

## **TRABAJO PRÁCTICO N°1**

### **Dinámica de Manifiesto Ágil y Lean**

**CARRERA:** Ingeniería en Sistemas de Información

**ASIGNATURA:** Ingeniería de Software

**CICLO LECTIVO:** 2020

**CURSO:** 4K1

**DOCENTES:**

- Meles, Silvia Judith
- Robles, Joaquín Leonel
- Crespo, María Mickaela

**GRUPO N°6:**

- Barbadillo Incerti, Oriana (72762)
- Dávila, Rodrigo Sebastian (76144)
- D'Uva, Micaela (74300)
- Molina, Matias Franco (75477)
- Santesso, Florencia (72257)
- Zanini, Gaspar (76639)

**FECHA DE ENTREGA:** 04/08/2020

# Principios relacionados a cada valor

## Individuos e interacciones por sobre procesos y herramientas:

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

El objetivo de este principio es obtener un equipo alineado, con objetivos claros y con un conocimiento común del propósito con el que se realiza el trabajo pudiendo conseguir un equipo que sea adaptable y de esta forma utilizar los procesos como guía y no depender puramente de ellos ya que no siempre se van a poder aplicar en todos los entornos de trabajo.

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

Si un equipo es autoorganizado genera un ambiente de comunicación, respeto, compromiso, eficacia y transparencia, y aunque los procesos y herramientas pasen a un segundo plano con respecto a las interacciones entre individuos, el equipo de desarrollo será capaz de generar y llevar a cabo buenas arquitecturas, diseños y requerimientos si tienen la actitud adecuada. No se producen ni se llegan a buenos resultados si el equipo no está motivado.

- ***Hacer proyectos con individuos motivados:***

Permite que cada uno de ellos sienta que posee un lugar significativo en el equipo de trabajo y a su vez perciba que tiene un compromiso con ese grupo. El conjunto de personas que trabaja motivadamente debería generar un ambiente confortable y eficiente, lo cual facilita la comunicación entre las partes y un crecimiento escalable como grupo de trabajo

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

La interacción presencial entre personas permite percibir a través del lenguaje corporal información que es imposible transmitir en palabras. Así podemos identificar mediante una correcta observación, si lo que una de las partes expresa verbalmente es realmente cierta, o si es necesario aplicar otras técnicas para adquirir información más precisa.

- ***Simplicidad - Maximización del trabajo no hecho:***

No se utilizan herramientas ni procesos complicados para gestionar el proyecto, el mismo se gestiona y se entrega de una manera simple

## Software funcionando por sobre documentación detallada:

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

Este valor se enfoca en que a través de releases tempranos nuestro cliente puede ir teniendo una visión anticipada del producto la que le permitirá determinar si el mismo

cumple con sus expectativas o si es necesario sugerir algún cambio. Esta interacción con el cliente supone que al momento de entregarle el producto final, el nivel de aceptación del mismo será alto, ya que el cliente interactúa iterativamente en el proceso de creación.

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

Lo importante de este principio es la release donde su fin es entregar una versión del software al cliente con una nueva funcionalidad y con ello obtener un feedback o una devolución por parte del cliente generando nuevas ideas para encarar funcionalidades más complejas. Por lo tanto estas entregas se deben realizar en periodos cortos de tiempo (entre 2 semanas y un mes), al ser tan dinámico el desarrollo y la entrega de software funcional se deja en un segundo plano el desarrollo de la documentación con todos los requisitos detallados ya que aporta menos valor que un producto funcional. Esto no quiere decir que no sea importante la documentación.

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

Si bien la documentación es importante, por lo cual se debe realizar, de nada serviría tener una documentación de un software que no funciona o que no cumple con lo que el cliente nos ha solicitado. Se debería documentar lo mínimo y necesario y aspirar a las funcionalidades del software principalmente, ya que es lo que principalmente le interesa al cliente.

- ***Simplicidad - Maximización del trabajo no hecho:***

Parte de evitar realizar más trabajo del necesario, pasa por reducir la cantidad de documentación a producir.

- ***Atención continua a la excelencia técnica:***

A pesar de que se realicen entregas de software funcional de manera frecuente, y de que se opte por documentar poco, todo desarrollo debe ser realizado con un muy alto nivel de calidad. Si se va a desarrollar documentando poco, entonces es necesario que el desarrollo tenga una alta calidad técnica.

### **Colaboración por sobre negociación con el cliente:**

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

Este principio, podemos relacionarlo con más de un valor, por un lado debemos considerar que, se debe tener una comunicación constante con el cliente, durante el desarrollo, para debatir acerca de si el software que se está desarrollando concuerda con lo que desea. A su vez, al trabajar con personal de varias áreas relacionadas, se debe lograr un ritmo de trabajo duradero a pesar de que el proyecto tenga una fecha tentativa de finalización.

- ***Hacer proyectos con individuos motivados:***

La motivación no solamente se refiere a quienes construirán el software, sino que también implica al cliente o dueños del negocio. Si quien encarga el proyecto no está interesado en el mismo, y no se involucra ni participa en el desarrollo, y simplemente espera que el equipo realice el trabajo sin comprometerse, entonces las metodologías ágiles no serían la forma adecuada de organizar ese proyecto, ya que el cliente no sería compatible con las mismas.

### **Responder a cambios por sobre seguir un plan:**

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

Hace referencia a que antes de desarrollar productos que tienen requerimientos inestables debido al cambio constante y las evoluciones continuas que hay en el medio, es preferible que los planes se adapten a la evolución de los requerimientos en vez de seguir un plan original el cual puede no ser el mismo que el planteado al inicio.

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

Hace referencia o lo relacionamos con este valor ya que nos basamos de responder a cambios por sobre seguir un plan, y el principio nos hace alusión a que deberíamos cada cierto tiempo, autoevaluarnos y/o tomar una propia retroalimentación como equipo para reajustar todas esas cosas o elementos que podemos hacer mejor para ser más efectivos, es decir, vamos a buscar que haya una mejora continua en nuestro proceso de desarrollo de software. Además, ya el hecho de pararse a reflexionar y hacerlo en equipo nos va a ayudar a reducir el riesgo significativamente.