

---

# Engenharia de Software

**Lucilena de Lima**

# Mercado de Trabalho

---

De acordo com Renato Trindade, gerente da Page Personnel, o destaque continuará em torno de profissões especializadas no uso de dados digitais, com foco nos profissionais de alta demanda e ainda escassos no mercado: os cientistas de dados.

Para Carolina Cadorin, diretora da Hays, setores como agronegócio, saúde e educação devem seguir o mesmo caminho das fintechs, startups do setor financeiro, tornando a tecnologia central e inovadora em cada área.

Segundo Leandro Bittioli, gerente sênior da divisão de TI e digital da Talenses, para esse ano, a segurança da informação passa a ser uma preocupação maior dentro das empresas, o que pode levar a ter um profissional especializado responsável pelo tema para impedir qualquer tipo de vazamento de dados ou uso inadequado deles”, comenta.

# Mercado de Trabalho

---

A área de tecnologia da informação deve ser a de maior sucesso em 2019. É o que mostra pesquisa realizada pela Confederação Nacional do Comércio de Bens, Serviços e Turismo (CNC). Segundo o estudo, funções ligadas à TI estiveram entre as carreiras que mais cresceram entre 2007 e 2017. No último ano, em 2018, houve muita demanda de vagas, mas pouca oferta de profissionais qualificados na área. A tendência é que essa procura aumente ainda mais em 2019.

Um estudo do Gartner apontou que serão gastos US\$ 3,8 trilhões apenas pelos departamentos de TI em 2019, um crescimento de 3,2% se comparado com 2018. Só a computação em nuvem deve contar com um crescimento de 22%.

Carreiras como **analista de informações**, **desenvolvedor mobile**, cientista de dados, analista de compliance e de SEO são algumas profissões em ascensão.

<https://www.terra.com.br/noticias/dino/tendencias-de-tecnologia-para-2019-por-que-ti-e-uma-das-areas-com-maior-crescimento> - Publicação:- abril de 2019 - acessado em 23-07-2019

---

# Mercado de Trabalho

Tecnologia Notícia da edição impressa de 16/11/2016. Novas tecnologias desafiam mercado de trabalho da TI - Jornal do Comércio –

Segundo Patricia Knebel - Jornal do Comércio

“Inteligência artificial, realidade aumentada, blockchain, drones, Internet das Coisas (IoT), robôs, impressão 3D são tecnologias que vão mudar os negócios nos próximos anos. Os impactos serão ainda maiores se pensarmos que vão atingir em cheio o mercado de trabalho, a partir da demanda pela criação de novas categorias profissionais - e do fim de outras.”

O setor emprega 1,3 milhão de pessoas, de acordo com dados da Associação Brasileira de Empresas de Tecnologia da Informação e Comunicação (Brasscom). Nos próximos quatro anos, o Brasil irá precisar de 750 mil profissionais.(Patricia Knebel)

([http://jcrs.uol.com.br/\\_conteudo/2016/11/economia/531409-novas-tecnologias-desafiam-mercado-de-trabalho-da-ti.html](http://jcrs.uol.com.br/_conteudo/2016/11/economia/531409-novas-tecnologias-desafiam-mercado-de-trabalho-da-ti.html))

---

# Profissional em TI

- raciocínio lógico.
- visão sistêmica.
- buscar sempre atualizações.
- ser dedicado.
- comprometido.
- trabalho em equipe.
- foco no cliente.
- iniciativa.
- Comunicação / liderança.
- qualidade do trabalho.
- conhecimentos em inglês e espanhol são diferenciais.

