignación de Ejecución de Test	163
Figura 10.2.19 Related list de Software de Ejecución de Test en formulario Story	164
Figura 10.2.20 UI Page Run Test	164
Figura 10.2.21 Run Test con Paso que necesita verificación en blocked	164
Figura 10.2.22 Database View Test Export	165
Figura 10.2.23 Database View Test Run Results	165

Índice de Tablas

Tabla 2.6.1 Historia de usuario crear proyecto.....	23
Tabla 2.6.2 Historia de Usuario crear tarea de proyecto.....	23
Tabla 2.6.3 Historia de usuario Crear Hojas de Tarifa de Gasto	24
Tabla 2.6.4 Historia de Usuario Añadir esfuerzo planeado por rol en una tarea de proyecto.....	24
Tabla 2.6.5 Historia de usuario ver esfuerzo y coste planeado a un proyecto.....	24
Tabla 2.6.6 Historia de usuario Ver esfuerzo y coste actual por rol en el proyecto.....	24
Tabla 2.6.7 Historia de usuario Ver esfuerzo y coste actual de un proyecto	25
Tabla 2.6.8 Historia de usuario Ver esfuerzo y coste proyectado por rol en una tarea de proyecto	25
Tabla 2.6.9 Historia de usuario Ver esfuerzo y coste proyectado de un proyecto.....	25
Tabla 2.6.10 Historia de usuario Crear una Time Card	25
Tabla 2.6.11 Historia de usuario Aprobar una Time Card.....	25
Tabla 2.6.12 Historia de usuario Rechazar una Time Card	26
Tabla 2.6.13 Historia de usuario crear un plan de recursos	26
Tabla 2.6.14 Historia de usuario Crear un plan de recursos.....	26
Tabla 2.6.15 Historia de usuario Crear asignaciones de recursos	26
Tabla 2.6.16 Historia de usuario Cancelar un plan de recursos.....	26
Tabla 2.6.17 Historia de usuario Completar un plan de recursos	27
Tabla 2.6.18 Historia de usuario Visualizar coste habitual de un proyecto	27
Tabla 2.6.19 Historia de Usuario Crear un Conjunto de Test	27
Tabla 2.6.20 Historia de usuario Crear un Test.....	27
Tabla 2.6.21 Historia de Usuario Crear una Versión de Test	27
Tabla 2.6.22 Historia de usuario Crear un Paso	28
Tabla 2.6.23 Historia de Usuario Crear un Entorno de Test	28
Tabla 2.6.24 Historia de usuario Asignar Test a los Conjuntos de Test	28
Tabla 2.6.25 Historia de usuario Actualizar los códigos de los Test de un conjunto de Test en bloque	28
Tabla 2.6.26 Historia de usuario Actualizar los códigos de los pasos automáticamente.....	28
Tabla 2.6.27 Historia de usuario Ejecutar un Test individualmente.....	28
Tabla 2.6.28 Historia de usuario Pausar la ejecución de un Test	29
Tabla 2.6.29 Historia de usuario Crear un Plan de Test.....	29
Tabla 2.6.30 Historia de usuario Crear un Ciclo de Test.....	29
Tabla 2.6.31 Historia de usuario Crear un Software de Ejecución de Test.....	29
Tabla 2.6.32 Historia de usuario Asignar un Software de Ejecución de Test a una story	29
Tabla 2.6.33 Historia de usuario Crear Asignaciones de Ejecución de Test	29
Tabla 2.6.34 Historia de usuario Ejecución de Test múltiple.....	30
Tabla 2.6.35 Historia de usuario Ejecución de Test múltiple.....	30
Tabla 2.7.1 Caso de uso crear proyecto.....	32
Tabla 2.7.2 Caso de uso crear tarea de proyecto	32
Tabla 2.7.3 Caso de uso Crear tareas de proyecto simultáneamente	33
Tabla 2.7.4 Caso de Uso crear una hoja de tarifa de gasto	33
Tabla 2.7.5 Caso de Uso actualizar los campos planned effort [rol] de una tarea de proyecto	34
Tabla 2.7.6 Caso de Uso crear una Time Card	34
Tabla 2.7.7 Caso de uso Aprobación de una Time Card	35
Tabla 2.7.8 Caso de uso rechazo de una Time Card	36
Tabla 2.7.9 Caso de uso creación de un Plan de Recursos desde tarea de proyecto.....	36

Tabla 2.7.10 Caso de uso creación de las asignaciones de recursos en un plan de recursos asociado a una tarea de proyecto	37
Tabla 2.7.11 Caso de uso cancelar un plan de recursos asociado a una tarea de proyectos	38
Tabla 2.7.12 Caso de uso completar un plan de recursos asociado a una tarea de proyectos	38
Tabla 2.7.13 Caso de uso creación de un Plan de Recursos desde demanda	39
Tabla 2.7.14 Caso de uso Creación de un Plan de Recursos desde demanda.....	40
Tabla 2.7.15 Creación de un Conjunto de Test.....	42
Tabla 2.7.16 Caso de uso Creación de un Test y Versión de Test	42
Tabla 2.7.17 Caso de uso Creación de un Paso	43
Tabla 2.7.18 Caso de uso Asignar Test a Conjunto de Test	43
Tabla 2.7.19 Caso de uso Calcular los códigos de Test de un Conjunto de Test	44
Tabla 2.7.20 Caso de uso Preparar una versión de test para ejecutar.....	44
Tabla 2.7.21 Caso de uso Ejecutar un test.....	45
Tabla 2.7.22 Caso de uso Crear un plan de test	45
Tabla 2.7.23 Caso de uso Crear un ciclo de test	46
Tabla 2.7.24 Caso de uso Crear un software de ejecución de test	46
Tabla 2.7.25 Caso de uso Asignar Conjuntos de Test a un Software de ejecución de Test	47
Tabla 2.7.26 Caso de uso Asignar Tests a un Software de ejecución de Test	48
Tabla 2.7.27 Caso de uso Ejecutar test de manera simultanea.....	49
Tabla 2.7.28 Caso de uso Continuar ejecutando un test pausado	49
Tabla 2.7.29 Caso de uso Crear Entorno de Test.....	50
Tabla 4.2.1 Tipos de Campo.....	68
Tabla 4.2.2 Tipos de UI Action	69
Tabla 5.2.1 Diagrama de clases de Test Management	81
Tabla 5.2.2 Relación de tablas Test Run Step Results	89
Tabla 5.2.3 Formulario de Test Run Step Results	89
Tabla 5.2.4 Prefijos Database View Test Exports	90
Tabla 5.2.5 Formulario de Test Exports.....	90
Tabla 5.2.6 Tipos de Reports utilizados	93
Tabla 6.1.1Funcionamiento de las Business Rules	97
Tabla 6.2.1 Campo type en los Client Script	99
Tabla 7.1.1 - Usuarios creados.....	103
Tabla 7.1.2 Registros Resource Role.....	104
Tabla 7.3.1 Registros de Compañías.....	117
Tabla 10.1.1 Lógica de campos de proyecto	136
Tabla 10.1.2 Formulario de proyecto	136
Tabla 10.1.3 Pestaña Project Information de Proyecto	137
Tabla 10.1.4 Pestaña Cost Price de Proyecto	137
Tabla 10.1.5 Vista de tabla de Proyecto	137
Tabla 10.1.6 Lógica de campos de tarea de proyecto	138
Tabla 10.1.7 Formulario de Tarea de proyecto	138
Tabla 10.1.8 Pestaña Role Cost de Tarea de Proyecto	138
Tabla 10.1.9 Pestaña Role Effort de Tara de Proyecto	139
Tabla 10.1.10 Vista de tabla de Tarea de Proyecto	139
Tabla 10.1.11 Vista de tabla de Tarea de Proyecto en la RL de Proyecto	139
Tabla 10.1.12 Lógica de campos de Time Cards	139

Tabla 10.1.13 Formulario general Time Card	140
Tabla 10.1.14 Vista de tabla de Time Card	140
Tabla 10.1.15 Lógica de campos de Planes de Recursos	141
Tabla 10.1.16 Formulario de Plan de Recursos.....	141
Tabla 10.1.17 Pestaña Cost en Plan de Recursos.....	141
Tabla 10.1.18 Vista de tabla de Resource Plan	141
Tabla 10.1.19 Vista de tabla de Plan de Recursos en RL de Proyecto	141
Tabla 10.1.20 Vista de tabla de Plan de Recursos en RL de Tarea de Proyecto	141
Tabla 10.1.21 Lógica de campos de Asignación de Recursos	142
Tabla 10.1.22 Formulario de Asignación de Recursos	142
Tabla 10.1.23 Vista de tabla de Asignación de Recursos.....	142
Tabla 10.1.24 Vista de tabla de Asignación de Recursos en la RL Plan de recursos.....	142
Tabla 10.1.25 Lógica de campos de Hojas de Tarifas de Gasto	142
Tabla 10.1.26 Formulario Hoja de Tarifa de Gasto	143
Tabla 10.1.27 Pestaña de General en Hoja de Tarifa de Gastos.....	143
Tabla 10.1.28 Vista de tabla de Hoja de Tarifa de Proyectos	143
Tabla 10.2.1 Lógica de campos de Conjunto de Test.....	149
Tabla 10.2.2 Formulario de Conjunto de Test	149
Tabla 10.2.3 Vista de tabla de Conjunto de Test	149
Tabla 10.2.4 Lógica de campos de Conjunto de Test Test	149
Tabla 10.2.5 Formulario de Conjunto de Test Test.....	149
Tabla 10.2.6 Vista de tabla de Conjunto de Test Test	149
Tabla 10.2.7 Vista de tabla de Conjunto de Test Test en RL Versiones de Test	149
Tabla 10.2.8 Lógica de campos de Test	150
Tabla 10.2.9 Formulario de Test	150
Tabla 10.2.10 Vista de tabla de Test	150
Tabla 10.2.11 Vista de tabla de Test en RL de Conjunto de Test.....	150
Tabla 10.2.12 Lógica de campos de Versión de Test	150
Tabla 10.2.13 Formulario de Versión de Test	151
Tabla 10.2.14 Vista de tabla de Versión de Test.....	151
Tabla 10.2.15 Vista de tabla de Versión de Test en RL de Test	151
Tabla 10.2.16 Lógica de campos de Paso.....	151
Tabla 10.2.17 Formulario de Paso	151
Tabla 10.2.18 Vista de tabla de Paso	151
Tabla 10.2.19 Vista de tabla de Paso en RL de Versión de Test	151
Tabla 10.2.20 Lógica de campos de Resultado de Test	151
Tabla 10.2.21 Formulario de Resultado de Test	152
Tabla 10.2.22 Vista de tabla de Resultado de Test	152
Tabla 10.2.23 Vista de tabla de Resultado de Test en RL de Versión de Test	152
Tabla 10.2.24 Vista de tabla de Resultado de Test en RL de Test Run	152
Tabla 10.2.25 Lógica de campos de Test Run	153
Tabla 10.2.26 Formulario de Test Run	153
Tabla 10.2.27 Pestaña Results en Test Run	153
Tabla 10.2.28 Vista de tabla de Test Run.....	153
Tabla 10.2.29 Lógica de campos de Entorno de Test	154
Tabla 10.2.30 Formulario de Entorno de Test	154

Tabla 10.2.31 Vista de tabla de Entorno de Test.....	154
Tabla 10.2.32 Lógica de campos de Plan de Test	154
Tabla 10.2.33 Formulario de Plan de Test	155
Tabla 10.2.34 Pestaña Progress en Plan de Test	155
Tabla 10.2.35 Vista de tabla de Plan de Test.....	155
Tabla 10.2.36 Lógica de campos de Ciclo de Test	155
Tabla 10.2.37 Formulario de Ciclo de Test	155
Tabla 10.2.38 Pestaña Progress en Ciclo de Test	156
Tabla 10.2.39 Vista de tabla de Ciclo de Test	156
Tabla 10.2.40 Vista de tabla de Ciclo de Test en RL de Plan Test.....	156
Tabla 10.2.41 Lógica de campos de Software de Ejecución de Test	156
Tabla 10.2.42 Formulario de Software de Ejecución de Test	156
Tabla 10.2.43 Vista de tabla de Software de Ejecución de Test.....	156
Tabla 10.2.44 Vista alternativa de tabla de Software de Ejecución de Test	157
Tabla 10.2.45 Vista de tabla de Software de Ejecución de Test en RL de Ciclo de Test.....	157
Tabla 10.2.46 Lógica de campos de Asignación de Ejecución de Test	157
Tabla 10.2.47 Formulario Asignación de Ejecución de Test	157
Tabla 10.2.48 Vista de tabla de Asignación de Ejecución de Test.....	157
Tabla 10.2.49 Vista de tabla de Asignación de Ejecución de Test en RL de Software de Ejecución de Test	157

1 Introducción

1.1 Motivación

Tras 8 meses trabajando en la empresa *SilverStorm* pensé que la herramienta *ServiceNow*, que se utiliza en la empresa, era una buena oportunidad para desarrollar el trabajo de fin de grado, ya que esta presenta muchas opciones de desarrollo. Además, me permitiría desarrollar mayores habilidades con una herramienta que utilizo diariamente.

Una vez decidí realizar el proyecto en la empresa, me reuní con los encargados de desarrollo interno de la empresa para ver que necesidades tenía la empresa que pudieran ser incluidas en un Trabajo de Fin de Grado y que además me motivarán personalmente para realizar el proyecto.

Las razones por las que me decante por el desarrollo de estas aplicaciones, es porque la empresa en la que me encuentro y para la que desarrollo este proyecto está ahora mismo en crecimiento, y especialmente durante el año 2020 hubo muchas incorporaciones y aumento el número de trabajadores en plantilla. Cuando una empresa crece, necesita nuevos métodos para analizar sus propios datos y gestionarse internamente, además de ir mejorando sus productos para poder llegar un mayor público.

La parte de Gestión de Proyectos permitirá a la empresa gestionar de una manera más eficiente sus recursos y controlar sus costes. El aumento del personal va ligado a un mayor número de proyectos en los que la empresa trabaja, y una mayor dificultad para asignar correctamente los recursos que con el sistema que se utiliza ahora mismo, y creo que será muy beneficioso para la empresa.

El desarrollo de *Test Management* viene por la creación de un nuevo departamento en la empresa, el departamento de QA (*Quality Assurance*) que se creó a principios de 2020 y en el que actualmente hay 2 personas trabajando. El objetivo de este departamento es dotar de una mayor calidad a los desarrollos que realizamos a través del *Testeo*. Este departamento trabajo el año pasado en exclusiva con el cliente NTT (*Nippon Telegraph and Telephone Corporation*), que fue el proyecto en el que yo estaba trabajando, y toda esta labor la realizamos el Excel. Tras haber trabajado en ello, creo que el desarrollo de esta aplicación nos permite un control mucho mayor del proceso de *Testeo*, pudiendo agrupar *Testeos* y pruebas en el propio *ServiceNow* y dará muchas facilidades a mis compañeras de este departamento para poder realizar un mejor trabajo y que este departamento siga creciendo como una parte importante de la empresa.

1.2 Objetivos

Los objetivos del Trabajo de Fin de Grado se dividen en dos partes, por un lado los objetivos centrados en los desarrollos realizados y lo que sé que quiere conseguir con los mismos, y por otro los objetivos personales que yo tengo haciendo este desarrollo.

1.2.1 Objetivos Gestión de proyectos

En el apartado de gestión de proyectos los objetivos son:

- Desarrollar una forma de gestión de los recursos de la empresa basada en el rol de cada usuario.
- Gestionar los recursos a través de un Proyecto a través de sus Tareas de Proyecto para trabajar en pequeñas tareas concretas.
- Sacar mayor rendimiento a la aplicación ITBM (*Business Management Informatic Technology*) de *ServiceNow*.
- Mejorar el método para medir la rentabilidad de los Proyecto.

1.2.2 Objetivos de *Test Management*

En el apartado de *Test Management* los objetivos son:

- Incluir la aplicación de *Test Management* en la instancia de *ServiceNow* perteneciente a *SilverStorm*.
- Proporcionar al departamento de QA de la empresa una forma de trabajo integrada en *ServiceNow*.
- Mejorar los planes de pruebas desarrollados para ofrecer mejores servicios a los clientes.

1.2.3 Objetivos personales

- Ampliar mis conocimientos sobre la herramienta *ServiceNow*.
- Mejorar mis habilidades en desarrollo JavaScript.
- Realizar un desarrollo en el cual sea el responsable principal del proyecto.

1.3 Contexto

La empresa *SilverStorm* con la que se realiza este Trabajo de Fin de Grado es una compañía española que tiene su sede principal en Valladolid, y está especializada en desarrollo digital, orientándose a ofrecer un servicio ágil y enfocado en la combinación de experiencias personalizadas de Usuarios, Negocios y Tecnología. Proveemos de Visión, Dirección y Experiencia a las organizaciones para impulsar su crecimiento, garantizar la continuidad de su negocio y mejorar la productividad de los empleados [1].

Entre las diferentes herramientas que *SilverStorm* utiliza para proporcionar servicios a sus clientes destaca *ServiceNow*, que es una compañía que se fundó en el año 2004 en Estados Unidos. En palabras de la propia empresa, proporciona flujos de trabajo digitales multiempresariales que conecten las personas, las funciones y los sistemas para acelerar la innovación, aumentar la agilidad y mejorar la productividad [2]. Para ello proporciona servicios en la nube, a través de diferentes servidores que permiten realizar otros servicios. Ofrecen servicios de ITSM (*Informatic Technology Service Management*) y de *Test Management*, que son los principales que estarán implicados en este proyecto, además de muchos otros que pueden abarcar cualquier área de las empresas actuales.

La plataforma de *ServiceNow* provee de una aplicación basada en Plataforma como servicio, mediante un servicio en la nube que proporciona para desarrollar, ejecutar y controlar aplicaciones. Para realizar estas tareas *ServiceNow* se basa en el uso de instancias, que pertenecen única y exclusivamente a un cliente. Cada cliente puede tener varias instancias, basándose en una arquitectura multi instancia [3].

Para el desarrollo dentro de la plataforma, por un lado, se realiza mediante elementos que no requieren desarrollo de código, ya que se desarrollan mediante interfaces predefinidas en las que se indican los requisitos para que determinadas acciones se produzcan.

En la parte de scripting, el lenguaje utilizado mayoritariamente es *JavaScript*, en el cual están definidas de caja distintos objetos y funciones para facilitar el desarrollo mediante este lenguaje. También se utilizará en determinadas entidades de la herramienta HTML, uniendo a la tecnología *Jelly* o *Angular* para programar algunos objetos.

ServiceNow proporciona una vista técnica basada principalmente en listas y formularios, y una vista de usuario (*Service Portal*) basado en widgets orientada a dar una experiencia de usuario más satisfactoria a los perfiles menos técnicos para utilizar la herramienta, que es principalmente la parte de la herramienta en la que se trabaja con *Angular*. Además, también proporciona vistas para trabajar tanto desde un ordenador como desde una *Tablet* o un *Smartphone*.

Para permitir el uso la aplicación se basa en un sistema de permisos, en el cual hay distintos usuarios que se asignan a distintos grupos de trabajo, y para los que se definen los roles que les permitirán realizar unas acciones determinadas, ya que el sistema *restringe* el acceso a sus tablas y registros principalmente a través de estos roles.

La parte del desarrollo de ITSM va orientado principalmente al trabajo que desarrollan en la empresa de *SilverStorm* las personas con el rol de *Program Delivery Management*, que son los encargados de la gestión de los proyectos, de controlar sus costes y de asignar los recursos disponibles para que estos se realicen correctamente.

En la parte de *Test Management*, se orienta a que todos los usuarios puedan llevar un control de las pruebas que se realizan en los desarrollos que la empresa hace para otras empresas, con un especial hincapié en el control que llevan sobre estos los departamentos de QA y los TAM.

1.4 Procedimiento

Para desarrollar este Trabajo de Fin de Grado lo primero ha sido realizar reuniones con distintos empleados de la empresa *SilverStorm* (en este caso los clientes) que aportaron los casos de uso y las necesidades que había tanto en la parte de Gestión de Proyectos como en la de *Test Management*. Tras ello, se plantearon los objetivos a cumplir, la lista inicial de requisitos y la definición de los procesos.

Se ha trabajado con un método *Agile* basado en *Scrum*, en el que se han realizado reuniones periódicas con el cliente para presentar los avances, hacer retrospectiva y analizar tanto los siguientes pasos como los errores producidos. A medida que se ha avanzado han ido apareciendo requisitos nuevos que se han desarrollado.

1.5 Tecnologías utilizadas

Las tecnologías que se han utilizado en la realización de este Trabajo de Fin de Grado han sido:

- *ServiceNow*: Herramienta de software en la que se ha realizado la implementación del proyecto. En ella se combinan elementos de caja que nos proporciona la propia herramienta con elementos de creación propia.
- *Microsoft Word*: herramienta en la que se desarrolla la memoria descriptiva del documento.
- *Microsoft Excel*: herramienta en la que se han almacenado datos para pruebas dentro de la herramienta.
- *Astah UML*: se ha utilizado para realizar diversos diagramas de la memoria, como el diagrama de clases.
- *Draw.io*: herramienta en la que se ha creado el diagrama temporal de la evolución del Trabajo de Fin de Grado y los ciclos de vida del proceso.
- Herramienta Recortes: herramienta proporcionada por Windows para realizar capturas de pantalla.

1.6 Estructura del documento

El documento se divide en distintas secciones en las que se especifica y se desarrolla todo lo trabajado durante el tiempo en el que se ha llevado a cabo el trabajo. Esta parte se divide en 6 capítulos distintos:

- Introducción: ofrece una visión global de las razones por las que se ha realizado este Trabajo de Fin de Grado y el marco en el que se han desarrollado las aplicaciones.

- Metodología de trabajo: desarrolla del sistema que se ha seleccionado para el Trabajo de Fin de Grado y como ha ido el proyecto en el tiempo.
- Ciclos de vida de las entidades: aparecen los ciclos de vida de los diferentes registros que hay en el proceso.
- Elementos técnicos de la aplicación: definición de las distintas entidades desarrolladas, sus contenidos y como se relacionan entre ellas.
- Implementación del proyecto en la instancia: como se han creado los campos y otras funcionalidades de la instancia.
- Casos de prueba: datos que se han utilizado para realizar las pruebas que se han llevado a cabo en el Trabajo de Fin de Grado.

Después se añaden las conclusiones sobre todo el trabajo realizado.

2 Metodología de trabajo

2.1 Scrum

En este Trabajo de Fin de grado se ha utilizado metodología *Scrum*. *Scrum* es un proceso de trabajo basado en entregas parciales y regulares del producto, en el que se priorizan las tareas según el beneficio que aportan al proyecto final, y que se utiliza en proyectos donde los requisitos son cambiantes o poco definidos, donde la innovación, la competitividad, la flexibilidad y la productividad son fundamentales [4].

En scrum se utilizan ciclos temporales cortos y con una duración fija, que suele ser entre 2 y 3 semanas. Por cada iteración se realiza una entrega al cliente que conlleve un incremento del producto final. En este Trabajo de Fin de Grado el ciclo, también conocido como sprint, tendrá una duración de 3 semanas [5].

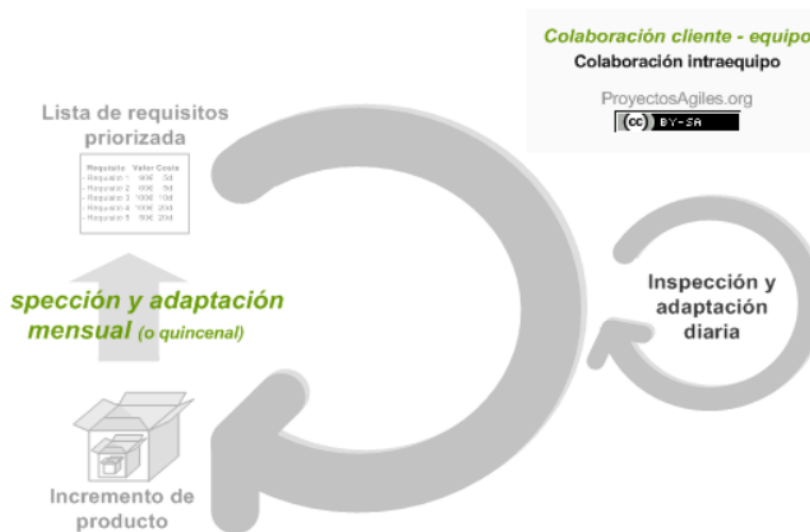


Figura 2.1.1 Metodología Scrum

Los beneficios de haber realizado el proyecto en *SCRUM* son la gestión de resultados de una forma más tangible, ya que los resultados se han mostrado al cliente periódicamente, flexibilidad y adaptación respecto a las necesidades de clientes, mitigación de los riesgos del cliente, así como una productividad y calidad y mayor facilidad para el entendimiento entre cliente y equipo de desarrollo (en este caso solo un desarrollador).

Habitualmente en un método de trabajo *SCRUM* encontraríamos 3 tipos de roles, el cliente que es la persona o empresa que requiere un nuevo servicio, el *scrum master*, que es el máximo responsable del proyecto que se realiza, y el desarrollador, que se encarga de implementar los requisitos a través de los documentos técnicos y de las historias de usuario. En el caso de este Trabajo de Fin de Grado, el papel tanto de cliente como de *scrum master* lo ha realizado la empresa *SilverStorm*, y el papel de desarrollador lo ha realizado el autor del Trabajo de Fin de Grado.

2.2 Comunicado de requisitos iniciales

Se realizaron dos reuniones iniciales con el objetivo de transmitir los primeros requisitos que se debían comenzar a implementar y que son los primeros sobre los que se realizaría una reunión de revisión. A partir de ese punto se crearían las primeras historias de usuario para trabajar.

2.3 Desarrollos de requisitos de cliente

A medida que se fueron desarrollando las reuniones, se fueron creando más requisitos que fueron reflejados en las distintas historias de usuario que se han ido creando. Podemos dividir los requisitos en dos partes, los que corresponde a la parte de Gestión de Proyectos y los que corresponden a la parte de *Test Management*. Antes de identificar los requisitos de cada área, se explican unos breves conceptos sobre ambas partes del proyecto.

2.3.1 Requisitos de Gestión de Proyectos

Antes de desarrollar los requisitos que tiene la Gestión de Proyectos es importante entender algunos conceptos propios de la empresa con los que se trabajan. En los proyectos con clientes externos trabaja el departamento de Delivery. En este departamento hay 4 tipos de roles que dividen a los trabajadores: DC (*Deployment Consultant*), SDC (*Senior Deployment Consultant*), PDM (*Program Delivery Manager*) y BPC (*Business Program Consultant*). Personas de estos 4 roles pueden estar asignadas a las distintas Tareas.

Los recursos para la Gestión de los Proyectos se generan en los Planes de Recursos, que pueden ser asignados a través de usuarios concretos, de grupos concretos o de roles. Además se pueden asociar a una Demanda o a una Tarea de Proyecto. Además, para trabajar con Tareas de Proyectos y los registros asociados a ellas, es necesario que existan Hojas de Tarifas de Gasto que establezcan el precio de la hora trabajada para esas Tareas.

Para el proceso de Gestión de Proyectos, los requisitos se fueron solicitando y transmitiendo al programador en el siguiente orden:

- Requisitos iniciales:
 - REQ001: disponer de distintos campos en los que mostrar cuál es el esfuerzo previsto para una tarea de proyecto para los roles DC, SDC, PDM y BPC.
 - REQ002: disponer de distintos campos en los que mostrar cuál es el coste previsto para una tarea de proyecto para los roles DC, SDC, PDM y BPC.
 - REQ003: disponer de distintos campos en los que mostrar cuál es el esfuerzo actual para una tarea de proyecto para los roles DC, SDC, PDM y BPC.
 - REQ004: disponer de distintos campos en los que mostrar cuál es el coste actual para una tarea de proyecto para los roles DC, SDC, PDM y BPC.
 - REQ005: cuando la *Task* asociada a un Plan de Recursos es de tipo Demanda, el campo *resource type* del plan de recursos debe valer por defecto Role y no poder editarse.
 - REQ006: cuando la *Task* asociada a un Plan de Recursos es de tipo Demanda, los valores del campo rol serán QA y PDM.
 - REQ007: cuando la *Task* asociada a un Plan de Recursos es de tipo Demanda, el campo members se ocultará.
 - REQ008: cuando la *Task* asociada a un Plan de Recursos es de tipo Demanda, aparecerá un campo *project task*, que podrá asociarse a una Tarea de Proyecto.
 - REQ009: cuando la *Task* asociada a un Plan de Recursos es de tipo Demanda, el campo *resource rate* será obligatorio y no se autocompletará.

- Requisitos punto de control 2:
 - REQ001: cuando la *Task* asociada a un Plan de recursos es de tipo Demanda, se mostrará la *Related list Requested Allocations*.
 - REQ002: cuando se crea un registro *Time Card* se debe mostrar el campo *resource plan* que será autocompletado si existe algún plan de recurso con el mismo usuario y Tarea de Proyecto de la *Time Card*. Además, la fecha de la *Time Card* debe estar entre las fechas de los campos *start date* y *end date* del plan de recursos.
 - REQ003: desde el formulario de Tarea de Proyecto, se tiene que poder crear un plan de recursos asociado a la misma.
- Requisitos punto de control 3:
 - REQ001: disponer de distintos campos en los que mostrar cuál es la proyección de esfuerzo prevista para una tarea de proyecto como resultado del esfuerzo realizado hasta ahora y el que queda por realizar para los roles DC, SDC, PDM y BPC.
 - REQ002: disponer de distintos campos en los que mostrar cuál es la proyección de costes prevista para una tarea de proyecto como resultado del coste invertido hasta ahora y el que queda por invertir para los roles DC, SDC, PDM y BPC.
 - REQ003: para calcular cual es la proyección de esfuerzo de un Proyecto, se deben sumar todas las proyecciones de esfuerzo por rol de las tares de proyecto asociadas.
 - REQ004: para calcular cual es el esfuerzo previsto realizar en un Proyecto, se deben sumar todas las previsiones de esfuerzo de las tares de proyecto asociadas.
 - REQ005: para calcular cual es el esfuerzo actualmente realizado en un proyecto, se deben sumar todos los campos de esfuerzo actualmente realizado de las tareas de proyecto asociadas.
 - REQ006: para calcular cual es el coste previsto realizar en un Proyecto, se deben sumar todos los campos de coste previsto de las tareas de proyecto asociadas.
 - REQ007: para calcular cual es el coste invertido actualmente en un Proyecto, se deben sumar todos los campos de coste invertido actualmente de las tareas de proyecto asociadas.
- Requisitos punto de control 4:
 - REQ001: cuando un Plan de Recursos se cierra, se deben recalcular cual es la proyección de esfuerzo de una Tarea de Proyecto para los roles DC, SDC, PDM y BPC.
 - REQ002: cuando un Plan de Recursos se cierra, se deben recalcular cual es la proyección de coste de una tarea de proyecto para los roles DC, SDC, PDM y BPC.
- Requisitos del punto de control 5:
 - REQ001: los campos que contienen el coste previsto de cada tarea de proyecto por rol deben ser no editables.
 - REQ002: los campos que contienen el coste previsto de cada tarea de proyecto por rol se calculan multiplicando el campo que contiene el esfuerzo previsto para esa tarea de proyecto.
 - REQ003: al menos un campo que contiene el esfuerzo previsto para cada rol tiene que ser mayor que 0 en cada tarea de proyecto.

- REQ004: el campo que contiene el esfuerzo general previsto para una tarea de proyecto debe ser no editable.
- REQ005: el campo que contiene el esfuerzo general previsto para una tarea de proyecto se calculara sumando el esfuerzo previsto para cada rol en esa tarea de proyecto.
- REQ006: el campo que contiene el coste general previsto para una tarea de proyecto debe ser no editable.
- REQ007: el campo que contiene el coste general previsto para una tarea de proyecto se calculara sumando el coste previsto para cada rol en esa tarea de proyecto.
- REQ008: el campo que contiene el esfuerzo general actualmente realizado para una tarea de proyecto debe ser no editable.
- REQ009: el campo que contiene el esfuerzo general actualmente realizado para una tarea de proyecto se calculara sumando el esfuerzo actualmente realizado para cada rol en esa tarea de proyecto.
- REQ010: el campo que contiene el coste general actualmente invertido para una tarea de proyecto debe ser no editable.
- REQ011: el campo que contiene el coste general actualmente invertido para una tarea de proyecto se calculara sumando el coste actualmente invertido para cada rol en esa tarea de proyecto.

2.3.2 Requisitos de *Test Management*

Antes de desarrollar los requisitos que tiene Test Management es importante entender algunos conceptos propios de la empresa con los que se trabajan. Test Management es una aplicación que ServiceNow proporciona de caja, y en el Trabajo de Fin de Grado se ha adaptado a las necesidades de SilverStorm. Los Entornos de Test se refieren a las instancias reales que tienen los distintos clientes en ServiceNow. Un *Trasnform Map* permite importar registros de fuera de la instancia a ella desde distintos formatos como Excel, lo que ahorra trabajo de creación de registros. Las Database Views no son registros ni tablas como tal, sino que asocian registros de varias tablas que tienen algún tipo de relación y los muestran en un mismo formulario.

Para el proceso de *Test Management*, los requisitos se fueron solicitando y transmitiendo al programador en el siguiente orden:

- Requisitos iniciales:
 - REQ001: creación de campos, *related list* y modificación de formularios y campos que se muestran en las tablas (1ª fase). Las tablas afectadas son *Test*, *Pasos*, *Versión de Test*, y *Asignación de Ejecución de Test*.
 - REQ002: modificación de datos del proceso de carga del mapa de transformación *SS – Test Importer 2* y creación de campos en tabla auxiliar denominada *SS - Test Importer 2*. (1ª fase).
 - REQ003: creación de la *Database View Test Step Results*.
 - REQ004: creación del *Dashboard Test Management*.
 - REQ005: creación y visualización de módulos y menús *Test Dashboard* y *Test Results*.
- Requisitos punto de control 2:

- REQ001: creación de campos, *related list* y modificación de formularios y campos que se muestran en las tablas (2ª fase). La tabla afectada es Conjunto de *Test*.
- REQ002: modificación de datos del proceso de carga del mapa de transformación *SS – Test Importer 2* y creación de campos en tabla auxiliar denominada *SS - Test Importer 2*. (2ª fase).
- REQ003: posibilidad de exportar los resultados de los *Test*.
- Requisitos punto de control 3:
 - REQ001: creación de campos, *related list* y modificación de formularios y campos que se muestran en las tablas (3ª fase). La tabla afectada son Asignación de Ejecución de *Test*, Software de Ejecución de *Test*, *Test Run*, Resultados de *Test* y Conjunto de *Test*.
 - REQ002: modificación de datos del proceso de carga del mapa de transformación *SS – Test Importer 2* y creación de campos en tabla auxiliar denominada *SS - Test Importer 2*. (3ª fase).
- Requisitos punto de control 4:
 - REQ001: creación de campos, *related list* y modificación de formularios y campos que se muestran en las tablas (4ª fase). Las tablas afectadas son Conjunto de *Test* y *Test Result*.
- Requisitos punto de control 5:
 - REQ001: creación de campos, *related list* y modificación de formularios y campos que se muestran en las tablas (5ª fase). La tabla afectada es Conjunto de *Test*.
 - REQ002: modificación de las reglas de negocia que asocian la tabla Conjunto de *Test* con la tabla *Test*.
 - REQ003: modificación de la funcionalidad de la *UI (User Interface) Action Create New Version* de la tabla Versión de *Test*.
 - REQ004: creación de campos, *related list* y modificación de formularios y campos que se muestran en las tablas (5ª fase). La tabla afectada es *Software de Ejecución de Test*.
 - REQ005: modificación de las reglas de negocia que asocian la tabla *Software de Ejecución de Test* con la tabla *Asignación de Ejecución de Test*.
 - REQ006: creación de campos, *related list* y modificación de formularios y campos que se muestran en las tablas (5ª fase). La tabla afectada es *Stories*.
 - REQ007: creación y visualización del módulo *Stories Execution Suites*.
 - REQ008: modificación del orden en el que se muestran los *Test* a realizar en la *UI Macro* activable con los botones *Run*.
- Requisitos punto de control 6:
 - REQ001: creación de una *UI Action* en la tabla *Versión de Test* que permita devolver un registro al estado *Draft*.
 - REQ002: creación de campos, *related list* y modificación de formularios y campos que se muestran en las tablas (6ª fase). La tabla afectada es Entorno.
 - REQ003: creación y visualización del módulo *Environments*.
 - REQ004: creación de un sistema para autogenerar los valores del campo *code* de la tabla *Test*.
 - REQ005: crear una nueva *Database View* que relacione un registro Paso con el *Resultado de Test* y el *Test*.

A medida que se generan los requisitos se buscan las formas más adecuadas de implementar cada uno de ellos.

2.4 Desarrollos de elementos técnicos

Esta fase se realizó teniendo en cuenta todos los requisitos que surgieron a lo largo de los puntos de control de desarrollo. Para poder realizarla el cliente (*SilverStorm*) proporcionó plantillas de dos estilos. Por un lado, documentos de tipo *Word* en los que se especificaron los distintos requisitos propios del desarrollo a realizar y por otro lado entidades *Stories* que especificaron los plazos y el encargado de supervisar esa parte. En esta entidad también especificaron tanto la descripción técnica de los requisitos como los criterios de aceptación que tenía que cumplir el desarrollo.

2.5 Roles

En este proyecto se trabaja con distintos roles. Para la parte de Gestión de Proyectos tenemos el rol admin, que pueden realizar todas las acciones que conlleva la aplicación. Además, existe el usuario sin roles, que puede solo crear las *Time Cards* que luego el *admin* aprobará o rechazará.

En la parte de *Test Managemnet*, nos encontramos con el rol test manager, que puede realizar todas las acciones que se aplican en esta parte de la aplicación

2.6 Desarrollo de historias de usuario

Tras conocer los requisitos y desarrollar los elementos técnicos, se comenzó con los desarrollos de historias de usuario. Primero empezó el desarrollo de *Test Management*, iniciando la customización de las entidades que ofrece *ServiceNow* para adaptarlas a las necesidades del cliente.

Más tarde continuó el desarrollo de Gestión de Proyectos, comenzando con la customización de las tareas de proyecto con los campos de esfuerzo y costes necesarios y más tarde, a medida que el proyecto avanzó, se incluyeron nuevas historias en ambas áreas de trabajo a la vez que las anteriores se iban cerrando o redefiniendo cuando es necesario.

2.6.1 Historias de usuario Gestión de Proyectos

Dentro de la gestión de proyectos se han establecido las siguientes historias de usuarios:

ID	HU01
Historia de Usuario	Crear proyecto
Descripción	Como admin, quiero crear proyectos para gestionar los distintos contratos con los clientes
Criterios de aceptación	El admin puede crear un proyecto

Tabla 2.6.1 Historia de usuario crear proyecto

ID	HU02
Historia de Usuario	Crear tarea de proyecto
Descripción	Como admin, quiero crear tareas de proyecto para dividir los proyectos en partes más pequeñas
Criterios de aceptación	El admin puede crear tareas de proyecto desde un proyecto

Tabla 2.6.2 Historia de Usuario crear tarea de proyecto

ID	HU03
Historia de Usuario	Crear Hojas de Tarifa de Gasto
Descripción	Como admin, quiero crear una hoja tarifa de gasto para saber cuánto se cobra a un cliente por cada hora de servicio de un usuario
Criterios de aceptación	El admin puede crear hojas de tarifas de gasto

Tabla 2.6.3 Historia de usuario Crear Hojas de Tarifa de Gasto

ID	HU04
Historia de Usuario	Añadir esfuerzo planeado por rol en una tarea de proyecto
Descripción	Como admin, quiero añadir el esfuerzo planeado a una tarea de proyecto dependiendo del rol para saber cuanto tiempo está previsto que invierta cada rol en un proyecto
Criterios de aceptación	El admin puede añadir esfuerzo planeado por rol en una tarea de proyecto. Cuando se añaden esfuerzos planeados en los roles, se calculan los costes planeados por rol en las tareas de proyecto. Cuando se modifican los esfuerzos y los costes planeados por rol, se calcula el esfuerzo planeado y el coste planeado general de la tarea de proyecto

Tabla 2.6.4 Historia de Usuario Añadir esfuerzo planeado por rol en una tarea de proyecto

ID	HU05
Historia de Usuario	Ver esfuerzo y coste planeado a un proyecto
Descripción	Como admin, quiero poder visualizar el esfuerzo y coste planeado en un proyecto para saber cuánto está previsto invertir en total en un proyecto
Criterios de aceptación	El admin puede ver el esfuerzo y el coste planeado en un proyecto

Tabla 2.6.5 Historia de usuario ver esfuerzo y coste planeado a un proyecto

ID	HU06
Historia de Usuario	Ver esfuerzo y coste actual en una tarea de proyecto
Descripción	Como admin, quiero poder visualizar el esfuerzo y coste actual de una tarea de proyecto para saber cuánto se ha invertido actualmente por rol y en total en una tarea proyecto
Criterios de aceptación	El admin puede ver el esfuerzo y el coste actual por [rol] en una tarea de proyecto. El admin puede ver el esfuerzo y el coste actual total en una tarea de proyecto.

Tabla 2.6.6 Historia de usuario Ver esfuerzo y coste actual por rol en el proyecto

ID	HU07
Historia de Usuario	Ver esfuerzo y coste actual de un proyecto
Descripción	Como admin, quiero poder visualizar el esfuerzo y coste actual de un proyecto para saber cuánto se ha invertido actualmente en total en un proyecto
Criterios de aceptación	El admin puede ver el esfuerzo y el coste actual de un proyecto

Tabla 2.6.7 Historia de usuario Ver esfuerzo y coste actual de un proyecto

ID	HU08
Historia de Usuario	Ver esfuerzo y coste proyectado por rol en una tarea de proyecto
Descripción	Como admin, quiero poder visualizar el esfuerzo y coste proyectado de una tarea de proyecto para saber cuánto se tiene previsto invertir teniendo en cuenta lo invertido hasta el momento por rol y en total en una tarea proyecto
Criterios de aceptación	El admin puede ver el esfuerzo y el coste proyectado por [rol] en una tarea de proyecto. El admin puede ver el esfuerzo y el coste proyectado total en una tarea de proyecto.

Tabla 2.6.8 Historia de usuario Ver esfuerzo y coste proyectado por rol en una tarea de proyecto

ID	HU09
Historia de Usuario	Ver esfuerzo y coste proyectado de un proyecto
Descripción	Como admin, quiero poder visualizar el esfuerzo y coste proyectado de un proyecto para saber cuánto se tiene previsto invertir teniendo en cuenta lo invertido hasta el momento en total en un proyecto
Criterios de aceptación	El admin puede ver el esfuerzo y el coste proyectado de un proyecto

Tabla 2.6.9 Historia de usuario Ver esfuerzo y coste proyectado de un proyecto

ID	HU10
Historia de Usuario	Crear una <i>Time Card</i>
Descripción	Como usuario, quiero que se puedan crear <i>time cards</i> para saber cuánto tiempo ha trabajado un usuario en una tarea de proyecto
Criterios de aceptación	El usuario puede crear una <i>Time Card</i> con lo trabajado durante una semana en una tarea de proyecto

Tabla 2.6.10 Historia de usuario Crear una *Time Card*

ID	HU11
Historia de Usuario	Aprobar una <i>Time Card</i>
Descripción	Como admin, quiero poder aprobar una <i>time card</i> para imputar el tiempo como invertido en la tarea de proyecto asociada
Criterios de aceptación	El admin puede aprobar <i>time cards</i> El tiempo se suma al esfuerzo y el coste actual de la tarea de proyecto asociada

Tabla 2.6.11 Historia de usuario Aprobar una *Time Card*

ID	HU12
Historia de Usuario	Rechazar una Time Card
Descripción	Como admin, quiero poder rechazar una time card para rechazar el tiempo como invertido en la tarea de proyecto asociada
Criterios de aceptación	El admin puede rechazar <i>time cards</i>

Tabla 2.6.12 Historia de usuario Rechazar una Time Card

ID	HU13
Historia de Usuario	Crear un plan de recursos
Descripción	Como admin, quiero poder crear planes de recursos asociados a una tarea de proyecto para gestionar la asignación de recursos en esta
Criterios de aceptación	El admin puede crear planes de recursos

Tabla 2.6.13 Historia de usuario crear un plan de recursos

ID	HU14
Historia de Usuario	Crear un plan de recursos
Descripción	Como admin, quiero poder crear planes de recursos asociados a una demanda para gestionar la asignación de recursos en esta
Criterios de aceptación	El admin puede crear planes de recursos

Tabla 2.6.14 Historia de usuario Crear un plan de recursos

ID	HU15
Historia de Usuario	Crear asignaciones de recursos
Descripción	Como admin, quiero poder crear asignaciones de recursos asociadas a los planes de prueba para prever las horas de trabajo a realizar semanalmente
Criterios de aceptación	El admin puede crear asignaciones de recursos asociadas a un plan de pruebas Las asignaciones de recursos modifican el esfuerzo y el coste proyectado por rol en la tarea de proyecto

Tabla 2.6.15 Historia de usuario Crear asignaciones de recursos

ID	HU16
Historia de Usuario	Cancelar un plan de recursos
Descripción	Como admin, quiero poder cancelar planes de recursos asociados a una tarea de proyecto, para marcar que ese plan de recursos no se llevará a cabo.
Criterios de aceptación	El admin puede crear planes de recursos La cancelación modifica el esfuerzo y el coste proyectado por rol en la tarea de proyecto.

Tabla 2.6.16 Historia de usuario Cancelar un plan de recursos

ID	HU17
Historia de Usuario	Completar un plan de recursos
Descripción	Como admin, quiero poder completar planes de recursos asociados a una tarea de proyecto, para marcar que ya no se está trabajando con esos planes
Criterios de aceptación	El admin puede completar planes de recursos Completar modifica el esfuerzo y el coste proyectado por rol en la tarea de proyecto.

