

# Análise e Modelagem de Sistemas

## Fundamentos da engenharia de software

Você sabia que seu material didático é interativo e multimídia? Isso significa que você pode interagir com o conteúdo de diversas formas, a qualquer hora e lugar. Na versão impressa, porém, alguns conteúdos interativos ficam desabilitados. Por essa razão, fique atento: sempre que possível, opte pela versão digital. Bons estudos!

Nesta webaula vamos ver os fundamentos da engenharia de software e entender a natureza e a evolução do software, a prática da engenharia de software, os princípios da análise de sistemas e a importância do papel do analista de sistemas no desenvolvimento de um software.

### Software

“

Softwares são programas de computadores com uma documentação associada e [que] os produtos de software podem ser desenvolvidos para um determinado cliente ou para um mercado mais generalizado.

— Sommerville (2011, p. 4).

”

Pressman (2016) afirma que um software de computador é um produto que profissionais da área da Tecnologia da Informação (TI) desenvolvem e para o qual darão suporte a longo prazo. Além disso, abrange qualquer tipo de mídia eletrônica, consistindo na união de três elementos:

Instruções	▼
Quando executadas, fornecem os atributos e funções de desempenhos desejados pelos usuários.	
Estrutura de dados	▼
Possibilitam aos programadores manipular as informações de forma mais adequada conforme a necessidade da aplicação.	
Documentação	▼
É toda a informação descritiva do software, a qual detalha a operação de uso dos programas, diagramas de funcionalidades, etc.	

[Saiba Mais](#)

### Evolução do software

A medida que novas tecnologias se tornam populares, maiores são as necessidades da produção de software para atender as demandas da sociedade.

#### Na década de 1990

Uma empresa precisava de um software e (talvez) de um site para que ficasse conhecida na internet; o site era utilizado como um cartão de visitas.

#### Na década de 2000

A mesma empresa precisava de um software com acesso à internet para disponibilizar recursos para seus clientes.
<a href="#">A partir de 2010</a> A empresa precisava de um software, um site e um aplicativo para que seus serviços pudessem ser acessados por diversos dispositivos e para que o armazenamento da base de dados estivesse na nuvem.
<a href="#">A partir de 2020</a> A mesma empresa já está pensando em utilizar a Inteligência Artificial para melhorar seus processos gerenciais.

As mudanças sempre ocorrerão ao longo do tempo de criação e uso de um software: durante o desenvolvimento, na fase da entrega e depois de entregue. Sempre há necessidade de ajustes e correções ou ainda pode ocorrer a necessidade de incluir novas funcionalidade ao software, as quais são, muitas vezes, requisitadas pelo cliente.

Além disso, o software passa por uma série de manutenções (isso acontece pois são realizadas novas solicitações do cliente) e, após realizar diversos ajustes (que podem levar a novos problemas), é possível que haja a necessidade de se criar um novo software.

### Prática da engenharia de software

De acordo com Sommerville (2011), a prática da engenharia de software se faz necessária porque cada vez mais a sociedade (e os indivíduos que a compõem) estão dependentes dos sistemas de software e, neste contexto, devemos produzir softwares mais confiáveis e de forma mais econômica, e a longo prazo é mais barato utilizar as técnicas da engenharia de software para evitar mudanças em algo que já tenha sido desenvolvido e principalmente que partes do software possam ser reutilizadas em outros sistemas.

Conforme Pressman (2016), a engenharia de software é uma tecnologia em quatro camadas que objetiva a disciplina, a adaptabilidade e a agilidade. As quatro camadas são:

Foco na qualidade	▼
É o objetivo final de toda a engenharia de software; toda a gestão de qualidade (das demais camadas) deve ser fundamentada em um comprometimento organizacional com a qualidade total de todas as etapas de desenvolvimento.	
Processo	▼
É a base da engenharia de software; o processo é a ligação entre as demais camadas; é o que mantém as camadas de tecnologia coesas, possibilitando o desenvolvimento de forma racional (e dentro dos prazos preestabelecidos) e é a base para o controle e gerenciamento de projetos de software. É nesta camada que são produzidos diversos artefatos (modelos, documentos, relatórios, formulários, etc.).	
Métodos	▼
Fornecem as informações técnicas para o desenvolvimento do software e envolvem uma grande quantidade de tarefas (comunicação, análise de requisitos, modelagem de projetos, codificação, testes e manutenção do software).	
Ferramentas	▼
Possibilitam um alicerce automatizado ou semi-automatizado para o processo e para os métodos; quando as ferramentas são integradas de modo que a informação criada possa ser usada por outra ferramenta, é estabelecido um suporte ao desenvolvimento de software, conhecido como engenharia de software auxiliado por computador.	

A engenharia de software, segundo Sommerville (2011), preocupa-se com todos os aspectos de produção do software.

Enquanto isso Pressman (2016) enfatiza que essa área engloba processos, métodos e ferramentas que possibilitam a construção de sistemas, incorporando cinco atividades específicas nesses processos: **comunicação, planejamento, modelagem, construção e entrega**.

