

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.

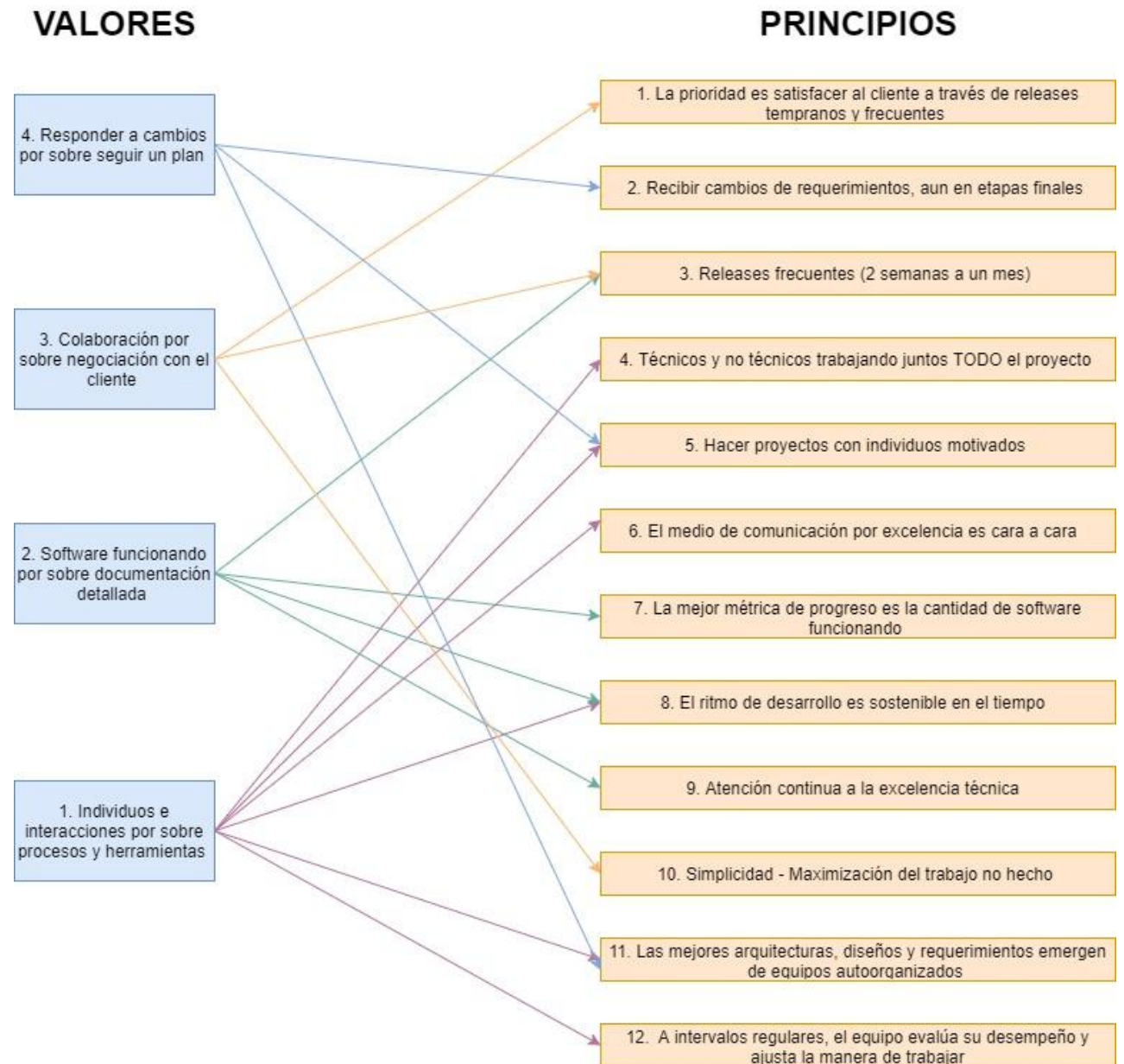
Instrucciones:

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.

INTEGRANTES GRUPO 6:

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

Se presenta la relación gráfica entre los valores y los principios:



A continuación, se detalla cada relación.

1. INDIVIDUOS E INTERACCIONES POR SOBRE PROCESOS Y HERRAMIENTAS

Se relaciona con los siguientes principios:

4 - Técnicos y no técnicos trabajando juntos TODO el proyecto:

El proyecto, desde que comienza hasta que finaliza, debe trabajarse en equipo. Todos los integrantes deben brindar sus conocimientos para que el proyecto crezca en su conjunto por lo que es importante que cada integrante tenga sus propias responsabilidades que ayuden a completar las tareas planteadas.

