

INSTITUTO FEDERAL
Paraíba

Campus
João Pessoa

Análise e Projeto de Sistemas

Engenharia de Software

Profa. Nadja Rodrigues

nadja.rodrigues@academico.ifpb.edu.br

Alguns questionamentos ...

- Por que precisamos de tanto tempo para concluir o desenvolvimento de um software?
- Por que os custos com desenvolvimento de software são tão altos?
- Por que não conseguimos encontrar todos os erros antes de entregar o software para o cliente?
- Por que ainda continuamos a ter dificuldades em medir o progresso de como um software está sendo desenvolvido ?

Natureza do Software

- O software é desenvolvido/projetado, ao invés de manufaturado no sentido clássico.
- Software não se desgasta com o uso, mas se deteriora com as solicitações de mudanças!

Natureza do Software

- Uma falha de software indica um erro no projeto ou no processo de desenvolvimento. Mas não existem peças de reposição!
- A maioria dos softwares é feita sob medida (por encomenda), ao invés de ser montada a partir de componentes existentes.

Natureza do Software

- Derivado do esforço intelectual.
- Software não pode ser visto ou tocado: para analisar o progresso de um projeto de software é preciso recorrer à sua documentação.

Software

- Programas de computador e todos os dados de documentação e configuração associados, necessários para que o programa opere corretamente;
- Produtos de software podem ser desenvolvidos para um cliente particular ou podem ser desenvolvidos para um mercado geral;

Engenharia de Software

- Disciplina de engenharia relacionada a todos os aspectos de produção de software, desde os estágios iniciais de especificação do sistema até sua manutenção, depois que entrar em operação;

Engenharia de Software

- Relacionada não só com os processos técnicos de desenvolvimento de software, mas também com atividades como o gerenciamento de projeto de software e o desenvolvimento de ferramentas, métodos e teorias que apoiem a produção de software;
- Recomenda a aplicação de ferramentas e técnicas/métodos apropriados dependendo do problema a ser solucionado, das restrições de desenvolvimento e dos recursos disponíveis.

Métodos, Ferramentas e Procedimentos

- Métodos de Engenharia de Software
 - Detalhes de “como fazer” para construir um software;
- Ferramentas de Engenharia de Software
 - Criadas para apoiar de forma automatizada os métodos;
 - Ex.: ferramentas CASE;
- Procedimentos da Engenharia de Software
 - Elo entre métodos e ferramentas;
 - Definem, por exemplo, a seqüência em que os métodos serão aplicados e os produtos a serem entregues;

Disciplinas

- Modelagem de Negócio
- Análise de Requisitos
- Análise e Projeto
- Implementação
- Testes
- Implantação/Distribuição
- Gerenciamento e Projetos
- Gerência de Configuração e Mudanças
- Manutenção

Alguns Objetivos da Engenharia de Software

- Controle sobre o desenvolvimento de software dentro de **custos**, **prazos** e níveis de **qualidade** desejados;
- Sistematização da produção, manutenção e evolução dos produtos
- Qualidade versus Produtividade;
 - Aprimoramento da qualidade de software;
 - Aumento da produtividade;

Alguns Problemas no Desenvolvimento de Software...

- Alguns projetos são cancelados mesmo antes de começar;
- O tempo de desenvolvimento e o custo são bem maiores do que o estimado;
- Os sistemas não funcionam como planejado;
- A qualidade é baixa;
- A manutenção e a reutilização são difíceis e custosas;
- Os problemas são proporcionais à complexidade dos sistemas;

Alguns Problemas no Desenvolvimento de Software...

Na sua opinião, quais são as causas desses problemas?

Algumas Causas...

- Essência do Software
 - Natureza da atividade e cultura dos clientes;
 - Desafio intelectual para a equipe de desenvolvimento de software;
 - Complexidade dos sistemas;
 - Dificuldade de formalização;
- Essência Humana
 - Métodos, linguagens, ferramentas, processos, e modelos de ciclo de vida inadequados ou mal utilizados;
 - Inexperiência/falta de habilidade da gerência de projetos;
 - Falta de qualificação técnica dos desenvolvedores (pouco treinamento formal).

