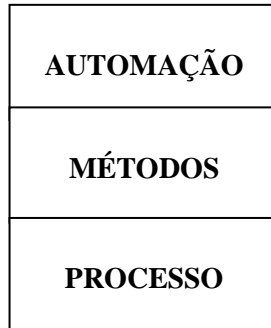
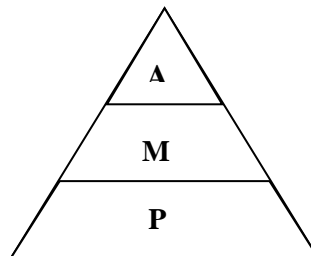


TIPOS DE AMBIENTE

1. AMBIENTE IDEAL

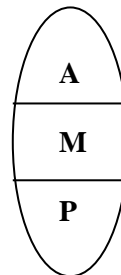


2. AMBIENTE TÍPICO



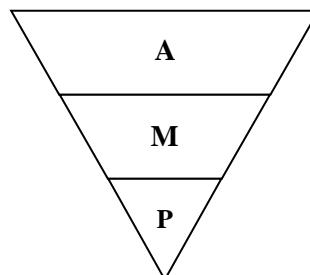
3. “BOLA DE FUTEBOL”

Muitos métodos, automação insuficiente, sem suporte para o processo.



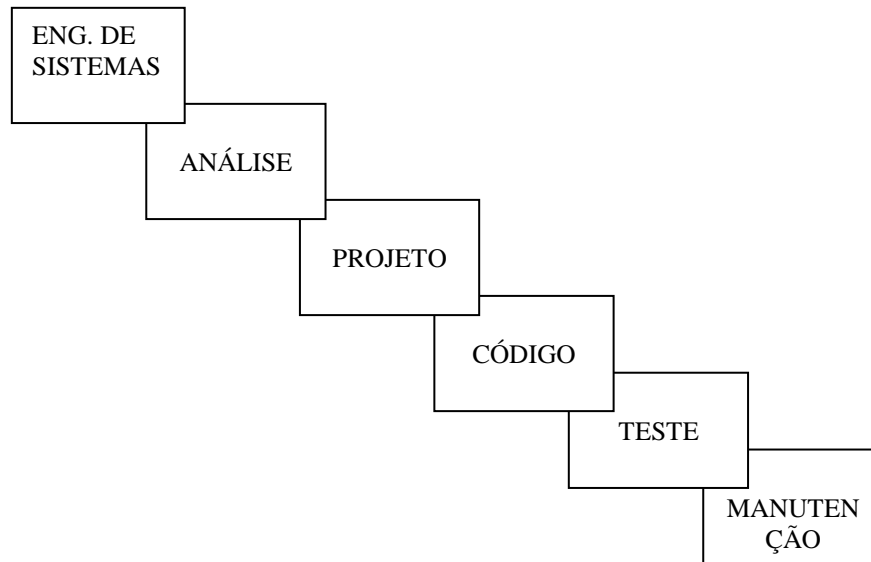
4. “PIRÂMIDE INVERTIDA”

Excesso de ferramentas CASE sem o suporte metodológico!



ALGUNS PROCESSOS

1 – Modelo “WATERFALL (Cascata)”



Problemas

- Ocorrência de iterações;
- Dificuldades em se explicitarem todos os requisitos no início do processo;
- Versão do problema só está disponível no fim do processo;
- Dificuldades no gerenciamento;
- Difícil de automatizar completamente.

COMO É O GERENCIAMENTO DO MODELO “WATERFALL” ?

- Cada fase necessita de recursos de:
 - Tempo
 - Mão-de-obra
 - Fundos
 - Facilidades
- Ao fim de cada fase realizam-se tarefas de:
 - Garantia de qualidade
 - Verificação e Validação
 - Gerência de Configuração

Apesar de tudo, o modelo “waterfall” é uma espécie de gabarito para aplicações de métodos para análise, projeto, codificação, teste e manutenção.

É MAIS ADEQUADO A PEQUENOS PROJETOS

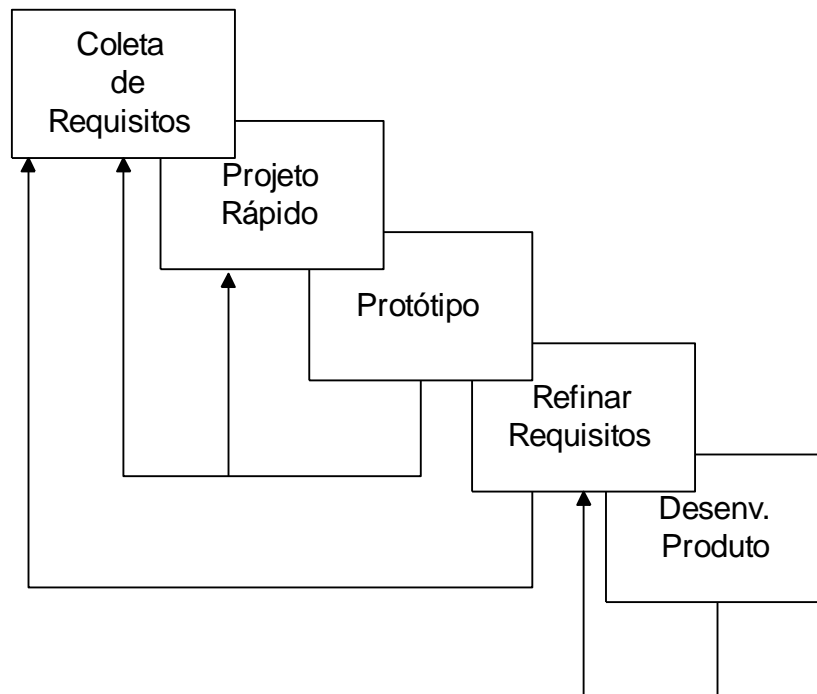
2 - A Prototipação

Permite ao analista criar um **modelo executável ou não** do software a ser construído.

