

Modelo de Casos de Uso

Testify

OSLO

Ojeda Valeria – Sly Eduardo
Levipichun Emilio – Oyarzo Malena



El modelo de casos de uso describe la funcionalidad propuesta del nuevo sistema.

Este modelo se basa en la descripción de elementos o usuarios externos al sistema (actores) y de la funcionalidad del sistema (casos de uso). Un Modelo de Casos de Uso describe los requerimientos funcionales de un actor en términos de las interacciones, la utilización de este modelo presenta el sistema desde la perspectiva de su uso y esquematiza como proporcionará valor a sus usuarios.

El modelo de casos de uso sirve como acuerdo entre clientes y desarrolladores para limitar las funciones con que dispondrá el sistema.

Tabla de contenido

Requerimientos	4
<i>Requerimientos Funcionales</i>	<i>4</i>
<i>Requerimientos no Funcionales</i>	<i>5</i>
Actores	5
<i>Administrador</i>	<i>5</i>
<i>Gestor de Prueba.....</i>	<i>6</i>
<i>Tester.....</i>	<i>6</i>
<i>Desarrollador</i>	<i>7</i>
<i>Invitado</i>	<i>7</i>
Casos de Uso	8
Diagramas Asociados.....	11
<i>Diagrama de casos de uso.....</i>	<i>11</i>
<i>Diagramas de Paquetes</i>	<i>¡Error! Marcador no definido.</i>
<i>Diagrama de componentes.....</i>	<i>¡Error! Marcador no definido.</i>
¡Error! Marcador no definido.	
<i>Diagrama de Clases.....</i>	<i>12</i>

Modelo de Casos de Uso

Requerimientos

Requerimientos Funcionales

Los requerimientos funcionales del sistema describen la funcionalidad o los servicios que se espera que éste provea. En este apartado se describe lo que el sistema Testify tendrá que hacer, los factores que afectan al producto y satisfacen los requerimientos.

A continuación, se narran los requerimientos funcionales detectados y luego se detallan en tablas individuales para cada uno:

RF01: El sistema debe permitir el alta, modificación y aprobado de los casos de prueba.

RF02: El sistema debe permitir la creación de múltiples proyectos para un mismo usuario. RF03: Un proyecto estará dividido en Hitos.

RF04: Cada Hito contendrá 1 o más de un caso de prueba.

RF05: Cada caso de prueba puede contener uno o más precondiciones.

RF06: Las precondiciones pueden estar activas o desestimadas.

RF07: Cada caso de prueba puede contener uno o más pasos a seguir.

RF08: Los pasos a seguir pueden estar activos o desestimados.

RF09: Los casos de prueba deberán ser identificados según su tipo: Documentación o Código.

RF10: El sistema contendrá los roles: Administrador, Gestor de Prueba, Tester, desarrollador e Invitado.

RF11: Los roles serán asignados a los usuarios y por proyecto, a excepción del rol de Administrador.

RF12: Solo el Rol Administrador podrá crear las cuentas de los usuarios.

RF13: El sistema deberá utilizar la API de google para acceder.

RF14: El gestor de prueba solo podrá visualizar los proyectos asignados a su usuario.

RF15: El Tester solo podrá visualizar los casos de prueba a los que fue asignado.

RF16: El desarrollador solo podrá visualizar los casos de prueba a los que fue asignado.

RF17: El Tester solo podrá actualizar el resultado de los casos de prueba y/o agregar comentarios.

RF18: El desarrollador podrá agregar comentarios a los casos de prueba a los que fue asignado.

RF19: La página principal del sistema debe mostrar los gráficos del proyecto de estado de pruebas y cantidad de pruebas, como su estado Aprobado o Pendiente.

RF20: El sistema debe permitir adjuntar documentos a los casos de prueba.

RF21: Cada proyecto puede tener configurado una tasa mínima de aprobación, el proyecto será marcado como Aprobado cuando el porcentaje de casos de uso aprobados supere la tasa.

RF22: Los casos de uso que se tendrán en cuenta para el cálculo de aprobación serán aquellos que fueron asignados a un Tester.