Mitos e Realidades do Software

- Existem três grandes grupos de mitos no desenvolvimento de software:
 - Mitos do Gerente de Projeto
 - Mitos do Cliente
 - Mitos do Desenvolvedor
- Vejamos alguns desses mitos...

Mitos e Realidades do Software



- **Mitos do Gerente de Projeto**

- **MITO:** “Se a equipe dispõe de um manual de padrões e procedimentos de desenvolvimento de software, então a equipe está apta a encaminhar bem o desenvolvimento”.
- **REALIDADE:** Só isso não é o suficiente. É preciso que a equipe aplique efetivamente os conhecimentos apresentados no manual... é necessário que o manual reflita a prática de desenvolvimento de software e que esta prática seja verificada com frequência, para confirmar o uso do conhecimento.
- **MITO:** “A equipe tem ferramentas de desenvolvimento de software de última geração, uma vez que ela dispõe de computadores de última geração”.
- **REALIDADE:** O fato da empresa ter hardware de última geração não garante que em nada a qualidade do software desenvolvido. Mais importante que ter um hardware de última geração é ter ferramentas para a automatização do desenvolvimento de software, as chamadas ferramentas CASE.
- **MITO:** “Se o desenvolvimento do software estiver atrasado, basta aumentar a equipe para honrar o prazo de desenvolvimento”.
- **REALIDADE:** Isto não condiz com a realidade. Quanto mais pessoas pegarem “o barco andando”, pior será, pois os novos profissionais deverão ser treinados e isto será feito por membros da equipe, o que vai implicar em maiores atrasos no cronograma.

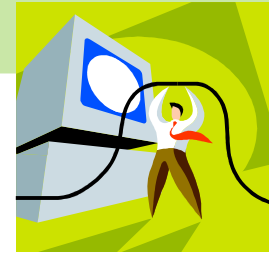
Mitos e Realidades do Software



- **Mitos do Cliente**

- **MITO:** "Uma descrição breve e geral dos requisitos do software é o suficiente para iniciar o seu projeto... maiores detalhes podem ser definidos posteriormente".
- **REALIDADE:** Este é um dos problemas que pode levar um projeto ao fracasso. O cliente deve ser questionado a fim de definir o mais precisamente possível os requisitos importantes para o software: funções, desempenho, interfaces, restrições de projeto e critérios de validação. Estes são pontos fundamentais para o sucesso do projeto.
- **MITO:** "Os requisitos de projeto mudam continuamente durante o seu desenvolvimento, mas isto não representa um problema, uma vez que o software é flexível e poderá suportar facilmente as alterações".
- **REALIDADE:** O software é mais flexível que a maioria dos produtos manufaturados, mas não existe software que suporte uma alteração significativa em seu escopo sem influenciar no custo e no prazo de entrega.

Mitos e Realidades do Software



- **Mitos do Desenvolvedor**

- **MITO:** “Após a construção do programa e a sua colocação em funcionamento, o trabalho está terminado”.
- **REALIDADE:** Isso não é verdade. Segundo estatísticas, após a implantação ocorrerão 50% a 70% do esforço do desenvolvimento de software (manutenção).
- **MITO:** “Enquanto o programa não entrar em funcionamento, é impossível avaliar a sua qualidade”.
- **REALIDADE:** A preocupação com a qualidade do software deve ocorrer durante todo o processo de desenvolvimento. Ao final de cada etapa deve ser feita uma verificação dos critérios de qualidade.
- **MITO:** “O produto a ser entregue no final do projeto é o programa funcionando”.
- **REALIDADE:** O produto final não é apenas o software funcionando, mas também toda a documentação do Software.

Processo de Desenvolvimento de Software

- Conjunto de atividades e resultados associados que produz um produto de software;
- São atividades comuns a todos os processos de software:
 - Especificação de software: definição do software a ser produzido;
 - Desenvolvimento de software: o software é projetado e programado;
 - Validação de software: o software é verificado para garantir que é o que o cliente deseja;
 - Evolução de software: adaptação do software às mudanças de requisitos.

Reflexões

1. Qual a importância de

Sistematizar as Atividades
do Desenvolvimento de
Sistemas?

Processo de Desenvolvimento de Software

- A utilização de um processo de software tem sido apontada como um fator primordial para o sucesso de empresas de desenvolvimento de software.
- O termo ciclo de vida evoluiu para processo.