Tabla 2.6.17 Historia de usuario Completar un plan de recursos

ID	HU18
Historia de Usuario	Visualizar coste habitual de un proyecto
Descripción	Como admin, quiero poder visualizar el coste planeado, actual y proyectado de un proyecto con las tarifas normales para poder compararlo con el coste debido a la tarifa asignada a ese proyecto o a sus tareas de proyecto.
Criterios de aceptación	El admin puede visualizar los costes con la tarifa habitual del proyecto

Tabla 2.6.18 Historia de usuario Visualizar coste habitual de un proyecto

2.6.2 Historias de usuario *Test Management*

ID	HU01
Historia de Usuario	Crear un Conjunto de Test
Descripción	Como test manager, quiero agrupar los test para poder distinguirlos según el proceso y la compañía a la que pertenecen
Criterios de aceptación	El test manager puede crear conjuntos de Test

Tabla 2.6.19 Historia de Usuario Crear un Conjunto de Test

ID	HU02
Historia de Usuario	Crear un Test
Descripción	Como test manager, quiero crear un Test para poder llevar a cabo el testeo de una implementación desde el mismo
Criterios de aceptación	El test manager puede crear Test

Tabla 2.6.20 Historia de usuario Crear un Test

ID	HU03
Historia de Usuario	Crear una Versión de Test
Descripción	Como test manager, quiero crear versiones de test asociadas a los test para poder tener distintas versiones de ejecución si un proceso va evolucionando
Criterios de aceptación	El test manager puede crear Versiones de Test

Tabla 2.6.21 Historia de Usuario Crear una Versión de Test

ID	HU04
Historia de Usuario	Crear un Paso
Descripción	Como test manager, quiero crear pasos asociados a las versiones de test para indicar los pasos a seguir en una ejecución
Criterios de aceptación	El test manager puede crear pasos

Tabla 2.6.22 Historia de usuario Crear un Paso

ID	HU05
Historia de Usuario	Crear un Entorno de Test
Descripción	Como test manager, quiero crear un entorno de test para indicar en que instancia se ejecutan los testeos
Criterios de aceptación	El test manager puede crear un entorno de test

Tabla 2.6.23 Historia de Usuario Crear un Entorno de Test

ID	HU06
Historia de Usuario	Asignar Test a los Conjuntos de Test
Descripción	Como test manager, quiero poder asignar test a los conjuntos de test para agrupar los que pertenecen a procesos similares
Criterios de aceptación	El test manager puede asignar Test a los Conjuntos de Test

Tabla 2.6.24 Historia de usuario Asignar Test a los Conjuntos de Test

ID	HU07
Historia de Usuario	Actualizar los códigos de los Test de un conjunto de Test en bloque
Descripción	Como test manager, quiero rellenar todos los códigos de los Test de un conjunto de test simultáneamente para ahorrar tiempo y evitar errores
Criterios de aceptación	El test manager puede generar todos los códigos de los test de un conjunto de test simultáneamente

Tabla 2.6.25 Historia de usuario Actualizar los códigos de los Test de un conjunto de Test en bloque

ID	HU08
Historia de Usuario	Actualizar los códigos de los pasos automáticamente
Descripción	Como test manager, quiero que cuando mueva un paso en la lista de pasos de una versión de test, el código se actualice según su posición para poder visualizarlos correctamente cuando voy a exportar
Criterios de aceptación	El código del paso se actualiza correctamente cuando este se mueve

Tabla 2.6.26 Historia de usuario Actualizar los códigos de los pasos automáticamente

ID	HU09
Historia de Usuario	Ejecutar un Test individualmente
Descripción	Como test manager, quiero ejecutar un test de forma individual para identificar si el proceso se ejecuta correctamente
Criterios de aceptación	El test manager puede ejecutar un test individualmente

Tabla 2.6.27 Historia de usuario Ejecutar un Test individualmente

ID	HU10
Historia de Usuario	Pausar la ejecución de un Test
Descripción	Como test manager, quiero pausar la ejecución de un test para realizar comprobaciones sobre el mismo antes de terminarlo
Criterios de aceptación	El test manager puede pausar la ejecución de un test

Tabla 2.6.28 Historia de usuario Pausar la ejecución de un Test

ID	HU11
Historia de Usuario	Crear un Plan de Test
Descripción	Como test manager, quiero crear un plan de test para especificar un conjunto de ciclos de test que se van a ejecutar en un periodo de tiempo determinado
Criterios de aceptación	El test manager puede crear un plan de test

Tabla 2.6.29 Historia de usuario Crear un Plan de Test

ID	HU12
Historia de Usuario	Crear un Ciclo de Test
Descripción	Como test manager, quiero crear un ciclo de test para especificar un conjunto de softwares de ejecución de test que se van a ejecutar en un periodo de tiempo determinado
Criterios de aceptación	El test manager puede crear un ciclo de test

Tabla 2.6.30 Historia de usuario Crear un Ciclo de Test

ID	HU13
Historia de Usuario	Crear un Software de Ejecución de Test
Descripción	Como test manager, quiero crear un Software de Ejecución de Test para poder agrupar Test y ejecutarlos a la vez
Criterios de aceptación	El test manager puede crear Software de Ejecución de Test

Tabla 2.6.31 Historia de usuario Crear un Software de Ejecución de Test

ID	HU14
Historia de Usuario	Asignar un Software de Ejecución de Test a una story
Descripción	Como test manager, quiero asociar un software de ejecución de test a una story para indicar que los test que lo componen pertenecen a esa story
Criterios de aceptación	El test manager puede asociar un Software de Ejecución de Test a una story

Tabla 2.6.32 Historia de usuario Asignar un Software de Ejecución de Test a una story

ID	HU15
Historia de Usuario	Crear Asignaciones de Ejecución de Test
Descripción	Como test manager, quiero crear Asignaciones de Ejecución de Test para poder asociar los Software de Ejecución de Test con las stories
Criterios de aceptación	El test manager puede crear Asignaciones de Ejecución de Test

Tabla 2.6.33 Historia de usuario Crear Asignaciones de Ejecución de Test

ID	HU16
Historia de Usuario	Ejecución de Test múltiple
Descripción	Como test manager, quiero ejecutar test simultáneos para poder registrar las ejecuciones de todos en un solo registros
Criterios de aceptación	El test manager puede hacer ejecuciones de test múltiples

Tabla 2.6.34 Historia de usuario Ejecución de Test múltiple

ID	HU17
Historia de Usuario	Visualización de Test dashboard
Descripción	Como test manager, quiero visualizar el test dashboard para saber los resultados generales de los test y poder filtrar por diferentes datos
Criterios de aceptación	El test manager puede visualizar el test dashboard

Tabla 2.6.35 Historia de usuario Ejecución de Test múltiple

2.7 Casos de uso

Un caso de uso es un conjunto de acciones realizadas secuencialmente entre el Sistema y un actor/usuario de este que esperan un resultado concreto. Además se pueden dar otros resultados dependiendo de las acciones ejecutadas. Los casos de uso se centran en definir como se reproducen los escenarios para una acción concreta.

2.7.1 Casos de uso Gestión de Proyectos

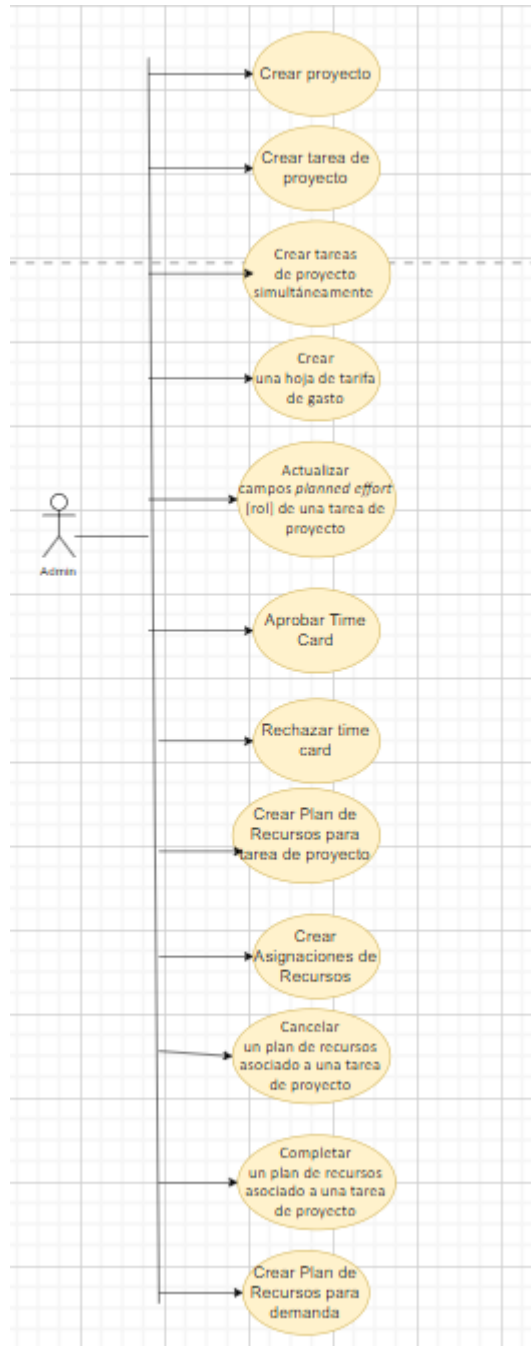


Figure 2.7.1 Casos de Uso Admin

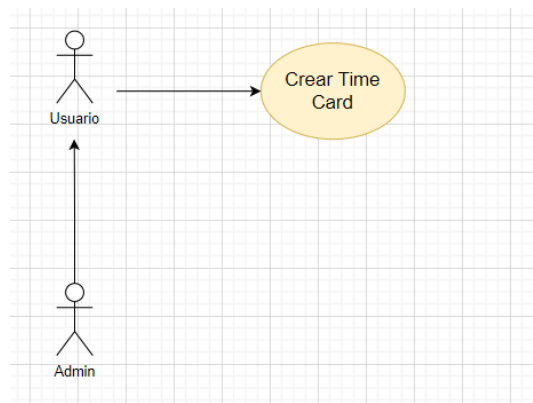


Figure 2.7.2 Casos de uso usuario

CU-01	Crear un proyecto	
Descripción	El sistema debe permitir crear un proyecto	
Precondición		
Secuencia normal	Paso	Acción
	1	El actor admin accede a <i>Project->Projects->All</i>
	2	El sistema muestra la lista de proyectos
	3	El actor admin pulsa New
	4	El sistema muestra el formulario de un nuevo proyecto
	5	El actor admin rellena los datos obligatorios
	6	El actor admin pulsa el botón <i>Save</i>
7	El sistema registra el proyecto	
Excepciones	Paso	Acción
	1a,3a,5a,6a	El actor admin cancela la operación y el caso de uso queda sin efecto
	7a	El sistema comprueba que hay datos obligatorios sin rellenar, muestra un mensaje de error y vuelve al paso 5

Tabla 2.7.1 Caso de uso crear proyecto

CU-02	Crear una tarea de proyecto	
Descripción	El sistema debe permitir crear una tarea de proyecto	
Precondición	El actor admin está en el formulario de un proyecto activo	
Secuencia normal	Paso	Acción
	1	El actor admin pulsa el botón <i>Create Test Phase</i>
	2	El sistema muestra el formulario de una nueva tarea de proyecto
	3	El actor admin rellena los datos obligatorios
	4	El actor admin pulsa el botón <i>Save</i>
5	El sistema registra la tarea de proyecto	
Excepciones	Paso	Acción
	1a,3a,4a	El actor admin cancela la operación y el caso de uso queda sin efecto
	5a	El sistema comprueba que hay datos obligatorios sin rellenar, muestra un mensaje de error y vuelve al paso 3

Tabla 2.7.2 Caso de uso crear tarea de proyecto

CU-03	Crear tareas de proyecto simultáneamente	
Descripción	El sistema debe permitir crear tareas de proyecto sin especificar los datos	
Precondición	El actor admin está en el formulario de un proyecto activo	
Secuencia normal	Paso	Acción
	1	El actor admin pulsa el botón del menú <i>Project Task Creator</i>
	2	El sistema muestra la ventana para crear las tareas de proyecto
	3	El actor admin introduce el número de tareas de proyecto para crear
	4	El actor admin pulsa el botón OK
5	El sistema registra la tareas de proyecto	
Excepciones	Paso	Acción
	1a,3a,4a	El actor admin cancela la operación y el caso de uso queda sin efecto
	5a	El sistema comprueba que no se ha introducido el número, muestra una alerta de error y vuelve al paso 3

Tabla 2.7.3 Caso de uso Crear tareas de proyecto simultáneamente

CU-04	Crear una hoja de tarifa de gasto	
Descripción	El sistema debe permitir crear una hoja de tarifa de gasto	
Precondición		
Secuencia normal	Paso	Acción
	1	El actor admin accede a <i>SilverStorm Implementation->Rates->Task Card Rate</i>
	2	El sistema muestra la lista de proyectos
	3	El actor admin pulsa <i>New</i>
	4	El sistema muestra el formulario de una nueva hoja de tarifa de gasto
	5	El actor admin rellena los datos obligatorios
	6	El actor admin pulsa el botón <i>Save</i>
7	El sistema registra la hoja de tarifa de gasto	
Excepciones	Paso	Acción
	1a,3a,5a,6a	El actor admin cancela la operación y el caso de uso queda sin efecto
	7a	El sistema comprueba que hay datos obligatorios sin rellenar, muestra un mensaje de error y vuelve al paso 5

Tabla 2.7.4 Caso de Uso crear una hoja de tarifa de gasto

CU-05	Actualizar campos <i>planned effort</i> [rol] de una tarea de proyecto	
Descripción	El sistema debe permitir actualizar los campos <i>planned effort</i> [rol] de una tarea de proyecto	
Precondición	El actor admin está en el formulario de una tarea de proyecto activa	
Secuencia normal	Paso	Acción
	1	El actor admin cambia los campos que considera
	2	El actor admin pulsa el botón <i>Save</i>
	3	El sistema calcula los campos <i>planned cost</i> [rol] de la tarea de proyecto con los datos <i>planned effort</i> [rol] introducidos.
	4	El sistema calcula los campos <i>planned effort</i> y <i>planned cost</i> de la tarea de proyecto y actualiza el registro
	5	El sistema calcula los campos <i>planned effort</i> , <i>planned cost</i> y <i>planned cost</i> (cost price) del proyecto asociado y actualiza el registro
Excepciones	Paso	Acción
	1a,2a	El actor admin cancela la operación y el caso de uso queda sin efecto
	3a	El sistema comprueba que hay datos obligatorios sin rellenar, muestra un mensaje de error y vuelve al paso 1
	3b	El sistema comprueba que todos los datos de campos <i>Planned Effort</i> [rol] son 0, muestra un mensaje de error y vuelve al paso 1
	3c	El sistema comprueba que uno o más campos <i>Planned Effort</i> [rol] mayor que 0 no tienen hoja de tarifa de gasto asociada, , muestra un mensaje de error y vuelve al paso 1

Tabla 2.7.5 Caso de Uso actualizar los campos *planned effort* [rol] de una tarea de proyecto

CU-06	Crear una <i>Time Card</i>	
Descripción	El sistema debe permitir crear una <i>time card</i>	
Precondición	El actor usuario está en el formulario de una tarea de proyecto activa	
Secuencia normal	Paso	Acción
	1	El actor usuario pulsa en la related list <i>Time Card</i>
	2	El sistema muestra la lista de <i>Time Cards</i> relacionadas con la tarea de proyecto
	3	El actor usuario pulsa el botón <i>New</i>
	4	El sistema muestra el formulario de creación de una <i>time card</i>
	5	El actor usuario rellena los campos
	6	El actor usuario pulsa el botón <i>Save</i>
	7	El sistema registra la <i>Time Card</i>
Excepciones	Paso	Acción
	1a,3a,5a,6a	El actor usuario cancela la operación y el caso de uso queda sin efecto
	7a	El sistema comprueba que hay datos obligatorios sin rellenar, muestra un mensaje de error y vuelve al paso 5
	7b	El sistema comprueba que no existe una hoja de tarifa de gasto asociada al rol y a la tarea de proyecto indicadas, muestra un mensaje de error y vuelve al paso 5

Tabla 2.7.6 Caso de Uso crear una *Time Card*

CU-07	Aprobación de una <i>Time Card</i>	
Descripción	El sistema debe permitir aprobar un registro <i>Time Card</i>	
Precondición	El sistema esta en una tarea de proyecto con <i>Time Cards</i> en estado Pending asociadas	
Secuencia normal	Paso	Acción
	1	El actor admin pulsa en la related list <i>Time Card</i>
	2	El sistema muestra la lista de <i>Time Cards</i> relacionadas con la tarea de proyecto
	3	El actor admin accede a una <i>Time Card</i>
	4	El sistema muestra el formulario de creación de una <i>time card</i>
	5	El actor pulsa el botón <i>Approved</i>
	6	El sistema cambia el estado de la <i>Time Card</i> a <i>processed</i> y la actualiza
	7	El sistema calcula los campos <i>actual effort</i> [rol] y <i>actual cost</i> [rol] de la tarea de proyecto asociada
	8	El sistema calcula los campos <i>actual effort</i> y <i>actual cost</i> de la tarea de proyecto asociada
	9	El sistema calcula los campos <i>projected effort</i> [rol] y <i>projected cost</i> [rol] de la tarea de proyecto asociada
	10	El sistema calcula los campos <i>projected effort</i> y <i>projected cost</i> de la tarea de proyecto asociada y actualiza el registro
11	El sistema calcula los campos <i>actual effort</i> , <i>actual cost</i> , <i>actual cost (cost price)</i> , <i>projected effort</i> , <i>projected cost</i> y <i>projected cost (cost price)</i> , del proyecto asociado y actualiza el registro	
Excepciones	Paso	Acción
	1a,3a,5a	El actor admin cancela la operación y el caso de uso queda sin efecto
	6a	El sistema comprueba que hay datos obligatorios sin rellenar, muestra un mensaje de error y vuelve al paso 5

Tabla 2.7.7 Caso de uso Aprobación de una *Time Card*

CU-08	Rechazo de una <i>Time Card</i>	
Descripción	El sistema debe permitir rechazar un registro <i>Time Card</i>	
Precondición	El sistema está en una tarea de proyecto con <i>Time Cards</i> en estado Pending asociadas	
Secuencia normal	Paso	Acción
	1	El actor admin pulsa en la related list <i>Time Card</i>
	2	El sistema muestra la lista de <i>Time Cards</i> relacionadas con la tarea de proyecto
	3	El actor admin accede a una <i>Time Card</i>
	4	El sistema muestra el formulario de creación de una <i>time card</i>
	5	El actor pulsa el botón <i>Rejected</i>
Excepciones	Paso	Acción
	1a,3a,5a	El actor admin cancela la operación y el caso de uso queda sin efecto
	6a	El sistema comprueba que hay datos obligatorios sin rellenar, muestra un mensaje de error y vuelve al paso 5

Tabla 2.7.8 Caso de uso rechazo de una *Time Card*

CU-09	Creación de un Plan de Recursos desde tarea de proyecto	
Descripción	El sistema debe permitir crear un registro Plan de Recursos	
Precondición	El actor admin está en el formulario de una tarea de proyecto activa	
Secuencia normal	Paso	Acción
	1	El actor admin pulsa en la related list <i>Resource Plans</i>
	2	El sistema muestra la lista de planes de recursos relacionadas con la tarea de proyecto
	3	El actor admin pulsa el botón <i>New</i>
	4	El sistema muestra el formulario de creación de un Plan de Recursos con la tarea de proyecto en el campo <i>Task</i>
	5	El actor admin completa los campos del formulario del Plan de recursos
	6	El actor admin pulsa el botón <i>Save</i>
Excepciones	Paso	Acción
	1a,3a,5a,6a	El actor admin cancela la operación y el caso de uso queda sin efecto
	7a	El sistema comprueba que hay datos obligatorios sin rellenar, muestra un mensaje de error y vuelve al paso 5
	7b	El sistema comprueba que no existe una hoja de tarifa de gasto asociada al rol y a la tarea de proyecto indicadas, muestra un mensaje de error y vuelve al paso 5

Tabla 2.7.9 Caso de uso creación de un Plan de Recursos desde tarea de proyecto

CU-10	Creación de las asignaciones de recursos	
Descripción	El sistema debe permitir crear registros de asignación de recursos para los planes de recursos	
Precondición	El actor admin está en el formulario de una tarea de proyecto con al menos un plan de recursos sin sus asignaciones creadas	
Secuencia normal	Paso	Acción
	1	El actor admin pulsa en la related list <i>Resource Plans</i>
	2	El sistema muestra la lista de planes de recursos relacionadas con la tarea de proyecto
	3	El actor admin pulsa en un Plan de Recursos
	4	El sistema muestra el formulario de un Plan de Recursos
	5	El actor pulsa el botón <i>Create Soft Allocation</i>
	6	El sistema crea las Asignaciones de Recursos asociadas al Plan de recursos
	7	El sistema suma el esfuerzo y el coste del plan de recursos a los campos projected effort [rol], projected effort, projected cost [rol] y projected cost en la tarea de proyecto asociada
8	El sistema calcula los campos projected effort, projected cost y projected cost (cost price) en el proyecto asociado a la tarea de proyecto	
Excepciones	Paso	Acción
	1a,3a,5a	El actor admin cancela la operación y el caso de uso queda sin efecto
	6a	El sistema comprueba que hay datos obligatorios sin rellenar, muestra un mensaje de error y vuelve al paso 5

Tabla 2.7.10 Caso de uso creación de las asignaciones de recursos en un plan de recursos asociado a una tarea de proyecto

CU-11	Cancelar un plan de recursos asociado a una tarea de proyecto	
Descripción	El sistema debe permitir cancelar un Plan de Recursos	
Precondición	El Plan de Recursos está en estado <i>Planning</i> .	
Secuencia normal	Paso	Acción
	1	El actor admin pulsa en la related list <i>Resource Plans</i>
	2	El sistema muestra la lista de planes de recursos relacionadas con la tarea de proyecto
	3	El actor admin pulsa en un Plan de Recursos
	4	El sistema muestra el formulario de un Plan de Recursos
	5	El actor pulsa el botón <i>Cancel</i>
	6	El sistema cambia el estado del Plan de recursos a <i>Cancelled</i> y actualiza el registro
	7	El sistema resta el esfuerzo y el coste del plan de recursos los campos <i>projected effort [rol]</i> , <i>projected effort</i> , <i>projected cost [rol]</i> y <i>projected cost</i> en la tarea de proyecto asociada
	8	El sistema calcula los campos <i>projected effort</i> , <i>projected cost</i> y <i>projected cost (cost price)</i> en el proyecto asociado a la tarea de proyecto
Excepciones	Paso	Acción
	1a,3a,5a	El actor admin cancela la operación y el caso de uso queda sin efecto
	6a	El sistema comprueba que hay datos obligatorios sin rellenar, muestra un mensaje de error y vuelve al paso 5

Tabla 2.7.11 Caso de uso cancelar un plan de recursos asociado a una tarea de proyectos

CU-12	Completar un plan de recursos asociado a una tarea de proyectos	
Descripción	El sistema debe permitir cancelar un Plan de Recursos	
Precondición	El Plan de Recursos está en estado <i>Allocated</i> .	
Secuencia normal	Paso	Acción
	1	El actor admin pulsa en la related list <i>Resource Plans</i>
	2	El sistema muestra la lista de planes de recursos relacionadas con la tarea de proyecto
	3	El actor admin pulsa en un Plan de Recursos
	4	El sistema muestra el formulario de un Plan de Recursos
	5	El actor pulsa el botón <i>Complete</i>
	6	El sistema cambia el estado del Plan de recursos a <i>Completed</i> y actualiza el registro
	7	El sistema resta el esfuerzo y el coste del plan de recursos los campos <i>projected effort [rol]</i> , <i>projected effort</i> , <i>projected cost [rol]</i> y <i>projected cost</i> en la tarea de proyecto asociada
	8	El sistema calcula los campos <i>projected effort</i> , <i>projected cost</i> y <i>projected cost (cost price)</i> en el proyecto asociado a la tarea de proyecto
Excepciones	Paso	Acción
	1a,3a,5a	El actor admin cancela la operación y el caso de uso queda sin efecto
	6a	El sistema comprueba que hay datos obligatorios sin rellenar, muestra un mensaje de error y vuelve al paso 5

Tabla 2.7.12 Caso de uso completar un plan de recursos asociado a una tarea de proyectos

CU-13	Creación de un Plan de Recursos desde demanda	
Descripción	El sistema debe permitir crear un registro Demanda	
Precondición	El actor admin está en el formulario de una demanda de tipo <i>project</i>	
Secuencia normal	Paso	Acción
	1	El actor admin pulsa en la related list <i>Resource Plans</i>
	2	El sistema muestra la lista de planes de recursos relacionadas con la tarea de proyecto
	3	El actor admin pulsa el botón <i>New</i>
	4	El sistema muestra el formulario de creación de un Plan de Recursos con la demand en el campo <i>Task</i> , el campo <i>resource type</i> con valor <i>Role Resource</i> y las opciones del campo <i>project task: 01-Initiate, 02-Prepare, 03-Create, 04-Transition, 05-Closed, 06-Management</i>
	5	El actor admin completa los campos del formulario del Plan de recursos
	6	El actor admin pulsa el botón <i>Save</i>
	7	El sistema registra el plan de recursos con state en <i>Planning</i>
Excepciones	Paso	Acción
	1a,3a,5a,6a	El actor admin cancela la operación y el caso de uso queda sin efecto
	7a	El sistema comprueba que hay datos obligatorios sin rellenar, muestra un mensaje de error y vuelve al paso 5

Tabla 2.7.13 Caso de uso creación de un Plan de Recursos desde demanda

CU-14	Creación de un Plan de Recursos desde demanda	
Descripción	El sistema debe permitir crear un registro Demanda	
Precondición	El actor admin está en el formulario de una demanda de tipo <i>service improvement</i>	
Secuencia normal	Paso	Acción
	1	El actor admin pulsa en la related list <i>Resource Plans</i>
	2	El sistema muestra la lista de planes de recursos relacionadas con la tarea de proyecto
	3	El actor admin pulsa el botón <i>New</i>
	4	El sistema muestra el formulario de creación de un Plan de Recursos con la demand en el campo <i>Task</i> , el campo <i>resource type</i> con valor <i>Role Resource</i> y las opciones del campo <i>project task: 00-SERVICE TRANSITION, 01-ADVANCE PLATFORM SUPPORT, 02-BASIC CONFIGURATION SUPPORT, 03-ENHANCEMENT SERVICES, 04-ARQUITECTURAL SERVICES, 05 -UPGRADE, 06 – MANAGEMENT</i>
	5	El actor admin completa los campos del formulario del Plan de recursos
	6	El actor admin pulsa el botón <i>Save</i>
	7	El sistema registra el plan de recursos con state en <i>Planning</i>
Excepciones	Paso	Acción
	1a,3a,5a,6a	El actor admin cancela la operación y el caso de uso queda sin efecto
	7a	El sistema comprueba que hay datos obligatorios sin rellenar, muestra un mensaje de error y vuelve al paso 5

Tabla 2.7.14 Caso de uso Creación de un Plan de Recursos desde demanda

2.7.2 Casos de uso Test Management

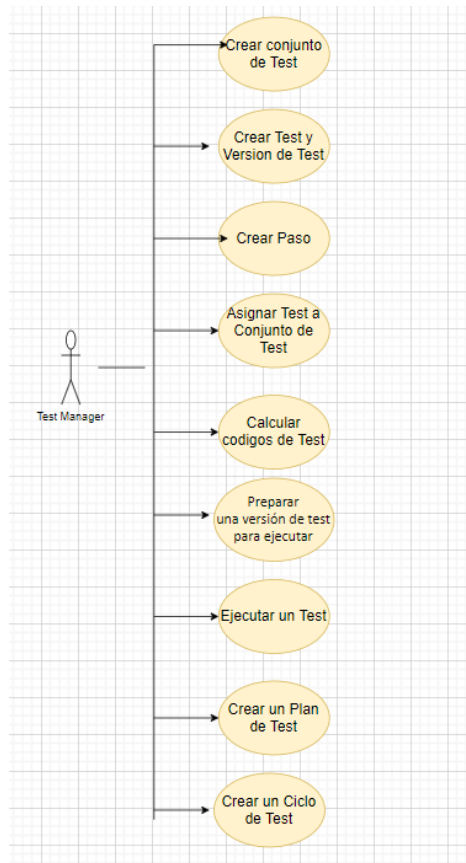


Figure 2.7.3 Casos de uso Test Manager Parte 1

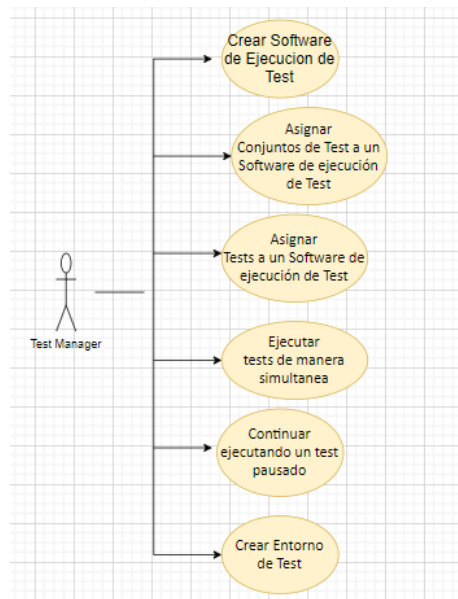


Figure 2.7.4 Casos de uso Test Manager Parte 2

CU-01	Creación de un Conjunto de Test	
Descripción	El sistema debe permitir registrar un Conjunto de Test	
Precondición		
Secuencia normal	Paso	Acción
	1	El actor test manager accede a <i>Test Management 2.0</i> -> <i>Test Set</i>
	2	El sistema muestra la lista de Conjuntos de Test
	3	El actor test manager pulsa New
	4	El sistema muestra el formulario de un nuevo conjunto de test
	5	El actor test manager rellena los datos obligatorios
	6	El actor test manager pulsa el botón <i>Save</i>
	7	El sistema registra el conjunto de test
Excepciones	Paso	Acción
	1a,3a,5a,6a	El actor test manager cancela la operación y el caso de uso queda sin efecto
	7a	El sistema comprueba que hay datos obligatorios sin rellenar, muestra un mensaje de error y vuelve al paso 5

Tabla 2.7.15 Creación de un Conjunto de Test

CU-02	Creación de un Test y Versión de Test	
Descripción	El sistema debe permitir registrar un Test y una Test Version	
Precondición		
Secuencia normal	Paso	Acción
	1	El actor test manager accede a <i>Test Management 2.0</i> -> <i>Test</i>
	2	El sistema muestra la lista de Test
	3	El actor test manager pulsa New
	4	El sistema muestra el formulario de una nueva versión de test
	5	El actor test manager rellena los datos obligatorios
	6	El actor test manager pulsa el botón <i>Save</i>
	7	El sistema registra el test
	8	El sistema registra la versión de test con versión 1
Excepciones	Paso	Acción
	1a,3a,5a,6a	El actor test manager cancela la operación y el caso de uso queda sin efecto
	7a	El sistema comprueba que hay datos obligatorios sin rellenar, muestra un mensaje de error y vuelve al paso 5

Tabla 2.7.16 Caso de uso Creación de un Test y Versión de Test

CU-03	Creación de un Paso	
Descripción	El sistema debe permitir crear un paso	
Precondición	El actor test manager se encuentra en un Test que tiene Test Version en estado Draft	
Secuencia normal	Paso	Acción
	1	El actor test manager pulsa en la <i>related list Test Version</i>
	2	El sistema muestra la lista de versión de test asociados al Test
	3	El actor test manager accede a la versión de Test
	4	El sistema carga el formulario de la versión de Test
	5	El actor test manager introduce el nuevo paso en el campo <i>Add Step</i>
6	El sistema crea un nuevo Paso y le asigna el código correspondiente a su orden	
Excepciones	Paso	Acción
	1a,3a,5a,	El actor test manager cancela la operación y el caso de uso queda sin efecto

Tabla 2.7.17 Caso de uso Creación de un Paso

CU-04	Asignar Test a Conjunto de Test	
Descripción	El sistema permite asignar Test a Conjuntos de Test	
Precondición		
Secuencia normal	Paso	Acción
	1	El actor test manager accede a Test Management 2.0 ->Test
	2	El sistema muestra la lista de Test
	3	El actor test manager selecciona los Test para asignar
	4	El actor test manager pulsa <i>Add to Test Set</i>
	5	El sistema muestra la ventana para seleccionar el conjunto de test
	6	El actor test manager rellena el test set
	7	El actor test manager pulsa el botón Add
8	El sistema asocia los Test seleccionados con el Conjunto de Test	
Excepciones	Paso	Acción
	1a,3a,4a,6a,7a	El actor test manager cancela la operación y el caso de uso queda sin efecto
	5a	El sistema comprueba que no se ha seleccionado ningún Test, muestra un mensaje de error y vuelve al paso 3
	8a	El sistema comprueba que hay datos obligatorios sin rellenar, muestra un mensaje de error y vuelve al paso 6
8b	El sistema comprueba que la compañía de algunos test no se corresponde con la del conjunto de test, no asigna esos test, y el caso de uso finaliza normalment	

Tabla 2.7.18 Caso de uso Asignar Test a Conjunto de Test

CU-05	Calcular los códigos de Test de un Conjunto de Test	
Descripción	El sistema debe permitir actualizar simultáneamente los test de un conjunto de test	
Precondición	El Conjunto de Test tiene Test asignados	
Secuencia normal	Paso	Acción
	1	El actor test manager accede a <i>Test Management 2.0</i> -> <i>Test Set</i>
	2	El sistema muestra la lista de Conjuntos de Test
	3	El actor test manager pulsa un conjunto de Test
	4	El sistema muestra el formulario del conjunto de test
	5	El actor test manager pulsa el botón Update Test Code
	6	El sistema muestra la ventana para generar los códigos
	7	El actor test manager rellena los datos
	8	El actor test manager pulsa Accept
9	El sistema actualiza el código de los test	
Excepciones	Paso	Acción
	1a,3a,5a,7a,8a	El actor test manager cancela la operación y el caso de uso queda sin efecto
	9a	El sistema comprueba que hay datos obligatorios sin rellenar, muestra un mensaje de error y vuelve al paso 7
	9b	El sistema comprueba que el dato introducido en <i>prefix</i> no es válido, muestra un mensaje de error y vuelve al paso 7
	9c	El sistema comprueba que el dato introducido en <i>number of digits</i> no es válido, muestra un mensaje de error y vuelve al paso 7
9d	El sistema comprueba que el dato introducido en <i>initial value</i> no es válido, muestra un mensaje de error y vuelve al paso 7	

Tabla 2.7.19 Caso de uso Calcular los códigos de Test de un Conjunto de Test

CU-06	Preparar una versión de test para ejecutar	
Descripción	El sistema debe permitir pasar una versión de test a estado ready	
Precondición	El test manager está en un test que contiene una versión de test en draft	
Secuencia normal	Paso	Acción
	1	El actor test manager pulsa en la <i>related list Test Version</i>
	2	El sistema muestra la lista de versión de test asociados al Test
	3	El actor test manager accede a la versión de Test
	4	El sistema carga el formulario de la versión de Test
	5	El actor test manager pulsa <i>Ready</i>
	6	El sistema pasa la versión de test a ready y la convierte en no editable
7	El sistema pasa cualquier otra versión de test de ese Test a <i>Retired</i>	
Excepciones	Paso	Acción
	1a,3a,5 ^a	El actor test manager cancela la operación y el caso de uso queda sin efecto
	6a	El sistema detecta que ningún paso de la versión de test esta catalogado como needs verification, muestra un mensaje de error y vuelve al paso 5

Tabla 2.7.20 Caso de uso Preparar una versión de test para ejecutar

CU-07	Ejecutar un test	
Descripción	El sistema permite ejecutar los test y almacenar los datos	
Precondición	El Test tiene una Version de Test en estado ready	
Secuencia normal	Paso	Acción
	1	El actor test manager pulsa en la <i>related list Test Version</i>
	2	El sistema muestra la lista de versión de test asociados al Test
	3	El actor test manager accede a la versión de Test en <i>Ready</i>
	4	El sistema carga el formulario de la versión de Test
	5	El actor test manager pulsa <i>Run</i>
	6	El sistema carga la ventana de ejecución de Test
	7	El actor test manager rellena los datos para comenzar
	8	El actor test manager pulsa <i>Run</i>
	9	El sistema carga la ventana con los pasos del Test
	10	El actor test manager completa los pasos que necesitan verificación
	11	El actor test manager pulsa el botón done
	12	El sistema crea un registro <i>Test Run</i> y otro resultado de test con el resultado del testeo
Excepciones	Paso	Acción
	1a,3a,5a,7a,8a,10a,11a	El actor test manager cancela la operación y el caso de uso queda sin efecto
	11b	El actor test manager pulsa el botón Pause y el caso de uso queda en pausa

Tabla 2.7.21 Caso de uso Ejecutar un test

CU-08	Crear un plan de test	
Descripción	El sistema permite crear un plan de test	
Precondición		
Secuencia normal	Paso	Acción
	1	El actor test manager accede a Test Management 2.0 -> Test Plan
	2	El sistema muestra la lista de Plan de Test
	3	El actor test manager pulsa New
	4	El sistema muestra el formulario de un nuevo plan de test
	5	El actor test manager rellena los datos
	6	El actor test manager pulsa el botón Save
	7	El sistema registra el plan de test
Excepciones	Paso	Acción
	1a,3a,5a,6a	El actor test manager cancela la operación y el caso de uso queda sin efecto
	7a	El sistema comprueba que hay datos obligatorios sin rellenar, muestra un mensaje de error y vuelve al paso 5

Tabla 2.7.22 Caso de uso Crear un plan de test

CU-09	Crear un ciclo de test	
Descripción	El sistema permite crear un ciclo de test	
Precondición	El actor test manager esta en un plan de test	
Secuencia normal	Paso	Acción
	1	El actor test manager pulsa en la <i>related list Test Cycles</i>
	2	El sistema muestra la lista de ciclos de test asociados al plan de test
	3	El actor test manager pulsa el botón <i>New</i>
	4	El sistema muestra el formulario de un nuevo plan de test
	5	El actor test manager rellena los datos
	6	El actor test manager pulsa el botón <i>Save</i>
	7	El sistema registra el ciclo de test
Excepciones	Paso	Acción
	1a,3a,5a,6a	El actor test manager cancela la operación y el caso de uso queda sin efecto
	7a	El sistema comprueba que hay datos obligatorios sin rellenar, muestra un mensaje de error y vuelve al paso 5

Tabla 2.7.23 Caso de uso Crear un ciclo de test

CU-10	Crear un software de ejecución de test	
Descripción	El sistema permite crear un software de ejecución de test	
Precondición	El actor test manager está en un ciclo de test	
Secuencia normal	Paso	Acción
	1	El actor test manager accede a <i>Test Management 2.0 -> Test Execution Suite</i>
	2	El sistema muestra la lista de software de ejecución de test
	3	El actor test manager pulsa <i>New</i>
	4	El sistema muestra el formulario de un nuevo software de ejecución de test
	5	El actor test manager rellena los datos
	6	El actor test manager pulsa el botón <i>Save</i>
	7	El sistema registra el ciclo de software de ejecución de test
Excepciones	Paso	Acción
	1a,3a,5a,6a	El actor test manager cancela la operación y el caso de uso queda sin efecto
	7a	El sistema comprueba que hay datos obligatorios sin rellenar, muestra un mensaje de error y vuelve al paso 5

Tabla 2.7.24 Caso de uso Crear un software de ejecución de test

CU-11	Asignar Conjuntos de Test a un Software de ejecución de Test	
Descripción	El sistema debe permitir asignar conjuntos de test a un software de ejecución de test	
Precondición		
Secuencia normal	Paso	Acción
	1	El actor test manager accede a <i>Test Management 2.0</i> -> <i>Test Execution Suite</i>
	2	El sistema muestra la lista de software de ejecución de test
	3	El actor test manager pulsa en un software de ejecución de test
	4	El sistema muestra el formulario del software de ejecución de test
	5	El actor test pulsa el botón <i>Add Test Sets</i>
	6	El sistema muestra el listado de Conjunto de Test y el botón <i>Add To Execution Suite</i>
	7	El actor test manager selecciona los conjuntos de test que quiere añadir al software de ejecución de test
	8	El actor test manager pulsa el botón <i>Add To Execution Suite</i>
	9	El sistema añade un registro asignación de ejecución de test por cada test asignado, relacionando cada test con el software de ejecución de test
Excepciones	Paso	Acción
	1a,3a,5a,7a,8a	El actor test manager cancela la operación y el caso de uso queda sin efecto
	9a	El sistema comprueba que no se ha seleccionado ningún conjunto de test, muestra un mensaje de error y vuelve al paso 7

Tabla 2.7.25 Caso de uso Asignar Conjuntos de Test a un Software de ejecución de Test

CU-12	Asignar Tests a un Software de ejecución de Test	
Descripción	El sistema debe permitir asignar tests a un software de ejecución de test	
Precondición		
Secuencia normal	Paso	Acción
	1	El actor test manager accede a <i>Test Management 2.0</i> -> <i>Test Execution Suite</i>
	2	El sistema muestra la lista de software de ejecución de test
	3	El actor test manager pulsa en un software de ejecución de test
	4	El sistema muestra el formulario del software de ejecución de test
	5	El actor test pulsa el botón <i>Add Test</i>
	6	El sistema muestra el listado de Test y el botón <i>Add To Execution Suite</i>
	7	El actor test manager selecciona los tests que quiere añadir al software de ejecución de test
	8	El actor test manager pulsa el botón <i>Add To Execution Suite</i>
	9	El sistema añade un registro asignación de ejecución de test por cada test asignado, relacionando cada test con el software de ejecución de test
Excepciones	Paso	Acción
	1a,3a,5a,7a,8a	El actor test manager cancela la operación y el caso de uso queda sin efecto
	9a	El sistema comprueba que no se ha seleccionado ningún test, muestra un mensaje de error y vuelve al paso 7

Tabla 2.7.26 Caso de uso Asignar Tests a un Software de ejecución de Test

CU-13	Ejecutar tests de manera simultanea	
Descripción	El sistema permite ejecutar test de forma simultánea	
Precondición	El actor test manager tiene software de ejecución de test asignados así mismo que están activos	
Secuencia normal	Paso	Acción
	1	El actor test manager accede a <i>Test Management 2.0</i> -> <i>Test Assigned To Me</i>
	2	El sistema muestra la lista de asignaciones de ejecución de test agrupadas según el software de ejecución de test al que están asignadas
	3	El actor test manager selecciona los test que quiere ejecutar
	4	El actor test manager pulsa el botón Run
	5	El sistema carga la ventana de ejecución de Test
	6	El actor test manager rellena los datos para comenzar
	7	El actor test manager pulsa <i>Run</i>
	8	El sistema carga la ventana con los pasos del primer Test
	9	El actor test manager completa los pasos que necesitan verificación de todos los test
	10	El actor test manager pulsa el botón done
11	El sistema crea un registro <i>Test Run general</i> y un registro resultado de test por cada test seleccionado con el resultado del testeo	
Excepciones	Paso	Acción
	1a,3a,4a,6a,9a,10a	El actor test manager cancela la operación y el caso de uso queda sin efecto
	10b	El actor test manager pulsa el botón Pause y el caso de uso queda en pausa

Tabla 2.7.27 Caso de uso Ejecutar test de manera simultanea

CU-14	Continuar ejecutando un test pausado	
Descripción	El sistema permite continuar ejecutando un test pausado	
Precondición	El actor test manager ha pausado algún testeo previamente	
Secuencia normal	Paso	Acción
	1	El actor test manager accede a <i>Test Management 2.0</i> -> <i>Runs</i>
	2	El sistema muestra la lista de <i>Test Run</i> ejecutados por el actor test manager
	3	El actor test manager pulsa en un <i>Test Run</i> en estado <i>In progress</i>
	4	El sistema muestra el formulario del <i>Test Run</i>
	5	El actor test manager pulsa el botón <i>Continue Run</i>
6	El sistema carga la ventana de ejecución de test en el mismo estado en el que se pauso	
Excepciones	Paso	Acción
	1a,3a,5a	El actor test manager cancela la operación y el caso de uso queda sin efecto

Tabla 2.7.28 Caso de uso Continuar ejecutando un test pausado

CU-15	Crear Entorno de Test	
Descripción	El sistema permite crear un Entorno de Test	
Precondición		
Secuencia normal	Paso	Acción
	1	El actor test manager accede a Test Management 2.0 -> Environment
	2	El sistema muestra la lista de Entornos de Test
	3	El actor test manager pulsa New
	4	El sistema muestra el formulario de un nuevo entorno de test
	5	El actor test manager rellena los datos obligatorios
	6	El actor test manager pulsa el botón Save
	7	El sistema registra el entorno de test
Excepciones	Paso	Acción
	1a,3a,5a,6a	El actor test manager cancela la operación y el caso de uso queda sin efecto
	7a	El sistema comprueba que hay datos obligatorios sin rellenar, muestra un mensaje de error y vuelve al paso 5

Tabla 2.7.29 Caso de uso Crear Entorno de Test

2.8 Implementación de los elementos definidos

Una vez se crearon las historias de usuario de un *sprint*, comenzó el desarrollo de estas. Durante esta parte el desarrollador se comunicó con el *Scrum Master* (en este caso el mismo que el cliente final) para resolver las dudas que surgieron, además de proponerle mejoras para que se cumplieran los distintos requisitos.

2.9 Diagramas de flujo

Para las aplicaciones se han desarrollado algunos diagramas de flujo que clarificaran más el funcionamiento de algunas partes de la story.

