

# De las Necesidades de Usuarios a los Requisitos de calidad: Hacia una herramienta basada en SQUARE aplicada a historias de usuario

## From User Needs to Quality Requirements: Towards a SQUARE-based tool applied to user stories

Presentación: 21/08/2024

**Javier Saldarini, Juan P. Mansilla, Gastón Ferreyra, Claudio Carrizo, Sofía Bovo,**

Universidad Tecnológica Nacional – Facultad Regional San Francisco

saldarinijavier@gmail.com; juampimansilla17@gmail.com; gastonferreyra49@gmail.com; cj carrizo77@gmail.com; sofibovo501@gmail.com

**Carlos Salgado, Alberto Sánchez, Mario Peralta**

Universidad Nacional de San Luis - Facultad de Ciencias Físico-Matemáticas y Naturales

csalgado@unsl.edu.ar, asanchez@unsl.edu.ar, mperalta@unsl.edu.ar

### Resumen

El uso de Historias de Usuario es una de las prácticas más comunes en el desarrollo de software guiado por los métodos ágiles; permiten introducir los requerimientos en el proceso de desarrollo, y consiste en una descripción corta de una funcionalidad valuada por un usuario o cliente de un sistema.

Habitualmente, no es común que las especificaciones de Historias de Usuario estén relacionadas a aspectos que no sean de tipo funcionales como, por ejemplo, atributos de calidad que debería tener un producto de software (Performance, Usabilidad, etc.)

En el presente trabajo se sientan las bases para el desarrollo de una web app que se integre a JIRA Software, cuya principal funcionalidad está centrada en mapear las necesidades de calidad de las partes interesadas con características de calidad de producto de software en el contexto de historias de usuario, tomando como marco de referencia la norma ISO/IEC 25030:2019.

**Palabras clave:** Historias de Usuario, Calidad de producto, ISO/IEC 25030, Métodos Ágiles

### Abstract

The creation of User Stories is one of the most common practices in software development guided by agile methods. They fulfill the purpose of introducing requirements into the development process and are a short description of a functionality valued by a user or client of a system.

In practice, it is not very common for User Story specifications to be related to aspects other than strictly functional ones, such as attributes related to the quality that a software product should have, for example: Performance, Usability, etc.

This work presents a proposal that lays the foundation for the development of a web app that integrates with JIRA, and whose main functionality is focused on mapping the quality needs of stakeholders with software product quality characteristics in the context of user stories and taking the ISO/IEC 25030:2019 standard as a reference framework.

**Keywords:** User Stories, Product Quality, ISO/IEC 25030, Agile Methods

### Introducción

Las metodologías ágiles para el desarrollo de software surgieron en la década de los noventa, están basadas en el denominado Manifiesto Ágil (Manifiesto for Agile Software Development, s.f.), y sus principales características están centradas en el desarrollo iterativo e incremental, las entregas frecuentes, la priorización de los requisitos, la constante interacción con el cliente, la adaptación al cambio, el trabajo colaborativo en equipo, etc.

Autores como Sommerville (Sommerville, 2011) y Pressman (Pressman, 2005) coinciden en que las más reconocidas en la industria del software son: Scrum (Scrum Alliance, s.f.), Programación Extrema (Extreme Programming, s.f.) y Crystal (CRYSTAL, Agile project management, s.f.).

En este sentido, Sommerville (Sommerville, 2011) expresa que el nuevo software se desarrolla rápidamente para aprovechar las actuales oportunidades. En consecuencia, en la actualidad la entrega y el desarrollo rápidos son por lo general el requerimiento fundamental de los sistemas de software. De hecho, muchas empresas están dispuestas a negociar la calidad del software y el compromiso con los requerimientos, para lograr con mayor celeridad la implementación que necesitan del software.

Según ISO/IEC 25030 (ISO/IEC 25030:2019, 2019) cuando hablamos de calidad del software es importante identificar y especificar los requisitos de calidad como parte de los requisitos del sistema, el software y los datos, porque encontrar el equilibrio adecuado entre los requisitos de calidad, además de los requisitos funcionales bien especificados, es un factor crítico de éxito para cumplir con los objetivos de las Partes Interesadas (PI).