Son virtuosas las interacciones entre desarrolladores, testers, analistas y otros en cualquier etapa del proyecto ya que cada uno podrá aportar una mirada diferente en los planteos de problemas y las propuestas de soluciones.

5 - Hacer proyectos con individuos motivados:

Es importante la interacción entre los individuos frente a los procesos y herramientas que se utilicen. Los individuos deben estar motivados ante las tareas del proyecto a realizar, debido a que así trabajan mejor, cumpliendo con mayor rigurosidad sus tareas al sentirse incluidos y en confianza con el resto del equipo.

De esta manera el proyecto se llevará a cabo de forma más eficiente consiguiendo un buen resultado final, facilitando además, la comunicación y resolución de situaciones problemáticas que se vayan presentando y/o cambios que se deban ir realizando.

6 - El medio de comunicación por excelencia es cara a cara:

Si bien los procesos y las herramientas pueden definir una forma de trabajo, a veces estos no logran arribar a la mejor solución y la falta de comunicación sobre esta situación puede generar conflictos. Estos se pueden resolver generando una incidencia o, de manera más rápida y eficiente, entablando una conversación con los interesados, para tomar medidas lo antes posible y encaminar de la forma que se quiere al producto.

Se valora mucho la relación existente entre la persona que va a colaborar por parte de la empresa y el equipo de desarrollo, coordinando cortas reuniones de forma frecuente para ir validando si lo que se está haciendo es lo que el cliente quiere, ver su lenguaje corporal y establecer juntos las próximas funcionalidades o cambios a establecer.

8 - El ritmo de desarrollo es sostenible en el tiempo:

Si bien en el desarrollo ágil no se priorizan los procesos y herramientas, debe existir un acuerdo entre los miembros del equipo para que haya buena comunicación y dos integrantes no hagan dos veces lo mismo o queden integrantes sin tareas asignadas.

También, es importante la interacción entre los individuos para identificar fortalezas y debilidades de cada uno, de esta forma los especialistas en ciertos aspectos podrán enfocarse en eso y podremos seguir entregando el producto en tiempo y forma, como se pactó. Sin la previa comunicación, la asignación de tareas podría no ser la óptima, generando retrasos o inconvenientes en el sprint.

11 - Las mejores arquitecturas, diseños y requerimientos emergen de equipos autoorganizados:

En el desarrollo ágil se enaltecen a los individuos y su interacción por sobre los procesos, en este caso la capacidad de autogestión y el conocimiento de los integrantes del equipo son más importantes que los procesos tradicionales.

La mejor manera de planificar un proyecto es con roles definidos de acuerdo a las habilidades que posea cada uno, para así obtener mejores resultados. La confianza entre individuos es la base de los equipos autoorganizados, ya que no es necesaria una estructura vertical que esté regulando a cada uno, la interacción entre los individuos es la base de esta confianza.

12 - A intervalos regulares, el equipo evalúa su desempeño y ajusta la manera de trabajar:

Al momento de trabajar hay cosas que pueden salir bien y otras mal (o muy mal), esto debe llevarnos a un nivel de conocimiento mayor, aprendiendo de los errores y buscando la solución en equipo de lo que sucedió, y lo más importante, intentar que no vuelva a pasar.

Realizar reuniones frecuentes del equipo hace que no se acumulen errores en etapas posteriores y se resuelvan de forma conjunta, por lo que la comunicación y colaboración dentro del mismo es fundamental.

2. SOFTWARE FUNCIONANDO POR SOBRE DOCUMENTACIÓN DETALLADA

Se relaciona con los siguientes principios:

3 - Releases frecuentes (2 semanas a un mes):

Se prioriza entregar al cliente software funcionando, aunque sea en menor cantidad de requerimientos cumplidos para mejorar la comunicación con el mismo y asegurarnos de estar respetando lo que él quiere.

Además, el cliente tiene un encuentro más cercano y tangible con el producto de software, de esta manera puede mejorar hasta su propio entendimiento de lo requerido al principio y no perdiendo tiempo en documentar excesivamente cosas que pueden sufrir muchos cambios.

7 - La mejor métrica de progreso es la cantidad de software funcionando:

El cliente sólo puede saber si lo que pidió se cumple cuando ve lo que el software hace en funcionamiento. Si esto no es así, se dificulta obtener una correcta retroalimentación y no podemos saber si entendimos los requerimientos correctamente. Además, el cliente desde el momento en que solicitó el software se encuentra ansioso y si no tiene cómo ver el avance del proyecto, cree que no está siendo escuchado o que no se está trabajando sobre lo que pidió.

