

AGILE MANIFESTO

Experiencias personales

- Intención:

- Experiencias personales en temas relacionados con *Agile*
- Casos de éxito y fracaso
- ¿ Es posible utilizarlo ?
- ¿ Es conveniente ?

- Guillermo Caro Murillo

- Ingeniero Civil
- Desarrollador de software durante 30 años
- Socio Estrada Caro y Cia
- Junta Acis
- Organizador maratón de programación

Plan de presentación

- Por qué importa en gerencia de proyectos.
- Origen.
- Manifiesto.
- Principios y comentarios.
- Características.
- Experiencias de otros.
- Resumen y conclusiones.

Origen del Agile Manifesto

- [<http://agilemanifesto.org/history.html>]
- Febrero 11-13 de 2001, Snowbird ski resort
- Se reúnen 17 personas a conversar, tomar, comer, descansar y esquiar.
- De la charla sale Agile “Software Development” Manifesto.
- Representantes de Extreme programming, SCRUM, DSDM, Crystal ...
- Según Alistair Cockburn el problema iba a ser poner de acuerdo a semejante grupo en algo significativo.
- Para sorpresa de ellos, estuvieron muy de acuerdo.

Origen del Agile Manifesto

- El fondo es promover organizaciones basadas en la gente, la colaboración, lugares donde uno quiere trabajar.
- Objetivo real: entregar productos a clientes operando en un ambiente que mas que hablar de que “nuestro capital más importante es la gente” realmente “actúe” pensando que lo es, y quite la palabra “capital”.

Temas de preocupación

- Satisfacción de expectativas del cliente Y del proveedor.
- Dificultades de comunicación, diferencias de lenguaje.
- Plazos impuestos.
- Metodologías pesadas
- Procesos “porque si”.
- Obligaciones “de libro”
- Resultados efectivos

Manifiesto for Agile Software Development

- **Individual and interactions** over process and tools
- **Working software** over comprehensive documentation
- **Customer collaboration** over contract negotiation.
- **Responding to change** over following a plan
- That is, while there is value in the items on the right, we value the left more
- **Individuos e interacciones** mas que procesos y herramientas.
- **Sistemas funcionando** antes que documentación detallada
- **Colaboración con el cliente** antes que negociación de contrato.
- **Responder al cambio** antes que seguir el plan.
- Esto es, aunque los items de la derecha son valiosos, les damos mas valor a los de la izquierda

Principios detrás del Manifiesto

- Nuestra prioridad es satisfacer al cliente entregándole desde temprano y con frecuencia software valioso.
- Éxito: BP
 - Primera versión funcionando en 29 días
 - Entregas incrementales y frecuentes durante años
 - Percepción del cliente:
 - Operación de corazón abierto, con el paciente corriendo y en excelente estado de salud.
 - 100% de satisfacción varios años consecutivos.

Principios detrás del Manifiesto

- Bienvenidos los cambios en los requerimientos, aún al final del desarrollo. El proceso ágil explota el cambio a favor de la ventaja competitiva del usuario.
 - Éxito: BP
 - Mantenemos una lista de requerimientos por atender muy grande y mensualmente definimos el foco del trabajo.
 - Error: Compras
 - Por actitud no quería aceptar el cambio, tuve serias dificultades con el contrato.

Principios detrás del Manifiesto

- Entregar software operando frecuentemente, entre un par de semanas y un par de meses, con preferencia del plazo más corto.
- Éxito: BP
 - Entregamos funcionalidades desarrolladas, documentadas, probadas en días, excepcionalmente demoran más de dos meses.
- Comentario:
 - El objetivo es ser más estricto con el plazo y menos con los entregables.
 - Es bastante difícil que el cliente “pierda” parte de su control

Principios detrás del Manifiesto

- Las personas del negocio y los desarrolladores deben trabajar juntos diariamente, a lo largo de todo el proyecto.
- Error: TD
 - Trabajo con el área técnica y no con los usuarios, al final no sirve.

Principios detrás del Manifiesto

- Construir proyectos basado en individuos motivados. Darles el ambiente y el apoyo, y creer que van a hacer el trabajo.
- Éxito: TD
 - El grupo hizo un excelente trabajo de desarrollo.
- Éxito: COMPRAS:
 - Excelente trabajo técnico.
- Comentario:
 - Se requiere un grupo de profesionales muy comprometidos.

Principios detrás del Manifiesto

- El método más eficiente y efectivo de intercambiar información a y entre el grupo es por conversaciones cara a cara.
- Éxito: BP
 - Mantenemos permanentes minireuniones y conversaciones, hay buen flujo de información.
- Comentario:
 - Los documentos son el soporte de las conversaciones, no su reemplazo.

Principios detrás del Manifiesto

- El software operando es la principal medida del progreso.
- Éxito: UJ – sistema de admisiones - 1973
 - Se desarrolló en llave permanente con el usuario, “se hace camino al andar”.
- Error: Segunda versión del mismo sistema.
 - Por inexperiencia del grupo se convirtió en reunionitis sin ningún resultado.
- Éxito: bp
 - Ha logrado sobrevivir muchos ataques y dificultades porque permanentemente entrega funcionalidad operando y correcta.

Principios detrás del Manifiesto

- Los procesos ágiles promueven desarrollo sostenido. Los promotores, usuarios y desarrolladores deben poder mantener un ritmo constante indefinidamente.
- Éxito: BP
 - Hemos mantenido años de desarrollo continuo.

Principios detrás del Manifiesto

- La atención continua a la excelencia técnica y buen diseño aumenta la agilidad.
- Comentario:
 - Es bien exigente en la calidad del grupo técnico, unidad de criterio y definición de arquitectura.

