



PRINCÍPIOS DA ENGENHARIA DE SOFTWARE – AULA 01

Prof.: Franklin M. Correia

E-mail:

franklin.correia@bonfim.ifbaiano.edu.br

APRESENTAÇÃO

- Franklin Matos Correia
- Bacharel em Ciência da Computação
 - Faculdade Ruy Barbosa – 2009
- Especialista em Engenharia de Software
 - Faculdade Ruy Barbosa 2013



OBJETIVO

Conhecer e aplicar as metodologias, ferramentas, procedimentos e boas práticas de Engenharia de Software.



AVALIAÇÕES

- 2 Provas + Trabalho
- Prova 1 (P1)
- Prova 2 (P2)
- Trabalho (Tb)
- Nota Final = $(P1 + P2) * 7 + Tb * 3$



DATAS AVALIAÇÕES

- Prova 1 - 12/09/2013*
 - Revisão caso tenham dúvidas na primeira aula
- Prova 2 - 10/10/2013
 - Revisão caso tenham dúvidas na primeira aula
 - Inclui também os assuntos da Prova 1
- Trabalho – Seminário / Estudo de Caso – 17 e 24/10/2013
 - 30 minutos.
- Prova final
 - Assuntos do semestre inteiro.



OBSERVAÇÕES & ALERTAS

- Itens negociáveis
 - Datas das provas
 - Nunca no dia da prova
 - Tipo de trabalho
 - Nunca no dia da apresentação / entrega
- Itens inegociáveis
 - Quantidade de provas
 - Trabalho
 - Data do trabalho
- 4 aulas/dia
 - 2 aulas +intervalo de 10 minutos + 2 aulas.
 - 1 aula = 50 minutos



O QUE TEMOS PRA HOJE??

- Introdução a Engenharia de Software
 - O que é software?
 - O que é Engenharia de Software?
 - Conceitos importantes
 - Tipos de Produtos
 - Processo de software
 - Fases do processo de software
 - Modelos de Processo de software



CRISE DO SOFTWARE

- Termo Engenharia de software usando 1968 durante a crise do software.
- Produção de um novo hardware usando circuitos integrados
- Aplicações inviáveis tornaram-se realizáveis
- Construção informal de software
- Atrasos exorbitantes
- Alto custo de construção de software
- Custo do hardware caindo e de software subindo
- Criação de técnicas para controle da produção do software

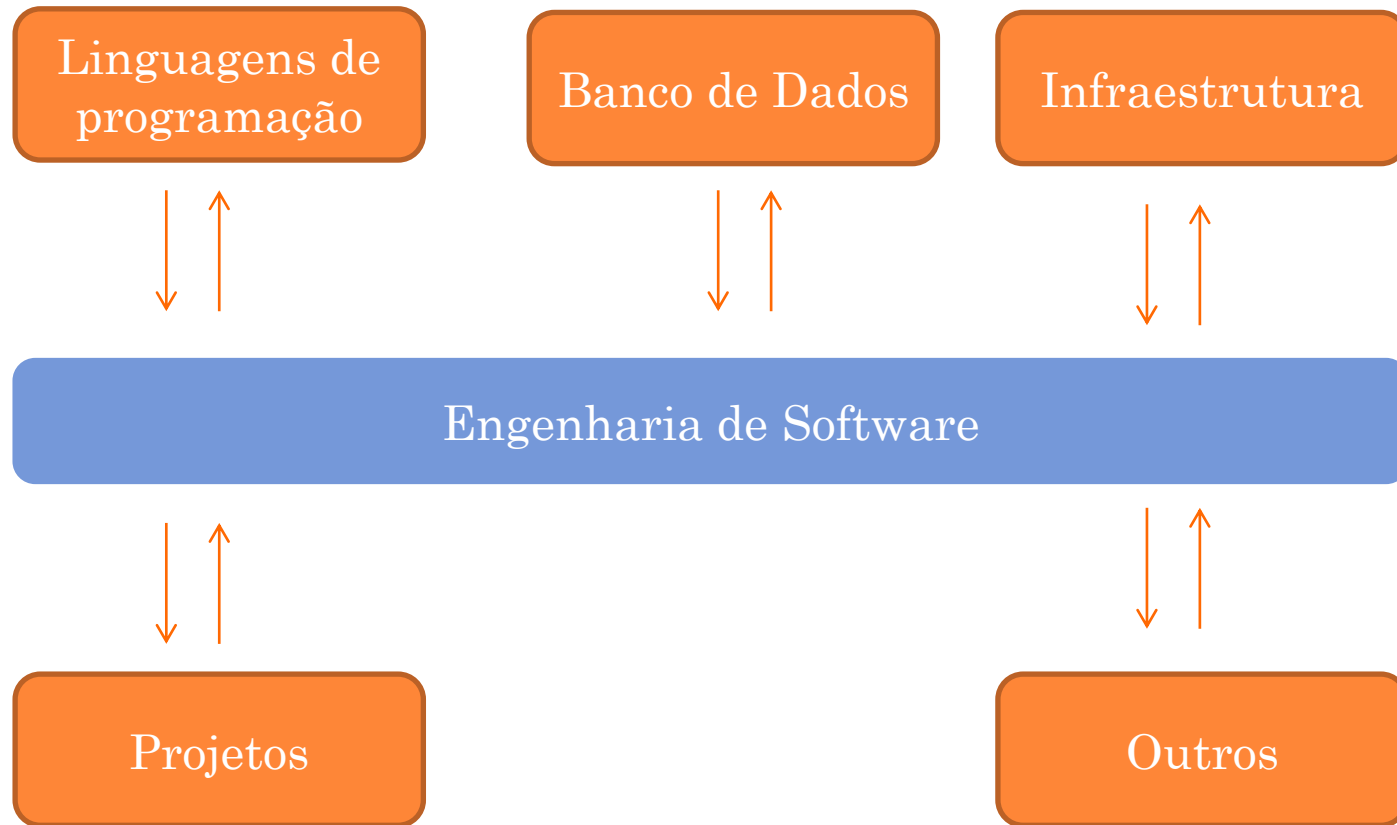


ENGENHARIA DE SOFTWARE

- O que é Engenharia de Software?
 - Ramo da engenharia cujo foco é o desenvolvimento dentro de custos adequados de sistemas de software de qualidade. Software é abstrato, intangível, não é limitado por materiais, ou controlado por leis físicas ou por processos de manufatura (Sommerville, 2003).