La mejor manera de saber si estamos haciendo lo correcto es si el cliente está satisfecho con lo que le estamos mostrando, por lo que se deben establecer reuniones que permitan validarlo.

8 - El ritmo de desarrollo es sostenible en el tiempo

Los integrantes del equipo deben mantener la regularidad con la que realizan sus tareas para desarrollar el proyecto de manera que sea mantenible y razonable en relación al tiempo y su complejidad. Es importante saber priorizar los requerimientos, de forma tal que se pueda cumplir con los diferentes releases acordados con el cliente.

9 - Atención continua a la excelencia técnica:

A pesar de que el software sea desarrollado en un período corto de tiempo, no debe dejarse de lado la calidad que tiene que tener. Por lo tanto, en vez de dedicar tiempo a realizar documentación detallada, debemos invertir ese tiempo en mejorar lo que ya tenemos, como el diseño para lograr una buena experiencia de usuario y también minimizar errores y/o fallas que el software pueda llegar a tener.

3. COLABORACIÓN POR SOBRE NEGOCIACIÓN CON EL CLIENTE

Se relaciona con los siguientes principios:

1 - La prioridad es satisfacer al cliente a través de releases tempranos y frecuentes:

Entregando versiones acotadas en funcionalidades se busca constantemente la aprobación del cliente/colaborador para que a partir de ese ida y vuelta se pueda llegar al producto que realmente quiere. Colaborando con él, para satisfacer sus requerimientos y corroborar que el producto está bien encaminado o si es necesario un cambio o mejora.

3 - Releases frecuentes:

Se busca entregar software funcional al cliente, preferentemente en un periodo de tiempo lo más corto posible, para que de esa forma pueda conocer periódicamente las nuevas entregas y así evaluar si satisfacen los requerimientos que él solicita, con el objetivo de aumentar la probabilidad de detectar de manera temprana modificaciones, permitiéndole al equipo llevar a cabo esos cambios a una escala manejable y temprana del desarrollo.

10 - Simplicidad - Maximización del trabajo no hecho:

Se busca no desperdiciar el tiempo en trabajo que no es importante y/o prioritario para este momento sino enfocar todos los recursos en lo que aquí nos compete.

Buscamos maximizar el trabajo presentado al cliente, es decir, lo que se compromete en el principio del release, se busca entregar con la mayor calidad posible. Además, hay que prestarle atención al trabajo restante en un proyecto, ya que una métrica del avance del producto brindará mucha mayor información para estimaciones y planificaciones si el cliente nos informa qué falta. Por ejemplo, para la gestión de un sprint se suele utilizar un gráfico Burndown, dándole mayor importancia al trabajo restante y no al realizado.

4. RESPONDER A CAMBIOS POR SOBRE SEGUIR UN PLAN

Se relaciona con los siguientes principios:

2 - Recibir cambios de requerimientos, aun en etapas finales:

Es importante tener en cuenta que los requerimientos que el cliente nos plantea al comenzar con el proyecto pueden variar constantemente, y esos cambios pueden ir apareciendo al momento de ir entregando software funcionando, por lo que debemos estar conscientes del hecho y estar preparados para esos cambios.

Los requerimientos se listan y se van atacando (analizando y depurando) según una priorización que le hayamos dado. El análisis se hace en ese momento porque pueden ser tan cambiantes que de esta manera no perdemos tiempo en requerimientos que pueden modificarse completamente o directamente ser eliminados de la lista.

5 - Hacer proyectos con individuos motivados:

Un factor clave en el desarrollo de un producto es contar con un equipo motivado. Poder generar en el equipo un ambiente donde se fomente la motivación a trabajar en conjunto, que busque jerarquizar las opiniones de los integrantes del equipo e involucrarlos en la responsabilidad de poder decidir el camino de acción es fundamental para enfrentarse a situaciones de cambio constante. Deberá tenerse el objetivo de contar con un grupo autogestionado que sea flexible para ser capaz de adaptarse a los cambios.

11 - Las mejores arquitecturas, diseños y requerimientos emergen de equipos autoorganizados

Se busca que el equipo esté dirigido y organizado por sus propios miembros, a los cuales se le dará las responsabilidades no sólo de dirigir y organizarse, sino también de gestionar y adaptarse a los cambios para corregir y mejorar sus actividades y así poder alcanzar los objetivos planteados. Si le damos las herramientas a las personas para decidir y manejarse por su cuenta resultará mejor para el proyecto en general.