RF23: El sistema permitirá exportar en un documento PDF los casos de prueba, gráficos y resultados.

Requerimientos no Funcionales

En este documento, se abordarán los requerimientos no funcionales del sistema, los cuales son cruciales para definir las características que pueden influir en el desempeño y las limitaciones del producto. Los requerimientos no funcionales se centran en aspectos que van más allá de las funcionalidades básicas del sistema y que impactan su rendimiento general, fiabilidad y usabilidad. En el caso de Testify, se identifican los siguientes:

RNF01: El sistema debe pensarse solamente para ser utilizado en computadoras de escritorio o notebooks, no deben considerarse la adaptación a otros dispositivos.

RNF02: El acceso al sistema debe ser vía Web y online.

Actores

En el sistema Testify, los actores representan los diferentes usuarios y sistemas que interactúan con las funcionalidades del sistema. Cada actor tiene un conjunto de características y relaciones específicas que definen cómo se relacionan con los casos de uso del sistema. A continuación, se describen los actores identificados:

Administrador

Actor	Administrador	A01
-------	---------------	-----

Descripción	<i>El Administrador tiene acceso completo al sistema y es responsable de gestionar usuarios, roles, permisos y proyectos. Además, puede crear, modificar y eliminar casos de prueba, hitos y otras entidades del sistema</i>
Características	<i>Acceso completo a todas las funcionalidades del sistema, incluido el mantenimiento de usuarios y roles.</i>
Relaciones	<i>Se relaciona con los casos de uso de gestión de usuarios, roles, permisos, proyectos y hitos.</i>
Referencias	<i>Accede a todos los casos de uso del sistema, diagramas de secuencia, etc.</i>

Comentarios
<i>Solo el Administrador puede crear cuentas de usuario.</i>

Gestor de Prueba

Actor	Gestor de Prueba	A02
Descripción	<i>El Gestor de Prueba está encargado de administrar los casos de uso asignados a su proyecto, y gestionar el progreso de las pruebas. Tiene permisos para crear y asignar casos de prueba a otros usuarios.</i>	
Características	<i>Acceso limitado a los proyectos y casos de prueba asignados a su usuario.</i>	
Relaciones	<i>Relación con la gestión de casos de prueba, asignación de pruebas y consulta de proyectos.</i>	
Referencias	<i>Participa en la creación y seguimiento de pruebas.</i>	

Comentarios
<i>Solo puede visualizar y administrar los proyectos y casos de uso asignados.</i>

Tester

Actor	Tester	A03
--------------	---------------	-----

Descripción	<i>El Tester está encargado de ejecutar pruebas en los casos de uso que se le han asignado y de reportar resultados. Puede agregar comentarios y actualizar el estado de las pruebas.</i>
Características	<i>Puede consultar y actualizar el resultado de las pruebas asignadas a su usuario.</i>
Relaciones	<i>Relación con los casos de prueba y la actualización de resultados de pruebas.</i>
Referencias	<i>Participa en la ejecución de pruebas y en la modificación de resultados.</i>

Comentarios
<i>Solo tiene acceso a los casos de uso que le han sido asignados.</i>

Desarrollador

Actor	Desarrollador	A04
Descripción	<i>El Desarrollador trabaja en los casos de uso asignados para su desarrollo o corrección. Puede agregar comentarios y consultar el progreso de sus casos de uso asignados.</i>	
Características	<i>Puede agregar comentarios sobre los casos de uso a los que está asignado y seguir su progreso.</i>	
Relaciones	<i>Relación con la gestión de casos de uso y el seguimiento del avance en las pruebas.</i>	
Referencias	<i>Participa en el desarrollo y análisis de los casos de uso.</i>	

Comentarios
<i>Solo tiene acceso a los casos de uso que le han sido asignados.</i>

