

Lição 1



Introdução à Engenharia de Software – Parte 2

Modelo de Processo de Software

- Estratégia que a equipe de desenvolvimento incorpora
- Escolhido baseado na natureza do projeto e aplicação

Tipos de Modelos de Processo de Software

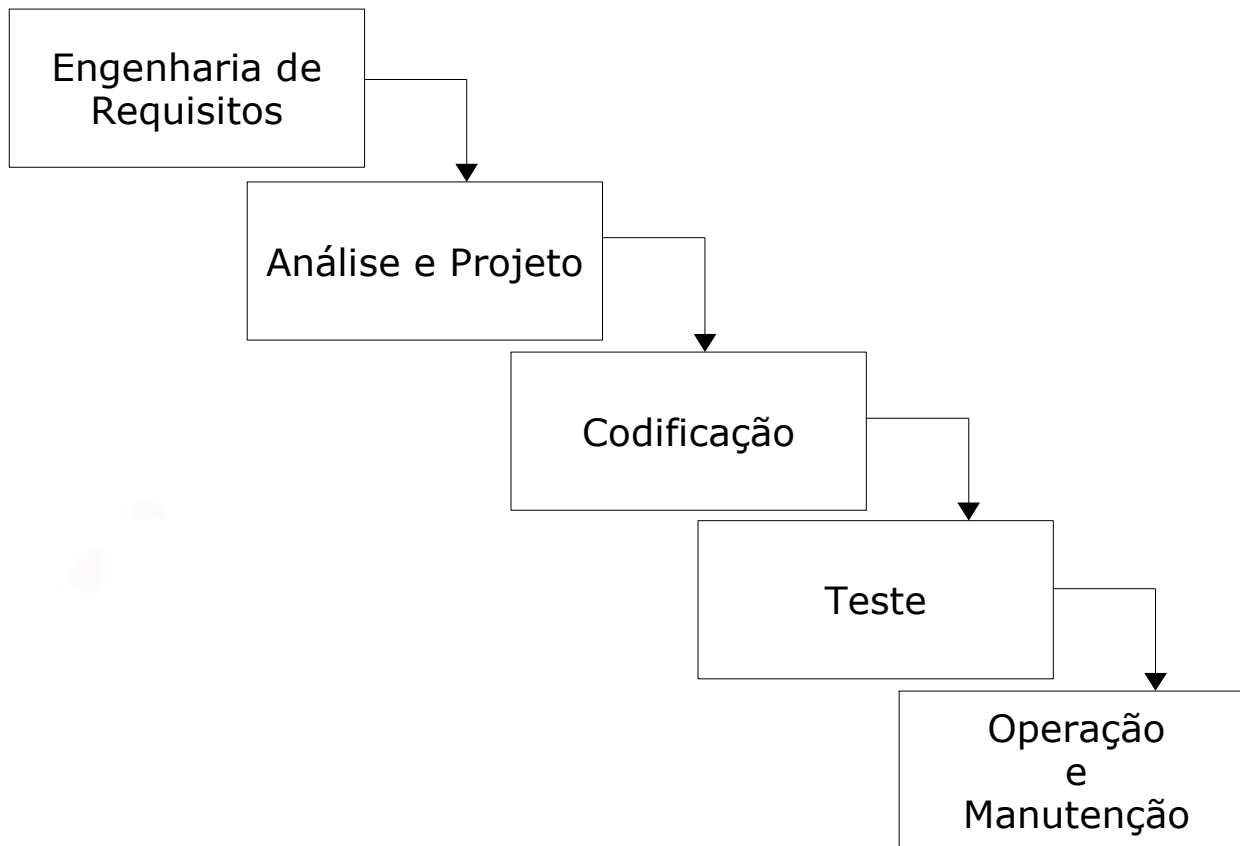
- Modelo Linear Seqüencial
- Modelo de Prototipação
- Modelo RAD (Desenvolvimento Rápido de Aplicações)
- Modelo de Processo de Software Evolucionário
 - Modelo Incremental
 - Modelo Espiral
 - Modelo de Montagem Baseada em Componentes
 - Modelo de Desenvolvimento Concorrente
- Modelo de Métodos Formais
- Técnicas de Quarta Geração



Modelo Linear Seqüencial

- Conhecido como modelo cascata ou ciclo de vida clássico
- Sugere uma abordagem sistemática e seqüencial para o desenvolvimento de software

Modelo Linear Seqüencial



Modelo Linear Seqüencial

- **Vantagens:**

- Melhor que nada
- Fornece uma base para outros Modelos de Engenharia de Software

- **Desvantagens:**

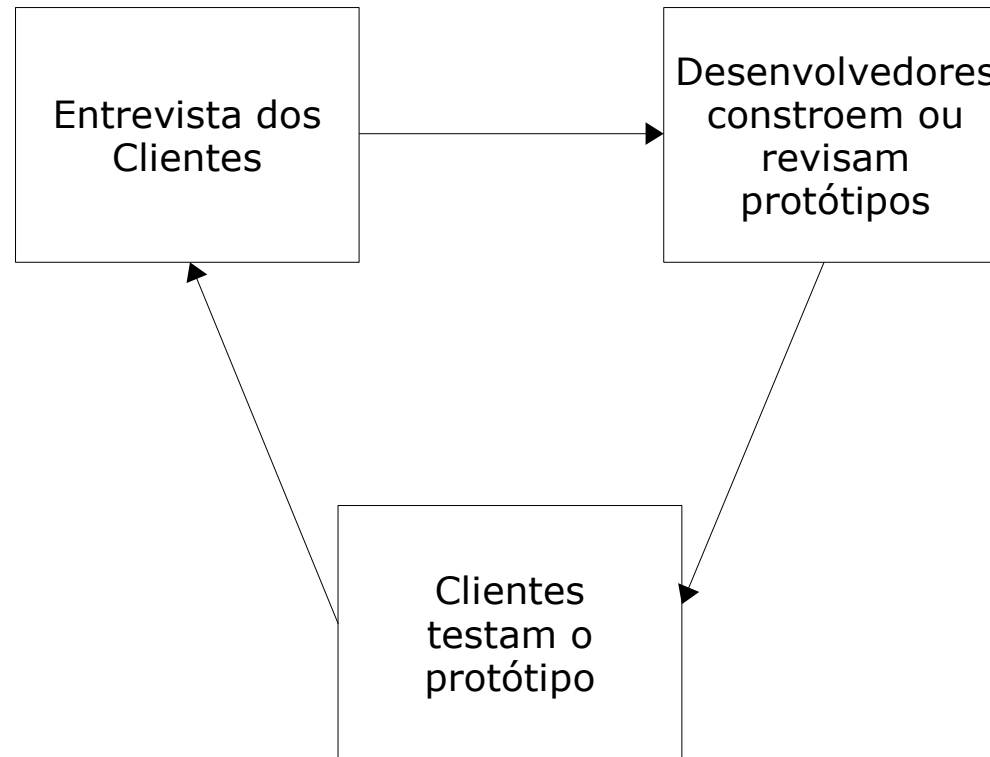
- Projetos reais raramente seguem uma seqüência
- Clientes têm dificuldade de definir todos os requisitos
- Envolvimento do cliente ocorre no início e fim do projeto
- Clientes devem ter paciência
- Desenvolvedores são com freqüência atrasados sem necessidade



