

# Teste de Software

# Nivelamento



**PUC Minas**

Instituto de Ciências Exatas  
e Informática

Prof. Lesandro Ponciano

Departamento de Engenharia de Software  
e Sistemas de Informação (DES)

# Objetivos da Aula

---

- Analisar conceitos associados aos softwares
  - Definições, tipos, defeitos
- Discutir o desenvolvimento de software
  - Bem-sucedido e malsucedido
  - Êxito no desenvolvimento de software
  - Surgimento da engenharia de software
- Contextualizar testes na Engenharia de Software
  - Atividades

# Software

---

```
private void btnOpen_Click(object sender, EventArgs e)
{
    string file_name = "C:\\test1.txt";
    string textLine = "";

    System.IO.StreamReader objReader;
    objReader = new System.IO.StreamReader(file_name);

    do
    {
        textLine = textLine + objReader.ReadLine() + "\r\n";
    } while (objReader.Peek() != -1);

    textBox1.Text = textLine;

    objReader.Close();
}
```

# Softwares

---

... são elementos do sistema lógico, não do sistema físico

... são instruções (programas de computador) que quando executadas fornecem características, funções e desempenho desejados

... incluem estruturas de dados que possibilitam aos programas manipular informações adequadamente

... incluem informações descritivas sobre a operação e uso dos programas

# Produtos de Software

---

- Produtos de software podem surgir da necessidade de um **usuário** ou de um **mercado**
- Produtos sob encomenda (personalizados)
  - Ex.: Sítios web, controles de processos específicos de uma organização
- Produtos genéricos (de prateleira, *stand-alone*)
  - Ex.: Editores de texto, pacotes gráficos, ferramentas de gerenciamento de projetos, jogos

# Tipos de Software

---

- O tipo do software determina
  - Interfaces para **usuários** e proximidade com o **cliente**
  - Funcionalidades, eficiência e custo da implementação
- Alguns tipos de software
  - Software de sistema e de aplicação
  - Software de engenharia ou científico
  - Software embutido (ou embarcado)
  - Software de linhas de produto
  - Aplicações para Web
  - Software de inteligência artificial

# Algoritmos

---

... são “descrição de um conjunto de comandos que, obedecidos, resultam numa sucessão finita de ações” (Farrer et al. 1999)

... são “Sequência de passos computacionais que transformam uma entrada em uma saída” (Cormen et al. 2002)

... são "um processo sistemático para a resolução de um problema". (Szwarcfiter et al. 2010)

# Estrutura de Dados

---

- É a forma de armazenamento e organização de dados que é usada para resolver o problema
  - A estrutura pode afetar a eficiência da solução
- Existem estruturas de dados estáticas e dinâmicas
  - Estruturas de dados **estáticas** são compostas por um número fixo (finito) de elementos
  - Estruturas de dados **dinâmicas** podem ter um número indefinido de elementos



# Softwares Grandes

---

- Máquinas de busca, como Google
  - Algoritmos como *page rank* e *map-reduce*
- Redes sociais, como Facebook
  - Algoritmos de processamento de grafo
- Sistemas Peer-to-Peer (P2P), como BitTorrent
  - Algoritmos baseados em reputação como o “olho por olho dentes por dentes” (*tit-for-tat*)

# Desenvolvimento de Software



# Desenvolvimento de Software

- Desenvolver um software significa
  - Transformar a **necessidade** de um usuário ou de um mercado em um produto de software
  - **Elaborar** e **implementar** um sistema computacional
- Um software pode ser bem-sucedido ou malsucedido

# Software

---

## Bem-sucedido

- Atende às necessidades dos usuários
- Opera perfeitamente durante um longo período
- É fácil de modificar
- É fácil de utilizar

## Malsucedido

- Seus usuários estão insatisfeitos
- É propenso a uma diversidade de defeitos
- É difícil modificá-lo
- É difícil utilizá-lo

# Engenharia de Software

---

- “é 1) aplicação de uma abordagem **sistemática**, **disciplinada**, e **quantificável** no desenvolvimento, na operação e na manutenção de software, 2) estudo e abordagem do descrito em 1” (IEEE\*, 1993)

\*IEEE é o Instituto de Engenheiros Eletricistas e Eletrônicos (*Institute of Electrical and Electronics Engineers*)

# Processo de Desenvolvimento

- Desenvolver um software significa
  - Transformar a **necessidade** de um usuário ou de um mercado em um produto de software
  - **Elaborar** e **implementar** um sistema computacional
- Processo de Desenvolvimento de Software, ou simplesmente
  - Processo de Desenvolvimento
  - Processo de Software

# Processo de Software

---

- Um processo de software consiste de um conjunto de **atividades** realizadas por **pessoas** e cujo objetivo é o desenvolvimento ou evolução de um **artefato**, que é o software e sua documentação



# Atividades Genéricas

---

## ■ Pressman (2011)

- Comunicação
- Planejamento
- Modelagem
- Construção
- Emprego

**Especificação**

**Projeto e  
Implementação**

**Validação**

## ■ Sommerville (2011)

- Especificação
- Projeto e implementação
- Validação
- Evolução

**Evolução**



# Atividade Especificação

- Requisitos de Software
  - O que o software deve fazer, serviços que deve oferecer, restrições ao seu funcionamento
- Ou, ainda..
  - Condição ou capacidade **necessária a um usuário** para resolver um problema ou alcançar um objetivo
  - Condição ou capacidade que deve ser alcançada ou **possuída por um sistema** para satisfazer um contrato
  - Uma representação documentada de uma condição ou capacidade como nos itens 1 ou 2