O QUE É ENGENHARIA DE SOFTWARE



O QUE É SOFTWARE?

- Software não é apenas o arquivo executável / programa
- É o Programa de computador, toda documentação associada(arquivos de configuração, manual de instalação e utilização) e o banco de dados.
- Podem ser desenvolvidos para um cliente específico ou para um mercado geral



TIPO DE PRODUTO SOFTWARE

- Produto de Software de Prateleira / Genéricos
 - Chamado de stand-alone
 - Criados de forma genérica, para qualquer empresa.
 - Controle de estoque
 - Controle de farmácia
 - Programas de Contabilidade
- Produtos sob encomenda / Personalizados
 - Software criados com objetivo de prover uma solução específica para um cliente específico
 - Software para dispositivos eletrônicos: Geladeiras, jogões, micro-ondas
 - Sistema de controle de tráfego aéreo



PROCESSO DE SOFTWARE

- Método utilizado para desenvolver ou produzir um software.
- Define **o que** faz, **como** será feito e **quando** será feito



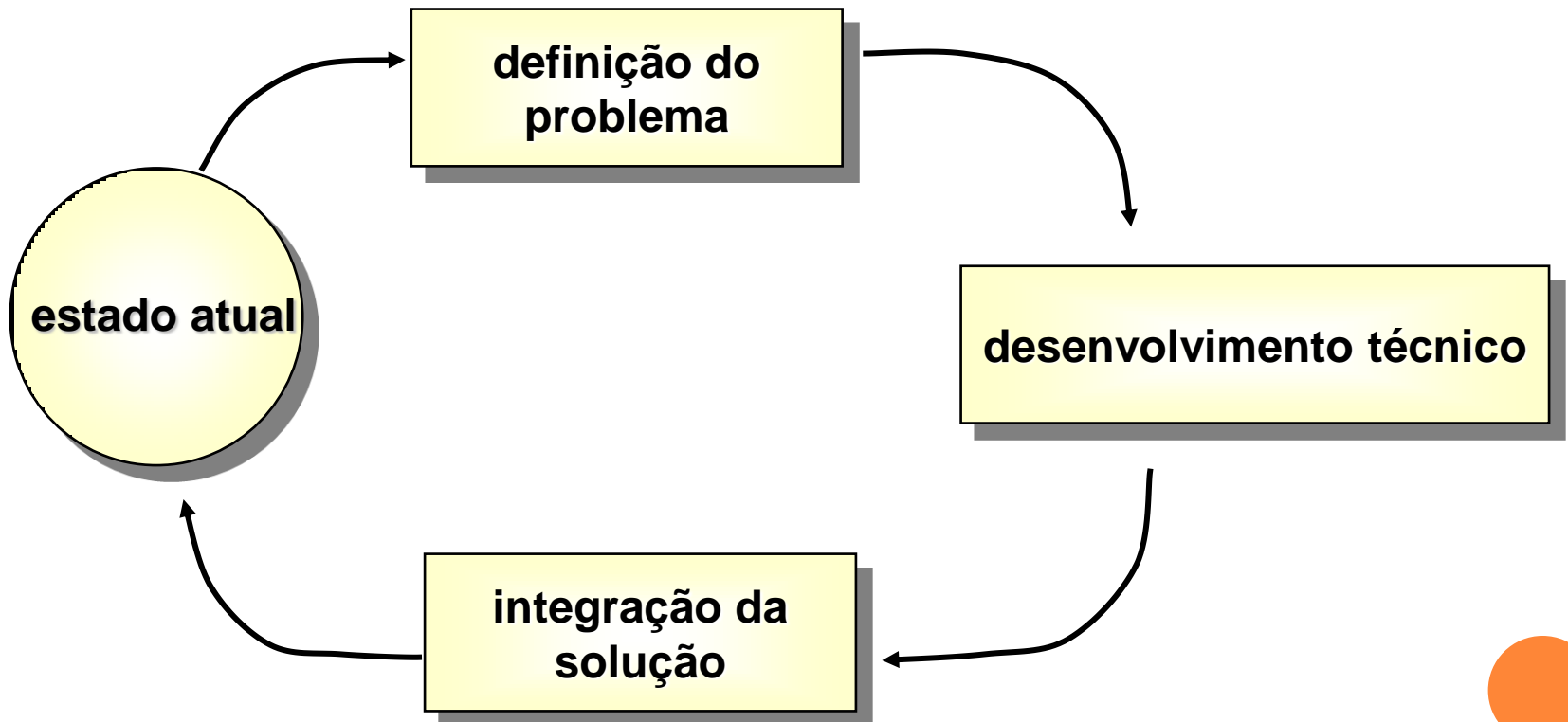
PROCESSO DE SOFTWARE

- Capaz de responder as perguntas:
- O que é feito? ====> Produto
- Como é feito? ====> Passos
- Por quem é feito? ====> Agente
- O que usa? ====> Insumos
- O que produz? ====> Resultados



