



Trabajo práctico nº14:

# Comparación de Enfoques para Gestión del Desarrollo de Productos de Software

GRUPO 6

INTEGRANTES:

Apellido y Nombre	Legajo	Mail
Amaranto Vilanova, Facundo	78659	facuvilanova98@gmail.com
Bonzano, Evangelina	80855	evabonzano@gmail.com
Carranza Duobaitis, Guadalupe	78698	guadi.carranza@gmail.com
Cornejo Ruata, Ángeles	78502	angeles.cornejo.ruata@gmail.com
Pérez, Candela	78293	candemperez13@gmail.com
Pérez, Leonardo	81295	fenixi1234567890@gmail.com
Petroni Olmos, Antonella	78340	antopetrolmos@gmail.com
Spalla, Gaspar	72010	gasparspalla@gmail.com

Características	 TRADICIONAL	 SCRUM	 KANBAN
Tipo de proceso	Proceso definido	Proceso empírico con enfoque Agile	Proceso empírico con enfoque Lean
Aplicación	Gestión de proyectos	Gestión de proyectos	Gestión de tickets/ cambios
Alcance del producto	Fijo (Variable Tiempo y Recursos)	Variable (Fijo Tiempo y Recursos)	Variable (Variable Tiempo Y Fijo Recursos <b>(aunque se puede variar en donde se van a utilizar)</b> )
Tamaño de proyecto	Proyectos de cualquier tamaño	Orientada a proyectos pequeños	No orientada a proyectos
Toma de decisiones	Lider de proyecto	Product Owner	Delegar decisiones y responsabilidades del producto en desarrollo al nivel más bajo posible. (auto asignación de decisiones)
Lineamientos	Plan de proyecto detallado desde el principio	Lo definido en la sprint planning e historias de usuario, que se van adaptando y refactorizando	Proceso de cambio evolutivo
Duración	Basado en lo que establezca el plan de proyecto	Basado en lo que se planifique, dividido en iteraciones de 2 a 4 semanas (sprints)	Basado en la necesidad
Finalización	Al finalizar el proyecto	Al finalizar el proyecto	Es para uso continuo
Entrega de valor	En hitos definidos	Al finalizar el Sprint (Tempranas y frecuentes)	Entregas continuas (Tempranas y frecuentes)
Aceptación	Cumple con el contrato establecido	Satisface al Product Owner	Satisface al cliente
Asignación de tareas	Las asigna el líder de proyecto	Tareas autoasignadas por parte del equipo	Tareas autoasignadas por parte del equipo
Equipo de trabajo	Roles fijos	Individuos multidisciplinares y autoorganizados	Distribución de responsabilidades, no de roles
Relación con el cliente	Poca interacción	Incorporado como parte del equipo	En reuniones periódicas
Limitaciones	Fecha de entrega	El sprint	El WIP (Work In Progress)
Reuniones	Bajo demanda	Planeadas, se consideran ceremonias	Bajo demanda y opcionales
Métricas	Esfuerzo Tiempo Costos Riesgos	Velocidad Capacidad RTF	Lead time Cycle time Touch time Eficiencia Proceso
Documentación	Detallada y rigurosa (ERS)	Solo documenta lo que genere valor al cliente	Solo documenta lo que genera valor y no sea desperdicio
Contenedor de requerimientos	ERS (Especificación de Requerimientos)	Product backlog	Flujo de trabajo
Requerimientos	Surgen de la ERS (Especificación de Requerimientos)	Surgen de reuniones cara a cara con el cliente	Surgen de optimizar el rendimiento del flujo
Priorización de requerimientos	No prioriza	Prioriza los que generan valor al cliente / negocio	Prioriza los que generan valor al cliente / negocio

Detalle	En la ERS (Especificación de Requerimientos)	En la conversación	En las Políticas expresas y Signal Cards.
Actitud frente al cambio y gestión del mismo	Resistencia/ Temor. No hay retroalimentación de problemas	Adaptación. El cambio forma parte del proceso. Se desarrolla en base a las necesidades.	Adaptación. El cambio forma parte del proceso
Costo de los cambios	Alto	Bajo	Bajo
¿Qué se estima?	Esfuerzo (horas calendario) Tamaño del producto Tiempo Costo	Tamaño (complejidad, esfuerzo e incertidumbre): Medido en SP (Es relativa) - Utiliza: serie de fibonacci, $2^n$ , talles de remera, tamaño por números	Define el WIP para saber cuántas tareas pueden estar en Progreso a la vez.
Retroalimentación	No existe	Se toma al final de cada Retrospective	Se busca feedback constante para continuar el proceso de mejora
Detección de errores	En etapas tardías	En etapas tempranas	En etapas tempranas
Enfoque del testing	El objetivo es encontrar defectos (Destruir el sw)	Busca prevenir defectos (Construir el mejor sw)	Busca prevenir defectos

## Conclusiones:

En esta dicotomía entre la gestión tradicional y la gestión ágil de proyectos, en los últimos años fue tomando fuerza la metodología ágil frente a la tradicional. Cuando se involucra al equipo completo y a los clientes en procesos importantes, el resultado final que se obtiene y la experiencia de trabajo para todos los involucrados es mucho más gratificante que el enfoque tradicional.

