



UNIVERSIDADE FEDERAL DO PIAUÍ – UFPI
CAMPUS SENADOR HELVÍDIO NUNES DE BARROS - CSHNB
CURSO DE BACHARELADO EM SISTEMAS DE INFORMAÇÃO



Engenharia de Software I

Engenharia e Ciclo de Vida do Software

Professora Pâmela Carvalho

06/12/22

Software

“Software é o produto que os engenheiros de software projetam e constroem. Abrangem programas que executam em computadores, de qualquer tamanho e arquitetura, documentos que incluem formas impressas e virtuais e dados que combinam números e texto, mas também incluem representações de informação em figuras, em vídeo e em áudio.” (Pressman)

Software

“Software são instruções que quando executadas fornecem a função e o desempenho desejados, estruturas de dados que permitem aos programas manipular adequadamente a informação e documentos que descrevem a operação e o uso dos programas” (Pressman)

Software

- Aplicações de Software.
 - Aplicável a qualquer situação para a qual um conjunto previamente especificado de procedimentos tenha sido definido.
 - **Determinação da informação:** previsibilidade da ordem e temporização da informação.
 - **Aplicações determinadas:** dados com ordem pré-definida, executa algoritmos de análise sem-interrupção e produz dados em forma de relatórios ou gráficos.
 - **Aplicações indeterminadas:** entradas com conteúdo variável e temporização arbitrária, algoritmos que podem ser interrompidos por forças externas e saídas que variam em função de ambiente e tempo.

Software

Aplicações de Software (tipos):

1. **Software de sistemas:** coleção de programas escritos para servir outros programas.
 - Caracterizado pela interação com o hardware do computador, uso de múltiplos usuários, operação concorrente, estruturas de dados complexas e interfaces externas múltiplas.
2. **Software de tempo real:** monitora / analisa / controla eventos do mundo real a medida que eles ocorrem.
 - Contem um componente de coleção de dados, um de análise, um de controle/saída e um de monitoração e controle de todos os outros.
3. **Software de tempo real:** apresentavam-se como sistemas discretos que evoluíram para sistemas de gestão de informação.
 - Reestruturam os dados existentes de modo a facilitar operações comerciais ou tomada de decisão de gestão de negócio.

Software

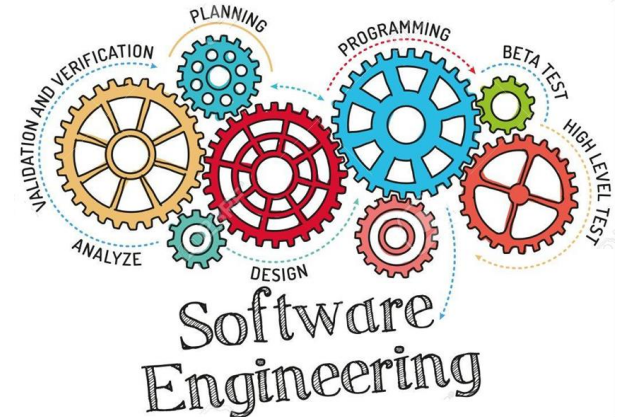
Aplicações de Software (tipos):

4. **Software científico e de engenharia:** caracterizado por algoritmos que processam números.
5. **Software embutido:** reside nas memórias ROM e é utilizado para controlar produtos e sistemas para o mercado consumidor e industrial.
6. **Software para Web:** a rede se transforma em um grande computador.
7. **Software para Inteligência Artificial:** utiliza algoritmos para resolver problemas complexos que não são passíveis de computação ou análise direta.
 - Representados inclusive por **sistemas especialistas, sistemas baseados em conhecimento e reconhecimento de padrões.**

Engenharia de Software

- “**Engenharia de Software** é a criação e utilização de sólidos princípios de engenharia a fim de obter software de maneira econômica, que seja confiável e que trabalhe eficientemente em máquinas reais.” (Fritz Bauer).
- É a “Aplicação de uma abordagem sistemática, disciplinada e quantificável, para o desenvolvimento, operação e manutenção do software; isto é, aplicação de engenharia de software.” (IEEE).

Engenharia de Software



- **Engenharia de Software?**

- É uma tecnologia em camadas.

- Tem seu foco na qualidade e melhoria contínua.
 - Processos mantêm unidas as camadas de tecnologia, define de áreas-chave de processo.
 - Métodos fornecem a técnica de como fazer para construir software.
 - Ferramentas de engenharia de software fornecem apoio automatizado ou semi-automatizado para o processos e para os métodos.