Principios detrás del Manifiesto

- Es esencial la simplicidad: el arte de maximizar los cantidad de trabajo que NO se debe hacer.
- Comentario:
 - Es un reto bien complejo.
 - Va en contra de la forma de pensar del técnico que le encantan las “Golosinas tecnológicas”
 -

Principios detrás del Manifiesto

- Las mejores arquitecturas, requerimientos y diseños surgen de grupos auto organizados.
- Éxito: US.
 - El equipo de desarrollo y los usuarios se han organizado informalmente y operan muy bien.
- Comentario:
 - Exige del líder el coraje de creer más en su gente que en el control personal.

Principios detrás del Manifiesto

- A intervalos regulares, el equipo reflexiona sobre cómo se más efectivo, entonces ajusta y sintoniza su comportamiento.
- Comentario:
 - Es un reto bien interesante.
 - Exige del líder entregar parte de su “poder” y “empoderar” al grupo.

Características de procesos ágiles

- Iterativo: el proyecto se desarrolla por medio de varias iteraciones, cada una es un mini proyecto completo, con entregables concretos y operativos.
 - Éxito: HipoAndes
 - Se inició con una versión del software que permitió iniciar el proyecto, durante varios años se agregó funcionalidad, respondiendo al negocio.
 - Error: CN.
 - Hice un desarrollo técnicamente muy lindo, lleno de golosinas tecnológicas, al final resultó que el usuario no podía obtener la información básica que pensé que era “obvio” que debería tener.

Características de procesos ágiles

- Incremental: La primera entrega es una versión mínima pero completa del sistema, en cada iteración se va creciendo

Características de procesos ágiles

- Auto organizado: el equipo tiene la autonomía para organizarse para desarrollar en la mejor forma sus resultados

Características de procesos ágiles

- Emergente: se permite que los requerimientos y las tecnologías surjan durante el desarrollo
- Éxito: TD
 - El reto era construir en muy corto plazo una gran cantidad de pantallas e informes, construimos un generador de código especializado para el proyecto.

Experiencias de otros

- eWorkshop on Agile Methods
- Empirical Findings in Agile Methods
- Mikael Lindvall, Vic Basili, Barry Boehm, Patricia Costa,
- Kathleen Dangle, Forrest Shull, Roseanne Tesoriero
- Laurie Williams, Mervin Zelkowitz

eWorkshop on Agile Methods

- Motivación:
 - Durante los 90 varios desarrolladores encuentran algunas metodologías pesadas, frustrantes y difíciles de seguir.
 - Las metodologías basadas en la planeación total desde el principio tienen dificultades manejando los cambios.
 - Como resultado han surgido varias propuestas que en conjunto forman parte del Agile Manifesto.
 - El objetivo de CeBASE Center of Empirically-Based Software Engineering es recolectar, analizar, documentar y diseminar información de casos reales.
-
-

eWorkshop on Agile Methods

- Definición de métodos ágiles

Iterativo

Incremental

Auto Organizado

Emergente

-

eWorkshop on Agile Methods

- **Tamaño del proyecto:**

Es el factor más importante para determinar si es aplicable

Hay bastantes experiencias hasta 12 personas.

Algunos casos alrededor de 25.

Pocos llegan a 100.

Casos aislados de más de 100, llegando a 800.

La percepción general es que cuando crece el grupo las dificultades de comunicación cara a cara son grandes, se requieren técnicas adicionales como partir los grupos.

—

•

•

eWorkshop on Agile Methods

- Personal.
 - El mayor debate está alrededor de que métodos ágiles requieren gente especialmente buena, y ésta produce resultados con cualquier metodología.
 - Hay consenso en la necesidad de 25% a 33% de personas competentes y experimentadas.
 - Competente:
 - Experiencia en el mundo real en la tecnología.
 - Ha construido sistemas similares en el pasado.
 - Tiene habilidades de comunicación.
-

eWorkshop on Agile Methods

- Criticidad, confiabilidad, seguridad.
 - Una de las críticas más fuertes es que los métodos ágiles no son adecuados cuando hay exigencias en éstos aspectos.
 - Parece que temas como desempeño se deben especificar desde el principio para que no sean un dolor de cabeza.
 - Hay algún consenso en que el énfasis en las pruebas hacen que sea aplicables con exigencias de confiabilidad.
-

eWorkshop on Agile Methods

- Necesidades de entrenamiento
 - La mayoría piensan que se requiere menos entrenamiento que con metodologías tradicionales.
 - Por ejemplo la programación por pares facilita el entrenamiento de uno a otro.
 - Algunos se han auto-entrenado satisfactoriamente.
-

eWorkshop on Agile Methods

- Factores de éxito.
 - Cultura:
 - control en los grupos, soporte de la gerencia, negociación.
 - Gente:
 - Se requieren miembros competentes, menos que en otras metodologías, pero más competentes.
 - Comunicación
 - Ágil, permanente.

eWorkshop on Agile Methods

- Documentación.

- Es un tema de discusión, con posiciones encontradas.

- ¿ es necesaria ?

- ¿ qué tanta ?

- Hay algún consenso:

- La documentación es una forma pobre de comunicación, pero en muchos casos necesaria.

- Es clave mantenerla bajo control para que realmente cumpla un cometido.

-

-

Resumen y Conclusiones

- Las metodologías ágiles tienen promesas interesantes.
- No son la panacea para todos los casos.
- Requieren claridad, coraje, soporte.
- Son más adecuadas para grupos relativamente pequeños, con disciplina se pueden aplicar a grupos grandes.
- Requieren un “Nucleo” de gente capaz.
- Es un tema en ebullición.