

PROJETO DE SOFTWARE

Angelita Moutin Segoria Gasparotto

INFORMAÇÃO E COMUNICAÇÃO

PROJETO DE SOFTWARE

Angelita Moutin Segoria Gasparotto

INFORMAÇÃO E COMUNICAÇÃO



Autora

Angelita Moutin Segoria Gasparotto

Bacharel em Tecnologia de Processamento de Dados e especialista em Educação pelo Centro Estadual de Educação Tecnológica Paula Souza. É mestre na área de Sistemas de Apoio à Decisão pelo Departamento de Engenharia de Produção da Escola de Engenharia de São Carlos (USP). Doutora na área de Processos e Gestão de Operações pelo Departamento de Engenharia de Produção, da Escola de Engenharia de São Carlos (USP). É docente da Faculdade de Tecnologia de Taquaritinga e de cursos de pós-graduação da Faculdade Anhanguerana de Matão. Atualmente, cursa pós-doutorado no Departamento de Economia, Administração e Educação UNESP de Jaboticabal. Suas áreas de interesse incluem gestão da qualidade, gestão tecnológica, modelagem de processos e redes colaborativas universidade-empresa.

Design Instrucional

Pedro Meneses
Sarah Resende

Projeto Gráfico

NT Editora

Revisão

Filipe Lopes
Ricardo Moura

Capa

NT Editora

Ilustração

Daniel Motta

Editoração Eletrônica

Danilo Oliveira

NT Editora, uma empresa do Grupo NT

SCS Quadra 2 – Bl. C – 4º andar – Ed. Cedro II

CEP 70.302-914 – Brasília – DF

Fone: (61) 3421-9200

sac@grupont.com.br

www.nteditora.com.br e www.grupont.com.br

Gasparotto, Angelita Moutin Segoria.

Projeto de Software / Angelita Moutin Segoria Gasparotto –
1. ed. reimpr. – Brasília: NT Editora, 2018.

210 p. il. ; 21,0 X 29,7 cm.

ISBN 978-85-8416-288-8

1. Software. 2. Desenvolvimento.

I. Título

Copyright © 2018 por NT Editora.

Nenhuma parte desta publicação poderá ser reproduzida por qualquer modo ou meio, seja eletrônico, fotográfico, mecânico ou outros, sem autorização prévia e escrita da NT Editora.

ÍCONES

Prezado(a) aluno(a),

Ao longo dos seus estudos, você encontrará alguns ícones na coluna lateral do material didático. A presença desses ícones o(a) ajudará a compreender melhor o conteúdo abordado e a fazer os exercícios propostos. Conheça os ícones logo abaixo:



Saiba mais

Esse ícone apontará para informações complementares sobre o assunto que você está estudando. Serão curiosidades, temas afins ou exemplos do cotidiano que o ajudarão a fixar o conteúdo estudado.



Importante

O conteúdo indicado com esse ícone tem bastante importância para seus estudos. Leia com atenção e, tendo dúvida, pergunte ao seu tutor.



Dicas

Esse ícone apresenta dicas de estudo.



Exercícios

Toda vez que você vir o ícone de exercícios, responda às questões propostas.



Exercícios

Ao final das lições, você deverá responder aos exercícios no seu livro.

Bons estudos!

Sumário

1 INTRODUÇÃO À ANÁLISE DE SISTEMAS	9
1.1 Teoria geral de sistemas: evolução e conceitos	9
1.2 Princípios de engenharia de <i>software</i>	14
1.3 Princípios de modelagem	17
1.4 Metodologias para desenvolvimento de sistemas	21
1.5 Ferramentas de engenharia de <i>software</i>	25
2 PROCESSO DE DESENVOLVIMENTO DE SOFTWARE.....	32
2.1 Modelos de ciclo de vida do <i>software</i>	32
2.2 Processos de desenvolvimento e garantia da qualidade de <i>software</i>	38
2.3 As etapas da engenharia de <i>software</i>	47
3 GERENCIAMENTO DE PROJETOS DE SOFTWARE	62
3.1 Atividades de gerenciamento	62
3.2 Cronograma de projeto	69
3.3 Gerenciamento de riscos	74
3.4 Noções de estimativas de custos	78
4 ENGENHARIA DE REQUISITOS	87
4.1 Requisitos de <i>software</i>	87
4.2 Documento de requisitos	90
4.3 Estudos de viabilidade	95
4.4 Obtenção, elicitação e análise de requisitos	96
4.5 Validação e gerenciamento de requisitos	101
5 MODELOS DE SISTEMAS	110
5.1 Modelos de contexto	110
5.2 Modelos de comportamento	117
5.3 Modelos de dados	121
5.4 Modelos de objetos	124
5.5 Métodos estruturados	129
6 PROTOTIPAÇÃO DE SOFTWARE	138
6.1 Definição e objetivos	138
6.2 Prototipação no processo de <i>software</i>	144
6.3 Técnicas e ferramentas de prototipação	148
6.4 Prototipação de interface com o usuário	151

7 PROJETO DE SOFTWARE	159
7.1 Projeto de arquitetura e projeto estruturado	159
7.2 Projeto orientado a objetos e projeto de tempo real	165
7.3 Projeto com reuso	171
7.4 Projeto de interfaces com o usuário.....	175
7.5 Projeto de sistemas críticos	179
8 DESENVOLVIMENTO ÁGIL DE PROJETOS DE SOFTWARE	185
8.1 Justificativas para o desenvolvimento ágil de projetos de <i>software</i>	185
8.2 Fases de desenvolvimento ágil	191
8.3 As equipes.....	197
8.4 O contexto da inovação no desenvolvimento ágil	201
GLOSSÁRIO.....	209
BIBLIOGRAFIA	210

Seja bem-vindo(a) ao **Projeto de Software!**

Deve ser de seu conhecimento que os empreendimentos, desde os pequenos até os maiores, requerem sistemas de qualidade. Isso significa que, atualmente, os sistemas devem operar com segurança, com agilidade e ser de fácil entendimento por aquele que os opera, desde um sistema para gerenciar a produção de uma indústria até um sistema para controlar uma aeronave que atravessa os continentes!