O que é mesmo um processo de software?

Processo de Desenvolvimento de Software

- Um processo de software pode ser entendido como um conjunto estruturado de atividades exigidas para desenvolver um produto de software.

Processo de Desenvolvimento de Software

- Características das atividades:
 - bem definidas;
 - com responsáveis;
 - com artefatos de entrada e saída;
 - com dependências entre as mesmas e ordem de execução;
 - com “modelo de ciclo de vida”.

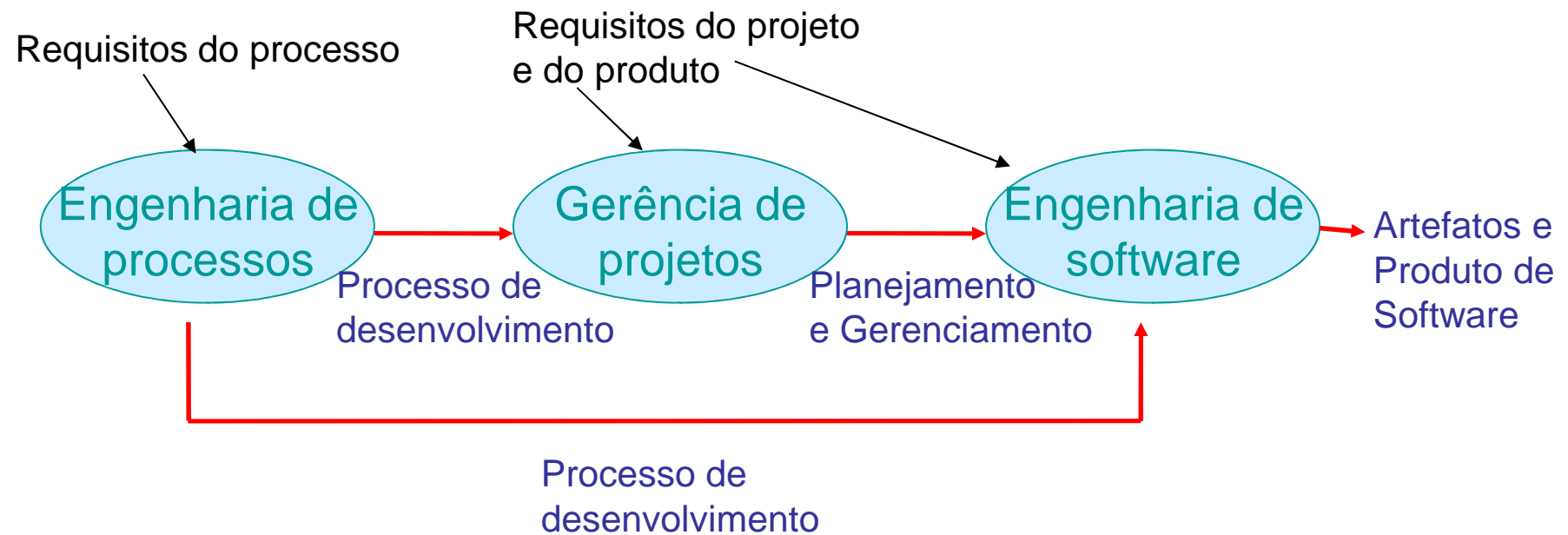
Processo de Desenvolvimento de Software

- Processo de software padrão é composto por:
 - Fases/atividades (entradas e saídas),
 - procedimentos/métodos,
 - e ferramentas, utilizados para construir um produto de software.

Processo de Desenvolvimento de Software

- Processo de software padrão pode ser dividido nos seguintes sub-processos:
 - Processos Fundamentais (Análise de Negócio, Requisitos, Análise e Projeto, Implementação, Testes e Implantação);
 - Processos de Apoio (Gerência de Projetos, Garantia da Qualidade, Gerência de Configuração e Mudança, Medição e Análise);
 - Processos Organizacionais (Infra-estrutura, Treinamento, Aquisição).

Processo de Desenvolvimento de Software



Processo de Desenvolvimento de Software

- Enfim, o processo de desenvolvimento de software define quem faz o quê, quando e como para atingir um certo objetivo.

