

# ***Engenharia de Software***

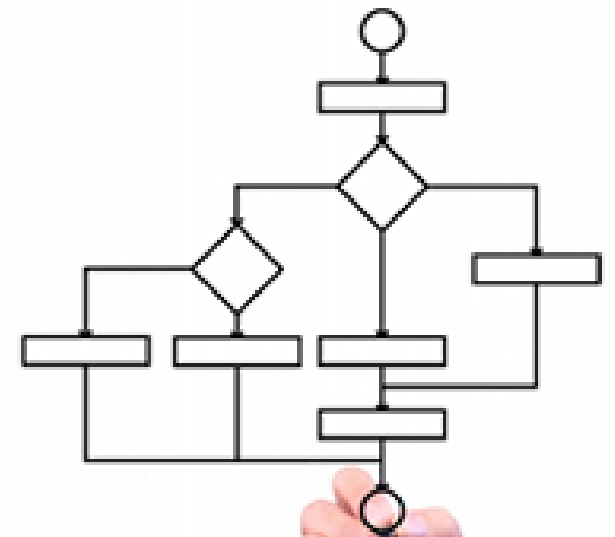
## *O processo*

Prof. Sanderson Macedo  
Aula 02

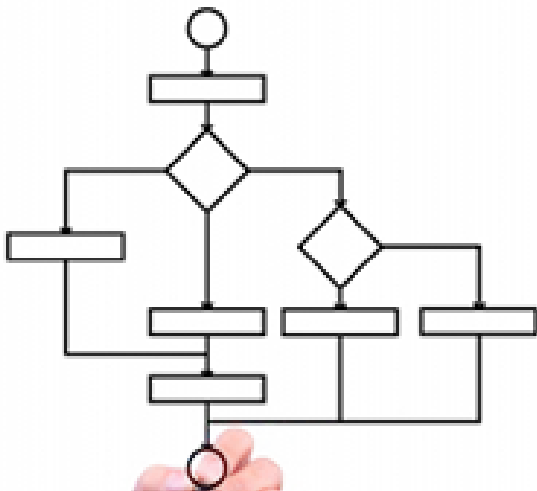
# ***O que vamos ver***

- O que é processo;
- Modelos de processo de software.

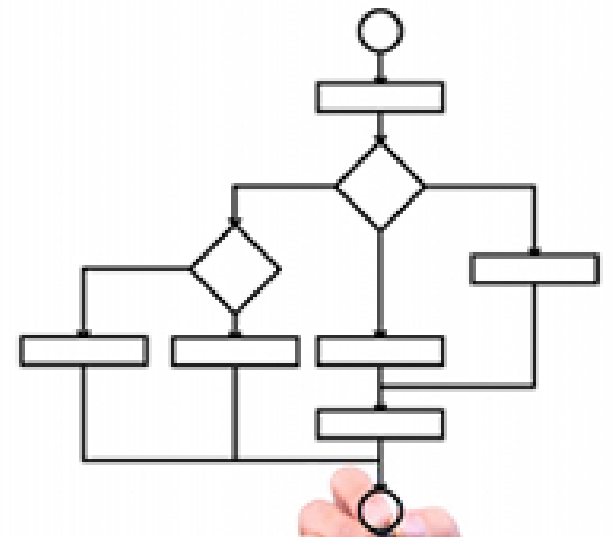
***Como você  
faz as coisas?***



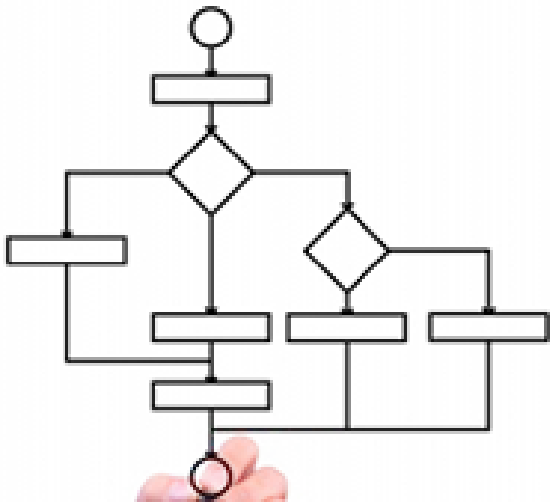
***Você repete certas  
atividades para  
fazer algo rotineiro?***



