**DevOps metologia agil**

es una forma de trabajo en desarrollo de software que busca unir (Development + Operations) el área de desarrollo (programadores) con el área de operaciones (infraestructura, servidores, despliegues, etc.) para trabajar de manera más colaborativa, rápida y automatizada.

En lugar de que los desarrolladores hagan un programa y luego "se lo pasen" a los de operaciones para que lo monten, en DevOps ambos trabajan juntos desde el inicio, usando automatización, integración continua y despliegue continuo.

**METODOLOGÍA ITIL**

ITIL es la abreviatura de Information Technology Infrastructure Library

es un conjunto de libros y archivos que documentan todo lo que las empresas del sector están haciendo para mejorar la calidad de los servicios a través de experiencias y nuevos aprendizajes.

La metodología consiste en identificar las prioridades informáticas de la organización para proponer formas de mejorar el trabajo, con ganancias para todos los procesos.

La biblioteca de buenas prácticas incluye operaciones, infraestructuras, mantenimiento e incluso atención al usuario.

La metodología gestiona un conjunto de competencias para coordinar recursos, con el objetivo de generar valor en los servicios.

las ventajas de esta metodologia:

* La metodología se adapta a todas las empresas, desde las pequeñas hasta las grandes.
* ITIL estandariza procesos y procedimientos.
* Esto significa menos tiempo en la ejecución de las actividades y una reducción de las fallas.
* Se refuerza la gestión.
* Los servicios se mejoran continuamente.
* Los clientes internos y externos están más satisfechos con las operaciones.
* Se reducen los costos operativos.
* Los empleados son más productivos y eficientes.

**METODOLOGÍA COBIT**

Está metodología está enlazada con las TIC

COBIT se enfoca en qué hacer, más que en cómo hacerlo. Esto significa que define los objetivos de control para los procesos de TI, pero deja la flexibilidad a cada organización para que implemente las técnicas y herramientas específicas que mejor se adapten a su contexto.

COBIT se utiliza en todo tipo de organizaciones, desde pequeñas empresas hasta grandes corporaciones, tanto en el sector público como en el privado. Es útil en entornos con regulaciones estrictas o donde la TI es un componente crítico para el éxito del negocio.

Se debe emplear COBIT cuando una organización necesita alinear estratégicamente la TI con sus objetivos de negocio

Es la herramienta a usar si la empresa busca:

* Mejorar la toma de decisiones relacionadas con la TI.
* Reducir los riesgos de seguridad y operacionales.
* Optimizar el valor que se obtiene de las inversiones en TI.
* Aumentar la transparencia y la rendición de cuentas en la gestión de la tecnología.

**METODOLOGIA INCREMENTAL**

El modelo incremental se centra en generar software operativo de forma rápida pero admisible. Los requisitos del proyecto tienen una prioridad asignada, cada cual entregado según el orden de incremento correspondiente.

Este enfoque, que se usó inicialmente para proyectos de software aunque más tarde se aplicó a otros sectores, establece entregas parciales mediante un calendario de plazos. En cada una de ellas, el producto debe mostrar una evolución con respecto a la fecha anterior, nunca puede ser igual

las caracteristicas de esta metodologia son:

* Los incrementos son pequeños.
* Permite una fácil administración de las tareas en cada iteración.
* La inversión se materializa a corto plazo.
* Es un modelo propicio a cambios o modificaciones.
* Se adapta a las necesidades que surjan.

**METODOLOGIA CANVAM**

es una técnica de tablero que sirve para organizar proyectos, se organiza visualmente, ayuda a identificar cuellos de botella y mejorar la calidad de entrega;Que hace haciendo hecho son las tre columnas del tablero.

**CARACTERÍSTICAS**

Visual y sencillo: utiliza un solo lienzo con bloques, lo que facilita su comprensión.

Enfoque integral: abarca desde clientes y propuesta de valor hasta costos e ingresos.

Flexibilidad: se adapta a cualquier tipo deempresa o proyecto (emprendimientos, startups, grandes compañías).

Colaborativo: fomenta el trabajo en equipo porque todos pueden aportar ideas en el mismo lienzo.

Dinámico: se puede modificar fácilmente al probar hipótesis o pivotar el modelo de negocio.

Basado en bloques: organiza la información en nueve secciones: socios clave, actividades clave, recursos clave, propuesta de valor, relaciones con clientes, canales, segmentos de clientes, estructura de costos y fuentes de ingresos.

**VENTAJAS**

* Muestra el negocio de forma clara y visual.
* Facilita la comunicación con el equipo o inversores.
* Permite innovar y detectar oportunidades.
* Es rápido y flexible para planear.
* Ayuda en la toma de decisiones estratégicas**.**

**DESVENTAJAS**

* Puede simplificar demasiado la realidad.
* Refleja un modelo estático, no su evolución.
* Depende de la correcta interpretación de los bloques.
* No reemplaza un plan de negocios completo.
* Puede basarse en suposiciones poco reales**.**

**METODOLOGIA XP**

Extreme Programming (XP) es una metodología ágil que optimiza el desarrollo

de software. Su objetivo principal es entregar productos de alta calidad

mediante la implementación de ciclos cortos, una colaboración constante y

una adaptabilidad excepcional a los cambios del proyecto.