E é por isso que os estudos sobre Projeto de Software tornam-se tão importantes! Aproveite ao máximo todo este material, cujo conteúdo foi projetado especialmente para atender às expectativas de um mercado cada vez mais exigente.

Bons estudos!

Angelita Moutin Segoria Gasparotto

1 INTRODUÇÃO À ANÁLISE DE SISTEMAS

Nesta seção, estudaremos a evolução e os principais conceitos que envolvem o desenvolvimento de sistemas de informações. Conhecer esses conceitos e saber situá-los no tempo é imprescindível para futuros profissionais da área.

Objetivos

Ao finalizar esta lição, você deverá ser capaz de:

- explicar a teoria geral de sistemas, destacando seus principais conceitos;
- discutir os princípios de engenharia de *software* e de modelagem atuais;
- destacar as metodologias e as ferramentas para desenvolvimento de sistemas e seus impactos para a garantia da qualidade do *software* produzido.

1.1 Teoria geral de sistemas: evolução e conceitos

As evoluções tecnológicas vividas por nossa sociedade atualmente têm evidenciado o valor crescente da informação e provocado uma utilização em massa dos computadores e tecnologias envolvidas. O uso vem crescendo para todos os tipos de computadores, no entanto, os de pequeno porte são os mais visados pelo mercado, pois são utilizados para funções que vão desde o simples entretenimento, até as operações mais precisas e complexas.



Por isso, com o passar dos anos (e até meses!), a tecnologia se tornará uma ferramenta imprescindível, sendo caracterizada como o agente responsável pelo processo de transformação da sociedade da informação. (MEIRELLES, 1994).

Porém, o uso dos recursos da informática ainda continua pouco explorado, devido principalmente ao baixo grau de informatização que muitas empresas possuem e, ainda, em relação ao pouco uso da tecnologia nas decisões estratégicas e de tomada de decisões. (PRATES; OSPINA, 2004).



De acordo com Segoria (2001), as realizações e as atividades desenvolvidas pelo homem em qualquer segmento produtivo são geradas pela percepção de suas necessidades, pelo acompanhamento da evolução tecnológica e pela adequação aos cenários e aos comportamentos da sociedade, seguindo uma abordagem global.



Dicas

Questões relacionadas à indústria de software, hoje caracterizadas como competências qualificadoras para a competitividade, se tornarão um conjunto de exigências básicas para a sobrevivência de longo prazo no mercado internacional.

Essa realidade faz parte das empresas do mundo todo, principalmente da empresa brasileira, dada a meta de desenvolvimento e melhoria da qualidade de vida estabelecida para a sociedade.



Importante

Meirelles (1994) destaca apropriadamente que, para entender os ciclos de evolução da informática e seus impactos, é necessário perceber a rapidez com que as transformações vêm ocorrendo nas últimas décadas.

O autor apresenta as seguintes características:

- as transformações na indústria dos computadores têm sido extremamente rápidas;
- os custos decrescentes da tecnologia aumentam a faixa de aplicações economicamente viáveis;
- as capacidades de processamento crescente em conjunto com sistemas de nível cada vez maior aumentam as aplicações tecnicamente viáveis.



- os impactos sobre as empresas e sobre as pessoas são muito grandes e podem provocar efeitos positivos, efeitos negativos e mudança no perfil e na estrutura da mão de obra.
 - » Efeitos positivos: aumento das informações disponíveis, redução da execução das tarefas, redução dos custos, aumento da produtividade e satisfação das partes interessadas (acionistas, clientes e fornecedores).
 - » Efeitos negativos: redução da flexibilidade, aumento da resistência e da insegurança.
 - » Mudanças no perfil e estrutura da mão de obra: isto é, causa mudanças na estrutura organizacional, nos processos gerenciais e no estilo administrativo.

Importante

Para Rezende (2005), todo sistema, usando ou não recursos de tecnologia de informação, que armazena dados e gera informação pode ser, genericamente considerado um sistema. Para conceituação inicial, informação é todo dado contido no universo, que seja útil, e com sentido para quem usa a informação. Quando a informação é trabalhada por pessoas e por recursos, possibilitando a geração de cenários e oportunidades, damos a ela o nome conhecimento.

A teoria geral de sistemas fundamenta-se em três premissas básicas:

- Os sistemas existem dentro dos sistemas, porque as moléculas estão dentro das células, as células dentro dos tecidos, os tecidos dentro dos órgãos, os órgãos dentro dos organismos.
- Os sistemas são abertos, porque são caracterizados por um processo de intercâmbio infinito com seu ambiente, que são os outros sistemas e, quando o intercâmbio finaliza, o sistema perde suas fontes de energia.
- As funções de um sistema dependem de sua estrutura, porque os sistemas são interdependentes e, à medida que suas funções se contraem ou expandem, sua estrutura as acompanha.

Dicas

Ainda segundo Rezende (2005), os sistemas de informação deverão contribuir significativamente para a solução de problemas organizacionais. Nesse sentido, o esforço das organizações



deve-se concentrar nos níveis superiores dos sistemas de informação, ou seja, sistemas de informação estratégico e de gestão.

Com mudanças tecnológicas cada vez mais velozes, crescimento na complexidade dos produtos, competição global, disseminação do uso da tecnologia e evolução nos conceitos da qualidade, torna-se necessário investigar os processos de desenvolvimento de sistemas, com o objetivo de alcançar produtos com qualidade superior. Essa investigação é justificada uma vez que a qualidade de um sistema está diretamente relacionada à qualidade do seu processo de desenvolvimento. (SEGORIA, 2001).



De acordo com Rezende (2010), a evolução dos sistemas ocorre por meio de ciclos, apresentados a seguir.

- Sistema de informação manual: os processos são realizados pelo ser humano, sem interferência da tecnologia e de suas ferramentas.
- Sistema de informação automatizado: são inseridos aos processos diários de trabalho, processos mecânicos, pneumáticos, elétricos e robóticos.



