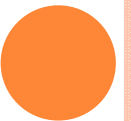


VAMOS FALAR DE
PIZZA!



DIA DE PIZZA

- Calabresa
- Americana
- Frango com Catupiry
- Quatro Queijos
- Filé
- Strogonoff
- Barriga Cheia
- Portuguesa
- Romeu e Julieta

R\$ 30,00



MONTE SUA PIZZA

R\$ 40,00

○ Escolha 5 ingredientes:

- Calabresa
- Frango
- Filé Mignon
- Milho
- Tomate
- Queijo “Mussarela”
- Queijo Cheddar
- Queijo Prato
- Queijo Catupiry
- Azeitonas
- Tomate
- Creme de Leite
- Ervilha
- Ovo cozido
- Palmito
- Orégano
- Cebola
- Presunto
- Batata Palha
- Pimentão
- Leite condensado
- Goiabada
- Banana
- Chocolate



MONTE SUA PIZZA

R\$ 50,00

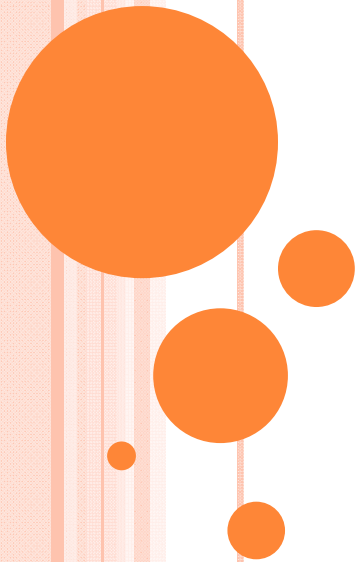
○ Escolha 10 ingredientes:

- Calabresa
- Frango
- Filé Mignon
- Milho
- Tomate
- Queijo “Mussarela”
- Queijo Cheddar
- Queijo Prato
- Queijo Catupiry
- Azeitonas
- Tomate
- Creme de Leite
- Ervilha
- Ovo cozido
- Palmito
- Orégano
- Cebola
- Presunto
- Batata Palha
- Pimentão
- Leite condensado
- Goiabada
- Banana
- Chocolate



ENGENHARIA DE SOFTWARE

REUSO DE SOFTWARE



REUSO DE SOFTWARE

- Reuso de Aplicações
- Reuso de Componentes
- Reuso de Objetos e Funções



REUSO DE SOFTWARE

- Em muitas disciplinas de engenharia, sistemas são projetados a partir da composição de componentes existentes que têm sido usados em outros sistemas.
- No entanto, é aceito atualmente que para produzir software com qualidade, mais rápido e com baixos custos, precisamos adotar um processo de design baseado no reuso sistemático de software.



ENGENHARIA DE SOFTWARE BASEADA EM REUSO

Reuso de sistemas de aplicações

Todo o sistema pode ser reutilizado pela sua incorporação, sem mudança, em outros sistemas (sistemas de prateleira) ou pelo desenvolvimento de famílias e aplicações.

Reuso de Componentes

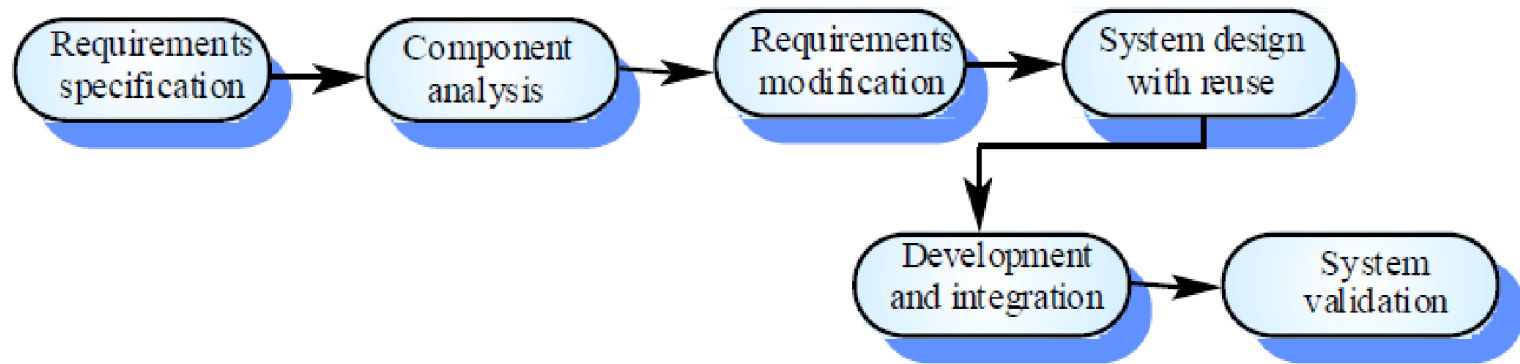
Componentes de uma aplicação que variam desde subsistemas até objetos isolados podem ser reutilizados.