En el trabajo denominado "*Herramientas y buenas prácticas para el aseguramiento de calidad de software con metodologías ágiles*" (Mercado-Ramos, 2015) se describe que existen distintos frameworks y/o herramientas para asegurar la calidad del software en un proyecto ágil, pero aún existen desafíos por solucionar y que además, que por las características y naturaleza de un proyecto desarrollado con metodologías ágiles es necesario contar con un conjunto de buenas prácticas, recomendaciones y herramientas, que, según sea el caso, se apliquen en determinado momento del proyecto.

En este último tiempo han surgido distintas herramientas o plataformas que permiten llevar a cabo la gestión de proyectos de desarrollo de software. En (IONOS Cloud S.L.U., s.f.) se describen y comparan herramientas como Jira, ClickUp, Sprintly, Redmine, Stackfield, Kanbanize, Open Project, Hive, entre otras. También se indica que con más de 65.000 usuarios alrededor del mundo, Jira Software (Atlassian, s.f.) se encuentra entre las soluciones más utilizadas para la gestión de proyectos y que esta solución facilita el desarrollo e integración de add-ons a través del marketplace que ofrece Atlassian (Atlassian, s.f.).

En este sentido, también se consideró conveniente investigar aquellas apps alojadas en el Marketplace de Atlassian (Atlassian, s.f.) para la gestión de la calidad y/o el testing. Principalmente se encontraron plugins dedicados al testing funcional tales como QMetry (QMetry, s.f.), Xray (Xray, s.f.) o Zephyr Squad (Zephyr Squad, s.f.), los cuales permiten el diseño y ejecución de test funcionales, integración con otras herramientas, reportes y analíticas. Sin embargo, estas soluciones no están específicamente enfocadas para la gestión de la calidad de producto de software o requisitos no funcionales.

Visto lo antes mencionado, es que, en este trabajo se presenta una propuesta que sienta las bases para el desarrollo de una web app que se integre a JIRA Software, cuya su principal funcionalidad está centrada en mapear las necesidades de calidad de las partes interesadas con las características de calidad de producto de software en el contexto de historias de usuario, tomando como marco de referencia la norma ISO/IEC 25030:2019 (ISO/IEC 25030:2007, 2007).

## Desarrollo

### **Especificación de Requisitos de Calidad del Software - ISO/IEC 25030:2019**

El propósito de la familia de normas ISO/IEC 25000 (ISO/IEC 25000:2014, 2014) es ayudar al desarrollo y adquisición del producto de software, estableciendo criterios para la *especificación de requerimientos de calidad del producto de software, su medición y evaluación*.

En ese contexto, la Norma ISO/IEC 25030 (ISO/IEC 25030:2019, 2019) permite llevar a cabo el proceso de especificación de requerimientos de calidad de software. Según se expresa en esta norma, los requisitos de calidad del software, así como todos los demás requisitos, no pueden ser vistos en forma aislada, sino que deben considerarse en un contexto más amplio.

En la Figura 1 se puede observar el proceso propuesto por ISO/IEC 25030 (ISO/IEC 25030:2019, 2019) para la obtención de los requisitos de calidad desde las necesidades de las partes interesadas hasta obtener los requisitos del sistema / software, también, esta norma proporciona una serie de pasos para guiar esa especificación, los cuáles se pueden observar en la Figura 2.

En esta última se puede ver que para cumplir con el paso (1) se debe interactuar con los modelos de calidad que propone la familia de normas ISO/IEC 25000 (ISO/IEC 25000:2014, 2014), para nuestro caso utilizaremos el modelo de calidad de producto establecido en la norma ISO/IEC 25010 (ISO/IEC 25010:2011, 2011) con la cual podremos identificar las características y sub-características de calidad necesarias. Del mismo modo, para cumplir con el paso (5) y establecer las métricas necesarias para medir los atributos de calidad de producto que se deriven del Paso (1) es que utilizaremos la norma ISO/IEC 25023 (ISO/IEC 25023:2016, 2016 (E)).

