

# Como desenvolver sistemas de informação

Capítulo 1

Paulo C. Masiero/Vania Neves  
2013

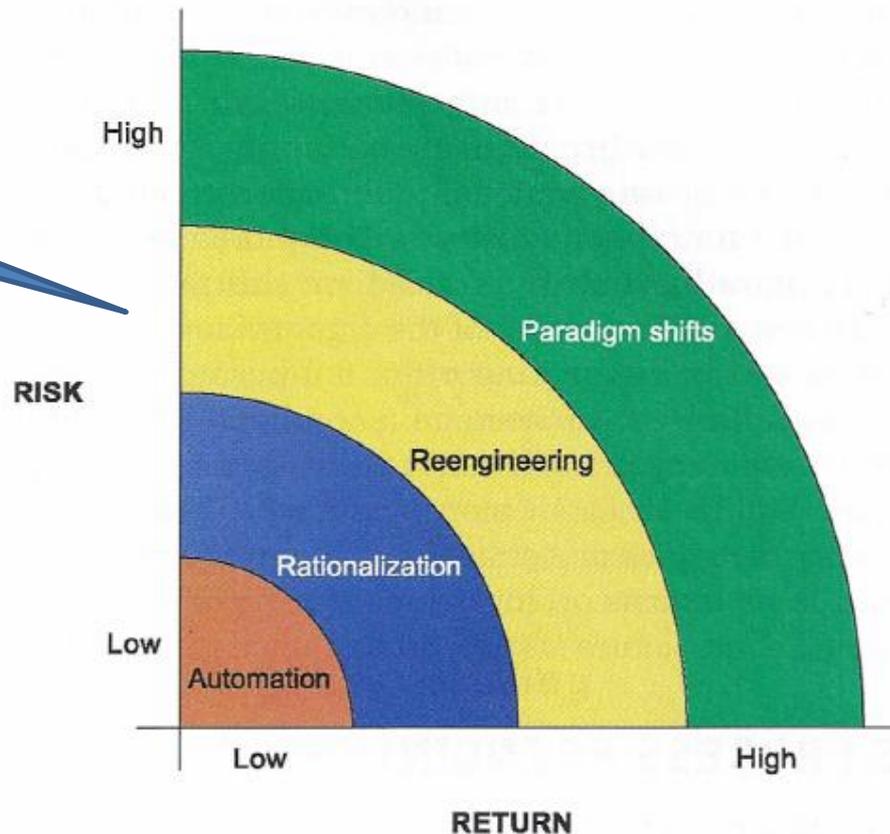
# Como desenvolver sistemas de informação

- A construção de um novo sistema de informação é um tipo de mudança organizacional planejada.
- A introdução de um novo sistema de informação envolve muito mais que novo hardware e software
  - Inclui mudanças em trabalhos, habilidades, gerenciamento e organização
- O projeto de um novo sistema de informação também “reprojeta” a organização.
- Os projetistas de sistema devem entender como um sistema irá afetar um processo de negócio específico e toda a organização .

# Desenvolvimento de sistemas e mudança organizacional

- A tecnologia da informação pode promover vários graus de mudança organizacional

**FIGURE 13-1 ORGANIZATIONAL CHANGE CARRIES RISKS AND REWARDS**



Redesign

# Desenvolvimento de sistemas e mudança organizacional

- Automação

- Envolve ajudar os funcionários a executar suas tarefas mais eficiente e efetivamente.

Automação  
Inicial

- Calcular salários e folhas de pagamento, etc..

- Pode revelar novos gargalos na produção e tornar o arranjo de procedimentos e estruturas dolorosamente pesado

- Racionalização de procedimentos

- Simplifica os procedimentos operacionais padrão

- Exemplo: racionalização a partir do uso de padrões de qualidade.

# Desenvolvimento de sistemas e mudança organizacional

- Reengenharia do processo de negócio
  - Os processos de negócio são analisados, simplificados e reprojutados.
  - É muito mais ambicioso que a racionalização de procedimentos e requer uma nova visão de como o processo será organizado.
  - Exemplo da Ford: Reengenharia do Processo de Contas a pagar:
    - Reduziu 75% de um total de 500 pessoas.
    - Antes os funcionárias gastavam muito tempo resolvendo discrepâncias entre os pedidos, documentos recebidos e faturas.

