

# TEXTO DE APOIO



## AULA 1

### Tópicos de Engenharia de Software

**Professor** Charles Boulhosa Rodamilans



Universidade Presbiteriana  
**Mackenzie**





# Sumário



<b>DESENVOLVIMENTO PROFISSIONAL DE SOFTWARE.....</b>	<b>3</b>
<b>INTRODUÇÃO.....</b>	<b>3</b>
<b>ENGENHARIA DE SOFTWARE .....</b>	<b>6</b>
<b>FUNDAMENTOS UNIVERSAIS .....</b>	<b>11</b>
<b>REFERÊNCIAS .....</b>	<b>12</b>

# **TÓPICOS DE ENGENHARIA DE SOFTWARE**

## **DESENVOLVIMENTO PROFISSIONAL DE SOFTWARE**

### **INTRODUÇÃO**

Este Texto de Apoio apresenta a Engenharia de Software, definindo diversos conceitos da área. Trataremos dos desafios do desenvolvimento de software e abordaremos, basicamente, os desafios da Engenharia de Software, entendendo porque ela é importante e porque a utilizamos.

Algumas perguntas são respondidas, como qual é a importância da Engenharia de Software? O que é um desenvolvimento de software profissional? Existem tipos diferentes de software? Existe uma forma única para desenvolver qualquer tipo de software? O contexto da disciplina começa a ser construído por meio do conteúdo deste material.

Hoje em dia, basicamente tudo que fazemos utiliza software. A sociedade moderna atual, sem software, passaria por dificuldades. Seu telefone, computador, o carro que você usa, o ônibus que você pega, tudo isso utiliza software.

A Engenharia de Software lida com essa ampla variedade de aplicações e possibilidades, em uma abordagem pragmática, que caracteriza as engenharias. Cada tipo de sistema de software requer uma abordagem de desenvolvimento diferente e a Engenharia de Software fornece subsídios para esses desenvolvimentos.

Quando pensamos no desenvolvimento de software, podemos ter pessoas que desenvolvem/programam o software por hobby. Mas estamos falando agora do desenvolvimento profissional de software, ou seja, estamos tratando do desenvolvimento por uma equipe de pessoas e queremos garantir que, ao final do trabalho dessa equipe, tenhamos um software funcional. Nosso desafio é construir um software profissional e, para isso, precisamos adotar uma abordagem sistemática da Engenharia de Software.

O software profissional não é composto apenas por programas de computador. Todo o conhecimento aplicado em seu projeto, como as várias partes que compõem o todo, toda a documentação que captura esse conhecimento, como dados e informação de configuração e testes, tudo isso também faz parte do software profissional. A Engenharia de Software se preocupará com todos os aspectos da produção de software, de tal forma a possibilitar a capacidade de produzir algo útil

para os usuários.

O desenvolvimento profissional de software deve considerar todos os aspectos da produção de software, estabelecendo um processo que inclui todas as atividades envolvidas no desenvolvimento do software. Esse processo de desenvolvimento incluirá todas as atividades envolvidas no desenvolvimento de software: desde o início, no qual temos que entender as necessidades de nosso usuário, até a entrega final de nosso software.

O que queremos identificar, na verdade, são algumas ideias fundamentais da Engenharia de Software que podemos aplicar, de forma universal, em todo o projeto de software. Algumas coisas que já sabemos serem importantes: definir o processo de software, trabalhar com os requisitos de confiança, proteção e reuso.

É muito importante termos em mente que, para os vários tipos de sistemas, também utilizamos diferentes ferramentas da Engenharia de Software. Não temos uma solução que resolve todos os problemas. Sendo assim, é necessário identificar quais são os fundamentos que se aplicam aos tipos de sistemas de software e, conseqüentemente, aos tipos de desenvolvimento de sistemas, pois para cada tipo de sistema, devemos utilizar ferramentas e técnicas da Engenharia de Software adequadas.

Além dos desafios da construção de um software, outro aspecto muito importante que também devemos considerar é a qualidade com relação ao uso do software que estamos desenvolvendo.

Sendo assim, podemos considerar as seguintes características no desenvolvimento profissional do software:

- preocupação com todos os aspectos da produção de software;
- o processo de software inclui todas as atividades envolvidas no desenvolvimento do software;
- ideias fundamentais da Engenharia de Software são universalmente aplicáveis para todos os tipos de desenvolvimento de sistemas (processos de software, confiança, proteção, requisitos e reuso);
- para os vários tipos de sistemas, devemos utilizar ferramentas e técnicas da Engenharia de Software adequadas;
- qualidade em uso do software deve ser considerada.