- Sistema de informação mecanizado: há a migração dos processos manuais para os processos mecanizados, sem melhorias, e ocorre apenas a racionalização das tarefas por meio da repetição.

- Sistema de informação informatizado: são inseridos aos processos dados inteligentes que agregam valor às informações (respostas), aumentando, assim, a confiabilidade dos resultados obtidos.
- Sistema de informação gerencial e estratégica: a alta direção passa a usar os sistemas como ferramentas de suporte às decisões estratégicas e gerenciais.

Importante

A natureza dos serviços prestados por um sistema varia muito em função das características particulares da organização ou do ambiente de uso e do nível procurado de transformação dos sistemas manuais em sistemas informatizados.

Além dos equipamentos e dos sistemas, existe outro aspecto importante para quase todas as aplicações, que é o suporte que os fabricantes oferecem, em especial, manutenção, documentação, instalação, treinamento e assistência pós-venda. (MEIRELLES, 1994). Esse aspecto tem grandes reflexos no principal componente dos sistemas: as pessoas que gerenciam e usam a tecnologia e os sistemas.



Em um sistema de processamento de dados, existem três componentes principais: o *hardware*, o *software* e o usuário.

- *Hardware*: unidade central de processamento e demais equipamentos, é o computador físico.
- *Software*: são os programas, as instruções e tarefas que a máquina (*hardware*) pode executar.
- *Usuário*: uma ou várias pessoas, realizando as tarefas necessárias para o funcionamento dos outros componentes do sistema.

O sistema resulta da combinação desses componentes necessários para que o computador funcione e realize algo que seja útil dentro do contexto no qual está inserido – organização, casa, profissão liberal ou outro ambiente de utilização.



Sistematizando o conhecimento

De acordo com Rezende (2005), o conhecimento pode ser considerado:

- a) uma informação trabalhada por pessoas e por recursos, possibilitando a geração de cenários e oportunidades.
- b) um dado que não pode ser trabalhado.
- c) algo que tem baixa aplicabilidade para a sociedade.
- d) um conjunto de recursos para um determinado objetivo.

Comentário: para Rezende (2005), quando a informação é trabalhada por pessoas e por recursos, o que possibilita a geração de cenários e oportunidades, damos a ela o nome conhecimento. A opção correta, portanto, é a alternativa "a".

1.2 Princípios de engenharia de software

De acordo com dois expoentes da nova Sociedade da Informação, permite-se definir o *software* como um forte agente de mudanças.

- Para Masuda (1982), o homem se encontra em um novo período, baseado na combinação das tecnologias de computadores e comunicações. Essa combinação vem dando origem à nova sociedade - a de Tecnologia de Informação, fundamentada na evolução da microeletrônica e guiada pela evolução do *software*.
- Segundo Lucena e Campos (1996), o *software* é uma tecnologia de suma importância para todas as áreas do conhecimento. A indústria se transforma no negócio mais competitivo da história contemporânea e seu componente - o *software* - é a força principal por trás da inovação.



Saiba mais

De acordo com Paula-Filho (2007), a Informática é o ramo da ciência cujo foco é o processamento da informação, por meio de máquinas (hardware). A Ciência, por sua vez, tem como objetivo a acumulação do conhecimento através do método científico, geralmente baseado em observações e experimentações. A Engenharia de Software, no que lhe concerne, utiliza resultados da Ciência e fornece problemas para o estudo desta.

Assim como o PMBOK (Corpo de Conhecimentos em Gestão de Projetos - <https://www.pmi.org/>) fora desenvolvido pelo PMI (*Project Management Institute*) o SWEBOK (Corpo de Conhecimentos em Engenharia de Software - <https://www.computer.org/web/swebok>) fora patrocinado pelo IEEE Computer Society (Instituto de Engenheiros Eletricistas e Eletrônicos).

Importante

Segundo Paulo-Filho (2007), as máquinas de tratamento de informações são organizadas em estruturas úteis, formando os sistemas de informática. O software é a parte programável de um sistema de informática, sendo um elemento central: realiza estruturas complexas e flexíveis que agregam valor ao sistema.

Essa abordagem foi originalmente sugerida em uma Conferência da OTAN em 1968 – conferência realizada com o objetivo de discutir o que foi então chamado de “a crise do *software*”. No sentido clássico, o termo engenharia refere-se à aplicação de princípios científicos ao projeto, à manufatura e à operação de estruturas e máquinas.

Voltado à Engenharia de *Software*, o termo é caracterizado pela aplicação dos princípios científicos, métodos, modelos, padrões e teorias que possibilitam gerenciar, modelar, projetar, implementar, medir, analisar, manter e melhorar continuamente sistemas de *software*. (PETERS; PEDRYCZ, 2001).



Saiba mais

Para Sommerville (2007), a Engenharia de Software é um ramo da Engenharia cujo foco é o desenvolvimento de sistemas dentro de custos adequados e com qualidade. Software é algo abstrato e intangível, não é limitado por materiais. Contudo, a falta de restrições significa que o software pode facilmente se tornar complexo e, assim, difícil de ser entendido.

A seguir serão apresentados alguns conceitos-chave que irão nortear nossas próximas conversas, adaptadas de Sommerville (2007).

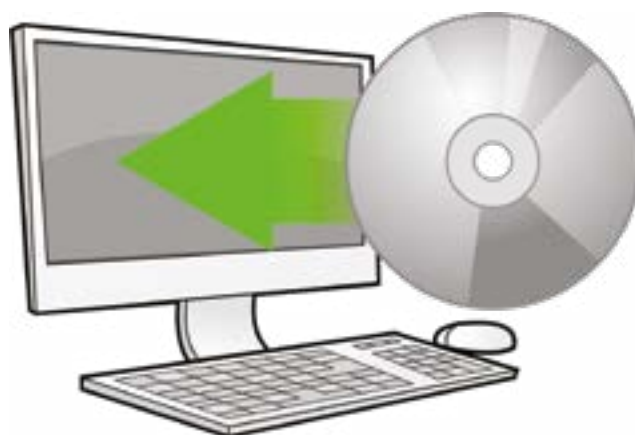
- *Software*: é um programa, sua documentação e configuração associados, necessários para que sua operação ocorra de maneira correta.
- Sistema de *software*: é um conjunto de programas, arquivos de configuração, documentação do sistema e documentação do usuário final.