***Já melhorou as atividades  
e no fim o resultado foi  
melhor?***



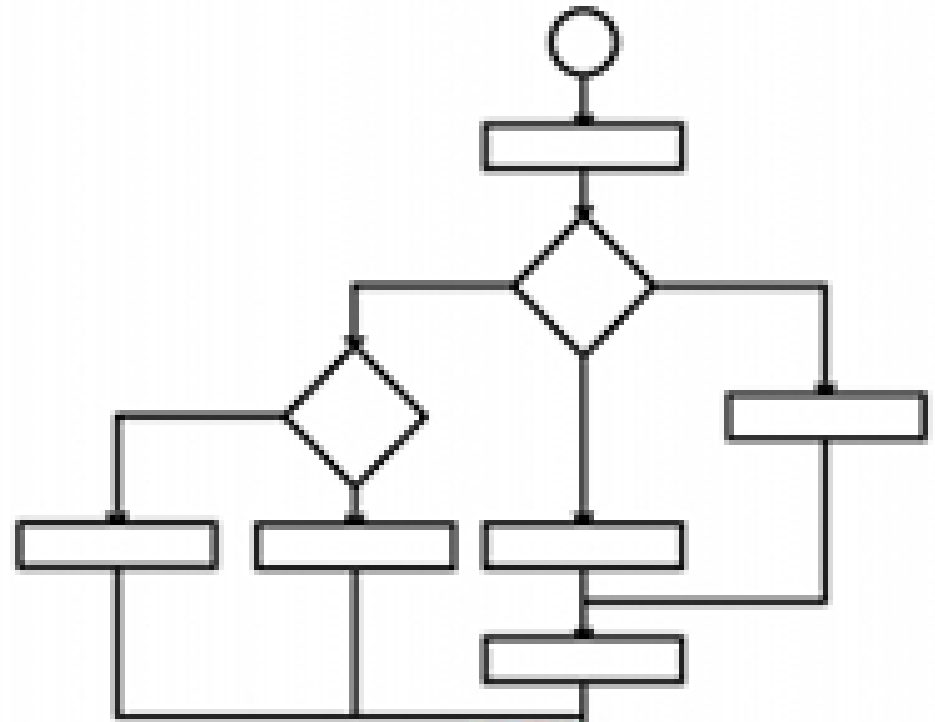
As atividades para  
se fazer algo  
são chamadas de  
***processo***