## PERGUNTAS QUE EXIGEM RESPOSTAS

Algumas perguntas são muito importantes para respondermos neste momento. São perguntas que devemos fazer e responder para iniciar um estudo sobre a engenharia de software e são apresentadas na Tabela 1.

**Tabela 1 – Perguntas Frequentes sobre Software**

Pergunta	Resposta
O que é software?	Software são programas de computador que escrevemos. Além disso, toda a documentação associada também faz parte do software.
Quais são os atributos de um bom software?	O primeiro grande atributo de um software é que precisa ser útil. O software precisa fazer aquilo que o usuário necessita. Além disso, precisa ser confiável, estar disponível quando o usuário solicitar e ser fácil de manter e utilizar.
O que é a Engenharia de Software?	É uma abordagem sistemática, utilizando o pragmatismo da engenharia, para entregar aquilo que foi prometido. Ela se preocupa com todos os aspectos da produção do software.
Quais são as principais atividades da Engenharia de Software?	Especificação, desenvolvimento, teste/validação e evolução do software.
Quais são os principais desafios da Engenharia de Software?	Atualmente, o principal desafio da Engenharia de Software é lidar com a diversidade de sistemas. Existem aplicativos em smartphones, softwares em pequenos sensores inteligentes e é necessário integrar esses vários tipos de sistema. Além disso, temos cada vez menos tempo para desenvolver uma solução de integração, além de o desenvolvimento de software precisar ser confiável.

Quais são os custos da Engenharia de Software?	Atualmente, o gasto médio do desenvolvimento de software é em torno de 60% do custo proveniente da parte de desenvolvimento/codificação e os outros 40% do custo, da parte de testes.  Em sistemas customizados, o custo proveniente da evolução do software é muito maior do que o custo do desenvolvimento.
Quais são as melhores técnicas e métodos da Engenharia de Software?	Aqui, vemos uma discussão quase ideológica e religiosa, na qual as pessoas defendem uma abordagem ou outra. O que precisamos entender é que não existe uma técnica melhor ou uma abordagem melhor. Existe a técnica/abordagem mais adequada para o contexto do problema apresentado.

Adaptado de: Sommerville (2011).

Outras questões: o que são produtos de software? Como eles são desenvolvidos e distribuídos?

Produtos de software são os softwares que podem ser vendidos ao cliente. Existem três tipos de produtos de software que podem ser desenvolvidos:

- produtos genéricos: a empresa desenvolve o software e disponibiliza para qualquer cliente que esteja interessado em adquirir. Ela é responsável por analisar, desenvolver e gerenciar a especificação. Exemplos: sistemas operacionais, bancos de dados, processadores de texto.
- produtos sob encomenda: um cliente solicita o desenvolvimento do software para outra empresa. A especificação é descrita pelo cliente e a empresa segue essa especificação. Exemplos: sistemas de apoio ao processo de negócio específico e sistemas de controle de tráfego aéreo.
- produtos genérico e sob encomenda (misto): os sistemas de softwares são desenvolvidos de forma genérica e adaptados para atender às necessidades dos clientes. Exemplos: sistemas de gestão empresarial (ERP) e sistemas de gestão de relacionamento com o cliente (CRM).

## ENGENHARIA DE SOFTWARE

A IEEE define a Engenharia de Software como:

[...] (1) A aplicação de uma abordagem sistemática, disciplinada e quantificável no desenvolvimento, na operação e na manutenção de software; isto é, a aplicação de engenharia ao software. (2) O estudo de abordagens como definido em (1). (IEEE, 1993).

Ressalta-se que uma “abordagem sistemática, disciplinada e quantificável” pode variar entre as equipes de desenvolvimento de software, a depender da agilidade e adaptabilidade necessária para atender as mudanças exigidas no contexto do problema (MAXIM; PRESSMAN, 2016).

A Engenharia de Software, segundo Sommerville (2011), também pode ser definida como:

[...] uma disciplina de engenharia cujo foco está em todos os aspectos da produção de software, desde os estágios iniciais da especificação do sistema até sua manutenção, quando o sistema já está sendo usado.