Invitado

Actor	Invitado	A05
Descripción	<i>El Invitado tiene acceso limitado al sistema, solo puede consultar información básica de los proyectos y los casos de uso sin poder realizar modificaciones.</i>	
Características	<i>Solo puede consultar información. No tiene permisos de modificación ni de creación.</i>	
Relaciones	<i>Relación con la consulta de casos de uso y proyectos.</i>	
Referencias	<i>Solo participa en la consulta de datos.</i>	

Comentarios
<i>Tiene acceso limitado a la información del sistema.</i>

Casos de Uso

En este apartado se detallan los Casos de Uso del sistema, los cuales describen las interacciones clave entre los usuarios y el sistema para alcanzar objetivos específicos. Cada caso de prueba especifica una funcionalidad particular del sistema, describiendo los pasos que un actor, o conjunto de actores, realiza para completar una tarea. Además, se identifican los actores involucrados en cada caso de prueba, ya sean personas o sistemas externos, que interactúan con la plataforma.

CU01 - CRUD Casos de Prueba	
Actores	Administrador, Gestor de Prueba
Descripción	Los usuarios pueden crear, modificar, eliminar y consultar casos de prueba dentro de un proyecto.

CU02 - Consultar Casos de Prueba	
Actores	Invitado, Desarrollador, Tester
Descripción	Los usuarios pueden visualizar todos los casos de prueba.

CU03 - Asignar Caso de Prueba	
Actores	Administrador, Gestor de Prueba, Desarrollador
Descripción	El Desarrollador asigna casos de uso a Testers o el Administrador asigna casos de prueba al Desarrollador para su evaluación o implementación.

CU04 - Adjuntar Documentos	
Actores	Administrador, Tester, Desarrollador, Gestor de Prueba
Descripción	El usuario puede subir y adjuntar documentos a los casos de prueba para respaldar los resultados de las pruebas.

CU05 - Comentar Caso de Prueba	
Actores	Administrador, Gestor de Prueba, Desarrollador, Tester
Descripción	El usuario puede agregar comentarios en los casos de prueba asignados.

CU06 - Exportar Casos de Prueba y Resultados	
Actores	Gestor de Prueba, Administrador
Descripción	El usuario puede exportar los casos de prueba, gráficos y resultados del proyecto en un archivo PDF.

CU07 - Actualizar Pasos a Seguir	
Actores	Administrador, Tester
Descripción	El usuario actualiza el resultado de un caso de prueba asignado, indicando si es satisfactorio o fallido. El usuario accede al caso de prueba, revisa los pasos a seguir de un CP y cambia el estado de Pendiente a Satisfactorio o de Pendiente a Error Detectado.

CU08 - CRUD Proyectos	
Actores	Administrador
Descripción	El usuario administra la creación, modificación, eliminación y búsqueda de proyectos en el sistema

	El usuario selecciona el proyecto y establece el porcentaje requerido para marcar el proyecto como satisfactorio.
--	---

CU09 - Consultar Proyectos	
Actores	Invitado
Descripción	El usuario puede visualizar la información general de todos los proyectos. Al consultar la sección de proyectos, pueden acceder a los detalles relevantes y seleccionar el proyecto que deseen revisar.

CU10 - Consultar Proyectos Asignados	
Actores	Gestor de Pruebas
Descripción	El usuario ve los proyectos que les han sido asignados. Al acceder a la sección de proyectos, los usuarios pueden revisar la información detallada de aquellos proyectos en los que están involucrados.

CU11 - CRUD Hitos	
Actores	Administrador
Descripción	El usuario administra los hitos del sistema, pudiendo elegir entre crear, modificar, eliminar y consultar hitos.

CU12 - CRUD Permisos	
Actores	Administrador
Descripción	El usuario administra los permisos asignados a los diferentes roles dentro del sistema, pudiendo crear, modificar, eliminar y buscar permisos.

CU13 - Consultar Hitos	
Actores	Invitado
Descripción	El usuario puede visualizar los hitos de un proyecto específico. Después de seleccionar un proyecto, acceden a la lista de hitos asociados, obteniendo una visión clara de su progreso.

CU14 - CRUD Roles	
-------------------	--

Actores	Administrador
Descripción	El usuario administra los roles en el sistema, pudiendo elegir entre crear, modificar, eliminar y buscar roles.