---

# Engenharia de Software

Conteúdo Programático:

- Introdução

- Ciclos de Vida

- Levantamento de Requisitos

- Análise e Documentação

- Prototipagem

---

# Critério de Avaliação

- Ao longo do semestre:
  - Exercícios aplicados em sala de aula e em laboratório;
  - Exercícios extraclasse;
  - Trabalhos - Apresentação de seminários;
  - Provas dissertativas individuais.
  - **Prova I e Prova II**
  - **Projeto e seminário**
  - **Fatec Aberta**
- Bibliografia Base:
  - Engenharia de Software, *Roger Pressman*, Makron Books

---

# *Introdução a Engenharia de Software*



---

# Introdução

“O Software ultrapassou o Hardware como chave para o sucesso de muitos sistemas baseados em computador” (Pressman, pg. 3, 1992)

---

# Introdução

- 1950-1970: Desafio principal era desenvolver Hardware
  - reduzisse o custo de processamento e armazenamento de dados
- Hoje: Reduzir o custo e melhorar a qualidade do Software (potencial da computação)
  - Funcionalidade
  - *Human-Friendly*

# Introdução

Engenharia de Software:

É a aplicação prática da ciência da computação, ciências gerenciais, abordagens de engenharia e economia para a análise, projeto, construção e manutenção de produtos de software e a documentação necessária para usar, operar e manter o sistema de software entregue. A tabela a seguir detalha este conceito [Nascimento, 1992].

Grupos	Funções
Desenvolvimento	Análise, Projeto, Codificação e Teste
Gerenciamento	Planejamento, Organização, Alocação de Pessoas, Direção e Controle
Métricas	Confiabilidade, Manutenibilidade, Usabilidade, Flexibilidade e Reusabilidade.
Manutenção	Erro, Correção e Modificação

---

# Introdução

Por que tanta demora para entregar o sistema?

Por que não se cumprem os prazos?

Por que os custos são altos?

Por que não achar todos os erros antes de entregar?

Por que é difícil medir o progresso do desenvolvimento de um software?

---

No mercado atual, não há dúvida de que os profissionais de TI envolvidos com projetos de desenvolvimento de software e soluções corporativas têm um claro desafio: PRODUZIR soluções mais rápidas, melhores e mais baratas que a concorrência.

---

# A importância do Software

- Durante as 3 primeiras décadas da era do computador, o principal desafio era desenvolver um **HARDWARE** de baixo custo e alto desempenho.
- Hoje o desafio é melhorar a qualidade (e reduzir os custos) das soluções baseadas em **SOFTWARE!**

---

# Filosofando... (Reinaldo Bianchi)

- A mudança de uma sociedade industrial para uma baseada na informação é uma Radical Mudança Econômica:
  - Material tem menos valor e Informação tem mais valor
- Antes: quanto menos pessoas possuísse algo, maior o valor.
- Hoje: quanto mais pessoas possuem algo, maior o valor.

---

# A evolução do software

## ○ Software é dividida em 4 Eras:

- Primeiros anos 1950 - 1960
- Segunda Era 1960 - 1970
- Terceira Era 1980 - 2000
- Quarta Era/atual 2000 - ...



---

<b>Período</b>	<b>Evolução</b>
1950-1960	Orientação a batch
	Software totalmente customizado
	Distribuição limitada
1960-1970	Multiusuários
	Tempo Real
	Banco de Dados

---

<b>Período</b>	<b>Evolução</b>
1980-1990	Sistemas distribuídos
	Inteligência Embutida
	Hardware de baixo custo
1990-2000	Sistemas de desktop poderosos
	Tecnologia orientada a objeto
	Sistemas Especialistas....

---

<b>Período</b>	<b>Evolução</b>
2000 - atual	As tecnologias orientadas a objetos
	Uso das técnicas de "quarta geração" para o desenvolvimento de software
	Os sistemas especialistas e o software de inteligência artificial.

# O que é Software?

Programas de computadores associados a documentação.

É um conjunto de soluções algorítmicas, codificadas numa linguagem de programação, executado numa máquina real.

# Características do Software

Complexidade

Conformidade

Mutabilidade

# Características do Software

## **Complexidade**

- Software é mais complexo do que qualquer outro produto construídos por seres humanos.

# Características do Software

## **Conformidade**

- O software deve ser desenvolvido conforme o ambiente. Não é o ambiente que deve se adaptar ao software.
- Desenvolvido ou projetado por engenharia, não manufaturado no sentido clássico (industrial).

Sucesso é medido pela qualidade e não quantidade.