Modelo de Prototipação

- Melhor abordagem para as seguintes situações:
 - Um cliente define um conjunto de objetivos gerais para o software mas não identifica detalhes de entrada, processamento ou requisitos de saída
 - Desenvolvedor não têm certeza da eficiência do algoritmo, da adaptabilidade do sistema operacional ou da forma que a interação homem-computador deve ser

Modelo de Prototipação



Modelo de Prototipação

- **Vantagens:**

- Usuários vêem a aparência atual do sistema

- **Desvantagens:**

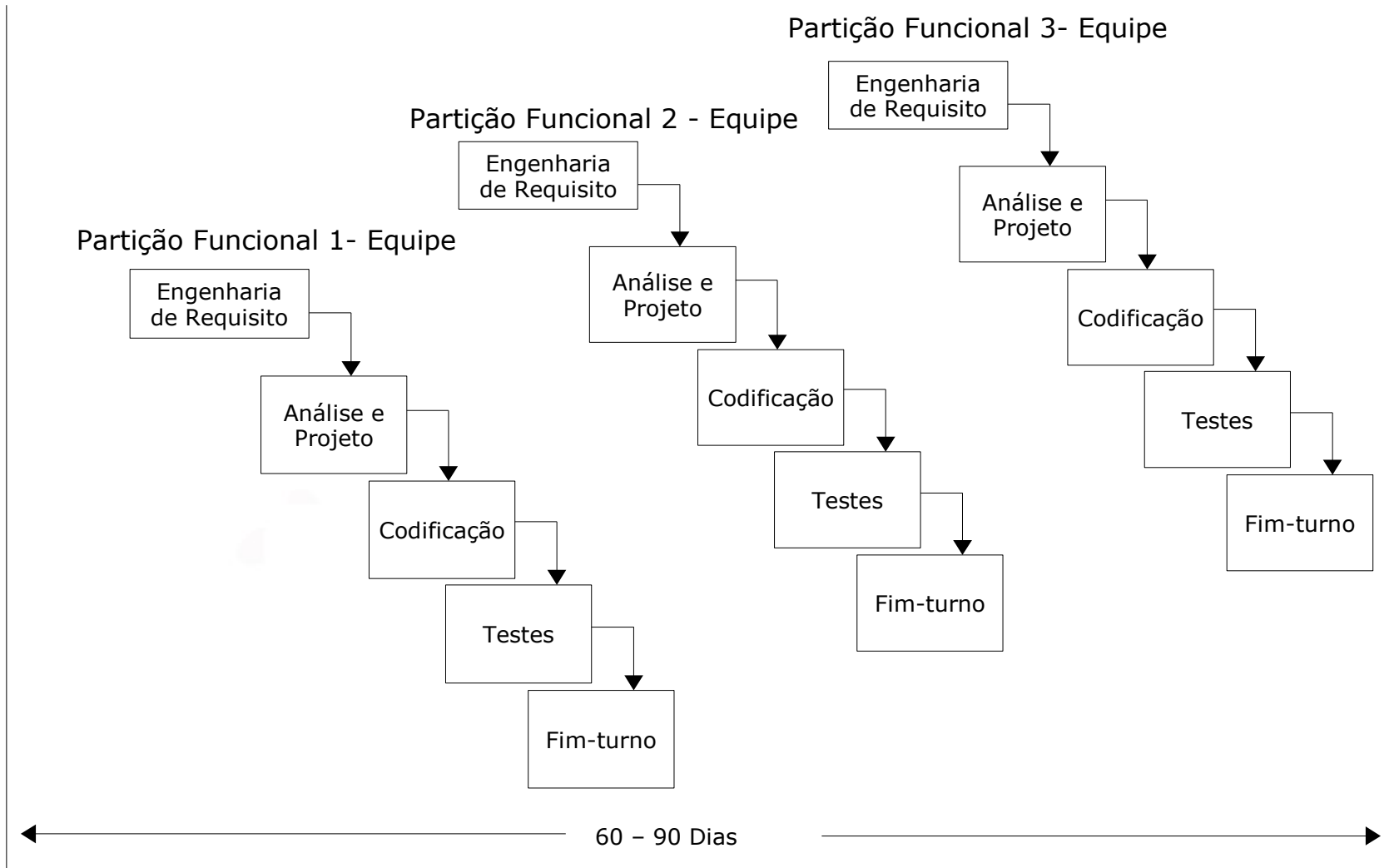
- Clientes vêem o que parece ser uma versão funcional do software sem consciência que o protótipo é mantido junto “com os artefatos descartáveis” sem considerar a qualidade do software e a manutenibilidade a longo prazo
- Desenvolvedores programam afim de ter um protótipo funcional