Esta definição está relacionada à engenharia, pois os engenheiros aplicam teorias e métodos para a resolução de problemas considerando os custos e prazos para a realização da solução. Ressalta-se que os engenheiros podem criar outras soluções quando as teorias e métodos não podem ser aplicados devido às restrições impostas. Segundo Sommerville (2011), “a Engenharia tem a ver com obter resultados de qualidade requeridos dentro do cronograma e do orçamento”.

Todos os aspectos de produção de software incluem os processos de desenvolvimento, as atividades para o gerenciamento do projeto de software e as teorias/métodos para dar suporte ao desenvolvimento do software.

**Figura 1 – Principais Atividades no Desenvolvimento de Software**



Fonte: Elaborada por autor

A utilização das atividades de forma sistêmica e sequenciada para a produção do software é denominada de processo de software. Conforme a Figura 1, as atividades fundamentais no desenvolvimento do software, comum a todos os processos de software, são:

- especificação: definição do que será produzido, dos objetivos e das restrições do software.
- desenvolvimento: projeto e programação do sistema de software de acordo



com as especificações determinadas.

- validação: verificar se o software desenvolvido atende às expectativas do cliente, com o intuito de garantir a satisfação do cliente e que suas necessidades foram realmente atendidas.
- evolução: o software é modificado para atender às mudanças de necessidades do cliente e para permanecer útil. Envolve a adição de novas funcionalidades e/ou correções do sistema.

## **DIVERSIDADE DE APLICAÇÕES**

A Engenharia de Software é uma abordagem sistemática, na qual buscamos estabelecer um processo claro de como produzir o software.

A Engenharia de Software precisa considerar algumas questões práticas. Temos que trabalhar com o custo e respeitar o prazo, além de haver confiança. E, também é importante atender às necessidades do cliente e de quem produz o software.

Como implementamos essa abordagem sistematizada pode variar de organização para organização. Você precisará avaliar como é a cultura da organização da empresa, de sua equipe, para, então, escolher a melhor abordagem para sua equipe e para o tipo de software que desenvolverá.

É importante reforçar que um dos principais desafios atuais da Engenharia de Software podem ser utilizadas. A Tabela 2 apresentará alguns exemplos de tipos de aplicações e sistemas de software.

Uma aplicação pode necessitar de levar em consideração as características de mais de um tipo. Como exemplo, um jogo para celular necessita de levar em consideração a usabilidade do usuário (sistemas de entretenimento) e precisa levar em consideração as restrições de execução do software no celular (sistema de controle embarcado), tais como, consumo de bateria, velocidade de rede e capacidade de processamento.

Cada tipo de sistema possui características diferentes e, por isso, necessita de técnicas de desenvolvimento de software diferentes. A seleção da técnica de desenvolvimento para um sistema de controle de tráfego aéreo ou sistema de controle de orientação de míssil deve levar em consideração a necessidade de especificações de software detalhadas antes da implementação do sistema. Em contrapartida, sistemas web podem ter os requisitos alterados com mais frequência, durante seu



desenvolvimento, para se adaptar às mudanças de necessidades do cliente e ao comportamento do mercado.

Software é a diversidade de tipos de soluções que temos que implementar, com diferentes níveis de confiabilidade, e também o tempo de entrega do software para o mercado que tem sido reduzido drasticamente.

É importante também reforçar que não é possível definir uma técnica para resolver tudo. O seguinte ditado: “Para quem tem um martelo, todo problema é prego”, não pode ser aplicado para a Engenharia de Software. Precisaremos procurar sempre a solução mais adequada. O tipo de software que será desenvolvido determina qual metodologia e quais técnicas da Engenharia de Software podem ser utilizadas. A Tabela 2 apresentará alguns exemplos de tipos de aplicações e sistemas de software.

Uma aplicação pode necessitar de levar em consideração as características de mais de um tipo. Como exemplo, um jogo para celular necessita de levar em consideração a usabilidade do usuário (sistemas de entretenimento) e precisa levar em consideração as restrições de execução do software no celular (sistema de controle embarcado), tais como, consumo de bateria, velocidade de rede e capacidade de processamento.

Cada tipo de sistema possui características diferentes e, por isso, necessita de técnicas de desenvolvimento de software diferentes. A seleção da técnica de desenvolvimento para um sistema de controle de tráfego aéreo ou sistema de controle de orientação de míssil deve levar em consideração a necessidade de especificações de software detalhadas antes da implementação do sistema. Em contrapartida, sistemas web podem ter os requisitos alterados com mais frequência, durante seu desenvolvimento, para se adaptar às mudanças de necessidades do cliente e ao comportamento do mercado.