2.9.1 Diagramas de flujo Gestión de proyectos

- Actualización tarea de proyecto

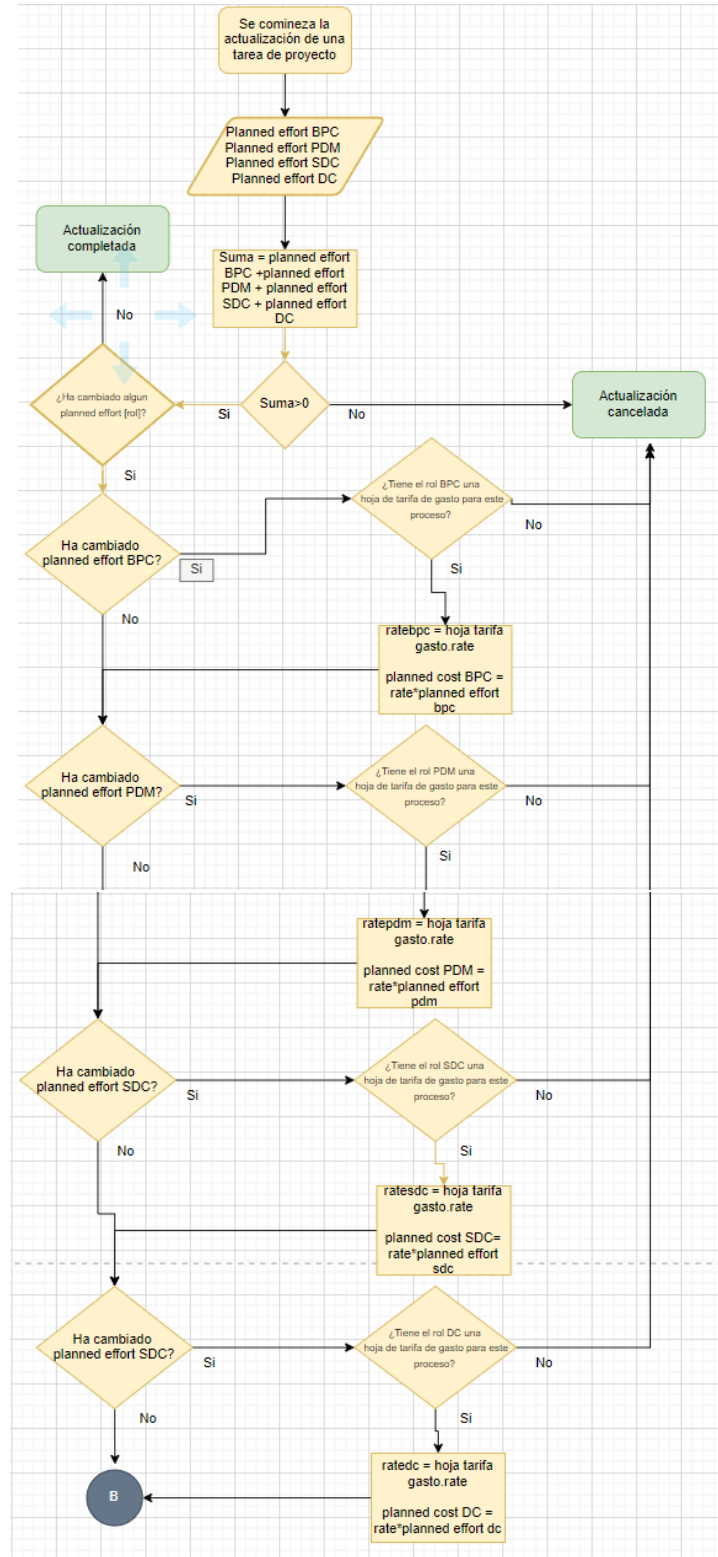


Figure 2.9.1 Actualización tarea de proyecto parte 1

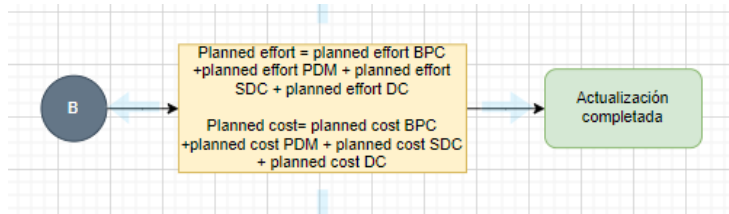


Figure 2.9.2 Actualización tarea de proyecto parte 2

- Aprobar Time Card

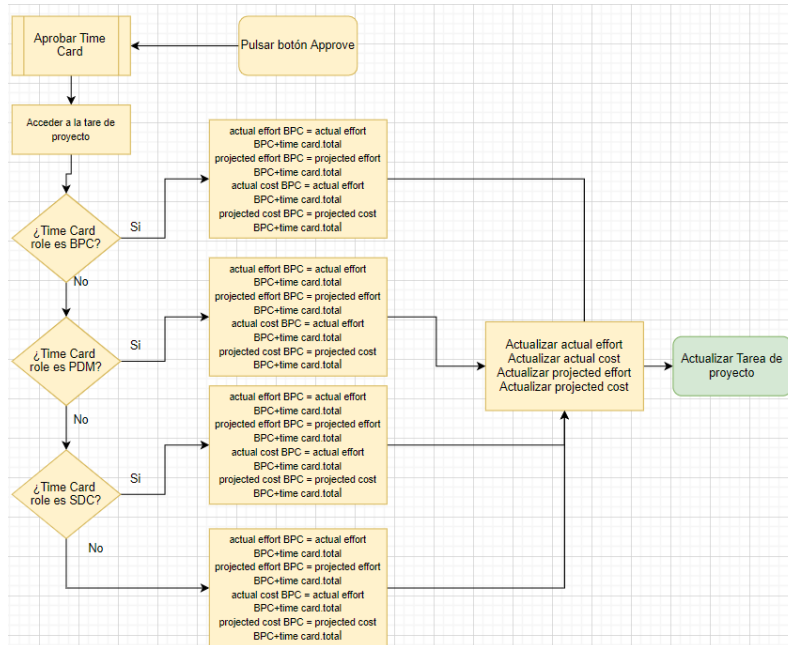


Figure 2.9.3 Aprobar Time Card

- Actualizar proyecto tras la actualización de una de sus tareas de proyecto

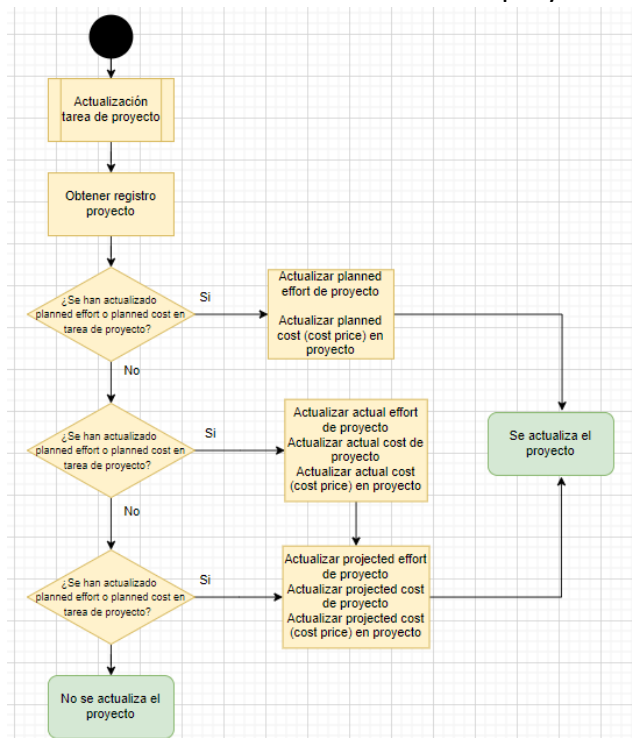


Figure 2.9.4 Actualizar proyecto tras la actualización de las tareas de proyecto

- Crear asignaciones de recursos:

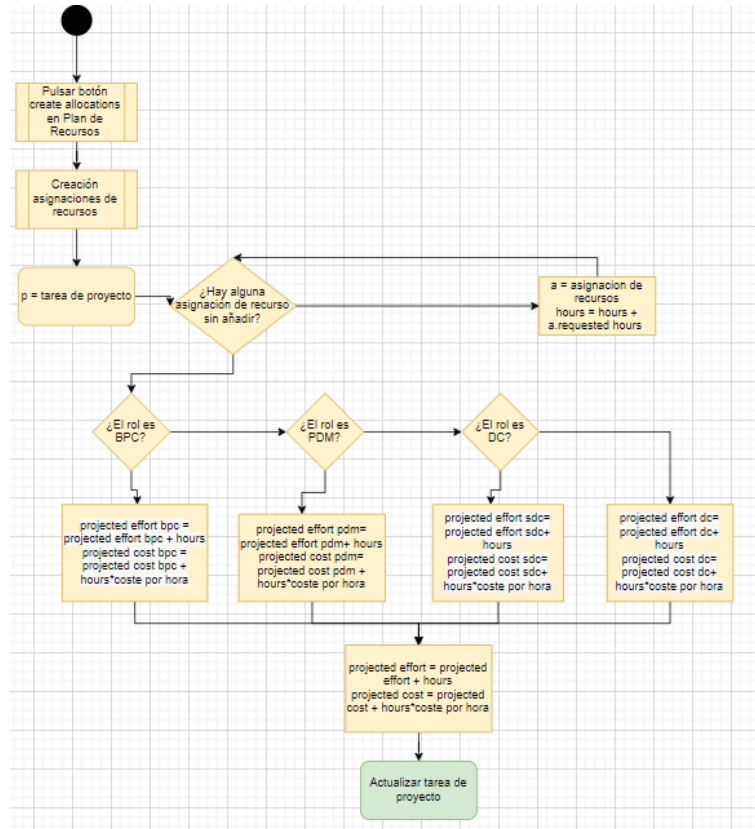


Figure 2.9.5 Crear asignaciones de recursos.

- Actualizar estado de un plan de recursos:

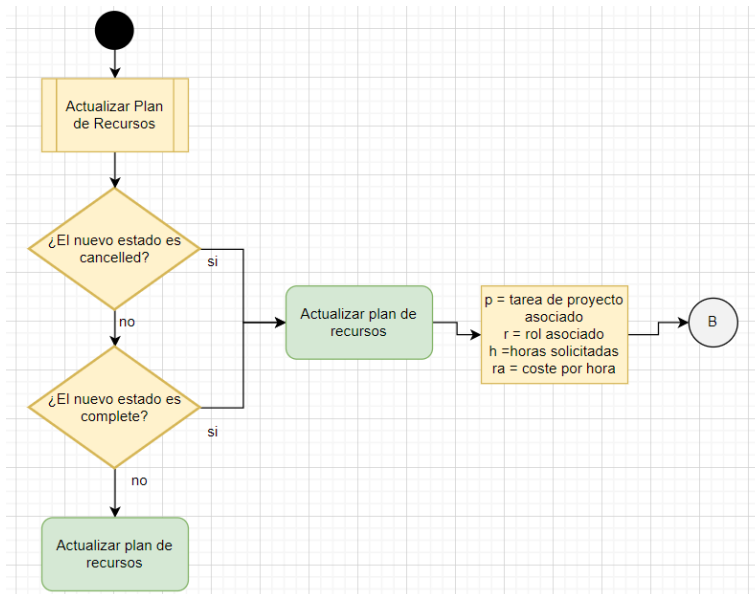


Figure 2.9.6 Actualizar estado de un plan de recursos parte 1

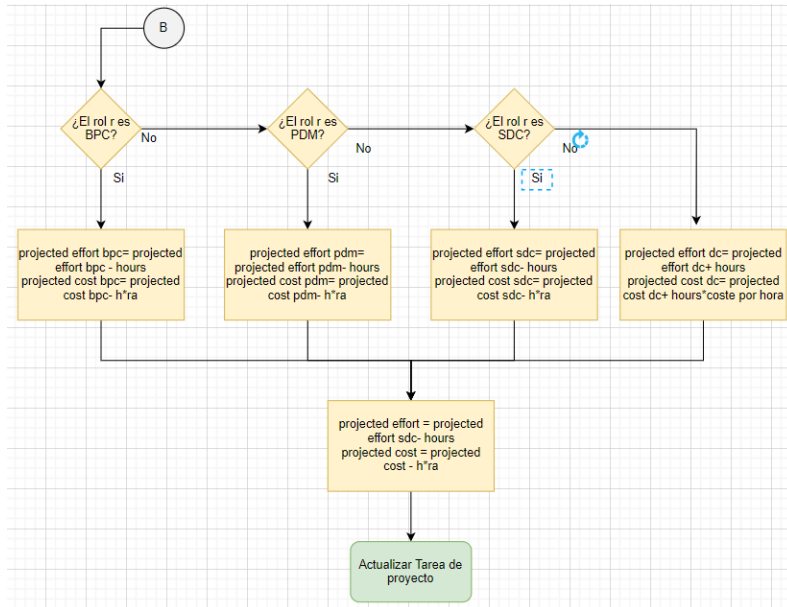


Figure 2.9.7 Actualizar estado de un plan de recursos parte 2

2.9.2 Diagramas de flujo Test Management

- Actualización de códigos de Test

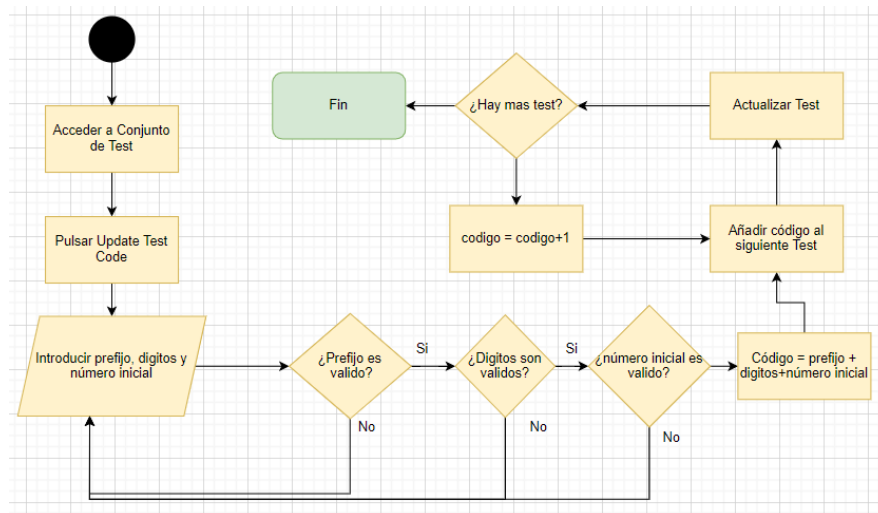


Figure 2.9.8 Actualización de códigos de Test

- Añadir Test a conjunto de Test

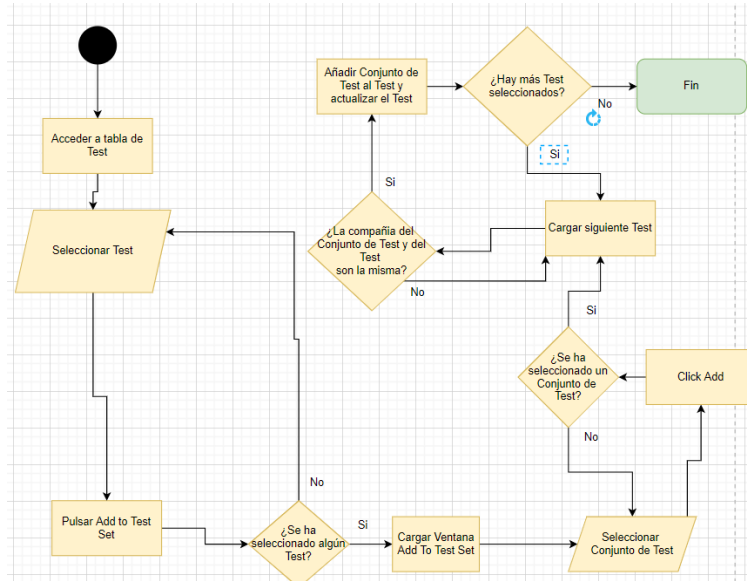


Figure 2.9.9 Añadir Test a conjunto de Test

- Preparación y ejecución de un Test (desde una Versión de Test ya creada:

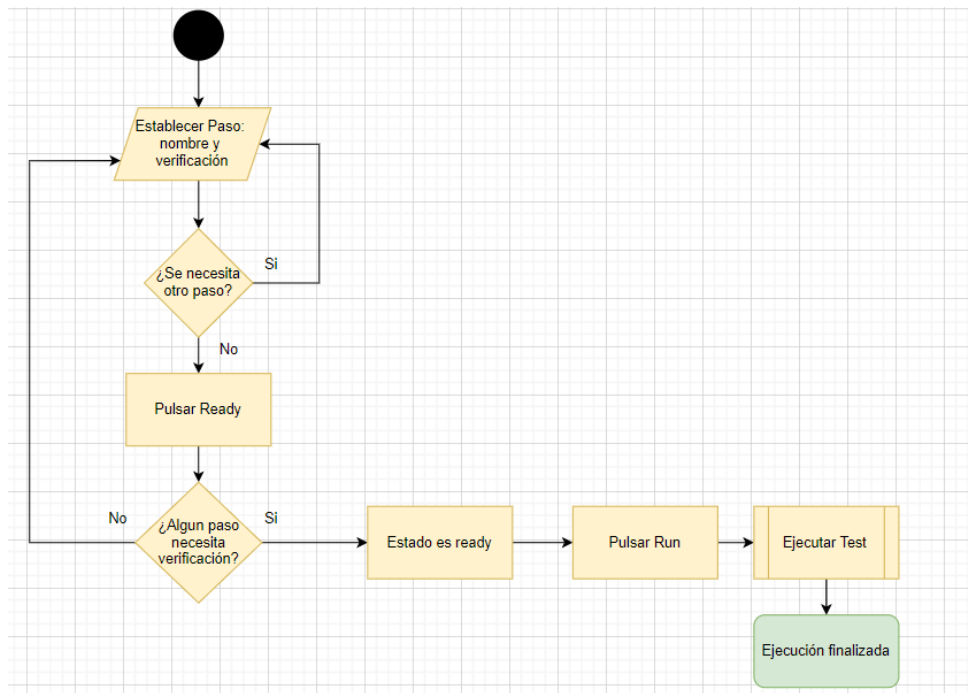


Figure 2.9.10 Preparar ejecución de Test

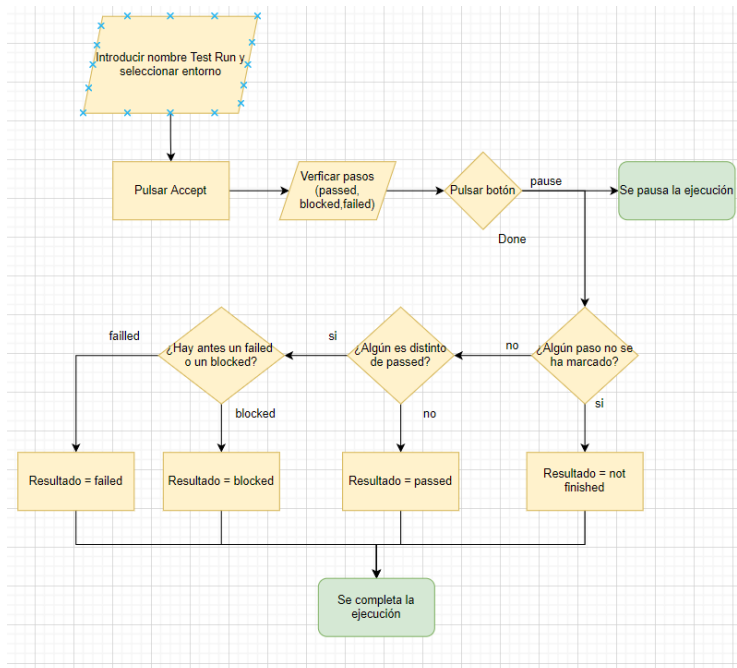


Figure 2.9.11 Ejecución de Test

2.10 Retrospectiva

Tras la realización de la implementación se realizaron reuniones de retrospectiva. En estas reuniones se mostró el trabajo realizado hasta el momento, y se evaluó la evolución de este, a la vez que se marcaron las pautas para el siguiente ciclo del proceso. Se fueron añadiendo requisitos cuando era necesario y se decidió que historias de usuario debidamente creadas se incluían en el siguiente *sprint*.

2.11 Despliegue y mantenimiento del desarrollo

Tras la finalización del proyecto, *SilverStorm*, como cliente, decidió si mover todo lo desarrollado a su instancia de producción, en la que los trabajadores tienen todas las funcionalidades desarrolladas disponibles para su uso diario. Desde allí pueden reportar defectos o fallos del sistema, que, de confirmarse, serán solucionados en la instancia de desarrollo, y si se confirma que la solución es correcta, subir está a la instancia de producción.

El despliegue no tiene por qué hacerse solo al final, sino que puede irse desplegando por partes a medida que van finalizando los *sprints*.

2.12 Línea temporal de trabajo

Durante el desarrollo del Trabajo de Fin de Grado se han realizado diversas reuniones y eventos que han ido marcando el avance de esta a lo largo de los meses que ha durado el desarrollo del Trabajo de Fin de Grado. Esos eventos son los siguientes:

- Reunión inicial de contacto: fue la primera reunión realizada con el cliente para tener una primera toma de contacto sobre los objetivos generales del Trabajo de Fin de Grado.
- Reunión inicial de requisitos: reunión con las diferentes partes de los responsables tanto de *Test Management* como de Gestión de Proyectos para fijar los requisitos iniciales del mismo.
- Reunión revisión documento de diseño inicial: se presentaron los primeros documentos de diseño con las distintas entidades, menús, *dashboards* y demás requerimientos para implementar.
- Reunión historias de usuario iniciales: se presentaron las historias de usuario que entraban en el desarrollo del primer *sprint*.

- Reuniones retrospectivas: se realizaron periódicamente para mostrar los avances realizados y determinar los requisitos e historias de usuario asignadas al siguiente *sprint*. También se han propuesto mejoras para el proyecto.
- Presentación demo final: demostración realizada ante el cliente de la herramienta solicitada con todos los requisitos implementados.

Además de todas estas reuniones se han hecho otras reuniones más a demanda del desarrollador de cara a consultar dudas que han ido surgiendo en el desarrollo de historias de usuario concretas, de cara a agilizar el desarrollo y que esta finalizara de la mejor manera posible.

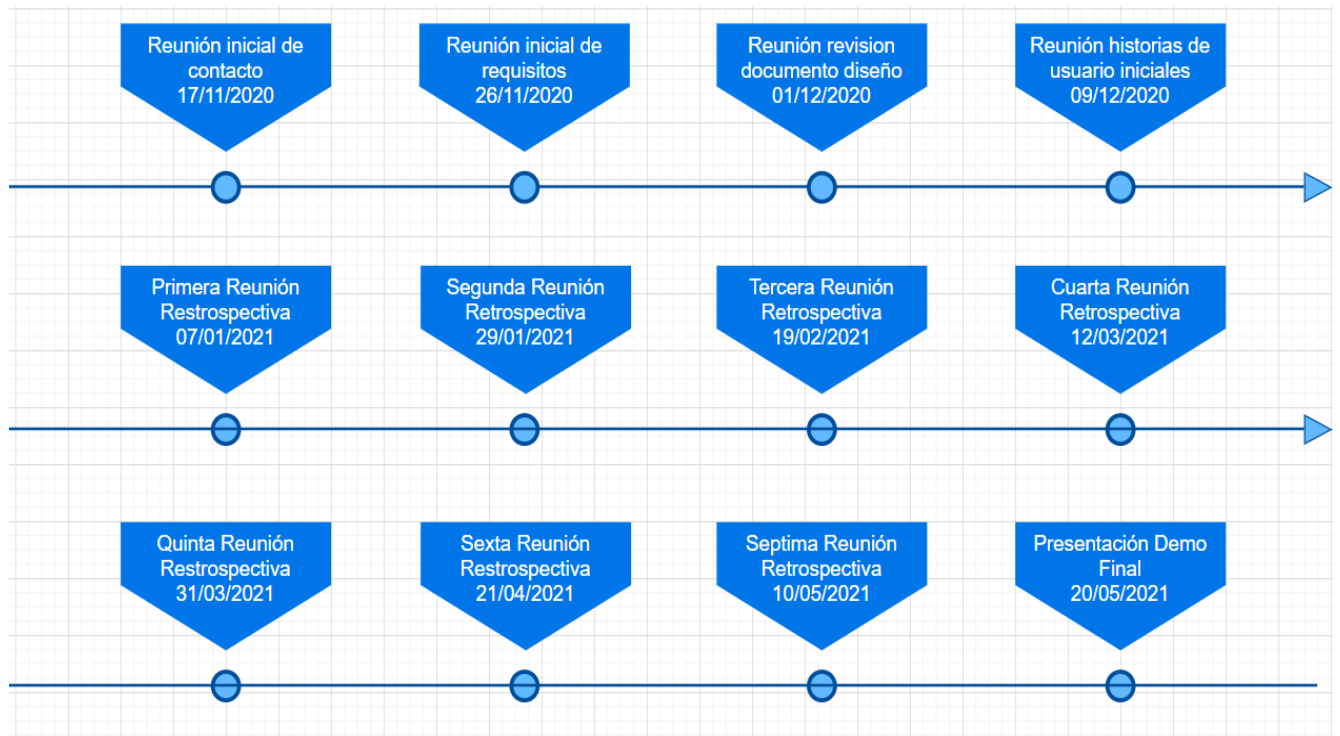


Figura 2.12.1 Diagrama de tiempo del desarrollo

3 ¿Por qué se necesitan estos desarrollos?

3.1 Desarrollo Gestión de Proyectos

En *Silverstorm* el área de trabajo principal es el área de *Delivery*. Más de la mitad de los trabajadores de la empresa están en esta área. Las personas que forman parte de este equipo se dedican a preparar, implementar y mantener los proyectos de los clientes de la empresa, y los beneficios provienen en gran parte del buen de trabajo en estos proyectos, realizando desarrollos eficientes.

Los líderes de este equipo junto con los PDM, que son los encargados de asignar las horas que trabaja cada persona en cada proyecto observaron que el proceso por el que se hacían estas asociaciones era excesivamente genérico, lo que provoca que sea difícil analizar la inversión tanto de esfuerzo como de coste de estos proyectos.

Se observó que para poder mejorar de una forma más sencilla, analizar en que áreas se puede mejorar se puede hacer trasladando la imputación de horas y asignación de recursos desde los proyectos hasta las tareas de proyectos que los componen. Aunque cada proyecto es diferente, hay una serie de tareas de proyecto que en general son comunes a todas, como *Initiate, Prepare o Create*. Analizando estas individualmente el análisis de que tareas de proyectos, se puede saber mejor en que áreas hay que centrarse para mejorar.

El desarrollo de los planes de recursos, también son mucho más sencillo de asignar cuando se crean asociados a las tareas de proyectos. Desde la empresa se ha observado que es difícil saber a qué labor están asignados los recursos cuando se asocian al proyecto, ya que no se define para que son exactamente. Hay tareas de proyecto, como pueden ser las tareas de *Initiate o Prepare* en las que la mayoría de las horas las realizan personas con el rol SDC o PDM y otras como *create* en la que los recursos son personas con rol DC. Al trasladar los planes de recursos a las tareas de proyecto, se sabe exactamente a donde está asignado cada persona, y por lo tanto cuantas horas de un rol determinado hay asignadas.

Otro punto importante es que hasta ahora los estudios de rendimiento se realizaban exclusivamente basándose por lo que está previsto invertir (*planned cost y planned effort*) y lo que has invertido hasta ahora (*actual cost y actual effort*). Además, se tienen las horas que se prevé asignar en ese momento hasta el final del proyecto. Muchos de los proyectos funcionan con bolsas de horas, es decir, que el cliente paga unas horas para el proyecto, que la empresa SilverStorm acepta, y las horas extras que se realicen no se pagan. Por ello se decidió implementar dos campos (*projected cost y projected effort*) que permitan conocer cuál es el gasto proyectado teniendo en cuenta lo invertido hasta ahora y lo que está previsto invertir. Esto permitirá ajustar los procesos a medida que se van realizando, y no solo al final.

Como en el resto de los puntos, toda la funcionalidad del proyecto se ha trasladado a las tareas del proyecto. Los campos indicados en anterior párrafo también se deciden trasladar, para que sea consistente con el resto de las implementaciones. Los mismos campos se mantienen en proyecto, pero en este caso se calcula basándose en la suma del valor de estos campos en todas las tareas de proyecto que contengan.

Finalmente, los responsables del departamento decidieron que les resultaba complicado ver cuantas horas se habían invertido en una tarea de proyecto por cada rol. Esta labor les llevaba mucho tiempo, ya que tenían que mirar todos esos datos de forma individual. Por ejemplo, para ver todas las

horas invertidas por un rol en una tarea de proyecto hasta el momento, tenían que ir a las *time cards* de todos los usuarios y filtrar por el rol y la tarea de proyecto. El coste, la única forma de calcularlo era manualmente. Por eso se decidió implementar el cálculo de todos los esfuerzos y costes divididos por rol, tanto el planeado como el actual o el proyectado. A través de estos campos de rol se calcularían los campos generales de la tarea de proyecto.

3.2 Desarrollo *Test Management*

La parte de *Test Management* es un módulo que se ha decidido implementar en la empresa a raíz de la creación de un departamento de QA que mejore la calidad de las entregas que realizamos a los clientes. Este departamento durante los primeros meses desde su creación ha trabajado con tablas Excel. Tanto desde el departamento como los encargados de la empresa decidieron que este no era el método adecuado, ya que conllevaba más horas de las necesarias para gestionar correctamente los planes de prueba y los testeos. Por ello se decidió añadir el módulo previamente comentado, ya que al ser una funcionalidad incluida dentro de *ServiceNow*, facilita tanto el trabajo de los empleados de *SilverStorm* como el entendimiento de la herramienta por parte de los clientes.

Desde el departamento de QA se notó la necesidad de poder juntar los Test agrupándolos para tener más claro a que correspondía cada Test y poder ejecutar pruebas similares juntas más fácilmente. Por esta razón se dio mucha importancia a los desarrollos relacionados con Conjuntos de Tests.

A su vez se notó la importancia de clasificar correctamente los testeos. En *SilverStorm* se trabajaba simultáneamente con un gran número de clientes, cada uno con sus desarrollos y sus necesidades propias, y también con sus propios planes de pruebas. Por ellos se dio suma importancia al desarrollo de los campos *company*, campos de tipo *reference*, que se asocian con la tabla de compañías. Estos campos se han desarrollado para tablas como Conjunto de Test, Test o Entorno de Test, con el objetivo de vincular exclusivamente registros relacionados con la misma compañía.

El departamento de *Delivery* comprobó que no tenía sentido que un Test pudiera estar relacionada con varios Conjuntos de Test, ya que se supone que cada Test tiene que estar incluido en un proceso completo. Por ello se realizó la implementación de las restricciones para que esta condición se cumpliera.

Por otra parte, se encontró la necesidad de ordenar los Test una vez agrupados por compañías y por Conjuntos de Test. Pero para realizar los procesos de testeo, se necesita también ordenar los Test de un mismo Conjunto de Test. Para ello se decidió crear el campo *code* en la tabla Test y el botón "*Update Test Codes*". Lo que hace este botón es crear códigos de Test para todos los pertenecientes a un mismo Conjunto de Test, basándose en un prefijo y en el número de dígitos. Este evita meter todos los códigos manualmente, lo que reduce el tiempo empleado y evita errores.

Desde el departamento de *Delivery* se notó que era necesario poder asociar los planes a las *stories*, para que cada trabajador sepa en que pruebas se basa el desarrollo que está realizando, y pueda probarlas también si es necesario. Por ello se decidió implementar el campo *Story* en la tabla Software de Ejecución de Test. Así se podía asociar cualquier *story* con los test relacionados.

Desde el departamento de QA, comprobando los resultados de test, se dieron cuenta de que comprobar uno por uno era lento y poco visual. Por ello se ha realizado el desarrollo de un *dashboard* en el que se muestran los resultados de los distintos test realizados. Posteriormente notaron que no se estaba aprovechando esta herramienta al 100 %, ya que solo podían ver los resultados generales,

así que se implementaron una serie de filtros interactivos para elegir sobre que compañías y testeos se enseñan los resultados de los gráficos.

Desde el departamento de QA querían poder añadir todas las pruebas creadas hasta ahora por el método antiguo a la nueva herramienta para tenerlas disponibles. Para ello se ha creado un importador en el que añadiendo un Excel genera todos los registros adecuados para poder testear con la herramienta *Test Management* sin tener que hacerlo manualmente.

Finalmente, desde el departamento de QA explicaron que seguían teniendo la necesidad de crear las hojas Excel, para poder compartir información con los clientes, pero que las tablas que existían no se adecuaban ni contenían los datos necesarios para que los clientes entendieran los procesos. Por ese motivo se crearon dos Database View, en las que se juntan varios registros relacionados con Test Management en uno solo, permitiendo la exportación de este.

4 Elaboración de los ciclos de vida de las entidades del proceso

En este apartado se muestran distintas figuras en las que se definen los ciclos de vida que tienen las distintas entidades del proceso y todos los estados por los que pueden pasar. No todas las entidades con las que se ha trabajado y que se han desarrollado en este proyecto tienen un ciclo de vida como tal, algunas se basan en un *checkbox* activo que determina si está activado y otras simplemente son un registro sin estado definido.

4.1 Ciclos de vida de Gestión de Proyectos

4.1.1 Ciclo de vida Proyecto

Para la entidad Proyecto, un registro puede estar en los estados:

- *Pending*: estado inicial del Proyecto, indica que está pendiente de aprobación.
- *Open*: estado cuando un Proyecto está aprobado, pero no ha empezado su desarrollo.
- *Work In Progress*: un Proyecto se está desarrollando.
- *Closed Complete*: el Proyecto ha terminado definitivamente.
- *Closed Incomplete*: el Proyecto ha terminado, pero han quedado cosas por hacer.
- *Closed Skipped*: el Proyecto ha terminado antes de tiempo.
- *Closed in Warranty*: el Proyecto ha terminado, pero está en garantía, y se arreglarán incidencias.

En esta entidad se puede pasar de cualquier estado a cualquier otro, por lo que no se puede definir un ciclo de vida como tal.

4.1.2 Ciclo de vida Tarea de Proyecto

Para la entidad Tarea de Proyecto, un registro puede estar en los estados:

- *Pending*: estado inicial de la Tarea de Proyecto, indica que está pendiente de aprobación.
- *Open*: estado cuando una Tarea de Proyecto está aprobado, pero no ha empezado su desarrollo.
- *Work In Progress*: una Tarea de Proyecto se está desarrollando.
- *Closed Complete*: la Tarea de Proyecto ha terminado definitivamente.
- *Closed Incomplete*: la Tarea de Proyecto ha terminado, pero han quedado cosas por hacer.
- *Closed Skipped*: la Tarea de Proyecto ha terminado antes de tiempo.
- *Closed in Warranty*: la Tarea de Proyecto ha terminado, pero está en garantía, y se arreglarán incidencias.

En esta entidad se puede pasar de cualquier estado a cualquier otro, por lo que no se puede definir un ciclo de vida como tal.

4.1.3 Ciclo de Vida Time Card

Para la entidad Time Card, un registro puede tener los siguientes estados:

- *Pending*: la Time Card está pendiente de aprobación para que se validen las horas imputadas.
- *Submitted*: la Time Card está completa y no se van a añadir más horas, queda pendiente de aprobación o de rechazo.
- *Approved*: la Time Card está aprobado y se populan las horas.

- *Rejected*: la *Time Card* queda rechazada.
- *Processed*: la *Time Card* se ha procesado y sus horas se han añadido a la Tarea de Proyecto correspondiente.

El ciclo de vida de la *Time Card* es el siguiente:

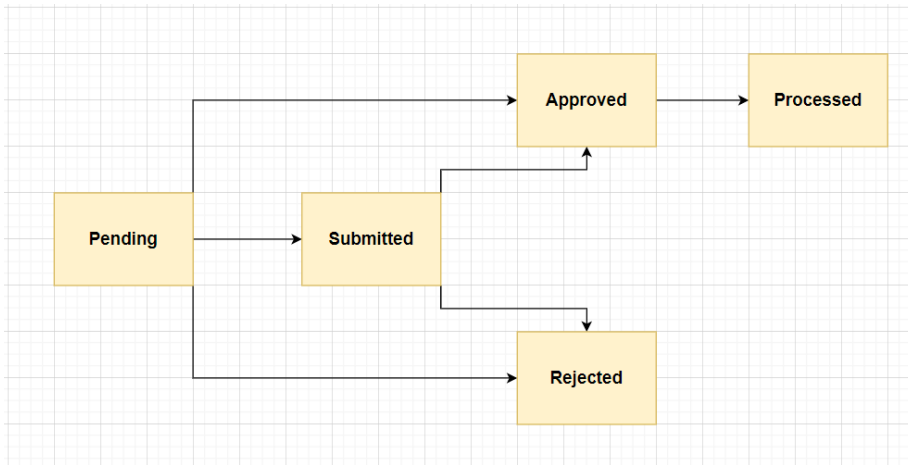


Figura 4.1.1 Ciclo de Vida Time Card

4.1.4 Ciclo de Vida Plan de Recursos

Para la entidad Plan de Recursos, un registro puede tener los siguientes estados:

- *Planning*: el Plan de Recursos está en proceso de planificación.
- *Requested*: se solicita a los encargados que se apruebe el Plan de Recurso.
- *Allocated*: los recursos solicitados por el proceso están asignados.
- *Completed*: el Plan de Recursos ha finalizado.
- *Cancelled*: se cancela el plan de recursos.

El ciclo de vida de la Plan de Recursos es el siguiente:

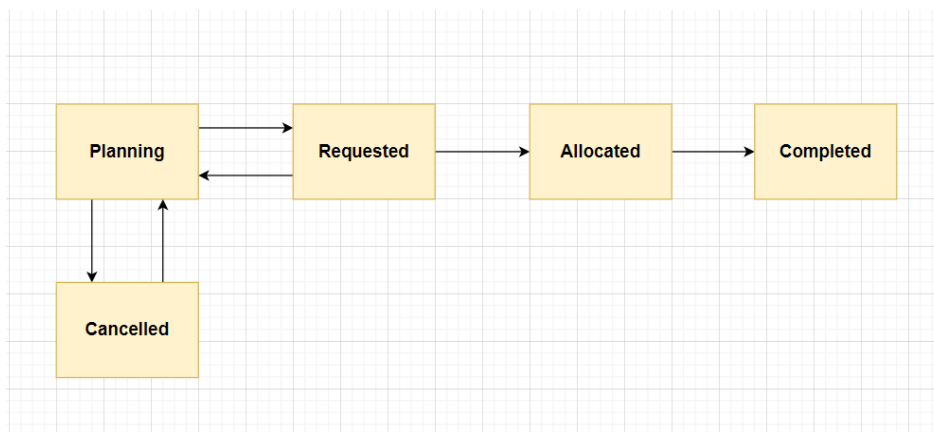


Figura 4.1.2 Ciclo de Vida Plan de Recursos

4.2 Ciclos de vida de *Test Management*

4.2.1 Ciclo de vida *Versión de Test*

Para la entidad *Versión de Test*, un registro puede estar en los estados:

- *Draft*: el registro todavía está en preparación.
- *Ready*: se están realizando *Testeos* basados en ese registro.
- *Retired*: el registro está ya retirado.

El ciclo de vida de la *Versión de Test* es el siguiente:

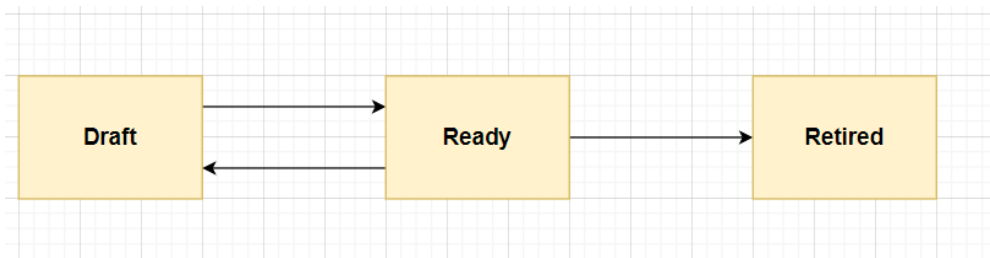


Figura 4.2.1 Ciclo de Vida de *Versión de Test*

Para la entidad *Test Run*, un registro puede estar en los estados:

- *In Progress*: el *Test Run* está en proceso.
- *Closed*: el *Test Run* está terminado.

4.2.2 Ciclo de vida de *Plan de Test*

Para la entidad *Plan de Test*, un registro puede estar en los estados:

- *Pending*: estado inicial del *Plan de Test*, indica que está pendiente de aprobación.
- *Open*: estado cuando un *Plan de Test* está aprobado, pero no ha empezado su desarrollo.
- *Work In Progress*: un *Plan de Test* se está desarrollando.
- *Closed Complete*: el *Plan de Test* ha terminado definitivamente.
- *Closed Incomplete*: el *Plan de Test* ha terminado, pero han quedado cosas por hacer.
- *Closed Skipped*: el *Plan de Test* ha terminado antes de tiempo.

En esta entidad se puede pasar de cualquier estado a cualquier otro, por lo que no se puede definir un ciclo de vida como tal.

4.2.3 Ciclo de Vida de *Ciclo de Test*

Para la entidad *Ciclo de Test*, un registro puede estar en los estados:

- *Pending*: estado inicial del *Ciclo de Test*, indica que está pendiente de aprobación.
- *Open*: estado cuando un *Ciclo de Test* está aprobado, pero no ha empezado su desarrollo.
- *Work In Progress*: un *Ciclo de Test* se está desarrollando.
- *Closed Complete*: el *Ciclo de Test* ha terminado definitivamente.
- *Closed Incomplete*: el *Ciclo de Test* ha terminado, pero han quedado cosas por hacer.
- *Closed Skipped*: el *Ciclo de Test* ha terminado antes de tiempo.

En esta entidad se puede pasar de cualquier estado a cualquier otro, por lo que no se puede definir un ciclo de vida como tal.

4.2.4 Ciclo de Vida de *Software de Ejecución de Test*

Para la entidad Proyecto, un registro puede estar en los estados:

- *Draft*: estado inicial del *Software de Ejecución de Test*, indica que está a la espera de planearse.
- *Planning*: estado cuando un *Software de Ejecución de Test* está planeándose.
- *Current*: un *Software de Ejecución de Test* se está desarrollando.
- *Complete*: el *Software de Ejecución de Test* ha terminado definitivamente.
- *Cancelled*: el *Software de Ejecución de Test* se ha cancelado.

En esta entidad se puede pasar de cualquier estado a cualquier otro, por lo que no se puede definir un ciclo de vida como tal.

5 Elementos técnicos de la aplicación

Dentro de la herramienta se utilizan distintos elementos técnicos que proporcionan la funcionalidad necesaria para poder implementar tanto los requisitos y objetivos de este trabajo de fin de grado como los necesarios para cualquier otro fin. Las distintas entidades con las que hemos contado para ello son:

- Campos: contienen los datos concretos que forman una entidad. Cada tabla o entidad tiene muchos de ellos. *ServiceNow* ofrece campos de muchos tipos. Para este trabajo estos son los tipos que han tenido relevancia [6].

Tipo	Contenido
<i>String</i>	Código alfanumérico
<i>Choice</i>	Selección de una opción entre distintas mostradas
<i>Reference</i>	Referencia a un registro, de otra tabla o de la misma
<i>Currency</i>	Valor monetario
<i>Integer</i>	Numero entero
<i>Decimal</i>	Numero decimal
<i>Percent Complete</i>	Porcentaje
<i>Date</i>	Fecha
<i>Datetime</i>	Fecha y hora
<i>Duration</i>	Registro que marca la duración, en distintos formatos de tiempo
<i>List</i>	Lista de registros asociados
<i>Checkbox</i>	Campo booleano
<i>URL</i>	URL
<i>True/false</i>	Campo booleano

Tabla 4.2.1 Tipos de Campo

Cada uno de los campos tiene asociados unas reglas de obligatoriedad (*mandatory*), editabilidad (*read only*) y visibilidad (*visibility*).

Los campos se crean asociados a unas tablas. Para acceder a las distintas tablas y crear los campos debemos usar el buscador del menú y buscar el módulo *System Definition > Tables*. En el podremos encontrar cualquier tabla que necesitemos configurar a través del *Label*.

Tras configurar los campos de la tabla, creamos el formulario con los campos necesarios que queramos mostrar, usando para ello la *Related Link* denominada *Form Layout* [7].

Para configurar las *Related list*, que son las listas de objetos relacionadas, se hace a través del botón *related list* [8].

La vista de los campos de un registro en la lista de registros de una tabla se configura a través del botón *Layout List*.

Para una misma entidad se pueden crear distintas vistas tanto para mostrar los campos en el formulario como para mostrar los campos en la lista. En el Trabajo de Fin de Grado se trabajará principalmente con las *Default View*. Algunas entidades pueden tener otras vistas cuando se crean los registros o según los roles que tenga el usuario.

- *UI Action*: son acciones que se llevan a cabo como respuesta a la acción de pulsar un botón en un formulario o una lista. Las *UI Actions* [8] pueden ser de distintos tipos y una misma *UI Action* puede ser de varios tipos a la vez. Los tipos son los siguientes:

Tipo	Descripción
<i>Form button</i>	Botón que aparece en el formulario
<i>Form Context Menu</i>	Opción que aparece en el menú desplegable de un formulario
<i>Form Link</i>	Botón que se muestra como un enlace en la parte de abajo del formulario
<i>List banner button</i>	Botón que aparece en el <i>banner</i> de la lista de registros de una tabla
<i>List bottom button</i>	Botón que aparece en la parte de debajo de la tabla que se muestra.
<i>List context menú</i>	Opción que aparece al seleccionar un registro de la tabla con el botón derecho
<i>List choice</i>	Opción que aparece en el menú desplegable situado debajo de una tabla.

Tabla 4.2.2 Tipos de *UI Action*

Hay una serie de *UI Actions* que son en general comunes a todas las tablas de registros y que se utilizan en este proyecto:

- *Submit*: *UI Action* de tipo *form button* que inserta un registro en el sistema y vuelve a la página cargada anteriormente.
- *Update*: *UI Action* de tipo *form button* que actualiza un registro en el sistema y vuelve a la página cargada anteriormente.
- *Delete*: *UI Action* de tipo *form button* que elimina un registro del sistema y vuelve a la página cargada anteriormente.
- *Cancel*: *UI Action* de tipo *form button* que cancela el proceso del registro en el que se activa.
- *Save*: *UI Action* de tipo *form context menú* que inserta o actualiza un registro del sistema y se mantiene en el registro.

- *Insert*: *UI Action* de tipo *form context menú* que inserta un registro en el sistema copiando los datos de un registro ya existente. Tras realizar la acción, la página se mantiene en el registro inicial.
 - *Insert and Stay*: *UI Action* de tipo *form context menú* que inserta un registro en el sistema copiando los datos de un registro ya existente. Tras realizar la acción, la página carga el nuevo registro.
 - *New*: *UI Action* de tipo *list banner button* que accede al formulario de creación de un registro de la lista en la que estaba el botón, para que se pueda rellenar.
- *Business Rule*: acciones que se producen en el servidor tras realizar una acción de inserción, actualización o borrado del registro de una tabla.
 - *Client Scripts*: acciones que se ejecutan en el cliente cuando se produce una acción de carga, cambio de algún campo o actualización de un registro de una tabla.
 - *Script Include*: registro que contiene distintas funciones y métodos que se ejecutan al ser llamadas desde otro.
 - *UI Policy*: registro de tipo cliente en el que se crean unas condiciones por las que se modifican las reglas asociadas a algún campo.
 - *UI Formatter*: es un elemento usado para mostrar información que no se encuentra en la propia entidad, y que se muestra a través del formulario.
 - *UI Page*: es un elemento que se usa para añadir una interfaz de usuario al formulario de una entidad. Contiene tres scripts para rellenar, el primero el HTML, en el que se programan en HTML los componentes de la interfaz, el *Client Script*, en el que se desarrolla en *JavaScript* los componentes de cliente y por último el *Processing Script*, donde se desarrollan, también en *JavaScript*, los componentes dedicados a servidor.
 - Menús y Módulos: estos permiten visualizar los distintos elementos y entidades que encontramos en la herramienta, concediéndonos el acceso a las mismas. El menú que nos ofrece esto se llama *Application Navigator*, y nos permite buscar por nombre los módulos a los que queremos acceder. También nos permite guardar favoritos y acceder a el historial de búsquedas y accesos que hemos realizado. Además, cuando accedemos a una lista o registro, el resto del contenido se divide en dos partes, una el *Banner Frame*, donde se muestran distintas opciones para que el usuario modifique los datos que se muestran (vistas, filtros) así como distintas *UI Actions*, y la otra el *Content Frame*, que muestra los datos requeridos.