Modelo de Desenvolvimento Rápido de Aplicações (RAD)

- Desenvolvimento de software linear seqüencial que enfatiza o ciclo de vida de desenvolvimento extremamente curto
- Dirigido à abordagem de construção baseada em componentes
- Melhor se todos os requisitos são bem entendidos e o escopo do projeto já está bem definido

Modelo RAD



Modelo RAD

- **Vantagens:**

- Sistema totalmente funcional é criado em um curto espaço de tempo

- **Desvantagens:**

- Para grandes, mas escaláveis projetos, o RAD requer recursos humanos suficiente para criar o número certo de equipes
- RAD requer desenvolvedores e clientes comprometidos com as atividades relâmpagos necessárias para completar o sistema em curtíssimo espaço de tempo
- Não é um bom modelo de processo de software para sistemas que não são modularizáveis
- Também não é um bom modelo de processo de software para sistemas que requerem alta performance
- Não apropriado para nova tecnologia ou quando o software requer alto grau de interoperabilidade



Modelo de Processo de Software Evolucionário

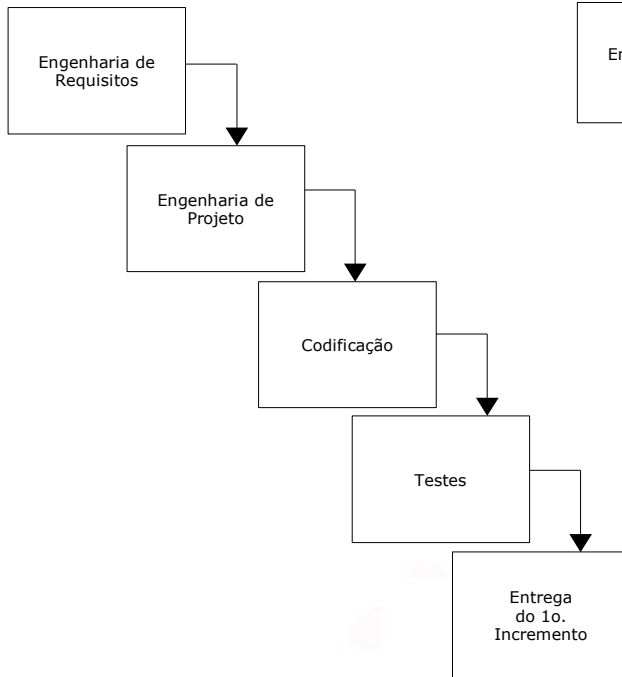
- Reconhece que softwares envolvem mais que um período de tempo
- Iterativos por natureza
- Permitem o desenvolvimento de versões do software crescentemente mais complexas
- Exemplos:
 1. Modelo Incremental
 2. Modelo Espiral
 3. Modelo de Montagem Baseada em Componentes

Modelo Incremental

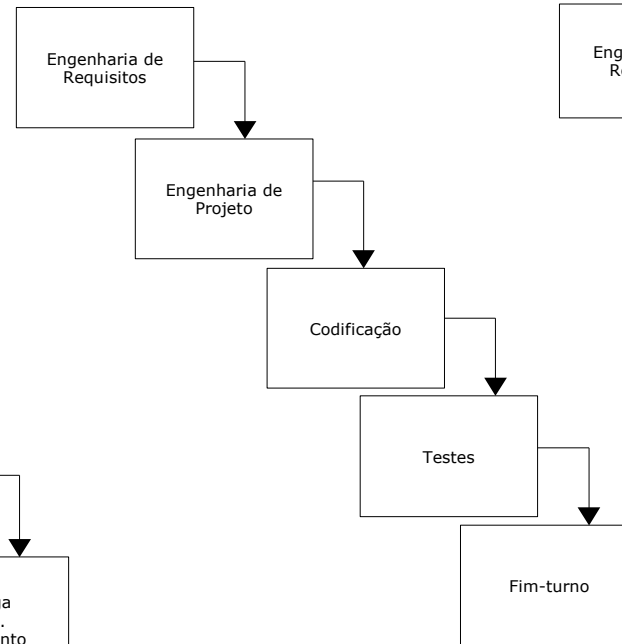
- Combina elementos do modelo linear seqüencial com a filosofia iterativa da prototipação
- Seqüências lineares são definidas, isto é, cada seqüência produzida é um “incremento” do software
- Diferente da prototipação, cada incremento é um produto operacional que poderá ser entregue

Modelo Incremental

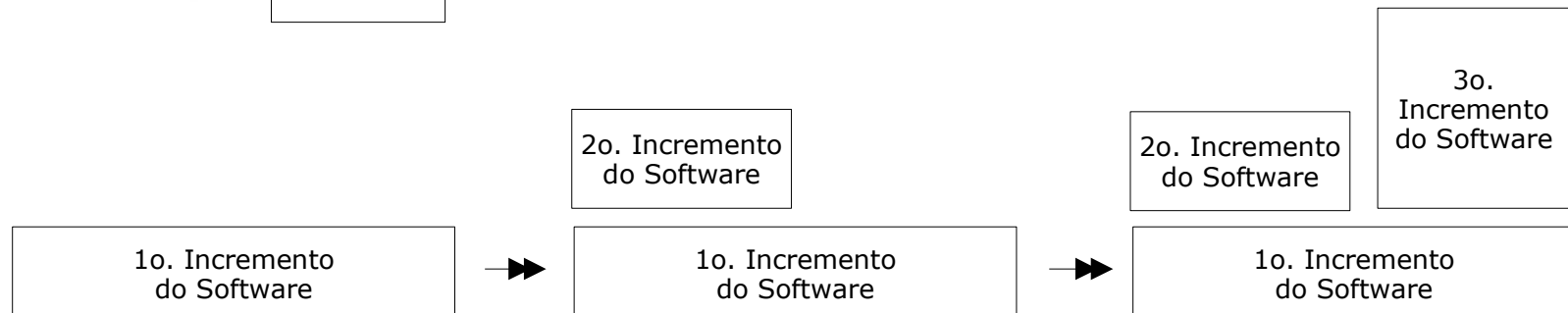
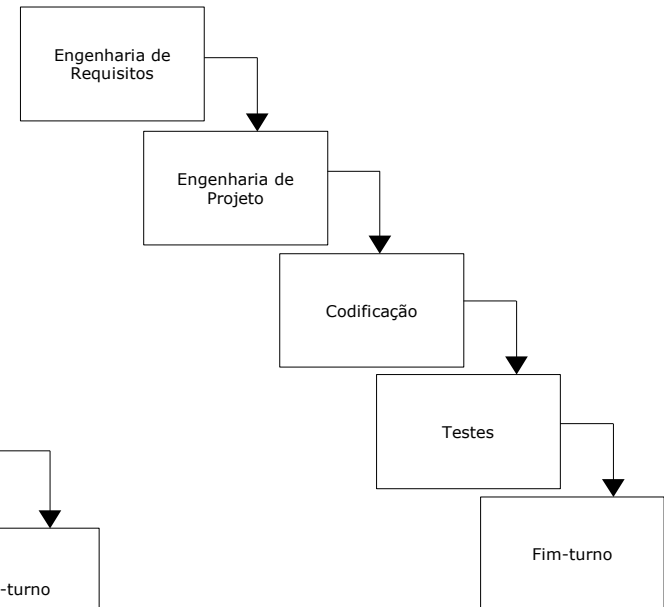
Primeiro Incremento



