

Ingeniería de Software

Clase 4 (20/03/2025)

Kanban

Metodología que optimiza el proceso de desarrollo de software por un flujo continuo de trabajo.

Kanban es un ciclo cerrado y todo lo que se compromete uno es lo que se tiene que entregar al final del sprint.

- Desarrollo incremental, dividiendo el trabajo en partes
-Backlog | Stories | To do | In progress | To verify | Done

Básicamente, dividir el trabajo de forma visual (con notas)

- Limitante principal es limitar el trabajo en progreso en cada etapa.
- Por columna poner ciertas restricciones para tener trazabilidad

Cuando el equipo es nuevo, se avanza en un trabajo simultáneamente, es decir, me excedo de trabajo en un área.

- **Tarjeta de señal:** Unidad de trabajo, la cual se mueve a traves del flujo de la organización solo cuando existe la capacidad de tomar la tarea e el sgte paso del proceso.
- Puede perder la visibilidad del objetivo final

Principios de Kanban

1. Visualizar el trabajo
Todos pueden visualizar el estado de cada tarea
Promueve la transparencia
2. Limita el trabajo en progreso (WIP)
3. Gestionar el flujo
Observa y optimiza el trabajo a traves del progreso
Identificación del bloqueo
4. Visualización de Políticas
Yo no puedo moverme a una columna sin que este
_todos los commits hechos, criterios de aceptación
5. Colaboración, evolución y experimentación
Colaboración del equipo para mejorar el proceso
Evaluación de resultados
Adaptación del proceso

Requerimientos funcionales y no funcionales

Existen dos requerimientos:

Usuario (funcionales)

Requerimientos: Descripciones de los servicios del sistema en función de las necesidades del cliente

Sistema (No funcionales)

Limitaciones: Netamente asociados a un sistema o dominio

La documentación es importante en cuanto a requerimientos. Forman un contrato de trabajo de un producto

Requerimientos Funcionales (FR)

El producto debe ofrecer un servicio para satisfacer necesidades

- Principales Lectores:
Gerentes, usuarios finales del sistema, ingenieros o analistas funcionales, contratistas (financieros).
- Requerimientos debe estar escrito a prueba de weones
- Puede incluir diagramas de servicio
- Al ser de alto nivel, pueden ser muy ambiguos
EJ) Un usuario debe ser capaz de buscar las listas de citas de todas las clinicas
MUY GENERAL (especificar los requerimientos)

Para escribir un FR se ocupa el sgte formato:

"El ... (tipo de usuario / sistema / componente del sistema) **NECESIDADES**

Verbo en futuro

Descripcion"

- Las **necesidades** pueden venir de un cliente o usuario

EJ) "El **administrador** podrá crear usuarios de tipo "Read Only" para que puedan acceder al sistema"

En desarrollo Ágil (US)

los requerimientos funcionales son denominados **Historias de Usuario**

- Estas historias de usuario tambo conviven en una línea temporal
EJ) Requerimiento para etapa 1, etapa 2, etapa 3...
- Forman parte de un Product Roadmap

Son INVEST

Independent

Negotiable

Valuable

Estimable

Small

Testable

Formato de escritura

Como	RoI
Quiero (necesito)	Funcionalidad
Para	Beneficio
Supuesto (opcional depende del contexto)	Relacion

Criterios de Aceptación

- Criterio 1
- Criterio 2
- Criterio 3

Requerimientos NO Funcionales (NFR)

Limitaciones a los servicios o funciones que ofrece el sistema o producto

- Al ser una capa un poco más "profunda", al ser mas restrictivo puedo evitar gastos innecesarios
- Se aplica al sistema en conjunto o bien a servicios individuales
- Definen las propiedades del sistema o producto, tiempo de respuesta y almacenamiento
-

EJ) Sistema operativo, version de navegador / libreria, lenguaje de programacion, etc
la app funciona en ios y para android (versiones especificas)

Requerimientos mas restrictivos

#Clasificado

Clasificaciones

Requerimientos del Producto

- Velocidad de ejecucion+
- Fiabilidad de datos
- % de error aceptado (casi 0%)