- Processo de *software*: é um conjunto de atividades e resultados associados que produz um sistema de *software*. Para FIORINI (1998), conhecer os processos significa conhecer como os produtos e serviços são planejados, produzidos e entregues ao cliente. Para que um processo de *software* seja efetivo para o cumprimento dos seus objetivos, torna-se necessário um planejamento detalhado que revele a realidade do ambiente de desenvolvimento do *software*. Devem-se considerar aspectos específicos do projeto, como metas e políticas, equipe de desenvolvimento, cronograma, disponibilidade de recursos humanos, técnicos e financeiros.



- Modelo de processo de *software*: é uma descrição simplificada do processo de *software*. Os modelos de processo incluem as atividades que fazem parte do processo de *software*, assim como os papéis das pessoas envolvidas na Engenharia de *Software*.
- Método de engenharia de *software*: é uma abordagem estruturada para o desenvolvimento de *software*, cujo objetivo é facilitar a produção desse *software*, dentro de prazos, custos e recursos disponíveis.
- Qualidade de *software*: segundo Segoria (2001), até a década de 70, a noção da qualidade estava ligada à parte interna do *software*, sendo avaliada em termos de linhas de código e tempo de execução. Nos anos 80, havia uma maior qualificação dos usuários. Nos anos 90, a discussão assumiu um caráter mais formal com a apresentação de normas como a ISO (*International Standardization Organization*). Atualmente, a discussão pela busca da qualidade leva as organizações a se preocuparem com a melhoria de seus produtos desde a sua concepção. Portanto, o enfoque principal dado às atividades de garantia da qualidade de *software* tem se concentrado na melhoria de seu processo de desenvolvimento.



- Ciclo de vida de *software*: de acordo com o IEEE (1991), ciclo de vida de um *software* é um período de tempo que se inicia quando um *software*-produto é concebido e que se finaliza quando ele não está mais disponível para uso. Um ciclo de vida de *software* típico inclui as fases de engenharia de sistemas, análise, projeto, codificação, testes e manutenção (PRESSMAN, 1992).

Importante

Uma abordagem de Engenharia à produção de software caracteriza-se, portanto, por um desenvolvimento prático, ordenado e medido. O principal objetivo dessa abordagem é produzir sistemas satisfatórios que cumpram prazos e custos.

Há um bom motivo para usarmos essa abordagem no planejamento, no desenvolvimento e avaliação e manutenção de *software* – evitamos assim que o *software* seja desenvolvido de maneira aleatória e pouco profissional.

Sistematizando o conhecimento

Para Peters e Pedryca (2001), a Engenharia de *Software* é voltada para:

- a) aplicação dos princípios científicos, métodos, modelos, padrões e teorias que possibilitam gerenciar, modelar, projetar, implementar, medir, analisar, manter e melhorar continuamente sistemas de *software*.
- b) aplicação dos princípios informais, métodos, modelos, padrões e teorias que possibilitam gerenciar, modelar, projetar, implementar, medir, analisar, manter e melhorar continuamente sistemas de *software*.
- c) a falta de aplicação dos princípios científicos, métodos, modelos, padrões e teorias que possibilitam gerenciar, modelar, projetar, implementar, medir, analisar, manter e melhorar continuamente sistemas de *software*.

d) aplicação dos princípios amadores, métodos, modelos, padrões e teorias que possibilitam gerenciar, modelar, projetar, implementar, medir, analisar, manter e melhorar continuamente sistemas de software.

Comentário: voltado à Engenharia de Software, o termo é caracterizado pela aplicação dos princípios científicos, métodos, modelos, padrões e teorias que possibilitam gerenciar, modelar, projetar, implementar, medir, analisar, manter e melhorar continuamente sistemas de software. (PETERS; PEDRYCZ, 2001). A opção correta, portanto, é a alternativa “a”.

1.3 Princípios de modelagem

A modelagem de sistemas é o processo de elaboração de modelos que representa as perspectivas de um sistema. Esse processo auxilia os desenvolvedores a entender as funcionalidades que esse futuro sistema conterá e como será a comunicação com os usuários finais.



Importante

De acordo com Pressman (2011), modelos são criados para se ter um melhor entendimento do que será realmente desenvolvido. Quando o elemento a ser construído for físico (como um edifício), pode-se construir um modelo idêntico ao real, porém, com uma escala menor.

Quando tratamos de desenvolver um sistema, tal modelo deve refletir as informações, assim como as características que os usuários necessitam.

Diversas são as perspectivas sobre modelagem de sistemas, entre elas destacam-se:

- perspectiva externa: na qual o ambiente do sistema ou seu contexto são modelados;
- perspectiva de interação: na qual são modelados os relacionamentos entre o sistema e o ambiente ou entre os componentes do sistema;
- perspectiva estrutural: na qual são modeladas as estruturas de dados que serão processadas pelo sistema;
- perspectiva comportamental: na qual é realizada a modelagem de como o sistema responderá a determinados eventos (acontecimentos).



No contexto da Engenharia de *Software*, podem ser criadas duas classes de modelos, que são os de requisitos e os de projetos:

- modelos de requisitos: também denominados modelos de análise, representam as necessidades (requisitos) dos clientes, descrevendo o sistema em três domínios diferentes: domínio da informação, domínio funcional e domínio comportamental;
- modelos de projetos: representam características do sistema que ajudam os desenvolvedores a construí-lo: a arquitetura, a interface com o usuário e os detalhes quanto aos componentes.