Segundo Incremento



Terceiro Incremento



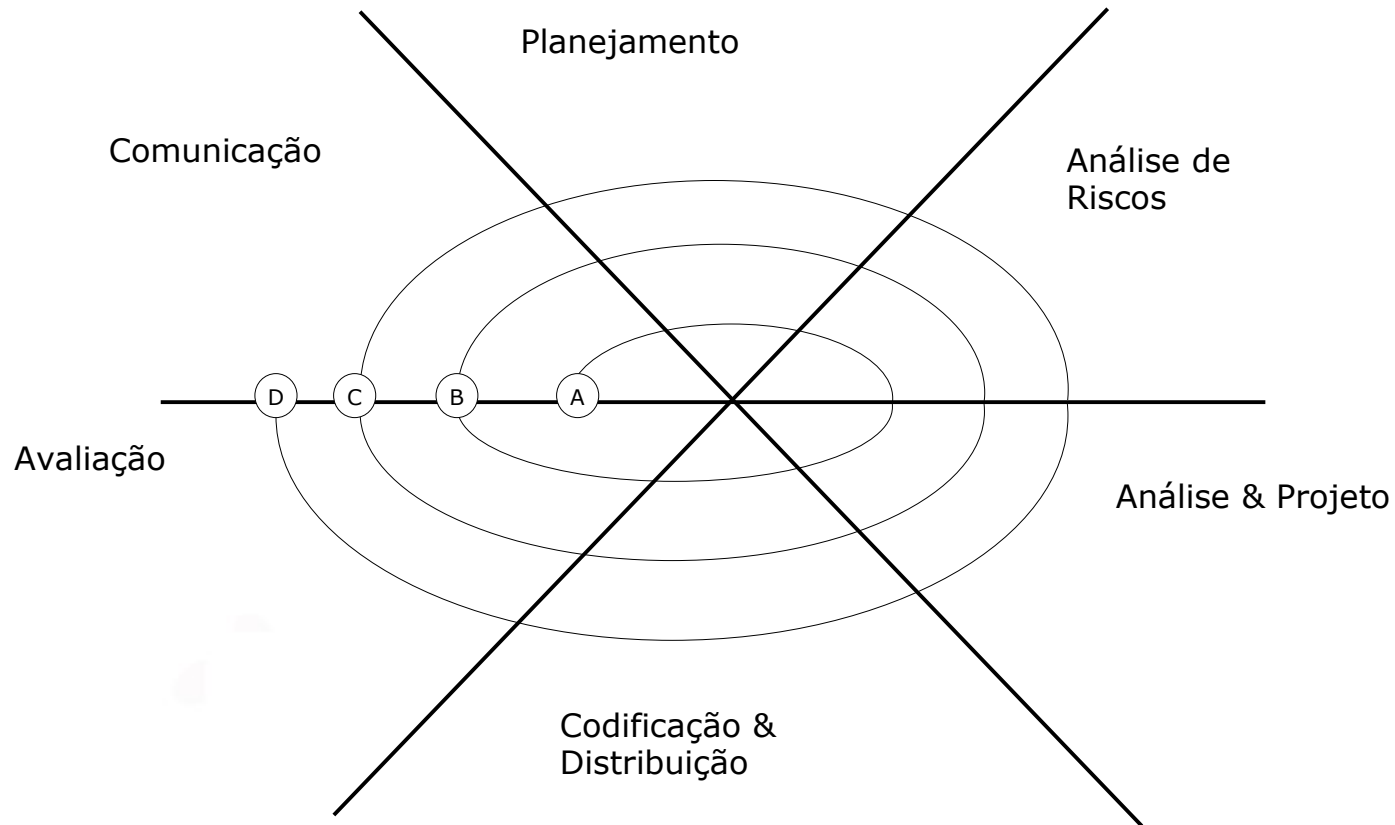
Modelo Espiral

- Originalmente proposto por *Boehm*
- Modelo de processo de software evolucionário que se acopla a natureza iterativa da prototipação com os aspectos de controle e sistemática do modelo linear seqüencial
- Provê potencial para rápido desenvolvimento de versões incrementais do software

Modelo Espiral

- Dividido em modelo estruturais também conhecido como regiões de tarefa:
 1. Comunicação com o Cliente
 2. Planejamento
 3. Análise de risco
 4. Engenharia
 5. Construção e Distribuição
 6. Avaliação do Cliente

Modelo Espiral



- A. Projeto Inicial do Software
- B. Manutenção do Novo Software
- C. Melhoramento do Software
- D. Desenvolvimento de outro sistema inter-relacionado

