

Universidad Tecnológica Nacional

Facultad Regional Córdoba

Ingeniería en Sistemas de Información

Cátedra: Ingeniería de Software

Trabajo Práctico N°1: Dinámica de Manifiesto Ágil

Curso: 4K1

GRUPO: 6

Nombre	Apellido	Legajo	Correo Electrónico
Facundo	Paz Fessia	78579	facupazfessia@gmail.com
Maximiliano	Saleh	78798	maxisaleh@outlook.com
Santiago	Landa Valle	78637	santi.land4@gmail.com
Leonardo	Pozzo	57453	leo.em09@gmail.com
Rodrigo	Tosco	69404	rodrigotosco95@gmail.com
Joaquin	Giron Pignol	60862	girjoes@gmail.com

Docentes:

- Meles, Silvia Judith (Titular)
- Boiero Rovera, Gerardo Javier (JTP)
- Crespo, María Mickaela (Ayudante 1ra)

Fecha de Presentación: Martes 10/08/2021



Unidad	Nro. 2: Gestión Lean-Ágil de Productos de Software
Consigna	Asocie cada uno de los valores del manifiesto ágil con el o los principios ágiles que considere que están relacionados
Objetivo	Comprender valores y principios del Manifiesto Ágil, para aplicarlos a ejemplos y prácticas concretas y cotidianas de trabajo.
Propósito	Familiarizarse con los valores y principios del Manifiesto Ágil.
Entradas	Conceptos teóricos sobre el tema, desarrollados en clase. Manifiesto Ágil. Bibliografía referenciada sobre el tema.
Salida	Comparación de los valores del manifiesto ágil con los principios del manifiesto ágil, y justificación asociada.
Instrucciones	1. Cada grupo discute individualmente qué valor del manifiesto ágil está asociado con qué principio ágil. 2. Cada grupo presenta al resto del curso sus conclusiones, justificando las relaciones presentadas.



Desarrollo

- Individuos e interacciones sobre procesos y herramientas

Principios:

- *Los proyectos se desarrollan en torno a individuos motivados. Hay que darles el entorno y el apoyo que necesitan, y confiarles la ejecución del trabajo.*
Se prioriza que los individuos que participan en el proyecto (tanto clientes como el equipo de trabajo), se encuentren motivados para el desarrollo del mismo. El valor relacionado habla de que son más importantes estos individuos y la interacción entre ellos, antes que apegarse a un proceso rígido o una herramienta determinada, lo cual pueda afectar la motivación de cada uno, según las preferencias o métodos de trabajo personales.
- *Las mejores arquitecturas, diseños y requerimientos emergen de equipos autoorganizados*
Cuando un equipo está bien definido, organizado, posee tareas claras y hay un marco de trabajo establecido en un entorno comunicativo, estos generan mayor aporte, y de mejor calidad. Además de existir una retroalimentación constante entre el mismo equipo.
- *Los responsables de negocio y los desarrolladores trabajamos juntos de forma cotidiana durante todo el proyecto.*
Esto es una manera eficiente de atender a los requerimientos cambiantes, de manera de poder obtener el feedback necesario para poder seguir con los próximos incrementos siempre de acuerdo con la opinión y necesidades de los clientes.
Así el desarrollo del proyecto estará basado en una comunicación continua y constante entre sus participantes.
- *El método más eficiente y efectivo de comunicar información al equipo de desarrollo y entre sus miembros es la conversación cara a cara.*
Este principio define cómo es conveniente que se den las **interacciones** entre las personas que desarrollan el software, en relación con la comunicación de información, a fin de que ésta sea correctamente transmitida y entendida por todos.