Entre os princípios da modelagem de requisitos, Pressman (2011) destaca:

- o universo de informações sobre um problema deve ser representado e compreendido: o universo de informações compreende os dados constantes no sistema (por exemplo, usuário final ou dispositivos externos), os dados que saem do sistema (como interface com o usuário, gráficos e relatórios), e a armazenagem de dados que coleta e armazena dados permanentemente;



- as funções que o sistema desempenha devem ser definidas e entendidas pela equipe: essas funções podem ser descritas em diferentes níveis de abstração, desde o mais genérico até o nível mais detalhado, uma vez que essas funções oferecem benefícios para os clientes e também para o suporte interno que desenvolve o sistema;
- o comportamento do *software* deve ser representado: o comportamento de um *software* é comandado por sua interação com o ambiente externo. Alimentações fornecidas pelos usuários finais, informações referentes a controles provenientes de um sistema externo, todos esses elementos determinam que o *software* opere de maneira específica;



- os modelos que descrevem informações, funções em comportamentos devem ser divididos para que revelem detalhes por camadas: a modelagem de requisitos permite que se entenda melhor o problema e se estabeleçam bases para a solução. Nesse sentido, o problema complexo é subdividido em subproblemas menores até que cada um seja compreendido;
- a análise deve partir da informação essencial para o detalhamento da implementação: a modelagem dos requisitos começa a partir das necessidades do cliente. Assim, os requisitos são levantados ouvindo a voz dos usuários que utilizarão o sistema, sem se preocupar com os detalhes de como será implementado.

Entre os princípios da modelagem de projetos, Pressman (2011) destaca:

- o projeto deve ser roteirizado para a modelagem dos requisitos: a modelagem de projetos traduz a modelagem de requisitos para um conjunto de subsistemas que implementam funções mais amplas e um conjunto de componentes que são a concretização das classes de requisitos;
- sempre considere a arquitetura do sistema a ser construído: a arquitetura do sistema é a espinha dorsal do sistema a ser desenvolvido, uma vez que ela pode afetar as interfaces, as estruturas de dados, o fluxo de controle dos programas, a maneira pela qual os testes devem ser conduzidos;
- o projeto de dados é tão importante quanto o projeto das funções de processamento: um projeto de dados bem estruturado ajuda a simplificar o fluxo do programa, torna mais fácil a elaboração do projeto e a implementação dos componentes do *software*, tornando o processamento mais eficiente;
- as interfaces devem ser cuidadosamente projetadas: isso facilita a integração e auxilia o responsável pelos testes quanto à validação das funções dos componentes que compõem o sistema;



- o projeto de interface com o usuário deve refletir as necessidades do usuário final, enfatizando a facilidade de uso do sistema: uma vez que um projeto pobre de interface, pode levar à percepção de um sistema ruim;
- o projeto no nível de componentes deve ser funcionalmente independente: isso significa que a funcionalidade do componente do sistema deve ser coesa, lógica, isto é, focada em apenas uma função;
- os componentes devem ser relacionados livremente: esse relacionamento pode ser obtido por meio, por exemplo, de mensagens via interface;



- representações de projetos (modelos) devem ser de fácil compreensão: o objetivo dessas representações é facilitar o desenvolvimento do sistema, os testes e a manutenção desse sistema no futuro;
- o projeto deve ser desenvolvido iterativamente: as primeiras iterações (repetições, o voltar a fazer) são realizadas para refinar o projeto e corrigir erros. As próximas já servirão para tornar o sistema cada vez mais simples.

Sistematizando o conhecimento

Qual o objetivo de se elaborar um modelo, sob a visão de Pressman (2011)?

- a) Um modelo é de custo mais baixo, que pode ser, mais tarde, vendido como sistema.
- b) Um modelo é criado para se ter um melhor entendimento do que será realmente projetado.
- c) Um modelo é criado para se ter um melhor entendimento do que será realmente desenvolvido.
- d) Um modelo é criado para se economizar com mão de obra de desenvolvimento especializada.

Comentário: de acordo com Pressman (2011), modelos são criados para se ter um melhor entendimento do que será realmente desenvolvido. Quando o elemento a ser construído for físico (como um edifício), pode-se construir um modelo idêntico ao real, porém com uma escala menor. A opção correta, portanto é a alternativa “c”.

1.4 Metodologias para desenvolvimento de sistemas

As organizações buscam constantemente as inovações com o objetivo de oferecer os melhores produtos que atendam às necessidades do mercado. Com o advento da internet e o avanço tecnológico no século XX, torna-se um desafio manter-se à frente dos concorrentes.



ADVENTO:
Chegada



Saiba mais

O Dicionário Aurélio nos mostra que o termo “metodologia” tem, em sua essência, a arte de dirigir o espírito na investigação da verdade. Segundo Rezende (1999), uma metodologia constitui-se de uma abordagem organizada para atingir um objetivo, por meio de passos preestabelecidos. É um roteiro, um processo dinâmico e interativo para o desenvolvimento de sistemas. A metodologia permite o uso de várias técnicas por opção dos desenvolvedores de sistemas.

A metodologia deve auxiliar o desenvolvimento de sistemas, de modo que atenda de maneira adequada às necessidades dos clientes, com os recursos disponíveis e dentro de um prazo ideal definido em conjunto com as partes interessadas. Não deve limitar a criatividade profissional, mas deve ser um instrumento que determine um planejamento metódico, que harmonize e coordene as demais áreas envolvidas.



Importante

Rezende (1999) destaca apropriadamente que todo e qualquer sistema deve ser elaborado por meio de uma metodologia bem estruturada, moderna e que ofereça documentação completa. Toda organização deve ter uma metodologia, visando a efetividade, continuidade, segurança e transparência, aceitas e implementadas pelos seus gestores, clientes e desenvolvedores.

As justificativas para o desenvolvimento de sistemas por meio de metodologias devem ser formalizadas de modo que seus produtos (REZENDE, 1999):

- forneçam a visão do estado do projeto a qualquer instante;
- sirvam como meio de comunicação entre os envolvidos;
- indiquem o nível de participação de todos os envolvidos;
- mantenham um histórico documental do sistema;
- sejam sempre base para as fases e subfases seguintes.



Soares-Neto (1993) destaca os princípios para uma metodologia de desenvolvimento de sistemas eficiente:

- Qualidade: está focada na capacidade que o sistema tem em atingir seus objetivos.
- Compromisso com o problema da organização: parte da premissa de que, primeiramente, se deve conhecer o problema, para, assim, desenvolver um sistema que contribua para

o alcance do objetivo proposto. Isso ocorre por meio do diagnóstico da situação atual do problema que se deseja resolver.