Pressman (2016) destaca sete grandes categorias de softwares e desafia a constante evolução das práticas empregadas na engenharia de software:



**Softwares legados**

Segundo Pressman (2016), são softwares antigos que continuam sendo utilizados, apresentam baixa qualidade, muita demora e funcionalidades que não são mais utilizadas pela empresa ou que são muito defasadas. Um software legado pode ter uma documentação deficitária ou inexistente, fazendo com que o processo de manutenção seja caro e muito demorado. O processo de evolução desse tipo de software é a criação de um novo com tecnologias mais recentes.

## Aplicativos

Software projetado para funcionar em uma determinada plataforma móvel. Os aplicativos possuem uma interatividade maior com os usuários, apresentam recursos disponíveis pelo fornecedor da plataforma móvel (iOS, Android, etc.) e também recursos baseados na web, permitindo uma vasta opção de informação e compartilhamento. Uma vantagem dos aplicativos é o fornecimento de recursos de armazenamento persistente dentro da plataforma. Um aplicativo pode acessar diretamente ferramentas de hardware, disponíveis no dispositivo, trazendo novas opções de programação utilizando, por exemplo, a câmera ou o GPS do dispositivo.

# Princípios da análise de sistemas

Os princípios da análise de sistemas fundamentam-se na necessidade de realizar estudos de processos para encontrar a melhor solução para a criação de um sistema. Conforme Roth, Dennis e Wixom (2014), a análise de sistemas baseia-se em métodos e técnicas de investigação e em especificação para encontrar a melhor solução para algum problema ou necessidade computacional de determinada área de negócio, a partir das funcionalidades levantadas pelo analista de sistemas.

Em uma visão mais generalizada das fases que envolvem os processos da análise de sistema, destacam-se, conforme Pressman (2005):

Análise	▼
Nesta fase são realizados estudos que objetivam a especificação do software, de modo a verificar a viabilidade (custo-benefício), definir as funcionalidades que o software deverá possuir e realizar o escopo, alocando recursos e realizando o orçamento do software. O resultado desta fase será utilizado nas próximas etapas.	
Projeto	▼
Nesta etapa há uma preocupação com a definição lógica do software, são elaborados os layouts de telas e relatórios e são criados a estrutura de banco de dados e os diagramas gráficos para o desenvolvimento do software.	
Implementação	▼
Nesta fase é realizada a codificação do software por meio de uma linguagem de programação (definida na fase de análise).	
Testes	▼
Objetivando a procura de erros, nesta fase, são realizados procedimentos de testes que verificam as funcionalidades dos itens codificados.	
Documentação	▼
Trata-se de documentar todos os processos (de todas as fases) e diagramas produzidos; são utilizados documentos padronizados (e personalizados por cada empresa de desenvolvimento), que servem como ferramenta de comunicação entre as pessoas envolvidas no desenvolvimento e também como parte de contrato entre as partes interessadas na produção do software.	
Manutenção	▼
Esta fase consiste em fazer o acompanhamento do software após ser implantado e entrar em funcionamento (durante um período), visando a registrar e corrigir falhas, propor melhorias ou incluir novas funcionalidades.	

As fases apontadas por Pressman (2005), devem apresentar um processo de homologação para validar os documentos gerados. O cliente deve estar ciente de que uma vez aprovada uma fase, caso seja alterada, haverá mudanças nos custos e implicará no dimensionamento do tempo (atrasos).

# O papel do analista de sistemas

O analista de sistemas é o profissional responsável por realizar atividades da análise de sistema como: pesquisas, planejamentos, coordenação de equipes de desenvolvimento e recomendação de alternativas de software de acordo com as necessidades de desenvolvimento ou de solução para problemas de negócios.

Possui como tarefas a criação, a implementação e a implantação de um software, de maneira que deve, primeiro, descobrir o que um sistema deverá fazer e depois entender e avaliar as necessidades e expectativas de cada usuário do software, a fim de que estas sejam organizadas, especificadas e documentas.

Cabe ao analista de sistemas colher informações com os usuários que utilizarão o software, interpretar essas informações e repassá-las aos programadores de forma técnica para que seja criado um software que atenda as expectativas do cliente e dos usuários.



Fonte: Shutterstock.

O profissional desse ramo precisa conhecer um pouco de cada área de negócio e, caso não tenha o domínio necessário sobre algum tema, deve ter a proatividade de procurar o máximo de conhecimento sobre a área que o software irá abranger.

Uma característica importante do analista de sistemas é ter uma boa visão empresarial para ajudar nos processos gerenciais da produção do software. As habilidades desejáveis para um analista de sistema são: conhecimento tecnológico atualizado, organização e método, visão gerencial, ótimo relacionamento interpessoal, entre outras.

Até aqui vimos que um software precisa evoluir para não ficar obsoleto e a necessidade de incluir práticas de engenharia de software e princípios da análise de sistemas no desenvolvimento de um software. Além disso, vimos o papel do analista de sistemas em todo esse processo.

## Pesquise mais!

Para colaborar na ampliação de seus conhecimentos a respeito do desenvolvimento de um software, o artigo apresentado a seguir trata de um mecanismo de visualização da evolução de software capaz de ser integrado em um processo de desenvolvimento distribuído, apresentando uma ferramenta por meio da qual desenvolvedores espalhados geograficamente podem compreender a evolução de seus projetos e identificar, através de uma visão global do projeto, diferentes características relevantes do projeto.

CEPÊDA, R.; MURTA, L.; WERNER, Cláudia. **Visualizando a evolução de software no desenvolvimento distribuído.**