# Características do Software

## **Mutabilidade**

- Existe sempre uma pressão para se fazer mudanças em um software.
- Não se “desgasta”, mas se deteriora devido as mudanças.
- A maioria é feita sob medida em vez de ser montada a partir de catálogos de componentes existentes (reusabilidade de software).



---

# Aplicações de Software

Pressman, página 20

- Software Básico
- Software de Tempo Real
- Software Comercial
- Software Científico ou de Engenharia
- Software Embutido
- Software de Computador Pessoal
- Software de Inteligência Artificial

---

# Crise do Software

Refere-se a um conjunto de problemas encontrados no desenvolvimento de software.

Problemas não se limitam ao software que não funciona adequadamente, mas abrange:

- desenvolvimento, testes, manutenção, suprimento etc.

---

# Crise do Software

Prazos ultrapassados

Custos acima do previsto

Não atendimento dos requisitos do usuário

---

# Crise do Software

## **Elevado custo de manutenção**

- 1/3 dos projetos são cancelados
- 2/3 dos projetos extrapolam o orçamento

## **Custo hardware x software**

- 1970 = 8:2
- 1991 = 2:8
- Hoje = 1:9

---

# Falta de qualidade...

Alguns fatos históricos...

## **Ariane 501**

Erros de software já foram responsáveis por prejuízos milionários e mesmo a perda de vidas humanas. A importância de garantir a qualidade é evidente à luz desta. [Pressman, 2002; Pfleeger, 2003]:

### **Ariane 501**

**Em 4 de junho de 1996, foi lançado o primeiro foguete Ariane 5. Decorridos 40 segundos da sequência de lançamento e a uma altitude de 3.700 metros, o foguete desviou-se de sua trajetória e se auto destruiu com uma explosão.**

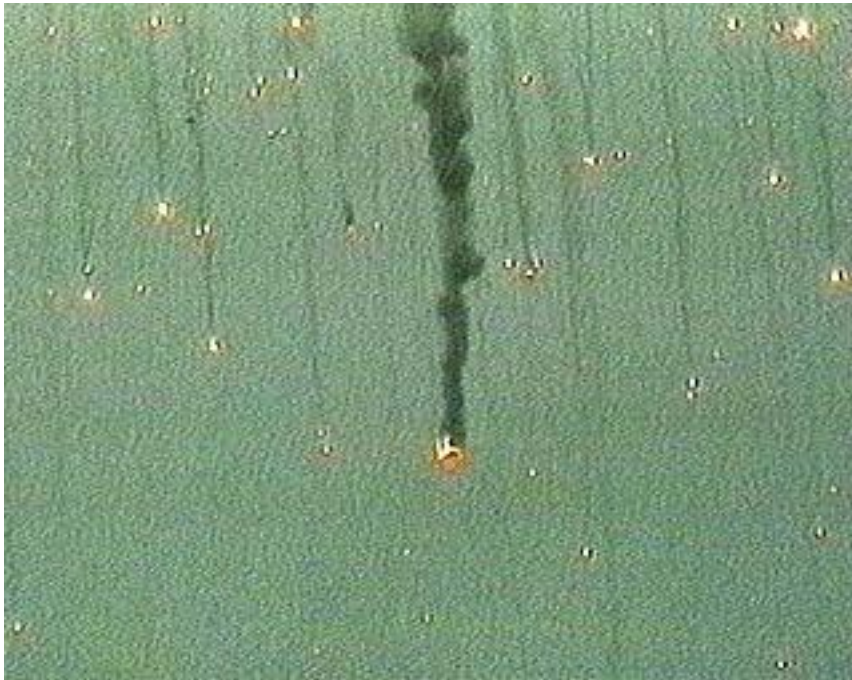
**A causa: a exceção, proveniente de um cálculo: um número em ponto flutuante representado com 64 bits foi convertido para um inteiro com sinal de 16 bits. O número era demasiado grande para ser representado com 16 bits e isso causou uma falha.**

**Continha quatro satélites: o custo foi de U\$ 500 milhões (quantia suficiente para pagar um salário de 2,5 mil dólares a cem programadores que trabalhassem durante um século.)**

## Ariane 501



## Ariane 501



- Explosão 40 segundos após a decolagem.
- Destruição do foguete e carga avaliada em US\$ 500 milhões.



