

TG Project Management: Seguimiento y documentación de actividades orientada al usuario desarrollador.

Valentín Peluso, Ariel Pasini¹, Nicolás Galdámez, Luciano Marrero²

Instituto de Investigación en Informática LIDI (III-LIDI)*
Facultad de Informática – Universidad Nacional de La Plata 50 y 120 La Plata Buenos Aires
*Centro Asociado Comisión de Investigaciones Científicas de la Pcia. de Bs. As. (CIC)

{apasini, ngaldamez, lmarrero}@lidi.info.unlp.edu.ar
valentinpeluso@gmail.com

Abstract: Los proyectos de software actuales no experimentan las mismas necesidades que los proyectos de hace algunos años. La evolución del mercado actual requiere nuevos tiempos de respuesta ante soluciones informáticas. Realizar una buena gestión del proyecto no garantiza su éxito, pero realizar una mala gestión seguramente lleve a su fracaso. Para que un proyecto resulte exitoso es necesario que su gestión y administración sean eficientes, para ello es fundamental la planificación, la experiencia, la buena comunicación y además es importante contar con herramientas de apoyo durante su gestión. Hoy en día se dispone de una gran variedad de herramientas para la gestión de proyectos, entre ellas, algunas muy populares como Trello y Jira. Ambas herramientas al intentar cubrir todo tipo de proyecto carecen de ciertos aspectos importantes, agilidad y facilidad de uso, pero poca estructura (Trello) o viceversa (Jira). La elección de herramientas en un proyecto debe ser un equilibrio entre las características del proyecto y las características de las herramientas elegidas con el objetivo de no modificar las reglas de negocio. El propósito de TG Project Management es nutrir el mercado actual con una herramienta orientada al usuario desarrollador, mejorando así, ciertos aspectos como el seguimiento del avance del proyecto, el control de tareas realizadas por el equipo, la estimación de esfuerzos, la organización del cronograma, etc. y una interacción fluida con herramientas de desarrollo colaborativo como Github.

Keywords: Ingeniería de Software, gestión de proyectos de software, metodologías ágiles, gestores de proyectos.

¹ ORCID: 0000-0002-4752-7112

² ORCID: 0000-0002-7507-1562

1 Introducción

En la Ingeniería de Software la gestión de proyectos de software es una temática de estudio e innovación constante. Gestionar un proyecto involucra la aplicación del conocimiento, habilidades, herramientas y técnicas a las actividades necesarias para alcanzar los objetivos deseados. En la gestión de proyectos existen dos tipos de procesos bien distinguidos: administración general y administración de proyecto. Si bien poseen aspectos en común, no deben ser confundidos. La administración general está pensada para un sistema de gestión de una organización a largo plazo y sin un límite conocido, en cambio, la administración de proyectos es finita y con objetivos específicos, que una vez cumplidos determinan su fin [1].

Administrar un proyecto de software implica atender diferentes procesos de gestión (alcance, recursos, cronograma, etc.), para asegurar que todo se ejecute de acuerdo con lo definido y conseguir los objetivos propuestos. Lograr el éxito de un proyecto involucra poner énfasis en varias cuestiones, desde la elección del personal que participará en el proyecto, pasando por la metodología de proceso aplicada, hasta las herramientas técnicas de apoyo que se utilizarán. Contar con personal de experiencia ligado al conocimiento y a las buenas prácticas, hace que su rendimiento mejore con respecto a proyectos previos. La metodología de gestión de un proyecto es un elemento clave para el conjunto de las actividades que involucra, elegir erróneamente la metodología a utilizar es costoso y puede llevar a su fracaso. Una metodología debe reflejar el tamaño, la duración y la complejidad del proyecto, como así también, adaptarse a la industria, la cultura y al nivel de madurez de la organización.