- Uma metodologia é composta por perguntas: a metodologia deve basear-se em um conjunto organizado de perguntas, capazes de orientar as partes interessadas para o reconhecimento, investigação e resolução do problema.



- A metodologia precisa apresentar uma natureza simples: deve ser de fácil utilização pelos responsáveis pelo desenvolvimento do sistema.
- A modelagem como auxílio ao pensamento: o processo de modelagem deve auxiliar o pensamento da equipe responsável pela análise, na busca das respostas consensuais às perguntas básicas da metodologia.
- A criação coletiva dos sistemas: sugere-se que haja um alto grau de participação e coautoria dos usuários na criação dos sistemas, o que proporciona, como resultado, um processo coletivo de criação.

Fournier (1994) já defendia a necessidade do uso de metodologias para o desenvolvimento de sistemas, com o propósito de disciplinar e flexibilizar a produção de sistemas com qualidade.

Importante

Nesse sentido, considerando as metodologias tradicionais e ágeis de desenvolvimento de sistemas presentes no mercado, cabe a cada empresa definir a que vai lhe retornar um resultado mais eficiente, atendendo às suas necessidades.

Segundo Royce (1970), uma metodologia tradicional também pode ser chamada de metodologia orientada à documentação.

Saiba mais

Na década de 70, o custo de fazer alteração/atualização em sistemas era muito alto, isso porque o acesso aos computadores era também muito limitado. Nesse sentido, e sem o apoio das modernas ferramentas de gestão de sistemas que somos providos atualmente, um sistema era totalmente planejado e documentado antes de ser desenvolvido. É o que chamamos de modelo clássico ou sequencial.

De acordo com Schwaber e Beedle (2002), o termo metodologia ágil tornou-se popular em 2001 quando os processos de *software* foram em parte representados por um método denominado SCRUM, e também pelo estabelecimento do manifesto ágil, cujos princípios são (AGILE MANIFESTO, 2004):

- indivíduos e interações, em vez de processos e ferramentas;
- colaboração do cliente, em vez de negociações de contratos;
- respostas rápidas às mudanças, em vez de seguir planos.



Importante

Para Pressman (2011), a Engenharia de Software ágil combina filosofia com um conjunto de princípios de desenvolvimento. Ela defende a satisfação do cliente e a entrega incremental, equipes de projeto pequenas e altamente motivadas, métodos informais e, acima de tudo, simplicidade no desenvolvimento.

Para termos agilidade, devemos seguir os 12 princípios estabelecidos pela Agile Alliance (2018). São eles:

- a prioridade está na satisfação do cliente por meio de entregas rápidas e contínuas;
- devemos sempre acolher os pedidos de mudanças no sistema;
- entregue *software* em intervalos curtos;
- os responsáveis pela área comercial e os desenvolvedores devem trabalhar colaborativamente;
- a motivação em trabalhar motivadamente é essencial para as equipes;
- seja claro e eficiente ao de transmitir as informações para as equipes;



- a principal medida de progresso do projeto é o sistema em funcionamento;
- os processos ágeis promovem desenvolvimento sustentável, ou seja, todos os envolvidos devem estar capacitados a manter um ritmo constante de trabalho e também objetivos;
- a atenção contínua com a excelência técnica aumenta a agilidade;
- manter a simplicidade é considerado essencial;
- as melhores arquiteturas, requisitos e projetos estão ligados às equipes auto-organizáveis;
- a equipe se autoavalia, em intervalos regulares, no intuito de se tornarem cada vez mais eficientes.

Sistematizando o conhecimento



Qual a definição de metodologia, sob a visão de Rezende (1999)?

- a) Uma metodologia constitui-se de uma abordagem organizada para atingir um objetivo, por meio de passos preestabelecidos. É um roteiro, um processo estático e interativo para o desenvolvimento de sistemas.
- b) Umametodologia constitui-se de uma abordagem organizada para atingir um objetivo, por meio de passos preestabelecidos. É um roteiro, um processo dinâmico e interativo para o desenvolvimento de sistemas.
- c) Uma metodologia constitui-se de uma abordagem desorganizada para atingir um objetivo, por meio de passos preestabelecidos. É um roteiro, um processo dinâmico e interativo para o desenvolvimento de sistemas.
- d) Uma metodologia constitui-se de uma abordagem organizada para atingir um objetivo, por meio de passos não planejados. É um roteiro, um processo dinâmico e interativo para o desenvolvimento de sistemas.

Comentário: segundo Rezende (1999), uma metodologia constitui-se de uma abordagem organizada para atingir um objetivo, por meio de passos preestabelecidos. É um roteiro, um processo dinâmico e interativo para o desenvolvimento de sistemas. Dessa forma, a opção correta é a alternativa “b”.

1.5 Ferramentas de engenharia de software

De acordo com Rezende (2010), para a execução das estratégias organizacionais, será necessário conhecer e estabelecer algumas ferramentas, que devem ser amplamente discutidas com todos os envolvidos, a fim de sistematizar os conhecimentos sobre o sistema a ser desenvolvido.

Saiba mais



Segundo Davis (1994), um sistema é um conjunto de dados sendo processados por pessoas, máquinas e métodos organizados, com o propósito de cumprir um certo número de funções específicas. De acordo com o Dicionário Aurélio, uma ferramenta é um conjunto de instrumentos e utensílios empregados em um determinado ofício. Nesse sentido, é imprescindível o uso de ferramentas que nos darão apoio durante o processo de desenvolvimento de sistemas. Será o nosso próximo assunto a ser abordado.

Para Rezende (2010), as principais ferramentas de estratégia organizacional voltadas à geração de informações são:

- planejamento estratégico empresarial;
- políticas empresariais e políticas gerais de informação e de Tecnologia de Informação;
- planejamento estratégico de informações;
- modelos de informações empresariais e mapas de conhecimento;

- metodologias para aquisição / desenvolvimento de sistemas e projetos em geral, e metodologia de tempos e custos de atividades;
- normas e padrões técnico-operacionais de Tecnologia de Informação;
- manuais e documentações.