---

## Ariane 501 - O que aconteceu?

Fato: o veículo detonou suas cargas explosivas de autodestruição e explodiu no ar. Por que?

Porque ele estava se quebrando devido às forças aerodinâmicas. Mas por que?

O foguete tinha perdido o controle de direção (atitude). Causa disso?

Os computadores principal e back-up deram shut-down ao mesmo tempo.

---

## Ariane 501 - O que aconteceu?

Por que o Shut-down? Ocorreria um *run time error* (out of range, overflow , ou outro) e ambos computadores se desligaram. De onde veio este erro?

Um programa que convertia um valor em ponto flutuante para um inteiro de 16 bits recebeu como entrada um valor que estava fora da faixa permitida.

---

# Ariane 501 - O que aconteceu?

## Especificamente: O que faltou?

strict precondition 1:

```
{  
    Set."x"=FLPT and Set."y"=INT16  
    and -32768 <= x <= +32767  
__}
```

program code:

```
y := int(x);
```

postcondition:

```
{Set."x"=FLPT and Set."y"=INT16 and  
y=int(x)}
```

## Therac-25

### Therac-25

Equipamento de Radioterapia.

O Therac-25 era uma máquina utilizada em terapia radiológica. Diferente de suas versões anteriores, era totalmente controlado por um computador, um PDP-11.

Entre 1985 e 1987 se envolveu em 6 acidentes, causando mortes por overdoses de radiação:- **Havia indicações de *no dose*, ou seja, nenhuma radiação teria sido emitida; e *treatment pause*, a máquina estava em simples pausa aguardando para continuar a operação. Como se ratava de mensagens comuns, o operador simplesmente ativou outra vez o Therac-25 (paciente faleceu. Uma autópsia revelou que a superexposição à radiação causou tantos danos que, se tivesse sobrevivido, seria preciso receber uma prótese para a cabeça do fêmur praticamente destruída pela radiação).** (página 382 do Pfleeger)

---

# Quais são os problemas? Por que erros acontecem...

A sofisticação do software ultrapassou nossa capacidade de construção.

Nossa capacidade de construir programas não acompanha a demanda por novos programas.

Nossa capacidade de manter programas é ameaçada por projetos ruins.

---

# Perguntas que Engenharia de Software quer responder:

Porque demora tanto para concluir um projeto (não cumprimos prazos)?

Porque custa tanto (uma ordem de magnitude a mais)?

Porque não descobrimos os erros antes de entregar o software ao cliente?

Porque temos dificuldade de medir o progresso enquanto o software está sendo desenvolvido?

---

# Crise de Software

Quando um componente de hardware se desgasta é substituído por uma “peça de reposição”

Não existe “peça de reposição” para software

- Toda falha indica um erro no projeto ou no processo de tradução para o código executável
- Manutenção do software é mais complexa do que a do hardware

---

# Crise de Software

**Refere-se a um conjunto de problemas encontrados no desenvolvimento de software:**

**(1) As estimativas de prazo e de custo freqüentemente são imprecisas**

**“Não dedicamos tempo para coletar dados sobre o processo de desenvolvimento de software”**

**“Sem nenhuma indicação sólida de produtividade, não podemos avaliar com precisão a eficácia de novas ferramentas, métodos ou padrões”**



---

# Crise de Software

(2) A produtividade das pessoas da área de software não tem acompanhado a demanda por seus serviços

**“Os projetos de desenvolvimento de software normalmente são efetuados apenas com um vago indício das exigências do cliente”**

---

# Crise de Software

(3) A qualidade de software às vezes é menos que adequada

**Só recentemente começam a surgir conceitos quantitativos sólidos de garantia de qualidade de software**

(4) O software existente é muito difícil de manter

**A tarefa de manutenção devora o orçamento destinado ao software**

**A facilidade de manutenção não foi enfatizada como um critério importante**

---

# Causas dos problemas associados à Crise de Software

## 1. próprio caráter do Software

**O software é um elemento de sistema lógico e não físico.**

**Conseqüentemente, o sucesso é medido pela qualidade de uma única entidade e não pela qualidade de muitas entidades manufaturadas**

