# SUMÁRIO

[**SUMÁRIO 1**](#_ygudko5r873u)

[**Engenharia de Software 2**](#_mvuizjjz9jgz)

[Objetivos gerais 2](#_v2qikn9n7ld8)

[**Livro Referência 3**](#_nmhr6wt8qz3c)

[**Hardware 3**](#_krk367m17ri5)

[**Software 3**](#_8g66ve2m03z2)

[Elementos de um software 3](#_wwuacb19vbjn)

[Instruções 3](#_7kqygbclj0bk)

[Estruturas de Dados 4](#_4kdebc5llr11)

[Documentos 4](#_gq4k7cj9dddr)

[Características de um Software 4](#_h08fb08ynr3a)

[Principais Características do Software 5](#_13a0fphwstts)

[**Tipo de software 5**](#_l0cev974pp98)

[Tipo de software - Categorias modernas 6](#_c1zguo85u7wc)

[**Curva de defeitos de hardware 6**](#_33ltfm18kt23)

[**Curva de defeitos do software 7**](#_v7toio0o6um)

# 

# Engenharia de Software

Engenharia de software é a disciplina tecnológica e gerencial preocupada com a produção sistemática e manutenção de produtos de software que são desenvolvidos e modificados no prazo estabelecido e dentro das estimativas de custo (Fairley, 1985). O objetivo da disciplina é ajudar o aluno a compreender e aplicar conhecimento científico para o projeto e construção de programas de computador e a documentação associada necessária para desenvolvê-los, operá-los e mantê-los (Boehm, 2011).

Esta disciplina ajudará o aluno a compreender o processo de engenharia de software, lhe possibilitará adquirir as habilidades e competências necessárias para entender os problemas das organizações, definir soluções tecnológicas, identificar e documentar as características de uma solução.

O processo de engenharia de software geralmente segue um ciclo de vida que inclui várias fases, como análise de requisitos, design, implementação, teste, manutenção e evolução. Cada fase tem seus próprios objetivos e atividades específicas para garantir que o software atenda aos requisitos do cliente, seja confiável, eficiente e fácil de manter.

Existem várias metodologias de desenvolvimento de software, como o modelo cascata, o modelo em espiral e as metodologias ágeis, como Scrum e Kanban. As metodologias ágeis têm ganhado destaque devido à sua abordagem flexível, colaborativa e iterativa, permitindo uma adaptação mais eficiente às mudanças nos requisitos durante o desenvolvimento.

Além disso, a engenharia de software também abrange práticas de garantia de qualidade, gerenciamento de configuração, gerenciamento de projetos e documentação adequada. A colaboração entre os membros da equipe, a comunicação eficaz e a adoção de boas práticas de codificação são fundamentais para o sucesso de projetos de engenharia de software.

No contexto atual, a engenharia de software enfrenta desafios constantes devido à rápida evolução das tecnologias, demandas crescentes por inovação e a necessidade de lidar com sistemas complexos. No entanto, a disciplina continua a ser crucial para o desenvolvimento bem-sucedido de software em diversas áreas, desde aplicações empresariais até sistemas embarcados e soluções de inteligência artificial.

Disciplina ministrada pelo Professor João de Deus Freire Júnior.

## Objetivos gerais

* Discernir Software e Hardware, bem como as suas características;
* Conhecer os diversos tipos de software;
* Ter ciência dos diversos problemas que caracterizam a crise do software;
* Conhecer os principais paradigmas da Engenharia de Software;
* Conhecer a essência dos principais processos de desenvolvimento de software;
* Conhecer técnicas e ferramentas eficazes para identificar e documentar as características de uma solução de software.

Antes de estudar a disciplina e o processo de Engenharia de Software, é muito importante compreender os principais elementos e características de um software. Cada elemento de um software é essencial para a operação, evolução e manutenção do software. Entender as características do software possibilita a compreensão de sua natureza peculiar, seu processo de fabricação e tempo de vida. Com esse conhecimento será possível ter um melhor aprendizado da Engenharia de Software.

# Livro Referência

PRESSMAN, R. S.(2011) Engenharia de Software: uma abordagem profissional. 7.ed. Porto Alegre: Bookman, 2016.

# Hardware

O hardware é toda a parte física de um computador ou outro dispositivo computacional. Ele permite que as instruções enviadas pelos programas sejam executadas, que os usuários interajam com as aplicações e, entre outras coisas, que dados e informações sejam armazenados. Exemplos de hardware são: mouse, teclado, fones de ouvido, monitores e etc. Devido a sua natureza física o hardware se desgasta ao longo do tempo o que o torna passível de reparação ou inutilização.

# Software

O software é a parte lógica que faz com que os computadores ou outros dispositivos computacionais funcionem. O software fornece instruções para o hardware de como ele deve funcionar e quais funções executar. Exemplos de software são: aplicativos para dispositivos móveis, sistemas operacionais, processadores de textos, jogos e diversos outros programas de computador. Pela sua natureza lógica, o software não se desgasta com o tempo, mas pode se deteriorar.