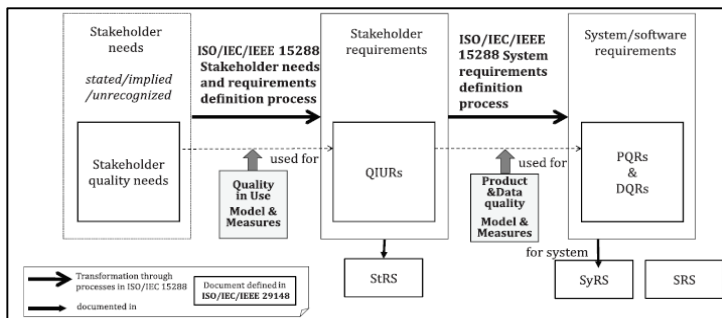


Figura 1. Desde las necesidades de las partes interesadas hasta los requisitos del sistema /

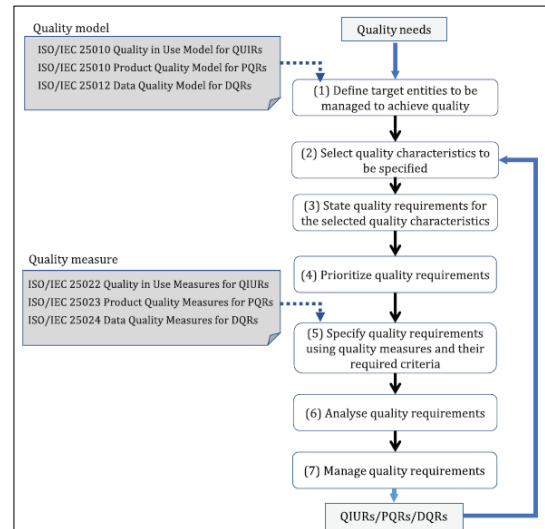


Figura 2. Pasos para definir los requisitos de calidad

### Modelo de Cardinalidad entre Entidades

A fin de poder modelar de mapeo entre las necesidades de la PI y los requisitos de calidad de software, obtenidos a través de proceso que contiene un serie de pasos, establecidos en ISO/IEC 25030 (ISO/IEC 25030:2019, 2019), se desarrolló un modelo de cardinalidad compuesto por Entidades, Atributos, Relaciones y su Cardinalidad (Figura 3).

A continuación, se muestra un ejemplo:

Un REQUISITO DE CALIDAD está compuesto por una o muchas NECESIDADES DE CALIDAD y una o muchas SUBCARACTERÍSTICAS DE CALIDAD

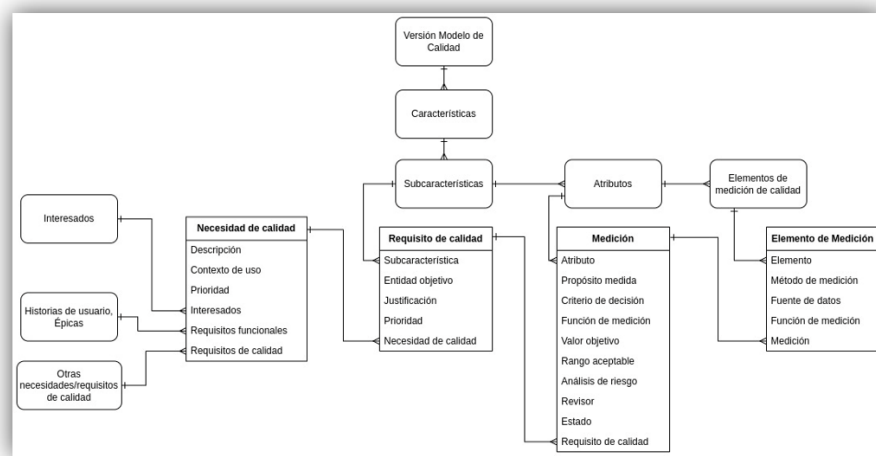


Figura 3. Modelo de cardinalidad basado en proceso de ISO/IEC 25030

### Funcionalidad de la Web App

En las Figuras 4 y 5 se presenta la interface de integración de la web app con Jira Software (Atlassian, s.f.), la cual permite mapear las necesidades de las PI con las características de calidad de producto, teniendo en cuenta el proceso descrito en ISO/IEC 25030 (ISO/IEC 25030:2019, 2019); también se utiliza como referencia el modelo de calidad de producto ISO/IEC 25010 (ISO/IEC 25010:2011, 2011), que permite identificar las características y sub-características de calidad, y la norma ISO/IEC 25023 (ISO/IEC 25023:2016, 2016 (E)) para establecer las medidas de calidad.

Para explicar las funcionalidades se detalla el paso a paso junto a un ejemplo práctico basado en una Épica que contiene tres historias de usuario.