# ***Camadas da Engenharia de SW***

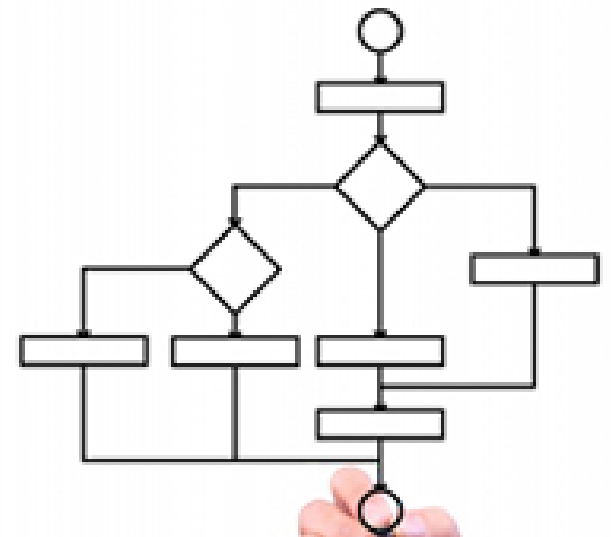


# ***Diagrama de actividades***



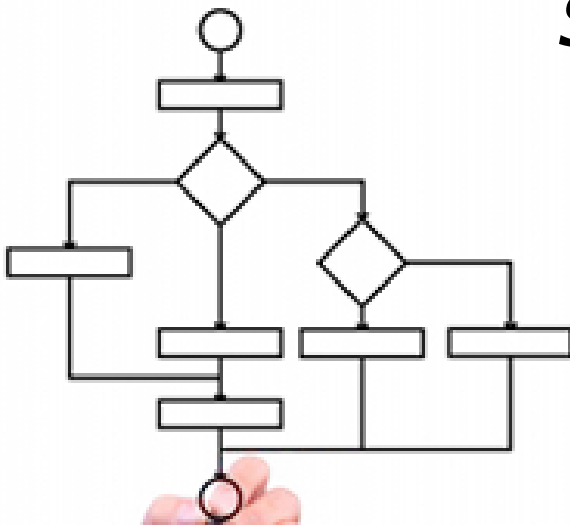


***O que é  
processo  
de software?***



# ***Processo de software***

*É um conjunto de atividades necessárias para construir um software de alta qualidade.*