# Requisitos Funcionais

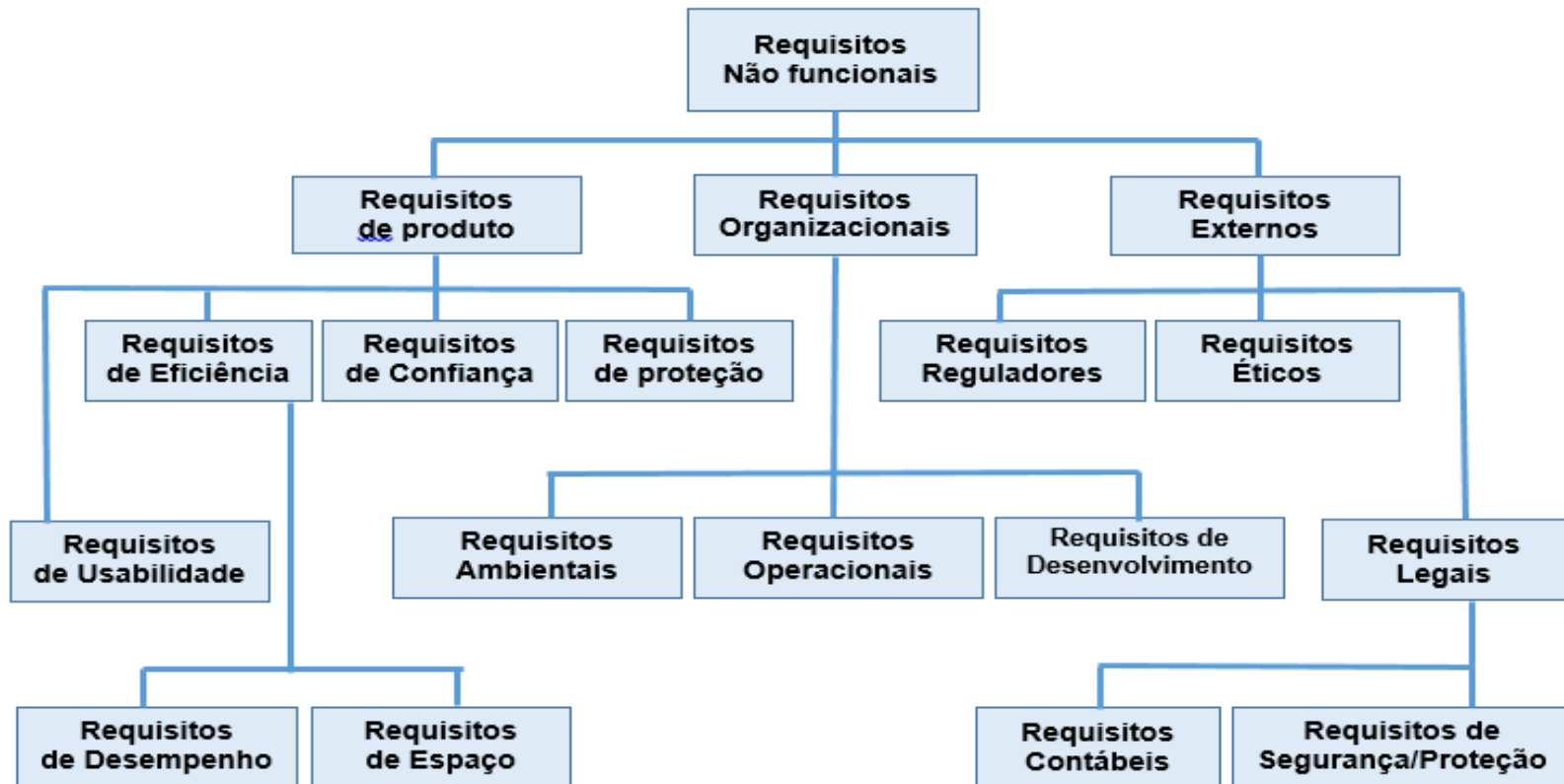
---

- Requisitos funcionais (RF) são declarações de
  - Serviços que o sistema deve oferecer
  - Como o sistema deve reagir a entradas específicas
  - Como o sistema deve se comportar em determinadas situações
- Estão diretamente relacionados aos **objetivos** do sistema

# Requisitos Não-Funcionais

- Requisitos não-funcionais (RNF),
  - **Restrições** aos serviços ou funções oferecidos pelo sistema, como tempo (*timing*), processo de desenvolvimento, plataforma, restrições legais
- Surgem de restrições de orçamento, políticas organizacionais, necessidade de interoperabilidade e fatores externos

# Tipos de Requisitos Não-Funcionais



# Atividade de Projeto e Implementação

- O software deve ser produzido para atender às especificações
- O projeto é uma extensão da especificação visando a implementação em um computador
  - É algo a ser compreendido pelo programador
- A implementação é a escrita do código
  - É uma tarefa relativamente simples se o projeto for bem-feito

# Atividade de Validação

---

- O software deve ser validado para se certificar de que ele atende os requisitos
  - Mostrar que o software se adéqua às especificações
  - “verificação e validação” / “teste de software”
- Inclui várias fases de teste do software
  - Testes feitos pelo próprio programador durante a programação
  - Testes feitos por equipes independentes
  - Testes manuais e automatizados

# Evolução

---

- O software deve evoluir para atender às necessidades de mudanças dos clientes
- A evolução inclui
  - Correção de defeitos
  - Inclusão de novas funcionalidades
  - Adaptação para novas plataformas

# Exercício de Fixação

---

- O que é um software?
- Quando o desenvolvimento de software é considerado bem sucedido?
- Qual a principal diferença entre a origem de defeitos em software e a origem de defeitos em hardware?
- Quais são as atividades genéricas de engenharia de software?