

El proceso de Ingeniería de Requisitos en el Ciclo Global del Software

Metodologías Agiles

Metodologías que dan soporte a los valores incluidos en el Manifiesto Ágil. Son iterativos de ciclo corto y flexible.

Se centran más en el código que en la documentación debido a:

- Son más adaptativos que predictivos: poca planificación
- Orientados a las personas: Confían en la experiencia, competencia y colaboración

Manifiesto Agil

Alternativa a las metodologías tradicionales:

- Valorar a los individuos y su interacción con procesos y herramientas
- Valorar software que funcione que sobre documentación exhaustiva
- Valorar la colaboración con el cliente sobre la negociación contractual
- Valorar la respuesta al cambio sobre el seguimiento de un plan.

Principios Agiles

1. La principal prioridad es la satisfacción del cliente, entregando al cliente lo que quiere rápidamente.
2. Son bienvenidos los requisitos cambiantes, esta metodología es constantemente cambiante.
3. Entregamos frecuentemente software que funcione.
4. Las personas de negocio y desarrolladores deben trabajar juntos
5. Se construyen los proyectos alrededor de individuos motivados.
6. Siempre se deben comunicar la información cara a cara con reuniones con el cliente y retrospectivas.
7. Se mide el avance en función a las validaciones del cliente.
8. Se propone un desarrollo sostenible
9. Atención continua a la excelencia, deben tener excelencia tanto el equipo de desarrollo como el producto resultante.
10. Las mejores arquitecturas siempre salen de los equipos que se autoorganizan.
11. En intervalos regulares el equipo se tiene que juntar y ver en que se puede mejorar el proyecto.

Tipos de metodologías Agiles

- Extreme Programming (XP)
 - Se prioriza la historia de usuario
 - Entrevistas
 - Brainstorming

- Modelado Agil (AM)
- Scrum

Metodología Agil VS Metodologías tradicionales con respecto a la captura de requisitos

- En la Agil no hay tanta documentación como en la tradicional
- Cambia mucho la prioridad, en la Agil son muy frecuentes los cambios de requisitos
- Especificación de requisitos (Agil)
 - Solamente se usa para entender ciertos procesos
 - Se usan historias de usuario
- Validación de requisitos
 - Se usan reuniones de revisión
- Gestión de requisitos
 - No hay trazabilidad total como en la tradicional
 - Aquí se pueden dejar requisitos To Be Defined (TBD)

Las metodologías tradicionales se utilizan en proyectos críticos ya que ahí es imposible utilizar metodologías Agiles. En Agile las fases no están definidas y hay muy poca documentación, lo que las hace fáciles de mantener.

Scrum

Es una metodología ágil “Framework” o un conjunto de buenas prácticas para la gestión de proyectos. Consiste en iteraciones cortas de tiempo e iteraciones incrementales.

Roles de Scrum

- Product Owner: el cliente
 - Gestiona las necesidades que serán satisfechas por el proyecto
 - Recoger y tener claras las necesidades de la aplicación
 - Fijar criterios de aceptación para cada historia de usuario
- Equipo de desarrollo: los que desarrollan el proyecto
 - Autoorganizado
 - Multifuncional
 - Autónomo
- Scrum Master: Se encarga de solucionar problemas en el sprint y de revisar que se cumplan los plazos
 - Sería equivalente a un jefe de proyecto
 - Soluciona todo lo que pasa, problemas, impedimentos, etc...
 - Actúa como mentor.

Artefactos de Scrum

- Historias de usuario: Similar a los requisitos, pero a más alto nivel

- Son requisitos ambiguos
- Descripciones breves de lo que quiere el cliente
- Ejemplo: “**Como** cliente **quiero** poder añadir elementos al carrito **para poder** revisarlos antes de comprar”
- Product Backlog: Documento de especificación de requisitos
 - Es como un documento de requisitos
 - Recopila historias de usuario y las organiza por prioridad
 - Puede crecer y decrecer, nunca va a estar completo.
 - El product Owner prioriza con Moscow: ← **Importante**
 - M (Must)
 - S (Should)
 - C (Could)
 - W (Wont)
- Sprint Backlog: El Subconjunto de tareas que se realizarán en un sprint
 - Tareas seleccionadas a realizar durante un sprint
 - Se coge un subconjunto de tareas del Product Backlog.
 - Las tarjetas tendrán puntos y el valor de las tarjetas tomadas no puede superar cierto valor para evitar sobresaturación.

Eventos de Scrum

- Sprint: período de tiempo en el que hay que hacer una entrega
 - De 1 a 4 Semanas.
 - El resultado será un producto de software potencialmente entregable (Prototipo operativo)
- Planificación del Sprint
 - Primer día del sprint
 - Product owner, Scrum Master y el equipo de desarrollo
- Reuniones de Sprint
 - Daily meeting:
 - No más de 15 minutos
 - Scrum Master y Equipo de desarrollo
 - Para solucionar problemas y ver como va el p'royecto
 - Revisión del sprint
 - No más de 4 horas
 - Product Owner, Scrum Master y Equipo de desarrollo
 - BurnDown Chart.
- Retrospectiva: Una vez terminado el sprint ver que se puede mejorar de este
 - Se reune todo el equipo para ver aspectos positivos y negativos del sprint.
 - Product Owner y equipo de desarrollo.

Scrum vs Requisitos

- No hay planificación
- Requisitos poco completos, a alto nivel
- Las historias de usuario se considerarán algo similar a los requisitos.

Last update:

2023/11/29 ingenieria_de_requisitos:procesoir_ciclo_global http://knoppia.net/doku.php?id=ingenieria_de_requisitos:procesoir_ciclo_global
08:11

From:

<http://knoppia.net/> - **Knoppia**



Permanent link:

http://knoppia.net/doku.php?id=ingenieria_de_requisitos:procesoir_ciclo_global

Last update: **2023/11/29 08:11**