MODELO DE PROCESSO DE SOFTWARE

- Deve incorporar uma estratégia de desenvolvimento

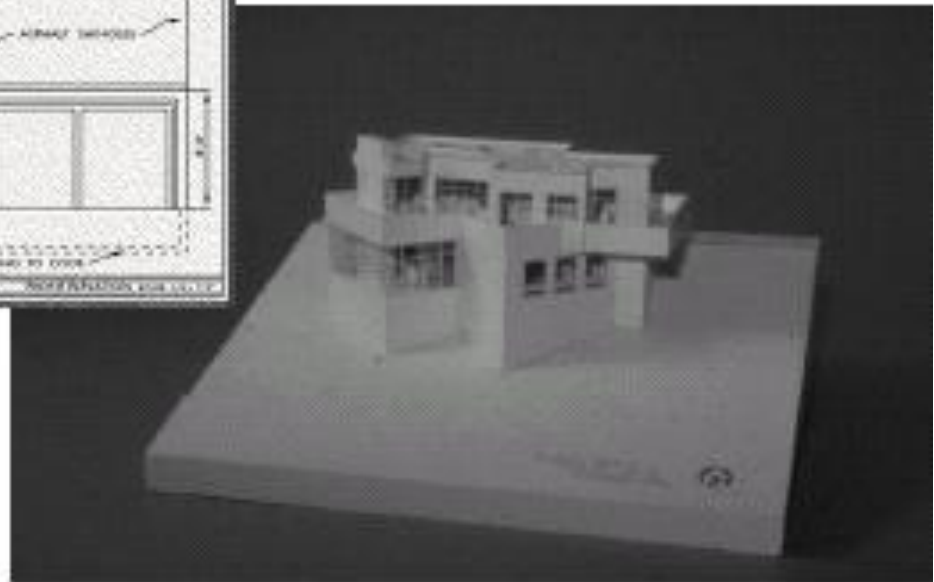


MODELO DE PROCESSO DE SOFTWARE

- A modelagem é uma técnica de engenharia bem aceita
 - modelos de arquitetura de casas e de grandes prédios
 - modelos matemáticos a fim de analisar os efeitos de ventos e tremores de terra --> causas



MODELO DE PROCESSO DE SOFTWARE



MODELO DE PROCESSO DE SOFTWARE

- O que é?
 - Simplificação da realidade
 - Planos podem ser:
 - Reais - Organização do sistema
 - Comportamentais – dinâmica do sistema
- Porque é importante construir modelos?
 - Melhor entendimento do sistema que está sendo construído
 - Especificar a estrutura e comportamento
 - Guia a construção do sistema
 - Documenta as decisões tomadas



MODELO DE PROCESSO DE SOFTWARE

○ Objetivo

- **Auxiliar ao gerente:** controlar o processo de desenvolvimento de sistemas de software.
- **Auxiliar ao desenvolvedor:** obter a base para produzir, de maneira eficiente, software que satisfaça os requisitos pré-estabelecidos.

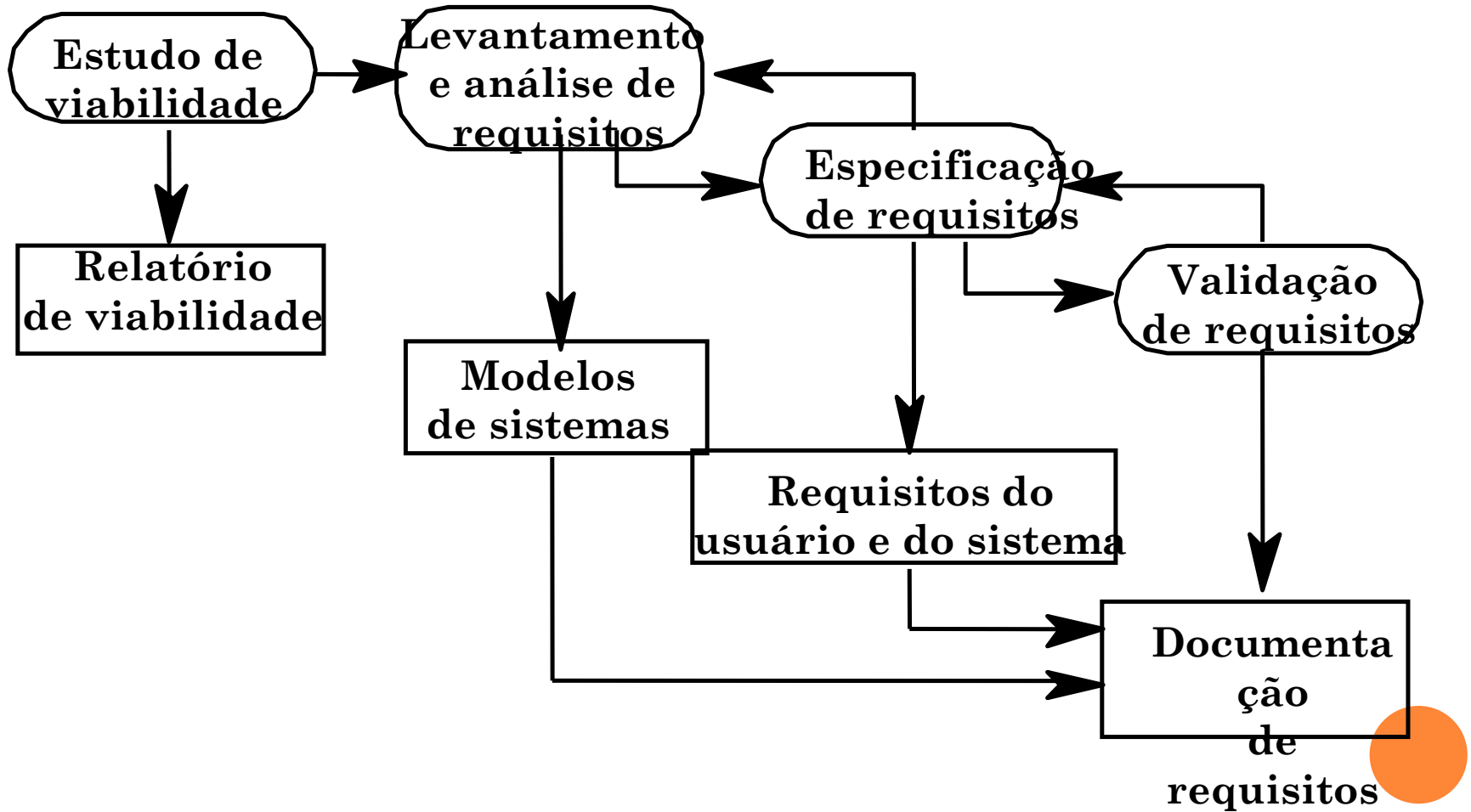


MODELO X PROCESSO

- Modelo de software : documento teórico, conjunto de possíveis ações
- Processo de software: deve determinar ações práticas a serem realizadas pela equipe como prazos definidos e métricas para se avaliar como elas estão sendo realizadas