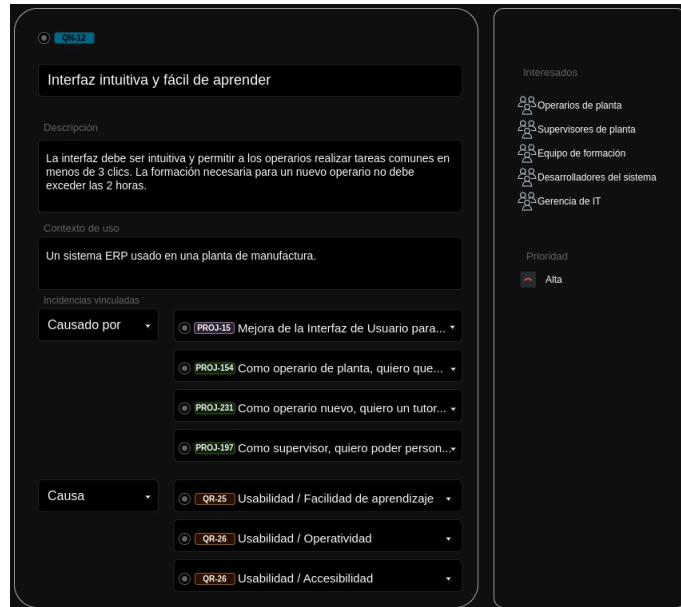


Figura 4. Interface 1 de integración con JIRA

1- Para especificar correctamente un requisito de calidad, se debe partir de una *Necesidad de Calidad*. Para ello se utiliza una Incidencia de Jira de tipo “*Necesidad de Calidad*”, la cual requiere los siguientes campos:

- **Título:** un nombre resumido que explique la necesidad en cuestión;
- **Descripción:** describir de forma detallada en qué consiste esa necesidad de calidad en términos del negocio  
*La interfaz debe ser intuitiva y permitir a los operarios realizar tareas comunes en menos de 3 clics. La formación necesaria para un nuevo operario no debe exceder las 2 horas.*
- **Contexto de uso:** describir el entorno en el que se enmarca esta necesidad de calidad  
*Un sistema ERP usado en una planta de manufactura*
- **Prioridad:** puede ser muy alta, alta, media, baja o muy baja  
*Alta*
- **Interesados:** seleccionar cuales de todos los interesados en este proyecto son afectados de alguna manera por esta necesidad de calidad
  - *Operarios de planta*
  - *Supervisores de planta*
  - *Equipo de formación*
  - *Desarrolladores del sistema*
  - *Gerencia de IT*
- **Incidentes Jira que causan esta necesidad de calidad:** una Necesidad de Calidad puede ser causada por Requisitos funcionales (Épicas, Historias de Usuario) o por otras Necesidades de Calidad. Para ello se utiliza la funcionalidad de Jira para “Vincular incidencias”, que permite relacionar incidencias entre sí y elegir un tipo de relación. En este caso se elige el tipo de relación “Causado por” junto con una épica, dos historias de usuario y dos necesidades de calidad.

#### ***Requisitos funcionales relacionados***

*Épica:* Mejora de la Interfaz de Usuario para Operarios

- *Historia de Usuario 1:* Como operario de planta, quiero que el sistema me permita completar tareas en no más de 3 clics para mejorar mi eficiencia.
- *Historia de Usuario 2:* Como operario nuevo, quiero un tutorial interactivo que me guíe a través de las funciones principales para poder empezar a usar el sistema rápidamente.
- *Historia de Usuario 3:* Como supervisor, quiero poder personalizar la interfaz para que se adapte mejor a las tareas diarias de mi equipo.

*Nota:* por una cuestión de espacio de aquí en adelante solo se detallan los pasos a seguir para la Historia de Usuario 2

- **Requisitos de calidad causados por la necesidad de calidad:** luego de analizar qué sub-características aplican para cumplir con la necesidad de calidad, se deben crear otras incidencias Jira de tipo “Requisito de Calidad”. Los Requisitos de Calidad generados a partir de la Necesidad de Calidad se vinculan mediante la relación “Causa”.
  - *Usabilidad / Facilidad de aprendizaje:*
    - **Justificación:** Dado que uno de los objetivos es que los operarios nuevos puedan aprender a usar el sistema rápidamente, la facilidad de aprendizaje es crucial. Esto incluye cómo el sistema apoya a los usuarios en la comprensión y dominio de sus funcionalidades.