Engenharia de Software

- **Engenharia de Software?**

- ES é a análise, o projeto, a construção, a verificação e a gestão de elementos técnicos (ou sociais).
 - Qual o problema a ser resolvido?
 - Que características do elemento são usadas para resolver o problema?
 - Como o elemento e a solução serão realizados?
 - Como o elemento vai ser construído?
 - Que abordagem será usada para descobrir erros de projeto e construção?
 - Como o elemento será mantido a longo prazo?

Engenharia de Software

- **Engenharia de Software?**
 - Fases genéricas:
 - **Definição** (o que): aqui ocorrerá a identificação das necessidades que o software precisa atender. Três tarefas principais:
 - Engenharia de sistemas ou de informação.
 - Planejamento do projeto de software.
 - Análise de requisitos.

Engenharia de Software

- Engenharia de Software?

- Fases genéricas:

- **Desenvolvimento** (como): definição de como os dados devem ser estruturados, como as necessidades serão atendidas. Três tarefas principais:

- Projeto do software.

- Geração de código.

- Teste de software.



Engenharia de Software

- **Engenharia de Software?**

- Fases genéricas:

- **Manutenção:** modificações associadas com a correção de erros, adaptações necessárias, e modificação de melhorias.
 - Correção: gera uma manutenção corretiva que altera o software para corrigir defeitos.
 - Adaptação: gera manutenção adaptativa que altera o software para acomodar mudanças no seu ambiente externo.
 - Aperfeiçoamento: gera manutenção perfectiva que aprimora o software além dos requisitos originais.
 - Prevenção: gera a manutenção preventiva (reengenharia de software) que faz modificações de modo que os programas possam ser mais facilmente corrigidos, adaptados e melhorados.

Engenharia de Software

- **Processo?**

- “Arcabouço para as tarefas que são necessárias para construir software de alta qualidade”. (Pressman).
- Depende do software que está sendo construído.
- Modelo de processo é escolhido com base na natureza do projeto e da aplicação, nos métodos e ferramentas a serem usados, e nos controles e nos produtos intermediários e finais que são requeridos. Descreve os processos que devem ser realizados para o desenvolvimento de um software.

Engenharia de Software

- **Gestão de Projetos?**

- “A gestão do projeto envolve o planejamento, a monitoração e o controle do pessoal processo e eventos que ocorrem à medida que o Software evolui de um conceito preliminar para uma implementação operacional.”

Engenharia de Software

- **Qualidade?**

- “Fazer a coisa certa no momento certo e do jeito certo.”

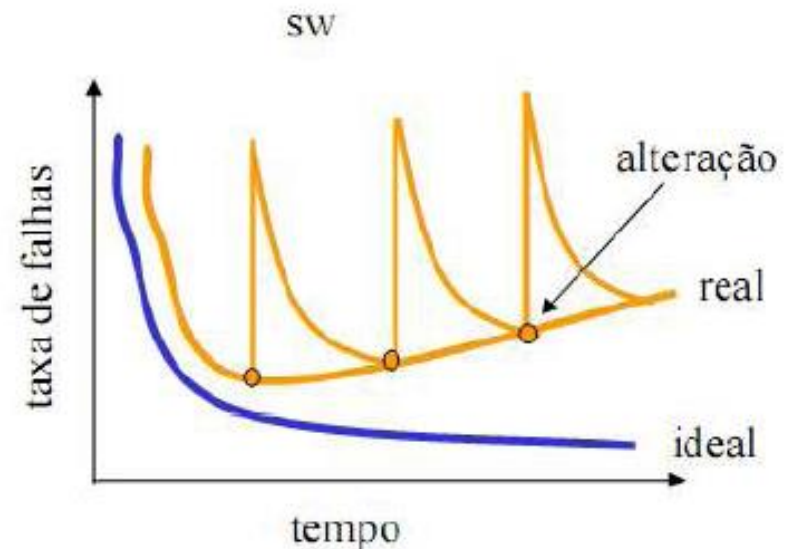


Engenharia de Software

- **Importância do Software:**
 - Um dos pontos fundamentais da importância do software é pelo seu uso cotidiano, aonde praticamente no mundo moderno, inexiste a possibilidade de não utilizá-lo.
 - Outro ponto é pela manipulação da informação (dado - informação - conhecimento), e quem a tem possui poder.

Engenharia de Software

- **Características do software:**
 - É desenvolvido/projetado por engenharia, não é fabricado.
 - Não se “desgasta”, mas deteriora.



Atributos de um bom Software

- Os atributos de um bom software refletem seu comportamento quando em funcionamento, a estrutura e a organização do programa fonte, e também a documentação associada (Sommerville).
- Como exemplos temos o tempo de resposta do software à consulta de um usuário e a facilidade de compreensão do código do programa.