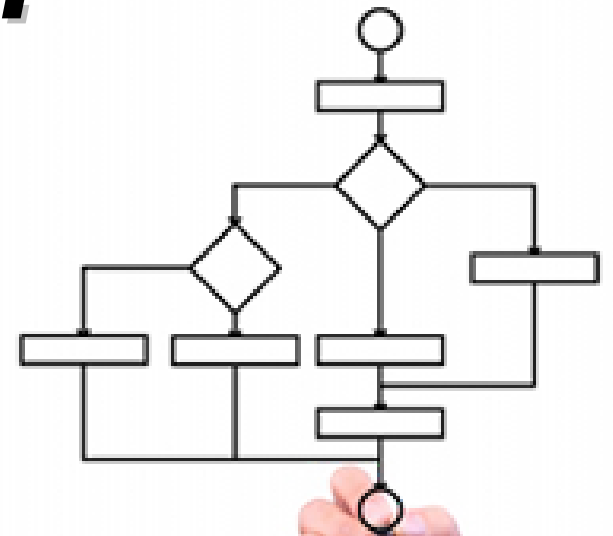


# ***Métodos***

Conjuntos de tarefas mais comuns na engenharia de SW

- Levantamento de requisitos;
- Análise dos requisitos;
- Projeto de software;
- Construção;
- Teste;
- Manutenção;
- Implantação.

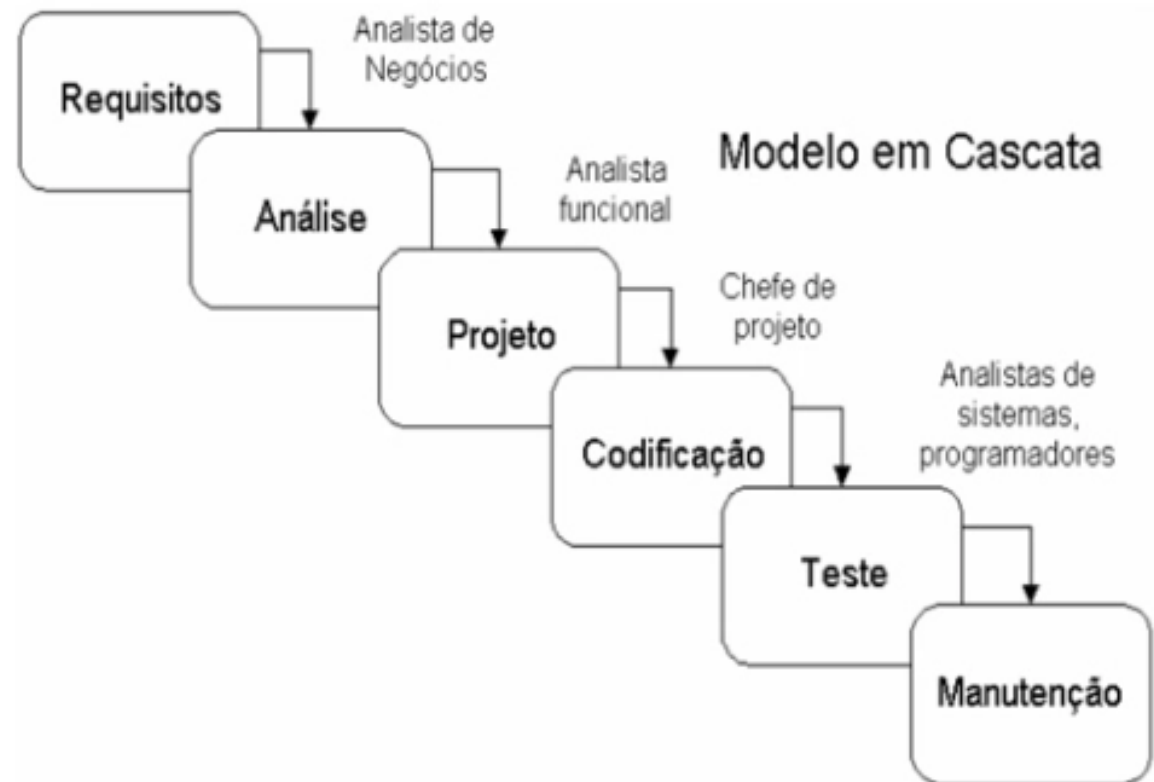
***Qual é a melhor  
sequência das tarefas?***