- *Prioridad: Alta*

2- En Figura 5 se muestra que Jira le asigna un identificador único a cada incidencia de forma automática. Las incidencias cuyo identificador comienza con “QN-” referencian a una Necesidad de Calidad y si comienza con “QR-” se está referenciando a un Requisito de Calidad.

Las incidencias de tipo “Requisito de Calidad” requieren los siguientes campos:

- **Título:** *Usabilidad / Facilidad de aprendizaje*
- **Justificación:** indicar porque esa sub-característica permite satisfacer la necesidad de calidad de origen  
*Dado que uno de los objetivos es que los operarios nuevos puedan aprender a usar el sistema rápidamente, la facilidad de aprendizaje es crucial. Esto incluye como el sistema apoya a los usuarios en la comprensión y dominio de sus funcionalidades.*
- **Entidad objetivo:** seleccionar un Componente de Jira el cual se vería afectado por esta sub-característica  
*Sistema de Gestión de operaciones en Planta (SGOP)*
- **Prioridad:**  
*Alta*
- **Vínculo “Causado por”:** generado automáticamente cuando se crea el Requisito de Calidad como contraparte del vínculo “Causa” en la Necesidad de calidad:  
*Interface intuitiva y fácil de aprender*
- **Medidas causadas por el requisito de calidad:** luego de analizar qué atributos de calidad aplican para medir este requisito de calidad, se deben crear otras incidencias Jira de tipo “Medición”. Las Mediciones generadas a partir del Requisito de Calidad se vinculan mediante la relación “Causa” y sus identificaciones comienzan con “QM-”.

#### *Exhaustividad de la guía del usuario*

- *Propósito de la Medida de la Calidad del Software (Información Necesaria):* Evaluar la cobertura de la documentación del usuario con el fin de asegurar que todas las funciones necesarias estén adecuadamente explicadas, permitiendo a los usuarios comprender y utilizar el sistema eficientemente.
- *Criterios de Decisión:* Si la exhaustividad de la guía del usuario es inferior al 90%, se deben realizar revisiones adicionales de la documentación para cubrir las funciones faltantes.
- *Función de Medición:*
  - $X=A/B$
  - *Elementos de Medida de la Calidad Utilizados:*
    - A: Número de funciones descritas en la documentación del usuario u otros elementos de ayuda según se requiera.
    - B: Número de funciones implementadas que se requieren para ser documentadas.
  - *Método de Medición:*
    - A: Conteo de funciones documentadas.
    - B: Conteo de funciones implementadas.
  - *Fuente de Dato(s):*
    - A: Revisión de la documentación del usuario.
    - B: Revisión del sistema para identificar todas las funciones implementadas.
  - *Valor objetivo:*
    - 95%
  - *Rango de valores aceptables:*
    - 90% - 100%

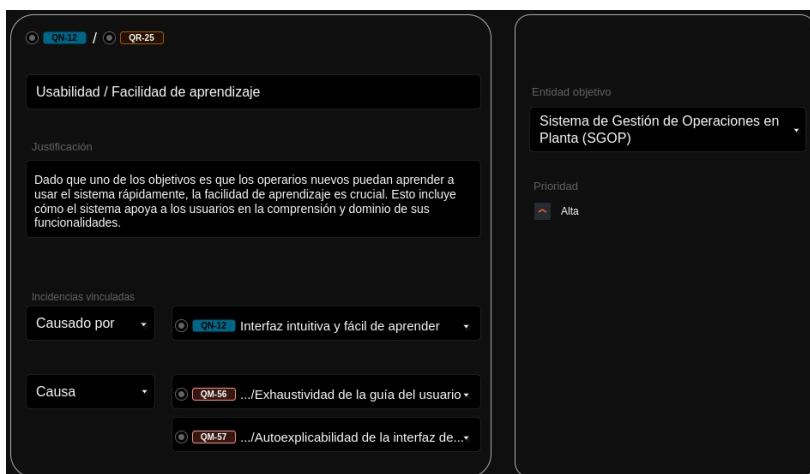


Figura 5. Interface 2 de integración con JIRA

## Conclusiones

Si bien los resultados de esta propuesta son iniciales, demuestran la utilidad de poder contar con herramientas que estén basadas en un estándar de calidad reconocido a nivel mundial, como lo es la familia de normas ISO/IEC 25000, y que las mismas se puedan integrar a herramientas de gestión de proyectos que dan soporte a los marcos ágiles para el desarrollo del software como lo es Jira Software.

