

## Conversación Exportada

Usuario: Explícame esto de la metodología Blue Watch.

Asistente: La metodología Blue WATCH describe un modelo de procesos que organiza y guía los pasos técnicos, de gestión y soporte necesarios para desarrollar aplicaciones empresariales. El propósito de este modelo es proporcionar al equipo de desarrollo un marco que explique cómo llevar a cabo dichos procesos.

Características clave del modelo:

### 1. Iterativo:

- Los procesos se ejecutan en ciclos repetitivos.
- Cada iteración busca acercarse más al resultado final deseado, mejorando y ajustando constantemente.

### 2. Versionado:

- El desarrollo de la aplicación se realiza en varias versiones o etapas.
- Cada versión es un paso dentro de un ciclo de desarrollo, permitiendo mejorar gradualmente.

### 3. Incremental:

- Los programas o funcionalidades se desarrollan y entregan de forma parcial o en pequeños incrementos.
- Esto facilita que el cliente reciba y valide componentes del producto de forma gradual.

Usuario: Explícame eso de iterativo, versionado e incremental como para niños.

Asistente: Claro! Imagina que estás construyendo un castillo de bloques de juguete. La metodología iterativa, versionada e incremental funciona así:

Iterativo:

Es como si armas un castillo, lo miras y piensas: "Mmm, le falta algo". Entonces lo desmontas un poco, lo mejoras y vuelves a intentarlo, una y otra vez, hasta que quede como te gusta. Es hacer y mejorar varias veces!

Versionado:

Piensa que haces el castillo por partes. Primero armas una torre pequeña y dices: "Esta es la versión 1". Luego agregas otra torre y llamas a eso la "versión 2". Cada vez es una versión nueva y mejor que la anterior.

Incremental:

Aquí vas construyendo el castillo poquito a poquito. Primero una torre, luego las paredes, después el puente. Cada vez entregas algo nuevo que ya funciona, aunque el castillo no está completo al principio.

Así, al final, tienes un castillo increíble porque lo construiste probando, mejorando y entregando pedacitos tiles en cada paso.

Usuario: ¿Cuál es la diferencia entre cada una?

Asistente: ¡Claro! Las diferencias entre iterativo, versionado e incremental son fáciles de entender si seguimos con el ejemplo del castillo de bloques:

1. Iterativo:

Qué significa: Hacer algo, revisarlo, mejorarlo y volverlo a hacer hasta que quede bien.

Ejemplo: Construyes una torre, la revisas y dices: "No queda tan firme". Entonces la desmontas un poco, cambias los bloques y pruebas otra vez. Repites esto hasta que estés satisfecho con la torre.

Enfoque: Repetir para perfeccionar.

## 2. Versionado:

Qu significa: Crear algo en "etapas" completas, pero cada etapa tiene mejoras respecto a la anterior.

Ejemplo: En la versin 1 del castillo haces solo una torre. En la versin 2, agregas una muralla. En la versin 3, le pones un puente. Cada versin es funcional, pero mejor que la anterior.

Enfoque: Avanzar por etapas completas.

## 3. Incremental:

Qu significa: Construir por pedacitos pequenos, aadiendo poco a poco nuevas piezas que funcionan por separado.

Ejemplo: Primero armas una torre y solo esa funciona. Luego, aades una muralla que tambn tiene sentido por s sola. Despus, el puente. Cada parte se agrega sin necesidad de rehacer las anteriores.

Enfoque: Agregar funcionalidad poco a poco.

Diferencias clave:

Aspecto:

Iterativo: Repetir lo mismo, mejorndolo.

Versionado: Crear versiones completas y mejores.

Incremental: Agregar nuevas partes tiles.

Resultado final:

Iterativo: Se obtiene despus de varias pruebas.

Versionado: Se obtiene versin tras versin.

Incremental: Crece poco a poco con partes funcionales.

Usuario: Cul es la diferencia entre versionado e incremental? El incremento tiene varias versiones?

Asistente: Buena pregunta! A veces versionado e incremental pueden parecer similares, pero tienen diferencias clave en cmo se estructuran y entregan los avances.

Diferencia principal:

Versionado: Se enfoca en desarrollar versiones completas y funcionales del producto en etapas.

Cada versin es como un "paquete" cerrado que incluye varias mejoras o caractersticas.

Incremental: Se enfoca en construir el producto en pequenas partes o piezas funcionales.

Cada incremento es independiente y funciona por s solo.

El incremento puede tener varias versiones?

S, un incremento puede mejorarse a lo largo del tiempo con varias versiones. Por ejemplo:

El primer incremento es una torre simple (versin 1).

Luego haces una versin 2 de la misma torre para que sea ms alta o ms resistente.

Usuario: Ahora explcame esto. [Texto adicional sobre Blue WATCH].

Asistente: El texto describe el modelo Blue WATCH con una metfora del funcionamiento de un reloj de pulsera para explicar cmo avanza el desarrollo de una aplicacin.

Metfora del reloj:

El reloj tiene un motor que mueve las horas, los minutos y los segundos:

Avanzar las horas sera como un ciclo de versin completo, un avance grande y significativo.

Avanzar los minutos representa un ciclo ms pequeo, dentro del ciclo de las horas.

Avanzar los segundos simboliza las iteraciones ms pequenas y constantes dentro de los ciclos

anteriores.

Ciclo del incremento:

Es una pieza funcional de código que se desarrolla en un tiempo corto (1-3 semanas).

Cada incremento incluye algunas funciones de la aplicación y se enfoca en un subconjunto pequeño.

Ciclo de la versión:

Una versión es más grande que un incremento. Integra varios incrementos en un conjunto funcional y operativo.

Es una etapa del producto que ya puede usarse completamente en la plataforma del usuario.

Evolución del producto:

A lo largo del tiempo, se van entregando versiones. Cada versión agrega más funcionalidades al producto final.