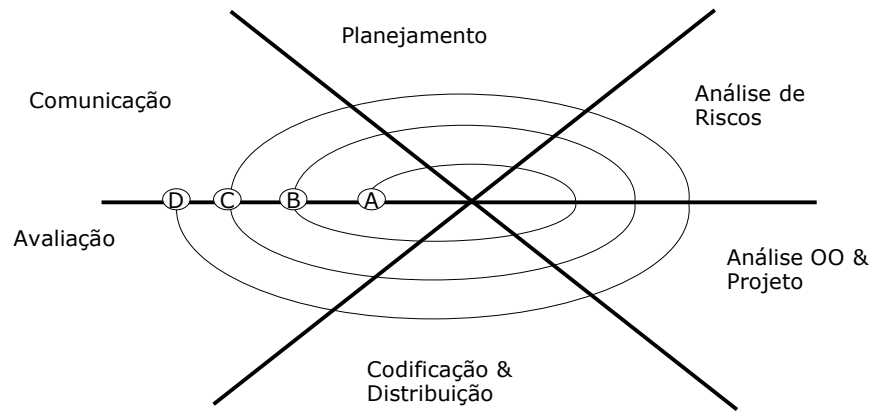


Modelo Baseado em Montagem de Componentes

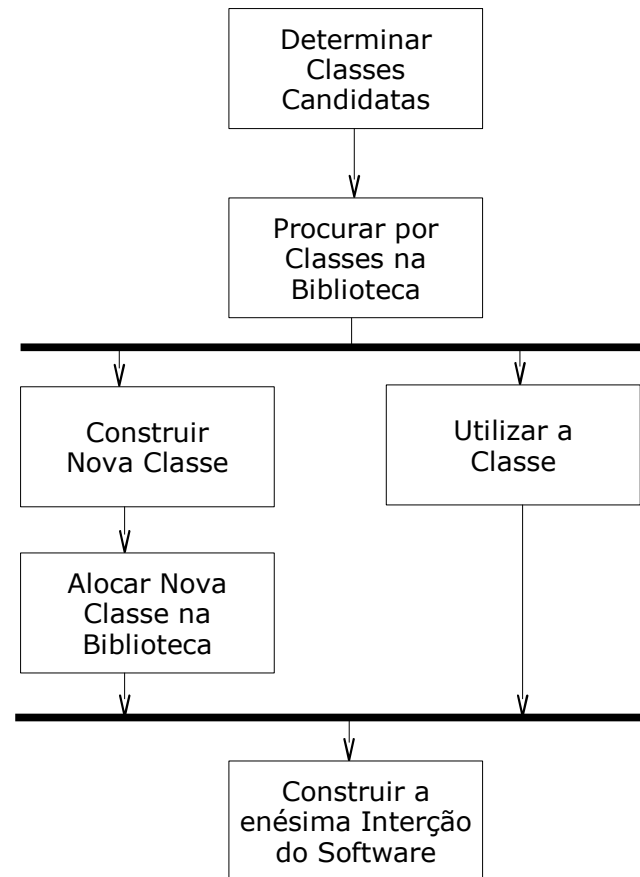
- Faz uso da tecnologia de objetos
- Enfatiza a criação de classes que encapsulam dados e algoritmos que manipulam estes dados
- Classes OO são reutilizadas entre diferentes aplicações e arquiteturas de sistemas baseadas em computadores
- Incorpora muitas das características do Modelo Espiral



Modelo Baseado em Montagem de Componentes



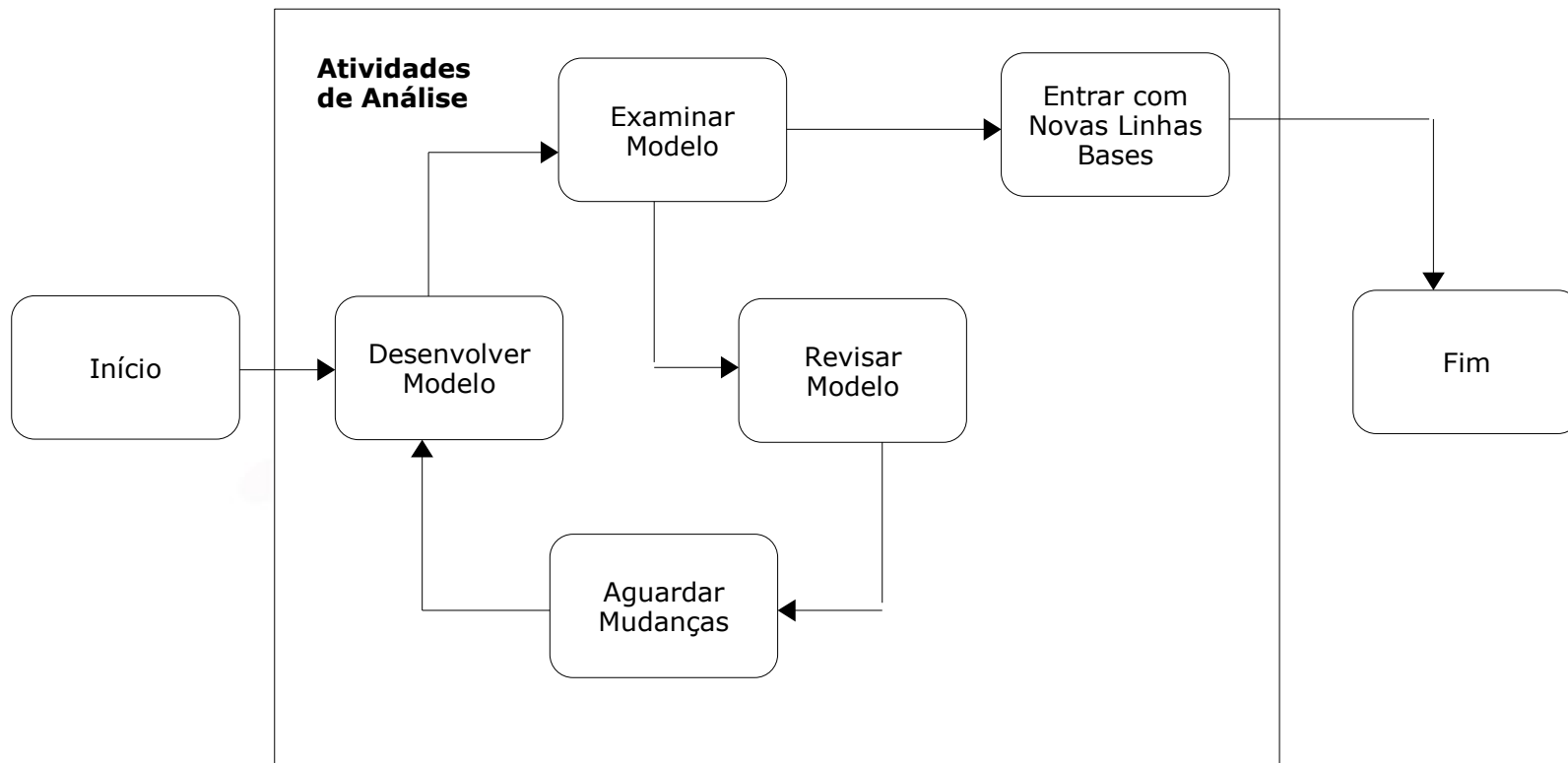
- A. Projeto Inicial do Software
- B. Manutenção do Novo Software
- C. Melhoramento do Software
- D. Desenvolvimento de outro sistema inter-relacionado