O software pode ser melhor definido pelas partes que o formam. Segue uma definição de software que segue essa lógica:

*Software consiste em: (1) instruções (programas de computador) que, quando executadas, fornecem características, funções e desempenho desejados; (2) estruturas de dados que possibilitam aos programas manipular informações adequadamente; e (3) informação descritiva, tanto na forma impressa como na virtual, descrevendo a operação e o uso dos programas. (PRESSMAN, 2011, p. 32 )*

Esta definição deixa claro os elementos que formam um software, que são: instruções, estrutura de dados e informação descritiva (documentação). Qualquer um desses elementos são essenciais para o funcionamento, evolução e manutenção do software.

O software é um produto produzido por profissionais especializados, ele pode dar suporte a operações computacionais, de negócios, médicas e etc. Ele pode ser disponibilizado gratuitamente para uso em massa, pode servir para divulgação de outros produtos, pode ser o diferencial competitivo de grandes corporações ou pode ser a única interação entre clientes e a empresa.

## Elementos de um software

Os principais elementos de um software são: instruções, estrutura de dados e documentos.

### Instruções

As instruções, quando executadas, produzem a função e desempenho desejados pelo software. As instruções dizem o que os dispositivos e máquinas devem fazer. O conjunto de instruções de um software é seu código-fonte. Este código pode ser escrito com uso de linguagem de programação como Java, Python e etc.

### Estruturas de Dados

As estruturas de dados possibilitam que os programas manipulem adequadamente os dados e produzam informações. As estruturas de dados armazenam dados de forma eficiente. Elas podem ser: filas, pilhas, árvores binárias, objetos relacionais e etc.

### Documentos

Os documentos descrevem a operação e uso do programa. A documentação do software detalha a estrutura do software, sua arquitetura, componentes, funcionamento e regras. A documentação bem elaborada é essencial para evolução e manutenção do software.

## Características de um Software

Software é desenvolvido ou passa por um processo de engenharia; ele não é fabricado no sentido clássico.

Tanto software como hardware requerem a construção de um produto, porém, seus métodos de fabricação são totalmente diferentes. Ambas as atividades são dependentes de pessoas, mas a relação entre pessoas envolvidas e trabalho realizado é completamente diferente nesses produtos. Os custos de software concentram-se no processo de engenharia. Isso significa que projetos de software não podem ser geridos como se fossem projetos de fabricação.

O Software não “se desgasta”, mas, se deteriora, à medida que o tempo passa, os componentes de um hardware sofrem os efeitos cumulativos de poeira, vibração, impactos, temperaturas extremas e vários outros males ambientais ele começa a desgastar-se.

O software não é suscetível aos males ambientais que fazem com que o hardware se desgaste. Portanto, ele não se desgasta. Mas, o ambiente de negócios, a tecnologia, os processos, as organizações, sociedades, leis, regras mudam e todos esses aspectos podem fazer com que o software fique obsoleto e se deteriore.

De acordo com Pressman (2011), outro aspecto de desgaste ilustra a diferença entre hardware e software. Quando um componente de hardware se desgasta, ele é substituído por uma peça de reposição. Não existem peças de reposição de software. Cada defeito de software indica um erro no projeto ou no processo pelo qual o projeto foi traduzido em código-fonte. Portanto, as tarefas de manutenção de software, que envolvem solicitações de mudanças, implicam em complexidade maior do que a de manutenção de hardware.

Embora a indústria caminhe para a construção com base em componentes, a maioria dos softwares continua a ser construída de forma personalizada (sob encomenda).

À medida que a disciplina de Engenharia de Software evolui, cada vez mais componentes reutilizáveis são criados. Muitas empresas inclusive comercializam componentes ou softwares prontos chamados “pacotes” para outras empresas. Desta forma, as organizações não precisam construir aplicações ou componentes de software já existentes no mercado, elas só precisam pagar pela aquisição ou uso dos serviços.

## Principais Características do Software

Intangibilidade: Diferentemente do hardware, o software não tem forma física. Ele é composto por código, dados que são armazenados eletronicamente e documentação.

Flexibilidade: Pode ser alterado e atualizado mais facilmente do que o hardware. Atualizações e correções de bugs podem ser distribuídas de maneira eficiente.

Dependência de Hardware: O software requer hardware para ser executado. Ele interage com o hardware para realizar tarefas específicas.

Abstração: Permite aos usuários interagir com o computador de maneira mais amigável, ocultando detalhes complexos do hardware.

Volatilidade: Pode ser facilmente modificado e atualizado, mas também pode ser perdido se não for armazenado adequadamente.

Linguagens de Programação: Os programas são escritos em linguagens de programação, que servem como meio de comunicação entre os desenvolvedores e o computador.

# Tipo de software

O software é a tecnologia única mais importante no cenário mundial. O software se tornou uma tecnologia indispensável para negócios, ciência e engenharia; ele viabilizou a criação de novas tecnologias (por exemplo, engenharia genética e nanotecnologia), a extensão de tecnologias existentes (por exemplo, telecomunicações) e a mudança radical nas tecnologias mais antigas (por exemplo, indústria gráfica).