**5 VALORES FUNDAMENTALES**

1. COMUNICACION
2. SIMPLICIDAD
3. RETROALIMENTACIÓN
4. CORAJE
5. RESPETO.

**FUNCIONALIDAD**

Funciona con un ciclo de 4 actividades

* Codificar:Transformar los requisitos en funcionalidades tangibles mediante la escritura de código limpio y eficiente.
* Diseñar:Mantener un diseño de software simple y flexible que facilite futuras modificaciones y expansiones.
* Probar:Realizar pruebas unitarias y de integración de manera constante para detectar y corregir errores tempranamente.
* Escuchar:Comprender a fondo las necesidades cambiantes del cliente y ajustar el desarrollo en función de su feedback

**BENEFICIOS**

* Mayor Calidad:Las pruebas continuas y las revisiones de código aseguran un software robusto y fiable.
* Alta Flexibilidad:El modelo se adapta con facilidad a los cambios en los requisitos sin afectar el ritmo del proyecto.
* Mejor Colaboración:Fomenta una comunicación fluida, reduciendo errores y malentendidos dentro del equipo.
* Ahorro:Minimiza retrabajos costosos y tardíos, optimizando el tiempo y los recursos invertidos.
* Satisfacción:Resultados que satisfacen tanto al equipo de desarrollo como al cliente final.

**METODOLOGIA DE PROTOTIPOS**

Un modelo de prototipo es el proceso de crear, probar y desarrollar una réplica funcional de un producto o servicio. En este caso, nos referimos a software web o de aplicaciones móviles . Es popular y uno de los modelos de Ciclo de Vida de Desarrollo de Software (SDLC) más utilizados . En este proceso, se crea una versión aceptable del software, se obtienen comentarios de los clientes y se perfecciona hasta que sea satisfactoria

1. Descarte rápido

2. Evolutivo

3. Incremental

4. Extremo

**VENTAJAS**

* Comentarios y participación: Los usuarios pueden opinar desde el inicio, lo que mejora el software y su satisfacción.
* Reutilizable: Los prototipos pueden aprovecharse en otros proyectos.
* Corrección temprana: Se detectan y solucionan errores antes del producto final.
* Mejor comprensión: Los usuarios aprenden sobre el desarrollo del software al participar activamente.
* Modelo sencillo: Es fácil de entender y no requiere gran experiencia técnica.

**DESVENTAJAS**:

* Requiere tiempo: El proceso es lento por las pruebas y los ajustes continuos.
* Costoso: Se crean varias versiones que a veces no se usan, generando gastos.
* Difícil implementar todo: No siempre se pueden incluir todas las sugerencias de los usuarios.
* Dependencia excesiva: Confiar demasiado en los comentarios puede causar requisitos poco claros al inicio.

**METODOLOGÍA RAD**

La metodología RAD (desarrollo rápido de aplicaciones) tiene enfoque sgul para el desarrollo de software, prioriza la creación fugaz de prototipos funcionales a través de iteraciones y la retroalimentación constante de los usuarios.

**OBJETIVOS**

* Asegurar la uniformidad y calidad del desarrollo y sistema
* Cumplir con las necesidades del usuario
* Tener un buen rendimiento y eficiencia en su trabajo
* Unificar criterios en la organización para el desarrollo del l sostén la de información
* Proporcionar puntos de control y revisión
* Ayudar a identificar lo antes posible cualquier cambio que se tenga que realizar dentro del desarrollo

**SUS 5 FASES**

1. Modelado de negocios
2. Modelado de datos
3. Modelado de procesos
4. Generación de aplicaciones
5. Pruebas y revisión

**BENEFICIOS**

* **los ciclos de desarrollo son más cortos de lo normal**
* El producto se crea en base a los comentarios del usuario
* Utiliza un enfoque flexible para adaptarse a los requisito cambiantes
* Los bloques de código reutilizables permiten realiza r pruebas más rápidas y reducir los defectos

**DESVENTAJASr**

* Requiere equipos muy capacitados y con experiencia.
* Proyectos grandes con muchos interesados pueden tener problemas de colaboración.
* Solo funciona bien en sistemas que se pueden dividir en módulos.
* Los requisitos del usuario deben estar claramente definidos durante todo el proyecto.
* Puede ser costoso por el modelado y generación automática de código.
* Es mejor para proyectos de corto plazo; en proyectos largos no es tan efectivo.

**METODOLOGÍA SCRUM**

**ROLES**

* product owner define y prioriza el trabajo
* Scrum Master quita obstáculos del objetivo
* Equipo de desarrollo trabaja en el proyecto

**SPRINTS**

* Sprint interacción 1 a 4 semanas de trabajo
* Sprint plannig planear el trabajo a realizar
* Sprint daily reunión diaria para acomodar roles
* Sprint rewiew ver el trabajo finalizado hacer retroalimentación
* Sprint retrospective buscar mejorías