**INTER-
DEPEN-
DÊNCIA:**
Dependên-
cia mútua

Pressman (2002) define ferramentas de gerência de projetos de *software* como as ferramentas de planejamento (estimativas de custo e esforço e planejamento do projeto) bem como as ferramentas de apoio ao acompanhamento e controle. Uma ferramenta de planejamento deve apoiar desde o cálculo do esforço estimado, a duração do projeto e o número de recursos recomendados, até permitir a definição de todas as tarefas do projeto e a **interdependência** entre elas.



Importante

Pressman (2002) afirma ainda que uma ferramenta de gerência de projeto deva ser utilizada para acompanhar e monitorar o cronograma do projeto e o plano de projeto em uma base contínua, além de fornecer ligações com outras ferramentas que dão suporte para outros processos de desenvolvimento de *software*.

No âmbito de desenvolvimento de *software*, as ferramentas CASE (que, no inglês, significam *Computer-Aided Software Engineering*) apoiam a criação, a edição e a análise das notações em engenharia de *software*. (SOMMERVILLE, 2007).

As classificações CASE ajudam a compreender os tipos de ferramentas CASE e seu papel em apoiar as atividades de processo de *software*, apresentadas a seguir a partir de três perspectivas.

- Uma perspectiva de função: em que as ferramentas CASE são classificadas de acordo com sua função específica.
- Uma perspectiva de processo: em que as ferramentas CASE são classificadas de acordo com as atividades de apoio que fornecem.
- Uma perspectiva de integração: em que as ferramentas CASE são classificadas de acordo com sua organização em unidades integradas que apoiam uma ou mais atividades do processo.

A tabela a seguir representa a classificação das ferramentas CASE segundo suas funções, por meio de uma quantidade de tipos diferentes de ferramentas e exemplos para cada uma delas.

Classificação das ferramentas CASE segundo suas funções

TIPO DE FERRAMENTA	EXEMPLOS DE APLICAÇÃO
Ferramentas de planejamento	Ferramentas PERT, ferramentas para estimativas, planilhas.
Ferramentas de edição	Editores de texto, editores de diagramas, processadores de texto.
Ferramentas de gerenciamento de mudanças	Ferramenta de controle de requisitos, sistemas de controle de mudanças.
Ferramentas de gerenciamento de configuração	Sistemas de gerenciamento de versões, ferramentas de construção de sistemas.
Ferramentas de prototipação	Linguagens de alto nível, geradores de interface com o usuário.
Ferramentas de apoio ao método	Editores de projeto, dicionário de dados, geradores de código.
Ferramentas de processamento de linguagens	Compiladores e interpretadores.
Ferramentas de análise de programas	Geradores de referência cruzada, analisadores estáticos e dinâmicos.
Ferramentas de testes	Geradores de dados de testes.
Ferramentas de depuração	Sistemas de depuração interativos.
Ferramentas de documentação	Programas de formatação de páginas, editores de imagens.
Ferramentas de reengenharia	Sistemas de referência cruzada, sistemas de reestruturação de programas.

Fonte: SOMMERVILLE (2007).

Já as ferramentas de desenvolvimento ágil auxiliam em um ou mais aspectos do desenvolvimento ágil com ênfase em facilitar a geração rápida de *software*. Segundo Pressman (2011), alguns dos autores da metodologia ágil argumentam que as ferramentas de *software* deveriam ser vistas como um suplemento menor para as atividades e não como um pivô para o sucesso das equipes.



Algumas dessas ferramentas são sociais. Iniciando-se no estágio de contratação de pessoal, algumas são tecnológicas, auxiliando equipes distribuídas geograficamente. Nesse sentido, as ferramentas são voltadas à comunicação e à colaboração, e incorporam mecanismos que fornecem informações entre as equipes.



Sistematizando o conhecimento

Pressman (2002) define ferramentas de gerência de projetos de *software* como:

- a) as ferramentas de documentação (estimativas de custo e esforço e planejamento do projeto) bem como as ferramentas de apoio ao acompanhamento e controle.
- b) as ferramentas de configuração (estimativas de custo e esforço e planejamento do projeto) bem como as ferramentas de apoio ao acompanhamento e controle.
- c) as ferramentas de planejamento (estimativas de custo e esforço e planejamento do projeto) bem como as ferramentas de apoio ao acompanhamento e controle.
- d) as ferramentas de execução (estimativas de custo e esforço e planejamento do projeto) bem como as ferramentas de apoio ao acompanhamento e controle.

Comentário: Pressman (2002) define ferramentas de gerência de projetos de *software* como as ferramentas de planejamento (estimativas de custo e esforço e planejamento do projeto) bem como as ferramentas de apoio ao acompanhamento e controle. Uma ferramenta de planejamento deve apoiar desde o cálculo do esforço estimado, a duração do projeto e o número de recursos recomendados, até permitir a definição de todas as tarefas do projeto e a interdependência entre elas. Portanto opção correta é a alternativa “c”.

Resumindo

Aprendemos nesta lição os principais conceitos sobre a teoria dos sistemas de informações, assim como os princípios de modelagem e as ferramentas que suportam os processos de engenharia

de *software*. Qualificar os desenvolvedores nas melhores práticas e conceitos sobre o tema são fundamentais para a produção de sistemas cada vez mais robustos, precisos e confiáveis aos seus clientes e usuários finais.

Veja se você se sente apto a:

- explicar a teoria geral de sistemas, destacando seus principais conceitos;
- discutir os princípios de engenharia de *software* e de modelagem atuais;
- destacar as metodologias e ferramentas para desenvolvimento de sistemas e seus impactos para a garantia da qualidade do *software* produzido.



Parabéns, você finalizou esta lição!

Agora responda às questões ao lado.