# Desenvolvimento de sistemas e mudança organizacional

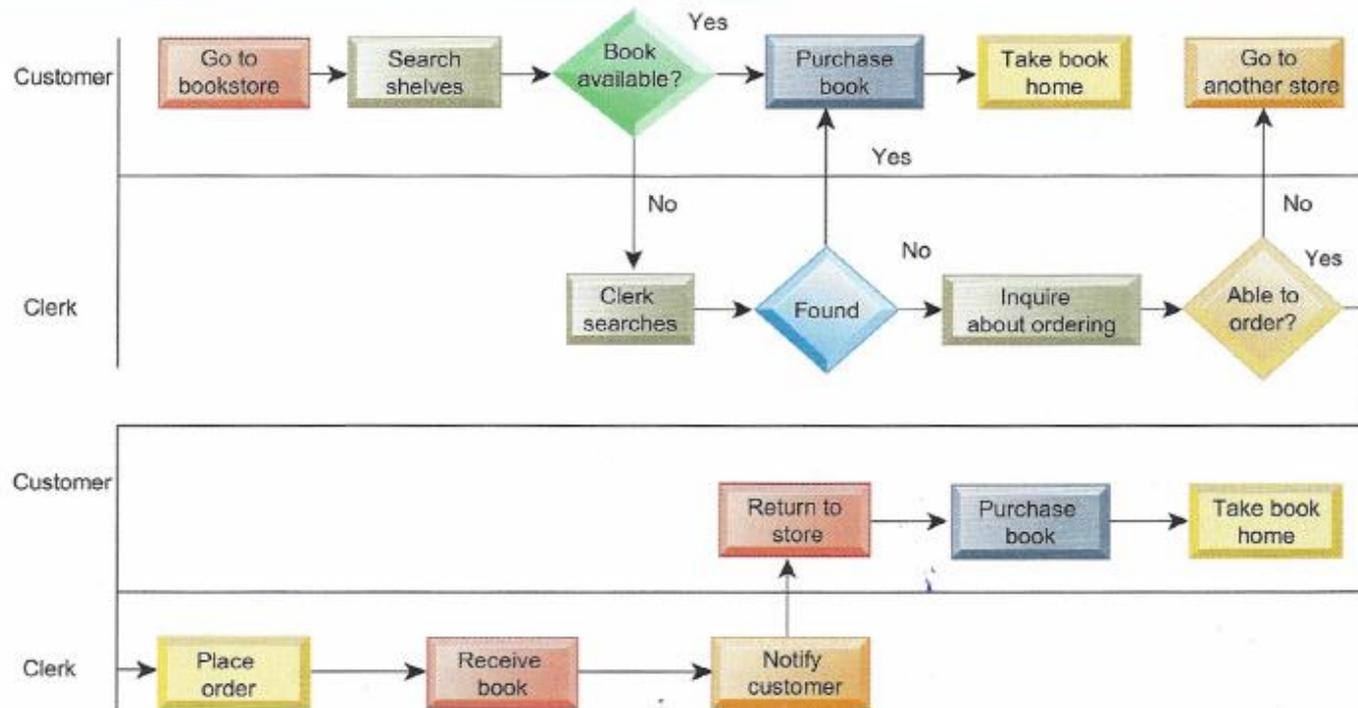
- Mudança de paradigma
  - Envolve repensar a natureza do negócio e a natureza da organização.
  - Exemplo: A empresa de transporte de cargas de longa distância Schneider National, mudou o seu negócio para gerenciar a logística de transporte de outras empresas.

# Reengenharia de processos de negócio

- Gerenciamento do processo de negócio (BPM) fornece uma série de ferramentas e metodologias para analisar processos existentes, projetar novos processos e otimizá-los.
  - Nunca é concluído
    - Melhoramento do processo requer mudanças contínuas
- 5 passos:
  - Identificar os processos de mudança
    - Entender quais processos precisam ser melhorados
  - Analisar os processos existentes
    - Processos de negócios existentes devem ser modelados e documentados
      - “As-is”