Durante el ciclo de vida de un proyecto se pueden utilizar diversas herramientas de apoyo y para diversos objetivos. Las herramientas para la gestión del proyecto (Trello y Jira) y las herramientas para el control de versiones (GIT) son el foco de estudio para este artículo. El uso de herramientas durante la gestión de proyectos puede enriquecer y potenciar la capacidad de los usuarios, como también asistir en el avance del proyecto, gestionar tareas del personal, realizar estimaciones, mejorar la comunicación con el cliente y mantener una documentación actualizada [1], [2] y [3].

En la sección dos se describen los dos desafíos más importantes de la gestión de los proyectos a gran escala, la administración de los recursos humanos y de la distribución de tareas entre los recursos. En la siguiente sección los inconvenientes encontrados en los principales gestores de proyectos. En la sección TG Project Management, se describe la herramienta propuesta para cubrir las falencias planteadas en el capítulo anterior. Por último las conclusiones y trabajos futuros.

2 Gestión en proyectos complejos y de amplio alcance.

2.1 Contexto

Actualmente la evolución de la industria del software crece a pasos agigantados. Se prioriza la agilidad y flexibilidad en el desarrollo, el producto se construye y al mismo tiempo se modifican e introducen requerimientos, los cuales cambian para adaptarse a un contexto versátil. El cliente generalmente posee una visión clara del proyecto, pero

la innovación y la velocidad a la que evoluciona el entorno de negocio, no le permite prever como será en detalle el resultado final. Se debe actuar rápidamente y utilizar tácticas orientadas a la respuesta ágil y flexible. Los proyectos de software se vuelven cada vez más complejos y a medida que aumenta la complejidad las estrategias deben actualizarse. El grado de adaptación a estos cambios es lo que permite mejorar y crecer sin perder la calidad de un producto para que perdure en el tiempo [1], [4] y [5].

2.2 Gestión del personal del proyecto

En un proyecto de software participan distintos tipos de usuarios, algunos vinculados directamente y otros de forma indirecta. Cuanto mayor mano de obra se necesita, mayor será el control a realizar, además, no siempre es sencillo contar con personas altamente capacitadas, lo habitual es que los más experimentados formen a los más novatos, para lograr esto es fundamental la buena comunicación de todo el equipo de trabajo. Es importante que los participantes de un proyecto se interesen en el mismo y que sepan trabajar en equipo, aunque el desempeño de una persona depende de muchos factores (motivos personales, ámbito laboral, etc.), es fundamental que el personal sienta la importancia de cada una de sus actividades y el impacto de estas en el proyecto, seguramente, así se aumenta la productividad.

Es común que durante el ciclo de vida de un proyecto, cambie alguno de sus requerimientos o sufra bajas y altas de personal. Estas situaciones impactan directamente sobre el equipo de trabajo. Se hace necesario tener estrategias y herramientas de gestión para absorber tales cambios con el menor costo posible [1] y [7].

Los grandes proyectos de software involucran un gran número de integrantes, lo que dificulta la organización, coordinación y asignación del trabajo a realizar. Es por esto por lo que la estrategia más utilizada es el trabajo en equipos organizados de forma autónoma para cumplir un objetivo dentro del proyecto, aun así, es importante contar con herramientas y estrategias de apoyo para gestionar la comunicación y la convivencia de los grupos, disminuyendo así la probabilidad de riesgos durante el proyecto.

2.3 Gestión de las tareas

En algunas ocasiones se puede confundir el concepto de tarea con el de proyecto, ya que ambos poseen atributos en común, personas, recursos escasos (tiempo, dinero, etc.), necesitan ser planificados y requieren de continuo control. En general las tareas son repetitivas y se mantienen en el tiempo mientras que los proyectos son únicos y temporales.

Es importante que un proyecto posea un director de proyecto, quién será el responsable del control y administración de todas las actividades durante todo el proceso de gestión. El director del proyecto deberá ser el integrador y facilitador, tanto en lo que respecta al proyecto como al equipo humano. Independientemente de la metodología de procesos elegida, tradicional o ágil, la dirección de los equipos de

trabajo suele ser piramidal. Cada equipo cuenta con un líder interno, quién conoce mejor el proyecto, las reglas de negocios y además es el responsable de la interacción con el resto del personal externo al equipo.

