Resumen de Lean Development

Lean Software Development o simplemente Lean Development es un modelo de desarrollo de software enmarcado dentro de los modelos ágiles como Scrum o Extreme Programming. El término “lean” se puede traducir del inglés como fino o esbelto, ya que se refiere a lo que queda tras eliminar todo lo que no es esencial en un proyecto.

Lean Development es heredero directo de Lean Manufacturing (sistema de producción industrial) y del Sistema de Producción de Toyota y tiene en sus precursores el matrimonio Poppendieck.

El desarrollo de Software Lean se basa en 7 principios fundamentales:

1.- Eliminar los desperdicios

Ejemplos: código y funciones innecesarias, empezar más de lo que se puede terminar, demoras, requisitos poco claros, burocracia, comunicación lenta, mala calidad.

Explicar los mapas de flujo de valor.

2.- Amplificar el aprendizaje

El desarrollo de software es entendido como un ejercicio de descubrimiento y aprendizaje. Podemos aprender de la interacción de los usuarios con la interfaz del programa. También podemos acelerar el proceso de aprendizaje mediante iteraciones cortas. La retroalimentación o feedback del cliente es esencial. En estas reuniones el cliente puede ver en qué fase está el proyecto o consensuar mejoras o nuevas funcionalidades para el futuro. También pude aclarar ambigüedades y aportar información que nos permita decidir que hacer frente a una decisión que habíamos inicialmente aplazado. Además, el cliente consigue comprender qué es lo que realmente necesita y los desarrolladores aprende a ajustarse a estas necesidades.

3.- Decidir lo más tarde posible

4.- Entregar tan rápido como sea posible

Para conseguir entregar nuestros proyectos lo antes posible, es esencial el uso de iteraciones cortas. Estas mejorarán el aprendizaje y la comunicación dentro del equipo. Además, la entrega rápida imprime una velocidad al proyecto que nos obliga a posponer decisiones importantes por falta de información (decidir lo más tarde posible). Otro añadido para apostar por la entrega lo antes posible es que el cliente quiere que se le presente el producto que quiere hoy y no el que quería ayer. Una frase que también representa muy bien el principio de entregar lo más rápido posible es que una solución al 80% hoy es mejor que una al 100% mañana. Esto se explica perfectamente con los cambios de opinión continuos de algunos clientes. Por si fuera poco, los clientes valoran muy positivamente la entrega rápida de un producto de calidad. La alta frecuencia de las iteraciones también evita largos periodos de aislamiento del equipo de desarrollo con el cliente, lo que implicaría una montaña de información que procesar cuando se produjera la próxima reunión.

5.- Capacitar al equipo

6.- Construir integridad intrínseca

Partimos de la necesidad de que el cliente tenga una visión general del sistema. El cliente quiere saber qué hace el sistema, si es intuitivo, cuánto cuesta mantenerlo… Para ello necesitamos la integridad conceptual: que todos los componentes del sistema que funcionan bien de manera separada lo hagan también juntos. Una forma de construir integridad intrínseca es tomar la información en pequeños lotes y prescindir en la medida de lo posible de la documentación, en favor de la comunicación cara a cara, tanto con el cliente como entre los miembros del grupo. Con ello conseguiremos un equipo más cohesionado y un trato más personal con el cliente. Otra forma de construir integridad intrínseca es la refactorización, es decir, limpiar el código de manera que no haya bloques de código repetido (desperdicios) o simplificar el cómo hacen una tarea las funciones sin modificar el qué hacen.

7.-Ver todo el conjunto