5.1 Entidades del proceso de proyecto

En este apartado se explica el funcionamiento de cada entidad relacionada con el proceso de proyecto, centrándonos en los campos de relevancia de cara al Trabajo de Fin de Grado que tienen influencia en el mismo.

Para entender este proceso, resaltar antes la tabla *User (sys_user)* es la tabla que proporciona *ServiceNow* en la que se muestran los distintos usuarios disponibles, los cuales son transversales a toda la herramienta. En dicha tabla existe el siguiente campo:

- *Role*: rol asociado a la empresa *Silverstorm* en el que se define la categoría en la que se encuentra el empleado en cuestión. De cara a este proyecto los roles relevantes son los siguientes: *Deployment Consultant (DC)*, *Senior Deployment Consultant (SDC)*, *Program Delivery Management (PDM)* y *Business Process Consultant (BPC)*.

5.1.1 Diagrama de clases

A continuación, se muestra el diagrama de clases que relaciona las distintas entidades en las que se basa la aplicación. La entidad inicial es la de Proyecto, cada uno de ellos relacionado con una o más Tareas de Proyecto. Cada tarea de proyecto también se relaciona con uno o más Planes de Recursos. Las Tareas de Proyectos también se relacionan con un número indeterminado de *Time Cards*, las cuales se relacionan con un solo usuario, y cada usuario se puede asociar a un número indeterminado de *Time Cards*. Cada *Time Card* se relaciona también con uno o ningún Plan de Recursos (que se relaciona con un número indeterminado de *Time Cards*). El Plan de Recursos se relaciona con los usuarios en una relación de varios a varios, al igual que lo hace con las Asignaciones de Recursos, siendo esta relación del tipo uno a varios.

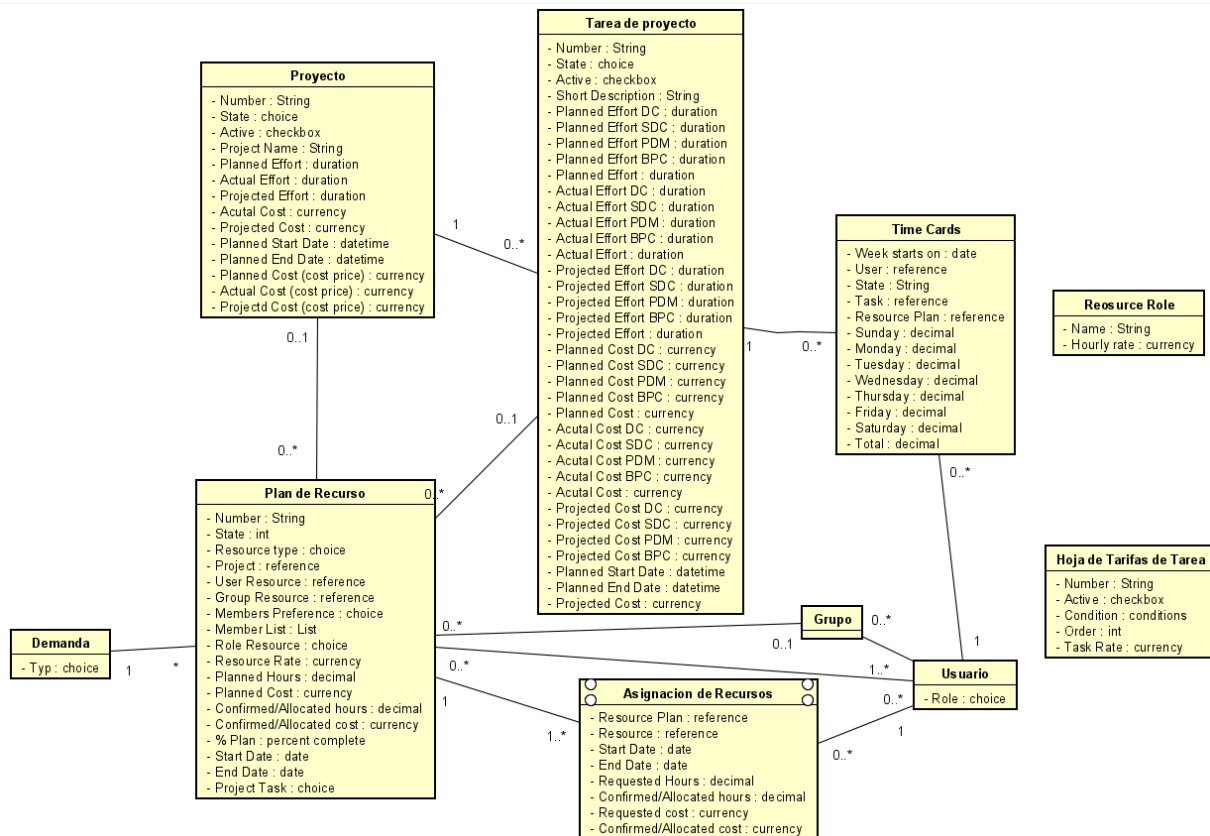


Figura 5.1.1 Diagrama de clases de Gestión de proyectos

5.1.2 Entidad Proyecto

El proyecto es la entidad en la que se muestran los datos generales sobre la realización de actividades relacionadas y ofrecerá una lista de las tareas de proyecto asociadas. Es una tabla que *ServiceNow* incluye directamente en sus servicios. Los campos asociados a esta tabla que a su vez tienen relevancia para la funcionalidad implementada son:

- **Number:** Autonumérico que proporciona *ServiceNow*, que se usa para identificar el número de registro dentro de la tabla.
- **State:** de tipo *choice* que muestra el estado actual en el que encuentra un proyecto. Las opciones son *Pending*, *Open*, *Work in progress*, *Closed Completed*, *Closed Incompleted*, *Closed Skipped*, *Closed in Warranty*.
- **Active:** de tipo *true/false* que marca si un proyecto está activo actualmente.
- **Project name:** *string* que contiene el nombre del proyecto.

- *Planned Effort*: campo de tipo *duration* que muestra el esfuerzo planeado a realizar por los usuarios en torno a un Proyecto. Es el resultado de la suma de todos los campos *Planned Effort* de cada tarea de proyecto asociada.
- *Actual effort*: campo de tipo *duration* que muestra el esfuerzo realizado actualmente para un Proyecto. Es el resultado de la suma de todos los campos *actual effort* de cada tarea de proyecto asociada.
- *Projected effort*: campo de tipo *duration* que muestra el esfuerzo previsto a realizar para un Proyecto. Es el resultado de la suma de todos los campos *Projected Effort* de cada tarea de proyecto asociada.
- *Actual Cost*: campo de tipo *currency* que muestra la inversión monetaria realizada hasta el momento en un Proyecto. Es la suma de los campos *Actual Cost* de cada tarea de proyecto asociada.
- *Projected Cost*: campo de tipo *currency* que muestra la inversión monetaria prevista a realizar en un Proyecto teniendo en cuenta la realizada hasta el momento. Es la suma de los campos *Projected Cost* de cada tarea de proyecto asociada.
- *Planned Start Date*: campo *Datetime*. Es la fecha en la que se prevé que empiece el Proyecto.
- *Planned End Date*: campo *Datetime*. Es la fecha en la que se prevé que termine el Proyecto.
- *Planned Cost (cost price)*: campo de tipo *currency* que muestra la suma de los campos *Planned Effort [Rol]* de cada Tarea de Proyecto asociada, multiplicando previamente cada una de estas por el precio que se indica en el registro de la entidad Recursos de Rol asociada a cada rol.
- *Actual Cost (cost price)*: campo de tipo *currency* que muestra la suma de los campos *Actual Effort [Rol]* de cada Tarea de Proyecto asociada, multiplicando previamente cada una de estas por el precio que se indica en el registro de la entidad Recursos de Rol asociada a cada rol.
- *Projected Cost (cost price)*: campo de tipo *currency* que muestra la suma de los campos *Projected Effort [Rol]* de cada Tarea de Proyecto asociada, multiplicando previamente cada una de estas por el precio que se indica en el registro de la entidad Recursos de Rol asociada a cada rol.

Esta tabla contiene una serie de listas relacionadas que muestran las siguientes relaciones:

- Tareas de proyectos: relación con las Tareas de Proyecto hijas del Proyecto.
- *Time Cards*: *Time Cards* asociados a cualquiera de las tareas de Proyecto, con las horas trabajadas en cada una ellas.
- Planes de recursos (Tarea de Proyecto): planes de recursos con los recursos previstos a asignar a las Tareas de Proyecto hijas de este Proyecto.

Además de las *UI Actions* nombradas como generales, la entidad proyecto tiene otras relacionadas exclusivamente para su uso:

- *Project Task Creator*: de tipo *context menú*, permite crear automáticamente el número de tareas de proyecto indicado, que se asociaran al proyecto.
- *Copy Project*: de tipo *context menú*, permite crear automáticamente una copia del proyecto actual, indicando un nuevo nombre.

- *Create Test Phase*: de tipo *form link*, permite crear una tarea de proyecto en la que se indican todos los datos referentes a ella.

5.1.3 Entidad Tarea de Proyecto

La tarea de proyecto son las subtareas en las que se divide un proyecto y que están asociadas al mismo. Cada tarea está asociado a un solo proyecto, pero cada proyecto puede estar asociado a distintas tareas. Los campos asociados a esta tabla que a su vez tienen relevancia para la funcionalidad son:

- *Number*: autonumérico que proporciona *ServiceNow*, que se usa para identificar el número de registro dentro de la tabla.
- *State*: de tipo *integer* que muestra el estado actual en el que encuentra la tarea de proyecto. Las opciones son *Pending*, *Open*, *Work in progress*, *Closed Completed*, *Closed Incompleted*, *Closed Skipped*, *Closed in Warranty*.
- *Short Description*: *string* que contiene una descripción de la Tarea de Proyecto.
- *Planned Effort DC*: campo de tipo *duration* que muestra el esfuerzo planeado a realizar por personas con el rol DC asociado en una Tarea de Proyecto.
- *Planned Effort SDC*: campo de tipo *duration* que muestra el esfuerzo planeado a realizar por personas con el rol SDC asociado en una Tarea de Proyecto.
- *Planned Effort PDM*: campo de tipo *duration* que muestra el esfuerzo planeado a realizar por personas con el rol PDM asociado en una Tarea de Proyecto.
- *Planned Effort BPC*: campo de tipo *duration* que muestra el esfuerzo planeado a realizar por personas con el rol BPC asociado en una Tarea de Proyecto.
- *Planned Effort*: campo de tipo *duration* que muestra el esfuerzo planeado a realizar por los usuarios en torno a una tarea de proyecto. Es el resultado de la suma de los campos *Planned Effort DC*, *Planned Effort SDC*, *Planned Effort PDM* y *Planned Effort BPC* de la Tarea de Proyecto.
- *Actual Effort DC*: campo de tipo *duration* que muestra el esfuerzo realizado actualmente por personas con el rol DC asociado en una tarea de proyecto. Es la suma de las horas totales invertidas de cada *Time Card* en estado *processed* con un usuario cuyo rol es DC.
- *Actual Effort SDC*: campo de tipo *duration* que muestra el esfuerzo realizado actualmente por personas con el rol SDC asociado en una tarea de proyecto. Es la suma de las horas totales invertidas de cada *Time Card* en estado *processed* con un usuario cuyo rol es SDC.
- *Actual Effort PDM*: campo de tipo *duration* que muestra el esfuerzo realizado actualmente por personas con el rol PDM asociado en una tarea de proyecto. Es la suma de las horas totales invertidas de cada *Time Card* en estado *processed* con un usuario cuyo rol es PDM.
- *Actual Effort BPC*: campo de tipo *duration* que muestra el esfuerzo realizado actualmente por personas con el rol BPC asociado en una tarea de proyecto. Es la suma de las horas totales invertidas de cada *Time Card* en estado *processed* con un usuario cuyo rol es BPC.
- *Actual effort*: campo de tipo *duration* que muestra el esfuerzo realizado actualmente para una tarea de proyecto. Es el resultado de la suma de los campos *Actual Effort DC*, *Actual Effort SDC*, *Actual Effort PDM* y *Actual Effort BPC* de la tarea de proyecto.
- *Projected Effort DC*: campo de tipo *duration* que muestra el esfuerzo que está previsto realizar por personas con el rol DC en una tarea de proyecto, teniendo en cuenta el realizado hasta el momento. Es la suma del *Actual Effort DC* y las horas previstas en las Asignaciones de Recursos activas para el rol DC asociados a los planes de recursos asociados a la tarea de

proyecto, a través del campo *Requested Hours* que contienen las Asignaciones de Recursos mencionadas.

- *Projected Effort SDC*: campo de tipo *duration* que muestra el esfuerzo que está previsto realizar por personas con el rol SDC en una tarea de proyecto, teniendo en cuenta el realizado hasta el momento. Es la suma del *Actual Effort SDC* y las horas previstas en las Asignaciones de Recursos activas para el rol SDC asociados a los planes de recursos asociados a la tarea de proyecto, a través del campo *Requested Hours* que contienen las Asignaciones de Recursos mencionadas.

- *Projected Effort PDM*: campo de tipo *duration* que muestra el esfuerzo que está previsto realizar por personas con el rol PDM en una tarea de proyecto teniendo en cuenta el realizado hasta el momento. Es la suma del *Actual Effort PDM* y las horas previstas en las Asignaciones de Recursos activas para el rol PDM asociados a los planes de recursos asociados a la tarea de proyecto, a través del campo *Requested Hours* que contienen las Asignaciones de Recursos mencionadas.

- *Projected Effort BPC*: campo de tipo *duration* que muestra el esfuerzo que está previsto realizar por personas con el rol BPC en una tarea de proyecto, teniendo en cuenta el realizado hasta el momento. Es la suma del *Actual Effort BPC* y las horas previstas en las Asignaciones de Recursos activas para el rol BPC asociados a los planes de recursos asociados a la tarea de proyecto, a través del campo *Requested Hours* que contienen las Asignaciones de Recursos mencionadas.

- *Projected Effort*: campo de tipo *duration* que muestra el esfuerzo que está previsto realizar para una determinada tarea de proyecto. Es la suma de cada campo *projected effort* asociado a cada rol en dicha tarea.

- *Planned Cost DC*: campo de tipo *currency* que muestra la inversión monetaria prevista a realizar en una tarea del proyecto por personas con el rol DC. Se calcula multiplicando el valor del campo *Planned Effort DC* por el coste de cada hora de este rol, que se encuentra en un registro de la tabla *Task Rate Cards*, en el campo *Task Rate*.

- *Planned Cost SDC*: campo de tipo *currency* que muestra la inversión monetaria prevista a realizar en una tarea del proyecto por personas con el rol SDC. Se calcula multiplicando el valor del campo *Planned Effort SDC* por el coste de cada hora de este rol, que se encuentra en un registro de la tabla *Task Rate Cards*, en el campo *Task Rate*.

- *Planned Cost PDM*: campo de tipo *currency* que muestra la inversión monetaria prevista a realizar en una tarea del proyecto por personas con el rol PDM. Se calcula multiplicando el valor del campo *Planned Effort PDM* por el coste de cada hora de este rol, que se encuentra en un registro de la tabla *Task Rate Cards*, en el campo *Task Rate*.

- *Planned Cost BPC*: campo de tipo *currency* que muestra la inversión monetaria prevista a realizar en una tarea del proyecto por personas con el rol BPC. Se calcula multiplicando el valor del campo *Planned Effort BPC* por el coste de cada hora de este rol, que se encuentra en un registro de la tabla *Task Rate Cards*, en el campo *Task Rate*.

- *Planned Cost*: campo de tipo *currency* que muestra la inversión monetaria total prevista a realizar en una tarea del proyecto. Se calcula sumando los campos *Planned Cost DC*, *Planned Cost SDC*, *Planned Cost PDM* y *Planned Cost BPC* de la tarea de proyecto.

- *Actual Cost DC*: campo de tipo *currency* que muestra la inversión monetaria realizada hasta el momento en una tarea del proyecto por personas con el rol DC. Se calcula multiplicando el valor del campo *Actual Effort DC* por el coste de cada hora de este rol, que se encuentra en un registro de la tabla *Task Rate Cards*, en el campo *Task Rate*.

- *Actual Cost SDC*: campo de tipo *currency* que muestra la inversión monetaria realizada hasta el momento en una tarea del proyecto por personas con el rol SDC. Se calcula multiplicando el valor del campo *Actual Effort SDC* por el coste de cada hora de este rol, que se encuentra en un registro de la tabla *Task Rate Cards*, en el campo *Task Rate*.
- *Actual Cost PDM*: campo de tipo *currency* que muestra la inversión monetaria realizada hasta el momento en una tarea del proyecto por personas con el rol PDM. Se calcula multiplicando el valor del campo *Actual Effort PDM* por el coste de cada hora de este rol, que se encuentra en un registro de la tabla *Task Rate Cards*, en el campo *Task Rate*.
- *Actual Cost BPC*: campo de tipo *currency* que muestra la inversión monetaria realizada hasta el momento en una tarea del proyecto por personas con el rol BPC. Se calcula multiplicando el valor del campo *Actual Effort BPC* por el coste de cada hora de este rol, que se encuentra en un registro de la tabla *Task Rate Cards*, en el campo *Task Rate*.
- *Actual Cost*: campo de tipo *currency* que muestra la inversión monetaria realizada hasta el momento en un proyecto. Se calcula sumando los campos *Actual Cost DC*, *Actual Cost SDC*, *Actual Cost PDM* y *Actual Cost BPC* de la tarea de proyecto.
- *Projected Cost DC*: campo de tipo *currency* que muestra la inversión monetaria prevista a realizar en una tarea del proyecto por personas con el rol DC, teniendo en cuenta la realizada hasta el momento. Se calcula sumando el valor del campo *Actual Effort DC* y el coste previsto en las Asignaciones de Recursos activas para el rol DC asociados a los planes de recursos asociados a la tarea de proyecto, a través del campo *Requested Cost* que contienen las Asignaciones de Recursos mencionadas.
- *Projected Cost SDC*: campo de tipo *currency* que muestra la inversión monetaria prevista a realizar en una tarea del proyecto por personas con el rol SDC, teniendo en cuenta la realizada hasta el momento. Se calcula sumando el valor del campo *Actual Effort SDC* y el coste previsto en las Asignaciones de Recursos activas para el rol SDC asociados a los planes de recursos asociados a la tarea de proyecto, a través del campo *Requested Cost* que contienen las Asignaciones de Recursos mencionadas.
- *Projected Cost PDM*: campo de tipo *currency* que muestra la inversión monetaria prevista a realizar en una tarea del proyecto por personas con el rol PDM, teniendo en cuenta la realizada hasta el momento. Se calcula sumando el valor del campo *Actual Effort PDM* y el coste previsto en las Asignaciones de Recursos activas para el rol PDM asociados a los planes de recursos asociados a la tarea de proyecto, a través del campo *Requested Cost* que contienen las Asignaciones de Recursos mencionadas.
- *Projected Cost BPC*: campo de tipo *currency* que muestra la inversión monetaria prevista a realizar en una tarea del proyecto por personas con el rol BPC, teniendo en cuenta la realizada hasta el momento. Se calcula sumando el valor del campo *Actual Effort BPC* y el coste previsto en las Asignaciones de Recursos activas para el rol BPC asociados a los planes de recursos asociados a la tarea de proyecto, a través del campo *Requested Cost* que contienen las Asignaciones de Recursos mencionadas.
- *Projected Cost*: campo de tipo *currency* que muestra la inversión monetaria total que está previsto realizar para una determinada tarea de proyecto. Es la suma de cada campo *projected cost* asociado a cada rol en dicha tarea.
- *Planned Start Date*: campo *Datetime*. Es la fecha en la que se prevé que empieza el proyecto.

- *Planned End Date*: campo *Datetime*. Es la fecha en la que se prevé que termine el proyecto.

Esta tabla contiene una serie de listas relacionadas que muestran las siguientes relaciones:

- Tareas de proyectos: relación con el resto de las tareas de proyecto hijas de la propia tarea de proyecto.
- *Time Cards*: *Time Cards* asociados a la tarea de proyecto, con las horas trabajadas en el mismo.
- Planes de recursos: planes de recursos con los recursos previstos a asignar a la tarea de proyecto.

Además de las *UI Actions* nombradas como generales, la entidad tarea de proyecto tiene otras relacionadas exclusivamente para su uso:

- *Project Task Creator*: de tipo *context menú*, permite crear automáticamente el número de tareas de proyecto indicado, que se asociaran a la tarea de proyecto.
- *Redirect to the top task*: de tipo *related list*, carga automáticamente el formulario del proyecto del que parte todas las tareas de proyecto en la que estas se encuentran.

5.1.4 Entidad *Time Card*

Esta entidad permite a un usuario indicar las horas trabajadas en una tarea de proyecto durante una semana concreta, indicando el día o días concretos en las que se han trabajado. Los campos asociados a esta tabla que a su vez tienen relevancia para la funcionalidad son:

- *Week starts on*: campo de tipo *date*, marca en que semana se han realizado las horas trabajadas.
- *User*: campo referencia a la tabla *User*, indica el usuario que hay realizado las horas trabajadas.
- *State*: campo de tipo *string* que indica el estado en el que esta la *Time Card* actualmente. Las opciones son *Pending*, *Submitted*, *Rejected*, *Approved*, *Processed* y *Recalled*.
- *Task*: campo de tipo referencia a la tabla *Task*, donde se guarda la tarea de proyecto en la que se imputan las horas trabajadas.
- *Resource plan*: campo referencia a la tabla Planes de Recursos, donde se muestra el plan de recursos asociado tanto a la tarea de proyecto del campo *task* como al usuario del campo *user*, en caso de que haya un plan de recursos asociado a ambas. Si no, el campo queda vacío.
- *Sunday*: campo tipo decimal, que indica las horas trabajadas el domingo.
- *Monday*: campo tipo decimal, que indica las horas trabajadas el lunes.
- *Tuesday*: campo tipo decimal, que indica las horas trabajadas el martes.
- *Wednesday*: campo tipo decimal, que indica las horas trabajadas el miércoles.
- *Thursday*: campo tipo decimal, que indica las horas trabajadas el jueves.
- *Friday*: campo tipo decimal, que indica las horas trabajadas el viernes.
- *Saturday*: campo tipo decimal, que indica las horas trabajadas el sábado.
- *Total*: campo tipo decimal, que indica las horas trabadas en total durante la semana

Además de las *UI Actions* nombradas como generales, la entidad *Time Card* tiene otras relacionadas exclusivamente para su uso:

- *Submit Time Card*: de tipo *form button*, actualiza el valor del campo *state* del registro a *Submitted*.
- *Approve*: de tipo *form button*, actualiza el valor del campo *state* del registro a *Approve* y autoriza a validar las horas indicadas de cara a la tarea de proyecto asociada. Estas horas se añaden a los campos correspondientes de dicha tarea.
- *Reject*: de tipo *form button*, actualiza el valor del campo *state* del registro a *Rejected* y rechaza validar las horas indicadas de cara a la tarea de proyecto asociada.

5.1.5 Entidad Plan de Recurso

Esta entidad contiene el plan de recursos que van a dedicar a un proyecto o a una tarea de proyecto en concreto. También se pueden crear asociados a una demanda. Los campos asociados a esta tabla que a su vez tienen relevancia para la funcionalidad son:

- *Number*: autonumérico que proporciona *ServiceNow*, que se usa para identificar el número de registro dentro de la tabla.
- *State*: campo de tipo *integer* que muestra el estado actual en el que se encuentra el plan de recurso. Los estados son *Planning*, *Requested*, *Allocated*, *Work in Progress*, *Completed*, *Cancelled* y *Confirmed*. Se considera que un Plan de Recursos está activo cuando su estado no es ni *Completed* ni *Cancelled*.
- *Project*: campo de tipo *reference* a la tabla *Task* que puede contener un proyecto, a una tarea de proyecto o a una demanda.
- *Project Task*: campo de tipo *choice* contiene las tareas de proyecto predeterminadas que suele contener un proyecto. Solo aparece cuando el campo *project* contiene un registro de tipo demanda.
- *Resource Type*: campo de tipo *choice* que marca a qué tipo de recurso se dedica el plan. Tiene tres opciones: *User Resource*, *Role Resource* y *Group Resource*.
- *User Resource*: campo referencia a la tabla *user* que contiene el usuario al que se destina el plan de recurso. Solo se muestra cuando el campo *Resource Type* tiene valor *User Resource*.
- *Group Resource*: campo referencia a la tabla *group* que contiene al grupo al que se destina el plan de recurso. Solo se muestra cuando el campo *Resource Type* tiene valor *Group Resource*.
- *Members Preference*: campo de tipo *choice* que solo se muestra cuando el campo *Resource Type* tiene valor *Group Resource*. Si el campo es *Any Member*, aplicará a todos los usuarios agregados al grupo, mientras que, si el campo es *Specific Member*, solo lo hará a los usuarios seleccionados en el campo *Member List*.
- *Member List*: campo de tipo *list* donde se especifican los usuarios que pertenecen al grupo indicado en el campo *Group Resource* para los que se va a aplicar el plan de recursos.
- *Role Resource*: campo de tipo *String* donde se especifica el rol para el que está destinado el plan de recursos. Solo se muestra si el campo *Resource Type* es *Role Resource*.
- *Resource Rate*: campo de tipo *currency* que indica el precio previsto por cada hora trabajada para ese Plan de Recursos.
- *Planned Hours*: campo de tipo decimal donde se indican las horas previstas a trabajar en el Plan de Recursos.
- *Planned Cost*: campo de tipo *currency* en el que se muestra el coste completo previsto para ese plan de recursos. Se obtiene multiplicando el valor del campo *Resource Rate* por el valor del campo *Planned Hours*.

- *Confirmed/Allocated hours*: campo de tipo decimal que indica las horas concedidas para el usuario para ese plan de recursos.
- *Confirmed/Allocated cost*: campo de tipo *currency* que indica el coste concedido por el usuario para ese plan de recursos.
- *% Plan*: campo de tipo *percent complete* que indica el porcentaje del plan de recursos que está previsto realizar.
- *Start Date*: campo *Date*. Es la fecha en la que se empieza el Plan de Recursos.
- *End Date*: campo *Date*. Es la fecha en la que termina el Plan de Recursos.

Esta tabla contiene una serie de listas relacionadas que muestran las siguientes relaciones:

- *Asignaciones de recursos*: las asignaciones de recursos dividen las horas planeadas entre las distintas semanas que durará el proyecto, teniendo en cuenta los campos *Start Date* y el *End Date*. Un plan de recursos tiene 1 o varias asignaciones de recursos, y las asignaciones de recursos tienen un solo plan de recursos.
- *Asignaciones requeridas*: se muestra solo cuando en el campo *project* del Plan de Recursos se asocia a una Demanda. Muestra las asignaciones requeridas para que el plan de recursos se lleve a cabo.

Además de las *UI Actions* nombradas como generales, la entidad Plan de Recursos tiene otras relacionadas exclusivamente para su uso:

- *Create Soft Allocation*: de tipo *form button*, genera automáticamente las entidades asignación de recursos que se asocian a un plan de recursos, dividiendo para ello las horas dedicadas entre las semanas que estará abierto el plan de recursos.
- *Request*: de tipo *form button*, *list context menu* y *list choice*, actualiza el valor del campo *state* del registro a *Requested*.
- *Reject*: de tipo *form button*, *list context menu* y *list choice*, actualiza el valor del campo *state* del registro a *Planning*, y rechaza el mismo para que se vuelvan a rellenar los datos de este.
- *Confirm and Allocate*: de tipo *form button*, *list context menu* y *list choice*, actualiza el valor del campo *state* del registro a *Allocated*, y confirma la validez de este. Asigna los recursos a los usuarios correspondientes.
- *Complete*: de tipo *form button*, actualiza el valor del campo *state* del registro a *Completed*, y da por cerrado en plan de recursos correspondiente.
- *Allocate*: de tipo *form button*, *list bottom button* y *list context menú*, asigna los planes de recursos seleccionados a los usuarios correspondientes.
- *Confirm*: de tipo *list context menú* y *list choice*, confirma que un plan de recursos es válido y será ejecutado por los usuarios.
- *Cancel*: de tipo *UI Action*, pasa el Plan de Recursos al *state Cancelled*.

5.1.6 Entidad Asignación de Recursos

Las asignaciones de recursos son el reparto semanal de recursos, tanto en horas como en coste para cada usuario. Los campos asociados a esta tabla que a su vez tienen relevancia para la funcionalidad son:

- *Resource Plan*: campo de tipo referencia a la tabla Planes de Recursos que muestra el plan de recursos asociado a la asignación de recursos.

- *Resource*: campo de tipo referencia a la tabla *user* que muestra el usuario para el que se ha generado la asignación de recursos.
- *Start Date*: campo *Date*. Es la fecha en la que se empieza dicha asignación de recursos.
- *End Date*: campo *Date*. Es la fecha en la que termina dicha asignación de recursos.
- *Requested hours*: campo de tipo decimal que indica las horas requeridas por el usuario para esa asignación de recursos.
- *Confirmed/Allocated hours*: campo de tipo decimal que indica las horas concedidas para el usuario para esa asignación de recursos.
- *Requested cost*: campo de tipo *currency* que indica el coste requerido por el usuario para esa asignación de recursos.
- *Confirmed/Allocated cost*: campo de tipo *currency* que indica el coste concedido por el usuario para esa asignación de recursos.

Las entidades Asignación de Recursos se crean todas asignadas un Plan de Recursos mediante *la UI Action Create Soft Allocation* previamente mencionada. Cuando se pulsa se crea un registro de esta tabla por semana que abarca el periodo entre el campo *start date* y el campo *end date* del Plan de Recursos. Para rellenar los campos de horas y coste, se divide las asignadas al Plan de Recursos entre el número de registros de Asignación de Recursos de forma equitativa, teniendo en cuenta cuantos días laborables comprende el periodo de la Asignación de Recursos. Todos estos campos son editables una vez el registro se ha creado, para dar libertad a los gestores del proyecto a la hora de asignarlos.

5.1.7 Entidad Hoja de Tarifas de Tarea

En esta entidad se define el coste que tiene cada hora invertida por un determinado rol en un determinado proyecto. Los campos asociados a esta tabla que a su vez tienen relevancia en el proyecto son:

- *Number*: autonumérico que proporciona *ServiceNow*, que se usa para identificar el número de registro dentro de la tabla.
- *Active*: campo de tipo *checkbox* que indica si la hoja de tarifas de la tarea esta activa.
- *Condition*: campo de tipo *condition* donde se especifica cuando se debe aplicar esta hoja de tarifa de tarea, tanto por proyecto a tarea de proyecto, como por rol.
- *Order*: campo de tipo *integer* que delimita que hoja de tarifa de tarea tiene prioridad en caso de que dos de estas cumplan las condiciones.
- *Task Rate*: campo de tipo *currency* que indica cuanto coste tiene por hora el proyecto y roles especificados en el campo *condition*.

5.1.8 Entidad Demanda

Esta entidad permite crear demandas de servicio a partir de las cuales se crean proyectos. En este Trabajo de Fin de Grado esta tabla tiene relevancia a la hora de relacionarlas con la tabla Planes de Recursos, ya que estos se pueden crear asociados a demanda a través del campo *project*. Para este proyecto ha sido relevante el siguiente campo:

- *Type*: de tipo *choice*, indica el tipo de la Demanda. Puede ser de tipo *Project*, *Service Improvement* o *Enhancement Services*.

Para hacer esto, esta entidad cuenta con la *related list*:

- Planes de recursos: planes de recursos con los recursos previstos a asignar a la demanda.

Los Planes de Recursos asociados a las Demandas tienen el valor del campo *resource type* como *Role* y convierte es campo en no editable. También muestra el campo *Project Task*, que nos muestra unas opciones u otras dependiendo del campo.

5.1.9 Entidad Recursos de Rol

Esta entidad indica el coste por hora trabajada asociado a cada rol de forma genérica, sin tener en cuenta el Proyecto concreto en el que se trabaja. Los campos asociados a esta tabla que a su vez tienen relevancia en el proyecto son:

- Name: de tipo *string*, indica el rol al que corresponde el registro.
- Hourly Rate: de tipo *currency* indica el coste por hora trabajada para ese rol.

5.1.10 Menús y Módulos relacionados con Gestión de Proyectos

Los menús y módulos por los que se puede acceder a las entidades de *Test* son los siguientes:

Project:

- *Project*: modulo separador que contiene módulos relacionados con *Project*.
 - *Rolled Up Report*: muestre un *dashboard* que contiene todos los proyectos y en el que se pueden ver distintos campos según esfuerzo requerido, duración, fecha...
 - *Project Workspace*: muestra un *dashboard* que contiene todos los proyectos activos los cuales tienen el campo *parent* vacío.
 - *Work In Progress*: muestra una lista que contiene todos los proyectos activos (el campo *active* tiene valor *true*).
 - *All*: muestra una lista que contiene todos los proyectos.
 - *Import*: permite importar un proyecto desde fuera de la herramienta.
- *Project Task*: modulo separador que contiene módulos relacionados con *Project Task*.
 - *Work In Progress*: muestra una lista que contiene todas las tareas de proyectos activas (el campo *active* tiene valor *true*).
 - *All*: muestra una lista que contiene todas las tareas de proyectos.

SilverStorm Implementations:

- *Rates*: modulo separador que contiene módulos relacionados con *Rates*.
 - *Task Rate Cards*: muestra una lista que contiene todas las Hoja de Tarifas de Tarea.
- *Resource Management*: modulo separador que contiene módulos relacionados con recursos.
 - *Resource Plans*: muestra una lista que contiene todos los planes de recursos.

5.2 Entidades del proceso de *Test Management*:

El proceso de gestión de pruebas es un proceso con el cual se prevé llevar un control de las pruebas que se realizan en los distintos desarrollos que se llevan a cabo en un proceso concreto. Este proceso permite gestionar las pruebas en grupos, guardar resultados y mostrar el progreso general en las distintas pruebas realizadas:

5.2.1 Diagrama de clases

En el proceso de gestión de pruebas nos encontramos las entidades que definen los *Test* a realizar para comprobar que un proceso funciona correctamente. Un Conjunto de *Test* es una entidad que está formado por un número indeterminado de *Test* con los que se relaciona mediante la clase asociativa Conjunto de *Test*. El *Test* a su vez este compuesto por un número indeterminado de Versiones de *Test*, cada una con un número de Pasos correspondiente, también indeterminado. Además, cada Versión de *Test* se relaciona con los Resultados de *Test* propios de la misma. Cada uno de estos resultados se asocia con un *Test Run*. Por otro lado, están las entidades que definen los plazos y las duraciones de los distintos *Test*. En este caso, la entidad Plan de *Test* nos muestra los datos generales, y se compone

de un número indeterminado de ciclos de *Test*, que a su vez se divide en un número indeterminado de software de ejecución de *Test*. Ambas partes se relacionan mediante la entidad Asignación de Ejecución de *Test*, la cual se relaciona con un *Test* y con un Software de ejecución de *Test*.

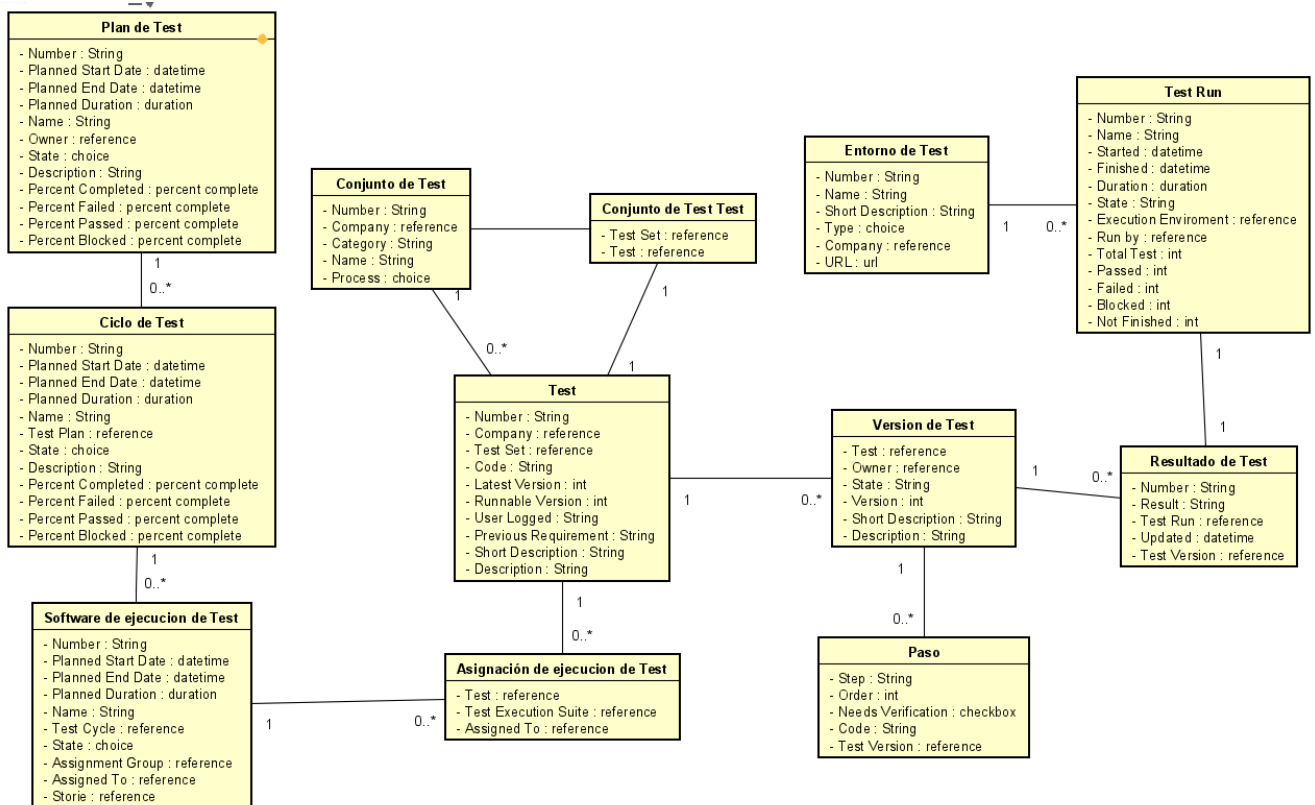


Tabla 5.2.1 Diagrama de clases de Test Management

5.2.2 Entidad Conjunto de Test

El conjunto de *Test* es una entidad cuya función es definir los datos generales de un grupo de *Test* que se tienen unas características en común. Los campos asociados a esta tabla que a su vez tienen relevancia en el proyecto son:

- **Number:** autonumérico que proporciona *ServiceNow*, que se usa para identificar el número de registro dentro de la tabla.
- **Company:** campo de tipo referencia a la tabla *Company*, que indica a que compañía están asociados los *Test*.
- **Process:** campo de tipo *choice* que indica el proceso de ejecución al que pertenece el Conjunto de *Test*.
- **Owner:** campo de tipo referencia a la tabla *User*, que indica que usuario es el responsable del Conjunto de *Test*
- **Category:** campo de tipo *string* que indica a que tipo pertenecen los *Test*.
- **Name:** campo de tipo *string* que indica el nombre que se le da al conjunto de *Test*

Esta tabla contiene una serie de listas relacionadas que muestran las siguientes relaciones:

- **Test Set Test:** tabla asociativa que relaciona un conjunto de *Test* con un *Test* concreto.

Además de las *UI Actions* nombradas como generales, la entidad Conjunto de *Test* tiene otras relacionadas exclusivamente para su uso:

- **Add to Execution Suite:** de tipo *list banner button*, solo se muestra cuando se ha accedido a la tabla *Test Set* desde el botón *Add Test Set* de un registro software de ejecución de *Test*. Los

Test que forman los Conjuntos de *Test* seleccionados se añadirán automáticamente al software de ejecución de *Test* desde el cual se accedió.

- *Update Code Test*: de tipo *form button*, dispara una interfaz a través de una UI Page para gestionar los campos *code* de las entidades *Test* asociadas[10]. La ventana muestra tres campos:

- *Prefix*: muestra el prefijo que tendrá el código que vamos a generar. Es obligatorio.
- *Number o digits*: determina el número de dígitos con el que va a contar el código. Es obligatorio y tiene que ser mayor que 0.
- *Initial Number*: indica el número en el que empezará el código. El campo puede estar vacío o ser un número entero mayor de 0.

Si se aceptan los datos, se generan automáticamente todos los *Test*.

5.2.3 Entidad Conjunto de *Test Test*

Entidad asociativa cuyo único objetivo es relacionar a los conjuntos de *Test* con los *Test* concretos. Los campos asociados a esta tabla que a su vez tienen relevancia en el proyecto son:

- *Test Set*: campo de tipo referencia a *Test Set*, que muestra el conjunto de *Test* asociado. Un conjunto de *Test* puede aparecer en cualquier número de registros en esta tabla.
- *Test*: campo de tipo referencia a *Test*, que muestra el *Test* asociado. Un *Test* puede aparecer en tan solo un registro en esta tabla.

Además de las *UI Actions* nombradas como generales, la entidad Conjunto de *Test Test* tiene otras relacionadas exclusivamente para su uso:

- *New*: de tipo *list banner button*, este *New* es distinto del resto, ya que no crea un registro en la tabla de Conjunto de *Test Test*, sino que nos muestra el formulario de Versión de *Test*, a través del cual se generaran los registros en ambas tablas y en la tabla *Test*. Se asocian al *Test* el *Test Set* y su *company*.

5.2.4 Entidad *Test*

La entidad *Test* muestra una prueba concreta que se realiza dentro un conjunto de *Test*. Cuando se crea una entidad *Test* no se crea directamente, sino que se crea una Versión de *Test* que mapea automáticamente los datos necesarios para generar el *Test*. Los campos asociados a esta tabla que a su vez tienen relevancia en el proyecto son:

- *Number*: autonumérico que proporciona *ServiceNow*, que se usa para identificar el número de registro dentro de la tabla.
- *Company*: campo de tipo referencia a la tabla *Company*, que indica a que compañía están asociados los *Test*.
- *Test Set*: campo de tipo referencia a la tabla *Test Set Test*, donde se asocia con el conjunto de *Test* correspondiente. Solo se puede asociar un *Test* con un conjunto de *Test* con el mismo valor para el campo *Company*.
- *Code*: campo de tipo *string* que muestra un código asociado al set.
- *Test Version*: campo de tipo referencia a la tabla *Test Version*, que indica cual es el la última Versión de *Test* que se ha creado para ese *Test*.
- *Runnable Version*: campo de tipo referencia a la tabla *Test Version*, que indica cual es el la Versión de *Test* que esta activa actualmente para ese *Test*.

- *User logged*: campo de tipo *string* que indica el usuario con el que se ha realizado un *Test*.
- *Previous requirements*: campo de tipo *string* que indica los requisitos previos a la realización de una prueba.
- *Short Descripción*: campo de tipo *string* que proporciona un breve resumen sobre el *Test*.
- *Description*: campo de tipo *string* que proporciona un resumen sobre el *Test*.

Esta tabla contiene una serie de listas relacionadas que muestran las siguientes relaciones:

- *Version Test*: versiones asociadas al mismo *Test* en las que se pueden ejecutar las pruebas.

Además de las *UI Actions* nombradas como generales, la entidad *Test* tiene otras relacionadas exclusivamente para su uso:

- *Add to Test Set*: de tipo *list banner button*, abre una UI page en la que se muestra un solo campo, *Test Set*, que hace referencia a dicha tabla. El *Test Set* seleccionado se mapea con el campo *Test Set* de los registros *Test* que se habían seleccionado, y crea en la tabla *Test Set* el registro que asocia ambos[11].
- *Add to Execution Suite*: de tipo *list banner button*, solo se muestra cuando se ha accedido a la tabla *Test* desde el botón *Add Test* de un registro software de ejecución de *Test*. Los *Test* seleccionados se añadirán automáticamente al software de ejecución de *Test* desde el cual se accedió.
- *New*: de tipo *list banner button*, este *button New* es distinto del resto, ya que no crea un registro en la tabla *Test*, sino que nos muestra el formulario de Versión de *Test*, a través del cual se generaran los registros.

Esta entidad tiene deshabilitados las *UI Action Insert* y *Insert and Stay*.

5.2.5 Entidad Versión de *Test*

La entidad Versión de *Test* es una variante de un *Test* concreto, que se puede ejecutar para ver el resultado de las pruebas. Los campos asociados a esta tabla que a su vez tienen relevancia en el proyecto son:

- *Test*: campo de tipo referencia a la tabla *Test*, que muestra el *Test* al que pertenece la versión.
- *Owner*: campo de tipo referencia a la tabla *User*, que muestra el usuario al que pertenece la versión del *Test*.
- *State*: campo de tipo *string* que muestra el estado en el que se encuentra la Versión de *Test*. Las opciones son *Draft*, *Ready* o *Retired*.
- *Version*: campo de tipo *integer* que muestra el número de versión con respecto al *Test* al que está asociada.
- *Short Description*: campo de tipo *string* que proporciona un breve resumen sobre la versión de *Test*.

Además, la tabla contiene varios campos gemelos de los campos de *Test* cuyo objetivo es mapear datos de un registro a otro.

Esta tabla contiene una serie de listas relacionadas que muestran las siguientes relaciones:

- *Other Version*: muestra el resto de las versiones que están asociadas al mismo *Test* que la versión actual.
- *Steps*: muestra los pasos a realizar para cumplimentar el *Test* asociado.
- *Test Results*: resultado de los *Test* realizados para la versión de *Test*.

Por otro lado, en esta tabla se muestra un *UI Formatter*, denominado *Test Version Steps*, el cual muestra a través de una UI Macro del mismo nombre los *Steps* relacionados con la versión de *Test*, y permite declarar cuales de ellos necesitan verificación. Solo se pueden añadir nuevos pasos a la versión de *Test* cuando esta se encuentra en estado *Draft*.

Además de las *UI Actions* nombradas como generales, la entidad Versión de *Test* tiene otras relacionadas exclusivamente para su uso:

- *Run*: de tipo *form button*, abre una ventana emergente en la que se permite guardar los datos de todos los pasos que se realizan en un *Test*, y guarda el resultado creando un registro *Test Result* asociado.
- *Ready*: de tipo *form button*, cambia el estado *Draft* de una versión de *Test* al estado *Ready*. En el momento que una versión de *Test* pasa a *Ready*, cualquier versión anterior del *Test* que estuviera en ese estado para a estar en estado *Retired*. Además, no se puede pasar a *Ready* ningún registro que no tenga al menos un Paso con el campo *needs verification* en true.
- *Edit Version*: de tipo *form button*, cambia el estado *Ready* de una versión de *Test* al estado *Draft*, para poder modificar la Versión de *Test* sin necesidad de crear una nueva.
- *Create New Version*: de tipo *form button*, crea un nuevo registro con los mismos datos que el registro origen, excepto el campo *state*, que será *Draft*, y el campo *Version*, cuyo valor será el número entera siguiente a la última versión creada para ese *Test*.
- *Delete Test*: de tipo *form button*, elimina el registro *Test* padre de la versión de *Test*, y todos los registros asociados.