Exercícios

Questão 1 - O uso dos recursos da informática ainda continua pouco explorado, devido principalmente:

- a) ao baixo grau de informatização que muitas empresas possuem, e também em relação ao pouco uso da tecnologia nas decisões estratégicas e na tomada de decisões.
- b) ao alto grau de informatização que muitas empresas possuem, e também em relação ao pouco uso da tecnologia nas decisões estratégicas e na tomada de decisões.
- c) ao baixo grau de informatização que muitas empresas possuem, e também em relação ao alto uso da tecnologia nas decisões estratégicas e na tomada de decisões.
- d) ao alto grau de informatização que muitas empresas possuem, e também em relação ao alto uso da tecnologia nas decisões estratégicas e na tomada de decisões.

Questão 2 - De acordo com Segoria (2001), as realizações e as atividades desenvolvidas pelo homem em qualquer segmento produtivo são geradas por:

- a) percepção de suas necessidades, acompanhamento da evolução tecnológica e adequação aos cenários e aos comportamentos da sociedade, seguindo uma abordagem local.
- b) percepção de suas necessidades, atraso tecnológico e adequação aos cenários e aos comportamentos da sociedade, seguindo uma abordagem global.
- c) falta de percepção de suas necessidades, acompanhamento da evolução tecnológica e adequação aos cenários e aos comportamentos da sociedade, seguindo uma abordagem global.
- d) percepção de suas necessidades, acompanhamento da evolução tecnológica e adequação aos cenários e aos comportamentos da sociedade, seguindo uma abordagem global.

Questão 3 - Meirelles (1994) destaca apropriadamente que, para entender os ciclos de evolução da informática e seus impactos, é necessário perceber a rapidez com que as transformações vêm ocorrendo nas últimas décadas. O autor apresenta as seguintes características:

- a) transformações na indústria dos computadores, custos crescentes da tecnologia, capacidades de processamento crescente e impactos sobre as empresas e sobre as pessoas.

b) transformações na indústria dos computadores, custos decrescentes da tecnologia, capacidades de processamento decrescente e impactos sobre as empresas e sobre as pessoas.

c) transformações na indústria dos computadores, custos decrescentes da tecnologia, capacidades de processamento crescente e impactos sobre as empresas e sobre as pessoas.

d) transformações na indústria dos computadores, custos decrescentes da tecnologia, capacidades de processamento crescente, e pouquíssimos impactos sobre as empresas e sobre as pessoas.

Questão 4 - Para Rezende (2005), o que é informação?

a) É todo dado contido no universo que seja útil e com sentido para quem usa a informação. Quando a informação é trabalhada por pessoas e por recursos, o que possibilita a geração de cenários e oportunidades, damos a ela o nome conhecimento.

b) É todo conhecimento contido no universo que seja útil e com sentido para quem usa a informação. Quando a informação é trabalhada por pessoas e por recursos, o que possibilita a geração de cenários e oportunidades, damos a ela o nome conhecimento.

c) É todo dado contido no universo que seja inútil e com sentido para quem usa a informação. Quando a informação é trabalhada por pessoas e por recursos, o que possibilita a geração de cenários e oportunidades, damos a ela o nome conhecimento.

d) É todo dado contido no universo que seja útil e sem sentido para quem usa a informação. Quando a informação é trabalhada por pessoas e por recursos, o que possibilita a geração de cenários e oportunidades, damos a ela o nome conhecimento.

Questão 5 - A teoria geral de sistemas fundamenta-se em três premissas básicas, que são:

a) os sistemas existem dentro dos sistemas, os sistemas não são abertos e as funções de um sistema dependem de sua estrutura.

b) os sistemas existem dentro dos sistemas, os sistemas são abertos e as funções de um sistema não dependem de sua estrutura.

c) os sistemas não existem dentro dos sistemas, os sistemas são abertos e as funções de um sistema dependem de sua estrutura.

d) os sistemas existem dentro dos sistemas, os sistemas são abertos e as funções de um sistema dependem de sua estrutura.

Questão 6 - O objetivo do SWEBOK é fornecer:

a) um corpo de dados obsoletos (atuais) em engenharia de *software*.

b) um corpo de conhecimentos em engenharia civil.

c) um corpo de conhecimentos em engenharia de *software*.

d) um corpo de conhecimentos em pesquisas de *software*.

Questão 7 - Segundo Rezende (2005), os sistemas de informação deverão contribuir significativamente para a solução de problemas organizacionais. Nesse sentido, o esforço das organizações deve ser concentrado:

- a) nos níveis inferiores dos sistemas de informação, ou seja, sistemas de informação estratégico e de gestão.
- b) nos níveis superiores dos sistemas de informação, ou seja, sistemas de informação estratégico e de gestão.
- c) nos níveis superiores dos sistemas de informação, ou seja, sistemas de informação operacional e de gestão.
- d) nos níveis superiores dos sistemas de dados, ou seja, sistemas de conhecimento estratégico e de gestão.

Questão 8 - Segundo Segoria (2001), a qualidade de um sistema está diretamente relacionada:

- a) aos altos custos do seu processo de desenvolvimento.
- b) à qualidade do seu processo de desenvolvimento.
- c) às muitas pessoas empregadas em seu processo de desenvolvimento.
- d) aos documentos obsoletos resultantes do seu processo de desenvolvimento.

Questão 9 - De acordo com Rezende (2010), em um sistema de informação manual:

- a) os processos são realizados por um robô, sem interferência humana.
- b) os processos são realizados pelo ser humano, com total interferência da tecnologia e de suas ferramentas.
- c) os processos são realizados pelo ser humano, sem interferência da tecnologia e de suas ferramentas.
- d) os documentos são gerados pela impressora, sem interferência da tecnologia e de suas ferramentas.

Questão 10 - A natureza dos serviços prestados por um sistema varia muito em função:

- a) das características particulares da organização ou do ambiente de uso e do nível procurado de transformação dos sistemas manuais em sistemas informatizados.
- b) das características genéricas da organização ou do ambiente de uso e do nível procurado de transformação dos sistemas informatizados em manuais.
- c) das características particulares da organização ou do ambiente de uso e do nível procurado de transformação dos sistemas elétricos em sistemas manuais.
- d) das características particulares do fornecedor e do nível procurado de transformação dos sistemas manuais em sistemas inflexíveis.