PROCESSO DE SOFTWARE



PROCESSO DE SOFTWARE

- Estudo de viabilidade
 - Econômica – relação custo/benefício;
 - Técnica – tecnologia e capacitação;
 - Jurídica – aspectos legais
- Levantamento de Análise de Requisitos
 - Entrevista
 - Observação
 - Reuniões



PROCESSO DE SOFTWARE

- Especificação de requisitos
 - Documento contendo os requisitos do usuário e do sistema
 - Funcionais e não funcionais
- Validação de requisitos
 - Avaliação do documento de requisitos – consistência e integridade



MODELO DE PROCESSO DE SOFTWARE

- Exemplo de modelos de processo:
 - Workflow – sucessão de atividades
 - Fluxo de dados – fluxo de informação
 - Papel / Ação – representa os papéis das pessoas e as atividades pelas quais elas são responsáveis



CICLO DE VIDA DE UM SOFTWARE

- Uma estratégia de desenvolvimento que englobe processos, métodos e ferramentas, e as fases de desenvolvimento...



CICLO DE VIDA DE UM SOFTWARE

- Modelo em Cascata - ciclo clássico
- Paradigma Evolucionário
 - Prototipação
 - Incremental
 - Espiral
 - Métodos Ágeis
- Modelos Formais
- Técnicas de 4ª Geração
- Orientado a Reuso

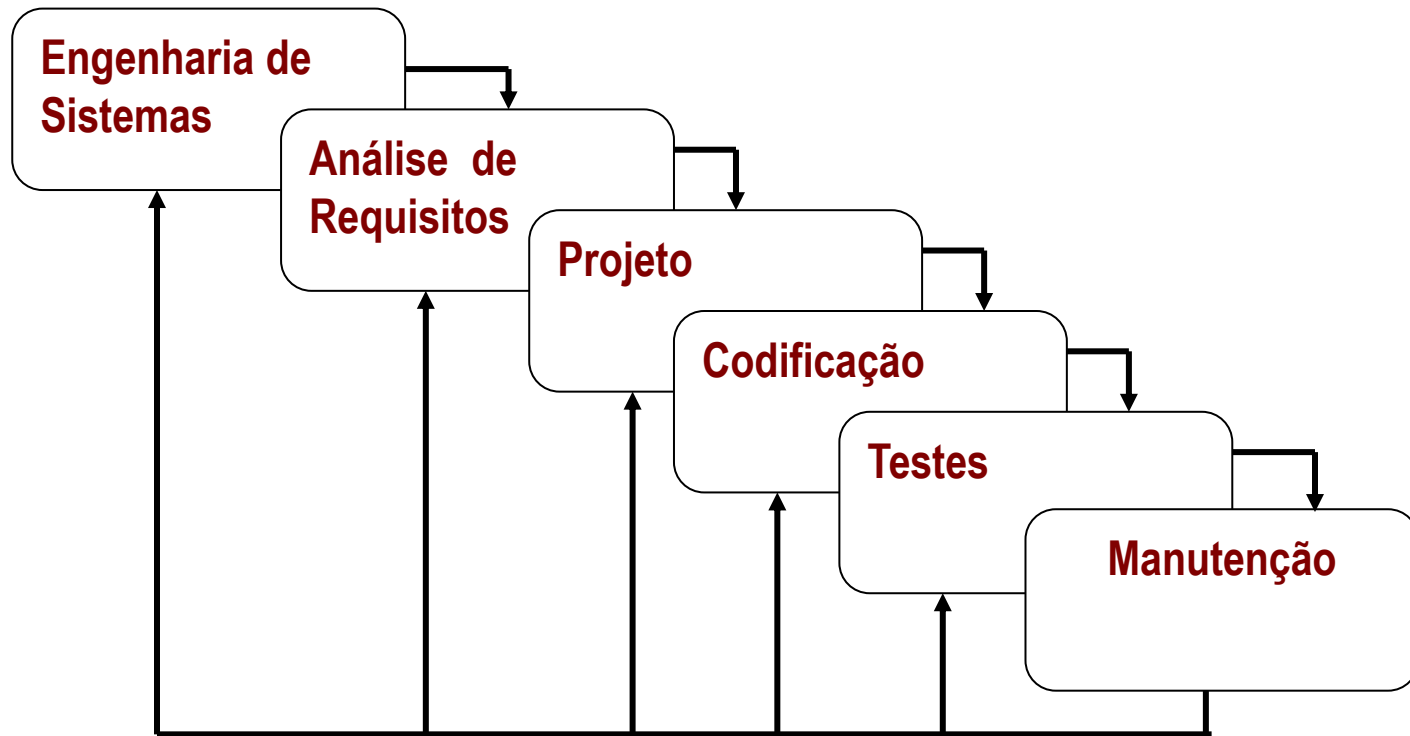


MODELO EM CASCATA (CICLO CLÁSSICO)