Esta entidad tiene deshabilitados las *UI Action Insert* y *Insert and Stay*.

5.2.6 Entidad Paso

La entidad Paso muestra todos los pasos que se han de realizar para completar un *Test* de manera individual. Los campos asociados a esta tabla que a su vez tienen relevancia en el proyecto son:

- *Step*: campo de tipo *string* que describe el paso a realizar.
- *Order*: campo de tipo *integer* que define el orden de los pasos, teniendo en cuenta el *Order* del resto y empezando por el más bajo.
- *Needs Verification*: campo de tipo *checkbox* que marco si un paso concreto necesita verificación.
- *Code*: campo de tipo *string* que indica el código del paso
- *Test Version*: campo de tipo referencia a la tabla *Test Version* que relaciona el paso con la versión de *Test* a la que pertenece.

5.2.7 Entidad Resultado de Test

La entidad Resultado de *Test* muestra los resultados de cada *Test* realizado por los usuarios. Los campos asociados a esta tabla que a su vez tienen relevancia en el proyecto son:

- *Number*: autonumérico que proporciona *ServiceNow*, que se usa para identificar el número de registro dentro de la tabla.
- *Result*: campo de tipo *string* que muestra en resultado de la última prueba realizada para ese *Test*. Los resultados posibles son *Passed*, *Blocked*, *Failed* y *Not Finished*.
- *Updated*: campo de tipo *datetime* que muestra el momento de ejecución del *Test*.
- *Test Run*: campo de tipo referencia a la tabla *Test Run* que muestra el registro que contiene los datos sobre la ejecución del *Test*.

- *Test Version*: campo de tipo referencia a la tabla *Test Version* que relaciona el resultado de *Test* con la versión de *Test* a la que pertenece.

Por otro lado, en esta tabla se muestra un UI Formatter, denominado *Test Step*, el cual muestra a través de una *UI Macro* del mismo nombre el resultado concreto de cada paso que se ha realizado en la ejecución del *Test*. En los campos que necesitan verificación, marca los resultados de este, así como muestra los comentarios añadidos por el usuario y los archivos adjuntos.

5.2.8 Entidad *Test Run*

La entidad *Test Run* muestra los datos del resultado de la ejecución de un *Test* en concreto. Los campos asociados a esta tabla que a su vez tienen relevancia en el proyecto son:

- *Number*: autonumérico que proporciona *ServiceNow*, que se usa para identificar el número de registro dentro de la tabla.
- *Name*: campo de tipo *string* que indica el nombre del funcionamiento de *Test* realizado.
- *Started*: campo del tipo *datetime* que indica momento del inicio del *Test*.
- *Finished*: campo del tipo *datetime* que indica momento del final del *Test*.
- *Duration*: campo de tipo *duration* que indica la duración del *Test*.
- *State*: campo de tipo *string* que indica la situación actual del *Test*. Las opciones son *In Progress* o *Closed*.
- *Execution Environment*: campo de tipo referencia a la tabla *Test Environment*, que muestra el entorno de ejecución en el que se ha realizado el *Test*. Solo se puede asociar uno que pertenezca a la Compañía del *Test*.
- *Run by*: campo de tipo referencia a la tabla *User* que muestra que usuario fue el que realizó la ejecución del *Test*.
- *Total Test*: campo de tipo *integer* que muestra el número de *Test* totales realizados para el *Test Run*.
- *Passed*: campo de tipo *integer* que muestra el número de *Test* correctos realizados para el *Test Run*.
- *Failed*: campo de tipo *integer* que muestra el número de *Test* fallidos realizados para el *Test Run*.
- *Blocked*: campo de tipo *integer* que muestra el número de *Test* bloqueados realizados para el *Test Run*.
- *Not Finished*: campo de tipo *integer* que muestra el número de *Test* no finalizados realizados para el *Test Run*.

Esta tabla contiene una serie de listas relacionadas que muestran las siguientes relaciones:

- *Test Results*: muestra todos los Resultados de *Test* asociados al *Test Run* en concreto.

Además de las *UI Actions* nombradas como generales, la entidad *Test Run* tiene otras relacionadas exclusivamente para su uso:

- *Continue Run*: de tipo *form button*, permite retomar la ejecución de un *Testeo* que se había quedado en pausa durante su ejecución.

5.2.9 Entidad Entorno de *Test*

Es una entidad en la que están registrados los distintos entornos disponibles en los que se ha podido realizar un *Test*. Los campos asociados a esta tabla que a su vez tienen relevancia en el proyecto son:

- *Number*: autonumérico que proporciona *ServiceNow*, que se usa para identificar el número de registro dentro de la tabla.
- *Name*: de tipo *string* que nombra el entorno.
- *Short Description*: campo de tipo *string* que aporta una pequeña descripción del entorno.
- *Type*: de tipo *string* que determina el tipo del que es un entorno. Las opciones son *Development, Production, QA, Staging, Support y UAT*.
- *Company*: de tipo *reference*, hace referencia a un registro de la tabla *Company*. Contiene la empresa a la que pertenece ese Entorno de *Test*.
- *URL*: de tipo *url*, indica la url para acceder al entorno.

5.2.10 Entidad Plan de *Test*

Entidad que nos muestra un plan de *Test* creado por un usuario concreto que permite juntar los *Test* en distintas agrupaciones. Los campos asociados a esta tabla que a su vez tienen relevancia en el proyecto son:

- *Number*: autonumérico que proporciona *ServiceNow*, que se usa para identificar el número de registro dentro de la tabla.
- *Planned Start Date*: campo de tipo *datetime* que indica cuando está previsto que empiece el plan de *Test*.
- *Planned End Date*: campo de tipo *datetime* que indica cuando está previsto que finalice el plan de *Test*.
- *Planned Duration*: campo de tipo *duration* que indica cual es la duración prevista.
- *Name*: campo de tipo *string* que nombra el plan de *Test*.
- *State*: campo de tipo *integer* que indica el estado actual en el que está el plan de *Test*. Las opciones son *Pending, Work in Progress, Open, Closed Completed, Closed Incompleted y Closed Skipped*.
- *Owner*: campo de tipo referencia a *User*, que indica que usuario es el encargado de la gestión de ese Plan de *Test* concreto.
- *Description*: campo de tipo *string* que aporta una explicación del plan de *Test*.
- *Percent Complete*: campo de tipo *percent complete* que indica que porcentaje del plan de *Test* se ha ejecutado.
- *Percent Passed*: campo de tipo *percent complete* que indica que porcentaje del plan de *Test* se ha ejecutado con éxito.
- *Percent Failed*: campo de tipo *percent complete* que indica que porcentaje del plan de *Test* se ha ejecutado fallidamente.
- *Percent Blocked*: campo de tipo *percent complete* que indica que porcentaje del plan de *Test* se ha bloqueado.

Esta tabla contiene una lista relacionada que muestra la siguiente relación:

- *Test Cycles*: establece la relación entre el plan de *Test* y los distintos ciclos de *Test* que componen el mismo.

5.2.11 Entidad Ciclo de *Test*

Entidad que nos muestra una parte de un plan de *Test* que está previsto ejecutar en un periodo concreto que se incluye dentro del plan de *Test*. Los campos asociados a esta tabla que a su vez tienen relevancia en el proyecto son:

- *Number*: autonumérico que proporciona *ServiceNow*, que se usa para identificar el número de registro dentro de la tabla.
- *Planned Start Date*: campo de tipo *datetime* que indica cuando está previsto que empiece el ciclo de *Test*.
- *Planned End Date*: campo de tipo *datetime* que indica cuando está previsto que finalice el ciclo de *Test*.
- *Planned Duration*: campo de tipo *duration* que indica cual es la duración prevista.
- *Name*: campo de tipo *string* que nombra el ciclo de *Test*.
- *State*: campo de tipo *integer* que indica el estado actual en el que está el ciclo de *Test*. Las opciones son *Pending*, *Work in Progress*, *Open*, *Closed Completed*, *Closed Incompleted* y *Closed Skipped*.
- *Test Plan*: campo de tipo referencia a *Test Plan*, que indica en que plan de *Test* está incluido en ciclo de *Test* concreto
- *Description*: campo de tipo *string* que aporta una explicación del ciclo de *Test*.
- *Percent Complete*: campo de tipo *percent complete* que indica que porcentaje del ciclo de *Test* se ha ejecutado.
- *Percent Passed*: campo de tipo *percent complete* que indica que porcentaje del ciclo de *Test* se ha ejecutado con éxito.
- *Percent Failed*: campo de tipo *percent complete* que indica que porcentaje del ciclo de *Test* se ha ejecutado fallidamente.
- *Percent Blocked*: campo de tipo *percent complete* que indica que porcentaje del ciclo de *Test* se ha bloqueado.

Esta tabla contiene una lista relacionada que muestra la siguiente relación:

- *Test Execution Suite*: establece la relación entre el ciclo de *Test* y los distintos softwares de ejecución de *Test* que componen el mismo.

5.2.12 Entidad Software de Ejecución de *Test*

Entidad que muestra un plan de ejecución de *Test*, asignando los encargados de ejecutar estos y delimitando los plazos en los que estos se tienen que ejecutar. Los campos asociados a esta tabla que a su vez tienen relevancia en el proyecto son:

- *Number*: autonumérico que proporciona *ServiceNow*, que se usa para identificar el número de registro dentro de la tabla.
- *Planned Start Date*: campo de tipo *datetime* que indica cuando está previsto que empiece el software de ejecución de *Test*.
- *Planned End Date*: campo de tipo *datetime* que indica cuando está previsto que finalice el software de ejecución de *Test*.
- *Planned Duration*: campo de tipo *duration* que indica cual es la duración prevista.
- *Name*: campo de tipo *string* que nombra el software de ejecución del *Test*.
- *Test Cycle*: campo de tipo referencia a la tabla *Test cycle* que muestra a que ciclo de *Test* está asociado el software de ejecución de *Test*.
- *State*: campo de tipo *choice* que indica el estado actual en el que está el software de ejecución de *Test*. Las opciones son *Draft*, *Planning*, *Current*, *Complete* y *Cancelled*.
- *Assignment Group*: campo de tipo referencia a la tabla *Group* que indica que grupo se encargará de ejecutar estas ejecuciones de *Test*.

- *Assigned To*: campo de tipo referencia a la tabla *User* que indica que usuario se encargará de ejecutar estas ejecuciones de *Test*.
- *Story*: campo de tipo referencia a la tabla *Storie*, que asocia el software de ejecución de *Test* con una *story* concreta.

Esta tabla contiene una lista relacionada que muestra la siguiente relación:

- *Test Execution Assignment*: establece la relación entre un *Test* y el software de ejecución de *Test* asociado correspondiente.

Además de las *UI Actions* nombradas como generales, la entidad Software de Ejecución de *Test* tiene otras relacionadas exclusivamente para su uso:

- *Add Test Set*: de tipo *form button*, te direcciona a la tabla *Test Set*, donde te muestra el listado de conjuntos de *Test* y el botón *Add To Execution Suite*.
- *Add Test*: de tipo *form button*, te direcciona a la tabla *Test*, donde te muestra el listado de *Test* y el botón *Add To Execution Suite*.

5.2.13 Entidad Asignación de ejecución de *Test*

Entidad que relaciona los distintos *Test* con un software de ejecución de *Test* y se lo asigna a un usuario. Los campos asociados a esta tabla que a su vez tienen relevancia en el proyecto son:

- *Test*: campo de tipo referencia a la tabla *Test* que indica el *Test* asociado.
- *Test Execution Suite*: campo de tipo referencia a la tabla *Test Execution Suite* que indica el software de ejecución de *Test* asociado.
- *Assigned To*: campo de tipo referencia a la tabla *User* que indica que usuario se encargará de este *Test* en concreto.

Además de las *UI Actions* nombradas como generales, la entidad Asignación de Ejecución de *Test* tiene otras relacionadas exclusivamente para su uso:

- *Run*: de tipo *list banner button*, permite la ejecución de una serie de *Test* que se muestran en la tabla uno detrás de otro, de forma que todos formen parte del mismo *Test Run*. Para que el botón se muestre, la tabla tiene que tener un filtro en el que solo se muestre las asignaciones de ejecución de *Test* de un mismo *Test* y que el usuario logueado sea el que se encuentra en el campo *Assigned To* de todos los registros. La versión de *Test* que ejecuta es la cual cuyo campo *state* tiene valor *Ready*.

5.2.14 Entidad *Storie*

Es una entidad en las que se definen los desarrollos técnicos que se tienen que realizar en una determinada tarea, además de los criterios de aceptación de esta. Además, define quien se debe encargar de ella, los plazos que tiene para ello y quien es el encargado de supervisar la correcta realización de la tarea.

El desarrollo de esta entidad no es realizado en sí para este Trabajo de Fin de Grado, pero si la capacidad de asociar esta entidad con la entidad Software de Ejecución de *Test*. Para ello, cuenta con la siguiente *Related list*:

- Software de ejecución de *Test*: permite crear un registro en la entidad *Test Execution Suites*, que tendrá el campo *Storie* completado con el registro de la entidad *Storie* desde el que se ha creado.

5.2.15 Database View Test Run Step Results

Test Run Step Results es una *Database View* que permite crear una relación entre distintas tablas basándose en campos referencia contenidas en las mismas. Las tablas utilizadas para crear esta *Database View* son *Test Set Test* (Conjunto de *Test Test*), *Test Version* (Version de *Test*), *Test Result* (Resultado de *Test*) y *Step Result* (Resultado de paso). Además, a cada tabla se le asocia un prefijo distinto para luego poder mostrar sus campos en caso de que estos tengan el mismo nombre. Los prefijos son las siguientes:

Tabla	Prefijo
<i>Test Set Test</i>	tsm2m
<i>Test Version</i>	tv
<i>Test Result</i>	tr
<i>Step Result</i>	sr

Tabla 5.2.2 Relación de tablas Test Run Step Results

Los distintos campos que se muestran en los formularios son los siguientes (se indica entre paréntesis de donde proviene el campo):

<i>Number</i> (<i>sr_Test_result.Test_run</i>)	<i>Run by</i> (<i>sr_Test_result.Test_run</i>)
<i>Test Run</i> (<i>sr_Test_result</i>)	<i>Updated</i> (<i>sr_Test_result</i>)
<i>Execution Environment</i> (<i>sr_Test_result.Test_run</i>)	
<i>Test Set</i> (<i>tsm2m_Test_set</i>)	<i>Company</i> (<i>tsm2m_Test_set</i>)
<i>Code</i> (<i>tsm2m_Test</i>)	
<i>User logged</i> (<i>tsm2m_Test</i>)	
<i>Previous requirement</i> (<i>tsm2m_Test</i>)	
<i>Short Description</i> (<i>tsm2m_Test</i>)	
<i>Description</i> (<i>tsm2m_Test</i>)	
<i>Code</i> (<i>sr_step</i>)	<i>Needs Verification</i> (<i>sr_step</i>)
<i>Step</i> (<i>sr_step</i>)	<i>Result</i> (<i>sr_result</i>)
<i>Comments</i> (<i>sr_comments</i>)	

Tabla 5.2.3 Formulario de Test Run Step Results

Los campos que se muestran en el listado del registro para esta *Database View* son los siguientes, mostrándose estos en el orden en el que señalan (entre paréntesis a que registro están asociados):

- *Company* (*Test Set*).
- *Owner*.
- *Test Run*.
- *Process* (*Test Set*).
- *Category* (*Test Set*).
- *Test Set*.
- *Code* (*Test Set*).
- *Short Description*.
- *Description* (*Test*).
- *Previous Requirements* (*Test*).
- *User logged* (*Test*).
- *Code* (*Step*).

- Step (Step).
- Result.
- Comment.
- Execution Environment (Test Result.Test Run).
- Updated (Test Result).
- Run by (Test Result.Test Run).

5.2.16 Database View Test Export

En esta *Database View* se asocian tres tablas con el objetivo de proporcionar una vista que permita exportar los datos de diferentes registros de diferentes tablas que tienen una relación entre ellos a través de distintos campos referencia. Las tablas que utilizamos en este caso son *Test*, Paso (Step Result) y Resultado de Test (Test Result). Los prefijos que se asocian a cada tabla son los siguientes:

Tabla	Prefijo
<i>Test</i>	<i>Test</i>
<i>Test Result</i>	tr
<i>Step Result</i>	sr

Tabla 5.2.4 Prefijos Database View Test Exports

Los distintos campos que se muestran en los formularios son los siguientes (se indica entre paréntesis de donde proviene el campo):

<i>Company (Test_u_Test_set)</i>	<i>Test Set (Test)</i>
<i>Owner (Test_u_Test_set)</i>	<i>Category (Test_u_Test_set)</i>
<i>Process (Test_u_Test_set)</i>	<i>Code (Test)</i>
<i>User logged (Test)</i>	
<i>Previous requirement (Test)</i>	
<i>Short Description (Test)</i>	
<i>Description (Test)</i>	
<i>Code (sr_step)</i>	<i>Needs Verification (sr_step)</i>
<i>Step (sr_step)</i>	
<i>Comments (sr_comments)</i>	
<i>Execution Environment (sr_Test_result.Test_run)</i>	<i>Updated (tr_sys_updated_on)</i>
<i>Run by (sr_Test_result.Test_run)</i>	

Tabla 5.2.5 Formulario de Test Exports

Los campos que se muestran en el listado del registro para esta *Database View* son los siguientes, mostrándose estos en el orden en el que señalan (entre paréntesis a que registro están asociados):

- *Company (Test_u_Test_set)*.
- *Owner (set)*.
- *Process (Test_u_Test_set)*.
- *Category (Test_u_Test_set)*.
- *Test Set (Test)*.
- *Code (Test)*.
- *User logged (Test)*.
- *Previous requirement (Test)*.
- *Short Description (Test)*.

- Description (*Test*).
- Code (*sr_step*).
- Needs Verification (*sr_step*).
- Comments (*sr_comments*).
- Execution Environment (*sr_Test_result.Test_run*).
- Updated (*tr_sys_updated_on*).
- *Run by* (*sr_Test_result.Test_run*) Updated (*tr_sys_updated_on*).

5.2.17 Menús y Módulos relacionados con *Test Management*

Los menús y módulos por los que se puede acceder a las entidades de *Test* son los siguientes:

Menú *Test Management 2.0*:

- *Test Board*: muestra un *dashboard* que contiene los Planes de *Test* en los que el usuario logueado aparece en el campo *Owner*.
- *Test Dashboard*: muestra un *dashboard* en el que se resume todo lo relacionado con la ejecución de *Test Results* y los datos de estas.
- *Test Run Results*: muestra todos los registros de la *Database View Test Run Step Results* en los que el campo *code* perteneciente a la entidad Paso este relleno.
- *Test Exports*: muestra todos los registros de la *Database View Test Exports*.
- *Test*: muestra todos los registros pertenecientes a la tabla *Test*.
- *Test Set*: muestra todos los registros pertenecientes a la tabla Conjunto de *Test*.
- *Test Plan*: muestra todos los registros pertenecientes a la tabla Planes de *Test* en los que el registro está activo (campo *active* con valor *true*).
- *Environment*: muestra todos los registros pertenecientes a la tabla de Entornos de *Test*.
- *Runs*: muestra todos los registros pertenecientes a la tabla *Test Run* en los que el campo *Run By* es el usuario logueado que accede al módulo.
- *Test Assigned to Me*: muestra todos los registros pertenecientes a la tabla Asignación de ejecución de *Test* en los que el campo *Assigned to me* es el usuario logueado que accede al módulo y el Software de Ejecución de *Test* está activo (campo *active* de este registro es *true*).
- *Test Execution Suites*: muestra todos los registros pertenecientes a la tabla Software de Ejecución de *Test*.
- *Stories Execution Suites*: muestra todos los registros pertenecientes a la tabla Software de Ejecución de *Test* en los que el campo *story* no está vacío.
- *Settings*: es un separador que incluye otros módulos dentro de él.
 - *Planning Console*: muestra relaciones entre los campos cuando se importan *Test* desde un archivo externo.
 - *Test Import Logs*: muestra los mensajes impresos como resultado de la importación de *Test*.
 - *Test Importer*: permite importar *Test* desde un archivo Excel a las tablas correspondientes.

5.2.18 Ejecución de *Test*

Todas estas tablas están orientadas a dar un servicio final que es la ejecución de *Test* en la herramienta. Para ejecutar los *Test*, se puede hacer desde dos puntos distintos, a través de las *UI Action* denominadas *Run*, que se han definido en sus respectivas tablas. Para ejecutar un solo *Test* en concreto, se puede hacer usando la *UI Action Run* del formulario versión de *Test*, que ejecuta una

versión en concreto, mientras que para ejecutar varios se hace desde la *UI Action Run* de la lista de asignación de software de ejecución de *Test*.

Cuando se pulsa uno de estos dos botones se activa la UI Page denominada *Test_step_list*, que se abre una ventana emergente y mediante la cual se realiza la ejecución en los siguientes pasos.

- Paso 1: aparece la ventana emergente que contiene dos campos, por un lado, el campo de tipo *string Run Name*, donde introduciremos el nombre que queramos asignarle al *Test Run* en concreto, y por otro el campo *Execution Environment*, que es un campo referencia donde se muestran las distintas instancias en las que se puede ejecutar el *Test*. Además, en la parte de debajo de la ventana, se muestran dos botones:
 - *Cancel*: cancela el *Test* que se pretendía realizar.
 - *Run*: inicia el *Test*. En el momento que se pulsa, se crean dos registros nuevos, por un lado, el *Test Run*, cuyos campos *finished* y *duration* aparecerán vacíos hasta que se termine de ejecutar el *Test*, y el campo *State* tiene valor *In Progress*. El segundo registro que se crea es el *Test Result*, que aparecerá con el campo *Result* vacío hasta que se acabe de ejecutar el *Test*.
- Paso 2: una vez que se haya pulsado el botón *Run* la ventana cargará todos los pasos que contengan las versiones de *Test* asociadas a la ejecución de *Test* concreta. Las versiones de *Test* se muestran de una en una. Los pasos que tienen activado el *checkbox Needs Verification* tienen tres opciones en las que se debe elegir una. El tic verde significa que ese paso se ha realizado correctamente, la cruz roja que el *Test* ha fallado en ese paso, y la admiración amarilla que el *Test* se ha bloqueado en ese paso. Cuando se determina que un paso ha fallado o es bloqueante, aparece un campo *Comments* de tipo *string*, en el que se puede indicar que falla o porque se produce el fallo o el bloqueo, además de poder añadir archivos adjuntos. Los botones que se muestran en este paso de la UI Macro son los siguientes:
 - *Pause*: permite pausar la ejecución del *Test*. Para retomarla hay que pulsar la *UI Action Continue Run* del formulario *Test Run* que se ha creado para ese *Test*.
 - *Done*: finaliza el *Test* que se visualiza en ese momento con lo indicado en cada uno de los pasos. Este botón finaliza el *Test Run* que se está realizando.

Además de estos dos botones, cuando se están ejecutando varios *Test* a la vez, aparecen dos botones con una flecha, que nos permite movernos entre los distintos *Test*. El orden en el que se muestran los *Test* viene determinado primero por el campo *Number* del conjunto de *Test* con el que el *Test* este asociado, segundo por el campo *code* del *Test* asociado, y tercero por el campo *Number* del *Test* asociado.

5.2.19 Dashboards y reportes

Además de las visualizaciones de los registros en formulario y registro, la aplicación también cuenta con una serie de tablas y gráficas que permiten comparar los distintos registros que se muestran en las tablas y hacer estudios sobre ellos. Para mostrar todos esos datos se utilizan distintos tipos de reportes. En estos *dashboard* se utilizarán los siguientes tipos [12]:

Nombre	Descripción
<i>Single Score</i>	Muestra un número entero
<i>List</i>	Muestra una lista de registros
<i>Pie</i>	Muestra una gráfica circular
<i>Multilevel Pivot</i>	Muestra datos dependiendo de varias variables distintas
<i>Spline</i>	Muestra los datos en una gráfica lineal de 2 ejes

Tabla 5.2.6 Tipos de Reports utilizados

Además, utilizaremos otro tipo de función que nos permite crear filtros interactivos. Estos filtros se aplican a todos los reportes que tengan activos los botones “Follow interactive *filter*” y “Show when *following filter*”. El primero se marca cuando se sigue a los filtros interactivos y el segundo muestra los momentos en el que se siguen. Cada reporte sigue solo a los filtros interactivos de su pestaña. En este caso todos los reportes que hay siguen a los filtros interactivos. El filtro interactivo puede ser de varios tipos, en este caso solo se utilizarán filtros de tipo *reference*, que hacen referencia a campos concretos de una tabla [13].

En *Test Management* hay dos *dashboards*.

El primer *dashboard* se denomina *Test Board*, y es ofrecido por *ServiceNow* para el análisis de planes de *Test*. En él se muestran los distintos planes de texto activos que se asocian mediante el campo *Ownership* con el usuario logueado.

Puedes acceder desde el mismo a *dashboard* a cualquier plan de *Testeo* en concreto, donde podrás ver los detalles del mismo y una gráfica en la que se visualiza como está diseñado el plan.

El segundo *dashboard* se denomina *Test Management* y se centra en mostrar los datos de la ejecución de *Test* a través de las tablas del mismo para poder analizar los resultados. Dentro del propio *dashboard* encontramos una serie de *reports* distintos. Estos se dividen en tres pestañas distintas:

- Pestaña “*Test Result*”:
 - *Test Run Executed: report* de tipo *Single Score* que muestra el número total de registros en la tabla *Test Run*, es decir, el número de pruebas totales ejecutadas.
 - *Test Executed: report* de tipo *Single Score* que muestra el número total de registros en la tabla *Test Result*, es decir, el número de pruebas totales ejecutadas.
 - *Test Passed: report* de tipo *Single Score* que muestra el número total de registros en la tabla *Test Result* cuyo campo *Result* tiene como valor *Passed*, es decir, el número de pruebas totales que se han superado con éxito. Tiene color verde.
 - *Test Failed: report* de tipo *Single Score* que muestra el número total de registros en la tabla *Test Result* cuyo campo *Result* tiene como valor *Failed*, es decir, el número de pruebas totales que se han fallado en su ejecución. Tiene color rojo.
 - *Test Blocked: report* de tipo *Single Score* que muestra el número total de registros en la tabla *Test Result* cuyo campo *Result* tiene como valor *Blocked*, es decir, el número de pruebas totales que se han bloqueado durante su ejecución. Tiene color amarillo.
 - *Test Result: report* de tipo *List* que muestra todos los registros de la tabla *Test Result* y da acceso a ellos.

- *Test Run*: filtro interactivo que permite filtrar todos los reportes mencionados en la pestaña. Lo hace a través de la tabla *Test Result*, usando el campo *Test Run*, que hace referencia a la tabla del mismo nombre [13] [14].
- *Company*: filtro interactivo que permite filtrar todos los reportes mencionados en la pestaña. Lo hace a través de la tabla *Test Result*, siguiendo el camino del campo *Test Run*, el campo *Execution Environment* que contiene, y acabando en el campo *company*, que hace referencia a la tabla del mismo nombre, y por el que se realiza el filtrado.
- Pestaña “*Test Trend*”:
 - *Test Trend: report* de tipo *Spline* que muestra una gráfica en la que se visualizan los *Test* ejecutados mensualmente, dividiendo estos en cuatro líneas según el resultado final del *Test*, *Passed* (color verde), *Failed* (color rojo), *Blocked* (color amarillo) y *Not Finished* (color azul).
 - *Current Quarter Result: report* de tipo *Multilevel Pivot* que muestra todos los registros de la tabla *Test Result*, dividiendo estos en cuatro líneas según el resultado final del *Test*. Los colores asociados son similares a los indicados en el *report Test Trend*.
 - *Test Run*: filtro interactivo que permite filtrar todos los reportes mencionados en la pestaña. Lo hace a través de la tabla *Test Result*, usando el campo *Test Run*, que hace referencia a la tabla del mismo nombre.
 - *Company*: filtro interactivo que permite filtrar todos los reportes mencionados en la pestaña. Lo hace a través de la tabla *Test Result*, siguiendo el camino del campo *Test Run*, el campo *execution Environment* que contiene, y acabando en el campo *company*, que hace referencia a la tabla del mismo nombre, y por el que se realiza el filtrado.
- Pestaña “*Usage*”:
 - *Test Executed by Month: report* de tipo *Spline* donde aparece el número total de *Test* ejecutados agrupándolos mensualmente.
 - *Test Created: report* de tipo *Spline* donde aparece el número total de *Test* creados agrupándolos mensualmente.
 - *Test not Owned by Managers: report* de tipo *Multilevel Pivot* que muestra todos los registros de la tabla *Test Version*, los cuales no hayan sido creados por las personas pertenecientes al departamento de QA de la empresa. Muestra el campo *created by* con el usuario que lo ha creado en las columnas, y la empresa para las que se crea el *Test* en las filas.
 - *Company*: filtro interactivo que permite filtrar todos los reportes mencionados en la pestaña. Lo hace a través de la tabla *Test Version*, siguiendo el camino del campo *Test*, y acabando en el campo *company*, que hace referencia a la tabla del mismo nombre, y por el que se realiza el filtrado.

5.2.20 Importador de *Test*

Los importadores nos permiten cargar una serie de datos externos a la herramienta en la misma. En este se ha creado un importador que nos permitirá, a través de un documento Excel, insertar los datos necesarios para crear los distintos registros asociados a los *Test*. En concreto, este importador nos permite crear registros para las tablas Conjunto de *Test*, *Test*, Versión de *Test* y Pasos [15].

Para ello se necesitan los siguientes componentes que llevarán a cabo la importación:

- *Data Source*: es un registro en el que se introducen los datos con los que se va a llevar a cabo la importación (tipo de fichero, nombre de la tabla a la que se realiza la importación...). En

el *Data Source* se carga el archivo Excel y este crea un *import set* con esos datos concretos. El *Data Source* para importar los *Test* se nombre como *SS – Test Importer*.

- *Import Set*: es un registro intermedio en el que se almacenan todos los datos que se van a subir a la plataforma. Para cada *Test* se crea un registro, a los que se puede acceder desde la *Related list Import Set Rows* que contiene el propio *Import Set*.
- Una tabla auxiliar denominada *SS - Test Importer 2*, la cual recoge todos los datos necesarios para crear los *Test* del Excel. Actúa como un espejo de las entidades de *Test*, ya que se utiliza para mapear los campo de la misma a las tablas *Test* con las que se trabaja habitualmente. Los campos que contiene esta tabla, que son todos de tipo *string*, son:
 - *u_categoria*: contiene la categoría que se añadirá al campo *category* de la entidad Conjunto de *Test*.
 - *Cod. CdP*: contiene el código que se añadirá al campo *code* de la entidad *Test*.
 - *Cod. Paso*: contiene el código que se añadirá al campo *code* de la entidad *Paso*.
 - *Datos de prueba / Usuario logueado*: contiene los datos que se añadirán en el campo *User logged* de la entidad *Test*.
 - *Descripción CdP*: contiene la descripción que se añadirá en el campo *description* de la entidad *Test*.
 - *Descripción Paso*: contiene la descripción que se añadirá en el campo *step* de la entidad *Paso*.
 - *Empresa*: contiene la empresa que se añadirá en los campos *company* de la entidad *Test* y de la entidad Conjunto de *Test*.
 - *Escenario*: contiene el nombre que se asignará en el campo *name* de la entidad Conjunto de *Test*.
 - *Nombre CdP*: contiene el nombre que se asignará en el campo *short Description* de la entidad *Test*.
 - *Owner*: contiene el nombre del usuario que se asignara al campo *Owner* de la entidad Conjunto de *Test*.
 - *Proceso*: contiene el proceso que se asignará al campo *process* de la entidad Conjunto de *Test*.
 - *Requisitos previos*: contiene los requisitos previos que se asignará al campo *previous requirements* de la entidad *Test*.
 - *Resultado Esperado*: contiene la descripción para rellenar el campo *Step* de la entidad *Paso*, siempre que el campo descripción paso se encuentre vacío.
- *Transform Map*: se encarga de crear los registros en las tablas destino mapeando los campos de la tabla *SS – Test Importer 2* con las tablas correspondientes. Lo hace a través de un *Script*.

6 Implementación de entidades de funcionalidad

Además de las tablas y los campos que sirven para crear entidades, hay otras en *ServiceNow* que aportan funcionalidad al proyecto. En estas entidades se programa a través de *JavaScript* [16] para conseguir que todos los procesos funcionen correctamente. Estas entidades se dividen en cliente, que son las que se ejecutan de cara a cliente y que no acceden a la base de datos de *ServiceNow* directamente, si no que tendrán que llamar a una entidad de servidor para que lo haga por ellas. Las llamadas desde cliente no pueden actualizar campos de otros registros que no sea en el que se está ejecutando. Las entidades de servidor son aquellas que si pueden acceder a la base de datos, obtener información de los distintos registros de cualquier entidad, y hacer cambios en cualquier registro. Además de las reglas normales de programación en *JavaScript*, *ServiceNow* proporciona una serie de objetos y funciones predefinidas que facilitan la labor de implementación en la herramienta. Estos objetos se dividen, de manera que hay algunos disponibles en el lado del cliente [17] y otros que solo lo están en el lado del servidor [18].

6.1 *Business Rules*:

Una *Business Rule* es una entidad de tipo servidor que contiene un script de servidor que se ejecuta cuando en el registro de una tabla asociada se produce una acción, como puede ser la creación, borrado o inserción de un registro [19]. Esta tabla es una tabla propia de *ServiceNow* y proporciona algunas funcionalidades predefinidas.

En las *Business Rules* hay varios campos que delimitan cuando se ejecutarán. Estos campos de especial relevancia son:

- *Table*: de tipo referencia, determina la tabla para la que se ejecuta la *Business Rule*.
- *When*: de tipo *string*, indica, en relación con otra serie de campos, cuando se ejecuta la *Business Rule*. Puede ser *Before*, *After*, *Display* o *Async*.
- *Insert*, *Update*, *Delete* y *Query*: campos de tipo *True/False* que determina cuando se ejecuta la *Business Rule* en combinación con el campo *when*. Puede haber varios marcados a la vez.

<i>Campo When</i>	<i>Campo True False marcado</i>	<i>Momento de ejecución</i>
<i>Before</i>	<i>Insert</i>	Antes de insertar el registro
	<i>Update</i>	Antes de actualizar el registro
	<i>Delete</i>	Antes de borrar el registro
	<i>Query</i>	Antes de la query
<i>After</i>	<i>Insert</i>	Después de insertar el registro
	<i>Update</i>	Después de actualizar el registro
	<i>Delete</i>	Después de borrar el registro
	<i>Query</i>	Después de la query
<i>Async</i>	<i>Insert</i>	Asíncronamente al insertar el registro
	<i>Update</i>	Asíncronamente al actualizar el registro
	<i>Delete</i>	Asíncronamente al borrar el registro
	<i>Query</i>	Asíncronamente al lanzar la query
<i>Display</i>		Al cargar el formulario

Tabla 6.1.1 Funcionamiento de las *Business Rules*

- *Filter condition*: de tipo *condition*, establece condiciones para la ejecución de la *Business Rule*.
- *Conditions*: de tipo *condition*, establece condiciones para la ejecución de la *Business Rule* a través de código escrito.
- *Script*: de tipo *script*, parte de código que se ejecutara cuando se cumplan todas las condiciones.

6.1.1 *Business Rules* de Gestión de Proyecto

Las *Business Rules* asociadas a las tablas de Gestión de Proyectos son las siguientes:

- *SS - Calculate Projected SDC*: calcula el valor de los campos *projected effort* y *cost* del SDC para una tarea de proyecto cuando los campos *Actual SDC* se actualizan.
- *SS - Calculate Projected BPC*: calcula el valor del campo *projected effort* y *cost* BPC para una tarea de proyecto cuando los campos *Actual BPC* se actualizan.
- *SS - Calculate Projected PDM*: calcula el valor del campo *projected effort* y *cost* PDM para una tarea de proyecto cuando los campos *Actual PDM* se actualizan.
- *SS - Calculate Projected DC*: calcula el valor del campo *projected effort* y *cost* DC para una tarea de proyecto cuando los campos *Actual DC* se actualizan.
- *SS - Projected Cost/Effort Project Task*: calcula el valor del campo *projected effort* y *cost* para una tarea de proyecto.
- *SS - Projected Cost/Effort Project*: calcula el valor del campo *projected effort* y *cost* para un proyecto.
- *SS - Calculated Planned Effort and cost*: calcula todos los campos *planned cost* asociados a cada rol, y además los campos *planned effort* y *planned cost* de la tarea de proyecto.
- *SS - Planned Effort Mandatory*: comprueba que al menos uno de los campos *planned effort* de rol de la tarea de proyecto es mayor que 0.
- *SS - Update Actual Cost and Effort*: calcula el valor de los campos *actual cost* y *effort* para los roles cuando se aprueba una *Time Card*.
- *SS - Calculated Projected Cost Effort RP*: calcula el valor de los campos *projected cost* y *effort* de una tarea de proyecto para todos los roles cuando un *Plan de Recursos* de aprueba o se cancela.
- *Populate Resource Plan*: popula el campo *resource plan* cuando se inserta o actualiza una *Time Card*.
- *SS - Calculate Projected SDC*: calcula el valor de los campos *projected effort* y *cost* del SDC para una tarea de proyecto cuando una *Asignación de Recursos* de ese rol se actualiza.
- *SS - Calculate Projected BPC*: calcula el valor del campo *projected effort* y *cost* BPC para una tarea de proyecto cuando una *Asignación de Recursos* de ese rol se actualiza.
- *SS - Calculate Projected PDM*: calcula el valor del campo *projected effort* y *cost* PDM para una tarea de proyecto cuando una *Asignación de Recursos* de ese rol se actualiza.
- *SS - Calculate Projected DC*: calcula el valor del campo *projected effort* y *cost* DC para una tarea de proyecto cuando una *Asignación de Recursos* de ese rol se actualiza.

6.1.2 *Business Rules* de Test Management

Las *Business Rules* asociadas a las tablas de *Test Management* son las siguientes:

- *SS - Autopopulate Test Version's Company*: popula el campo *Company* de las entidades *Versión de Test* asociadas a un *Test*.
- *SS - Updated Relation Test Set – Test*: crea y elimina registros de la tabla *Conjunto de Test Test*, de forma que cada entidad *Test* solo pueda estar asociado a un registro de esta tabla.
- *Create Test*: cuando se crea una *Versión de Test* original (número de versión 1) se crea el *Test* asociado al que se asocia, populando todos los campos.
- *SS - Insert Code Step*: popula los *code Steps* cuando se crea una copia de una *Versión de Test*.

- *SS - Update Assignment To*: actualiza los campos *Assigned To* de los registros Asignación de Ejecución de *Test* con el mismo campo del registro Software de Ejecución de *Test* con el que se relaciona.

6.2 Client Script

Un *Client Script* es una entidad de tipo cliente que contiene un script de cliente que se ejecuta cuando en el registro de una tabla asociada se produce una acción, como puede ser la carga del formulario o la modificación de algún campo [20]. Esta tabla es una tabla propia de *ServiceNow* y proporciona algunas funcionalidades predefinidas.

En los *Client Scripts* hay varios campos que delimitan cuando se ejecutarán. Estos campos de especial relevancia son:

- *Table*: de tipo referencia, determina la tabla para la que se ejecuta el *Client Script*.
- *Inherited*: de tipo *True/False*, si está activo todas las tablas que extiendan de la tabla indicada en el campo *table* también tendrán dicha funcionalidad.
- *Type*: de tipo *string*, determina cuando se ejecuta el *Client Script*, como se indica en la siguiente tabla:

Nombre	Cuando se ejecuta
<i>onLoad</i>	Cuando se carga el formulario
<i>onSubmit</i>	Cuando se actualiza el formulario
<i>onChange</i>	Cuando un campo designado en el campo <i>field name</i> cambia, pudiendo incluir durante la carga del formulario.
<i>onCellEdit</i>	Cuando un campo designado en el campo <i>field name</i> es modificado dese el listado.

Tabla 6.2.1 Campo *type* en los *Client Script*

- *Field Name*: de tipo *field*, indica que campo de la tabla hará que se dispare la ejecución (solo aplica en *Client Scripts* de tipo *onChange* o *onCellEdit*).
- *Script*: de tipo *script*, parte de código que se ejecutara cuando se cumplan todas las condiciones.

6.2.1 Client Scripts de Gestión de Proyecto

Los *Client Scripts* asociados a las tablas de Gestión de Proyectos son los siguientes:

- *SS - Resource Type "Role"*: de tipo *onChange*, en un Plan de Recursos si el campo *project* hace referencia a la tabla Demanda, el campo *resource type* tiene valor *Role Type* y pasa a ser un campo no editable.
- *SS - Change Resource Type*: de tipo *onChange*, en un Plan de Recursos si el campo *resource type* cambia, el campo *resource rate* se vacía automáticamente.
- *SS - R.Rate & Message Member List Role*: de tipo *onChange*, en un Plan de Recursos si el campo *member list* cambia, se popula el campo *resource rate* con el valor encontrado en la tabla Hojas de Tarifas de Gasto para el rol que tengan los miembros del campo. También comprueba si todos los usuarios de la lista tienen el mismo rol.
- *SS - Resource Rate & Message User Role*: de tipo *onChange*, en un Plan de Recursos si el campo *user* cambia, se popula el campo *resource rate* con el valor encontrado en la tabla Hojas de Tarifas de Gasto para el rol que tenga el usuario.

- *SS - Set Resource Rate When Task Changes*: de tipo *onChange*, en un Plan de Recursos si el campo *project* cambia, se popula el campo *resource rate* con el valor encontrado en la tabla Hojas de Tarifas de Gasto para la tarea.
- *SS - Check Rate Before Approve*: de tipo *onSubmit*, cuando se aprueba una *Time Card* se comprueba que hay una Hoja de Tarifa de gasto asociada a los datos que contiene.

6.2.2 Client Scripts de Test Management

Los *Client Scripts* asociados a las tablas de *Test Management* son los siguientes:

- *SS - Short Description Editable*: de tipo *onLoad*, cuando se crea un *Test* el campo *short Description* es editable.
- *SS - Test Version New View Form*: de tipo *onLoad*, en un registro Versión de *Test* se muestra unos campos u otros para la vista *New View* según si el registro es nuevo o no.

6.3 Script Includes

Los *scripts includes* son registros en los que se desarrolla código JavaScript para proveer de funciones y clases que aportan funcionalidad a la instancia en el lado del servidor [21].

Para que un *Script Include* se le pueda llamar desde el lado de cliente debe tener activado el campo de tipo True/False *client callable*. Cuando un *Script Include* se marca como *Client Callable* se suele finalizar su nombre con el sufijo Ajax.

Normalmente un *Script Include* con el campo *client callable* activo, tiene una pareja con el mismo desactivado. Esta pareja comparte nombre, y el que tiene el campo *client callable* activo añade el sufijo Ajax, como se ha indicado en el párrafo anterior.

6.3.1 Script Includes de Gestión de Proyecto

Los *Script Includes* que desarrollan funcionalidad para la parte de Gestión de Proyectos son:

- *SSTimeCardAutoUtilAjax*: da funcionalidad a las *Time Card*. Es el *script include* de tipo *client callable*. Aporta funcionalidad para buscar y obtener Planes de Recursos asociados a través de la función *populateResourcePlan()*.
- *SSTimeCardAutoUtil*: da funcionalidad a las *Time Card* para completar campos de las mismas. Entre las funciones que tiene:
 - *getStartOfWeek()*: calcula el día que empieza la semana en la que se genera la *Time Card*.
 - *populateResourcePlan()*: popula el campo *resource plan* de la *Time Card*.
 - *getResourcePlan()*: obtiene el *resource plan*.
- *SS_TimeCardTaskRateProcessor*: da funcionalidad a las *Time Cards* para comprobar el buen funcionamiento de estas. Entre las funciones que tiene:
 - *processTaskTimecard()*: procesa la *Time Card* cuando este ya está completada y no va tener más modificaciones.
 - *getRate()*: comprueba que existe una Hoja de Tarifa de Gasto asociada al *Time Card* mediante el *Task* y el rol del *User* asociado.
- *SS_ProjectedCostInProject*: proporciona funcionalidad para los campos *projected effort* y *projected cost* dentro de *Project*. Entre las funciones que tiene:
 - *setProjectedCostAndEffort()*: calcula los campos *projected cost* y *projected effort* de Proyecto.
- *SS_ResourcePlanMessagesAjax*: proporciona funcionalidad para rellenar los campos de la entidad Plan de Recursos. Entre las funciones que tiene están:
 - *validateTaskRate()*: obtiene el valor del campo *task rate* de la Hoja de Tarifa de Gasto asociada para rellenar el campo *task rate* del Plan de Recursos.

- `getProjectClass()`: determina a que entidad pertenece el campo *project*.
- `getUserRole()`: determina el rol del usuario del campo *user*.
- `validateUser()`: determina que el usuario del campo *user* es válido.
- `validateValuesMember()`: determina que dos usuarios del campo *member list* tienen el mismo rol.
- `SS_ProjectedCostInProjectTask`: proporciona funcionalidad para los distintos campos de esfuerzo y coste que se encuentran dentro de la entidad Tarea de Proyecto. Entre las funciones que tiene están:
 - `calculatedCostEffortProjectTaskDC()`: modifica los campos *projected cost* y *projected effort* del rol DC y actualiza la Tarea de Proyecto.
 - `calculatedCostEffortProjectTaskSDC()`: modifica los campos *projected cost* y *projected effort* del rol SDC y actualiza la Tarea de Proyecto.
 - `calculatedCostEffortProjectTaskPDM()`: modifica los campos *projected cost* y *projected effort* del rol PDM y actualiza la Tarea de Proyecto.
 - `calculatedCostEffortProjectTaskBPC()`: modifica los campos *projected cost* y *projected effort* del rol BPC y actualiza la Tarea de Proyecto.
 - `calculateProjectedCostEffort()`: calcula el valor de los campos *projected cost* y *projected effort* basándose en un rol y en una Tarea de Proyecto.
 - `calculatedPlannedCostRol()`: calcula el valor del campo *projected cost* de un rol para una Tarea de Proyecto, un rol, y un esfuerzo planeado.

6.3.2 Script Includes de Test Management

Los *Script Includes* que desarrollan funcionalidad para la parte de *Test Management* son:

- `SS_TestUtils`: aporta funcionalidad a todo el módulo de *Test Management*, incluyendo la creación de registros a través del importador. Entre las funciones que tiene están:
 - `createTestSetRecord()`: es llamada desde el importador para crear un registro Conjunto de *Test* en caso de que este no exista previamente.
 - `createTestRecord()`: es llamada desde el importador para crear un registro *Test* en caso de que este no exista previamente.
 - `createTestVersionRecord()`: es llamada desde el importador para crear un registro Versión de *Test* en caso de que este no exista previamente.
 - `createTestStepRecord()`: es llamada desde el importador para crear un registro Paso en caso de que este no exista previamente.
 - `createRelationTestTestSet()`: crea un registro Conjunto de *Test Test*.
 - `deleteRelationTestTestSet()`: elimina un registro Conjunto de *Test Test*.
 - `updateTestAssignmentTo()`: actualiza el campo *assigned to* de los registros Asignación Ejecución de *Test* a través de un registro Software de Ejecución de *Test*.
 - `canUpdateCodes()`: comprueba si existen registros Conjunto de *Test Test* asociados a un Conjunto de *Test*.