En proyectos de gran escala, es fundamental la supervisión, comunicación y articulación de los líderes de equipos, tanto con sus dirigidos como con sus superiores y pares. Contar con estrategias y herramientas de apoyo para cada tarea involucrada en la dirección de un equipo es esencial para su éxito [4] y [5].

3 Gestores de proyectos

Existen numerosas posibilidades al momento de elegir una herramienta para gestionar proyectos de software, pero no todas se adaptan a todos los tipos de proyectos, dicha elección es una balanza entre las características del proyecto y las características ofrecidas por la herramienta.

Las organizaciones deben enfrentar algunos inconvenientes al momento de gestionar proyectos grandes y complejos. Algunos de ellos son:

Configuración por defecto: con la ambición de abarcar todo tipo de proyectos, algunas herramientas poseen configuraciones muy genéricas.

Control del proyecto: en la medida que un proyecto aumenta en complejidad y tamaño, es común realizar divisiones de tareas en equipos de trabajo y aplicar diferentes estrategias de gestión y control. En algunas ocasiones existe un alto acoplamiento entre los equipos de trabajo, lo que agrega complejidad en el uso de la herramienta para supervisar el trabajo conjunto y le quita agilidad a la gestión del proyecto.

Equipos y roles: un punto a tener en cuenta es la división y administración del personal en equipos, similar al problema anterior pero enfocado a la autogestión de sus participantes. Disponer de equipos con muchos integrantes dificulta la organización y control de tareas.

Proceso de desarrollo, planificación y verificación: la alta competitividad del mercado hace que las organizaciones intenten lograr la mayor agilidad en la implementación de los cambios con el objetivo de alcanzar sus metas. Más allá de la moda del desarrollo ágil, en algunas ocasiones es necesario realizar un “mix” con las metodologías tradicionales. Esto es un reto al momento de elegir una herramienta que permita mantener la satisfacción de todos los stakeholders.

Búsqueda de historias de usuario: en los proyectos complejos y de gran escala, el número de historias de usuario es alto, por lo tanto, se necesita algún método para agilizar la búsqueda de alguna de ellas. Seguramente existan muchas historias con características en común en su resolución, lo cual, al momento de resolver una historia, se necesite consultar por otras.

La mayoría de los gestores de proyectos son genéricos y admiten configuraciones para cualquier tipo de proyecto. Pretenden abarcar la mayor cantidad de tipos de proyectos posibles. En muchas ocasiones se dejan de lado cuestiones importantes

como el proceso de licitación, la verificación del software, la agilidad en ciertos puntos del desarrollo, etc. [1], [8] y [9],

4 TG Project Management (TG)

TG Project Management, es una nueva propuesta en herramientas de software para la gestión de proyectos complejos que tiene su foco en la eficiencia del proceso de gestión y en la posibilidad de acotar la complejidad de las cuestiones anteriormente descriptas [1] y [9].

4.1 Configuraciones por defecto

TG adopta las mejores prácticas de las metodologías ágiles (Scrum y Kanban), aplicando características de cada metodología según sea conveniente. Con esto mejora la agilidad en el proceso de gestión al eliminar la necesidad de definir una configuración determinada [8] y [10].

TG posee métodos para organizar y administrar el trabajo a través de la definición de roles, equipos y módulos de trabajo.

TG permite la definición de requerimientos e historias de usuarios asociadas a los requerimientos.

TG implementa el modelo de iteraciones (Sprints) de Scrum, permitiendo que cada equipo de trabajo organice y administre el desarrollo de sus historias independientemente de los demás.

TG es una herramienta orientada al usuario desarrollador. Posee características esenciales que aumentan la agilidad durante el proceso de desarrollo.

4.2 Creación de un proyecto

Crear un proyecto en TG es un proceso simple y ágil, simplemente con definir un nombre para identificar al mismo se tiene un nuevo proyecto con las configuraciones por defecto definidas. Figura 1.