**Tabela 2 – Exemplos de Tipos de Aplicações e Sistemas**

Aplicações/ Sistemas de Software	Descrição
Aplicações <i>stand-alone</i>	São aplicações executadas localmente (no computador ou dispositivo móvel) sem a necessidade de se conectar com a rede. Exemplos: programas de escritório, programas CAD, softwares de manipulação de imagem.

Aplicações interativas baseadas em transações	São aplicações executadas remotamente, sendo acessadas por dispositivos do próprio usuário (computador, celular e tablet). Muitas dessas aplicações são aplicações web que podem ser acessadas pelo navegador, sem precisar da instalação de outro software. Exemplos: sistemas de e-mail, sistemas de comércio eletrônico, sistemas de armazenamento de dados.
Sistemas de controle embarcados	São sistemas de controle de software para controlar e gerenciar dispositivos de hardware. Exemplos: softwares para celular, softwares que controlam o freio ABS, softwares que controlam o aquecimento do forno micro-ondas.
Sistemas de processamento em lotes ( <i>batches</i> )	São sistemas para processar dados em grandes lotes. Exemplos: faturamento periódico, tais como, contas de telefone e folhas de pagamento.
Sistemas de entretenimento	São sistemas para entreter o usuário, geralmente jogos, podendo ser executados em consoles específicos para essa finalidade.
Sistemas para modelagem e simulação	São sistemas desenvolvidos por cientistas e engenheiros para modelar processos físicos e interação entre objetos. Necessitam de um alto poder de processamento para realizar as modelagens e simulações. Exemplo: sistemas de interação entre moléculas para analisar o comportamento de um determinado remédio.
Sistemas para modelagem e simulação Sistemas de coleta de dados e análise	São sistemas que coletam dados em um ambiente e enviam esses dados para que possam ser analisados por outros sistemas. Exemplo: sensores meteorológicos espalhados na cidade para previsão do tempo.

Sistemas de sistemas	Sistemas de softwares compostos por outros sistemas de software. Um sistema pode ser dividido em sistemas menores (devido a sua complexidade) e/ou sistemas podem ser integrados para a criação de um novo sistema.
----------------------	---

Adaptado de: Sommerville (2011).

## FUNDAMENTOS UNIVERSAIS

Felizmente, temos alguns fundamentos que são universais para todos os processos. Se não tivéssemos, seria complicado ter uma abordagem sistematizada. Esses fundamentos são:

- **processo gerenciado e compreendido:** toda empresa precisa ter um processo de desenvolvimento que seja gerenciado e compreendido por todos que participam desse processo. Isso é importante porque se você tiver pessoas no desenvolvimento que não entendem o porquê delas estarem realizando determinadas tarefas, essas pessoas não trabalharão da melhor forma possível para atingir o que prometemos ao cliente;
- **confiança e desempenho:** precisamos entregar um software confiável e que tenha o desempenho esperado. Isso envolve um trabalho que vai além de um aspecto puramente funcional;
- **entender e gerenciar a especificação e os requisitos de software (o que o software deve fazer):** quando estamos fazendo a gestão de requisitos, estamos gerindo o que entregaremos para o usuário e para o cliente. Isso, na verdade, é uma gestão de expectativa. Logo, se não soubermos gerenciar as expectativas, é provável que nosso projeto não será executado dentro do prazo e custo;
- **melhor uso dos recursos existentes (reuso):** devemos priorizar o uso efetivo e eficiente dos recursos que temos. Sempre que possível, devemos tentar reutilizar soluções que já construímos e evitar construir tudo novamente.

## CONSIDERAÇÕES FINAIS

O objetivo deste primeiro Texto de Apoio é apresentar claramente onde estamos inseridos, qual é o papel da Engenharia de Software e qual é o grande desafio que temos pela frente.

## REFERÊNCIAS

IEEE *Standards Collection: Software Engineering*, IEEE Standard 610.12-1990, IEEE, 1993.

MAXIM, B. R.; PRESSMAN, R. S. *Engenharia de Software: uma abordagem profissional*. 8. ed. Porto Alegre: AMGH, 2016.

SOMMERVILLE, I. *Engenharia de Software*. 9. ed. São Paulo: Pearson Education do Brasil, 2011.