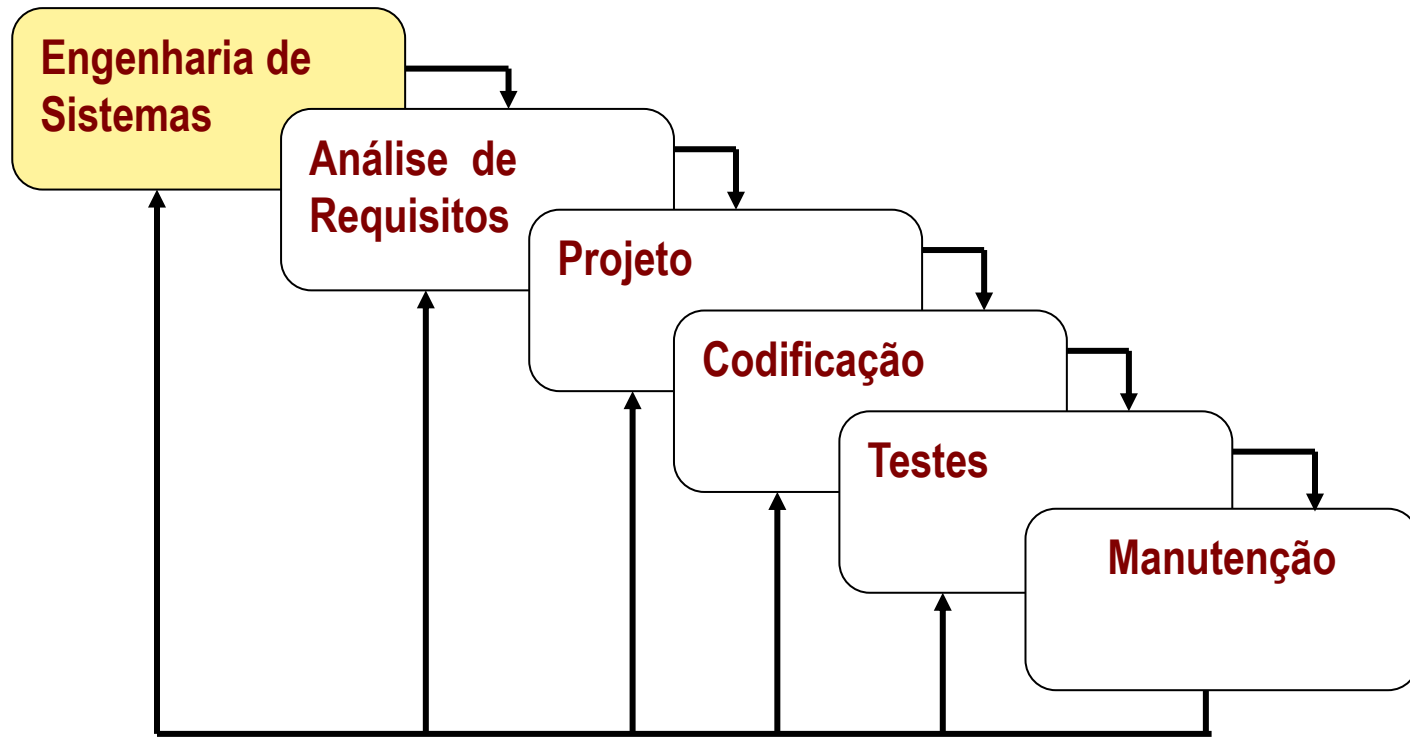
- Método sistemático e sequencial
- O resultado de uma fase se constitui na entrada da outra
- Cada fase é estruturada como um conjunto de atividades que podem ser executadas por pessoas diferentes



MODELO EM CASCATA (CICLO CLÁSSICO)



MODELO EM CASCATA (CICLO CLÁSSICO)