Modelo de Desenvolvimento Concorrente

- Conhecido como *engenharia concorrente*
- Faz uso de diagramas de estado que representam a relação concorrente entre atividades associadas com um evento específico
- Pode ser representado esquematicamente como uma série de importantes atividades técnicas, tarefas e seus estados associados
- Dirigido pela necessidade do usuário, decisões gerenciais e revisão de resultados

Modelo de Desenvolvimento Concorrente



Métodos Formais

- Incorpora um conjunto de atividades que são guiados por uma especificação matemática do software
- Fornece um mecanismo para eliminar muitos dos problemas que são difíceis de encontrar usando outro paradigma de engenharia de software
- Utilizado como um programa de verificação e evidencia a descoberta e correção de erros que de outra forma não seriam detectados

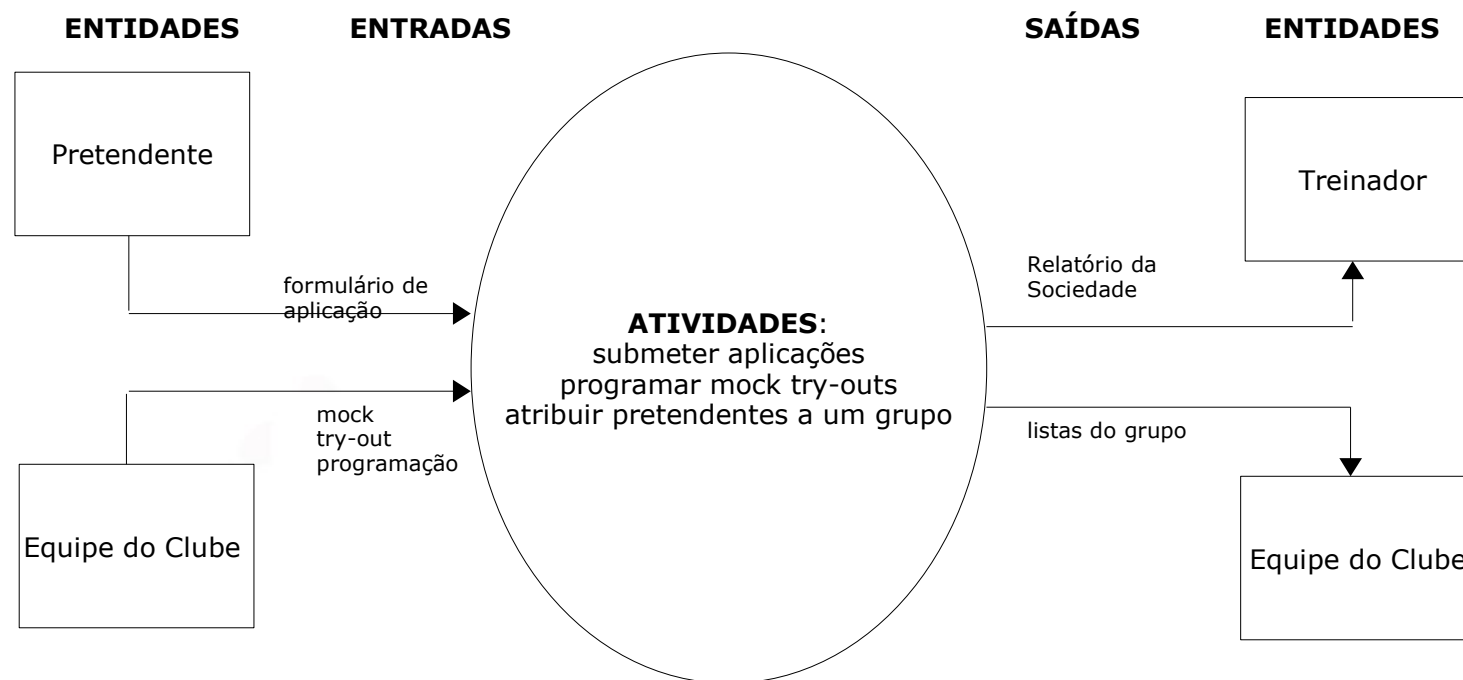
Fatores que Afetam a Escolha do Modelo de Processo

- Tipo do Projeto
- Métodos e Ferramentas a Utilizar
- Requisitos e Pessoas-Chave do Negócio
- Senso comum e Julgamento