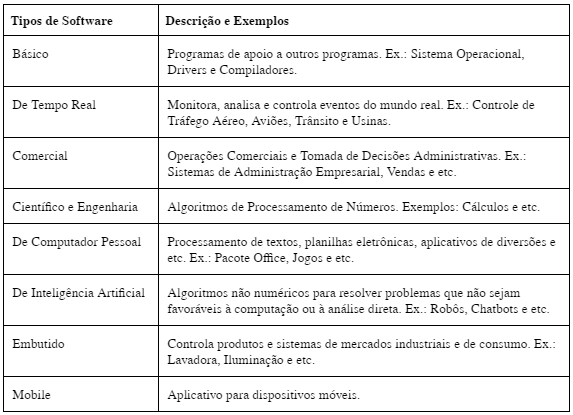
Se tornou a força motriz por trás da revolução do computador pessoal, já vemos pacotes de software sendo comprados pelos consumidores em lojas de bairro.

O software evoluiu lentamente de produto para serviço, na medida que empresas de software ofereceram funcionalidade imediata (just-in-time), via um navegador Web ou aplicativos móveis.

Companhias de software se tornaram as maiores e mais influentes companhias da era industrial criando a era digital, a Internet evoluiu e modificou tudo: de pesquisa em bibliotecas a compras feitas pelos consumidores, incluindo discurso político, hábitos de namoros de jovens e de adultos não tão jovens.

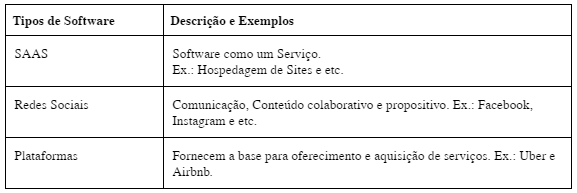
O software foi incorporado em sistemas de todas as áreas: transportes, medicina, telecomunicações, militar, industrial, entretenimento, máquinas de escritório e etc. Para atender toda essa demanda foram criados vários tipos de software.

Segue na tabela abaixo os tipos mais importantes:



## Tipo de software - Categorias modernas

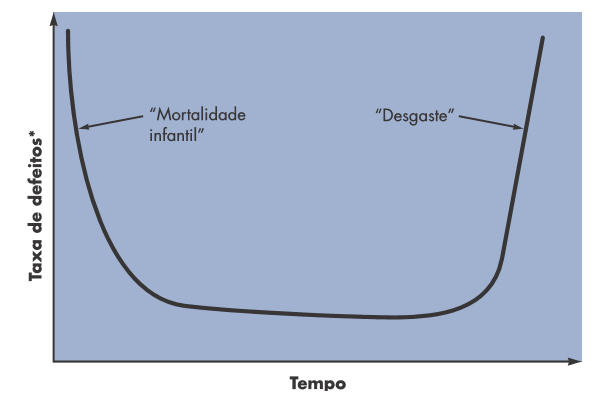
Com a evolução da tecnologia é possível observar novos tipos de software que, mesmo que possam ser classificados nas categorias apresentadas no tópico anterior, devem ser destacados separadamente também devido ao grande número de corporações e usuários que os utilizam. Segue na tabela abaixo esses tipos de software mais modernos:



# Curva de defeitos de hardware

A curva de defeitos de um hardware tem um comportamento bem diferente da curva de defeitos de um software. Ela está relacionada à natureza física do hardware e de muitos outros produtos físicos. O hardware apresenta uma alta taxa de defeitos no início de seu ciclo de vida, isso acontece até que o equipamento e seus componentes sejam ajustados corretamente a sua função e ambiente.

Após a correção desses defeitos e estabilização do produto, a taxa de defeitos do hardware se mantém estável e baixa até o fim de seu ciclo de vida quando é verificada uma alta taxa de defeitos novamente devido ao desgaste dos produtos por efeitos cumulativos de poeira, vibração, impactos e temperaturas extremas e vários outros males ambientais. Essa curva de defeitos do hardware é chamada de “curva da banheira”. Vejam na figura abaixo a curva de defeitos do hardware ao longo de seu ciclo de vida.



# Curva de defeitos do software

A curva de defeitos de um software é bastante peculiar e tem um comportamento bem diferente da curva de defeitos de um hardware. Ela também tem uma alta taxa de defeitos no início do ciclo de vida do software assim como o hardware, porém, ela não se estabiliza ao longo do tempo como a do hardware.

Diferente do hardware, o software sofre diversas mudanças durante seu ciclo de vida. Essas mudanças acontecem para corrigir defeitos identificados tardiamente, para desenvolver melhorias no software, adaptações e evoluções. A cada mudança, como efeito colateral, há um aumento na taxa de defeitos daquele software. O software é estabilizado, mas o processo se reinicia com uma nova mudança. Vejam na figura abaixo a curva de defeitos do software ao longo de seu ciclo de vida.