Tipos de Modelos

- No papel \Rightarrow interface homem-máquina.
- Programa semelhante no proposto (Rulm).
- Executável.

Seqüência de Eventos



Problemas

- Gerenciamento obscuro.
- A imagem de produto “já pronto” gera ilusão no cliente.
- O protótipo executável é gerado com técnicas pouco eficientes. (Muitas vezes isto ocorre!)

3 - Desenvolvimento Incremental (Prototipação Rápida)

- **Conceito básico:** desenvolvimento incremental \Rightarrow criação do software em incrementos pequenos, onde cada novo incremento é o sistema antigo com funcionalidades adicionais. Cada incremento é um sistema funcionando.

Vantagens

- Aumento do poder descritivo.
- Ressaltam-se conceitos importantes em E.S. com verificação e validação.
- Visibilidade do sistema final é antecipada.

Desvantagens

- Se não houver verificação, **alta propagação de erros**.
- Gerenciamento é complexo, por se construírem muitos subsistemas.

4 - Modelo Operacional

- Utiliza uma **linguagem** que produz uma **especificação executável**.
- Esta especificação não é um protótipo, mas uma forma de se **simular** operações executadas pelo sistema que expressem o que o sistema deve fazer.

Vantagens

- Alto poder descritivo: as descrições são formais, rigorosas e podem ser analisadas!
- Modelo executável pode ser facilmente avaliados pelo cliente e pelo desenvolvedor.
- Pode ser facilmente automatizável!
- Comportamento do sistema está implícito na especificação.

Desvantagens

- Fronteira Requisitos - Implementação é mal definida: desenvolver pode ser induzido a pensar em implementações, esquecendo requisitos.
- Gerenciamento obscuro.

5 - Modelo Baseado em Conhecimento

- Utiliza Inteligência Artificial.
- O computador age com um “assistente especialista” guiando o desenvolvedor através do processo.
- Separa **domínio da engenharia de software** do **domínio da aplicação**.
- Regras baseadas em técnicas de engenharia de software.
- Regas formadas a partir do conhecimento da aplicação.

Dois sistemas especialistas:

- com conhecimento de engenharia de software.
- com conhecimento da aplicação.

Vantagens

- Alto poder descritivo, generalidade e adequabilidade à automação.

Desvantagem

- Tecnologia para a implementação

COMPARAÇÃO ENTRE OS PROCESSOS VISTOS

| PROCESSO | PODER DESCRITIVO | GENERALIDADE | ADEQUABILIDADE X AUTOMAÇÃO |
|--------------------------------|-------------------------|---------------------|-----------------------------------|
| Cascata | B | B | B |
| Prototipação | M | M | M/A |
| Desenv. Incremental | M | M | M/A |
| Operacional | M/A | M | M |
| Baseado em conhecimento | A | A | A |

B = Baixa
M = Média
A = Alta

PROCESSO PADRÃO DO DEPTO. DE DEFESA AMERICANO

O processo padrão utilizado pelo U.S. Department of Defense é inteiramente embasado na norma denominada DOD - STD - 21474.

Esta norma norteia todo o processo de desenvolvimento adotado, deixando bem claro todas as atividades a serem realizadas e todos os produtos a serem emitidos durante o ciclo-de-vida.

FASES:

SISTEMA

- Análise de Requisitos
- Projeto

SOFTWARE

- Análise de Requisitos
- Projeto Preliminar
- Projeto Detalhado
- Codificação e testes de C.S.U.
- Integração e testes de C.S.C.
- Testes de C.S.C.I.

HARDWARE

- Mesmas fases até projeto detalhado
- Fabricação
- Testes de Integração de HWCI.

COMUNS

- Integração de Sistema.
- Avaliação e teste operacional.

- Produção

Características

- Não determina quais métodos devem ser utilizados. Especifica apenas o processo.
- Os métodos são detalhados pelo desenvolvedor no chamado plano de desenvolvimento de software emitido após a fase de projeto do sistema e antes da análise de requisitos de software.
- As atividades são realizadas, bem como alguns produtos, para cada um dos chamados itens de configuração.
- Os métodos detalhados no plano de desenvolvimento de software devem se aplicar a atividades de desenvolvimento técnico, gerência de configuração, avaliação de qualidade e gerenciamento de custos/tempo.
- A gerência de configuração é feita sobre as chamadas “baselines” (marcos) do desenvolvimento. Estas “baselines” terão sua configuração identificada, controlada, mapeada e auditada no decorrer do desenvolvimento. Estas mesmas são, normalmente, documentos ou códigos-fontes. São as referências, metas para todo o ciclo-de-vida.
- Os itens de Configuração (C.S.C.I.) são partes do software colocadas e implementadas em conjunto, escolhidas por critérios muito bem definidos por outra norma (ISSP-B), que estão sob o que chamamos controle de configuração. São divididos, para efeito de simplificação na implementação, em C.S.C. (componentes de software), que por sua vez se dividem em C.S.U. (Unidades de Software - Módulos).

Por que dividir o software em C.S.C.I.?

- Devido ao grande tamanho dos softwares desenvolvidos pela instituição, torna-se necessário a divisão de forma a garantir-se melhor controle sobre as atividades de gerenciamento, desenvolvimento e qualidade.