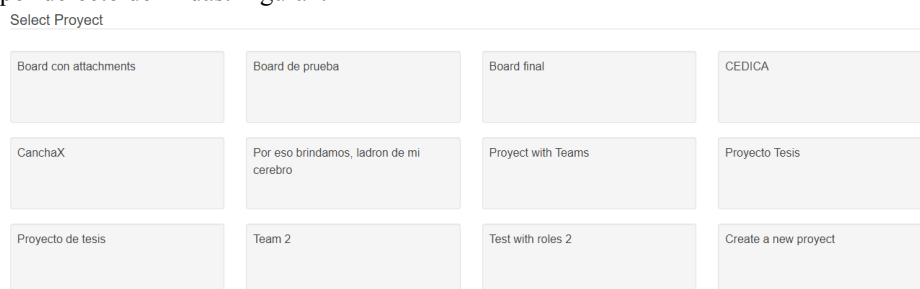


Fig. 1. Listado de proyectos creados.

4.3 Gestión del trabajo y del personal

TG es apto para la gestión de grandes proyectos de software, lo cual brinda la posibilidad de dividir los proyectos en módulos y el personal en equipos. TG permite la creación de equipos en cualquier etapa del proyecto. Una vez definido el equipo, se le pueden asignar miembros y a cada miembro el rol (master, user, owner y other) que cumplirá en el equipo (Figura 2). Hay que tener en cuenta que un mismo usuario no puede pertenecer a dos equipos simultáneamente. Antes de asignar un usuario a un equipo, TG solicita un correo electrónico para realizar la verificación de que el usuario este registrado en Trello y en Github.

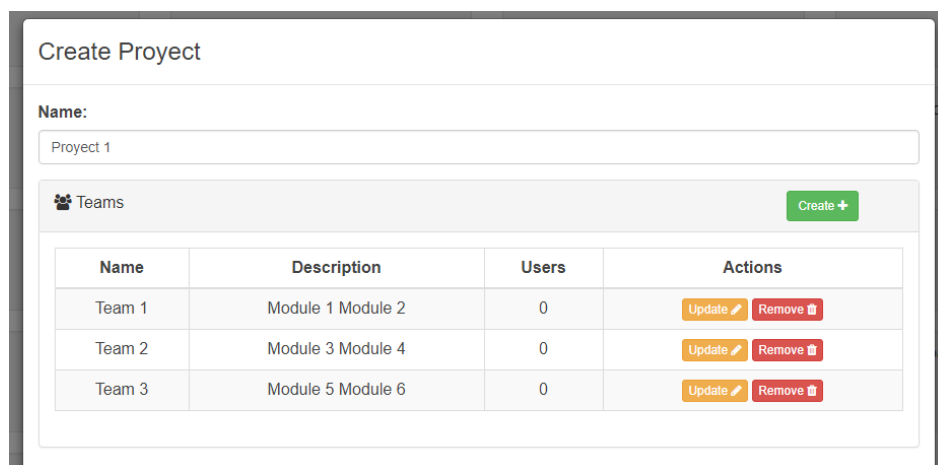


Fig. 2. Creación de equipos para un proyecto en particular.

4.4 Gestión del desarrollo

Una vez creado el proyecto y haber asignado el personal que participará del mismo, TG establece el modelo de proceso de desarrollo con el cual se trabajará. Se optó por el modelo en etapas de la Ingeniería de Software: requisitos, diseño, implementación, verificación y mantenimiento. TG traduce estas etapas en: crear un requerimiento, crear las historias de usuarios asociadas al requerimiento, asignar responsables, asignar historias de usuario a una iteración (Sprint), desarrollar la historia de usuario, vincular la historia de usuario con Github (a través de un pull request), verificar la historia de usuario y validar la solución propuesta.

TG organiza estas etapas en cuatro grandes secciones o vistas, independizando algunos procesos con el objetivo de reducir la complejidad de la herramienta. Figura 3.