# Reengenharia de processos de negócio

**FIGURE 13.2 AS-IS BUSINESS PROCESS FOR PURCHASING A BOOK FROM A PHYSICAL BOOKSTORE**

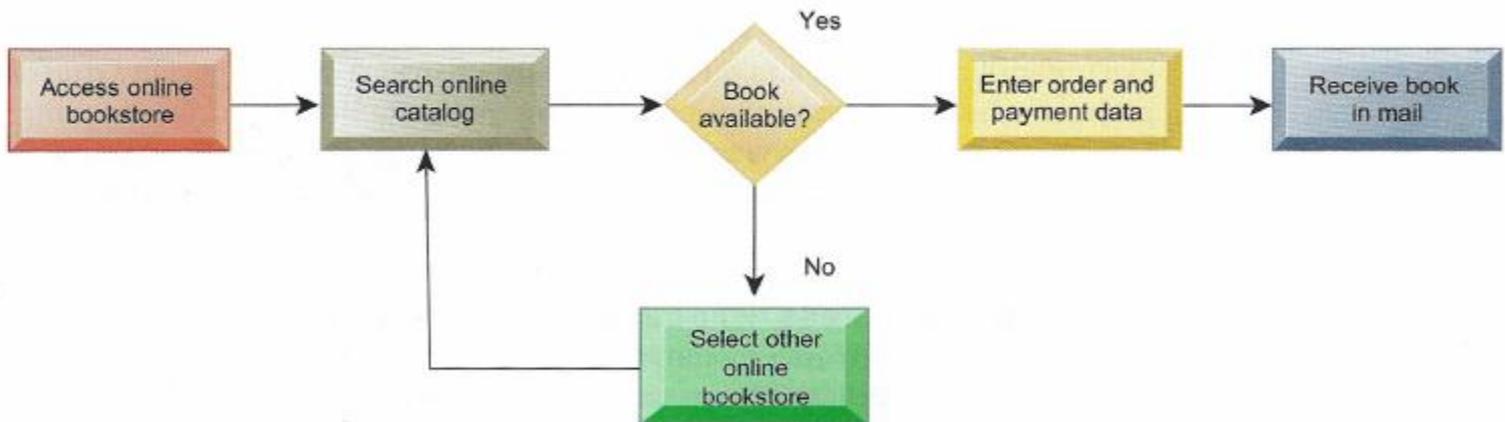


Purchasing a book from a physical bookstore requires many steps to be performed by both the seller and the customer.

# Reengenharia do processo de negócio

- Projetar o novo processo
  - Tentar melhorar o projeto existente projetando um novo
  - “To-be”

**FIGURE 13.3 REDESIGNED PROCESS FOR PURCHASING A BOOK ONLINE**

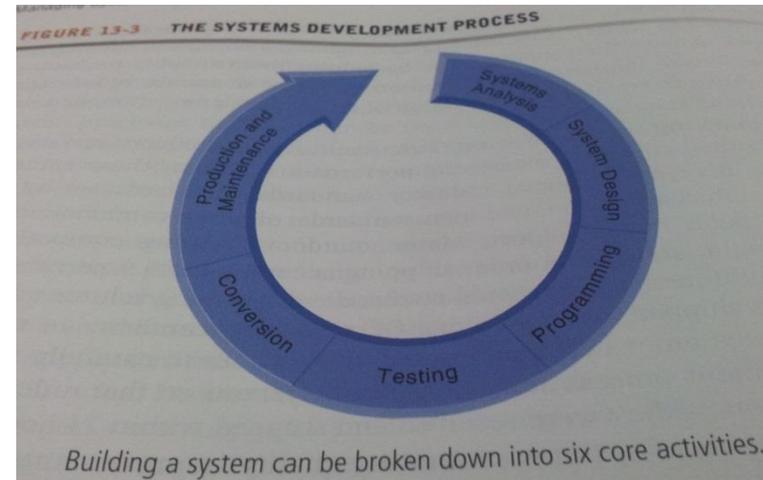


# Reengenharia do processo de negócio

- Implementar o novo processo
  - Uma vez modelado e analisado, o novo processo deve ser traduzido em um novo conjunto de procedimentos e regras de trabalho
- Medição contínua
  - Processos podem se deteriorar ao longo do tempo
  - Empregados podem voltar aos métodos antigos
  - Não ter mais a mesma eficácia se a empresa sofre outras mudanças

## Visão Geral dos Sistemas de Desenvolvimento

- Sistemas de desenvolvimento é um tipo estruturado da solução do problema com atividades distintas
  - Análise de sistemas
  - Projeto de sistemas
  - Programação
  - Teste
  - Conversão
  - Produção e manutenção



# Análise de Sistema

- É a análise do problema que a organização tenta resolver com um sistema de informação.
- Consiste de:
  - definir o problema
  - identificar suas causas
  - especificar uma solução
  - identificar o conjunto de requisitos que devem ser atendidos por uma dada solução.
- Descreve o que um sistema deve fazer para encontrar os requisitos de informação

# Análise de Sistema

- O analista de sistemas deve:
  - Identificar os responsáveis por (dono/cliente) e usuários primários dos dados.
  - Hardware e software existente
- Em seguida:
  - Detalhar os problemas do sistemas existentes.
  - Identificar os objetivos e áreas problemáticas para encontrar uma solução.

# Análise de Sistema (Cont.)

- Realizar um estudo de viabilidade
  - Técnica
  - Econômica
- Definir várias possíveis soluções (em linhas gerais)
- Para cada solução:
  - Relatório escrito
  - Custos e benefícios
  - Vantagens e desvantagens
- A gerência é que deve determinar a combinação desses elementos a escolher.

# Projeto de Sistemas

- Mostra **como** o sistema irá cumprir os objetivos definidos na fase de análise
- Produz um modelo ou plano geral do sistema
- Consiste de uma especificação que dá forma e estrutura ao sistema
- A especificação trata dos aspectos organizacionais, gerenciais e componentes tecnológicos da solução do sistema

# Projeto de Sistemas

- Os requisitos de informação são os que direcionam o esforço de desenvolvimento do sistema.
- O papel dos usuários é assegurar que o sistema atenda as suas necessidades de informação e prioridades de negócio
  - não crie viés do pessoal técnico.
- A participação do usuário no processo de desenvolvimento aumenta seu entendimento e aceitação do sistema.