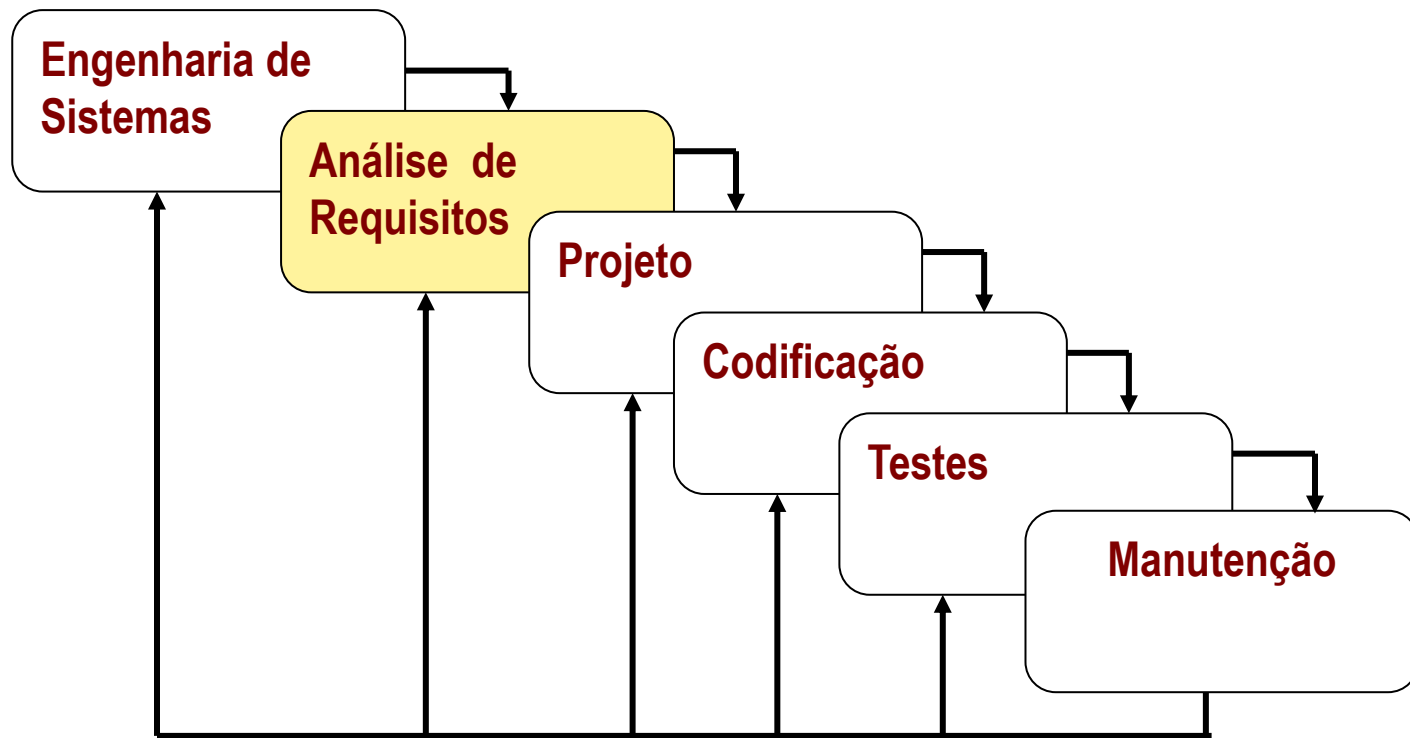


MODELO EM CASCATA (CICLO CLÁSSICO)

- Engenharia de Sistemas
 - Envolve a coleta de requisitos (nível de sistemas)
 - Pequena quantidade de projetos
 - Análise de alto nível
 - Importante quando o sistema fizer interface com outros elementos (hardware, pessoas e banco de dados)



MODELO EM CASCATA (CICLO CLÁSSICO)

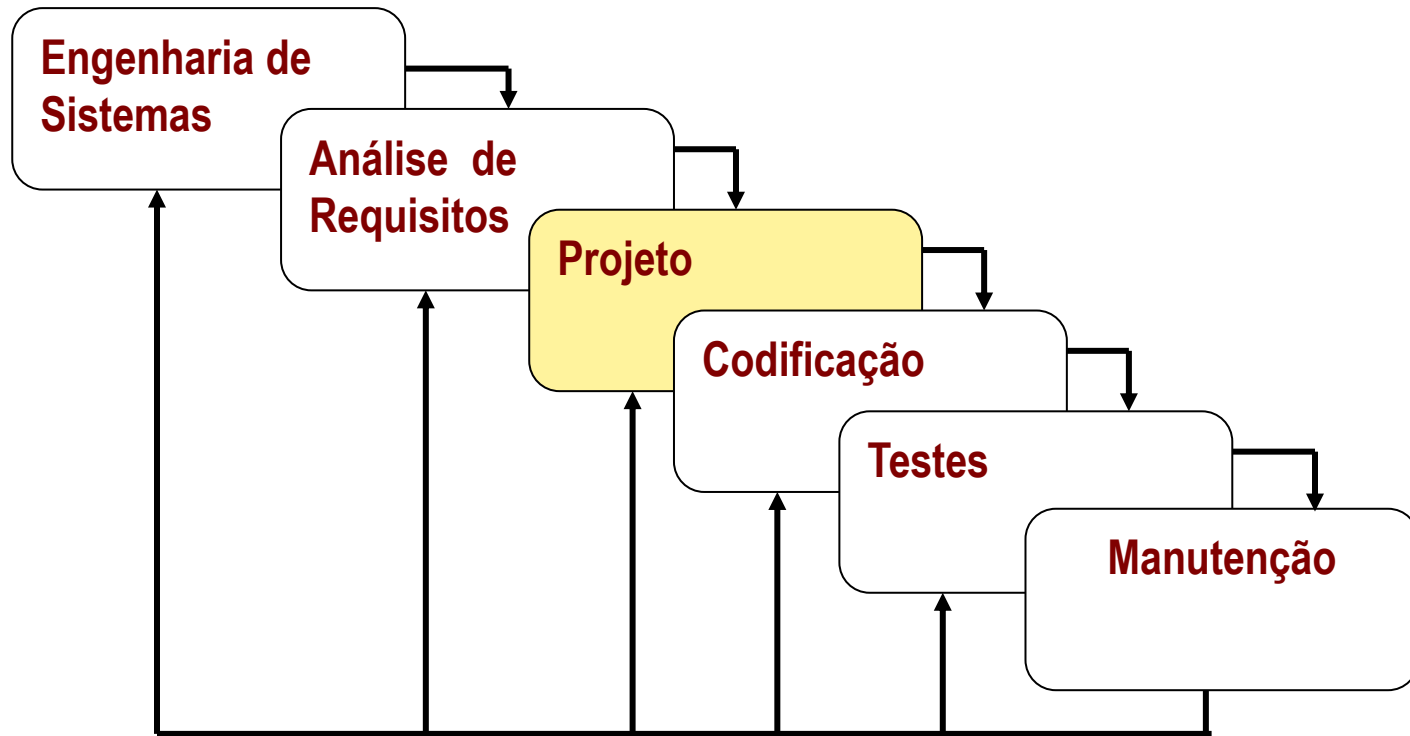


MODELO EM CASCATA (CICLO CLÁSSICO)

- **Análise de Requisitos**
 - Envolve a coleta de requisitos (nível de usuário) de forma intensa
 - Compreensão do domínio, função, desempenho e interface necessários
 - Os requisitos são documentados e revistos com o cliente



MODELO EM CASCATA (CICLO CLÁSSICO)