# ***Modelo Cascata - Tradicional***

O modelo em cascata é um modelo de desenvolvimento de software sequencial no qual o desenvolvimento é visto como um fluir constante para frente.

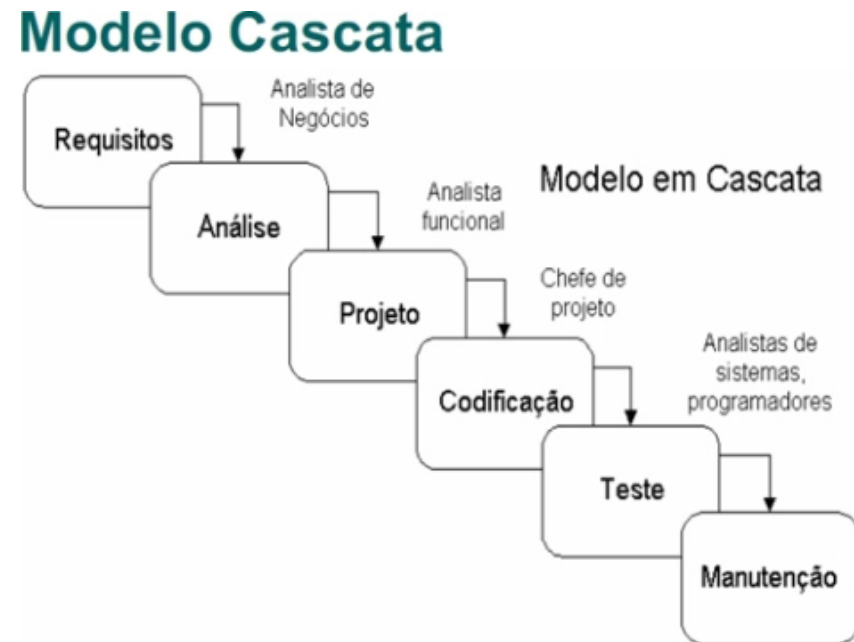
## **Modelo Cascata**



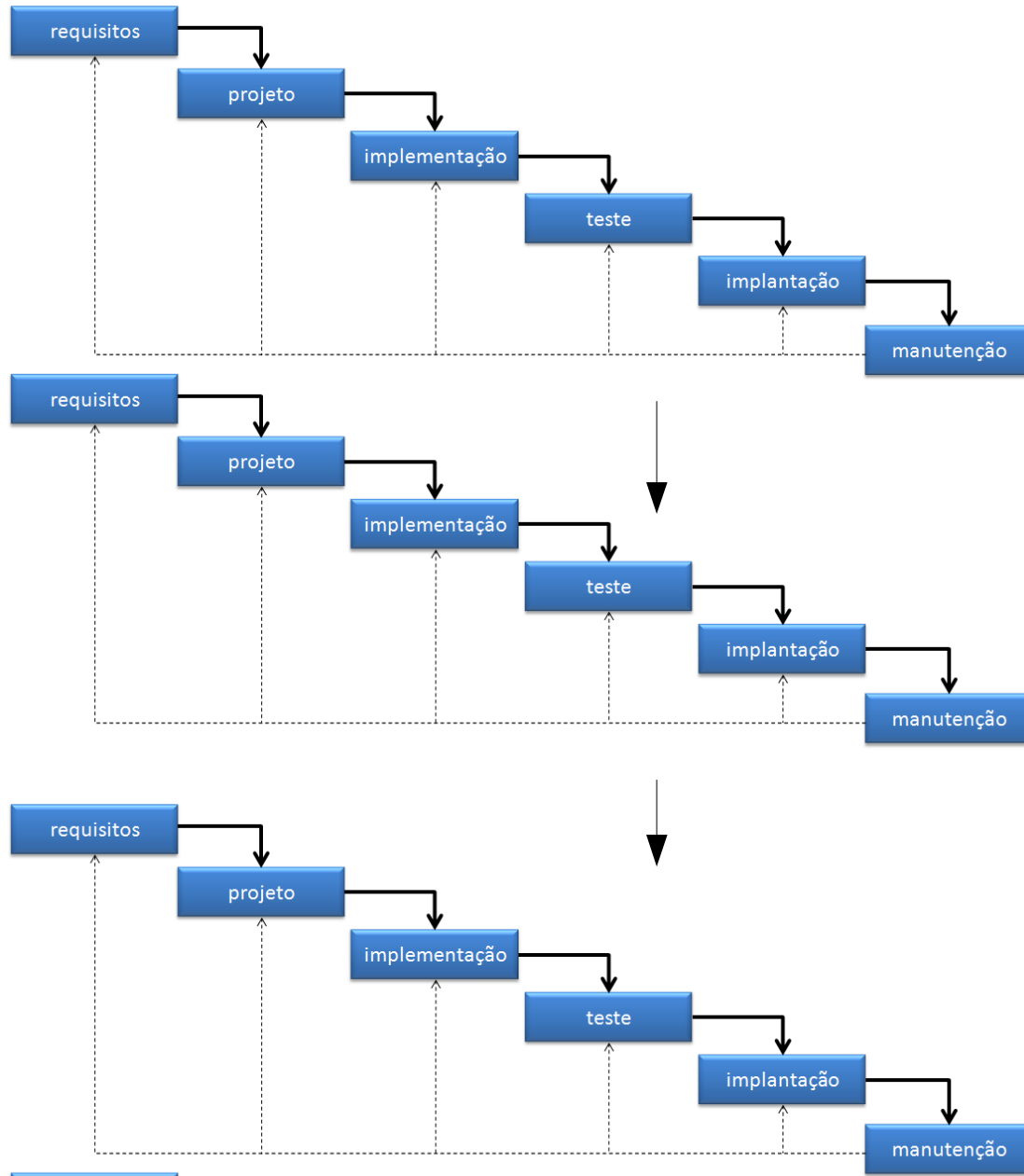
# Modelo Cascata

Foi originalmente concebido para trazer ordem ao caos que existia na área de desenvolvimento de software. Mas ao longo do tempo mostrou-se ineficiente.

- Cliente não sabe de todos os requisitos;
- Bloqueio das equipes;
- Cliente só recebe o SW no fim.