7 Plan de pruebas

Durante la implementación de los distintos procesos que se han realizado en el Trabajo de Fin de Grado, se ha ido ejecutando un plan de pruebas para comprobar el correcto funcionamiento de los desarrollos y las posibles mejoras a realizar. Los planes de prueba de Gestión de Proyectos y los de *Test Management* se probaron de manera independiente.

Este plan de pruebas se adapta a lo dispuesto por el cliente para la creación de este[22]. No cuenta con pruebas de carácter no funcional, ya que el cliente *SilverStorm* no ha considerado necesario centrarse a los aspectos que abarcan estas pruebas durante los desarrollos. Tampoco se han desarrollado pruebas para empleados con poco conocimiento técnico, ya que todas las personas que van a utilizar las funcionalidades desarrolladas son personas con conocimiento tanto en informática como en *ServiceNow*, y dominan el vocabulario que contienen estas pruebas[23].

Además, al finalizar la implementación, se reproduce un proceso completo en el que se incluyó todo lo implementado en la herramienta de *ServiceNow*, para comprobar que los desarrollos realizados funcionaban correctamente y todos los cambios estaban correctamente integrados entre sí. Los procesos de Gestión de Proyectos y los de *Test Management*, igual que en el caso anterior se probaron de manera independiente. Además, al formar parte de un proyecto más grande, en la demo se pasará por puntos del proceso que no han desarrollado específicamente para este proceso.

Todo el plan de pruebas se ha llevado a cabo en la instancia de desarrollo de *SilverStorm*. Esta instancia es en la que se realizan todos los desarrollos internos de la compañía, y es un espejo de la instancia de producción. Cuando un desarrollo se termina y cumple los requisitos, se sube desde esta instancia a la de producción. Periódicamente se clona la instancia de producción en la de desarrollo, para que sea lo más parecida posible.

7.1 Plan de Prueba Gestión de Proyectos

7.1.1 Pasos Previos:

Para poder probar el proceso necesitamos crear previamente una serie de datos registrados que permitan comprobar la funcionalidad.

Se han creado una serie de usuarios con los roles asociados para poder realizar el proceso:

Nombre	Rol
<i>User DC</i>	DC
<i>User SDC</i>	SDC
<i>User PDM</i>	PDM
<i>User BPC</i>	BPC

Tabla 7.1.1 - Usuarios creados

También se ha creado un grupo al que pertenecen los cuatro usuarios creados previamente, que se denomina como "TFG Users Group".

Además, en la tabla Resources Roles contamos con los siguientes registros:

Nombre	Ratio Hora
DC	25€
SDC	50€
PDM	50€
BPC	55€

Tabla 7.1.2 Registros Resource Role

que nos permitirán calcular algunos campos del proceso más adelante.

7.1.2 Creación de Proyecto

Caso 1:

- Descripción: a través de la tabla de proyectos creamos un nuevo proyecto.
- Pasos a seguir:
 - En el Menú, accedemos a *Project->All*.
 - Click en *New*.
 - El formulario se carga correctamente.
 - Se rellenan los campos obligatorios.
 - Click en *Save*.
- Pruebas realizadas:
 - Prueba 1: crear proyecto
 - Resultado esperado (salida): se crea un proyecto con los datos incluidos.
 - Resultado obtenido: se ha creado un proyecto con el número PRJ0010683. OK.

7.1.3 Creación de Tarea de Proyecto

Caso 1:

- Descripción: a través de la *related link Create Test Phase* de proyectos creamos una nueva Tarea de Proyecto.
- Requisitos previos: hemos accedido al Proyecto PRJ0010683.
- Pasos a seguir:
 - Click en *Create Test Phase*.
 - El formulario de Tarea de Proyecto se carga correctamente.
 - Se rellenan los campos obligatorios.
 - Click en *Save*.
- Pruebas realizadas:
 - Prueba 1: crear Tarea de Proyecto a través de *Create Test Phase*
 - Resultado esperado: se crea una Tarea de Proyecto con los datos incluidos.
 - Resultado obtenido: se ha creado una Tarea de Proyecto con el número PRJTASK0015657. OK.

Caso 2:

- Descripción: a través del context menú *Project Task Creator* del formulario de proyecto creamos nuevas Tarea de Proyecto.
- Requisitos previos: hemos accedido al Proyecto PRJ0010683.

- Pasos a seguir:
 - *Click* con botón derecho en el banner.
 - *Click* en *Project Task Creator*.
 - La *UI Page* se carga correctamente.
 - Se rellenan los datos correctamente.
 - Se crean las tareas de proyecto marcadas.
- Pruebas realizadas:
 - Prueba 1: crear Tarea de Proyecto a través de *Project Task Creator*
 - Datos de entrada: *Number of Task: 2*.
 - Resultado esperado (salida): se crea un proyecto con los datos incluidos.
 - Resultado obtenido: se han creado dos Tareas de Proyecto con el número PRJTASK0015658 y PRJTASK0015659. OK.

7.1.4 Cálculo de campos *Planned Cost* y *Planned Effort* en las Tareas de Proyecto

Caso 1:

- Descripción: actualizar los campos de *planned effort* y *planned cost* relacionados con los roles (DC, SDC, PDM y BPC).
- Requisitos previos: acceder a la Tarea de Proyecto PRJTASK0015657.
- Pasos a seguir:
 - Editar campos de *planned effort*.
 - Editar campos de *planned cost*.
 - *Click* en *Save*.
- Pruebas realizadas:
 - Prueba 1: actualizar campos *planned effort* [rol] y *planned cost* [rol].
 - Datos de entrada: *Planned Effort DC: 2 horas. Planned Cost DC: 500€ Planned Cost SDC 700€*.
 - Resultado esperado: se actualizan los campos indicados. El campo *Planned Cost* se actualiza con la suma de los campos *Planned Cost* y el campo *Planned Effort* se actualiza con los campos de *Planned Effort*. En proyecto se actualizan los campos *Planned Cost* y *Planned Effort*. En Proyecto se actualizan *Planned Effort* y *Planned Cost (cost price)*.
 - Resultado obtenido: el registro se ha actualizado correctamente. OK.
 - Prueba 2: actualizar campos *planned effort* [rol] sin Hojas de Tarifas de Proyecto Asociadas.
 - Requisitos previos adicionales: no existen Hojas de Tarifas de Proyecto asociadas.
 - Datos de entrada: *Planned Effort DC: 2 horas. Planned Effort SDC: 2 horas*
 - Resultado esperado: la Tarea de Proyecto no se actualiza y se imprime el mensaje de error "*There are not Task Rate Cards associated with roles: DC, SDC*".
 - Resultado obtenido: se imprime el mensaje y el registro no se actualiza. OK.
 - Prueba 3: actualizar campos *planned effort* [rol] con Hojas de Tarifas de Proyecto Asociadas.
 - Requisitos previos adicionales: existen Hojas de Tarifas de Proyecto asociadas a todos los roles (creadas en el apartado 6.1.5).
 - Datos de entrada: *Planned Effort DC: 2 horas, Planned Effort SDC: 2 horas, Planned Effort DC: 2 horas, Planned Effort SDC: 2 horas*.

- Resultado esperado: se actualizan los campos indicados. En Tarea de Proyecto, los campos *Planned Cost [rol]* se actualizan con el valor del *Planned Effort [rol]* multiplicado por el valor de la Hoja de Tarifa de Gasto. El campo *Planned Cost* se actualiza con la suma de los campos *Planned Cost [rol]* y el campo *Planned Effort* se actualiza con los campos de *Planned Effort [rol]*. En Proyecto se actualizan *Planned Effort* y *Planned Cost (cost price)*.
- Resultado obtenido: el registro se ha actualizado correctamente. OK.
- Prueba 4: actualizar campos *planned effort [rol]* a 0.
 - Requisitos previos adicionales: existen Hojas de Tarifas de Proyecto asociadas a todos los roles.
 - Datos de entrada: *Planned Effort DC: 0 horas, Planned Effort SDC: 0 horas, Planned Effort DC: 0 horas, Planned Effort SDC: 0 horas.*
 - Resultado esperado: la Tarea de Proyecto no se actualiza y se imprime el mensaje de error “*At least one planned effort role have to be more than 0*”.
 - Resultado obtenido: se imprime el mensaje y el registro no se actualiza. OK.

7.1.5 Creación de hojas de tarifa de gasto

Caso 1:

- Descripción: crear registros en la tabla Hojas de Tarifas de Gasto.
- Pasos a seguir:
 - En el Menú, acceder a *SilverStorm Implementations > Rates > Task Rate Cards*.
 - Click en New.
 - Rellenar todos los campos.
 - Click en Save.
- Pruebas realizadas:
 - Prueba 1: creación de Hoja de Tarifa de Proyecto para DC y Tarea de Proyecto
 - Datos de entrada: *Name: PRJTASK0015657 DC; Rate: 10€; Condition: la Tarea de Proyecto es PRJTASK0015657 y el rol es DC.*
 - Resultado esperado: se crea una Hoja de Tarifa de Proyecto con los datos incluidos.
 - Resultado obtenido: OK.
 - Prueba 2: creación de Hoja de Tarifa de Proyecto para SDC y Proyecto.
 - Datos de entrada: *Name: PRJ0010683 SDC; Rate: 20€; Condition: el Proyecto es PRJ0010683 y el rol es SDC.*
 - Resultado esperado: se crea una Hoja de Tarifa de Proyecto con los datos incluidos.
 - Resultado obtenido: OK.
 - Prueba 3: creación de Hoja de Tarifa de Proyecto para PDM y BPC y Proyecto.
 - Datos de entrada: *Name: PRJ0010683 PDM y BPC; Rate: 50€; Condition: el Proyecto es PRJ0010683 y el rol es PDM o BPC.*
 - Resultado esperado: se crea una Hoja de Tarifa de Proyecto con los datos incluidos.
 - Resultado obtenido: OK.
 - Prueba 4: creación de Hoja de Tarifa de Proyecto para DC y Proyecto.
 - Datos de entrada: *Name: PRJ0010683 DC; Rate: 5€; Condition: el Proyecto es PRJ0010683 y el rol es DC.*

- Resultado esperado: se crea una Hoja de Tarifa de Proyecto con los datos incluidos.
- Resultado obtenido: OK.

7.1.6 Crear Time Cards

Caso 1:

- Descripción: crear registro en la tabla *Time Cards*.
- Requisitos previos: acceder a la Tarea de Proyecto PRJTASK0015657.
- Pasos a seguir:
 - Ir a la *related list Time Cards*.
 - Click en *New*.
 - Rellenar los campos.
 - Click en *Save*.
- Pruebas realizadas:
 - Prueba 1: crear *Time Card* para usuario del rol PDM sin Hoja de Tarifa de Proyecto asociada.
 - Requisitos previos adicionales: desactivar las Hojas de Tarifas de Gastos asociadas a la Tarea de proyecto y al proyecto del que proviene.
 - Datos de entrada: *Task: PRJTASK0015657; User: User PDM; Monday: 2 horas*
 - Resultado esperado: el registro no se crea y se imprime el mensaje *"This Time Card does not have a task rate associated"*.
 - Resultado obtenido: OK.
 - Prueba 2: crear *Time Card* para usuario del rol DC sin Hoja de Tarifa de Proyecto asociada.
 - Requisitos previos adicionales: desactivar las Hojas de Tarifas de Gastos asociadas a la Tarea de proyecto y al proyecto del que proviene.
 - Datos de entrada: *Task: PRJTASK0015657; User: User DC; Monday: 2 horas*.
 - Resultado esperado: el registro no se crea y se imprime el mensaje *"This Time Card does not have a task rate associated"*.
 - Resultado obtenido: OK.
 - Prueba 3: crear *Time Card* para usuario del rol SDC sin Hoja de Tarifa de Proyecto asociada.
 - Requisitos previos adicionales: desactivar las Hojas de Tarifas de Gastos asociadas a la Tarea de proyecto y al proyecto del que proviene.
 - Datos de entrada: *Task: PRJTASK0015657; User: User SDC; Monday: 2 horas*.
 - Resultado esperado: el registro no se crea y se imprime el mensaje *"This Time Card does not have a task rate associated"*.
 - Resultado obtenido: OK.
 - Prueba 4: crear *Time Card* para usuario del rol BPC sin Hoja de Tarifa de Proyecto asociada.
 - Requisitos previos adicionales: desactivar las Hojas de Tarifas de Gastos asociadas a la Tarea de proyecto y al proyecto del que proviene.
 - Datos de entrada: *Task: PRJTASK0015657; User: User BPC; Monday: 2 horas*.

- Resultado esperado: el registro no se crea y se imprime el mensaje “*This Time Card does not have a task rate associated*”.
 - Resultado obtenido: OK.
- Prueba 5: crear *Time Card* para usuario del rol PDM con Hoja de Tarifa de Proyecto asociada.
 - Requisitos previos adicionales: activar las Hojas de Tarifas de Gastos asociadas a la Tarea de proyecto y al proyecto del que proviene.
 - Datos de entrada: *Task: PRJTASK0015657; User: User PDM; Monday: 2 horas*
 - Resultado esperado: se crea un registro Time Card con los datos indicados TIM0000111615.
 - Resultado obtenido: OK.
- Prueba 6: crear *Time Card* para usuario del rol DC con Hoja de Tarifa de Proyecto asociada.
 - Requisitos previos adicionales: activar las Hojas de Tarifas de Gastos asociadas a la Tarea de proyecto y al proyecto del que proviene.
 - Datos de entrada: *Task: PRJTASK0015657; User: User DC; Monday: 2 horas*
 - Resultado esperado: se crea un registro Time Card con los datos indicados TIM0000111616.
 - Resultado obtenido: OK.
- Prueba 7: crear *Time Card* para usuario del rol SDC con Hoja de Tarifa de Proyecto asociada.
 - Requisitos previos adicionales: activar las Hojas de Tarifas de Gastos asociadas a la Tarea de proyecto y al proyecto del que proviene.
 - Datos de entrada: *Task: PRJTASK0015657; User: User SDC; Monday: 2 horas*
 - Resultado esperado: se crea un registro Time Card con los datos indicados TIM0000111617.
 - Resultado obtenido: OK.
- Prueba 8: crear *Time Card* para usuario del rol BPC con Hoja de Tarifa de Proyecto asociada.
 - Requisitos previos adicionales: activar las Hojas de Tarifas de Gastos asociadas a la Tarea de proyecto y al proyecto del que proviene.
 - Datos de entrada: *Task: PRJTASK0015657; User: User BPC; Monday: 2 horas*
 - Resultado esperado: se crea un registro Time Card con los datos indicados TIM0000111618.
 - Resultado obtenido: OK.

7.1.7 Aprobar Time Cards

Caso 1:

- Descripción: aprobar registro en la tabla *Time Cards*.
- Requisitos previos: acceder a un registro de la tabla *Time Card* en estado Pending.

- Pasos a seguir:
 - Acceder a la Time Card.
 - *Click* en *Approve*.
- Pruebas realizadas:
 - Prueba 1: aprobar Time Card con usuario con rol PDM.
 - Requisitos previos adicionales: acceder a Time Card TIM0000111615
 - Resultado esperado: el estado de la Time Card pasa a ser *Processed*. En la Tarea de Proyecto, se suma el tiempo total de la Time Card en Actual Effort PDM y Projected Effort PDM, y se añade el coste en Actual Cost PDM y Projected Cost PDM. En Actual Cost, Actual Effort, Projected Cost y Projected Effort se suman los valores correspondientes recientemente añadidos. En el Actual Cost, Actual Effort, *Actual Cost (cost Price)*, Projected Cost, Projected Effort y *Projected Cost (cost price)*. con los nuevos valores de la Tarea de Proyecto.
 - Resultado obtenido: OK.
 - Prueba 2: aprobar Time Card con usuario con rol DC.
 - Requisitos previos adicionales: acceder a Time Card TIM0000111616.
 - Resultado esperado: el estado de la Time Card pasa a ser *Processed*. En la Tarea de Proyecto, se suma el tiempo total de la Time Card en Actual Effort DC y Projected Effort DC, y se añade el coste en Actual Cost DC y Projected Cost DC. En Actual Cost, Actual Effort, Projected Cost y Projected Effort se suman los valores correspondientes recientemente añadidos. En el Proyecto se actualizan Actual Cost, Actual Effort, *Actual Cost (cost Price)*, Projected Cost, Projected Effort y *Projected Cost (cost price)*. con los nuevos valores de la Tarea de Proyecto.
 - Resultado obtenido: OK.
 - Prueba 3: aprobar Time Card con usuario con rol SDC.
 - Requisitos previos adicionales: acceder a Time Card TIM0000111617.
 - Resultado esperado: el estado de la Time Card pasa a ser *Processed*. En la Tarea de Proyecto, se suma el tiempo total de la Time Card en Actual Effort SDC y Projected Effort SDC, y se añade el coste en Actual Cost SDC y Projected Cost SDC. En Actual Cost, Actual Effort, Projected Cost y Projected Effort se suman los valores correspondientes recientemente añadidos. En el Proyecto se actualizan Actual Cost, Actual Effort, *Actual Cost (cost Price)*, Projected Cost, Projected Effort y *Projected Cost (cost price)* con los nuevos valores de la Tarea de Proyecto.
 - Resultado obtenido: OK.

7.1.8 Rechazar Time Cards

Caso 1:

- Descripción: aprobar registro en la tabla *Time Cards*.
- Requisitos previos: acceder a un registro de la tabla *Time Card* en estado Pending.

- Pasos a seguir:
 - Acceder a la Time Card.
 - *Click* en *Reject*.
- Pruebas realizadas:
 - Prueba 1: rechazar Time Card con usuario con rol BPC.
 - Requisitos previos adicionales: acceder a Time Card TIM00001116158.
 - Resultado esperado: el estado de la Time Card pasa a ser *Rejected*.
 - Resultado obtenido: OK.

7.1.9 Crear Planes de Recursos

Caso 1:

- Descripción: crear registro en la tabla Planes de Recursos desde Tareas de Proyecto.
- Requisitos previos: acceder a la Tarea de Proyecto PRJTASK0015657.
- Pasos a seguir:
 - Ir a la *related list* Planes de Recursos.
 - *Click* en *New*.
 - Rellenar los campos.
 - *Click* en *Save*.
- Pruebas realizadas:
 - Prueba 1: crear Plan de Recursos para usuario con rol BPC sin Hoja de Tarifas de Gastos asociada.
 - Requisitos previos adicionales: desactivar las Hojas de Tarifas de Gastos asociadas a la Tarea de proyecto y al proyecto del que proviene.
 - Datos de entrada: *Project*: PRJTASK0015657; *Resource Type*: *User Resource*; *User* BPC; *End Date*: después del día actual.
 - Resultado esperado: el registro no se crea y se imprime el mensaje “, *the selected user has the role BPC. Please complete the field with the associated rate*”.
 - Resultado obtenido: OK.
 - Prueba 2: crear Plan de Recursos para usuario con rol DC sin Hoja de Tarifas de Gastos asociada.
 - Requisitos previos adicionales: desactivar las Hojas de Tarifas de Gastos asociadas a la Tarea de proyecto y al proyecto del que proviene.
 - Datos de entrada: *Project*: PRJTASK0015657; *Resource Type*: *User Resource*; *User* DC; *End Date*: después del día actual.
 - Resultado esperado: el registro no se crea y se imprime el mensaje “, *the selected user has the role DC. Please complete the field with the associated rate*”.
 - Resultado obtenido: OK.
 - Prueba 3: crear Plan de Recursos para usuario con rol PDM sin Hoja de Tarifas de Gastos asociada.
 - Requisitos previos adicionales: desactivar las Hojas de Tarifas de Gastos asociadas a la Tarea de proyecto y al proyecto del que proviene.
 - Datos de entrada: *Project*: PRJTASK0015657; *Resource Type*: *User Resource*; *User* PDM; *End Date*: después del día actual.
 - Resultado esperado: el registro no se crea y se imprime el mensaje “, *the selected user has the role BPC. Please complete the field with the associated rate*”.
 - Resultado obtenido: OK.

- Prueba 4: crear Plan de Recursos para usuario con rol SDC sin Hoja de Tarifas de Gastos asociada.
 - Requisitos previos adicionales: desactivar las Hojas de Tarifas de Gastos asociadas a la Tarea de proyecto y al proyecto del que proviene.
 - Datos de entrada: *Project: PRJTASK0015657; Resource Type: User Resource; User SDC; End Date: después del día actual.*
 - Resultado esperado: el registro no se crea y se imprime el mensaje “, *the selected user has the role SDC. Please complete the field with the associated rate*”.
 - Resultado obtenido: OK.
- Prueba 5: crear Plan de Recursos para grupo añadiendo usuarios con distintos roles a la lista.
 - Datos de entrada: *Project: PRJTASK0015657; Resource Type: Group Resource; User SDC; Members Preference: specific members, Members List: User BPC, User SDC.*
 - Resultado esperado: al añadir el segundo usuario en *Member List*, se elimina automáticamente el segundo y aparece el mensaje “*The selected users have different roles, you need to create a new resource plan for the user with different role.*”
 - Resultado obtenido: OK.
- Prueba 6: crear Plan de Recursos para usuario con rol BPC con Hoja de Tarifas de Gastos asociada.
 - Requisitos previos adicionales: activar las Hojas de Tarifas de Gastos asociadas a la Tarea de proyecto y al proyecto del que proviene.
 - Datos de entrada: *Project: PRJTASK0015657; Resource Type: User Resource; User BPC; End Date: después del día actual.*
 - Resultado esperado: el registro se crea con los datos correspondientes. En el campo *Resource Rate* se carga el gasto asociado en la Hoja de Tarifas de Gasto. El campo *state* tiene valor *Planning*.
 - Resultado obtenido: OK.
- Prueba 7: crear Plan de Recursos para usuario con rol DC con Hoja de Tarifas de Gastos asociada.
 - Requisitos previos adicionales: activar las Hojas de Tarifas de Gastos asociadas a la Tarea de proyecto y al proyecto del que proviene.
 - Datos de entrada: *Project: PRJTASK0015657; Resource Type: User Resource; User DC; End Date: después del día actual.*
 - Resultado esperado: el registro se crea con los datos correspondientes. En el campo *Resource Rate* se carga el gasto asociado en la Hoja de Tarifas de Gasto. El campo *state* tiene valor *Planning*.
 - Resultado obtenido: OK.
- Prueba 8: crear Plan de Recursos para usuario con rol PDM con Hoja de Tarifas de Gastos asociada.
 - Requisitos previos adicionales: activar las Hojas de Tarifas de Gastos asociadas a la Tarea de proyecto y al proyecto del que proviene.
 - Datos de entrada: *Project: PRJTASK0015657; Resource Type: User Resource; User PDM; End Date: después del día actual.*

- Resultado esperado: el registro se crea con los datos correspondientes. En el campo Resource Rate se carga el gasto asociado en la Hoja de Tarifas de Gasto. El campo state tiene valor Planning.
 - Resultado obtenido: OK.
- Prueba 9: crear Plan de Recursos para usuario con rol SDC con Hoja de Tarifas de Gastos asociada.
 - Requisitos previos adicionales: activar las Hojas de Tarifas de Gastos asociadas a la Tarea de proyecto y al proyecto del que proviene.
 - Datos de entrada: *Project*: PRJTASK0015657; *Resource Type*: *User Resource*; *User* SDC; *End Date*: después del día actual.
 - Resultado esperado: el registro se crea con los datos correspondientes. En el campo Resource Rate se carga el gasto asociado en la Hoja de Tarifas de Gasto. El campo state tiene valor Planning.
 - Resultado obtenido: OK.

Caso 2:

- Descripción: crear registro en la tabla Planes de Recursos desde Demanda.
- Requisitos previos: acceder a la Demanda y abrir una demanda.
- Pasos a seguir:
 - Ir a la *related list* Planes de Recursos.
 - Click en *New*.
 - Rellenar los campos.
 - Click en *Save*.
- Pruebas realizadas:
 - Prueba 1: crear Plan de Recursos para una demanda con el campo *type* con valor *Project*.
 - Requisitos previos adicionales: la demanda es de tipo *Project*.
 - Datos de entrada: *End Date*: después del día actual.
 - Resultado esperado: *Resource Type* = *Role Resource*; *Task type*: *Demand*; Opciones *Project Task*: 01-Initiate, 02-Prepare, 03-Create, 04-Transition, 05-Closed, 06-Management, Resource Rate: se rellena manualmente.
 - Resultado obtenido: OK. Todos los campos cumplen los parámetros esperados. El registro se crea correctamente con ellos.
 - Prueba 2: crear Plan de Recursos para una demanda con el campo *type* con valor *Service Improvement – Enhancement*.
 - Requisitos previos adicionales: la demanda es de tipo *Service Improvement – Enhancement*.
 - Datos de entrada: *End Date*: después del día actual.
 - Resultado esperado: *Resource Type* = *Role Resource*; *Task type*: *Demand*; Opciones *Project Task*: 01-Initiate, 02-Prepare, 03-Create, 04-Transition, 05-Closed, 06-Management, Resource Rate: se rellena manualmente.
 - Resultado obtenido: OK. Todos los campos cumplen los parámetros esperados. El registro se crea correctamente con ellos.
 - Prueba 2: crear Plan de Recursos para una demanda con el campo *type* con valor *Service Improvement*.
 - Requisitos previos adicionales: la demanda es de tipo *Service Improvement – Enhancement*.

- Datos de entrada: *End Date*: después del día actual.
- Resultado esperado: *Resource Type = Role Resource; Task type: Demand*; Opciones *Project Task*: 00 - SERVICE TRANSITION, 01 - ADVANCE PLATFORM SUPPORT, 02 - BASIC CONFIGURATION SUPPORT, 03 - ENHANCEMENT SERVICES, 04 - ARCHITECTURAL SERVICES, 05 - UPGRADE, 06 – MANAGEMENT; *Resource Rate*: se rellena manualmente.
- Resultado obtenido: OK. Todos los campos cumplen los parámetros esperados. El registro se crea correctamente con ellos.

7.1.10 Creación de Asignaciones de Recursos:

Caso 1:

- Descripción: crear registros en la tabla Asignación de Recursos asociado a un Plan de Recursos creado previamente.
- Requisitos previos: acceder a un registro de la tabla Plan de Recursos para el que no se han creado las asignaciones de recursos.
- Pasos a seguir:
 - Ir a *SilverStorm Implementation>Resource Management>Resource Plans*.
 - Acceder a un registro.
 - *Click en Create Soft Allocation*.
- Pruebas realizadas:
 - Prueba 1: crear las Asignaciones de Recursos para un Plan de Recursos asociado a una Tarea de Proyecto para un usuario DC.
 - Requisitos previos adicionales: el campo *Task* del Plan de Recursos es una Tarea de Proyecto, el usuario asociado tiene rol DC.
 - Resultado esperado: en la *related list Resource Allocation* se generan los registros que Asignan los recursos al usuario dividiéndolo semanalmente La suma total de los campos *requested hours* de estos registros son iguales a las horas planificadas. En la Tarea de Proyecto asociada los campos *projected effort DC* y *projected cost DC* se actualizan sumando las nuevas horas y coste previsto. Además, los campos *projected effort* y *projected cost* se les suman también. En el Proyecto asociado, se actualizan los campos *projected effort*, *projected cost* y *projected cost (cost price)*.
 - Resultado obtenido: OK.
 - Prueba 2: crear las Asignaciones de Recursos para un Plan de Recursos asociado a una Tarea de Proyecto para un usuario PDM.
 - Requisitos previos adicionales: el campo *Task* del Plan de Recursos es una Tarea de Proyecto, el usuario asociado tiene rol PDM.
 - Resultado esperado: en la *related list Resource Allocation* se generan los registros que Asignan los recursos al usuario dividiéndolo semanalmente, la suma total de los campos *requested hours* de estos registros son iguales a las horas planificadas. En la Tarea de Proyecto asociada los campos *projected effort PDM* y *projected cost PDM* se actualizan sumando las nuevas horas y coste previsto. Además, los campos *projected effort* y *projected cost* se les suman también. En el Proyecto asociado, se actualizan los campos *projected effort*, *projected cost* y *projected cost (cost price)*.
 - Resultado obtenido: OK

- Prueba 3: crear las Asignaciones de Recursos para un Plan de Recursos asociado a una demanda.
 - Requisitos previos adicionales: el campo Task del Plan de Recursos es una Demanda.
 - Resultado esperado: en la *related list Resource Allocation* se generan los registros que Asignan los recursos al usuario dividiéndolo semanalmente, la suma total de los campos *requested hours* de estos registros son iguales a las horas planificadas.
 - Resultado obtenido: OK.

7.1.11 Solicitar Plan de Recursos

Caso 1:

- Descripción: un plan de Recursos pasa a estado Requested.
- Requisitos previos: acceder a un registro de la tabla Plan de Recursos en estado *Planning*.
- Pasos a seguir:
 - Ir a *SilverStorm Implementation>Resource Management>Resource Plans*.
 - Acceder a un registro.
 - *Click en Request*.
- Pruebas realizadas:
 - Prueba 1: solicitar la aprobación de un Plan de Recursos.
 - Resultado esperado: el estado del Plan de Recursos pasa a ser Requested.
 - Resultado obtenido: OK.

7.1.12 Asignar Plan de Recursos

- Descripción: un plan de Recursos pasa a estado Allocated.
- Requisitos previos: acceder a un registro de la tabla Plan de Recursos en estado *Requested*.
- Pasos a seguir:
 - Ir a *SilverStorm Implementation>Resource Management>Resource Plans*.
 - Acceder a un registro.
 - *Click en Confirm and Allocation*.
- Pruebas realizadas:
 - Prueba 1: solicitar la aprobación de un Plan de Recursos.
 - Resultado esperado: el estado del Plan de Recursos pasa a ser Allocated.
 - Resultado obtenido: OK.

7.1.13 Rechazar Plan de Recursos

- Descripción: un plan de Recursos es rechazado.
- Requisitos previos: acceder a un registro de la tabla Plan de Recursos en estado *Requested*.
- Pasos a seguir:
 - Ir a *SilverStorm Implementation>Resource Management>Resource Plans*.
 - Acceder a un registro.
 - *Click en Reject*.
- Pruebas realizadas:

- Prueba 1: solicitar la aprobación de un Plan de Recursos.
 - Resultado esperado: el estado del Plan de Recursos pasa a ser *Planning*.
 - Resultado obtenido: OK.

7.1.14 Finalización de Plan de Recursos

Caso 1:

- Descripción: completar un Plan de Recursos
- Requisitos previos: el Plan de Recursos se encuentra en estado *Allocated*
- Pasos a seguir:
 - Ir a *SilverStorm Implementation>Resource Management>Resource Plans*
 - Acceder a un registro
 - *Click en Complete.*
- Pruebas realizadas:
 - Prueba 1: completar un Plan de Recursos asociado a una Tarea de Proyecto con *user* con rol DC.
 - Requisitos previos adicionales: el campo *task* contiene un registro de tipo Tarea de Proyecto y el campo *user* un registro con rol DC.
 - Resultado esperado: en el Plan de Recursos, el estado pasa a *Complete*. En la Tarea de Proyecto asociada los campos *projected effort DC* y *projected cost DC* se actualizan restando horas y coste que estaban asignadas a ese plan. Además, los campos *projected effort* y *projected cost* se les restan también. En el Proyecto asociado, se actualizan los campos *projected effort*, *projected cost* y *projected cost (cost price)*.
 - Resultado obtenido: OK.
 - Prueba 2: completar un Plan de Recursos asociado a una Tarea de Proyecto con *user* con rol PDM.
 - Requisitos previos adicionales: el campo *task* contiene un registro de tipo Tarea de Proyecto y el campo *user* un registro con rol PDM.
 - Resultado esperado: en el Plan de Recursos, el estado pasa a *Complete*. En la Tarea de Proyecto asociada los campos *projected effort PDM* y *projected cost PDM* se actualizan restando horas y coste que estaban asignadas a ese plan. Además, los campos *projected effort* y *projected cost* se les restan también. En el Proyecto asociado, se actualizan los campos *projected effort*, *projected cost* y *projected cost (cost price)*.
 - Resultado obtenido: OK.
 - Prueba 3: completar un Plan de Recursos asociado a una Demanda.
 - Requisitos previos adicionales: el campo *task* contiene un registro de tipo Demanda.
 - Resultado esperado: en el Plan de Recursos, el estado pasa a *Complete*.
 - Resultado obtenido: OK.

Caso 2:

- Descripción: cancelar un Plan de Recursos .
- Requisitos previos: el Plan de Recursos se encuentra en estado *Planning*.

- Pasos a seguir:
 - Ir a *SilverStorm Implementation>Resource Management>Resource Plans*.
 - Acceder a un registro.
 - *Click en Cancel..*
- Pruebas realizadas:
 - Prueba 1: completar un Plan de Recursos asociado a una Tarea de Proyecto con user con rol SDC.
 - Requisitos previos adicionales: el campo *task* contiene un registro de tipo Tarea de Proyecto y el campo *user* un registro con rol SDC.
 - Resultado esperado: en el Plan de Recursos, el estado pasa a Cancelled. En la Tarea de Proyecto asociada los campos *projected effort* SDC y *projected cost* SDC se actualizan restando horas y coste que estaban asignadas a ese plan. Además, los campos *projected effort* y *projected cost* se les restan también. En el Proyecto asociado, se actualizan los campos *projected effort*, *projected cost* y *projected cost (cost price)*.
 - Resultado obtenido: OK.
 - Prueba 2: completar un Plan de Recursos asociado a una Tarea de Proyecto con user con rol BPC.
 - Requisitos previos adicionales: el campo *task* contiene un registro de tipo Tarea de Proyecto y el campo *user* un registro con rol BPC.
 - Resultado esperado: en el Plan de Recursos, el estado pasa a Cancelled. En la Tarea de Proyecto asociada los campos *projected effort* BPC y *projected cost* BPC se actualizan restando horas y coste que estaban asignadas a ese plan. Además, los campos *projected effort* y *projected cost* se les restan también. En el Proyecto asociado, se actualizan los campos *projected effort*, *projected cost* y *projected cost (cost price)*.
 - Resultado obtenido: OK.
 - Prueba 3: completar un Plan de Recursos asociado a una Demanda.
 - Requisitos previos adicionales: el campo *task* contiene un registro de tipo Demanda.
 - Resultado esperado: en el Plan de Recursos, el estado pasa a Cancelled.
 - Resultado obtenido: OK.

7.2 Mejoras para Gestión de Proyectos

Durante el proceso de plan de pruebas se han observado algunas mejoras que se podrían realizar en el proceso para sacar un mayor partido a los desarrollos realizados:

- Añadir las horas y el coste del Plan de Recursos antes de crear las asignaciones: en el desarrollo las horas de los Planes de Recursos se añaden a los campos correspondientes en Tareas de Proyecto al crear las Asignaciones de Recursos. Si se hace desde que se crea el plan, se podrían prever los esfuerzos y costes con mayor facilidad.
- Crear unos ciclos de vida para Proyectos y Tareas de Proyectos más lineal: en estos registros se puede pasar de unos estados a otros como se quiera. Organizando mejor estos cambios de estados, se mediría mejor en qué momento está el proyecto.

7.3 Datos de Prueba *Test Management*

7.3.1 Pasos previos

Antes de realizar el proceso crearemos registros en distintas entidades Entorno de para tenerlos disponibles de cara a hacer las pruebas.

Primero en la tabla de compañías (*Company*) creamos los siguientes registros:

Name	Code
TFG DEMO 1	tfg_Number_1
TFG DEMO 2	tfg_Number_2
TFG DEMO 3	tfg_Number_3

Tabla 7.3.1 Registros de Compañías

7.3.2 Creación de Conjuntos de *Test*

Caso 1:

- Descripción: creación de un Conjunto de *Test*.
- Pasos a seguir:
 - Ir a *Test Management 2.0>Test Set*.
 - Click en New.
 - Rellenar los campos.
 - *Click en Save*.
- Pruebas realizadas:
 - Prueba 1: crear un Conjunto de *Test*.
 - Resultado esperado: se crea un Conjunto de *Test* con los datos indicados.
 - Resultado obtenido: OK.

7.3.3 Creación de *Test* y Versiones de *Test* versión 1

Caso 1:

- Descripción: creación de un *Test* y Versión de *Test* 1 desde la tabla *Test*.
- Pasos a seguir:
 - Ir a *Test Management 2.0>Test*.
 - Click en New.
 - Se abre el formulario de Versión de *Test*.
 - Rellenar los campos.
 - *Click en Save*.
- Pruebas realizadas:
 - Prueba 1: crear un *Test* desde la lista de la tabla *Test*.
 - Resultado esperado: se crea un *Test* y una Version de *Test* con el valor 1 en el campo version.
 - Resultado obtenido: OK.

Caso 2:

- Descripción: creación de un *Test* y Versión de *Test* 1 desde un Conjunto de *Test*.
- Pasos a seguir:
 - Ir a *Test Management 2.0>Test Set*.
 - Abrir un Conjunto de *Test*.
 - Ir a la Related List *Test*.

- *Click en New.*
- Rellenar los campos.
- *Click en Save.*
- Pruebas realizadas:
 - Prueba 1: crear un *Test* desde un Conjunto de *Test*.
 - Resultado esperado: se crea un *Test* y una *Version de Test* con el valor 1 en el campo *version*. El campo *company* es el mismo que el conjunto de *Test* y el campo *Test Set* es el conjunto de *Test* desde el que se ha creado la prueba. Ambos son no editables.
 - Resultado obtenido: OK.

7.3.4 Asignación de *Test ya creados* a Conjuntos de *Test*

Caso 1:

- Descripción: creación de un *Test* y *Versión de Test* 1 desde la tabla *Test*.
- Pasos a seguir:
 - Ir a *Test Management 2.0>Test*.
 - Seleccionar *Test* para añadir al Conjunto de *Test*.
 - Click en *Add to Test Set*.
 - Se abre una UI Page.
 - Seleccionar el Conjunto de *Test*.
 - *Click en Add*.
- Pruebas realizadas:
 - Prueba 1: no seleccionar ningún *Test* antes de pulsar *Add To Test Set*.
 - Resultado esperado: se imprime en la pantalla el mensaje "*Please, select a Test*"
 - Resultado obtenido: OK.
 - Prueba 2: seleccionamos *Test* antes de pulsar *Add To Test Set*. La *company* del Conjunto de *Test* es distinta que la del *Test*.
 - Resultado esperado: no se modifica el campo *Test Set* de los *Test* seleccionados.
 - Resultado obtenido: OK.
 - Prueba 3: seleccionamos *Test* antes de pulsar *Add To Test Set*. La *company* del Conjunto de *Test* es la misma que la de los *Test*.
 - Resultado esperado: se modifica el campo *Test Set* de los *Test* seleccionados cambiándolo al Conjunto de *Test* elegido.
 - Resultado obtenido: OK.

7.3.5 Actualización campo *code* en *Test desde Conjunto de Test*

- Descripción: actualización del campo *code* de los *Test* a través del botón *Update Test Codes*.
- Requisitos previos: el Conjunto de *Test* tiene varios *Test* asociados.
- Pasos a seguir:
 - Ir a *Test Management 2.0>Test Set*.
 - Acceder a un Conjunto de *Test*.
 - *Click en Update Test Codes*.
 - Se carga la ventana "*Select Code Condition*".
 - Rellenar los campos de "*Select Code Condition*".
 - *Click en Accept*.

- Pruebas realizadas:
 - Prueba 1: dejar vacíos los campos obligatorios *Prefix* y *Number of digits*.
 - Datos de entrada: *Prefix:""; Number of digits:""; Initial Value.*
 - Resultado esperado: aparece un mensaje de error con las normas para rellenar los campos.
 - Resultado obtenido: OK.
 - Prueba 2: el campo *Number of digits* no es un número.
 - Datos de entrada: *Prefix: ap ; Number of digits: ap ; Initial Value.*
 - Resultado esperado: aparece un mensaje de error con las normas para rellenar.
 - Resultado obtenido: OK.
 - Prueba 3: el campo *Number of digits* es menor que 0.
 - Datos de entrada: *Prefix: ap ; Number of digits: -1 ; Initial Value.*
 - Resultado esperado: aparece un mensaje de error con las normas para rellenar.
 - Resultado obtenido: OK
 - Prueba 4: el campo *Number of digits* es un número decimal.
 - Datos de entrada: *Prefix: ap; Number of digits: 3,4 ; Initial Value.*
 - Resultado esperado: aparece un mensaje de error con las normas para rellenar.
 - Resultado obtenido: OK.
 - Prueba 5: el campo *Initial Value* no es un número.
 - Datos de entrada: *Prefix: ap ; Number of digits: 2 ; Initial Value: ap.*
 - Resultado esperado: aparece un mensaje de error con las normas para rellenar.
 - Resultado obtenido: OK.
 - Prueba 6: el campo *Initial Value* es menor que 0.
 - Datos de entrada: *Prefix: ap ; Number of digits: 2 ; Initial Value:-1.*
 - Resultado esperado: aparece un mensaje de error con las normas para rellenar.
 - Resultado obtenido: OK.
 - Prueba 7: el campo *Initial Value* es un número decimal.
 - Datos de entrada: *Prefix: ap; Number of digits: 2 ; Initial Value: 3,4.*
 - Resultado esperado: aparece un mensaje de error con las normas para rellenar.
 - Resultado obtenido: OK.
 - Prueba 8: el campo *Initial Value* es un número mayor que 10 elevado al campo *Number of digits*.
 - Datos de entrada: *Prefix: ap; Number of digits: 2 ; Initial Value: 102.*
 - Resultado esperado: aparece un mensaje de error con las normas para rellenar.
 - Resultado obtenido: OK.
 - Prueba 9: los campos se han rellenado correctamente
 - Requisitos previos adicionales: el Conjunto de *Test* tiene 3 *Test* asociados.
 - Datos de entrada: *Prefix: ap; Number of digits: 2; Initial Value: 10.*
 - Resultado esperado: los *Test* se rellenan con los códigos ap10, ap11 y ap12.
 - Resultado obtenido: OK.

7.3.6 Creación de Pasos

Caso 1

- Descripción: creación de un Paso.
- Pasos a seguir:
 - Ir a *Test Management 2.0>Test*.
 - Acceder a un *Test*.
 - Ir a la *related list Test Version*.
 - Acceder a la *Version de Test*.
 - En el formulario click en *Add Step*.
 - Rellenar los campos.
- Pruebas realizadas:
 - Prueba 1: la *Version de Test* está en un estado distinto de *Draft*.
 - Resultado esperado: No se pueden crear Pasos.
 - Resultado obtenido: OK.
 - Prueba 2: la *Version de Test* está en estado de *Draft*, no se marca *Needs Verification*.
 - Resultado esperado: se crea un Paso que no necesita verificación.
 - Resultado obtenido: OK.
 - Prueba 3: la *Version de Test* está en estado de *Draft*, se marca *Needs Verification*.
 - Resultado esperado: se crea un Paso que necesita verificación.
 - Resultado obtenido: OK.

7.3.7 Ordenar Pasos

Caso 1

- Descripción: ordenar los pasos a través del *Test Step*.
- Requisitos previos: la *Versión de Test* está en estado *Draft*.
- Pasos a seguir:
 - Ir a *Test Management 2.0>Test*.
 - Acceder a un *Test*.
 - Ir a la *related list Test Version*.
 - Acceder a la *Version de Test*.
 - En *Test Steps*, se mueven los Pasos entre sí.
- Pruebas realizadas:
 - Prueba 1: en la *Versión de Test* se mueven los Pasos entre si.
 - Resultado esperado: El order de los registros Pasos cambia teniendo el del Paso que esta primero el menor y el que está el último el mayor.
 - Resultado obtenido: OK.

Caso 2

- Descripción: ordenar los Pasos a través de los propios registros Pasos.
- Requisitos previos: la *Versión de Test* está en estado *Draft*.
- Pasos a seguir:
 - Ir a *Test Management 2.0>Test*.
 - Acceder a un *Test*.
 - Ir a la *related list Test Version*.
 - Acceder a la *Version de Test*.

- Ir a la *related list Steps*.
- Modificar el campo *order*.
- *Click Update*.
- Pruebas realizadas:
 - Prueba 1: acceder a un Paso que no tenga el *order* más alto, y ponerle el más alto.
 - Resultado esperado: En la Versión de *Test*, el Paso modificado aparece el último de la lista.
 - Resultado obtenido: OK.
 - Prueba 2: acceder a un Paso que no tenga el *order* más bajo, y ponerle el más bajo.
 - Resultado esperado: En la Versión de *Test*, el Paso modificado aparece el primero de la lista.
 - Resultado obtenido: OK.

7.3.8 Crear Versiones de *Test* procedentes de otra

Caso 1

- Descripción: ordenar los pasos a través del *Test Step*.
- Requisitos previos: la Versión de *Test* está en estado *Draft* o *Ready*.
- Pasos a seguir:
 - Ir a *Test Management 2.0>Test*.
 - Acceder a un *Test*.
 - Ir a la *related list Test Version*.
 - Acceder a la Versión de *Test*.
 - *Click Create New Version*.
- Pruebas realizadas:
 - Prueba 1: el *Test* al que está asociado la Versión de *Test* donde pulsamos el botón solo tiene una Versiones de *Test*.
 - Resultado esperado: se crea una nueva Versión de *Test* y todos los pasos asociados a ella iguales a la versión 1. El número del campo versión es 2. En el *Test* asociado, *la Test versión* es el 2.
 - Resultado obtenido: OK.
 - Prueba 2: el *Test* al que está asociado la Versión de *Test* donde pulsamos el botón tiene tres Versiones de *Test*.
 - Resultado esperado: se crea una nueva Versión de *Test* y todos los pasos asociados a ella iguales a la versión 1. El número del campo versión es 4. En el *Test* asociado, *la Test versión* es el 4.
 - Resultado obtenido: OK.

7.3.9 Establecer Versión de *Test* como la actual en ejecución

Caso 1

- Descripción: pasar Versiones de *Test* a estado *Ready*.
- Requisitos previos: la Versión de *Test* está en estado *Draft*. El *Test* tiene al menos un Paso con el campo *Needs Verification* marcado.
- Pasos a seguir:
 - Ir a *Test Management 2.0>Test*.
 - Acceder a un *Test*.