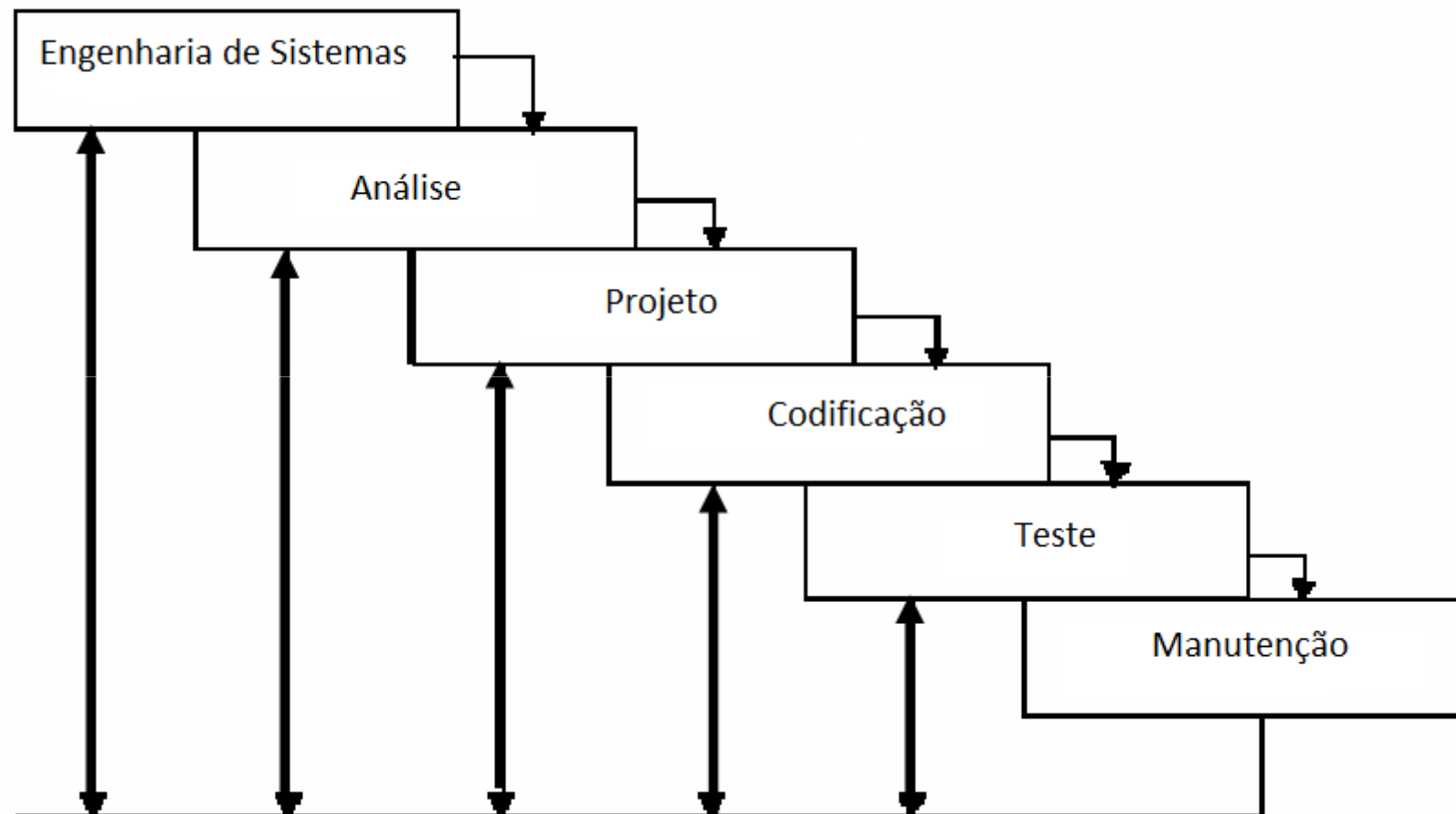
Processo de Desenvolvimento de Software

- Algumas razões para a definição e utilização de um processo:
 - Redução dos problemas relacionados a treinamento, revisões e suporte à ferramentas;
 - As experiências adquiridas nos projetos são incorporadas ao processo padrão e contribuem para melhorias em todos os processos definidos;
 - Economia de tempo e esforço na definição de novos processos adequados a projetos.

Modelos de Processos de Software

- Vejamos alguns dos mais conhecidos modelos de processos de software...

