**Universidad Tecnológica Nacional**

**Facultad Regional Córdoba**

**Ingeniería en Sistemas de Información**

Cátedra: Ingeniería de Software

PRÁCTICO 1 - Dinámica de Manifiesto Ágil y Lean

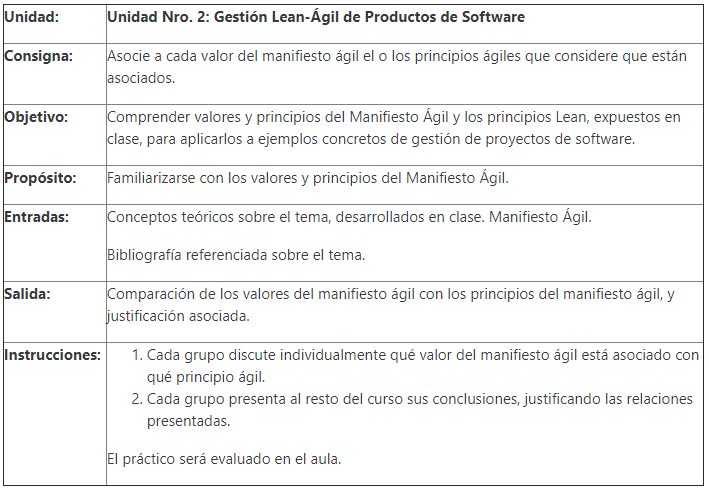
**Curso: 4K1**

Integrantes:

* Bossio, Mateo 75428
* Gallardo, Ignacio Alejandro 77491
* Garzón Funes, Álbaro 75652
* Lastra Levián, Francisco 72369
* Márquez, Darío 75566
* Morardo, Diego 75527

Grupo N° 5

Fecha de Entrega: 4/08/2020



**VALORES DEL MANIFIESTO ÁGIL**

* **Individuos e interacciones sobre procesos y herramienta:** Es importante entender que un desarrollo de software es llevado a cabo por personas, por lo cual es importante tener en cuenta sus emociones, necesidades, motivaciones, etc, para de esta forma poder influenciarlos positivamente y que desempeñen su trabajo de la mejor forma, algo que no solo es importante para la persona sino para todo el equipo, ya que de esta manera se crea un mejor clima laboral.
* **Software funcionando sobre documentación extensiva:** Se debe hacer hincapié en obtener funcionalidad para el cliente, si bien está claro que la documentación es importante en un proyecto de desarrollo de software, lo que se valora es un producto funcional, ya que además de ser de suma importancia para el cliente, también es clave para los equipos de desarrollo, ya que un producto funcional les aporta claridad gracias al feedback temprano que reciben por parte del cliente o usuarios, por lo que se debe encontrar un balance entre la documentación y la funcionalidad, documentando únicamente lo que es necesario, es decir lo que aporta valor al desarrollo.
* **Colaboración con el cliente sobre negociación contractual:** Este valor se centra en la colaboración más que en la negociación, lo que esto quiere decir es que es importante que tanto el equipo de desarrollo como el cliente trabajen en conjunto para lograr un producto que cumpla con los requerimientos de la forma más eficiente. Esto se logra conversando de antemano y planificando la forma en que equipo y cliente trabajarán juntos, otorgando mayor importancia de esta forma al diálogo continuo y a las decisiones que se toman en las diferentes etapas del proceso que a lo que se haya pactado a través de un contrato.
* **Respuesta ante el cambio sobre seguir un plan:** Actualmente las cosas cambian a una gran velocidad, volviéndose cada vez más complejas. La tecnología cambia todos los días, el mercado, los competidores y las necesidades están en constante cambio, y obviamente el software no se escapa de esto, o al menos no debería escaparse ya que un software que no cambia se vuelve obsoleto. En un desarrollo de software existe una gran incertidumbre y complejidad, lo cual nos lleva a tener que trabajar muchas veces aprendiendo sobre la marcha y adaptando nuestro producto a medida que se avanza y surgen cambios. Para esto es necesario ser ágiles y flexibles frente al cambio, ya sea un cambio de requerimientos, cambios dentro del equipo o de cualquier factor que afecte al desarrollo de software.