**O software não se desgasta, mas se deteriora!!!**

---

# Causas dos problemas associados à Crise de Software

## 2. falhas das pessoas responsáveis pelo desenvolvimento de Software

**Gerentes sem nenhum *background* em software**

**Os profissionais da área de software têm recebido pouco treinamento formal em novas técnicas para o desenvolvimento de software**

**Resistência a mudanças**

---

# Causas dos problemas associados à Crise de Software

## 3. mitos do Software

**propagaram desinformação e  
confusão**

- ▶ *administrativos*
- ▶ *cliente*
- ▶ *profissional*

---

# Mitos do Software - Administrativos

**Já temos um manual repleto de padrões e procedimentos para a construção de software. Isso não oferecerá ao meu pessoal tudo o que eles precisam saber?**

*Realidade:*

Será que o manual é usado?

Os profissionais sabem que ele existe?

Ele reflete a prática moderna de desenvolvimento de software?

Ele é completo?

---

# Mitos do Software - Administrativos

**Meu pessoal tem ferramentas de desenvolvimento de software de última geração; afinal lhes compramos os mais novos computadores.**

*Realidade:*

É preciso muito mais do que os mais recentes computadores para se fazer um desenvolvimento de software de alta qualidade.

---

# Mitos do Software - Administrativos

**Se nós estamos atrasados nos prazos,  
podemos adicionar mais programadores e  
tirar o atraso.**

**Realidade:**

O desenvolvimento de software não é um processo mecânico igual à manufatura.

Acrescentar pessoas em um projeto torna-o ainda mais atrasado. Pessoas podem ser acrescentadas, mas somente de uma forma planejada.



---

# Mitos do Software - do Cliente

**Uma declaração geral dos objetivos é suficiente para se começar a escrever programas - podemos preencher os detalhes mais tarde.**

*Realidade:*

Uma definição inicial ruim é a principal causa de fracassos dos esforços de desenvolvimento de software.

É fundamental uma descrição formal e detalhada do domínio da informação, função, desempenho, interfaces, restrições de projeto e critérios de validação.

---

# Mitos do Software - do Cliente

**Os requisitos de projeto modificam-se continuamente, mas as mudanças podem ser facilmente acomodadas, porque o software é flexível.**

**Realidade:**

Uma mudança, quando solicitada tardiamente num projeto, pode ser maior do que mais do que uma ordem de magnitude mais dispendiosa do que a mesma mudança solicitada nas fases iniciais.

---

# Mitos do Software - do Profissional

**Assim que escrevermos o programa e o colocarmos em funcionamento nosso trabalho estará completo.**

**Realidade:**

Os dados da indústria indicam que entre 50 e 70% de todo esforço gasto num programa serão despendidos depois que ele for entregue pela primeira vez ao cliente.

---

# Mitos do Software - do Profissional

**Enquanto não tiver o programa "funcionando", eu não terei realmente nenhuma maneira de avaliar sua qualidade.**

**Realidade:**

Um programa funcionando é somente uma parte de uma Configuração de Software que inclui todos os itens de informação produzidos durante a construção e manutenção do software.

---

# Engenharia de Software: Definição

- “Engenharia de Software é o estabelecimento e uso de sólidos princípios de engenharia para que se possa obter economicamente um software que seja confiável e que funcione eficientemente em máquinas reais”
- Resumindo... É METODOLOGIA!

---

# Exercícios:

1-Pesquisar, descrever e apresentar, (ao menos) 2 casos de problemas causados por erros em software que você conhece

2-Pesquisar, descrever e apresentar, (ao menos) 2 produtos (que não seja um PC) no qual o software faz a diferença.

3-Pressman, página 20, apresenta a relação, abaixo apresentada, de tipos/características de software. Explane cada um (Acrescente outros se desejar.):

Software Básico;

Software de Tempo Real;

Software Comercial;

Software Científico ou de Engenharia;

Software Embutido;

Software de Computador Pessoal;

Software de Inteligência Artificial e etc.

Atividade extra entrega (manuscrita) data: será discutida e acordada em aula com a turma)