CU15 - CRUD Usuarios	
Actores	Administrador
Descripción	El usuario administra los usuarios del sistema, pudiendo crear, modificar, buscar y eliminar, y el sistema actualiza la lista de usuarios.

CU16 - Gestionar Rol	
Actores	Administrador
Descripción	El usuario gestiona los roles asignados a un usuario específico. Desde esta sección, puede modificar, eliminar y buscar roles asignados, asegurando que cada usuario tenga los permisos adecuados para sus tareas.

Diagramas Asociados

En el marco del desarrollo del sistema Testify, es fundamental describir cómo se organizan y relacionan sus distintos componentes a través de diagramas de UML. Estos diagramas no solo ofrecen una visión clara de la arquitectura interna del sistema, sino que también facilitan la comprensión de las interacciones entre los elementos que lo componen. A continuación, se presentan los diagramas de casos de uso, de paquetes, de componentes y de clases, los cuales permiten visualizar las diferentes capas de abstracción del sistema.

Diagrama de casos de uso

El diagrama de casos de uso representa gráficamente los casos de uso que tiene un sistema. Se define un caso de uso como cada interacción supuesta con el sistema a desarrollar, donde se representan los requisitos funcionales. Es decir, se está diciendo lo que tiene que hacer un sistema y cómo.

En la siguiente figura se muestra un ejemplo de casos de uso, donde se muestran cinco actores (Administrador, Gestor de Prueba, Tester, Desarrollador e Invitado) y las operaciones que pueden realizar sus roles.

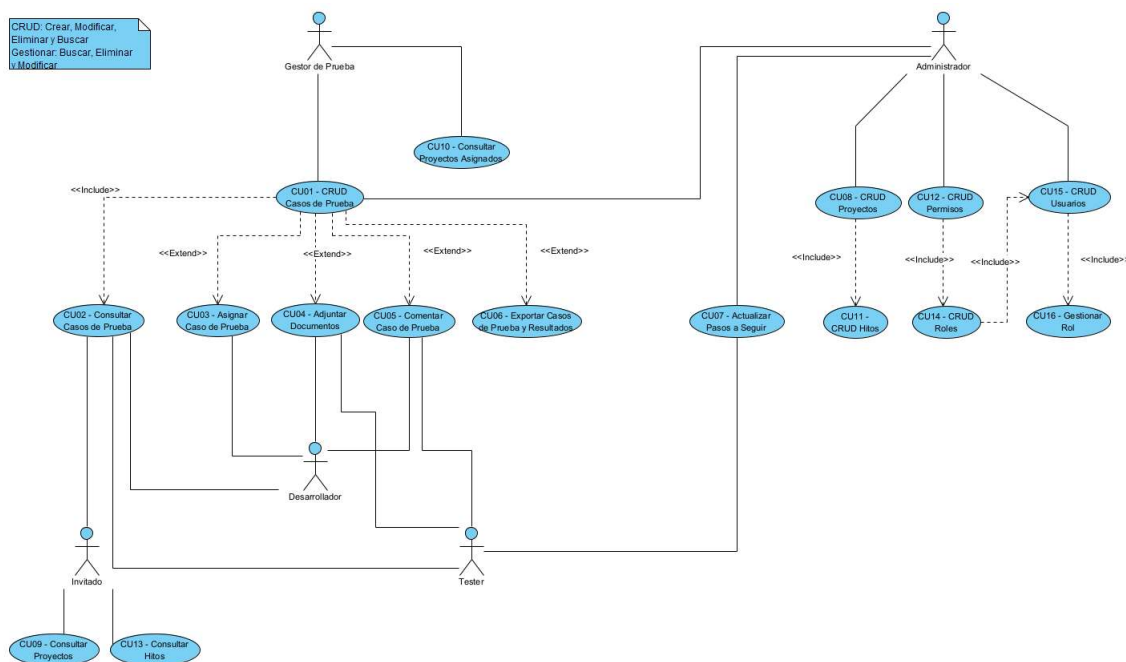


Diagrama de Clases

El diagrama que se presenta es un diagrama de clases UML, el cual modela la estructura de un sistema de gestión de proyectos, roles, casos de prueba y permisos. A continuación, se describen las principales características y relaciones:

Clases Principales

1. Proyecto

- Atributos: id, nombre, estado, aprobacion.
- Métodos: agregarHito(Hito), agregarCasoDePrueba(CasoDePrueba), calcularTasaAprobacion().
- Relaciones: Un proyecto contiene una colección de hitos y casos de prueba, y puede calcular su tasa de aprobación basada en los casos de prueba que contiene.

2. Usuario

- Atributos: id, nombre, email, esAdmin.
- Métodos: autenticar().
- Relaciones: Un usuario está asociado a uno o varios roles a través de la clase RolesAsignados. Los usuarios pueden ser desarrolladores o testers dentro de los casos de prueba.

3. RolesAsignados

- Atributos: id, usuario, rol, proyecto.
- Relaciones: Esta clase actúa como intermediaria entre usuarios, roles y proyectos, definiendo qué rol tiene cada usuario en un proyecto específico.

4. Rol

- Atributos: id, nombre.
- Relaciones: Un rol puede tener múltiples permisos a través de la clase RolPermiso. Está asociado con usuarios en proyectos mediante RolesAsignados.

5. Permiso

- Atributos: id, nombre.
- Relaciones: Un permiso puede estar relacionado con uno o más roles a través de la clase RolPermiso, que establece qué permisos tiene cada rol en el sistema.

6. Hito

- Atributos: id, nombre, proyecto.
- Relaciones: Un hito pertenece a un proyecto y puede incluir múltiples casos de prueba.

7. CasoDePrueba

Atributos: id, nombre, hito, tipoCasoPrueba, comentario, resultadoEsperado, resultadoObtenido, desarrollador, tester, estado.

Métodos: adjuntarDocumentos(), asignarCasoDePrueba(), exportarCasoDePrueba(), actualizarPasos().

Relaciones: Un caso de prueba pertenece a un hito y puede tener uno o más pasos y precondiciones. Los usuarios que participan en el desarrollo y pruebas de los casos son los desarrolladores y testers.

8. Precondicion

Atributos: id, orden, descripcion, estado.

Relaciones: Cada precondición pertenece a un caso de prueba y define requisitos previos que deben cumplirse para ejecutar el caso de prueba.

9. Paso

Atributos: id, orden, descripcion, estado.

Relaciones: Un paso también pertenece a un caso de prueba y define una secuencia de acciones que deben realizarse dentro del caso.

10. Adjunto

Atributos: id, descripcion, document.

Relaciones: Los adjuntos están relacionados con casos de prueba, permitiendo la asociación de documentos como soporte o evidencia.

Relaciones Entre Clases

Un usuario puede tener múltiples roles en diferentes proyectos a través de RolesAsignados.

Un rol puede tener múltiples permisos a través de la relación RolPermiso, lo que define los niveles de acceso o funcionalidad disponibles para el usuario que tiene ese rol.

Un proyecto está compuesto por varios hitos, y cada hito puede contener uno o más casos de prueba.

Un caso de prueba puede estar vinculado a un tester y a un desarrollador, lo que representa los usuarios responsables de las pruebas y el desarrollo del caso.

Cada caso de prueba tiene asociadas una serie de precondiciones y pasos, los cuales detallan los requisitos y la secuencia de acciones que deben seguirse para validar el caso.

Los adjuntos se asocian a los casos de prueba como documentos relevantes para el contexto de las pruebas.

Enumeraciones

EstadoProyectoEnum: Define los posibles estados en los que puede estar un proyecto: pendiente, en_progreso, completado, cancelado.

EstadoTipoCasoPruebaEnum: Enumera los tipos de casos de prueba posibles: funcionalidad, componente, modulo, clase, documentacion.

EstadoCasoPruebaEnum: Enumera los posibles estados de un caso de prueba: pendiente, en_pruebas, corregir, aprobado.