# Programação

- A especificação preparada na atividade de projeto é traduzida para o código de programa.
- Hoje, muitas empresas não mais programam o seu próprio sistema.
  - Compram de terceiros os softwares que atendem aos requisitos.

# Programação (Cont.)

- Isso pode acontecer na forma de:
  - Pacotes de software de empresas de software
  - Provedores de serviços de aplicação
  - Empresas terceirizadas que desenvolvem a aplicação sob medida para o cliente.

# TESTE

- O teste procura responder à seguinte questão: o sistema produzirá os resultados desejados sob condições conhecidas?
- Isto contribui para a qualidade do sistema
- O teste consome muitos recursos:
  - Casos de teste devem ser preparados
  - Os resultados do teste devem ser analisados
  - Correções podem ter que ser feitas
- É muito perigoso negligenciar o teste.

# TESTE (Continuação)

- A atividade de teste pode ser dividida em três fases:
  - Teste de unidade
  - Teste de integração
  - Teste de Sistema
- O teste de sistema pode ter propósitos diferentes. Exemplos:
  - Teste Alfa, Teste Beta
  - Teste de aceitação
  - Teste de stress

# Implantação (ou Conversão)

- É o processo de mudar do sistema antigo pelo novo (Ou: implantar o novo sistema)
- Há quatro estratégias principais de conversão:
  - Em paralelo
  - Transição direta
  - Projeto piloto: introduz em uma área limitada
  - Em fases
    - Por funções
    - Por unidades organizacionais

# Implantação (ou Conversão)

- A mudança exige que os usuários sejam treinados no novo sistema.
- **Documentação detalhada** deve estar disponível informando como o sistema funciona
  - Do ponto de vista técnico
  - Do ponto de vista do usuário

# Produção e Manutenção

- Depois que o processo de manutenção termina, diz-se que o sistema está em operação.
- Então, ele é executado ou operado, diariamente.
- Em alguns casos pode haver uma auditoria de pós-implantação, gerando um documento de análises e recomendações.

# Manutenção

- Depois da implantação e dos ajustes iniciais, o sistema também pode passar por manutenções.
- Enquanto o sistema está em operação podem ocorrer:
  - Falhas no sistema que precisam ser corrigidas.
  - Mudanças no hardware e software básico
  - Novos requisitos surgirem
  - O Desempenho da operação precisa ser melhorado, etc.

# Manutenção (Cont.)

- De um modo geral, as manutenções se dividem em:
  - 20% para corrigir emergências relativas à operação do sistema
  - 60% por necessidades de melhorias no sistema (novos requisitos)
  - 20% em mudanças relacionadas a software, arquivos, relatórios, hardware, software básico...
- Apenas as categorias 2 e 3 podem se beneficiar de um melhor processo de análise e projeto.

# Um resumo das atividades

**TABLE 13-2 SYSTEMS DEVELOPMENT**

CORE ACTIVITY	DESCRIPTION
Systems analysis	Identify problem(s) Specify solutions Establish information requirements
Systems design	Create design specifications
Programming	Translate design specifications into program code
Testing	Unit test Systems test Acceptance test
Conversion	Plan conversion Prepare documentation Train users and technical staff
Production and maintenance	Operate the system Evaluate the system Modify the system

# Análise e Projeto de Sistema: Modelos

- Metodologias estruturadas
  - Técnicas são passo a passo
  - Análise e projeto estruturado
  - DFD
- Desenvolvimento orientado a objeto
  - Utiliza o objeto como a unidade básica de desenvolvimento dos sistemas
  - Análise e projeto orientado a objetos: Processo Unificado, Rational, Larman
- Engenharia de software assistida por computador (CASE)
  - Fornece ferramentas de software para automatizar as metodologias
    - Reduzir o número de trabalho repetitivo que o desenvolvedor precisa fazer