Atributos de um bom Software

ATRIBUTOS	Descrição
Facilidade de Manutenção	O software deve ser escrito de modo que possa evoluir para atender às necessidades mutáveis dos clientes. Esse é um atributo crucial, porque as modificações em um software são uma consequência inevitável de um ambiente de negócios em constante mutação.
Nível de Confiança	O nível de confiança do software tem uma gama de características que incluem confiabilidade, proteção e segurança. O software confiável não deve ocasionar danos físicos ou econômicos, no caso de um defeito no sistema.
Eficiência	O software não deve desperdiçar os recursos do sistema, como memória e ciclos do processador. A eficiência, portanto, inclui a rapidez de resposta, o tempo de processamento, a utilização da memória, entre outros.
Facilidade de Uso	O software deve ser utilizável, sem esforços indevidos, pelo tipo de usuário para quem foi projetado. Isso significa que ele deve dispor de uma interface apropriada com o usuário e de documentação adequada.

Problemas Enfrentados pela Engenharia de Software

- Identificar adequadamente os requisitos do Sistema, ou seja, saber o que o software deve fazer;
- Que ferramentas, linguagem, sistema operacional usar;
- Como diminuir os tempos e custos de desenvolvimento;
- Prever falhas antes da entrega final;
- Como fazer manutenção e controlar versões;
- Inexistência de histórico, ou documentação, no desenvolvimento de Sistemas;
- Comunicação com os usuários precária;
- Conceitos quantitativos inexistentes tais como confiabilidade, qualidade e reusabilidade.

Desafios da Engenharia de Software

- Atualmente os principais desafios da Engenharia de Software, conforme Sommerville, são:

DESAFIOS	Descrição
O desafio do legado	Ainda os grandes sistemas de software existentes foram desenvolvidos no passado, e com importantes funções corporativas. O desafio é fazer a manutenção e atualização desses softwares à custos baixos e com qualidade.
O desafio da heterogeneidade	Os sistemas exigem em ambientes distribuídos por redes de diferentes tipos de computadores e sistemas de apoio. O desafio é desenvolver técnicas para construir softwares flexíveis para lidar com a heterogeneidade.
O desafio do fornecimento	Nos dias atuais existe uma demanda enorme de sistemas que sejam desenvolvidos no menor tempo possível e com facilidade de adaptação. O desafio é fornecer sistemas cada vez maiores e complexos com a qualidade desejada, e em curto espaço de tempo.

Ciclo de Vida do Software

Ciclo de Vida do Software

- Quando pensamos no desenvolvimento de um software queremos ir logo para a parte do desenvolvimento em si. Porém, certas etapas são importantes de serem realizadas antes de colocar a mão na massa.



Ciclo de Vida do Software

- Ciclo de vida do software é o termo utilizado para definir o **conjunto de etapas que ocorrem entre a concepção de um sistema e o instante em que ele é descontinuado** pelo desenvolvedor.
- Ele ajuda a orientar a equipe de desenvolvedores, assim como o direcionamento de recursos.

Ciclo de Vida do Software

- A gestão do ciclo de vida do sistema é importante por permitir ao negócio ter um planejamento inteligente e capaz de identificar quando é a melhor hora de executar cada tarefa que envolve a criação e a manutenção de um sistema.
- Ou seja, torna o processo de gestão do aplicativo mais robusto e organizado. Assim, o time pode ter maior controle sobre a aplicação e evitar cenários de risco.

Ciclo de Vida do Software

Existem 3 etapas básicas de um ciclo de software:

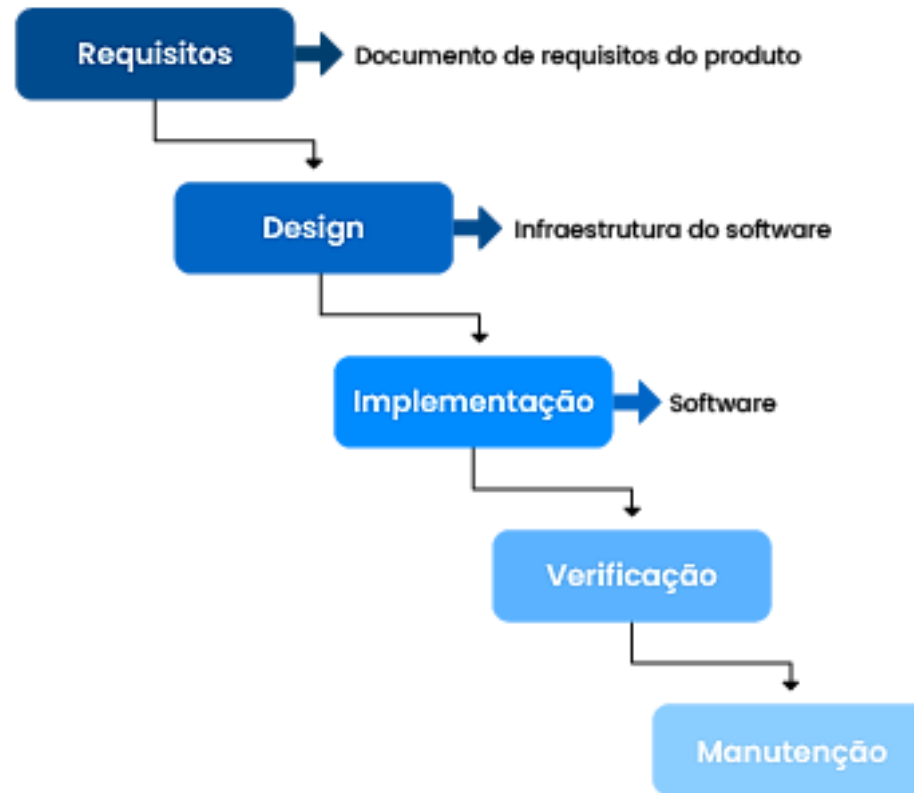
1. **Definição:** Identificação do problema para buscar uma resolução do mesmo. É na definição que você fará a modelagem dos processos e a análise do sistema.
2. **Desenvolvimento:** Envolve atividades relacionadas a design, prototipagem, codificação, testes, entre outras atividades que forem necessárias, como por exemplo, a integração com um outro sistema.
3. **Operação:** O software já estará em produção e você dará o devido suporte aos usuários e, claro, corrigirá possíveis bugs que possam aparecer.

Ciclo de Vida do Software

De forma geral, essas 3 etapas dividem-se em 5 fases principais em qualquer um dos modelos de desenvolvimento adotados:

1. **Fase de requisitos:** levanta os requisitos mínimos, estuda a viabilidade e define o modelo a ser usado;
2. **Fase de projeto:** Envolve atividades de concepção, especificação, design da interface, prototipação, design da arquitetura;
3. **Fase de implementação:** tradução para uma linguagem de programação das funcionalidades definidas durante as fases anteriores;
4. **Fase de testes:** realização de testes no que foi desenvolvido de acordo com os requisitos;
5. **Fase de produção:** implantação em produção do produto final;

Ciclo de Vida do Software



O ciclo de vida de um software é uma estrutura que indica **processos e atividades** envolvidas no desenvolvimento, operação e manutenção de um software, **abrangendo de fato toda a vida do sistema**.

Ciclo de Vida do Software

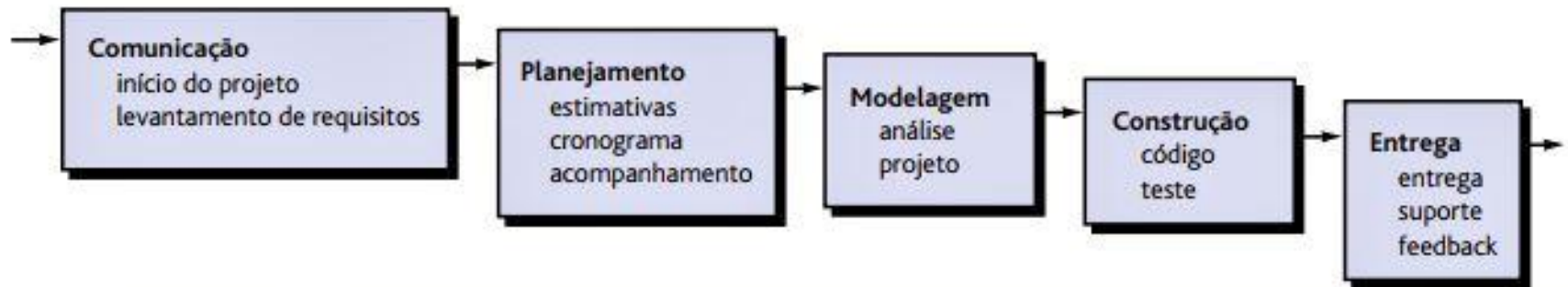


FIGURA 4.1 O modelo cascata.

(Pressman, 2016)

O Modelo Cascata é considerado o ciclo de vida clássico e sugere uma abordagem sequencial.

Ciclo de Vida do Software

- Em resumo, muitas são as metodologias de desenvolvimento de softwares.
- Existem as clássicas (antigas), que são mais estáveis de serem executadas através de diversos ciclos de vida prescritivo, seguindo um único caminho de trabalho, e aquelas metodologias ágeis, que possuem diversas formas dinâmicas de execução, exigindo maior experiência dos envolvidos.

Até a próxima aula...