**Desenvolvimento Iterativo e Incremental**

O Scrum é baseado no desenvolvimento **iterativo** e **incremental**.

A entrega é por parte no scrum, a iteração dentro de uma sprint é baseado nos processos, requisitos, planejamento e design, desenvolvimento e testes, feedback e integração.

Ao final desta iteração iremos para próxima sprint que é incremental a sprint anterior.

Lembrando que os pilares transparência, inspeção e adaptação fazem parte do scrum a cada iteração o time deve ficar melhor tendo em vista a execução repetitiva do processo e a aplicação dos pilares para o melhor desenvolvimento.

**Incremental vs. Sequencial**

O desenvolvimento de software pode ser baseado em dos tipos de metodologias para realizações de projetos.

**Sequencial**

O projeto é desenvolvido em uma sequência linear, com cada etapa sendo concluída antes da próxima começar.

**Vantagens:**

* É mais fácil de planejar e gerenciar do que o desenvolvimento incremental.
* Requer menos comunicação entre a equipe de desenvolvimento e o cliente.
* É mais fácil documentar o processo de desenvolvimento.

**Desvantagens:**

* O cliente só pode fornecer feedback após o projeto ter sido concluído.
* É mais difícil de adaptar o projeto às mudanças nas necessidades.
* O cliente só pode começar a usar o produto após o projeto ter sido concluído.



**Incremental**

O projeto é dividido em partes menores, chamadas de incrementos, que são desenvolvidos e entregues de forma iterativa.

**Vantagens:**

* Permite um feedback mais frequente do cliente e a adaptação do projeto às mudanças nas necessidades.
* Reduz o risco de erros e problemas, pois cada incremento é testado antes de passar para o próximo.
* Permite que o cliente comece a usar o produto mais cedo, mesmo que ainda não esteja completo.

**Desvantagens:**

* Pode ser mais difícil de planejar e gerenciar do que o desenvolvimento sequencial.
* Requer uma comunicação mais frequente entre a equipe de desenvolvimento e o cliente.
* Pode ser mais difícil de documentar o processo de desenvolvimento.



A escolha entre o desenvolvimento incremental e o desenvolvimento sequencial depende de uma série de fatores, como a natureza do projeto, as necessidades do cliente e a experiência da equipe de desenvolvimento.

Os dois tipos de desenvolvimentos podem ter o mesmo tempo de execução de entrega, porém o feedback constante no desenvolvimento incremental pode ser mais vantajoso e o desenvolvimento pode sofrer maiores intervenções para melhorias e correções de bugs.

No incremental ainda é possível gerir melhor o prazo de entrega do projeto, tendo em vista a divisão das entregas através das sprints e a definição de entregas e a possibilidade de melhor “visualizar” se o prazo das entregas estão corretas, podendo desta forma ainda se necessário ser compensado através das entregas das próximas sprints.

Aqui estão alguns exemplos de quando cada abordagem pode ser mais adequada:

**Desenvolvimento incremental**

* Desenvolvimento de software com requisitos que podem mudar ao longo do tempo.
* Desenvolvimento de produtos com um ciclo de vida curto.
* Desenvolvimento de produtos com um alto grau de incerteza.

**Desenvolvimento sequencial:**

* Desenvolvimento de produtos com requisitos bem definidos.
* Desenvolvimento de produtos com um ciclo de vida longo.
* Desenvolvimento de produtos com um baixo grau de incerteza.

**Processos Preditivos e Empíricos**

Os processos de desenvolvimento de software podem ser divididos em duas categorias principais: preditivos e empíricos.

**Processos preditivos**

Baseiam-se na estimativa e no planejamento de todo o trabalho a ser feito antes do início do desenvolvimento. Assumem que os requisitos são bem conhecidos e que não mudarão significativamente ao longo do projeto.



**Exemplos:**

* Cascata
* V-Model

**Processos empíricos**

Baseiam-se no desenvolvimento iterativo e incremental do software, com feedback frequente do cliente. Assumem que os requisitos são incompletos e que podem mudar ao longo do projeto.



Sequência pequenas e definidas de especificação e desenvolvimento.

**Exemplos:**

* Scrum
* Kanban

Diferenças entre os processos:

| **Característica** | **Processos Preditivos** | **Processos Empíricos** |
| --- | --- | --- |
| **Foco** | Planejamento e controle | Adaptabilidade e mudança |
| **Requisitos** | Completos e estáveis | Incompletos e mutáveis |
| **Abordagem** | Sequencial | Iterativa |
| **Feedback** | No final do projeto | Frequente |
| **Risco** | Maior | Menor |
| **Custo** | Menor | Maior |
| **Complexidade** | Menor | Maior |
|  |  |  |

Aqui estão alguns exemplos de quando cada abordagem pode ser mais adequada:

**Processos preditivos:**

* Projetos com requisitos bem definidos e estáveis.
* Projetos com baixo risco de mudança.
* Projetos com prazos e orçamentos apertados.

**Processos empíricos:**

* Projetos com requisitos incompletos ou mutáveis.
* Projetos com alto risco de mudança.
* Projetos que exigem flexibilidade e adaptabilidade.

No desenvolvimento as funcionalidades e os projetos:

* 35% dos requerimentos mudam ao longo do projeto;
* 65% das funcionalidades não são ou raramente são utilizadas.

Um protótipo navegável pode garantir que o projeto seja desenvolvido da forma que o cliente idealizou em sua cabeça o que ele deseja.