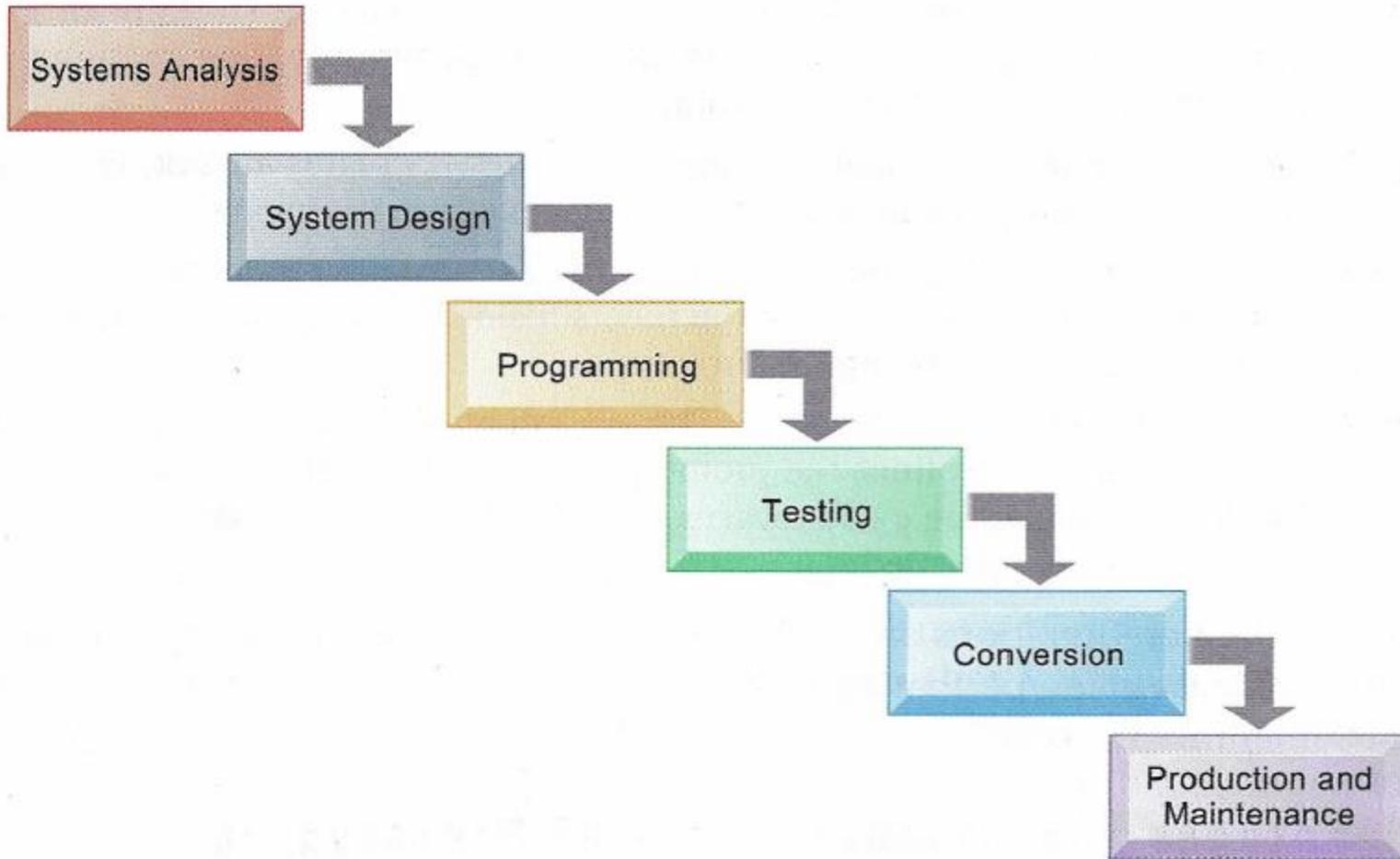
# Abordagens alternativas para desenvolvimento de sistemas

- Sistemas diferem em termos de:
  - Tamanho
  - Complexidade Tecnológica
  - Problemas Organizacionais
- Métodos alternativos
  - Ciclos de vida de sistemas tradicionais
  - Prototipação
  - Pacotes de aplicativos de software
  - Desenvolvimento pelo usuário final
  - Terceirização

## Ciclo de vida tradicional de desenvolvimento de sistemas

- Método mais antigo para o desenvolvimento de SI
- Abordagem em fases
- Divisão muito formal entre usuários e especialistas em SI
- Utilizada para desenvolvimento de sistemas de grande complexidade
  - Requer análise de requisitos rigorosa e formal
- Pode ser custosa, consumir muito tempo e inflexível
- Pode encorajar o “congelamento” das especificações relativamente cedo no processo de desenvolvimento