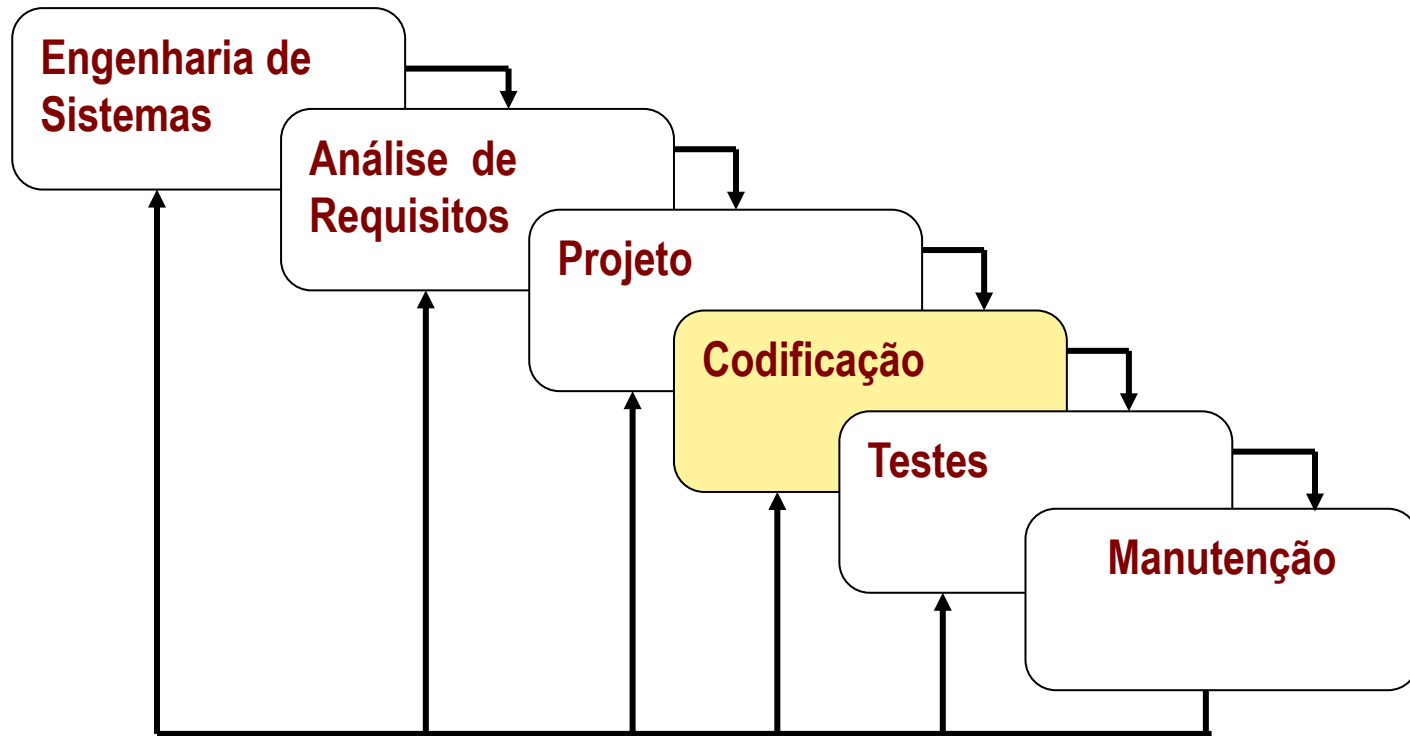


MODELO EM CASCATA (CICLO CLÁSSICO)

- Projeto
- Requisitos do software -> Representações
 - Avaliação de qualidade
 - Anterior a codificação
- Concentram em 4 atributos
 - Estrutura de dados
 - Arquitetura
 - Detalhes de procedimentos
 - Caracterização de interface



MODELO EM CASCATA (CICLO CLÁSSICO)



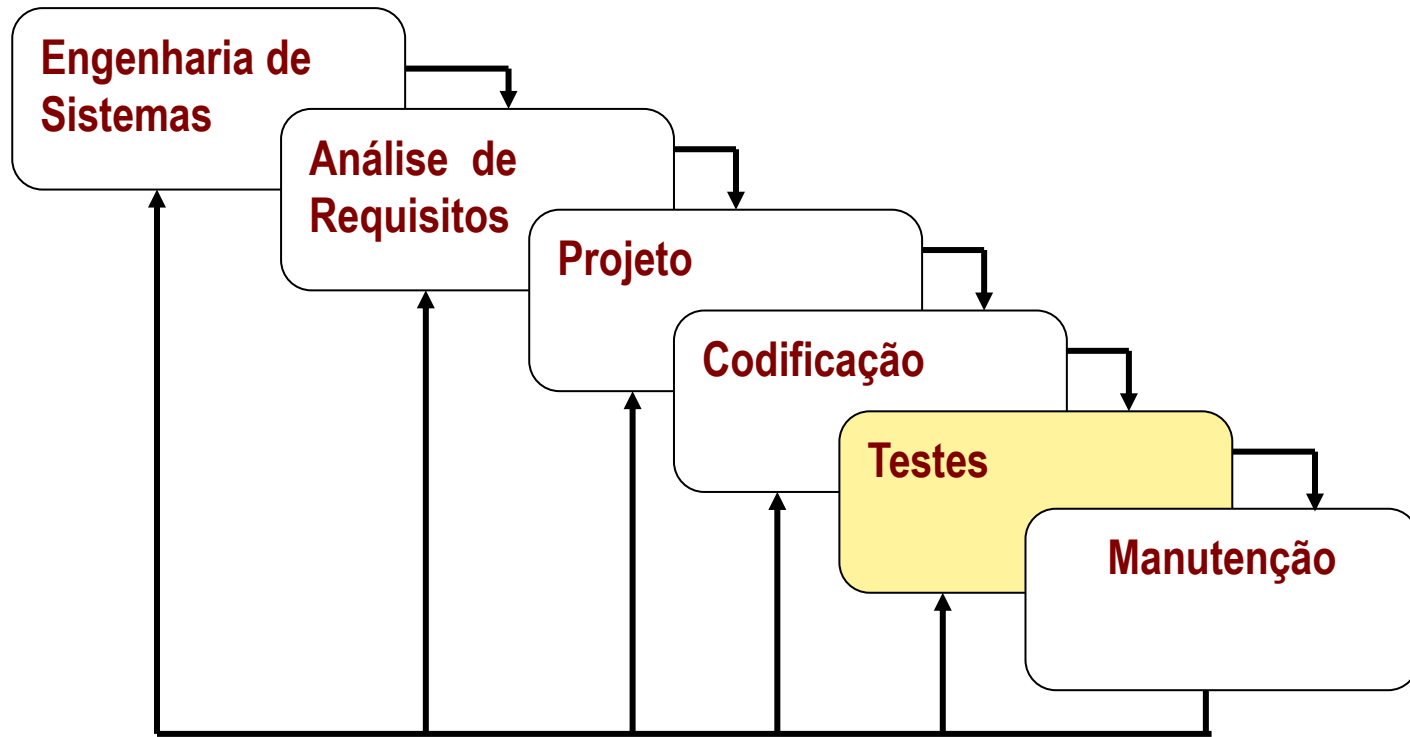
MODELO EM CASCATA (CICLO CLÁSSICO)