Cascata



Cascata – Principais características

■ Atividades:

□ Análise e Engenharia de Sistemas:

- Envolve a coleta dos requisitos em nível de sistema;
- Busca a compreensão do sistema como um todo, incluindo hardware, software, bancos de dados e pessoas envolvidas;

□ Análise de Requisitos e Software:

- Intensifica e concentra a definição dos requisitos do software;
- O resultado será utilizado nas etapas posteriores de Projeto, Construção, Testes e Manutenção;

Cascata – Principais características

■ Atividades:

□ Projeto:

- Transforma os requisitos da análise, o modelo lógico, em modelo físico;
- Define a arquitetura do software, detalhes procedimentais e de interface, dentre outros;

□ Codificação:

- Implementa os programas e relatórios do sistemas com base nas especificações das fases anteriores;

Cascata – Principais características

- Atividades:

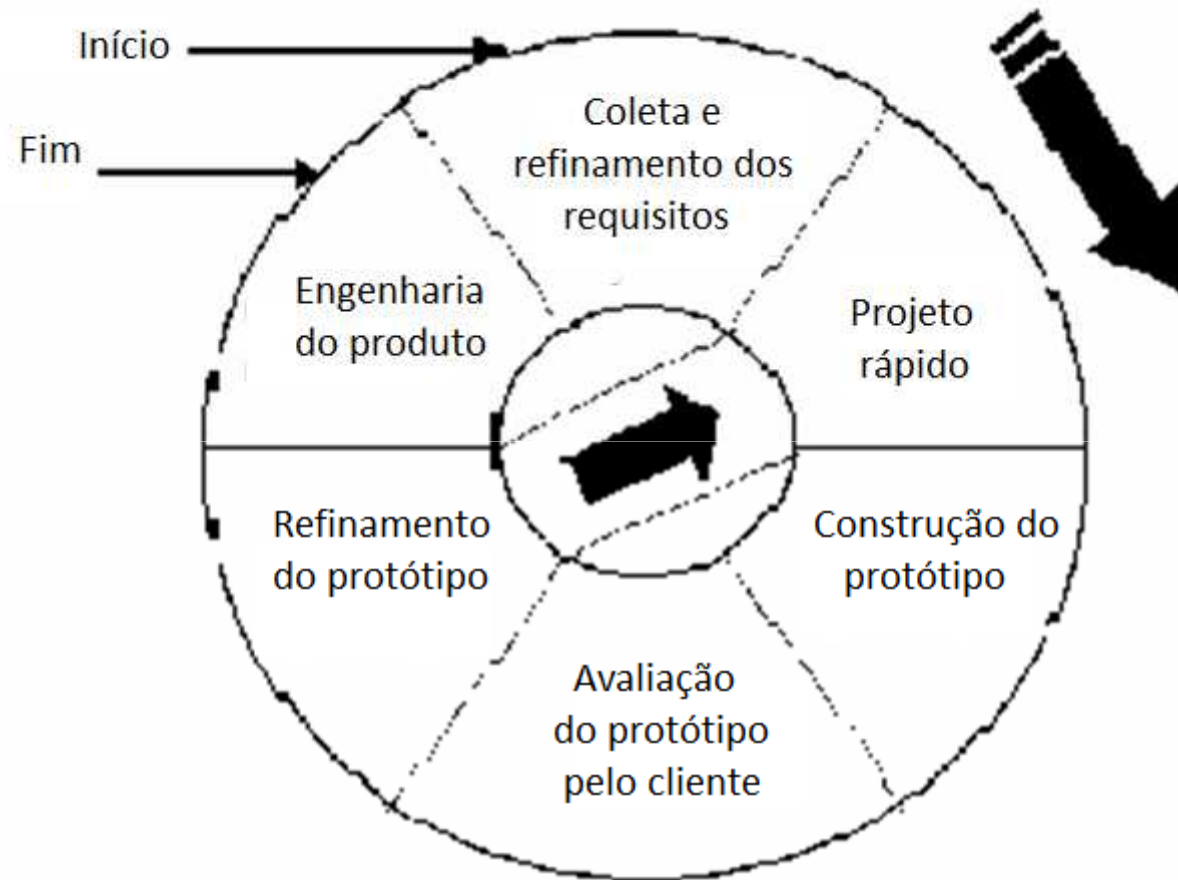
- Testes:

- Realiza os testes do sistema (aspectos lógicos internos e aspectos funcionais externos);

- Manutenção:

- Constrói alterações no software que visem a solução de erros, melhorias ou inclusão de novas funcionalidades.