**PRINCIPIOS ÁGILES:**

1. Nuestra principal prioridad es satisfacer al cliente a través de la entrega temprana y continua de software de valor.
2. Son bienvenidos los requisitos cambiantes, incluso si llegan tarde al desarrollo. Los procesos ágiles se doblegan al cambio como ventaja competitiva para el cliente.
3. Entregar con frecuencia software que funcione, en periodos de un par de semanas hasta un par de meses, con preferencia en los períodos breves.
4. Las personas del negocio y los desarrolladores deben trabajar juntos de forma cotidiana a través del proyecto.
5. Construcción de proyectos en torno a individuos motivados, dándoles la oportunidad y el respaldo que necesitan y procurándoles confianza para que realicen la tarea.
6. La forma más eficiente y efectiva de comunicar información de ida y vuelta dentro de un equipo de desarrollo es mediante la conversación cara a cara.
7. El software que funciona es la principal medida del progreso.
8. Los procesos ágiles promueven el desarrollo sostenido. Los patrocinadores, desarrolladores y usuarios deben mantener un ritmo constante de forma indefinida.
9. La atención continua a la excelencia técnica enaltece la agilidad.
10. La simplicidad como arte de maximizar la cantidad de trabajo que se hace, es esencial.
11. Las mejores arquitecturas, requisitos y diseños emergen de equipos que se autoorganizan.
12. En intervalos regulares, el equipo reflexiona sobre la forma de ser más efectivo y ajusta su conducta en consecuencia.

**RELACIONES**

1. **Individuos e interacciones sobre procesos y herramientas:** **4,5,6, 8, 9,11,12**

* Relación con 4 (Trabajo en equipo interdisciplinario): los individuos del equipo deben comunicarse para poder trabajar de forma eficiente en un proyecto de desarrollo de software. Un grupo de trabajo tiene que poder compartir sus avances con los demás grupos con el fin de avanzar en el cumplimiento del objetivo del cliente.
* Relación con 5 (Desarrollo de proyectos en torno a individuos motivados): Este punto se basa en la idea de que los miembros de un grupo realizan su trabajo correctamente si se encuentran motivados para hacerlo, para lograr esto las distintas personas que integran el proyecto deben sentirse respaldadas por el resto del equipo y por los líderes del proyecto.
* Relación con 6 (Conversación cara a cara): En este contexto se hace referencia a la comunicación entre los miembros del equipo y su importancia, ya que de nada sirve si cada uno de ellos se desempeña de forma individual y no como grupo, haciendo que el trabajo se vuelva más pesado/lento. Además esta técnica elimina las dificultades que suelen surgir en la comunicación entre dos partes que tienen puntos de vista diferentes, lo cual se soluciona hablando cara a cara las diferencias.
* Relación con 8 (Desarrollo sostenido): El ciclo de desarrollo de software es uno constante que requiere trabajo continuo por parte de los integrantes de un grupo en donde se van presentando entregas al cliente a lo largo del mismo y se van corrigiendo las distintas desviaciones que se tomaron a partir de los requerimientos del cliente.
* Relación con 9 (Atención a la excelencia técnica): Para que un equipo de trabajo funcione eficientemente y otorgue un producto de calidad debe poseer un conjunto de habilidades, conocimientos y experiencias que les permitan ser capaces de adaptarse a los cambios, que se pueden producir en cualquier momento, de forma rápida y efectiva. Por eso es importante conocer y entender las fortalezas y debilidades de todos los integrantes y cómo reaccionan ante las distintas situaciones que se puedan encontrar.
* Relación con 11 (Equipos autoorganizados): Los equipos autoorganizados permiten una organización y dan lugar a nuevas arquitecturas o diseños que pueden ser más eficientes, pero para que pueda existir esto se debe permitir la interacción y una cierta libertad entre los individuos que van a encargarse del trabajo.
* Relación con 12 ( Reflexión sobre efectividad y conducta): Muy ligado al principio anterior, se necesita que los grupos de trabajo reflexionen frecuentemente sobre sus formas de trabajo y sus conductas, para que, siempre que sea necesario, las modifiquen para funcionar de una forma más eficiente.