Product Backlog Sprints Development Verification and Validation

Fig. 3. Organización de la gestión del desarrollo

Pila de producto (product backlog): desde aquí se crean, modifican y eliminan requerimientos de usuarios. Esta posibilidad solo la tienen aquellos usuarios con roles “master” u “owner”. Para cada requerimiento se puede especificar: título, descripción, prioridad, estimación del esfuerzo, razón de existencia, origen y dependencias (Figura 4).

Fig. 4. Pila de producto.

Iteraciones (sprints): desde aquí se gestionan las historias de usuario. Esta posibilidad solo la tienen los usuarios con rol “master”. Para cada historia se puede indicar: título, descripción, reglas de negocio, responsable, cantidad de días estimados, reporte de estado, prioridad, relaciones con otras historias, archivos adjuntos e información adicional para la historia. Además, se cuenta con información importante, como las wikis, las cuales brindan información de apoyo al desarrollador en el objetivo de las historias y la pila de producto, pero desde una perspectiva del “master”. Figura 5.

Además, en esta sección TG lista los requerimientos de la pila de producto, de esta manera se consigue agilizar el proceso de identificación de un requerimiento en particular. Las historias de usuario son un elemento importante de comunicación en las metodologías ágiles, son el medio para brindar la información que el desarrollador necesita para incorporar o modificar una funcionalidad del producto de software. En TG todas las historias de usuario son creadas a partir de un requerimiento de la pila de producto.

En TG todas las historias de usuario pasan por los siguientes estados: “Not started”, “Ready for dev”, “Dev completed”, “Ready for test” y “Closed”. Estos estados son utilizados para la gestión del desarrollo y marcan la etapa en la que se encuentra una historia.

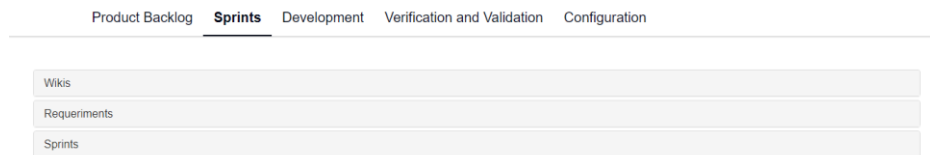


Fig. 5. Gestión de las iteraciones (Sprints)

Desarrollo (Development): desde aquí se realiza la gestión del proceso de desarrollo en la etapa de implementación. Esta solapa está destinada a que cada integrante del equipo sea el responsable de administrar y gestionar el desarrollo de las historias asignadas. Esta vista es accedida solamente desde el rol “member”.

Verificación y Validación (Verification and Validation): desde aquí se puede decir que la historia se encuentra aceptada desde un punto de vista de la calidad de la solución propuesta, pero aún debe pasar por un ambiente de prueba. Desde esta vista TG le permite al “master” seleccionar aquellas historias de la iteración actual y aceptarlas o rechazarlas en base a la validación que han recibido las mismas. Validar una historia, consiste en tratar de descubrir defectos funcionales, comprobando que la solución propuesta cumple con las expectativas del cliente. Si una historia es rechazada, significa, que no cumple con las expectativas del cliente o que no se encuentra acorde a alguna regla definida, es aquí, que el “master” debe regresar la historia para desarrollar una nueva solución y reparar los errores encontrados. Una vez que la historia ha sido aceptada, TG asume que se ha generado un incremento dentro del sistema, esto es que el requerimiento al cual se encuentra asociado la historia de usuario avanza y además la iteración actual posee una historia más resuelta. Cuando se llega a este punto se considera finalizado el proceso de gestión del desarrollo de una historia. TG almacena todas las historias finalizadas para futuras búsquedas.

4.5 Búsqueda de una historia de usuario

La búsqueda de una historia de usuario es importante debido a que en algunas ocasiones suelen compartir ciertas características (reglas de negocio, implementación, problemas, etc.) y en su desarrollo puede resultar muy útil referenciar a alguna historia que ya se haya realizado. TG posee un potente buscador de historias de usuario. La principal característica es que se apoya en la relación entre historias cuando las mismas son creadas. Si esta opción se utiliza adecuadamente, simplemente con navegar a través de las historias se obtendría rápidamente lo buscado. No obstante, no es la única forma de buscar historias, TG almacena las iteraciones (Sprints) finalizadas, esto permite acotar la búsqueda, visualizando únicamente una iteración en particular. Otra posibilidad, es buscar a partir de un requerimiento, en TG todas las historias nacen a partir de un requerimiento, TG mantiene esta relación permitiendo acotar la búsqueda a las historias derivadas de dicho requerimiento.

Contar con más de un proceso para realizar búsquedas de historias, añaden a TG una característica de agilidad para el desarrollador al momento de necesitar encontrar una historia de usuario en particular.

5 Conclusiones

TG Project Management se suma como una opción competitiva, al momento de seleccionar una herramienta para la gestión de proyectos de software amplios y complejos. TG se presenta como una herramienta de apoyo, que prioriza la agilidad y flexibilidad durante el desarrollo, en lugar de definir procesos largos y estructurados. TG permite gestionar y administrar todo lo relacionado al proyecto de software, controlar el desarrollo, gestionar requerimientos, historias de usuario, equipos, actividades, etc.

Aunque los proyectos de software actuales aumentan su complejidad a medida que avanzan en su ciclo de vida, TG se encuentra diseñado para adaptarse fácilmente al impacto del incremento de la complejidad.

La principal característica y distinción de TG (con respecto a Trello y Jira) es permitir la gestión del proyecto desde su inicio hasta su finalización, sin tener que preocuparse por el momento del ciclo de vida en el que se encuentra, la cantidad de integrantes o la magnitud del mismo. TG se adapta al proyecto, no es necesario adaptar el proyecto a TG.

6 Trabajos Futuros

Como trabajo futuro, se propone ampliar las características de TG para beneficiar la agilidad y el avance del proyecto. Algunas de estas características son:

- Incorporar gráficos estadísticos para el progreso del proyecto e iteraciones.
- Definir estrategias para identificar, analizar y definir riesgos para el proyecto. Mejorando así los tiempos estimados y la eficiencia de la gestión.
- Incorporar análisis de costos.
- Incorporar estrategias para que el usuario desarrollador participe en diversas tareas.
- Vincular TG con una herramienta de correo como por ejemplo Gmail.

7 Referencias

1. Pablo Lledó y Gustavo Rivarola. (2007). *Gestión de proyectos / Pablo Lledó y Gustavo*. Buenos Aires: Prentice Hall - Pearson Education.
2. Schaul, S. F. (2011). EL “DESARROLLO DE SOFTWARE” COMO “INGENIERÍA DE SOFTWARE”. *Ing. USBMed*, 4.

3. Alexander Menzinsky, Gertrudis López, Juan Palacio. (2016). *Scrum Manager*. Iubaris Info 4 Media SL.
4. Schwaber, K. (1995). Scrum. *SCRUM Development Process*, 23.
5. Lledó, P. &. (2007). *Gestión de Proyectos - Cómo dirigir proyectos exitosos, coordinar los recursos humanos y administrar los riesgos*. California: Pearson. Prentice Hall.
6. Almunia, P. (22 de Febrero de 2016). *itmplatform*. Recuperado el 25 de Junio de 2017, de itmplatform: <http://www.itmplatform.com>
7. Pfleeger, S. L. (2002). *Ingeniería de software teoría y práctica*. Buenos Aires: Pearson Educación.
8. Figuerola, N. (2015). Cómo seleccionar una Metodología de Project Management. *Project Management*, 7
9. Pressman, P. R. (2003). *Ingeniería del software - Un enfoque práctico*. Mexico: Mc Graw Hill.
10. Highsmith, M. F. (2001). The Agile Manifesto. *The Agile Manifesto*, 7.