

INGENIERÍA DE SOFTWARE

TP N°1

Dinámica de Manifiesto Ágil

Profesores:

- Ing. Judith Meles
- Ing. Massano, María Cecilia
- Ing. Joaquin Leonel Robles

Alumnos:

➤ Bordón Eliana	71336
➤ Cruz Karen	61539
➤ Konicoff Sebastián	67335
➤ López Ríos María Sol	71871
➤ Palacio Priscila	63018
➤ Salaberri Marcelo	57659

Grupo N° 7

Curso: 4K2

Año: 2020

PRÁCTICO 1 - Dinámica de Manifiesto Ágil y Lean (Evaluable)

Unidad Nro. 2: Gestión Lean-Ágil de Productos de Software

Consigna:

Asocie a cada valor del manifiesto ágil el o los principios ágiles que considere que están asociados.

Objetivo:

Comprender valores y principios del Manifiesto Ágil y los principios Lean, expuestos en clase, para aplicarlos a ejemplos concretos de gestión de proyectos de software.

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.

Manifiesto Ágil

VALORES

- Individuos e interacciones por sobre procesos y herramientas
- Software funcionando por sobre documentación detallada
- Colaboración por sobre negociación con el cliente
- Responder a cambios por sobre seguir un plan

PRINCIPIOS

1. La prioridad es satisfacer al cliente a través de releases tempranos y frecuentes
2. Recibir cambios de requerimientos, aun en etapas finales
3. Releases frecuentes (2 semanas a un mes)
4. Técnicos y no técnicos trabajando juntos TODO el proyecto
5. Hacer proyectos con individuos motivados
6. El medio de comunicación por excelencia es cara a cara
7. La mejor métrica de progreso es la cantidad de software funcionando
8. El ritmo de desarrollo es sostenible en el tiempo
9. Atención continua a la excelencia técnica
10. Simplicidad - Maximización del trabajo no hecho
11. Las mejores arquitecturas, diseños y requerimientos emergen de equipos autoorganiza

DESARROLLO

VALORES	PRINCIPIOS	JUSTIFICACIÓN
Individuos e interacciones por sobre procesos y herramientas	4- Técnicos y no técnicos trabajando juntos TODO el proyecto.	4- Se le da una gran importancia a las personas, considerando que somos un solo equipo sinérgico y que tenemos un mismo objetivo.
	5- Hacer proyectos con individuos motivados.	5- Si los individuos se mantienen contentos y cómodos, serán más productivos, esto se logra con autonomía, confianza, etc.
	6- El medio de comunicación por excelencia es el cara a cara.	6- Los equipos trabajando juntos en el mismo espacio y tiempo permiten resolver los conflictos y dificultades más rápidamente, además se logra un mejor entendimiento con el cliente.
	8- El ritmo de desarrollo es sostenible en el tiempo.	8- Los individuos al estar motivados permiten que sea más ágil el desarrollo y que se mantenga un mismo ritmo o nivel.
	11- Las mejores arquitecturas, diseños y requerimientos emergen de equipos autoorganizados.	11- Implica confiar en que la gente y el equipo se pueden autoorganizar, brindándoles autonomía.
	12- A intervalos regulares, el equipo evalúa su desempeño y ajusta la manera de trabajar.	12- El equipo es el que cada cierto tiempo tiene que evaluar qué es lo que se hizo, ver en qué se tiene que concentrar y mejorar.

<p>Software funcionando por sobre documentación detallada</p>	<p>1- La prioridad es satisfacer al cliente a través de releases tempranos y frecuentes.</p> <p>3- Releases frecuentes (2 semanas a un mes).</p> <p>7- La mejor métrica de progreso es la cantidad de software funcionando.</p> <p>10- Simplicidad - Maximización del trabajo no hecho.</p> <p>8- El ritmo de desarrollo es sostenible en el tiempo.</p> <p>9- Atención continua a la excelencia técnica.</p>	<p>1 y 3- Con releases tempranos nos aseguramos de validar que el sw esté funcionando o si hay detalles para pulir antes de que se presente algún error.</p> <p>7- Es la mejor forma de controlar que estamos avanzando correctamente con el proyecto, viendo cuantas características tenemos terminadas.</p> <p>10- Evitamos que se genere trabajo innecesario que pueda llevar a consumir tiempo y no generar ningún impacto positivo en el proyecto.</p> <p>8- Se hacen release cada cierto tiempo lo que permite que sea más ágil el desarrollo y que se mantenga un mismo ritmo.</p> <p>9- Brindarle continuidad y atención al desarrollo del software garantiza que el mismo funcione de manera correcta y lleva a que logremos la excelencia.</p>
<p>Colaboración por sobre negociación con el cliente</p>	<p>3- Releases frecuentes (2 semanas a un mes).</p> <p>4- Técnicos y no técnicos trabajando juntos TODO el proyecto.</p>	<p>3- Al hacer release frecuentes puedo obtener un feedback temprano del cliente, permitiendo responder a los cambios rápidamente.</p> <p>4- Asocia justamente la colaboración entre el personal del proyecto técnico con conocimientos y el no técnico, haciendo referencia a los clientes que forman parte de manera importante. Todos forman un equipo con un objetivo</p>

	6- El medio de comunicación por excelencia es cara a cara.	en común. 6- Es más rápido y más conveniente comunicarse con el cliente cara a cara enfatizando la colaboración para mejores resultados antes que escribir largos documentos de requerimientos o contratos.
Responder a cambios por sobre seguir un plan	2- Recibir cambios de requerimientos, aun en etapas finales.	2- Se acepta el cambio en todo momento, siempre y cuando se evalúe que se pueda hacer o no.