Reuso de Funções

Componentes de software que implementam uma única função podem ser reutilizados



REUSO DE COMPONENTES



BENEFÍCIOS DO REUSO

Maior confiabilidade

Os componentes já foram experimentados e testados em sistemas que já estão funcionando.

Redução dos riscos de processo

Menos incertezas sobre os custos de desenvolvimento.

Uso efetivo de especialistas

Reuso de componentes ao invés de pessoas.



BENEFÍCIOS DO REUSO

Conformidade com padrões

Os padrões são embutidos ao se reutilizar componentes.

Desenvolvimento acelerado

Evita o desenvolvimento e validação, acelerando a produção



PROBLEMAS COM REUSO

- **Aumento nos custos de manutenção**

Elementos reusados do sistema podem tornar-se cada vez mais incompatíveis.

- **Falta de ferramentas de apoio**

O conjunto de ferramentas CASE podem não apoiar o desenvolvimento com reuso.

- **Síndrome do “não foi inventado aqui”**

Alguns engenheiros de software preferem reescrever componentes por que acreditam que podem aprimorá-los.



ABORDAGENS QUE DÃO SUPORTE AO REUSO

- Padrões de Projeto (Design patterns)
- Frameworks
- Desenvolvimento baseado em Aplicações
 - Integração de COTS (Commercial Off-The-Shelf)
 - Linhas de produtos



FATORES NO PLANEJAMENTO DO REUSO

- Cronograma para o desenvolvimento do software.
- Expectativa para o tempo de vida do software.
- Conhecimento, habilidades e experiência da equipe de desenvolvimento.
- Importância do software e seus requisitos não-funcionais.
- Domínio da aplicação.



DESIGN PATTERNS

- Um **padrão de projeto** ou **padrão de desenho** (do inglês **design pattern**) é uma solução geral reutilizável para um problema que ocorre com frequência dentro de um determinado contexto
- Não é um projeto finalizado que pode ser diretamente transformado em código fonte ou de máquina, ele é uma descrição ou modelo de como resolver um problema que pode ser usado em muitas situações diferentes.
- São melhores práticas formalizadas que o programador pode usar para resolver problemas comuns quando projetar uma aplicação ou sistema.
- Padrões de projeto residem no **domínio de módulos e interconexões**. Em um nível mais alto há padrões arquiteturais que são maiores em escopo, usualmente descrevendo um padrão global seguido por um sistema inteiro.



FRAMEWORKS

- ***Frameworks*** representam um projeto de subsistema constituído de um conjunto de classes abstratas e concretas e da interface entre elas.
- ***Frameworks*** são entidades moderadamente grandes que podem ser reutilizadas.
- Os subsistemas são implementados **com o acréscimo de componentes** e o fornecimento da implementação concreta das classes abstratas nos *frameworks*.



REUSO DE APLICAÇÕES

- Envolve o reuso de uma aplicação completa através da configuração de um sistema para um ambiente ou através da integração de 2 ou mais sistemas para criar uma aplicação.
- Algumas abordagens:
 - Integração de COTS;
 - Desenvolvimento de Linhas de Produto



REUSO DE COTS

- **COTS** - Commercial Off-The-Shelf Systems.
- **COTS** são usualmente aplicações completas que oferecem uma API (Application Programming Interface).

O principal benefício é o desenvolvimento rápido e a baixo custo.



LINHAS DE PRODUTO DE SOFTWARE

- Aplicações com funcionalidade genérica que podem ser adaptadas e configuradas para uso em um contexto específico.
- Tipos de adaptações:
 - Especialização de plataforma e de Ambiente;
 - Especialização Funcional
 - Especialização de Processo



ESPECIALIZAÇÃO DE LINHAS DE PRODUTO DE SOFTWARE

- **Especialização de plataforma e ambiente**
 - Versões da aplicação são desenvolvidas para plataformas diferentes.
 - Versões são criadas para lidar com ambientes operacionais e dispositivos específicos.



ESPECIALIZAÇÃO DE LINHAS DE PRODUTO DE SOFTWARE

- **Especialização funcional**
 - Versões são criadas para clientes específicos com requisitos diferentes.
- **Especialização de processo**
 - O sistema é adaptado para lidar com processos de negócio específicos.



REFERÊNCIAS

- **SOMMERVILLE**, Ian Engenharia de Software 8ª ed. São Paulo: Pearson, 2007.