Sistemas

- Grupo de entidades ou componentes
- Compostos de subsistemas menores

Sociedade do Clube Limites de Manutenção do Sistema



Dois tipos de Sistemas

- Sistemas Sintéticos
 - Imperfeitos
 - Possuem sempre áreas de correção e aperfeiçoamento
- Sistemas Automatizados

Princípios Gerais de Sistemas

- Quanto mais especializado é um sistema, menos capaz ele é de adaptar-se às circunstâncias diferentes
- Quanto maior o sistema é, mais recursos devem ser devotados a sua manutenção regular
- Sistemas são sempre parte de Sistemas maiores, e podem sempre ser particionados em sistemas menores

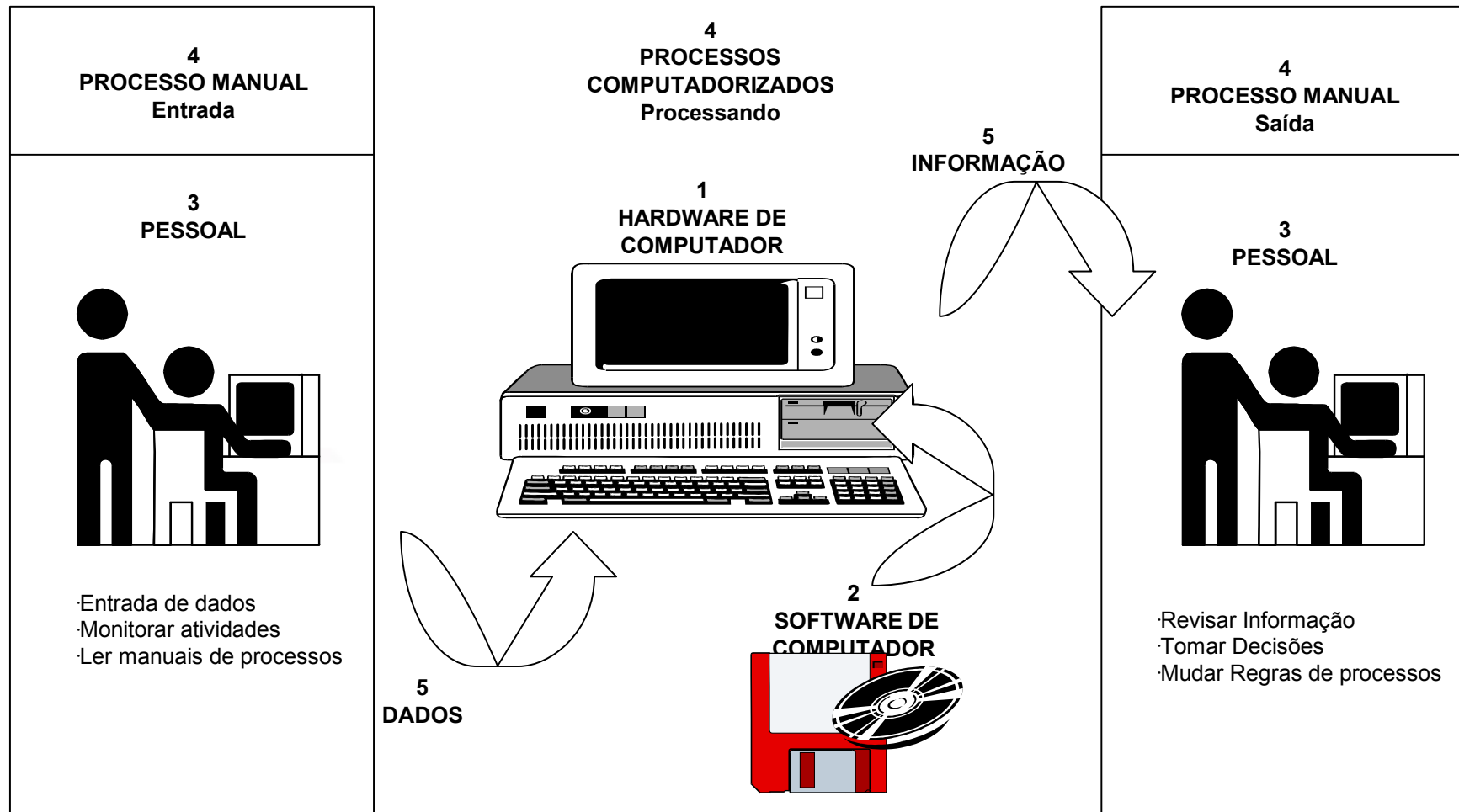


Componentes de um Sistema Automatizado

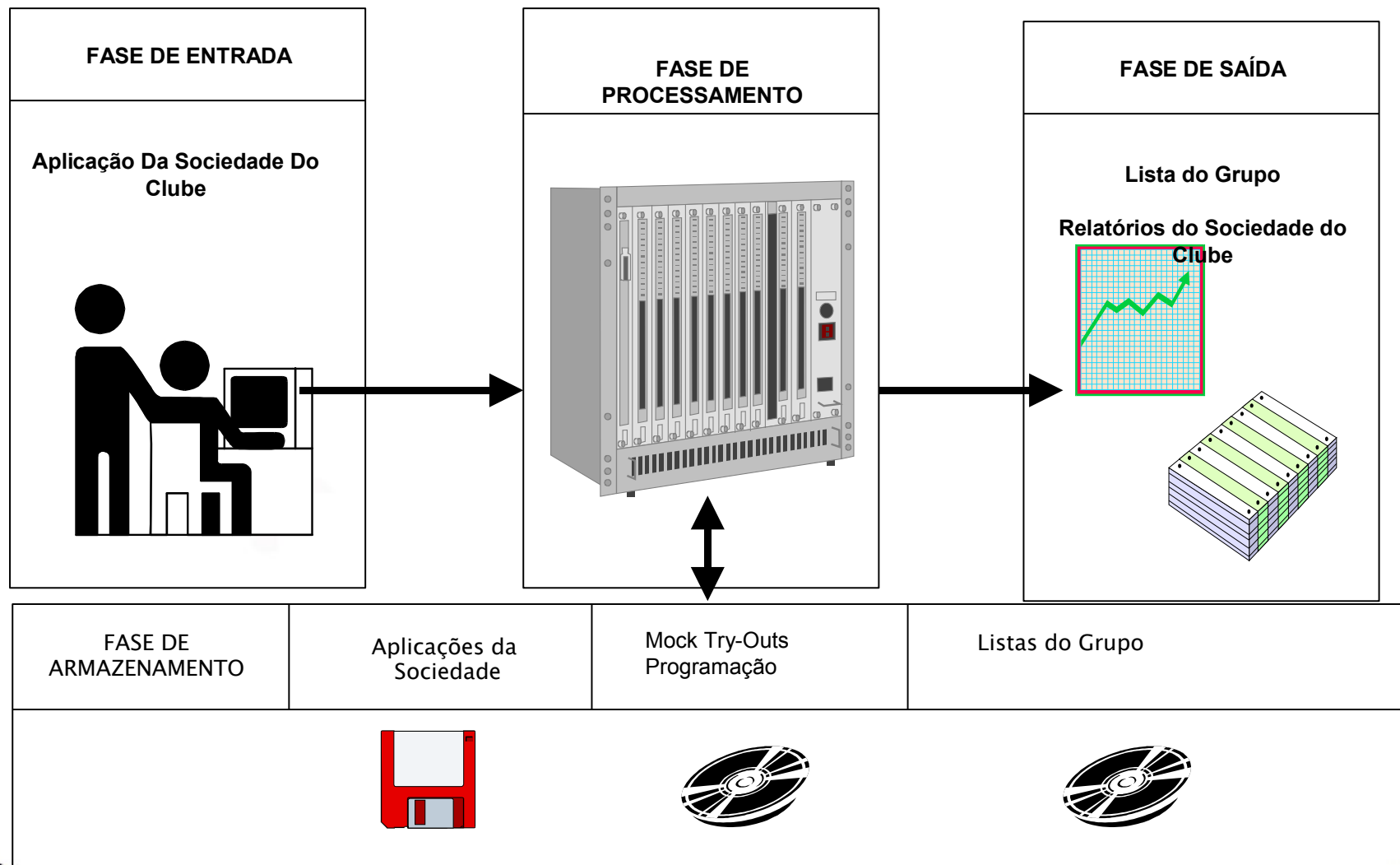
- Hardware de Computador
- Software de Computador
- Pessoas
- Dados/Informação
- Processos
- Conectividade



Relacionamento entre Componentes



Processamento de Aplicação da Sociedade do Clube



Principais Grupos

- Usuários Finais
- Equipe de Desenvolvimento

Trabalho

- Trabalhadores Operacionais
- Supervisor
- Executivos
- Gerência:
 - Usuários Gerentes
 - Gerentes EDP/MIS
 - Gerentes Gerais

Guias Gerais e os Usuários Finais

- O mais alto nível de gerência pouco se importa com a tecnologia de computadores
- Os objetivos e prioridades da gerência podem entrar em conflito com os dos usuários, especialmente o supervisor e usuários operacionais
- A gerência pode não fornecer recursos, fundos ou tempo para construir um sistema efetivo
- Gerentes têm diferentes visões e opiniões e muito freqüentemente diferentes metas e objetivos



Auditores, Portadores Comuns, Garantia de Qualidade

- Objetivo geral deste grupo é certificar que o sistema está sendo desenvolvido de acordo com vários conjuntos de padrão, como:
 - padrões contábeis desenvolvido pela firma de contabilidade da organização
 - padrões desenvolvidos por outros departamentos dentro da organização ou por clientes/usuários que herdaram o sistema
 - vários padrões impostos pelas agências regulatórias do governamentais
- Alguns possíveis problemas:
 - não se envolvem com o projeto até o fim do projeto
 - variações na notação e formato da documentação
 - podem ter mais interesse em substância

Equipe de Desenvolvimento

- Analista de Sistemas
- Projetista
- Programadores
- Testadores



Documentação

- Equipe:
 - *Webmaster*
 - Profissionais de TI
- Propósitos:
 - Oferecer um argumento racional e permanente da estrutura ou comportamento de um sistema
 - Servir como documentos transitórios que são parte da infraestrutura envolvida em execução de projetos reais

Critério para Medição de Usabilidade de Documentos

- Disponibilidade
- Adequação
- Acessibilidade
- Legibilidade

Importância dos Manuais e Documentos

- Economia
- Ferramentas de Venda e Marketing
- Entregas Tangíveis
- Obrigações Contratuais
- Cobertura de Segurança
- Auxílio à Implementação e Testes
- Utilizados para comparar sistemas antigos e novos



Sumário

- Engenharia de Software
- Uma camada de Visão
- Qualidade
- Três perspectivas sobre qualidade
- Endereçando pontos importante sobre qualidade
- Garantia de qualidade de software e definir qualidade de software
- Atividades da qualidade do software
- Técnicas formais de revisão
 - Método de inspeção Fagan
 - Walkthrough (Revisão em pares)
- Processo de Software



Sumário

- Tipos de Processo de Software
 - Modelo Linear Seqüencial
 - Modelo de Prototipação
 - Modelo de Desenvolvimento Rápido de Aplicações (RAD)
 - Modelo de Processo Evolucionário
- Tipos de Modelo de Processo de Software
 - Métodos Formais
- Fatores que afetam a escolha do processo de desenvolvimento de software
- Sistema
- Princípios Gerais do Sistema
- Dois Tipos de Sistema



Sumário

- Fatores a Considerar na Automação
- Componentes de um Sistema Automatizado
- Relacionamento entre os componentes de um sistema automatizado
- Dois Principais Grupos
- Usuários Finais
 - Diretamente Envolvido no Negócio
 - Indiretamente Envolvido no Negócio
- Equipe de Desenvolvimento
- Documentação



Parceiros

- Os seguintes parceiros tornaram JEDITM possível em Língua Portuguesa:

