

Análise e Projetos de Sistemas

Prof. Ederson Luiz Silva

- **Metodologias de desenvolvimento de software tradicionais**
- Metodologias de desenvolvimento de software tradicionais referem-se a abordagens e práticas estabelecidas ao longo do tempo para desenvolver software. Essas metodologias são baseadas em processos estruturados e sequenciais que visam alcançar resultados eficazes e de alta qualidade. Elas fornecem diretrizes e etapas definidas para o desenvolvimento de software, desde a concepção até a entrega final

- **Existem várias metodologias de desenvolvimento de software tradicionais**
- Existem várias metodologias de desenvolvimento de software tradicionais, cada uma com suas próprias características e processos distintos. Alguns exemplos populares incluem o Modelo em Cascata, Modelo em V, Desenvolvimento Racional de Software (RUP) e o Modelo Espiral. Essas metodologias têm sido amplamente adotadas na indústria de tecnologia e têm sido utilizadas com sucesso em diversos projetos de desenvolvimento de software.

- **Outra característica importante dessas metodologias**
- Outra característica importante dessas metodologias é a ênfase na documentação detalhada. Durante todo o processo de desenvolvimento, é esperado que os desenvolvedores e equipes envolvidas criem documentação abrangente, incluindo especificações de requisitos, documentos de design, manuais de usuário e relatórios de teste. Essa documentação ajuda a garantir a rastreabilidade e facilita a manutenção do software no futuro.

- **Importância de metodologias de desenvolvimento de software tradicionais na indústria de tecnologia**
- As metodologias de desenvolvimento de software tradicionais desempenham um papel fundamental na indústria de tecnologia. Elas fornecem uma base sólida e estruturada para o desenvolvimento de software, permitindo que as equipes de desenvolvimento sigam um conjunto de processos definidos e eficazes

- **Ao adotar uma metodologia de desenvolvimento de software tradicional, as empresas podem se beneficiar de várias maneiras:**
- **Padronização:** As metodologias tradicionais estabelecem um conjunto de diretrizes e processos padronizados que ajudam a garantir uma abordagem consistente e uniforme para o desenvolvimento de software em toda a organização.
- **Controle de qualidade:** Com as etapas e processos bem definidos, as metodologias tradicionais permitem um melhor controle de qualidade ao longo do ciclo de vida do desenvolvimento de software.

- Gerenciamento de projetos: Ao seguir uma metodologia tradicional, os projetos de desenvolvimento de software podem ser planejados e gerenciados de forma mais eficiente.
- Comunicação eficaz: As metodologias tradicionais também facilitam a comunicação eficaz entre as partes interessadas envolvidas no desenvolvimento de software.

- **Modelo de Cascata**

- É o modelo mais antigo, sendo o primeiro a ser conhecido na engenharia de software, e continua sendo implantado até hoje.
- Trata-se de um modelo sequencial, onde cada passo deve ser completado antes que o próximo seja iniciado. Os nomes de cada passo variam, mas basicamente inicia-se com a análise de requisitos, seguindo para a fase de desenho, codificação, implementação, testes e manutenção.

- Um problema neste modelo é quanto aos requisitos, por ser uma única etapa e estes mudarem constantemente. Foi verificada então a necessidade de dar uma realimentação aos requisitos.

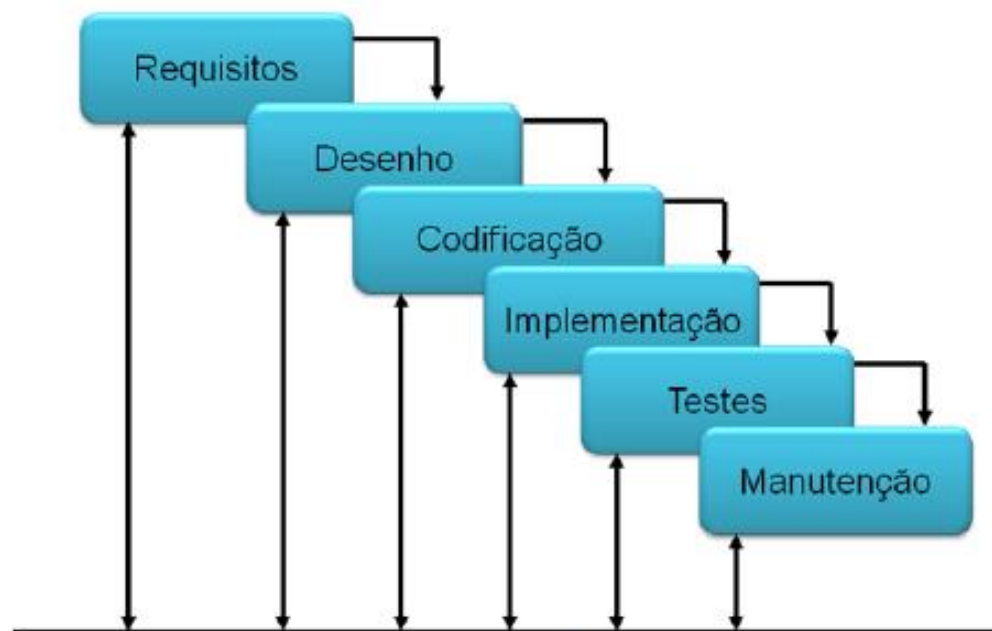


Figura 13: Modelo Cascata

- **Modelo em Espiral**

- É uma variação do modelo de Cascata. Quem introduziu este modelo foi Barry Boehm em 1988. A imagem do modelo é de uma espiral que começa no meio e continuamente revisa as tarefas básicas do usuário: comunicação, planejamento, análise de risco, engenharia, construção e implantação, e avaliação do usuário.

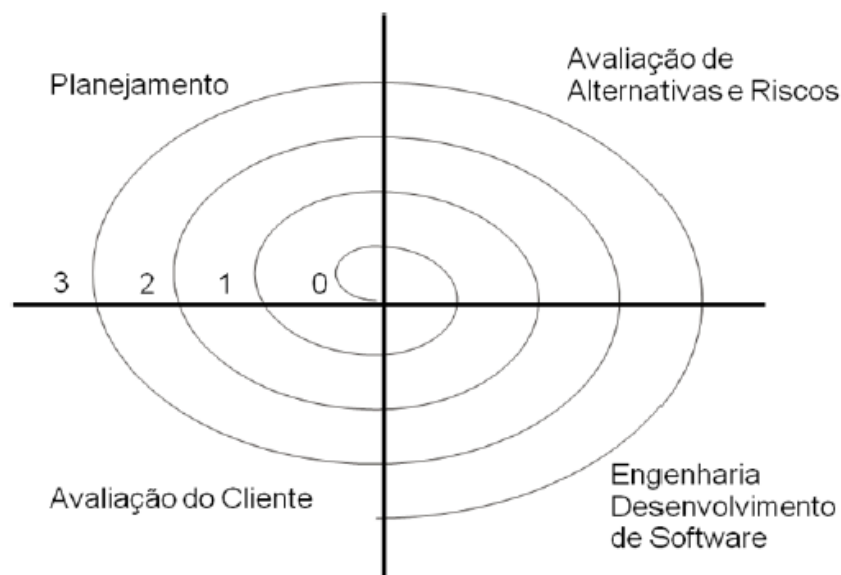


Figura 14: Modelo Espiral

- **Modelo de Prototipagem**

Esse modelo constrói uma versão descartável, ou seja, um protótipo. Esse protótipo vai testar conceitos e requisitos e será usado para demonstrar o comportamento aos clientes. Após a concordância dos clientes, o software é desenvolvido seguindo as mesmas fases do modelo de Cascata.

- O esforço despendido no protótipo geralmente é compensado pelo não desenvolvimento de características desnecessárias.



Figura 15: Modelo Prototipagem

- **Modelo Incremental**
- O objetivo era projetar e entregar ao cliente um conjunto mínimo do sistema que continuasse a ser um sistema usável. O processo continuará a interagir ao longo de todo o ciclo e vida com incrementos adicionais mínimos. As vantagens incluem fornecer logo ao cliente um sistema e novas versões de trabalho

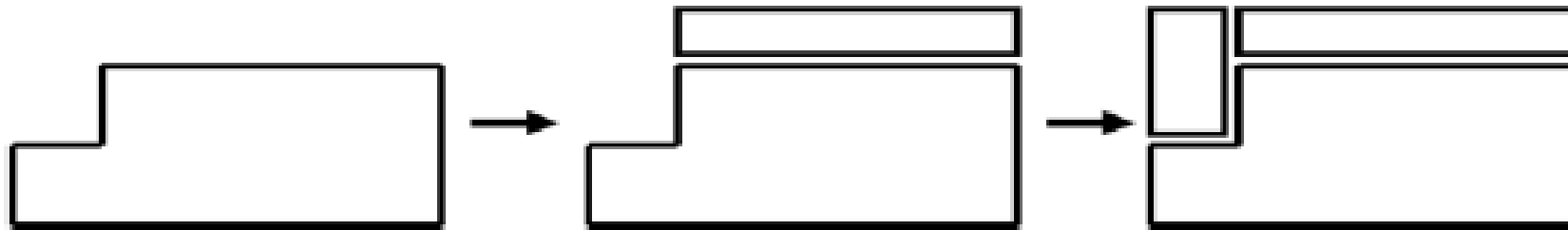


Figura 16: Modelo Incremental

1.4.5 Comparação dos modelos

<i>Modelo</i>	<i>Vantagens</i>	<i>Desvantagens</i>
Cascata	Minimiza o tempo de planejamento. Funciona bem para equipes tecnicamente mais fracas.	Inflexível. Torna-se difícil voltar atrás para corrigir erros.
Espiral	As interações iniciais são as mais baratas, permitindo que as tarefas de maior risco tenham custos reduzidos. Cada iteração da espiral pode ser customizada para as necessidades específicas de cada projeto.	É complexo e requer atenção e conhecimentos especiais para ser levado à risca.
Prototipagem	Os usuários conseguem ver os progressos. É útil quando os requisitos mudam rapidamente e o usuário está relutante em aceitar um conjunto de requisitos.	É impossível determinar com exatidão o tempo que o projeto vai demorar. Não há forma de saber o número de iterações que serão necessárias.
Incremental	É impossível determinar com exatidão o tempo que o projeto vai demorar. Não há forma de saber o número de iterações que serão necessárias.	Perigoso. Não há forma de assegurar qualidade e identificar os riscos. Falhas fundamentais não percebidas, imediatamente resultando em desperdício de trabalho.

- **Referências**
- <https://awari.com.br/metodologias-de-desenvolvimento-de-software-tradicionais-um-olhar-detalhado/>