○ Codificação

- Implementação
- Tradução do projeto em código computacional
 - Instruções executáveis pelo computador
 - Linguagens de programação (alto ou baixo nível)
- Quanto mais coeso o projeto e os requisitos mais rápida é a codificação



MODELO EM CASCATA (CICLO CLÁSSICO)



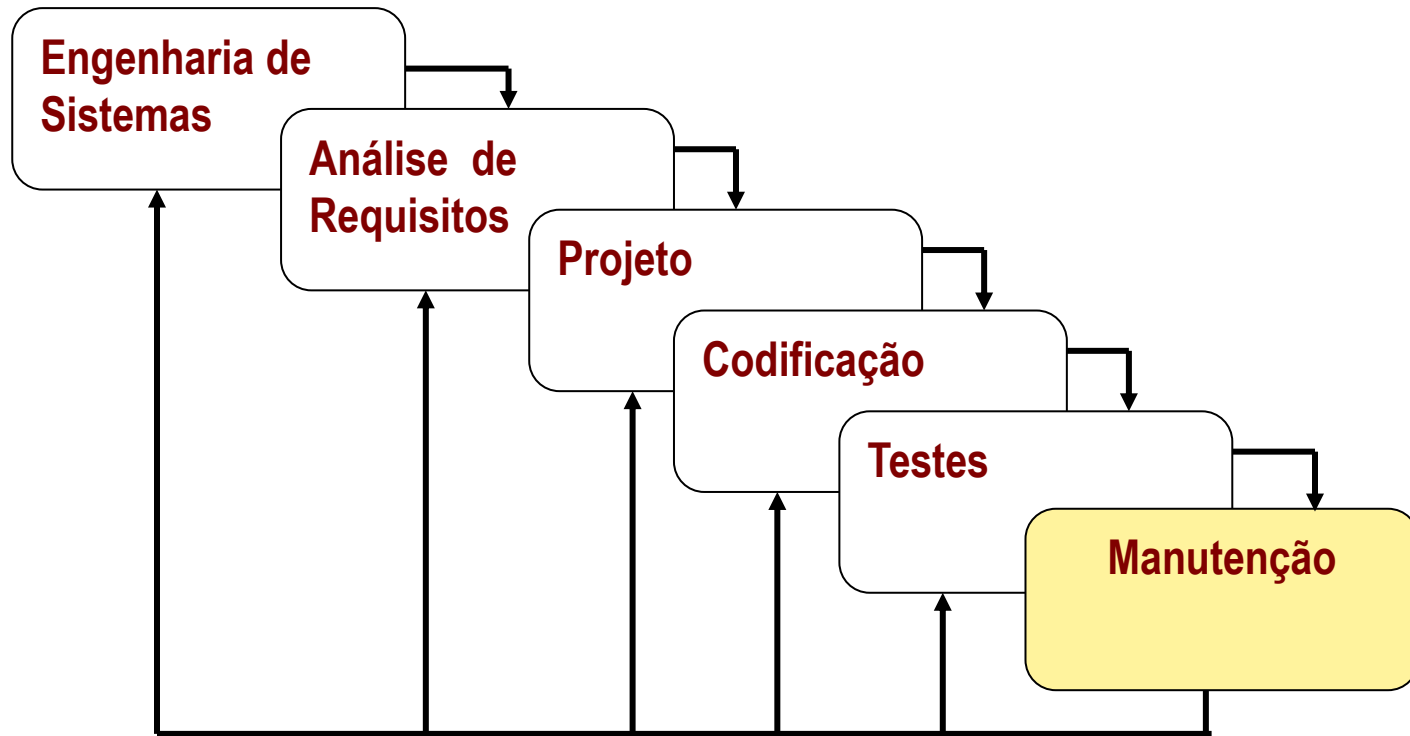
MODELO EM CASCATA (CICLO CLÁSSICO)

○ Testes

- Concentra os aspectos lógicos internos
 - Garante o teste de funcionalidade (código)
- Nos aspectos funcionais externos
 - Descobrir erros (teste de funcionalidade)
 - Entrada x produz saída y
 - Garantir a confiabilidade



MODELO EM CASCATA (CICLO CLÁSSICO)



MODELO EM CASCATA (CICLO CLÁSSICO)

○ Manutenção

- Alterações depois de entrega efetuada
- Mudanças ocorrem por:
 - Erros
 - Adaptação para acomodação de mudanças em processo organizacional
 - Exigência do cliente para acréscimo funcional
 - Em decorrência do desempenho



PROBLEMAS COM MODELO EM CASCATA

- Projetos raramente seguem o fluxo do modelo
- Dificuldade de estabelecer os requisitos no início do projeto
- O cliente deve ter paciência
 - Uma versão do produto só ficará disponível numa etapa avançada de desenvolvimento



MODELO EM CASCATA – COMENTÁRIO

- Mesmo com as fragilidades, ele é significativamente melhor que uma abordagem aleatória de desenvolvimento.
- Embora a entrega de uma versão “beta” seja tardia o resultado é satisfatório porem demorado.