Las Metodologías Ágiles se orientan a la obtención rápida de resultados y a la satisfacción del cliente, se prescinde de la excesiva burocracia, se tiende a la simplicidad y a la mejora continua, tanto del producto del proyecto como del propio proceso de gestión del proyecto de software con entregas tempranas, continuas y que otorguen valor.

Otros beneficios de la metodología ágil son: la alta calidad (dado que está embebida dentro de todo el proyecto), una mayor productividad, un mejor valor comercial, menos costos y velocidades más rápidas de comercialización. Además, en los tiempos que corren, las empresas están frente a cambios constantes, seguir una metodología ágil sería una muy buena opción a tener en cuenta.

La gestión tradicional de proyectos es una metodología establecida donde los proyectos se ejecutan en un ciclo de vida secuencial que incluye las siguientes etapas: iniciación, planificación, ejecución, monitoreo y cierre.

Los proyectos, en su totalidad, son planeados de antemano, sin posibilidad de realizar cambios en los requerimientos/necesidades. Este enfoque asume que tanto tiempo como costos, son variables y que, por el contrario, los requisitos son fijos. Esta es entonces la razón por la cual la gestión tradicional de proyectos se enfrenta o suele enfrentarse problemas en cuestiones de presupuesto y plazos.

En contraposición, la gestión de proyectos Ágil prioriza al equipo de trabajo, la colaboración con el cliente, la comunicación y la flexibilidad. Este es un proceso iterativo que se enfoca en el feedback de los clientes y las versiones continuas en cada iteración de un proyecto de desarrollo de software.

Lo esencial detrás del desarrollo de software ágil, es que hace hincapié en los cambios evolutivos y en el esfuerzo colaborativo para obtener resultados acordes a lo que el cliente necesita, en lugar de un proceso predefinido. Se dedica menos tiempo a la planificación y priorización inicial, ya que Agile es más flexible en términos de cambios y desarrollos en las especificaciones.

Scrum y Kanban, son dos de los marcos de trabajo Agile más utilizados, muy conocidos por fomentar la toma de decisiones y evitar el consumo de tiempo en variables que están destinadas a cambiar. Estos métodos hacen hincapié en la satisfacción de los clientes y usan los equipos disponibles para acelerar el desarrollo de software en cada etapa. Pero a diferencia de Scrum, Kanban es una manera de aplicar la filosofía Lean, que apunta a un trabajo con flujo continuo, más parecido a un proceso donde se tienen tareas para realizar vinculadas a la gestión de requerimientos. Una de sus características más importantes es que siempre se parte de un proceso que ya existe, como base para aplicar sus principios y mejorar lo que se tiene. Siendo destacada la diferencia entre Scrum que posee reglas prescriptivas de aplicación y trabajo, mientras que Kanban es más libre, con menos reglas y apunta a la gestión del cambio y no una metodología de administración de proyecto.

Por último, podemos afirmar que no hay un enfoque que sea mejor que otro, tenemos que tener en cuenta cuál es el tipo de proyecto se va a llevar a cabo para decidir, en base a las necesidades y los recursos disponibles, qué enfoque deberíamos aplicar para lograr la satisfacción del cliente y entregarle el mejor producto posible. Es decir, lo primero que debemos hacer antes de empezar a trabajar es analizar y decidir qué metodología vamos a utilizar basándonos en el contexto que tenemos y queremos tener.

## Bibliografía

- Hito Master S.L. (s. f.). ágiles vs tradicionales. MDAP. Recuperado 11 de junio de 2021, de <https://uv-mdap.com/programa-desarrollado/bloque-iv-metodologias-agiles/metodologias-agiles-vs-tradicionales/>
- Feda, E. D. N. (2019, 20 mayo). *GESTIÓN ÁGIL vs GESTIÓN TRADICIONAL DE PROYECTOS ¿CÓMO ELEGIR?* - Escuela de Negocios FEDA. Escuela de negocios FEDA. <https://www.escueladenegociosfeda.com/blog/50-la-huella-de-nuestros-docentes/471-gestion-agil-vs-gestion-tradicional-de-proyectos-como-elegir#:~:text=En%20la%20metodolog%C3%ADa%20%C3%A1gil%2C%20cada.comparte%20la%20propiedad%20del%20proyecto.&text=En%20el%20enfoque%20tradicional%2C%20cada,del%20tiempo%20y%20presupuesto%20estimados>
- Apuntes de Aprendiz (10 de julio,2013) - *Estimación y Medición en Kanban* <https://jesuslc.com/2013/07/10/546/#:~:text=Ya%20hemos%20visto%20que%20el,en%20el%20flujo%20de%20trabajo>
- Manifiesto Ágil <http://agilemanifesto.org/iso/es/>
- Sommerville, Ian - INGENIERÍA DE SOFTWARE - Novena Edición (Editorial Addison-Wesley Año 2011). Capítulo 2, 3 ,4, 22, 23