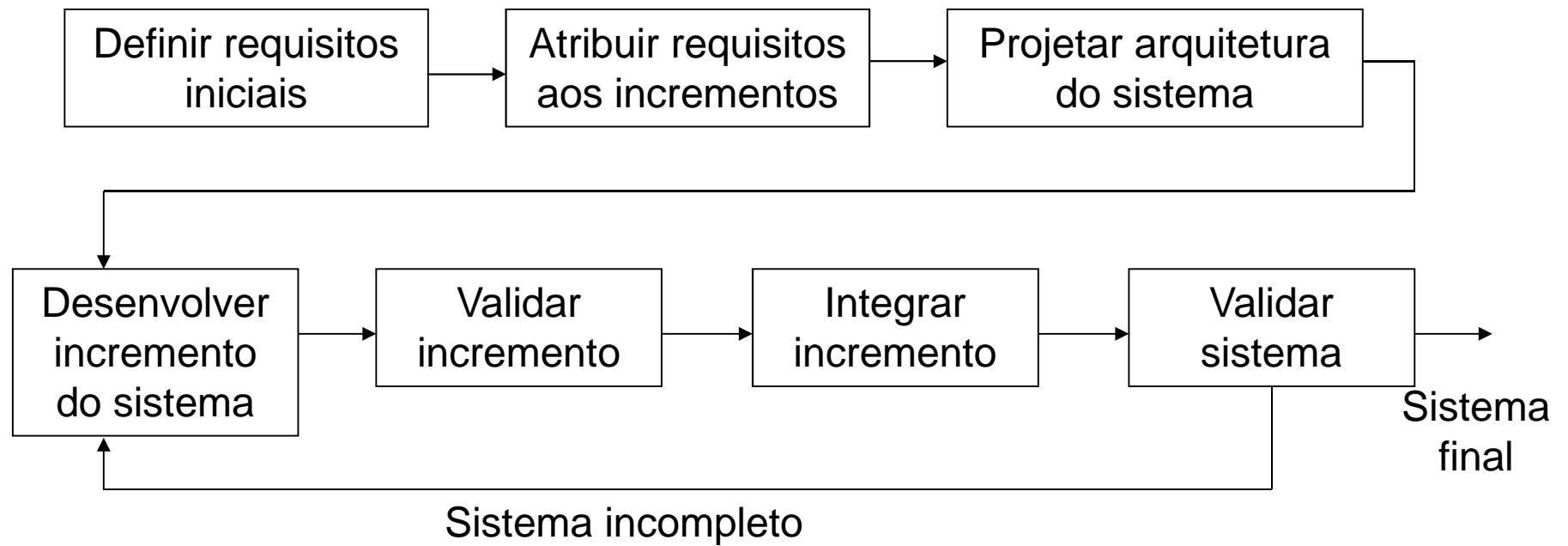
Prototipação



Prototipação – Principais características

- Capacita o desenvolvedor a criar um modelo do software que será implementado;
- Tem o objetivo de resolver problemas ou limitações existentes na análise de requisitos, antes do início da implementação do software;
- O protótipo pode ser apresentado ao cliente na forma de papel ou modelo executável no computador, retratando a interface do software;
- Com o protótipo, o cliente poderá verificar se os requisitos especificados são os desejados ou reformulá-los se houver necessidade.