1. **Software sobre documentación: 1, 3, 7, 9, 10**

* Relación con 1.(Satisfacer al Cliente). Se debe establecer una retroalimentación con el cliente (para validar y aprobar el software entregable, entre otros) para permitir que el producto final se adapte rápidamente a lo que el cliente realmente quiere, obteniendo así su satisfacción.
* Relación con 3.(Entregar con frecuencia software que funcione): Se debe entregar software **funcional** frecuentemente, entre dos semanas y dos meses (y mejor si es al periodo de tiempo más corto posible). Lo mejor es entregar las tareas en forma secuencial en el tiempo lo antes posible, así el cliente pueda evaluarlas cómo entregadas y realizar sus modificaciones lo antes posible.
* Relación con 7.(Sw funcionando como medida de éxito): Es el software funcionando el que indica toda la funcionalidad conseguida y es lo que le permite al cliente marcar una tarea como terminada o no.

En las metodologías tradicionales se necesita mucho tiempo para elaborar la documentación antes de que se comience a codificar el producto. Esto al cliente le supone más un atraso que un avance porque no le permite ir probando la funcionalidad que ha solicitado.

* Relación con 9.(Atención Continua a la excelencia técnica y al buen diseño): este principio se relaciona principalmente con el hecho de poder garantizar la calidad del producto, la comprobación continua y la aplicación de este tipo de técnicas, como las pruebas automatizadas, le permite al equipo poder realizar entregas de calidad en menos tiempo y con menos costo.
* Relación con 10.(La simplicidad es esencial): Esta relación se ve reflejada en que se debe evitar perder tiempo con documentación excesiva y compleja, si bien es muy importante y este valor no indica que no se debe documentar, sí nos está diciendo que se debe documentar solo lo necesario, e implementar el tiempo que se perdería con documentación compleja que no aporta valor al producto, en el desarrollo de la funcionalidad, que al fin y al cabo es lo que al cliente le interesa.

1. **Colaboración con el cliente sobre negociación contractual: 1, 2, 6**

* Relación con principio 1 (Satisfacer al Cliente): La entrega temprana y continua de software de valor está estrechamente relacionada con la colaboración con el cliente, ya que si se deja de lado esta relación y se sigue estrictamente lo pactado en un contrato es muy probable que a la hora de entregar el producto, el cliente no quede del todo satisfecho. En cambio, si se mantiene cierta comunicación, e incluso el cliente se “integra” con el equipo de trabajo, este participara en la toma de decisiones de las diferentes etapas del desarrollo, y podrá hacer un seguimiento del proyecto, lo cual evitará disconformidades a la hora de la entrega.
* Relación con principio 2 (Requisitos cambiantes): Si se sigue estrictamente lo pactado en un contrato y no existe comunicación constante con el cliente, los requisitos cambiantes no llegarían hasta la entrega final del producto, y se perdería la ventaja que representan los mismos.
* Relación con principio 6 (Conversación cara a cara): Este valor del manifiesto ágil se centra en la comunicación constante con el cliente, y si bien no es necesario que para cada detalle o toma de decisiones dicha comunicación sea cara a cara, si es necesario y beneficioso pactar reuniones con el cliente para que de su feedback de forma presencial, ya que como lo dice el principio, es la forma más eficiente y efectiva de comunicar.

1. **Respuesta ante el cambio sobre seguir un plan: 2, 11, 12**

* Relación con 2 (Requisitos cambian): la relación con este principio se basa principalmente en los requerimientos que cambian en el tiempo. Es fundamental atender a estos, inclusive si llegan tarde al proceso de desarrollo, teniendo en cuenta su prioridad y significación para el negocio. Los requerimientos representan las necesidades de información que tiene el cliente, de ahí la ventaja competitiva que representan.
* Relación con 11 (Mejoras emergen de grupos auto-organizados): la autoorganización en un equipo de desarrollo lleva a que en cada iteración o entrega realizada de un producto, surjan en el proyecto mejoras en cuanto a la arquitectura y diseño, así como también el modo en que se tratan y afrontan nuevos requisitos o requisitos que cambian a lo largo del tiempo.
* Relación con 12 (Mejora continua del proceso): Como la ideología ágil se basa en la continua respuesta al cambio podemos decir que las salidas del proceso en una iteración se utilizan como entradas en la siguiente, reflexionando sobre la forma de lograr una mayor eficiencia y ajustando los comportamientos anteriores.