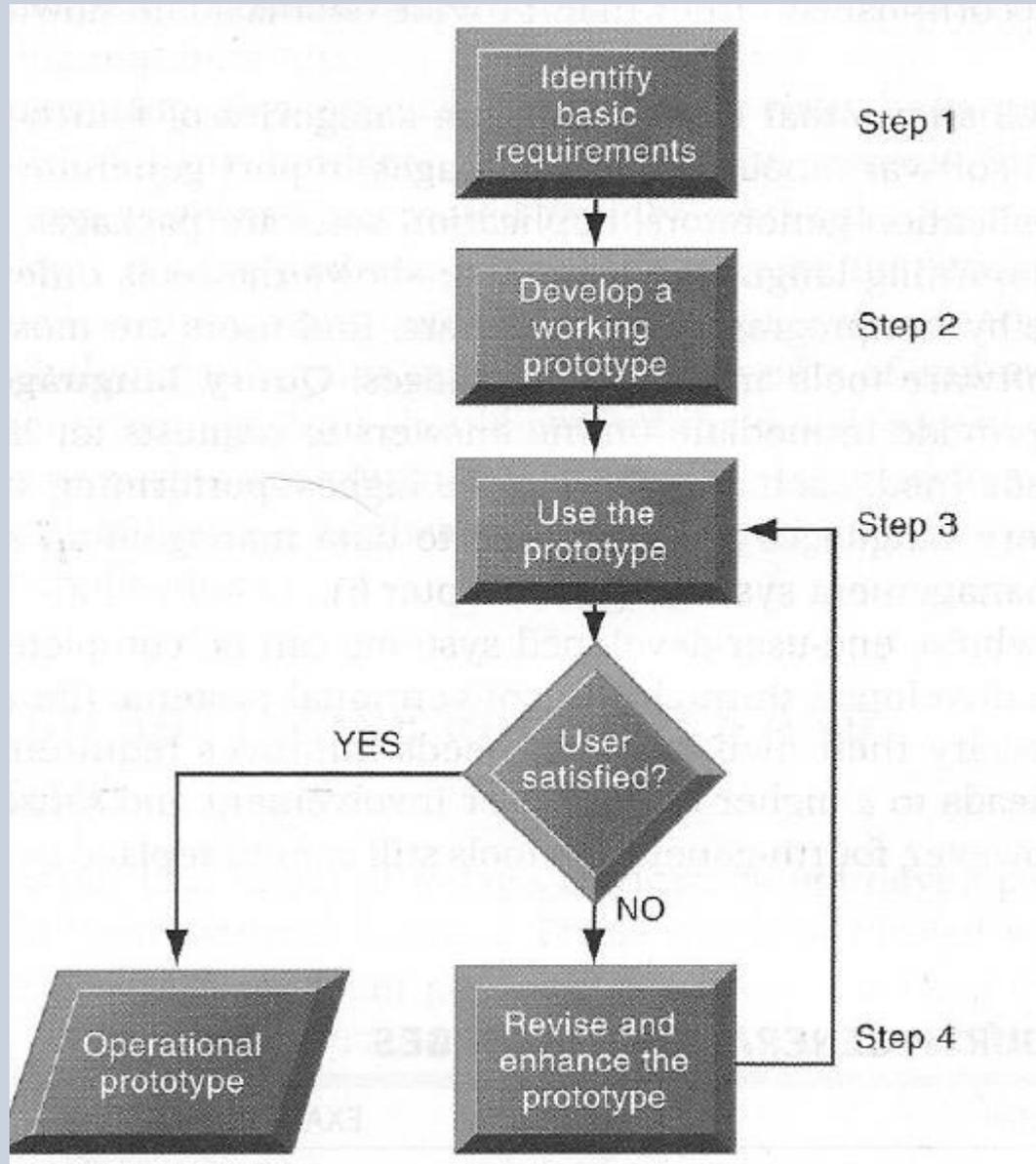
**FIGURE 13.9 THE TRADITIONAL SYSTEMS DEVELOPMENT LIFE CYCLE**



# Prototipação

- Construção rápida e barata de um sistema experimental para os usuários avaliarem
  - Usuários podem ter uma ideia melhor de seus requisitos
- Vantagens
  - Útil quando há incertezas em relação aos requisitos
  - Mais provável que se produza sistemas que cumpra com os requisitos do usuário
- Desvantagens
  - Pode encobrir passos essenciais no desenvolvimento do sistema
  - Se o protótipo funciona adequadamente, o usuário não vê necessidade de construir um novo sistema
    - Pode não acomodar grande quantidade de dados ou grande número de usuários, documentação inadequada, etc.