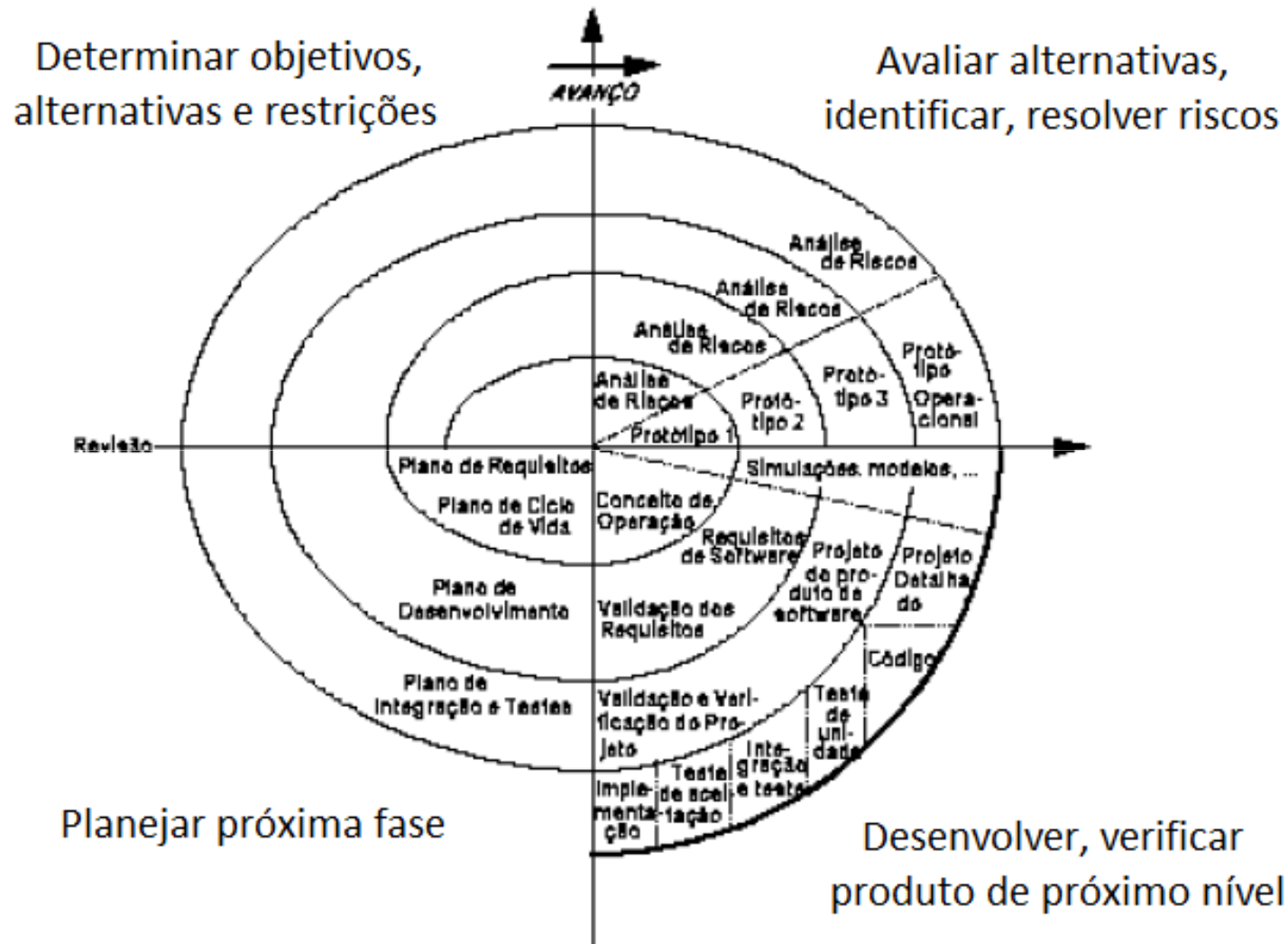
Modelo Incremental



Modelo Incremental – Principais características

- O desenvolvimento e a entrega são divididos em incrementos, com cada incremento adicionando ao sistema novas capacidades funcionais, até a obtenção do sistema final;
- Requisitos dos usuários são priorizados e os requisitos de mais alta prioridade são incluídos nas iterações iniciais;
- A arquitetura global deve estar pronta antes da construção do primeiro incremento.

Modelo Espiral



Modelo Espiral – Principais características

- Desenvolvido para abranger as melhores características do ciclo de vida clássico e da prototipação;
- Acrescenta um novo elemento: a análise de riscos;
- O processo é representado como uma espiral em vez de uma seqüência de atividades;
- Cada volta na espiral representa uma fase no processo.

Reflexões

2. Análise Crítica dos Modelos de Processos de Software.

Modelos de Processos de Software

- Nas próximas aulas falaremos mais sobre modelos de processos de software...