En base a lo desarrollado, se pudo observar que es factible poder establecer características, sub-características y medidas de calidad partiendo de las necesidades de calidad de las partes interesadas desde las historias de usuario, utilizando como marco lo establecido en ISO/IEC 25030 y su relación con las normas ISO/IEC 25010 y ISO/IEC 25023.

Se plantea a futuro seguir trabajando en esta línea de investigación, por un lado, se pretende terminar de manera completa la web app y realizar ensayos en múltiples proyectos de desarrollo de software guiados por metodologías ágiles, por otro, desarrollar la Incidencia de tipo "Medición", la cual, estará configurada para ser testeada, lo que implica un tratamiento similar al que dan herramientas de testing como XRay a los "Test Case". Esto permitirá planificar las tareas de medición, registrar los valores de cada elemento de la función de medición, documentar los resultados y verificar el cumplimiento de los requisitos y necesidades de calidad.

## Referencias

- Atlassian. (n.d.). Atlassian. Retrieved from <https://bit.ly/3SwMmDd>
- CRYSTAL, Agile project management. (n.d.). Retrieved from <http://crystalmethodologies.org>
- Estayno, M., & Meles, J. (2014). El Rol del Product Owner en la definición y validación de las user stories. *Ciencia y Tecnología*(14), 145-162.
- Extreme Programming. (n.d.). Retrieved from <https://www.agilealliance.org/glossary/xp>
- IONOS Cloud S.L.U. (n.d.). IONOS Cloud. Retrieved from <https://www.ionos.es/digitalguide/paginas-web/desarrollo-web/alternativas-a-jira/>
- ISO/IEC 25000:2014. (2014). Systems and software engineering-Systems and software Quality Requirements and Evaluation (SQuaRE). ISO-Guide to SQuaRE.
- ISO/IEC 25010:2011. (2011). Systems and software engineering-Systems and software Quality Requirements and Evaluation (SQuaRE)-System and software quality models.
- ISO/IEC 25023:2016. (2016 (E)). Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Measurement of system and software product quality.
- ISO/IEC 25030:2007. (2007). Software engineering — Software product Quality Requirements and Evaluation (SQuaRE) — Quality requirements. ISO.
- ISO/IEC 25030:2019. (2019). Ingeniería de Software – Requisitos de la calidad del producto de software (SQuaRE) – Requisitos de Calidad. ISO.
- Jeffries, R. (2019). Three-C's Revisited. Retrieved from <https://ronjeffries.com/articles/019-01ff/3cs-revisited/>
- Manifiesto for Agile Software Development. (n.d.). Retrieved from <http://agilemanifesto.org/>
- Mercado-Ramos, V. Z. (2015). HERRAMIENTAS Y BUENAS PRÁCTICAS PARA EL ASEGURAMIENTO DE CALIDAD DE SOFTWARE CON METODOLOGÍAS ÁGILES. 6(1), 73-83.
- Pressman, R. (2005). Ingeniería de Software. (6 ed.). Mcgraw-Hill.
- QMetry. (n.d.). QMetry - Gestión de testing para Jira. Retrieved from <https://marketplace.atlassian.com/apps/1215144/qmetry-test-management-for-jira-qtm4j?tab=overview&hosting=cloud>
- Scrum Alliance. (n.d.). Retrieved from <https://resources.scrumalliance.org/Article/quick-guide-things-scrum>
- Sommerville, I. (2011). Ingeniería de Software (9° ed.). México: PEARSON EDUCACIÓN.
- Xray. (n.d.). Xray - Gestión de testing para Jira. Retrieved from <https://marketplace.atlassian.com/apps/1211769/xray-test-management-for-jira?tab=overview&hosting=cloud>
- Zephyd Squad. (n.d.). Zephyd Squad - Gestión de testing para Jira . Retrieved from <https://marketplace.atlassian.com/apps/1014681/zephyr-squad-test-management-for-jira?tab=overview&hosting=cloud>