

Proceso de pruebas desde los requerimientos

Planificación → Identificación → Especificación → Ejecución → Análisis de fallas → Fin de las pruebas.

casos de prueba
métodos
personas

DISÑO

hay defectos!

dejamos de probar?

Entregables - plan de prueba
- casos de prueba (datos)
- reporte de incidentes
- informe final.

cuántas líneas de código necesitamos para hacer testing → 0

en ciclo de vida

en V: testamos en todas las etapas desde los requerimientos.

desarrolla desde lo general a lo particular, se prueba de lo particular a lo general (pruebas unitarias)

Testing ≠ QA - testing =

cuánto testing es suficiente. - no podemos abarcar todo pero no es excusa.

- ↳ depende - costo - beneficio (al final)
- ↳ riesgo: criticidad del sistema

↳ el criterio de aceptación - acuerdo de que tanto testing → varía en un caso de prueba.

Principios

- ↳ presencia de defectos
- ↳ exhaustivo es imposible
- ↳ testing temprano
- agrupamiento de defectos
- paradoja del pesticida
- dependiente del contexto
- autor revisar código propio.

Clase 12/10

Lean y Kanban

Libro: lean software development. - cada libro es un principio.

parecido al manifiesto ágil. cada framework puede variar en art.

- más viejo, Japón 1946

- surgió en industrias tangibles. (Toyota system)

compatibilidad con ágil.

eliminación
reducción de
desperdicio

kanban - generación de servicios. cualquier intangible

* ① todo aquello que no genera valor

Lean

Principios

- ① *
- ② amplificar aprendizaje: compartir el aprendizaje
- ③ Embeber la inteligencia conceptual: calidad del producto presente siempre
- ④ Diferir compromisos: hasta el último momento responsable
 - ↳ toma de decisiones - basadas en información
 - ↳ postergar los compromisos hasta que tengamos más info.
 - ↳ si esperamos al momento perfecto nunca lo vamos a hacer.

Just in time
asignación de
trabajo

- ⑤ Dar el poder al equipo - empoderarlo
- ⑥ ver el todo - necesitar tener una visión completa del proceso y producto.
 - ↳ saber donde estamos parados.
 - ↳ importante en los intangibles
- ⑦ Entregar lo antes posible - retroalimentación / feedback

10

DESPERDICIOS

Costos en producción en productores lean

- talento no utilizado -
- inventario
- movimiento - personal y recursos
- espera
- transporte
- defectos
- sobreproducción
- sobreprocesamiento - hacer de más

en software

- caract. extra
- trabajo a medias
- proceso extra: lo que no genera valor
- movimientos

- defectos
- esperas

- cambio de tareas - multitasking: una cosa por vez
- talento no utilizado o subutilizado
- equipos de desarrolladores pasados a análisis p.ej.

binario
↓
terminado
no

kanban - no metodología - método o framework

signo canal
kan-ban → serializado

KANBAN - sist. de trabajo.

mejoras evolutivas a nuestros procesos.

- ↳ software o físicos
- ↳ entrega de servicios
- ↳ mejora gradual de los procesos

nace en Toyota 1940 - Taiichi Ohno

- ↳ reducción de costos de almacenamiento
- ↳ just in time: cuando haya falta

valores

- ↳ transparencia
- ↳ equilibrio
- ↳ colaboración
- ↳ foco en el cliente

- ↳ flujo
- ↳ liderazgo
- ↳ entrenamiento
- ↳ acuerdo

- respeto

principios

gestión de cambios

- ↳ comenzar con lo que tienes ahora - tu proceso actual
- ↳ no "todo lo de antes no sirve"
- ↳ propone cambios evolutivos
- ↳ fomentar lazos de liderazgo

↑ entrega continua
feedback

entrega de servicios

- ↳ comprender expectativas del cliente
- ↳ auto organización de los trabajadores
- ↳ mejora de productores entregador
- ↳ red de servicios y políticas

prácticas generales - cambios evolutivos (complicado).

① Visualizar - debemos poder ver todo el proceso. todo lo mto necesaria disponible

↳ TRANSPARENCIA

↳ tablero - cada paso del proceso es una columna.

↳ kanban: señalización

↓
tello por ej o físico

② Limitar el WIP (work in progress)

↳ cantidad de trabajo en cada columna.

↳ si agregamos algo debemos retirar otro (si está lleno). - avanzamos con el proyecto.

③ Ayudar a que el trabajo fluya.

↳ asignamos límites explícitos de cuántos items puede haber en progreso en cada estado del flujo de trabajo (mejor en todas)

④ Gestionar el flujo del trabajo

↳ pull no push

↳ asignación de trabajo: el trabajador va y busca
NO LE DAN.

KANBAN
David
Anderson

WIP
colas de proceso → en progreso

Tipos de trabajo

- requerimientos
- defectos
- desarrollo
- solicitudes

asignando capacidad en fn. de la demanda.
podemos tener una fila por cada tipo de trabajo.

④ Hacer explícitas las políticas - pocas pero explícitas

DEBEN SER

- deben ser acordadas entre todas las partes
- expuestas
- acuerdos
- expreso
- fecha fija ^{si no manejamos bien queda expreso} - que llegue en la fecha pactada
- Estándar - calidad de servicio estándar - FIFO
- Intangible - difíciles de explicar

Pocas
Sencillas

Visibles
aplicables en
todo
momento

Bien definidas
fácilmente
modificables
(adaptaciones
continuas)

⑤ Mejorar colaborativamente - retrospectiva de scrum - mejora del sist. de Trabajo.

que se pide y se entrega.

"kanban university"

lead time - tiempo entre ~~pedido~~ de ~~entrega~~ de ~~entrega~~.

¿cómo aplicar Kanban?

- L proceso
- L trabajo
- L WIP ^{mejoramos con experiencia}
- políticas
- vellos de botella - ^{intercambio} swarming - ^{después de hacer lo que estás haciendo para ayudar a uno.}
- clase de servicio
- cadencia

~~mejoramos con experiencia~~

scrum - iterativo por proyecto

Flujo continuo - No way sprint, fluye por un camino. No se limpia el tablero.

↳ "deja de empezar y comienza a terminar"

Políticas ^{define el que} - reposición de trabajo... - DoD, DoR
↳ clase de servicio.

cadencias - ceremonias de retroalimentación

- Team Kanban meeting (daily)
- Team retrospective (^{scrum sprint}retrospective)
- Team Replenishment meeting (^{como sprint}planning)

métricas

- Lead time - vista del cliente (tiempo de entrega)
 - ↳ desde que entra hasta que sale
 - ↳ tiempo de entrega
- Cycle time - vista interna
 - ↳ desde que ~~entra~~ inicia hasta que termine el proceso
 - ↳ tiempo de terminación.