- Ir a la *related list Test Version*.
- Acceder a la Version de *Test*.
- *Click Ready*.
- Pruebas realizadas:
 - Prueba 1: pasar a *Ready* la Version de *Test* tiene el campo *versión* con valor 2.
 - Resultado esperado: el estado de la Version de *Test* pasa a *Ready* y todos sus campos a no editables. En el *Test* asociado, el campo *runnable version* pasa a ser 2 .
 - Resultado obtenido: OK.
 - Prueba 2: pasar a *Ready* la Version de *Test* tiene el campo *versión* con valor 3.
 - Requisitos previos adicionales: el *Test* tiene ya una Versión de *Test* asociada con estado *Ready*.
 - Resultado esperado: el estado de la Version de *Test* pasa a *Ready* y todos sus campos no editables. En el *Test* asociado, el campo *runnable version* pasa a ser 3. El estado de la Version de *Test* que estaba previamente en *Ready* pasa a ser *Retired*.
 - Resultado obtenido: OK.
 - Prueba 3: pasar a *Ready* la Version de *Test* que no tiene asociado un Paso con el campo *needs verification* marcado.
 - Resultado esperado: en la pantalla se imprime el mensaje “*The Test has to have at least one verification step to be marked as Ready*”.
 - Resultado obtenido: OK.
-

7.3.10 Establecer Version de *Test* en estado Draft

Caso 1

- Descripción: pasar Versiones de *Test* a estado *Draft*.
- Requisitos previos: la Versión de *Test* está en estado *Ready*.
- Pasos a seguir:
 - Ir a *Test Management 2.0>Test*.
 - Acceder a un *Test*.
 - Ir a la *related list Test Version*.
 - Acceder a la Version de *Test*.
 - *Click Edit Version*.
- Pruebas realizadas:
 - Prueba 1: pasar a *Draft* una Versión de *Test*.
 - Resultado esperado: el estado de la Version de *Test* pasa a *Draft* y sus campos son editables.
 - Resultado obtenido: OK.

7.3.11 Crear Entorno de *Test*

Caso 1

- Descripción: crear Entorno de *Test* desde la tabla *Environment Test*.
- Pasos a seguir:
 - Ir a *Test Management 2.0> Environment*.
 - Click en *New*.

- Rellenar los campos.
- *Click en Save.*
- Pruebas realizadas:
 - Prueba 1: crear un Entorno de *Test* a través de su tabla.
 - Resultado esperado: se crea un registro Entorno de *Test*.
 - Resultado obtenido: OK.

7.3.12 Crear Plan de *Test*

Caso 1

- Descripción: crear Plan de *Test* desde la tabla Planes de *Test*.
- Pasos a seguir:
 - Ir a *Test Management 2.0 > Test Plan*.
 - Click en *New*.
 - Rellenar los campos.
 - *Click en Save*.
- Pruebas realizadas:
 - Prueba 1: crear un Plan de *Test* a través de su tabla.
 - Resultado esperado: se crea un registro Plan de *Test*.
 - Resultado obtenido: OK.

7.3.13 Crear Ciclo de *Test*

Caso 1

- Descripción: crear Ciclo de *Test* desde un Plan de *Test*.
- Pasos a seguir:
 - Ir a *Test Management 2.0 > Test Plan*.
 - Acceder a un Plan de *Test*.
 - Ir a la *related list Test Cycles*.
 - *Click en New*.
 - Rellenar los campos.
 - *Click en Save*.
- Pruebas realizadas:
 - Prueba 1: crear un Ciclo de *Test* a través de un Plan de *Test*.
 - Resultado esperado: se crea un registro Ciclo de *Test* con el campo *Test Plan* poblado con el Plan de *Test* desde el que se ha creado.
 - Resultado obtenido: OK.

7.3.14 Crear Software de Ejecución de *Test*

Caso 1

- Descripción: crear Software de Ejecución de *Test* desde un Ciclo de *Test*.
- Requisitos previos: el Plan de *Test* al menos un Ciclo de *Test* asociado.
- Pasos a seguir:
 - Ir a *Test Management 2.0 > Test Plan*.
 - Acceder a un Plan de *Test*.
 - Ir a la *related list Test Cycles*.
 - *Acceder a un Ciclo de Test*.

- Ir a la *related list Suite Execution Suite*.
- Click en *New*.
- Rellenar los campos.
- Click en *Save*.
- Pruebas realizadas:
 - Prueba 1: crear un Software de Ejecución de *Test* a través de un Ciclo de *Test*.
 - Resultado esperado: se crea un registro Software de Ejecución de *Test* con el campo *Test Cycle* populado con el Ciclo de *Test* desde el que se ha creado.
 - Resultado obtenido: OK.

Caso 2

- Descripción: crear Software de Ejecución de *Test* desde el módulo *Test Execution Suite*.
- Requisitos previos: el Plan de *Test* al menos un Ciclo de *Test* asociado.
- Pasos a seguir:
 - Ir a *Test Management 2.0> Test Execution Suite*.
 - Click en *New*.
 - Rellenar los campos.
 - Click en *Save*.
- Pruebas realizadas:
 - Prueba 1: crear un Software de Ejecución de *Test* a través del módulo *Test Execution Suite*.
 - Resultado esperado: se crea un registro Software de Ejecución de *Test*.
 - Resultado obtenido: OK.

Caso 3

- Descripción: crear Software de Ejecución de *Test* desde el módulo *Stories Execution Suite*.
- Requisitos previos: el Plan de *Test* al menos un Ciclo de *Test* asociado.
- Pasos a seguir:
 - Ir a *Test Management 2.0> Stories Execution Suite*.
 - Click en *New*.
 - Rellenar los campos.
 - Click en *Save*.
- Pruebas realizadas:
 - Prueba 1: crear un Software de Ejecución de *Test* a través del módulo *Stories Execution Suite*.
 - Resultado esperado: se crea un registro Software de Ejecución de *Test*.
 - Resultado obtenido: OK.

Caso 4

- Descripción: crear Software de Ejecución de *Test* desde un registro *Story*.
- Requisitos previos: el Plan de *Test* al menos un Ciclo de *Test* asociado.
- Pasos a seguir:
 - Ir a *SilverStorms Implementation>Stories>Open Stories*
 - Acceder a una *Story*.
 - Ir a la *related list Suite Execution Suite*.
 - Click en *New*.

- Rellenar los campos.
- *Click en Save.*
- Pruebas realizadas:
 - Prueba 1: crear un Software de Ejecución de *Test* desde una story.
 - Resultado esperado: se crea un registro Software de Ejecución de *Test*. El campo story del registro está poblado con la story desde la que se ha creado.
 - Resultado obtenido: OK.

7.3.15 Asignación de *Test* a Software de Ejecución de *Test*

Caso 1

- Descripción: agregar registros Conjuntos de *Test* a un Software de Ejecución de *Test*.
- Pasos a seguir:
 - Ir a *Test Management 2.0> Test Execution Suite*.
 - Acceder a un Software de Ejecución de *Test*.
 - *Click Add Test Sets*.
 - Seleccionar Conjuntos de *Test*.
 - *Click en Add to Execution Suite* .
- Pruebas realizadas:
 - Prueba 1: pulsar Add to Execution Suite sin seleccionar Conjuntos de *Test*
 - Resultado esperado: en la pantalla se imprime el mensaje “Please Select *Test Set*”.
 - Resultado obtenido: OK.
 - Prueba 2: pulsar Add to Execution Suite seleccionando antes Conjuntos de *Test*.
 - Resultado esperado: se crea un registro en la tabla Asignación de Ejecución de *Test* por cada *Test* que contienen los Conjuntos de *Test*, que relaciona el *Test* con el Software de Ejecución de *Test*. El campo Assigned To en estos registros es el mismo que el de Software de Ejecución de *Test*.
 - Resultado obtenido: OK.

Caso 2

- Descripción: agregar *Test* individuales a un Software de Ejecución de *Test*.
- Pasos a seguir:
 - Ir a *Test Management 2.0> Test Execution Suite*.
 - Acceder a un Software de Ejecución de *Test*.
 - *Click Add Test*.
 - Seleccionar Conjuntos de *Test*.
 - *Click en Add to Execution Suite*.
- Pruebas realizadas:
 - Prueba 1: pulsar Add to Execution Suite sin seleccionar *Test*.
 - Resultado esperado: en la pantalla se imprime el mensaje “Please Select *Test*”.
 - Resultado obtenido: OK.
 - Prueba 2: pulsar Add to Execution Suite seleccionando antes *Test*.
 - Resultado esperado: se crea un registro en la tabla Asignación de Ejecución de *Test* por cada *Test* seleccionado, que relaciona el *Test* con el Software de Ejecución de *Test*. El campo *Assigned To* en estos registros es el mismo que el de Software de Ejecución de *Test*.

- Resultado obtenido: OK.

7.3.16 Ejecución de *Test Run*

Caso 1

- Descripción: ejecutar *Test Runs* desde una Versión de *Test*.
- Requisitos previos: la Versión de *Test* está en estado *Ready*.
- Pasos a seguir:
 - Ir a *Test Management 2.0>Test*.
 - Acceder a un *Test*.
 - Ir a la *related list Test Version*.
 - Acceder a la Versión de *Test*.
 - Click *Run*.
 - Rellenar los campos.
 - Click *Run*.
 - Completar los resultados de los Pasos que necesitan verificación.
 - Click *Done*.
- Pruebas realizadas:
 - Prueba 1: ejecutar una prueba marcando todos los Pasos que necesitan verificación como *Passed*.
 - Resultado esperado: se crea un registro *Test Run* con el campo *total Test* en 1 y el campo *passed* en 1. Se crea un registro Resultado de *Test* con campo *Result* con valor *passed*. Los pasos que necesitan verificación aparecen en verde si son "*passed*", en amarillo si son "*blocked*", en rojo si son "*failed*" y en negro si son "*not executed*".
 - Resultado obtenido: OK.
 - Prueba 2: ejecutar una prueba marcando el primer Paso que necesita verificación como *Failed*.
 - Resultado esperado: se crea un registro *Test Run* con el campo *total Test* en 1 y el campo *failed* en 1. Se crea un registro Resultado de *Test* con campo *Result* con valor *failed*. Los pasos que necesitan verificación aparecen en verde si son "*passed*", en amarillo si son "*blocked*", en rojo si son "*failed*" y en negro si son "*not executed*".
 - Resultado obtenido: OK.
 - Prueba 3: ejecutar una prueba marcando el primer Paso que necesita verificación como *Blocked*.
 - Resultado esperado: se crea un registro *Test Run* con el campo *total Test* en 1 y el campo *blocked* en 1. Se crea un registro Resultado de *Test* con campo *Result* con valor *blocked*. Los pasos que necesitan verificación aparecen en verde si son "*passed*", en amarillo si son "*blocked*", en rojo si son "*failed*" y en negro si son "*not executed*".
 - Resultado obtenido: OK.
 - Prueba 4: ejecutar una prueba sin marcar todos los Pasos que necesitan.
 - Resultado esperado: se crea un registro *Test Run* con el campo *total Test* en 1 y el campo *not finished* en 1. Se crea un registro Resultado de *Test* con campo *Result* con valor *not finished*. Los pasos que necesitan verificación aparecen en

verde si son “*passed*”, en amarillo si son “*blocked*”, en rojo si son “*failed*” y en negro si son “*not executed*”.

- Resultado obtenido: OK.

Caso 2

- Descripción: ejecutar *Test Runs* desde una la tabla de Asignación de Ejecución de *Test*.
- Requisitos previos: el usuario tiene registros asignados en esta tabla.
- Pasos a seguir:
 - Ir a *Test Management 2.0>Test Assigned To Me*.
 - Seleccionar *Tests*.
 - *Click Run*.
 - Rellenar los campos.
 - *Click Run*.
 - Completar los resultados de los Pasos que necesitan verificación en todos los *Test*.
 - *Click Done*.
- Pruebas realizadas:
 - Prueba 1: ejecutar 4 *Test* y dejar cada uno en un estado.
 - Resultado esperado: aparece un mensaje indicando la pause. Se crea un registro *Test Run* con el campo *total Test* en 4, y los campos *passed*, *failed*, *blocked* y *not finished* a 1. Se crea un registro Resultado de *Test* por cada *Test* con campo *Result* con el resultado correspondiente. Los pasos que necesitan verificación aparecen en verde si son “*passed*”, en amarillo si son “*blocked*”, en rojo si son “*failed*” y en negro si son “*not executed*”.
 - Resultado obtenido: OK.

7.3.17 Pausar la Ejecución de un *Test*

Caso 1

- Descripción: pausar la ejecución de *Test Runs* desde una Versión de *Test*.
- Requisitos previos: la Versión de *Test* está en estado *Ready*.
- Pasos a seguir:
 - Ir a *Test Management 2.0>Test*.
 - Acceder a un *Test*.
 - Ir a la *related list Test Version*.
 - Acceder a la Versión de *Test*.
 - *Click Run*.
 - Rellenar los campos.
 - *Click Pause*.
- Pruebas realizadas:
 - Prueba 1: ejecutar un *Test* y pulsar el botón *Pause*.
 - Resultado esperado: aparece un mensaje indicando la pause. Se crea un registro *Test Run* con el campo *total Test* en 1, el campo *state* como *In Progress*. Se crea un registro Resultado de *Test* con campo *Result* vacío. Los pasos que necesitan verificación aparecen en verde si son “*passed*”, en amarillo si son “*blocked*”, en rojo si son “*failed*” y en negro si son “*not executed*”.
 - Resultado obtenido: OK.

Caso 2

- Descripción: pausar la ejecución de *Test Runs* desde una la tabla de Asignación de Ejecución de *Test*.
- Requisitos previos: el usuario tiene registros asignados en esta tabla.
- Pasos a seguir:
 - Ir a *Test Management 2.0>Test Assigned To Me*.
 - Seleccionar *Tests*.
 - *Click Run*.
 - Rellenar los campos.
 - *Click Pause*.
- Pruebas realizadas:
 - Prueba 1: ejecutar 3 *Test* y pulsar el botón *Pause*.
 - Resultado esperado: aparece un mensaje indicando la *pause*. Se crea un registro *Test Run* con el campo *total Test* en 3, el campo *state* como *In Progress*. Se crea un registro Resultado de *Test* por cada *Test* con campo *Result* vacío. Los pasos que necesitan verificación aparecen en verde si son “*passed*”, en amarillo si son “*blocked*”, en rojo si son “*failed*” y en negro si son “*not executed*”.
 - Resultado obtenido: OK.

7.3.18 Reiniciar la Ejecución de un *Test* pausado

Caso 1

- Descripción: continuar la ejecución de *Test Runs* pausado.
- Requisitos previos: el estado del *Test Run* es *In Progress*.
- Pasos a seguir:
 - Ir a *Test Management 2.0>Test Runs*.
 - Acceder a un *Test Run*.
 - *Click en Continue Run*.
- Pruebas realizadas:
 - Prueba 1: continuar ejecutando una prueba pausada.
 - Resultado esperado: se vuelve a abrir la ventana de ejecución del *Test Run* en el estado en el que estaba y se termina de manera normal.
 - Resultado obtenido: OK.

7.4 Mejoras para *Test Management*

Durante el proceso de plan de pruebas se han observado algunas mejoras que se podrían realizar en el proceso para sacar un mayor partido a los desarrollos realizados:

- Hacer *Test* desde los Conjuntos de *Test*: al contener los Conjuntos de *Test Test* que tienen una relación entre sí, sería interesante implementar un botón *Run* desde el propio conjunto de *Test*. Ahora mismo se podría realizar añadiendo el Conjunto de *Test* a un Software de Ejecución de *Test*, pero el método propuesto es más rápido e intuitivo.
- Controlar los *Test* que se añaden a un Software de Ejecución de *Test* y los que se ejecutan en conjuntos: ahora mismo se permite seleccionar cualquier tipo de *Test* en estas opciones, pero

para que el proceso sea eficiente solo deberían poder seleccionarse Test del mismo cliente (campo *company*).

- Entornos de Test seleccionables en la ejecución: igualmente, cuando se ejecuta un Test se puede seleccionar un entorno de cualquier cliente. Sería mejor controlar este campo para que solo mostrara los Entornos de Test cuya *company* es la misma que el Test ejecutado, ya que esas pruebas solo serán válidas en instancias de ese cliente.
- Campo *runnable versión* en Test: cuando en una Versión de Test en *Ready* se pulsa el botón *Edit Version*, el campo *runnable versión* del Test asociado no se borra, cuando en ese momento no se puede ejecutar.

8 Conclusiones

El trabajo de fin de grado me ha servido para trabajar varios conceptos que creo que me pueden ser muy útiles tanto en mi vida laboral como en la vida diaria.

El uso de la herramienta *ServiceNow* me ha servido para ampliar mis conocimientos sobre los requisitos y las capacidades con las que cuenta esta aplicación, lo que creo que me será muy útil actualmente ya que es la herramienta con la que trabajo diariamente. Tener un mejor conocimiento sobre esta me permitirá ser más eficiente en mis tareas diarias y conocer módulos de esta que habría tardado más en dominar y en conocer.

En referencia a la parte más técnica de programación, he ampliado mis conocimientos en *Javascript* y HTML5, además de Angular. Todos ellos son lenguajes que ahora mismo se usan en muchas empresas y entornos, por lo que un buen conocimiento sobre ellos me permitirá tener más opciones una vez finalizado este Trabajo de Fin de Grado.

El trabajo con metodologías ágiles y *Scrum* es muy útil ya que es un sistema que ahora mismo se usa en muchas empresas y entidades, y se el desarrollador de este Trabajo de Fin de Grado me ha permitido mejorar a nivel de organización de trabajo, de las relaciones con un cliente, y también a la hora de dar *feedback* a un cliente y proponer mejoras sobre los desarrollos que se están realizando, que es algo que no es mi papel diariamente, y me vendrá bien cuando en el futuro tenga que asumir más responsabilidades.

Finalmente, creo que este proyecto ha motivado mi responsabilidad personal a la hora de hacer tareas cuando no hay nadie detrás exigiendo resultados tan de continuo, ya que es algo de lo que soy el único responsable (a nivel de estudios).

9 Bibliografía

- [1] SilverStorm. “¿QUIÉNES SOMOS?”. SilverStorm. <https://silver-storm.com/conocenos/> (acceso: 9 de mayo de 2021)
- [2] ServiceNow. “ServiceNow”. ServiceNow. <https://www.ServiceNow.es/now-platform.html> (acceso: 9 de mayo de 2021)
- [3] ServiceNow. *ServiceNow Fundamentals Participant Guide*. 1st Edition. Santa Clara, CA: ServiceNow, 2019
- [4] K. Schwaber y J. Sutherland. *La guía de scrum*. 2020
- [5] “Qué es SCRUM”. Proyectosagiles.org. <https://proyectosagiles.org/que-es-scrum/> (acceso: 10 de abril de 2021).
- [6] ServiceNow. “Field Types”. ServiceNow Product Documentation. https://docs.ServiceNow.com/bundle/quebec-platform-administration/page/administer/reference-pages/reference/r_FieldTypes.html (acceso: 20 de enero de 2021).
- [7] ServiceNow. “Form Layout”. *ServiceNow Product Documentation* <https://docs.ServiceNow.com/bundle/quebec-platform-administration/page/administer/form-administration/concept/configure-form-layout.html> (acceso: 10 de febrero de 2021)
- [8] ServiceNow. “Related lists”. *ServiceNow Product Documentation*. https://docs.ServiceNow.com/bundle/quebec-platform-user-interface/page/use/using-forms/concept/c_RelatedLists.html (acceso: 20 de marzo de 2021).
- [9] ServiceNow. “ UI actions”. *ServiceNow Product Documentation*. https://docs.servicenow.com/bundle/quebec-platform-administration/page/administer/list-administration/concept/c_UIActions.html (acceso: 17 de febrero de 2021).
- [10] “Introducción a Bootstrap”. *Bootstrap*. <https://getbootstrap.com/docs/5.0/getting-started/introduction/> (acceso: 12 de abril de 2021).
- [11] ServiceNow . “Extensions to Jelly syntax”. *ServiceNow Product Documentation*. https://docs.ServiceNow.com/bundle/paris-application-development/page/script/general-scripting/concept/c_ExtensionsToJellySyntax.html (acceso: 17 de mayo de 2021).
- [12] ServiceNow. “ Create a widget that displays a ServiceNow UI page”. *ServiceNow Product Documentation*. https://docs.ServiceNow.com/bundle/quebec-now-intelligence/page/use/dashboards/task/create_widget_displays_webpage.html (acceso: 2 de febrero de 2021).
- [13] ServiceNow. “ Interactive Filters” . *ServiceNow Product Documentation*. https://docs.servicenow.com/bundle/quebec-now-intelligence/page/use/dashboards/concept/c_HomepagePublishers.html (acceso: 15 de abril de 2021).
- [14] ServiceNow. “ Create a reference field interactive filter” . *ServiceNow Product Documentation* https://docs.ServiceNow.com/bundle/quebec-now-intelligence/page/use/dashboards/task/t_CreateAReferenceFieldPublisher.html (acceso: 15 de abril de 2021).
- [15] ServiceNow. “Transform Maps”. *ServiceNow Product Documentation* https://docs.ServiceNow.com/bundle/quebec-platform-administration/page/script/server-scripting/concept/c_CreatingNewTransformMaps.html (acceso: 15 de enero de 2021).

[16] ServiceNow. *Scripting in ServiceNow Fundamentals Participant Guide*. 1st Edition. Santa Clara, CA: ServiceNow, 2019.

[17] ServiceNow. "API Reference – Client". *ServiceNow Developer*
<https://developer.ServiceNow.com/dev.do#!/reference/api/quebec/client/> (acceso: 10 de mayo de 2021).

[18] ServiceNow. "API Reference – Server Global". *ServiceNow Developer*
https://developer.ServiceNow.com/dev.do#!/reference/api/quebec/server_legacy/ (acceso: 12 de mayo de 2021).

[15] ServiceNow. "Business Rule". *ServiceNow Product Documentation*.
https://docs.ServiceNow.com/bundle/quebec-application-development/page/script/business-rules/concept/c_BusinessRules.html (acceso: 10 de abril de 2021).

[16] ServiceNow. "Client Scripts". *ServiceNow Product Documentation*.
<https://docs.ServiceNow.com/bundle/paris-application-development/page/script/client-scripts/concept/client-scripts.html> (acceso: 13 de abril de 2021).

[16] ServiceNow. "Scripts Include". *ServiceNow Product Documentation*.
https://docs.ServiceNow.com/bundle/paris-application-development/page/script/server-scripting/concept/c_ScriptIncludes.html (acceso: 17 de abril de 2021).

[22] "Pruebas de software: 10 pasos para elaborar el plan de pruebas". PMOinformatica.com.
<http://www.pmoinformatica.com/2016/01/elaborar-plan-pruebas-software.html> (acceso: 30 de abril de 2021).

[23] J.M Sanchez Peño. Pruebas de Software. Fundamentos y Técnicas. 1ªed. Madrid: Universidad Politécnica de Madrid, 2015.

[24] "Guía temática sobre citas bibliográficas UC3M: IEEE v01.29.2021". Universidad Carlos III de Madrid. https://uc3m.libguides.com/guias_tematicas/citas_bibliograficas/IEEE#s-lg-box-wrapper-13407990 (acceso: 18 de junio de 2021).

[25] IEEE Publishing Operations. IEEE REFERENCE GUIDE. 1ªed. Piscataway, NJ: Institute of Electrical and Electronics Engineers.

10 Anexos

10.1 Anexos Gestión de proyectos

10.1.1 Lógica de Proyecto

Campos: La tabla de proyectos es la base para la gestión de lo desarrollado. En ella se utilizarán los siguientes campos para realizar la gestión del proyecto:

Nombre del campo	Type	Mandatory	Read Only
Number	String	No	Si
State	Choice	No	No
Active	True/False	No	No
Project Name	String	Si	No
Planned Effort	Duration	No	Si
Actual Effort	Duration	No	Si
Projected Effort	Duration	No	Si
Actual Cost	Currency	No	Si
Projected Cost	Currency	No	Si
Planned Start Date	Datetime	No	No
Planned End Date	Datetime	No	No
Planned Cost (cost Price)	Currency	No	Si
Actual Cost (cost Price)	Currency	No	Si
Projected Cost (cost Price)	Currency	No	Si

Tabla 10.1.1 Lógica de campos de proyecto

Estos campos se muestran en el formulario principal del siguiente modo (se incluyen campos que no han sido mencionados en el proyecto, ya que no se han desarrollado para el mismo):

Number	Special follow up
Deployment ServiceNow	State
Offer	Active
Po Number	Status
Project Net Amount	Project Manager
Company	Commercial Assigned
Language	Business Process Consultant
Type	Quality Assurance Consultant
Schedule	Senior Deployment Consultant
	Senior Deployment Consultant
Project Name	
Description	

Tabla 10.1.2 Formulario de proyecto

A parte del formulario principal, encontramos dos formularios secundarios que se muestran como pestañas:

La pestaña Project Information:

<i>Planned Effort</i>	<i>Calculation</i>
<i>Actual Effort</i>	<i>Plannd Start Date</i>
<i>Projected Effort</i>	<i>Planned End Date</i>
<i>Actual Cost</i>	<i>Invoiced</i>
<i>Projected Cost</i>	<i>Pending</i>

Tabla 10.1.3 Pestaña Project Information de Proyecto

La pestaña Project Task incluye la lista relacionada de tareas de proyecto que son hijas del proyecto en el que se encuentra.

La pestaña Cost Price:

<i>Planned Cost (cost Price)</i>	<i>Actual Cost (cost Price)</i>
	<i>Projected Cost (cost Price)</i>

Tabla 10.1.4 Pestaña Cost Price de Proyecto

En la lista general de la tabla, se muestran los siguientes campos en cada registro:

<i>Number</i>	<i>Company</i>	<i>Project Manager</i>	<i>Project Name</i>	<i>State</i>	<i>Plannd Start Date</i>	<i>Planned End Date</i>	<i>Percent Complete</i>
---------------	----------------	------------------------	---------------------	--------------	--------------------------	-------------------------	-------------------------

<i>Planned Effort</i>	<i>Actual Effort</i>	<i>Planned Cost</i>	<i>Actual Cost</i>	<i>Service Aware Candidate</i>	<i>Net Amount</i>	<i>Project Net Amount</i>	<i>Invoiced</i>	<i>Pending</i>
-----------------------	----------------------	---------------------	--------------------	--------------------------------	-------------------	---------------------------	-----------------	----------------

Tabla 10.1.5 Vista de tabla de Proyecto

10.1.2 Lógica de Tareas de Proyecto

Campos: los campos relacionados con este proyecto de la entidad de tareas de proyecto son:

Nombre del campo	Type	Mandatory	Read Only
<i>Number</i>	<i>String</i>	No	Si
<i>State</i>	<i>Integer</i>	No	No
<i>Short Description</i>	<i>String</i>	No	No
<i>Planned Effort DC</i>	<i>Duration</i>	Solo 1 de los 4 campos	No
<i>Planned Effort SDC</i>	<i>Duration</i>		No
<i>Planned Effort PDM</i>	<i>Duration</i>		No
<i>Planned Effort BPC</i>	<i>Duration</i>		No
<i>Planned Effort</i>	<i>Duration</i>	No	Si
<i>Planned Cost DC</i>	<i>Currency</i>	No	Si
<i>Planned Cost SDC</i>	<i>Currency</i>	No	Si
<i>Planned Cost PDM</i>	<i>Currency</i>	No	Si
<i>Planned Cost BPC</i>	<i>Currency</i>	No	Si
<i>Planned Cost</i>	<i>Currency</i>	No	Si
<i>Actual Effort DC</i>	<i>Duration</i>	No	Si
<i>Actual Effort SDC</i>	<i>Duration</i>	No	Si
<i>Actual Effort PDM</i>	<i>Duration</i>	No	Si
<i>Actual Effort BPC</i>	<i>Duration</i>	No	Si
<i>Actual Effort</i>	<i>Duration</i>	No	Si

<i>Actual Cost DC</i>	<i>Currency</i>	No	Si
<i>Actual Cost SDC</i>	<i>Currency</i>	No	Si
<i>Actual Cost PDM</i>	<i>Currency</i>	No	Si
<i>Actual Cost BPC</i>	<i>Currency</i>	No	Si
<i>Actual Cost</i>	<i>Currency</i>	No	Si
<i>Projected Effort DC</i>	<i>Duration</i>	No	Si
<i>Projected Effort SDC</i>	<i>Duration</i>	No	Si
<i>Projected Effort PDM</i>	<i>Duration</i>	No	Si
<i>Projected Effort BPC</i>	<i>Duration</i>	No	Si
<i>Projected Effort</i>	<i>Duration</i>	No	Si
<i>Projected Cost DC</i>	<i>Currency</i>	No	Si
<i>Projected Cost SDC</i>	<i>Currency</i>	No	Si
<i>Projected Cost PDM</i>	<i>Currency</i>	No	Si
<i>Projected Cost BPC</i>	<i>Currency</i>	No	Si
<i>Projected Cost</i>	<i>Currency</i>	No	Si
<i>Planned Start Date</i>	<i>Datetime</i>	No	No
<i>Planned End Date</i>	<i>Datetime</i>	No	No

Tabla 10.1.6 Lógica de campos de tarea de proyecto

Estos campos se muestran en el formulario principal del siguiente modo (se incluyen campos que no han sido mencionados en las tareas de proyecto, ya que no se han desarrollado para el mismo):

<i>Number</i>	<i>State</i>
<i>Time constraint</i>	<i>Planned Effort</i>
<i>Planned Start Date</i>	<i>Actual Effort</i>
<i>Planned End Date</i>	<i>Remaining effort</i>
<i>Planned Duration</i>	<i>Projected Effort</i>
<i>Planned Cost</i>	
<i>Actual Cost</i>	
<i>Projected Cost</i>	
<i>Short Description</i>	
<i>Description</i>	

Tabla 10.1.7 Formulario de Tarea de proyecto

A parte del formulario principal, encontramos dos formularios secundarios que se muestran como pestañas:

La pestaña Role Cost:

<i>Planned cost BPC</i>	<i>Actual Cost BPC</i>
	<i>Projected Cost BPC</i>
<i>Planned cost PDM</i>	<i>Actual Cost PDM</i>
	<i>Projected Cost PDM</i>
<i>Planned cost SDC</i>	<i>Actual Cost SDC</i>
	<i>Projected Cost SDC</i>
<i>Planned cost DC</i>	<i>Actual Cost DC</i>
	<i>Projected Cost DC</i>

Tabla 10.1.8 Pestaña Role Cost de Tarea de Proyecto

La pestaña Role Effort:

<i>Planned cost BPC</i>	<i>Actual Cost BPC</i>
	<i>Projected Cost BPC</i>
<i>Planned cost PDM</i>	<i>Actual Cost PDM</i>
	<i>Projected Cost PDM</i>
<i>Planned cost SDC</i>	<i>Actual Cost SDC</i>
	<i>Projected Cost SDC</i>
<i>Planned cost DC</i>	<i>Actual Cost DC</i>
	<i>Projected Cost DC</i>

Tabla 10.1.9 Pestaña Role Effort de Tara de Proyecto

En la lista general de la tabla, se muestran los siguientes campos en cada registro:

<i>Created</i>	<i>Task</i>	<i>Project</i>	<i>Assigned To</i>	<i>Dependency</i>	<i>Planned Start Date</i>	<i>Planned End Date</i>	<i>Planned Duration</i>	<i>Percent Complete</i>	<i>State</i>
----------------	-------------	----------------	--------------------	-------------------	---------------------------	-------------------------	-------------------------	-------------------------	--------------

Tabla 10.1.10 Vista de tabla de Tarea de Proyecto

Además, la tabla presenta una vista distinta en la *Related list* que se muestra desde la tabla de Proyecto:

<i>Number</i>	<i>Short Description</i>	<i>State</i>	<i>Planned Start Date</i>	<i>Planned End Date</i>	<i>Percent Complete</i>	<i>Planned Effort</i>	<i>Actual Effort</i>	<i>Remaining Effort</i>
---------------	--------------------------	--------------	---------------------------	-------------------------	-------------------------	-----------------------	----------------------	-------------------------

Tabla 10.1.11 Vista de tabla de Tarea de Proyecto en la RL de Proyecto

10.1.3 Lógica de *Time Cards*

Campos: los campos relacionados con este proyecto de la entidad de *Time Card* son:

Nombre del campo	Type	Mandatory	Read Only	Reference
<i>Week Stars On</i>	<i>Date</i>	No	No	
<i>User</i>	<i>Reference</i>	Si	Depende de permisos	<i>sys_user</i>
<i>State</i>	<i>String</i>	No	Si	
<i>Task</i>	<i>Reference</i>	Si	No	<i>Task</i>
<i>Resource Plan</i>	<i>Reference</i>	No	Si	<i>resource_plan</i>
<i>Sunday</i>	<i>Decimal</i>	Si	No	
<i>Monday</i>	<i>Decimal</i>	Si	No	
<i>Tuesday</i>	<i>Decimal</i>	Si	No	
<i>Wednesday</i>	<i>Decimal</i>	Si	No	
<i>Thursday</i>	<i>Decimal</i>	Si	No	
<i>Friday</i>	<i>Decimal</i>	Si	No	
<i>Saturday</i>	<i>Decimal</i>	Si	No	
<i>Total</i>	<i>Decimal</i>	No	Si	

Tabla 10.1.12 Lógica de campos de *Time Cards*

Estos campos se muestran en el formulario principal del siguiente modo (se incluyen campos que no han sido mencionados en las *Time Cards* ya que no se han desarrollado para el mismo):

<i>Week Stars On</i>	<i>Sunday</i>
<i>User</i>	<i>Monday</i>
<i>State</i>	<i>Tuesday</i>
<i>Type</i>	<i>Wednesday</i>
<i>Task</i>	<i>Thursday</i>
<i>Case</i>	<i>Friday</i>
<i>Story</i>	<i>Saturday</i>
<i>Resource Plan</i>	<i>Total</i>
<i>Notes</i>	

Tabla 10.1.13 Formulario general Time Card

En la lista general de la tabla, se muestran los siguientes campos en cada registro:

<i>Week Stars On</i>	<i>User</i>	<i>User.Role</i>	<i>Task.Parent.Company</i>	<i>Task.Parent.Short Description</i>	<i>Task</i>	<i>Task.Short Description</i>	<i>State</i>
----------------------	-------------	------------------	----------------------------	--------------------------------------	-------------	-------------------------------	--------------

<i>Sunday</i>	<i>Monday</i>	<i>Tuesday</i>	<i>Wednesday</i>	<i>Thursday</i>	<i>Friday</i>	<i>Saturday</i>	<i>Total</i>	<i>Notes</i>
---------------	---------------	----------------	------------------	-----------------	---------------	-----------------	--------------	--------------

Tabla 10.1.14 Vista de tabla de Time Card

Además, la tabla presenta la misma vista en la *Related list* que se muestra desde la tabla de Proyectos y Tareas de Proyecto.

10.1.4 Lógica de Planes de Recursos

Campos: los campos relacionados con este proyecto de la entidad de Plan de Recurso son:

Nombre del campo	Type	Mandatory	Read Only	Reference
<i>Number</i>	<i>String</i>	No	Si	
<i>State</i>	<i>Integer</i>	No	Si	
<i>Project</i>	<i>Reference</i>	Si	Si	<i>Task</i>
<i>Project Task</i>	<i>Choice</i>	Cuando el Project es una demanda	No	
<i>Resource Type</i>	<i>Choice</i>	Si	No	
<i>User Resource</i>	<i>Reference</i>	Si Resource Type = User Resource	No	<i>Sys_user</i>
<i>Group Resource</i>	<i>Reference</i>	Si Resource Type = Group Resource	No	<i>sys_user_group</i>
<i>Members Preference</i>	<i>Choice</i>	Si Resource Type = Group Resource	No	
<i>Members List</i>	<i>List</i>	Si Members Preference = Specific Members	No	
<i>Role Resource</i>	<i>String</i>	Si Resource Type = Role Resource	No	
<i>Resource Rate</i>	<i>Currency</i>	Si	No	
<i>Planned Hours</i>	<i>Decimal</i>	Si	No	
<i>Planned Cost</i>	<i>Currency</i>	No	Si	
<i>Confirmed/Allocated hours</i>	<i>Decimal</i>	No	No	
<i>Confirmed/Allocated cost</i>	<i>Currency</i>	No	Si	

<i>%Plan</i>	<i>Percent complete</i>	No	Si	
<i>Start Date</i>	<i>Date</i>	Si	No	
<i>End Date</i>	<i>Date</i>	Si	No	

Tabla 10.1.15 Lógica de campos de Planes de Recursos

Estos campos se muestran en el formulario principal del siguiente modo (se incluyen campos que no han sido mencionados en los planes de recursos, ya que no se han desarrollado para el mismo):

<i>Number</i>	<i>State</i>
<i>Resource Type</i>	<i>Task Type</i>
<i>Group/User/Role Resource</i>	<i>Project</i>
<i>Members Preference</i>	<i>Start Date</i>
<i>Members List</i>	<i>End Date</i>
<i>Request type</i>	
<i>Allocation Spread</i>	
<i>Allocation Type</i>	
<i>Planned Hours</i>	

Tabla 10.1.16 Formulario de Plan de Recursos

A parte del formulario principal, encontramos un formulario secundario que se muestra como pestaña:

La pestaña Cost:

<i>Confirmed/Allocated hours</i>	<i>Planned Cost</i>
<i>Actual Hours</i>	<i>Confirmed/Allocated Cost</i>
<i>% Plan</i>	<i>Actual Cost</i>
<i>Notes</i>	

Tabla 10.1.17 Pestaña Cost en Plan de Recursos

En la lista general de la tabla, se muestran los siguientes campos en cada registro:

<i>Number</i>	<i>Name</i>	<i>Project</i>	<i>Offer</i>	<i>Resource Type</i>	<i>Group</i>	<i>User</i>	<i>Start Date</i>
---------------	-------------	----------------	--------------	----------------------	--------------	-------------	-------------------

<i>End Date</i>	<i>Requested Type</i>	<i>FTE</i>	<i>Planned Hours</i>	<i>Planned Cost</i>	<i>State</i>
-----------------	-----------------------	------------	----------------------	---------------------	--------------

Tabla 10.1.18 Vista de tabla de Resource Plan

Además, la tabla presenta una vista distinta en las distintas *Related list*:

Related list de Proyecto:

<i>Number</i>	<i>Name</i>	<i>Role Resource</i>	<i>State</i>	<i>Planned Hours</i>	<i>Planned Cost</i>	<i>Confirmed/Allocated hours</i>	<i>Confirmed/Allocated Cost</i>	<i>Start Date</i>	<i>End Date</i>
---------------	-------------	----------------------	--------------	----------------------	---------------------	----------------------------------	---------------------------------	-------------------	-----------------

Tabla 10.1.19 Vista de tabla de Plan de Recursos en RL de Proyecto

Related list de Tarea de Proyecto:

<i>Number</i>	<i>Name</i>	<i>Group</i>	<i>User</i>	<i>Start Date</i>	<i>End Date</i>	<i>FTE</i>	<i>Planned Hours</i>	<i>Planned Cost</i>	<i>State</i>
---------------	-------------	--------------	-------------	-------------------	-----------------	------------	----------------------	---------------------	--------------

Tabla 10.1.20 Vista de tabla de Plan de Recursos en RL de Tarea de Proyecto

10.1.5 Lógica de Asignaciones de Recursos

Campos: los campos relacionados con este proyecto de la entidad de Asignación de Recursos son:

Nombre del campo	Type	Mandatory	Read Only	Reference
Resource plan	Reference	No	Si	resource_plan
Resource	Reference	Si	No	sys_user
Start date	Date	No	Si	
End date	Date	No	Si	
Requested hours	Decimal	Si	No	
Confirmed/Allocated Hours	Decimal	No	No	
Requested Cost	Currency	No	Si	
Confirmed/Allocated Cost	Currency	No	Si	

Tabla 10.1.21 Lógica de campos de Asignación de Recursos

Estos campos se muestran en el formulario principal del siguiente modo (se incluyen campos que no han sido mencionados en las asignaciones de recursos, ya que no se han desarrollado para el mismo):

Resource Plan	Requested Hours
Resource	Confirmed/Allocated Hours
Start Date	
End Date	

Tabla 10.1.22 Formulario de Asignación de Recursos

En la lista general de la tabla, se muestran los siguientes campos en cada registro:

Resource Plan	Resource Plan.Name	Start Date	End Date	Requested Hours	Confirmed /Allocated Hours	Task	Task. Company	Task.Short Description	Resource
---------------	--------------------	------------	----------	-----------------	----------------------------	------	---------------	------------------------	----------

Tabla 10.1.23 Vista de tabla de Asignación de Recursos

Además, la tabla presenta una vista distinta en la *Related list* de los Planes de Recursos:

Number	Resource Plan.Name	Resource	Start Date	End Date	Requested Hours	Confirmed/Allocated Hours	Bookin Type
--------	--------------------	----------	------------	----------	-----------------	---------------------------	-------------

Tabla 10.1.24 Vista de tabla de Asignación de Recursos en la RL Plan de recursos

10.1.6 Lógica de Hojas de Tarifas de Gasto

Campos: los campos relacionados con este proyecto de la entidad de Hojas de Tarifa de Gasto son:

Nombre del campo	Type	Mandatory	Read Only
Number	String	No	Si
Active	True/False	No	No
Condition	Conditions	Si	No
Order	Integer	No	No
Task Rate	Currency	Si	No

Tabla 10.1.25 Lógica de campos de Hojas de Tarifas de Gasto

Estos campos se muestran en el formulario principal del siguiente modo (se incluyen campos que no han sido mencionados en las hojas de tarifas de gastos, ya que no se han desarrollado para el mismo):

<i>Number</i>	<i>Name</i>
<i>Table</i>	<i>Order</i>
<i>Summary Type</i>	<i>Active</i>
<i>Short Description</i>	

Tabla 10.1.26 Formulario Hoja de Tarifa de Gasto

A parte del formulario principal, encontramos un formulario secundario que se muestra como pestaña:

La pestaña General:

<i>Use Time Worked</i>	<i>Task Rate</i>
<i>Condition</i>	
<i>Description</i>	

Tabla 10.1.27 Pestaña de General en Hoja de Tarifa de Gastos

En la lista general de la tabla, se muestran los siguientes campos en cada registro:

<i>Name</i>	<i>Table</i>	<i>Task Rate</i>	<i>Use Time Worked</i>	<i>Default Labor Rate</i>	<i>Summary Type</i>	<i>Active</i>	<i>Order</i>
-------------	--------------	------------------	------------------------	---------------------------	---------------------	---------------	--------------

Tabla 10.1.28 Vista de tabla de Hoja de Tarifa de Proyectos

10.1.7 Imágenes de la aplicación Gestión de proyectos

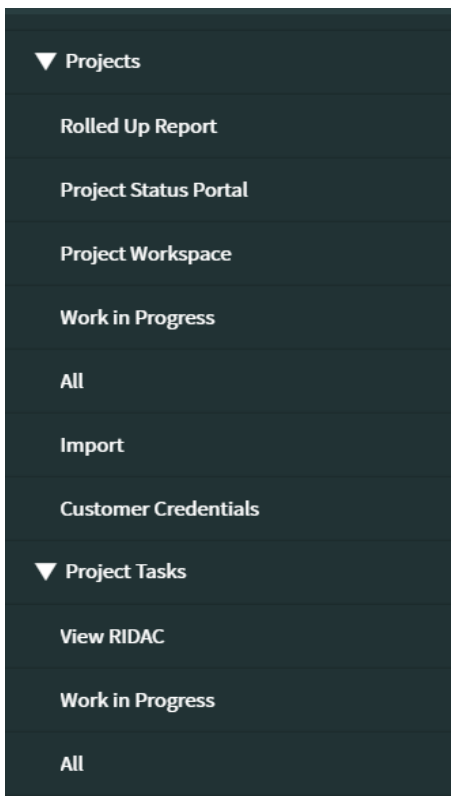


Figura 10.1.2 Menu Project y Project Task

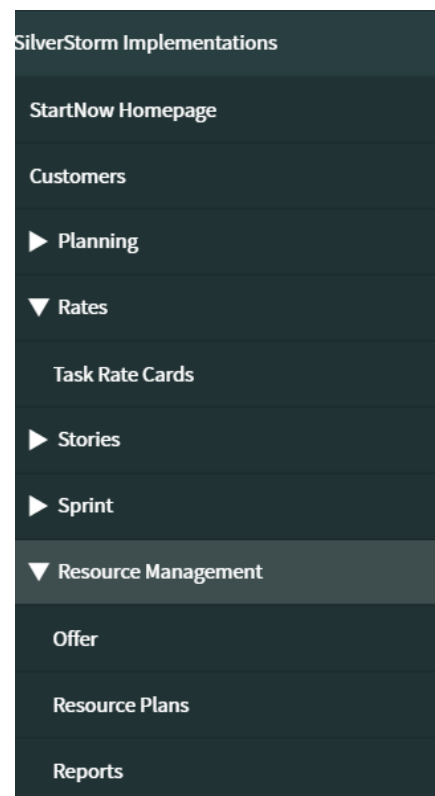


Figura 10.1.1 Menu SilverStorm Implementation

Number	Company	Project Name	State	Planned effort	Actual effort	Planned cost	Actual cost	Project Net Amount	Projected C
PRJ0010675	SILVERSTORM	NTT - Nuevo proyecto para TFG	Work in Progress			0,00 €	0,00 €	17.500,00 €	
PRJ0010666	SILVERSTORM	TFG project	Work in Progress			0,00 €	0,00 €	0,00 €	
PRJ0010678	(empty)	test project TFG	Pending			0,00 €	0,00 €	0,00 €	
PRJ0010683	SILVERSTORM	Project demo TFG	Pending	60 Hours	10 Hours	1.400,00 €	500,00 €	0,00 €	1.40
PRJ0010679	SILVERSTORM	test project TFG II	Work in Progress	109 Hours	25 Hours	430,00 €	1.050,00 €	5.800,00 €	97
PRJ0010648	SILVERSTORM	TFG - Mejoras ITBM OneStorm	Closed Incomplete	2 Hours	0 Seconds	0,00 €	0,00 €	0,00 €	

Figura 10.1.3 Tabla de proyectos

Project Informations	Project Task	Applications/Services	Cost Price
Planned effort	<input type="text"/>	Calculation	Automatic
Actual effort	<input type="text"/>	Planned start date	04-01-2021 08:00:00
Projected Effort	<input type="text"/>	Planned end date	30-04-2021 17:00:00
Actual cost	0,00 €	Invoiced	0,00 €
Projected Cost	0,00 €	Pending	17.500,00 €

Figura 10.1.4 Campos Coste y Esfuerzo de Proyecto

Project Informations	Project Task	Applications/Services	Cost Price
Planned Cost (cost price)	0,00 €	Actual Cost (cost price)	0,00 €
		Projected Cost (cost price)	0,00 €

Figura 10.1.5 Pestaña Cost Price en Proyecto

Project Task	PRJTASK0015631
Parents	test project TFG II > PRJTASK0015631
Number	PRJTASK0015631
Time constraint	Start ASAP
Planned start date	10-05-2021 08:00:00
Planned end date	02-07-2021 17:00:00
Planned duration	39 Days
Planned cost	240,00 €
Actual cost	120,00 €
Projected Cost	120,00 €
* Short Description	06 - MANAGEMENT
Description	This covers the following activities: - Service Management task (SLAs, KPIs follow-up, Task follow-up, review task, etc...) - Service Review
State	Work in Progress
Planned effort	5 Hours
Actual effort	3 Hours
Remaining effort	2 Hours
Projected Effort	5 Hours