- Software funcionando sobre documentación extensiva
 - *Nuestra mayor prioridad es satisfacer al cliente mediante la entrega temprana y continua de software con valor.*

Ya que priorizamos el desarrollo del software para tener entregas en periodos más cortos refinando los requisitos con el cliente evitando documentación innecesaria que sería en vano si los requisitos cambian, o bien evitando que la acción de documentar conlleve un tiempo que pueda retrasar la entrega de una nueva versión del software al cliente.
 - *Entregamos software funcional frecuentemente, entre dos semanas y dos meses, con preferencia al periodo de tiempo más corto posible.*

Documentar las decisiones de forma exhaustiva puede llevar a ocupar mucho tiempo del tiempo de desarrollo produciendo un aumento de la probabilidad de que no se pueda entregar un producto funcionando a tiempo. Por ende, el trabajo debe estar enfocado en poder presentar este software funcional, con todos los cambios necesarios identificados
 - *El software funcionando es la medida principal de progreso.*

Un producto terminado al 99% es un producto que está 0% completo o hecho, si queremos ser capaces de desarrollar en un entorno ágil debemos poder proveer al cliente con software funcional de manera continua, para que éste evalúe nuestro progreso, nos provea de feedback y permita cambiar los requerimientos según sus necesidades que es en definitiva quién lo usará, documentar de forma exhaustiva sólo retrasara el proceso de desarrollo en cada sprint.
 - *La atención continua a la excelencia técnica y al buen diseño mejora la agilidad.*

Con la aplicación de patrones para la buena organización y el reuso ante situaciones ya conocidas podemos agilizar el desarrollo, y además contribuyen a la calidad del software a desarrollar y entregar
 - *La simplicidad, o el arte de maximizar la cantidad de trabajo no realizado, es esencial.*

Este principio refiere al hecho de que más allá de las necesidades de documentación, arquitectura y refinamiento, es importante el hecho de concretar un producto sólido, el cual cumpla con los requisitos del cliente, sin caer en la excesiva complejidad o el abarcar más terreno del necesario para el funcionamiento del mismo.
El cliente aprecia y valora sobre todas las cosas que el software abarque los requisitos solicitados y brinde soluciones simples y efectivas a sus problemas.
 - *El método más eficiente y efectivo de comunicar información al equipo de desarrollo y entre sus miembros es la conversación cara a cara.*

Ya que podemos aclarar y tomar decisiones en grupo manteniendo sincronía sobre lo que se debe hacer en vez de alternativas como comentarios en algún artefacto. Además evita malentendidos y posibles errores en el mismo producto.



- Colaboración con el cliente sobre negociación contractual
 - *Nuestra mayor prioridad es satisfacer al cliente mediante la entrega temprana y continua de software con valor.*
La interacción continua con el cliente nos permite identificar requisitos y cambios necesarios, con entregas incrementales aportamos componentes que ya pueden ser utilizados que pueden servir para feedback.
- Respuesta ante el cambio sobre seguir un plan
 - *Aceptamos que los requisitos cambien, incluso en etapas tardías del desarrollo*
Ya que las necesidades del cliente van cambiando y aparecen medidas que notan que necesitan, el mayor costo en caso de error es por la identificación errónea de los requisitos.
Por otro lado, este cambio de planes ante un nuevo requisito, no será rechazado por el equipo, sino que será asimilado por encima del plan anterior a fin de poder dar una respuesta concreta a la necesidad del cliente (dirigido por los casos de uso)
 - *Los procesos Ágiles promueven el desarrollo sostenible. Los promotores, desarrolladores y usuarios debemos ser capaces de mantener un ritmo constante de forma indefinida.*
En las metodologías ágiles se trata de establecer un ritmo de trabajo constante a lo largo de todo el proceso. En función de la velocidad obtenida en cada iteración, se establecen los objetivos para la siguiente tratando de llevar un ritmo constante desde el inicio. Así, la planificación de las actividades de cada iteración se modifica en base al ritmo de trabajo, que a su vez cambia en cada iteración según se vea en la experiencia del grupo, por lo que el plan no se mantiene fijo
 - *A intervalos regulares el equipo reflexiona sobre cómo ser más efectivo para a continuación ajustar y perfeccionar su comportamiento en consecuencia.*
Este método permite evitar desperdiciar tiempo en el desarrollo de cosas que realmente no cumplen con lo solicitado o requerido.
A su vez, este constante ajuste y perfección del comportamiento permite que el equipo esté listo ante eventuales cambios de requerimientos y pueda adaptar el plan a estos.



Correcciones:

22 Valor palabra clave. Tiene que haber un producto de valor