Requerimientos Organizacionales

- Políticas y procedimientos organizacionales
- Estandares
- Requisitos de implementacion

Requerimientos Externos

Externos al sistema

- Proceso de desarrollo
- Interoperabilidad
- Legislativo (si hay conflicto de contrato)

Métricas

Propiedad	
Velocidad	
Tamaño	
Facilidad de uso	
Reliability	Tiempo medio de fallo / Disponibilidad
Confiabilidad	
Portabilidad	

Hay apps que murieron por ser lentas

Clase 5 (24/03/2025)

Requerimientos de Dominio

Derivan del dominio de la aplicación del sistema o producto.

El entorno es un dominio (puede ser dominio de negocio)

En el dominio también hay requerimientos, por lo tanto restricciones

EJ) Sistema de control de trenes tiene en cuenta características de frenado en distintas condiciones climáticas.

EJ 2) Sistema de control tiene características de la pista, vientos, condiciones climáticas extremas.

Van asociados al contexto del negocio donde está mi producto

Tiende a restringir más que dar aperturas

Muchas veces si los requerimientos de dominio no están satisfechos, el sistema o producto queda inútil.

Documentación de Requerimientos

Es un documento oficial que los desarrollos de software implementan.

Incluyen requerimientos funcionales, no funcionales y de dominio

- Relevante cuando el desarrollo de software se externaliza
 - En frameworks ágiles, los documentos de requerimiento pasan a ser las historias de usuario
- Por qué ocurre esto?
- Producto y sus funcionalidades pueden cambiar en el tiempo
 - Backlog vivo y versionamiento de roadmap de producto.

Debe incluir:

- Requerimientos de los clientes (audiencias)
- Detalle para desarrollo de las funcionalidades
- Información relevante del sistema

Usuarios de un documento de requerimientos:

Clientes del sistema

Especifican los requerimientos y los leen para comprobar que cubren sus necesidades

Administradores

Usan doc de req. para planear una cotización para el sistema y el proceso de desarrollo del sist.

Ingenieros del sistema

Ven los req. para ver que se desarrolla

Ingenieros de prueba del sistema

Ven los req. para desarrollar pruebas de validación para el sist.

Contenido

1. Prefacio
2. Introduccion
3. Glosario
 - Diccionario de terminos abordados*
4. Requerimientos de usuario
5. Arquitectura
6. Especificación de Requerimientos
7. Modelos del Sistema
8. Evolución
9. Apéndices (info adicional de config)
10. Índice

Levantamiento de Requerimientos

Muchas veces no se sabe lo que se requiere.

Muchas veces son necesidades y decisiones no basadas en numeros, sino de "guata"

Muchas veces los stakeholders no estan alineados respecto a la vision estrategica.

Ciclo exotico:

1. Descubrimiento de requerimientos

Interacción con las partes interesadas y/o stakeholders para descubrir las necesidades (Req funcionales, no funcionales y dominio)

2. Clasificación y organizacion de requerimientos

Se agrupan y ordenan en grupos coherentes

3. Priorización y negociacion de requerimientos

Se da prioridad y se abordan los conflictos

4. Especificación de requerimientos

Se documentan y se van generando versiones.

Muchas veces se genera una nueva iteración.

Técnicas de Levantamiento de requerimientos

****Entrevistas**

- Formales o informales
- Usuarios claves o stakeholders
 - **Levantamiento del proceso de negocio**
- Como funciona hoy y como puede ser mas eficiente
- Procesos a alto nivel y de a poco va abordando más detalle (dependiendo del problema)
- Identificar usuarios claves para cada una de las interacciones / etapas

Si voy a hablar con una persona que no esta al 100% con el proyecto, perdida de tiempo

Workshops

Encuestas de Satisfacción

User Experience / Se basa en la competencia

Prototipado

Para temas de agilidad.

Preguntas para criterios de aceptacion

- ¿Es pequeño?
- Falta el supuesto
- Falta el contexto
- Cada criterio debiera ser su propia historia
- Como funciona ahora -> Como quiero que funcione (Cambio de alcance)