Figura 10.1.6 Formulario Tarea de proyecto

Role Cost		Role Effort	
Planned cost BPC	0,00 €	Actual cost BPC	0,00 €
Planned cost PDM	50,00 €	Projected Cost BPC	0,00 €
Planned cost SDC	150,00 €	Actual cost PDM	0,00 €
Planned cost DC	40,00 €	Projected Cost PDM	0,00 €
		Actual cost SDC	0,00 €
		Projected Cost SDC	0,00 €
		Actual cost DC	120,00 €
		Projected Cost DC	120,00 €

Figura 10.1.7 Costes por Rol de Tarea de Proyecto

Role Cost		Role Effort	
Planned effort BPC	Hours 0 00 00	Actual effort BPC	
Planned effort PDM	Hours 1 00 00	Projected Effort BPC	2 Hours
Planned effort SDC	Hours 3 00 00	Actual effort PDM	
Planned effort DC	Hours 1 00 00	Projected Effort PDM	0 Seconds
		Actual effort SDC	
		Projected Effort SDC	0 Seconds
		Actual effort DC	3 Hours
		Projected Effort DC	3 Hours

Figura 10.1.8 Esfuerzo por Rol de Tareas de Proyecto

Project Tasks		Time Cards (2)		Expense Lines (2)		Stories		Cases		Resource Plans (2)		
Task = PRJTASK0015631												
Week starts on	User	Role	Company	Short Description	Short Description	State	Total	Sunday	Monday	Tuesday	Wednesday	Thursday
16-05-2021	Alejandro Muñoz García	Deployment Consultant	SILVERSTORM	test project TFG II	06 - MANAGEMENT	Processed	1	0	0	0	1	0
16-05-2021	Alejandro Muñoz García	Deployment Consultant	SILVERSTORM	test project TFG II	06 - MANAGEMENT	Processed	2	0	0	2	0	0
Sum							3	0	0	2	1	0

Figura 10.1.9 Related list de Time Card en Tareas de Proyecto

Project Tasks		Time Cards (3)		Expense Lines (1)		Stories		Cases		Resource Plans (6)	
Project = PRJTASK0015658											
Number	Name	Group	User	Start date	End date	FTE	Planned hours	Planned cost	State		
RPLN0003094	RPLN0003094 - User DC	(empty)	User DC	27-05-2021	18-06-2021	0	0	0,00 €	Cancelled		
RPLN0003092	RPLN0003092 - Alberto Benito	(empty)	User SDC	25-05-2021	27-05-2021	0,63	15	300,00 €	Completed		
RPLN0003097	RPLN0003097 - User PDM	(empty)	User PDM	25-05-2021	27-05-2021	0,17	4	200,00 €	Completed		
RPLN0003095	RPLN0003095 - User BPC	(empty)	User BPC	27-05-2021	04-06-2021		10	500,00 €	Planning		
RPLN0003096	RPLN0003096 - User PDM	(empty)	User PDM	25-05-2021	25-05-2021		15	0,00 €	Completed		
RPLN0003093	RPLN0003093 - Alberto Benito	Titans	Alberto Benito	25-05-2021	04-06-2021	0	0	0,00 €	Cancelled		

Figura 10.1.10 Related list de Planes de Prueba en Tareas de Proyecto

Resource Plan
New record [New view]

Number:

* Resource type: -- None --

Request type: Hours

* Planned hours: 20

Name:

Task type: Project

Project: PRJTASK0015658

* Start date: 25-05-2021

* End date: 25-05-2021

Allocation type: Weekly

Allocation spread: Even

State: Planning

Rate override:

* Resource rate: € 0,00

Save Submit Create Soft Allocation

Figura 10.1.11 Formulario de creación de Plan de Recursos

Resource Plan
New record [New view]

Number:

* Resource type: Role Resource

* Role:

If group name is not known, it can be left blank

Request type: Hours

* Planned hours: 0

Name:

Task type: Demand

Project: DMND0001104

* Project Task: -- None --

* Start date:

* End date:

Allocation type: Weekly

Allocation spread: Even

State: Planning

Rate override:

* Resource rate: € 0,00

Save Submit Create Soft Allocation

Figura 10.1.12 Formulario de Plan de Recursos creado desde Demanda

Resource Plan
RPLN0003094

Number: RPLN0003094

Resource type: User Resource

User: User DC

Request type: Hours

Allocation spread: Even

Allocation type: Weekly

Planned hours: 0

State: Cancelled

Task type: Project

Project: PRJTASK0015658

Start date: 27-05-2021

End date: 18-06-2021

Cost

Confirmed/Allocated hours: 0

Actual hours: 0

% Plan: 100

Planned cost: 0,00 €

Confirmed/Allocated cost: 0,00 €

Actual cost: 0,00 €

Figura 10.1.13 Formulario de Plan de Recursos

Resource Allocations Search Number Search 1 to 1 of 1

Resource plan = RPLN0003097

	Number	Name	Resource	Start date	End date	Requested cost	Requested hours	Confirmed/Allocated hours	Booking type
	RALLO0025507	RPLN0003097 - User PDM	User PDM	25-05-2021	27-05-2021	250,00 €	5	4	Hard
	Sum						5		

Figura 10.1.14 Related list de Asignación de Recursos en Plan de Recursos

Resource Allocation
RALLOC0025507

Resource plan: RPLN0003097
Resource: User PDM
Start date: 25-05-2021
End date: 27-05-2021

Requested hours: 5
Confirmed/Allocated hours: 4

Related Links
Add to Update Set

Resource Allocations Daily

Name	Date	Hours	Notes
TFG Project Task 2	26-05-2021	1	
TFG Project Task 2	27-05-2021	1	
TFG Project Task 2	25-05-2021	2	

Figura 10.1.15 Formulario de Asignación de Recursos

Time Card
TIM0000111614

Week starts on: 23-05-2021

User: User PDM
State: Pending
Type: No Defect
Task: PRJTASK0015658
Resource plan: RPLN0003097

* Sunday	0
* Monday	0
* Tuesday	10
* Wednesday	0
* Thursday	0
* Friday	0
* Saturday	0
Total	10

Update Submit Time Card Approve Reject

Figura 10.1.16 Formulario de Time Card

Task Rate Cards

Name	Table	Task rate	Use time worked	Default labor rate	Summary type	Active	Order
*TFG	Search	Search	Search	Search	Search	Search	Search
Project demo TFG - DC	Time Card [time_card]	5,00 €	false	0,00 €		true	100
TFG Project Task 1 - DC	Time Card [time_card]	10,00 €	false	0,00 €		true	100
Project demo TFG - SDC	Time Card [time_card]	20,00 €	false	0,00 €		true	100
test project TFG.II - PDMRATE	Time Card [time_card]	50,00 €	false	0,00 €		true	100
test project TFG.II - SDC/BPC RATE	Time Card [time_card]	50,00 €	false	0,00 €		true	100
test project TFG.II - DC RATE	Time Card [time_card]	40,00 €	false	0,00 €		true	100
test project TFG.II - SDC/BPC RATE	Time Card [time_card]	50,00 €	false	0,00 €		true	100
test project TFG.II - PDMRATE	Time Card [time_card]	50,00 €	false	0,00 €		true	100

Figura 10.1.17 Lista de Hojas de Tarifa de Gasto

Task Rate Card
TKRC0010566

Number: TKRC0010566 Name: Project demo TFG - DC

Table: Time Card [time_card] Order: 100

Summary type: -- None -- Active:

Short description:

General

Use time worked: * Task rate: € 5,00

* Condition: 0 records match condition

Add Filter Condition Add "OR" Clause

All of these conditions must be met

Task(task).Parent is PRJ0010683 AND OR X

User.Role is Deployment Consultant AND OR X

Description:

Update

Figura 10.1.18 Formulario de Hoja de Tarifa de Gasto

Resource Roles New Search: Name [Search]

All > Name != QA

	Name	Description	Hourly rate
<input type="checkbox"/>	BPC	Business Process Consultant	55,00 €
<input type="checkbox"/>	PDM	Technical Account Manager	50,00 €
<input type="checkbox"/>	SDC	Senior Deployment Consultant	50,00 €
<input type="checkbox"/>	DC	Deployment Consultant	25,00 €

Figura 10.1.19 Lista de Recursos de Rol

10.2 Anexos *Test Management*

10.2.1 Lógica de Conjunto de *Test*

Campos: los campos relacionados con este proyecto de la entidad de Conjunto de *Test* son:

Nombre del campo	Type	Mandatory	Read Only	Reference
<i>Number</i>	<i>String</i>	No	Si	
<i>Company</i>	<i>Reference</i>	No	No	<i>core_company</i>
<i>Process</i>	<i>Choice</i>	No	No	
<i>Owner</i>	<i>Reference</i>	No	No	<i>sys_user</i>
<i>Category</i>	<i>String</i>	No	No	
<i>Name</i>	<i>String</i>	No	No	

Tabla 10.2.1 Lógica de campos de Conjunto de *Test*

Estos campos se muestran en el formulario principal del siguiente modo (se incluyen campos que no han sido mencionados en el Conjunto de *Test Test*, ya que no se han desarrollado para el mismo):

<i>Number</i>	<i>Owner</i>
<i>Company</i>	<i>Category</i>
<i>Process</i>	
<i>Name</i>	

Tabla 10.2.2 Formulario de Conjunto de *Test*

En la lista general de la tabla, se muestran los siguientes campos en cada registro:

<i>Number</i>	<i>Name</i>	<i>Process</i>	<i>Company</i>	<i>Owner</i>
---------------	-------------	----------------	----------------	--------------

Tabla 10.2.3 Vista de tabla de Conjunto de *Test*

10.2.2 Lógica de Conjunto de *Test Test*

Campos: los campos relacionados con este proyecto de la entidad de Conjunto de *Test Test* son:

Nombre del campo	Type	Mandatory	Read Only	Reference
<i>Test Set</i>	<i>Reference</i>	Si	No	Conjunto de <i>Test</i>
<i>Test</i>	<i>Reference</i>	Si	No	<i>Test</i>

Tabla 10.2.4 Lógica de campos de Conjunto de *Test Test*

Estos campos se muestran en el formulario principal del siguiente modo (se incluyen campos que no han sido mencionados en el conjunto de *Test Test*, ya que no se han desarrollado para el mismo):

<i>Domain</i>	<i>Test Set</i>
<i>Test</i>	

Tabla 10.2.5 Formulario de Conjunto de *Test Test*

En la lista general de la tabla, se muestran los siguientes campos en cada registro:

<i>Domain</i>	<i>Test</i>	<i>Test Set</i>
---------------	-------------	-----------------

Tabla 10.2.6 Vista de tabla de Conjunto de *Test Test*

Además, la tabla presenta una vista distinta en la *Related list* de las Versiones de *Test*:

<i>Test Set</i>

Tabla 10.2.7 Vista de tabla de Conjunto de *Test Test* en *RL Versiones de Test*

10.2.3 Lógica de *Test*

Campos: los campos relacionados con este proyecto de la entidad de *Test* son:

Nombre del campo	Type	Mandatory	Read Only	Reference
<i>Number</i>	<i>String</i>	No	Si	

<i>Runnable Version</i>	<i>Reference</i>	No	Si	<i>Versión de Test</i>
<i>Company</i>	<i>Reference</i>	No	No	<i>Company</i>
<i>Test Set</i>	<i>Reference</i>	No	No	<i>Conjunto de Test</i>
<i>Latest Version</i>	<i>Reference</i>	No	Si	<i>Versión de Test</i>
<i>Code</i>	<i>String</i>	Si	No	
<i>User logged</i>	<i>String</i>	No	No	
<i>Previous Requirements</i>	<i>String</i>	No	No	
<i>Short Description</i>	<i>String</i>	No	Si	
<i>Description</i>	<i>String</i>	No	No	

Tabla 10.2.8 Lógica de campos de Test

Estos campos se muestran en el formulario principal del siguiente modo (se incluyen campos que no han sido mencionados en el *Test*, ya que no se han desarrollado para el mismo):

<i>Number</i>	<i>LaTest Version</i>
<i>Runnable Version</i>	<i>LaTest Version.Result</i>
<i>Company</i>	<i>Code</i>
<i>Test Set</i>	
<i>User logged</i>	
<i>Previous Requirements</i>	
<i>Short Description</i>	
<i>Description</i>	

Tabla 10.2.9 Formulario de Test

En la lista general de la tabla, se muestran los siguientes campos en cada registro:

<i>Number</i>	<i>Company</i>	<i>Test Set</i>	<i>Short Description</i>	<i>LaTest Version</i>	<i>Runnable Version</i>
---------------	----------------	-----------------	--------------------------	-----------------------	-------------------------

Tabla 10.2.10 Vista de tabla de Test

Además, la tabla presenta una vista distinta en la *Related list* de los Conjuntos de *Test*:

<i>Test</i>	<i>Code</i>
-------------	-------------

Tabla 10.2.11 Vista de tabla de Test en RL de Conjunto de Test

10.2.4 Lógica de Versión de Test

Campos: los campos relacionados con este proyecto de la entidad de Versión de *Test* son:

Nombre del campo	Type	Mandatory	Read Only	Reference
<i>Test</i>	<i>Reference</i>	No	Si	<i>Test</i>
<i>Owner</i>	<i>Reference</i>	No	Si	<i>User</i>
<i>State</i>	<i>Choice</i>	No	Si	
<i>Version</i>	<i>Integer</i>	No	Si	
<i>Short Description</i>	<i>String</i>	No	Si state i= Draft	

Tabla 10.2.12 Lógica de campos de Versión de Test

Estos campos se muestran en el formulario principal del siguiente modo (se incluyen campos que no han sido mencionados en la Versión de *Test*, ya que no se han desarrollado para el mismo):

<i>Test</i>	<i>State</i>
<i>Owner</i>	<i>Version</i>
<i>Test.Company</i>	<i>Test.Code</i>
<i>Test.User logged</i>	
<i>Test.Previous Requirements</i>	

Short Description
Test.Description
Test Steps

Tabla 10.2.13 Formulario de Versión de Test

En la lista general de la tabla, se muestran los siguientes campos en cada registro:

Short Description	State	Owner	Version	Test
-------------------	-------	-------	---------	------

Tabla 10.2.14 Vista de tabla de Versión de Test

Además, la tabla presenta una vista distinta en la *Related list* de los Conjuntos de Test:

Short Description	State	Version
-------------------	-------	---------

Tabla 10.2.15 Vista de tabla de Versión de Test en RL de Test

También aparece en la *Related list* de la propia entidad Versión de Test, denominada Other Version. En este caso la lista de campos es la misma que la lista general de la tabla.

10.2.5 Lógica de Paso

Campos: los campos relacionados con este proyecto de la entidad de Paso son:

Nombre del campo	Type	Mandatory	Read Only	Reference
Order	String	No	No	
Code	String	No	No	
Step	String	No	No	
Needs Verification	True/False	No	No	
Test Version	Reference	Si	Si	Versión de Test

Tabla 10.2.16 Lógica de campos de Paso

Estos campos se muestran en el formulario principal del siguiente modo (se incluyen campos que no han sido mencionados en el Paso, ya que no se han desarrollado para el mismo):

Order	Code
Step	
Needs Verification	

Tabla 10.2.17 Formulario de Paso

En la lista general de la tabla, se muestran los siguientes campos en cada registro:

Needs Verification	Order	Step	Domain	Test Version
--------------------	-------	------	--------	--------------

Tabla 10.2.18 Vista de tabla de Paso

Además, la tabla presenta una vista distinta en la *Related list* de la Versión de Test:

Order	Step	Needs Verification	Code
-------	------	--------------------	------

Tabla 10.2.19 Vista de tabla de Paso en RL de Versión de Test

10.2.6 Lógica de Resultado de Test

Campos: los campos relacionados con este proyecto de la entidad de los Resultados de Test son:

Nombre del campo	Type	Mandatory	Read Only	Reference
Number	String	No	Si	
Result	String	No	Si	
Updated	Datetime	No	Si	
Test Run	Reference	No	Si	Test Run
Test Version	Reference	Si	Si	Versión de Test

Tabla 10.2.20 Lógica de campos de Resultado de Test

Estos campos se muestran en el formulario principal del siguiente modo (se incluyen campos que no han sido mencionados en el Resultado de *Test*, ya que no se han desarrollado para el mismo):

<i>Number</i>	<i>Test Version.Test</i>
<i>Result</i>	<i>Test Version.Version</i>
<i>Test Run.Execution Enviroment</i>	<i>Updated</i>
<i>Test Run.Run by</i>	<i>Test Run</i>
<i>Steps</i>	

Tabla 10.2.21 Formulario de Resultado de Test

En la lista general de la tabla, se muestran los siguientes campos en cada registro:

<i>Number</i>	<i>Test Version.Short Description</i>	<i>Execution Status</i>	<i>Result</i>
---------------	---------------------------------------	-------------------------	---------------

Tabla 10.2.22 Vista de tabla de Resultado de Test

Además, la tabla presenta una vista distinta en las distintas *Related list*:

Related list de Versión de *Test*:

<i>Number</i>	<i>Result</i>	<i>Test Run</i>	<i>Updated</i>
---------------	---------------	-----------------	----------------

Tabla 10.2.23 Vista de tabla de Resultado de Test en RL de Versión de Test

Related list de *Test Run*:

<i>Number</i>	<i>Test Version.Short Description</i>	<i>Test Version.Test.Code</i>	<i>Result</i>
---------------	---------------------------------------	-------------------------------	---------------

Tabla 10.2.24 Vista de tabla de Resultado de Test en RL de Test Run

10.2.7 Lógica de *Test Run*

Campos: los campos relacionados con este proyecto de la entidad de los *Test Run* son:

Nombre del campo	Type	Mandatory	Read Only	Reference
<i>Number</i>	<i>String</i>	No	Si	
<i>Started</i>	<i>Datetime</i>	No	Si	
<i>Finished</i>	<i>Datetime</i>	No	Si	
<i>Duration</i>	<i>Duration</i>	No	Si	
<i>State</i>	<i>String</i>	No	Si	
<i>Name</i>	<i>String</i>	No	Si	
<i>Execution Environment</i>	<i>Reference</i>	No	Si	<i>Entorno de Test</i>
<i>Run by</i>	<i>Reference</i>	No	Si	<i>User</i>
<i>Total Sets</i>	<i>Integer</i>	No	Si	
<i>Passed</i>	<i>Integer</i>	No	Si	
<i>Failed</i>	<i>Integer</i>	No	Si	
<i>Blocked</i>	<i>Integer</i>	No	Si	
<i>Not Finished</i>	<i>Integer</i>	No	Si	

Tabla 10.2.25 Lógica de campos de *Test Run*

Estos campos se muestran en el formulario principal del siguiente modo (se incluyen campos que no han sido mencionados en el *Test Run*, ya que no se han desarrollado para el mismo):

<i>Name</i>	
<i>Started</i>	<i>Number</i>
<i>Finished</i>	<i>Execution Environment</i>
<i>Duration</i>	<i>Run by</i>
<i>Closed</i>	<i>Total Test</i>

Tabla 10.2.26 Formulario de *Test Run*

A parte del formulario principal, encontramos el formulario secundario *Results*:

<i>Passed</i>	<i>Blocked</i>
<i>Failed</i>	<i>Not Finished</i>

Tabla 10.2.27 Pestaña *Results* en *Test Run*

En la lista general de la tabla, se muestran los siguientes campos en cada registro:

<i>Number</i>	<i>Name</i>	<i>State</i>	<i>Started</i>	<i>Finished</i>	<i>Duration</i>
<i>Execution Environment</i>	<i>Execution Environment. Company</i>	<i>Passed</i>	<i>Failed</i>	<i>Blocked</i>	<i>Not Finished</i>

Tabla 10.2.28 Vista de tabla de *Test Run*

10.2.8 Lógica de Entorno de Test

Campos: los campos relacionados con este proyecto de la entidad de los Entornos de Test son:

Nombre del campo	Type	Mandatory	Read Only	Reference
Number	String	No	Si	
Name	String	No	No	
Short Description	String	No	No	
Type	String	No	No	
Company	Reference	No	No	Company
URL	Url	No	No	

Tabla 10.2.29 Lógica de campos de Entorno de Test

Estos campos se muestran en el formulario principal del siguiente modo (se incluyen campos que no han sido mencionados en el Entorno de Test, ya que no se han desarrollado para el mismo):

Number
Name
Short Description
Type
Company
URL

Tabla 10.2.30 Formulario de Entorno de Test

En la lista general de la tabla, se muestran los siguientes campos en cada registro:

Number	Name	Company	Type
--------	------	---------	------

Tabla 10.2.31 Vista de tabla de Entorno de Test

10.2.9 Lógica de Plan de Test

Campos: los campos relacionados con este proyecto de la entidad de los Planes de Test son:

Nombre del campo	Type	Mandatory	Read Only	Reference
Number	String	No	Si	
Name	String	No	No	
Planned Start Date	Datetime	No	No	
Planned End Time	Datetime	No	No	
Planned Duration	Duration	No	No	Company
State	Integer	No	No	
Owner	Reference	No	No	User
Description	String	No	No	
Percent Complete	Percent Complete	No	Si	
Percent passed	Percent Complete	No	Si	
Percent Failed	Percent Complete	No	Si	
Percent Blocked	Percent Complete	No	Si	

Tabla 10.2.32 Lógica de campos de Plan de Test

Estos campos se muestran en el formulario principal del siguiente modo (se incluyen campos que no han sido mencionados en el Plan de Test, ya que no se han desarrollado para el mismo):

<i>Name</i>	
<i>Planned Start Date</i>	<i>Number</i>
<i>Planned End Date</i>	<i>Owner</i>
<i>Planned Duration</i>	<i>State</i>
<i>Description</i>	

Tabla 10.2.33 Formulario de Plan de Test

A parte del formulario principal, encontramos el formulario secundario Progress:

<i>Percent Complete</i>	<i>Percent passed</i>
	<i>Percent Failed</i>
	<i>Percent Blocked</i>

Tabla 10.2.34 Pestaña Progress en Plan de Test

En la lista general de la tabla, se muestran los siguientes campos en cada registro:

<i>Number</i>	<i>Name</i>	<i>State</i>	<i>Percent Complete</i>
---------------	-------------	--------------	-------------------------

Tabla 10.2.35 Vista de tabla de Plan de Test

10.2.10 Lógica de Ciclo de Test

Campos: los campos relacionados con este proyecto de la entidad de los Ciclos de Test son:

Nombre del campo	Type	Mandatory	Read Only	Reference
<i>Number</i>	<i>String</i>	No	Si	
<i>Name</i>	<i>String</i>	No	No	
<i>Planned Start Date</i>	<i>Datetime</i>	No	No	
<i>Planned End Time</i>	<i>Datetime</i>	No	No	
<i>Planned Duration</i>	<i>Duration</i>	No	No	<i>Company</i>
<i>State</i>	<i>Integer</i>	No	No	
<i>Test Plan</i>	<i>Reference</i>	No	No	<i>Plan de Test</i>
<i>Description</i>	<i>String</i>	No	No	
<i>Percent Complete</i>	<i>Percent Complete</i>	No	Si	
<i>Percent passed</i>	<i>Percent Complete</i>	No	Si	
<i>Percent Failed</i>	<i>Percent Complete</i>	No	Si	
<i>Percent Blocked</i>	<i>Percent Complete</i>	No	Si	

Tabla 10.2.36 Lógica de campos de Ciclo de Test

Estos campos se muestran en el formulario principal del siguiente modo (se incluyen campos que no han sido mencionados en el Ciclos de Test, ya que no se han desarrollado para el mismo):

<i>Name</i>	
<i>Planned Start Date</i>	<i>Number</i>
<i>Planned End Date</i>	<i>Test Plan</i>
<i>Planned Duration</i>	<i>State</i>
<i>Description</i>	

Tabla 10.2.37 Formulario de Ciclo de Test

A parte del formulario principal, encontramos el formulario secundario Progress:

<i>Percent Complete</i>	<i>Percent passed</i>
	<i>Percent Failed</i>
	<i>Percent Blocked</i>

Tabla 10.2.38 Pestaña Progress en Ciclo de Test

En la lista general de la tabla, se muestran los siguientes campos en cada registro:

<i>Number</i>	<i>Name</i>	<i>State</i>	<i>Percent Complete</i>	<i>Percent passed</i>	<i>Percent Failed</i>	<i>Percent Blocked</i>
---------------	-------------	--------------	-------------------------	-----------------------	-----------------------	------------------------

Tabla 10.2.39 Vista de tabla de Ciclo de Test

Además, la tabla presenta una vista distinta en la *Related list* de Plan de Test

<i>Number</i>	<i>Name</i>	<i>State</i>	<i>Percent Complete</i>
---------------	-------------	--------------	-------------------------

Tabla 10.2.40 Vista de tabla de Ciclo de Test en RL de Plan Test

10.2.11 Lógica de Software de Ejecución de Test

Campos: los campos relacionados con este proyecto de la entidad de los Software de Ejecución de Test son:

Nombre del campo	Type	Mandatory	Read Only	Reference
<i>Number</i>	<i>String</i>	No	Si	
<i>Name</i>	<i>String</i>	No	No	
<i>Planned Start Date</i>	<i>Datetime</i>	No	No	
<i>Planned End Time</i>	<i>Datetime</i>	No	No	
<i>Planned Duration</i>	<i>Duration</i>	No	No	<i>Company</i>
<i>State</i>	<i>Integer</i>	No	No	
<i>Test Cycle</i>	<i>Reference</i>	No	No	<i>Ciclo de Test</i>
<i>Description</i>	<i>String</i>	No	No	
<i>Assignment Group</i>	<i>Reference</i>	No	No	<i>Group</i>
<i>Assigned to</i>	<i>Reference</i>	No	No	<i>User</i>
<i>Storie</i>	<i>Reference</i>	No	No	<i>Storie</i>

Tabla 10.2.41 Lógica de campos de Software de Ejecución de Test

Estos campos se muestran en el formulario principal del siguiente modo (se incluyen campos que no han sido mencionados en el Software de Ejecución de Test, ya que no se han desarrollado para el mismo):

<i>Name</i>	
<i>Planned Start Date</i>	<i>Number</i>
<i>Planned End Date</i>	<i>Test Cycle</i>
<i>Planned Duration</i>	<i>State</i>
	<i>Story</i>
	<i>Assignment Group</i>
	<i>Assigned To</i>

Tabla 10.2.42 Formulario de Software de Ejecución de Test

En la lista general de la tabla, se muestran los siguientes campos en cada registro:

<i>Number</i>	<i>Name</i>	<i>Assignment Group</i>	<i>State</i>
---------------	-------------	-------------------------	--------------

Tabla 10.2.43 Vista de tabla de Software de Ejecución de Test

En este caso, contamos con una segunda vista que se no muestra por defecto cuando accedemos al módulo Stories Execution Suite:

<i>Number</i>	<i>Name</i>	<i>Story</i>	<i>Story. Sprint</i>	<i>Assignment Group</i>	<i>Assigned To</i>	<i>State</i>
---------------	-------------	--------------	----------------------	-------------------------	--------------------	--------------

Tabla 10.2.44 Vista alternativa de tabla de Software de Ejecución de Test

Además, la tabla presenta una vista distinta en la *Related list* de Ciclo de Test:

<i>Number</i>	<i>Name</i>	<i>Percent Complete</i>	<i>Planned Start Date</i>	<i>Planned End Date</i>
---------------	-------------	-------------------------	---------------------------	-------------------------

Tabla 10.2.45 Vista de tabla de Software de Ejecución de Test en RL de Ciclo de Test

10.2.12 Lógica de Asignación de Ejecución de Test

Campos: los campos relacionados con este proyecto de la entidad de los Asignación de Ejecución de Test son:

Nombre del campo	Type	Mandatory	Read Only	Reference
<i>Test</i>	<i>Reference</i>	<i>No</i>	<i>No</i>	<i>Test</i>
<i>Test Execution Suite</i>	<i>Reference</i>	<i>No</i>	<i>No</i>	<i>Software de Ejecución de Test</i>
<i>Assigned to</i>	<i>Reference</i>	<i>No</i>	<i>No</i>	<i>User</i>

Tabla 10.2.46 Lógica de campos de Asignación de Ejecución de Test

Estos campos se muestran en el formulario principal del siguiente modo (se incluyen campos que no han sido mencionados en el Asignación de Ejecución de Test, ya que no se han desarrollado para el mismo):

<i>Test</i>	<i>Assigned To</i>
<i>Test Execution Suite</i>	

Tabla 10.2.47 Formulario Asignación de Ejecución de Test

En la lista general de la tabla, se muestran los siguientes campos en cada registro:

<i>Test.Short Description</i>	<i>Test Execution Suite</i>	<i>Test.Company</i>	<i>Test.Result</i>	<i>Test. Runnable Version</i>
-------------------------------	-----------------------------	---------------------	--------------------	-------------------------------

<i>Assigned To</i>	<i>Test. Previous Requirement</i>	<i>Test. User Logged</i>	<i>Test.Description</i>
--------------------	-----------------------------------	--------------------------	-------------------------

Tabla 10.2.48 Vista de tabla de Asignación de Ejecución de Test

Además, la tabla presenta una vista distinta en la *Related list* de Software de Ejecución de Test:

<i>Test.Short Description</i>	<i>Test.Code</i>	<i>Test. Test Set</i>	<i>Assigned To</i>	<i>Result. Result</i>	<i>Updated</i>	<i>Updated By</i>
-------------------------------	------------------	-----------------------	--------------------	-----------------------	----------------	-------------------

Tabla 10.2.49 Vista de tabla de Asignación de Ejecución de Test en RL de Software de Ejecución de Test

10.2.13 Imágenes de la aplicación Test Management

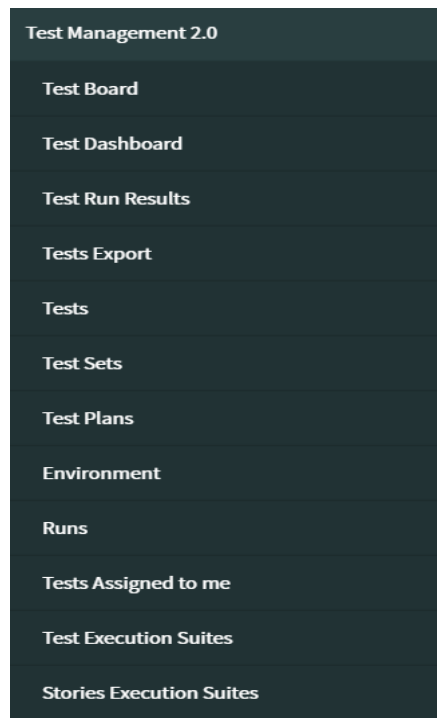


Figura 10.2.1 Menú Test Management 2.0

The dashboard shows a summary of test results and a table of test runs. The summary includes: Tests Runs Executed (46), Tests Executed (143), Tests Passed (44), Tests Failed (30), and Tests Blocked (17). The table below lists individual test runs with columns for Test run, Execution environment, Company, Test, Result, Run by, and Updated.

Test run	Execution environment	Company	Test	Result	Run by	Updated
Test Export - Test Run 2	SilverStorm - Dev Instance	SILVERSTORM	Test MR - Testing Export	Passed	Miguel Brazález	27-05-2021 09:57:53
Test Export - Test Run 1	SilverStorm - Dev Instance	SILVERSTORM	Test MR - Testing Export	Failed	Miguel Brazález	27-05-2021 09:49:52
asd	SilverStorm - Spare Instance	SILVERSTORM	Test Version #101	Passed	Alejandro Muñoz Garcia	26-05-2021 20:10:11
Test Failed	(empty)	(empty)	Related list "CARs"	(empty)	Miguel Brazález	12-05-2021 12:16:14
Test Failed	(empty)	(empty)	Test-Bea	(empty)	Miguel Brazález	12-05-2021 12:16:14
Test Mike	PBE	(empty)	Test-Bea	Failed	Miguel Brazález	12-05-2021 12:15:30
Test Mike	PBE	(empty)	CONF - Model Categories - Check the type...	Passed	Miguel Brazález	12-05-2021 12:15:30
Test Mike	PBE	(empty)	Related List de Cloud Account	Passed	Miguel Brazález	12-05-2021 12:15:30

Figura 10.2.2 Dashboard Test Management pestaña 1



Figura 10.2.3 Dashboard Test Management pestaña 2

The screenshot shows the 'Test Versions not created by Test Managers' table and a 'Company' filter dropdown.

Test Versions not created by Test Managers Table:

Company	Created by	Test Manager Test Role	Tester Test Role	abenito	amunoz	emartinez	mbrazalez	Count
(empty)		3	1	4	13	3	1	25
NTT ESPAÑA	6	1	1	8	62	2	2	80
SICPA	5	0	2	6	2	0	0	15
SILVERSTORM	0	1	0	5	0	4	0	10
TFG DEMO 1	0	0	0	3	0	0	0	3
TFG DEMO 2	0	0	0	5	0	0	0	5
TFG DEMO 3	0	0	0	1	0	0	0	1
Count	14	3	7	41	67	7	7	139

Figura 10.2.4 Dashboard Test Management pestaña 3

The screenshot shows a list of test sets with columns for Number, Name, Process, Company, and Owner.

Test sets Table:

Number	Name	Process	Company	Owner
TMTS0001549	Conjunto Test 4		TFG DEMO 3	User BPC
TMTS0001548	Conjunto Test 3		TFG DEMO 2	User BPC
TMTS0001547	Conjunto Test 2		TFG DEMO 1	User BPC
TMTS0001546	Conjunto Test 1		TFG DEMO 1	User BPC

Figura 10.2.5 Lista de Conjunto de Test

Test set
Conjunto Test 4

Number: TMTS0001549

Company: TFG DEMO

Process: -- None --

Name: Conjunto Test 4

Owner: User BPC

Category:

Update Update Test Codes

Related Links

Add to Update Set

Tests New Search Test Code Search

Test set = Conjunto Test 4

Test Code

TFG demo Test

1 to 1 of 1

Figura 10.2.6 Formulario Conjunto de Test y Related list a Test

Select Code Condition

Number of Digits:
-Is Mandatory field.
-Has to be a Integer number higher than 0.

Initial Value:
-Has to be a number.
-Has to be a Integer number greater than 0.
-Has to be less than higher number of digits.

Total length of code has to be less than 40

*Prefix

* Number of digits

Initial value

Cancel Accept

fig_04

Figura 10.2.8 UI Page Update Test Code con mensaje de error

Select Test Set

*Test set

Cancel Add

Figura 10.2.7 UI Page Add Test to Test Set

Tests New Add to Test Set Search Number Search

All > Company Name contains TFG

	Number	Company	Test set	User logged	Short description	Latest version	Runnable version
<input checked="" type="checkbox"/>	TEMT0003731	TFG DEMO 1	Conjunto Test 2	TFG demo Test	TFG demo Test	2	(empty)
<input type="checkbox"/>	TEMT0003730	TFG DEMO 1	Conjunto Test 2	TFG demo Test	TFG demo Test	1	(empty)
<input type="checkbox"/>	TEMT0003729	TFG DEMO 2	Conjunto Test 3	TFG demo Test	TFG demo Test	5	3
<input type="checkbox"/>	TEMT0003728	TFG DEMO 3	Conjunto Test 4	test	TFG demo Test	1	(empty)

1 to 4 of 4

Figura 10.2.9 Lista de Test

Test TFG demo Test

Number: TEMENT0003731 Latest version: 2

Runnable version: Result: -- None --

Company: TFG DEMO 1 Code: TFG010

Test set: Conjunto Test 2

User logged: TFG demo Test

Previous Requirements: TFG demo Test

Short description: TFG demo Test

Description: TFG demo Test

Update

Related Links

Add to Update Set

Test versions Search Short description Search

Test = TFG demo Test

Short description	State	Version
TFG demo Test	Draft	1
TFG demo Test	Draft	2

Figura 10.2.10 Formulario de Test con related list Versiones de Test

Test version

Test: TFG demo Test State: Draft

Owner: Alejandro Muñoz Garcia Version: 2

Company: TFG DEMO 1 Code: TFG010

Test set: Conjunto Test 2

User logged: TFG demo Test

Previous Requirements: TFG demo Test

* Short description: TFG demo Test

Description: TFG demo Test

Test Steps

Step	Needs verification
Test Step 1	false
Test Step 2	true

Add Step

Update Ready Create New Version Delete Test

Related Links

Add to Update Set

Other Versions (1) Steps (2) Test results

Steps New Search Order Search

Test version = 2

Step	Needs verification	Code
100 Test Step 1	false	
200 Test Step 2	true	

Figura 10.2.11 Formulario de Version de Test y related list Paso

Steps Test Step 1

Order: 100 Code:

Step: Test Step 1

Needs verification:

Update

Figura 10.2.12 Formulario de Paso

Test result - TMTR0001382

Number: TMTR0001382

Test: TFG demo Test

Result: Failed

Version: 1

Execution environment: Entorno Demo 1

Updated: 31-05-2021 22:40:38

Run by: User SDC

Test run: tst demo 1

Test paso 1

Test paso 2: Failed

Test paso 3

Test paso 4: Passed

Figura 10.2.13 Formulario Resultado de Test

Test run - tst demo 1

Name: tst demo 1

Started: 31-05-2021 22:40:27

Finished: 31-05-2021 22:40:38

Duration: 11 Seconds

State: Closed

Number: TRUN0001127

Execution environment: Entorno Demo 1

Run by: User SDC

Total tests: 4

Results

Passed: 1

Failed: 1

Blocked: 0

Not finished: 2

Test results Search Number Search

Number	Short description	Code	Result
TMTR0001382	TFG demo Test	TFG	Failed
TMTR0001383			Not finished
TMTR0001384	test prueba	cp1	Passed

Figura 10.2.14 Formulario Test Run con related list Resultados de Test

Test plan - TPLAN0001023

Name: Test demo 1

Planned start date: 01-06-2021 00:00:00

Planned end date: 02-06-2021 00:00:00

Planned duration: 1 Day

Description:

Number: TPLAN0001023

Owner: Alejandro Muñoz García

State: Open

Progress

Percent complete: 50

Percent passed: 25

Percent failed: 25

Percent blocked: 0

Related Links

Planning console

Test cycles Search Number Search

Number	Name	State	Percent complete
TCY10001023	test demo cycle	Pending	50%

Figura 10.2.15 Formulario Plan de Test con related list Ciclo de Test

Test cycle - TCYL0001038

Name: test demo cycle

Planned start date: 01-06-2021 00:00:00

Planned end date: 02-06-2021 00:00:00

Planned duration: 1 Day

Description: test demo cycle

Number: TCYL0001038

Test plan: TPLAN0001023

State: Pending

Progress

Percent complete: 50

Percent passed: 25

Percent failed: 25

Percent blocked: 0

Test execution suites

Number	Name	Percent complete	Planned start date	Planned end date
TEXS0001047	Test execution Suite 1	50%	01-06-2021 00:00:00	02-06-2021 00:00:00

Figura 10.2.16 Formulario Ciclo de Test con related list Software de Ejecución de Test

Test execution suite - Test execution Suite 1

Name: Test execution Suite 1

Planned start date: 01-06-2021 00:00:00

Planned end date: 02-06-2021 00:00:00

Planned duration: 1 Day

Number: TEXS0001047

Test cycle: TCYL0001038

State: Current

Story:

Assignment group: TFG User Groups

Assigned to: User SDC

Test execution assignments

Short description	Code	Test set	Assigned to	Result	Updated	Updated by
TFG demo Test	TFG	Conjunto Test 2	User SDC	Failed	31-05-2021 22:40:27	User SDC
test	5	Conjunto Test 1	User SDC	Not finished	31-05-2021 22:40:27	User SDC
test prueba	cp1	Conjunto Test 1	User SDC	Passed	31-05-2021 22:40:27	User SDC
TFG demo Test	TFG010	Conjunto Test 2	User SDC	Not finished	31-05-2021 22:40:27	User SDC

Figura 10.2.17 Formulario Software de Ejecución de Test con related list Asignación Ejecución de Test

Test execution assignments

Run

Search: Test Short description

All > Assigned to is User SDC > Test execution suite Active = true

4 total Test execution assignments

Short description	Test execution suite	Company	Result	Runnable version	Assigned to	Previous Requirements	User logged	Description
TFG demo Test	Test execution Suite 1	TFG DEMO 1	Failed	1	User SDC	TFG demo Test	TFG demo Test	TFG demo Test
test	Test execution Suite 1	TFG DEMO 1	Not finished	(empty)	User SDC	sadsd	sad	sad
test prueba	Test execution Suite 1	TFG DEMO 1	Passed	1	User SDC			
TFG demo Test	Test execution Suite 1	TFG DEMO 1	Not finished	1	User SDC	TFG demo Test	TFG demo Test	TFG demo Test

Figura 10.2.18 Tabla Asignación de Ejecución de Test

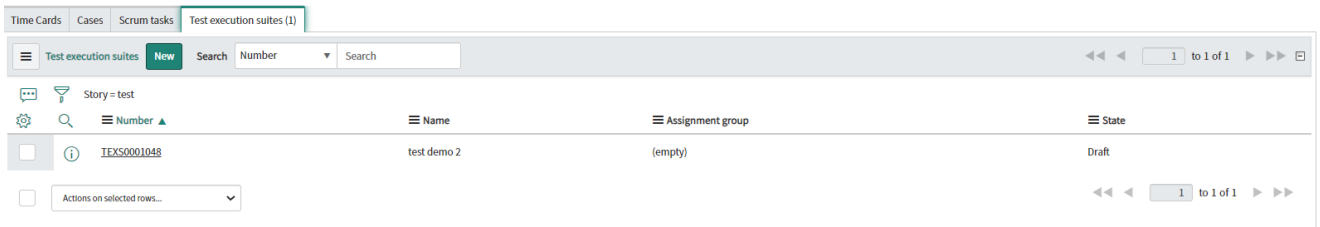


Figura 10.2.19 Related list de Software de Ejecución de Test en formulario Story

Run Tests

Run name

Execution environment

Cancel Run

Figura 10.2.20 UI Page Run Test

TFG demo Test

TEST 1

Test 2

Test 3

Comment

Pause Done

Figura 10.2.21 Run Test con Paso que necesita verificación en blocked

Tests Exports										
Search										Updated
Search										1 to 10 of 10
All > Company = TFG DEMO 1										
Company Owner Process Category Test set Code Short description Description Previous Requirements User logged										
=TFG DEMO 1 Search Search Search Search Search Search Search Search Search Search										
<input type="checkbox"/>	<i>i</i>	TFG DEMO 1	User BPC	(empty)	Conjunto Test 2	TFG010	TFG demo Test	TFG demo Test	TFG demo Test	TFG demo Test
<input type="checkbox"/>	<i>i</i>	TFG DEMO 1	User BPC	(empty)	Conjunto Test 2	TFG010	TFG demo Test	TFG demo Test	TFG demo Test	TFG demo Test
<input type="checkbox"/>	<i>i</i>	TFG DEMO 1	User BPC	(empty)	Conjunto Test 2	TFG010	TFG demo Test	TFG demo Test	TFG demo Test	TFG demo Test
<input type="checkbox"/>	<i>i</i>	TFG DEMO 1	User BPC	(empty)	Conjunto Test 2	TFG	TFG demo Test	TFG demo Test	TFG demo Test	TFG demo Test
<input type="checkbox"/>	<i>i</i>	TFG DEMO 1	User BPC	(empty)	Conjunto Test 2	TFG	TFG demo Test	TFG demo Test	TFG demo Test	TFG demo Test
<input type="checkbox"/>	<i>i</i>	TFG DEMO 1	User BPC	(empty)	Conjunto Test 2	TFG	TFG demo Test	TFG demo Test	TFG demo Test	TFG demo Test
<input type="checkbox"/>	<i>i</i>	TFG DEMO 1	User BPC	(empty)	Conjunto Test 2	TFG	TFG demo Test	TFG demo Test	TFG demo Test	TFG demo Test
<input type="checkbox"/>	<i>i</i>	TFG DEMO 1	User BPC	(empty)	Conjunto Test 1	cp1	test prueba			
<input type="checkbox"/>	<i>i</i>	TFG DEMO 1	User BPC	(empty)	Conjunto Test 1	cp1	test prueba			
<input type="checkbox"/>	<i>i</i>	TFG DEMO 1	User BPC	(empty)	Conjunto Test 1	cp1	test prueba			

Figura 10.2.22 Database View Test Export

Test Run Step Results										
Search										Test set Company
Search										1 to 20 of 30
All > Step Code is not empty > Test set Company = TFG DEMO 1										
Company Owner Test run Process Category Test set Code Short description Description Previous Requirements User logged										
=TFG DEMO 1 Search Search Search Search Search Search Search Search Search Search										
<input type="checkbox"/>	<i>i</i>	TFG DEMO 1	Alejandro Muñoz Garcia	Test demo 2	(empty)	Conjunto Test 2	TFG	TFG demo Test	TFG demo Test	TFG demo Test
<input type="checkbox"/>	<i>i</i>	TFG DEMO 1	Alejandro Muñoz Garcia	Test demo 2	(empty)	Conjunto Test 2	TFG	TFG demo Test	TFG demo Test	TFG demo Test
<input type="checkbox"/>	<i>i</i>	TFG DEMO 1	Alejandro Muñoz Garcia	Test demo 2	(empty)	Conjunto Test 2	TFG	TFG demo Test	TFG demo Test	TFG demo Test
<input type="checkbox"/>	<i>i</i>	TFG DEMO 1	Alejandro Muñoz Garcia	Test demo 2	(empty)	Conjunto Test 2	TFG	TFG demo Test	TFG demo Test	TFG demo Test
<input type="checkbox"/>	<i>i</i>	TFG DEMO 1	Alejandro Muñoz Garcia	test emo 1	(empty)	Conjunto Test 2	TFG	TFG demo Test	TFG demo Test	TFG demo Test
<input type="checkbox"/>	<i>i</i>	TFG DEMO 1	Alejandro Muñoz Garcia	test emo 1	(empty)	Conjunto Test 2	TFG	TFG demo Test	TFG demo Test	TFG demo Test
<input type="checkbox"/>	<i>i</i>	TFG DEMO 1	Alejandro Muñoz Garcia	test emo 1	(empty)	Conjunto Test 2	TFG	TFG demo Test	TFG demo Test	TFG demo Test
<input type="checkbox"/>	<i>i</i>	TFG DEMO 1	Alejandro Muñoz Garcia	test emo 1	(empty)	Conjunto Test 2	TFG	TFG demo Test	TFG demo Test	TFG demo Test
<input type="checkbox"/>	<i>i</i>	TFG DEMO 1	Alejandro Muñoz Garcia	Test demo 2	(empty)	Conjunto Test 2	TFG	TFG demo Test	TFG demo Test	TFG demo Test

Figura 10.2.23 Database View Test Run Results