# Prototipação



# Desenvolvimento pelo Usuário Final

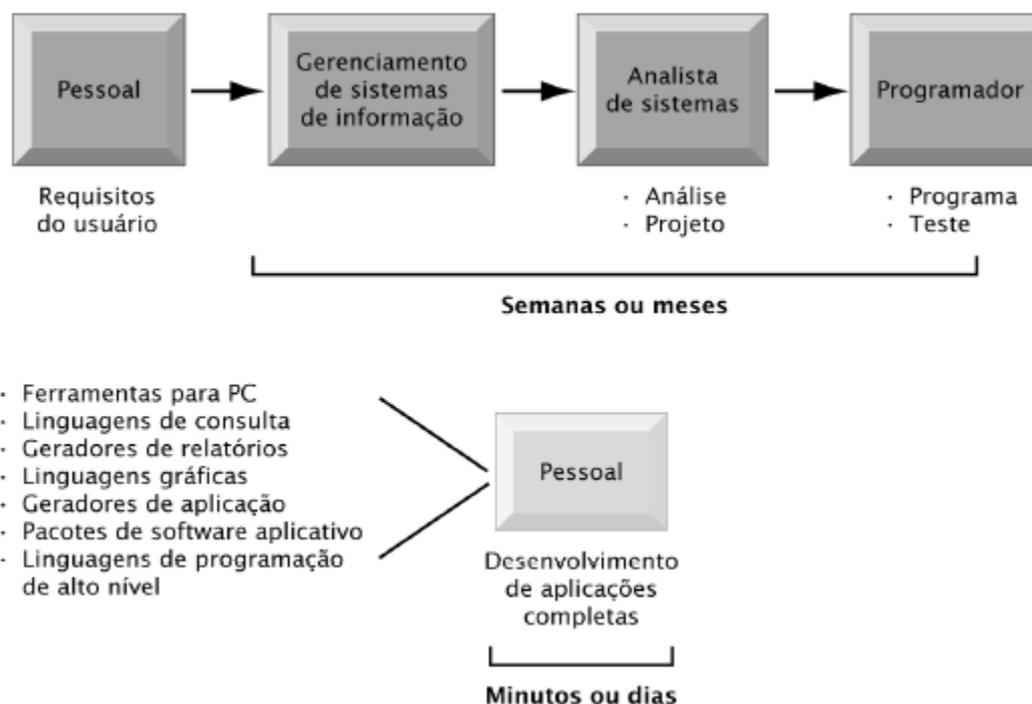
- Desenvolvimento de sistemas simples pelos usuários finais com pouca ou nenhuma ajuda de especialistas técnicos
  - Possível graças ao uso de linguagens de quarta geração
    - Ferramentas de software para PC
    - Query languages
    - Geradores de relatórios
    - Linguagens gráficas,
    - Linguagens de programação de bastante alto nível
- Concluído mais rapidamente do que os sistemas desenvolvidos com ferramentas convencionais
- Riscos organizacionais

# Desenvolvimento pelo Usuário Final

**Figura 11.7**

Desenvolvimento pelo usuário final.

*Ferramentas de software de quarta geração possibilitam aos usuários finais desenvolver sistemas simples que podem ser concluídos mais rapidamente do que aqueles desenvolvidos por meio do tradicional ciclo de vida de desenvolvimento de sistemas.*





## Pacotes Aplicativos de Software

- Como muitos processos de negócios são os mesmos nas empresas e são relacionados às funções básicas, é possível comprar soluções prontas que podem ser adaptadas.
  - Exemplos: ERPs, Suítes ...
- Capacidades de adaptação
- Adaptabilidade vs Flexibilidade de Processos de negócios

## Pacotes Aplicativos de Software

- Quando esse tipo de solução é adotada, a Análise de Sistema deve incluir um esforço de avaliação do pacote:
  - Funções oferecidas pelo pacote
  - Flexibilidade
  - Facilidade de uso
  - Recursos de hardware e software
  - Requisitos de bancos de dados
  - Instalação e manutenção
  - Qualidade do vendedor
  - Documentação
  - Custo

## Pacotes Aplicativos de Software

- A avaliação do pacote é baseada numa chamada de propostas (ou Request for Proposals – RFP), que é uma lista detalhada de questões submetidas aos vendedores de pacotes.
- Ao adotar uma solução desse tipo, a empresa perde o controle sobre o processo de projetar o sistema de informação (o pacote).

## Terceirização (outsourcing)

- Se uma empresa não quer usar seus próprios recursos para construir ou operar seus SIs, ela pode contratar (terceirizar) o trabalho para uma organização externa que é especializada em prestar esse serviço. Exemplos de formas de terceirização:
  - Computação em nuvem (cloud computing)
  - Software como serviço (Software as Service)

# Terceirização

- O relacionamento entre a empresa e o terceirizado é baseado em um contrato que regula os serviços a serem prestados
  - Contrato de nível de serviço (Service Level Agreement – SLA).
- Um SLA especifica:
  - Natureza e nível dos serviços
  - Critérios de desempenho
  - Provisões de suporte
  - Provisões para recuperação de desastres

## Terceirização (outsourcing) (Cont.)

- Outra forma: contratar um vendedor externo para projetar e construir o sistema, mas a empresa o opera em seus próprios computadores (pode ou não fazer manutenção – propriedade do código fonte)
- A terceirização pode ser doméstica ou internacional (offshore) → menores custos?

## Terceirização (outsourcing) (Cont.)

- Muitas empresas subestimam o custo de identificar e avaliar vendedores de TI, mudar de vendedor, etc.
- Os custos envolvem:
  - documentação de requisitos
  - Preparar e executar RFP
  - Despesas de viagens
  - Negociação de contratos
  - Gestão de projeto
- Na prática, demora de 3 meses a 1 ano para transferir o trabalho e o vendedor entender o negócio.

## Terceirização (outsourcing) (Cont.)

- Terceirização internacional envolve ainda:
  - Lidar com diferenças culturais (diminui a produtividade)
  - Tratar as questões de recursos humanos
- Todos esses custos “escondidos” drenam os possíveis benefícios antecipados da terceirização
- Em especial, deve-se ter muito cuidado ao terceirizar aplicações que fornecem algum tipo de vantagem competitiva.

### Custo total do outsourcing internacional

Custo do contrato de outsourcing	\$10.000,000			
Custos ocultos	Cenário otimista	Custo adicional (\$)	Cenário pessimista	Custo adicional (\$)
1. Seleção de fornecedores	0,2%	20.000	2%	200.000
2. Custos de transição	2%	200.000	3%	300.000
3. Demissões e retenção	3%	300.000	5%	500.000
4. Perda de produtividade/problemas culturais	3%	300.000	27%	2.700.000
5. Aperfeiçoamento de processos de desenvolvimento	1%	100.000	10%	1.000.000
6. Gestão do contrato	6%	600.000	10%	1.000.000
<b>Custos adicionais totais</b>		<b>1.520.000</b>		<b>5.700.000</b>
	Contrato em aberto (\$)	Custo adicional (\$)	Custo total (\$)	Custo adicional
Custo total do outsourcing no cenário otimista	10.000.000	1.520.000	11.520.000	15,2%
Custo total do outsourcing no cenário pessimista	10.000.000	5.700.000	15.700.000	57,0%

#### Figura 11.8

Custo total do outsourcing internacional.

*Se uma empresa gastar 10 milhões de dólares em contratos de outsourcing internacional, essa empresa na verdade gastará 15,2 por cento a mais em custos extras, mesmo no cenário mais otimista. No cenário mais pessimista, em que existe uma queda drástica na produtividade, combinada com custos de demissão e transição excepcionalmente altos, a empresa pode esperar desembolsar 57 por cento a mais em custos extras, além dos 10 milhões pagos pelo contrato internacional.*

## Desenvolvimento de Aplicações para a Empresa Digital

- As empresas precisam atualmente de agilidade e escalabilidade
- Responder mais rapidamente às oportunidades
- Ciclos de desenvolvimento mais rápidos
- Busca de alternativas além de ERPs e pacotes

## Desenvolvimento Rápido de Aplicações para E-business (RAD)

- Termo usado para descrever o processo de criação de sistemas aptos a funcionar em curto espaço de tempo
- Engloba o uso de programação visual e outras ferramentas para construir interfaces, prototipação , geração de código, trabalho em grupos muito unidos etc.

## Projeto conjunto de aplicações (Joint application design – JAD)

- Usado para acelerar a elicitação dos requisitos de informação
- E para desenvolver o projeto inicial do sistema.
- Usuários finais e especialistas em sistemas de informação trabalhando juntos no projeto

## Desenvolvimento ágil de software

- Foca a entrega rápida de software.
- Divide um projeto grande em pequenos sub-projetos que são entregues rapidamente usando iterações e feedback contínuo.
- Cada subprojeto é responsabilidade de um grupo (ciclos incrementais)
- Foca comunicação face-a-face ao invés de documentos escritos.
- Exemplos: XP e SCRUM

# Desenvolvimento baseado em Componentes

- Para acelerar ainda mais a criação de software, grupos de objetos tem sido reunidos para fornecer componentes de software para funções comuns que podem ser combinadas para criar aplicações de negócio em larga escala.
  - Interface gráfica do usuário
  - Pedidos on-line
- Combinação de componentes disponíveis comercialmente
  - Carrinhos de compras
  - Autenticação de usuário
  - Motores de busca

## Web Services e Computação Orientada a Serviço

- Componentes de software reusáveis que utiliza XML ou outros protocolos e padrões abertos que possibilitam que uma aplicação se comunique com outra sem nenhum requerer nenhuma programação customizada para compartilhar dados e serviços
- Pode ser utilizada como uma ferramenta para construir novos SI ou melhorar existentes
- Podem executar determinadas funções por conta própria
  - Ou encarregar que outros web services completem transações mais complexas
- Exemplo: Nota Fiscal Eletrônica

## Desenvolvimento de Aplicações Móveis

- Diferente do desenvolvimento para computadores com telas grandes.
- Aplicações móveis devem
  - ser otimizadas para as tarefas que devem executar
  - não fazer muitas tarefas
  - projetadas para usabilidade.
- Prioridade: economizar recursos.

## Desenvolvimento de Aplicações Móveis

- As aplicações devem ser projetadas para cada sistema principal: Android, iPhone/iPad e Windows phone
- A aplicação reconhece que celular o usuário está usando.

# Desenvolvimento de Aplicações Móveis

- Usar “responsive Web Design”
  - Mudar o leiaute de acordo com a resolução da tela.
  - Otimizar o projeto para diferentes contextos de visão.

## Desenvolvimento de Aplicações Móveis

- O desenvolvimento de aplicações móveis pode ser terceirizado.
- Exemplo: Red Foundry (cobra até 20.000 dólares para fazer uma aplicação)