# ***Modelo Iterativo Incremental***



1º incremento

2º incremento

3º incremento

nº incremento

# ***Modelo Iterativo Incremental***

## ***Vantagens:***

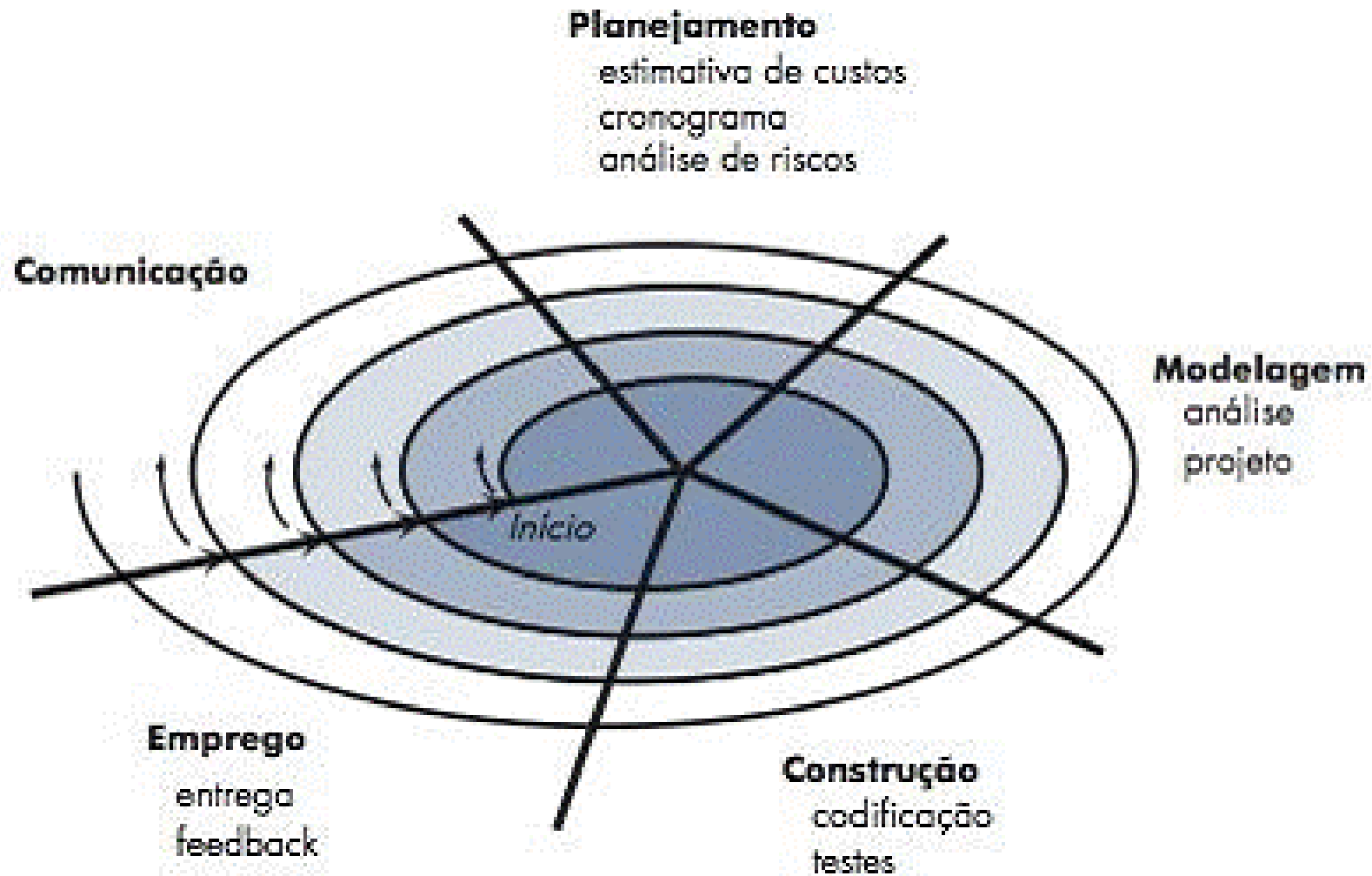
- Composto por pequenas cascatas;
- Criado em resposta às fraquezas do modelo em cascata;
- Entrega do SW por partes;
- Cliente recebe rapidamente uma porção do SW;

## ***Desvantagens:***

- O usuário pode se entusiasmar excessivamente com a primeira versão do sistema;
- Termina quando o SW é entregue.



# ***Modelo Espiral***



# ***Modelo Espiral***

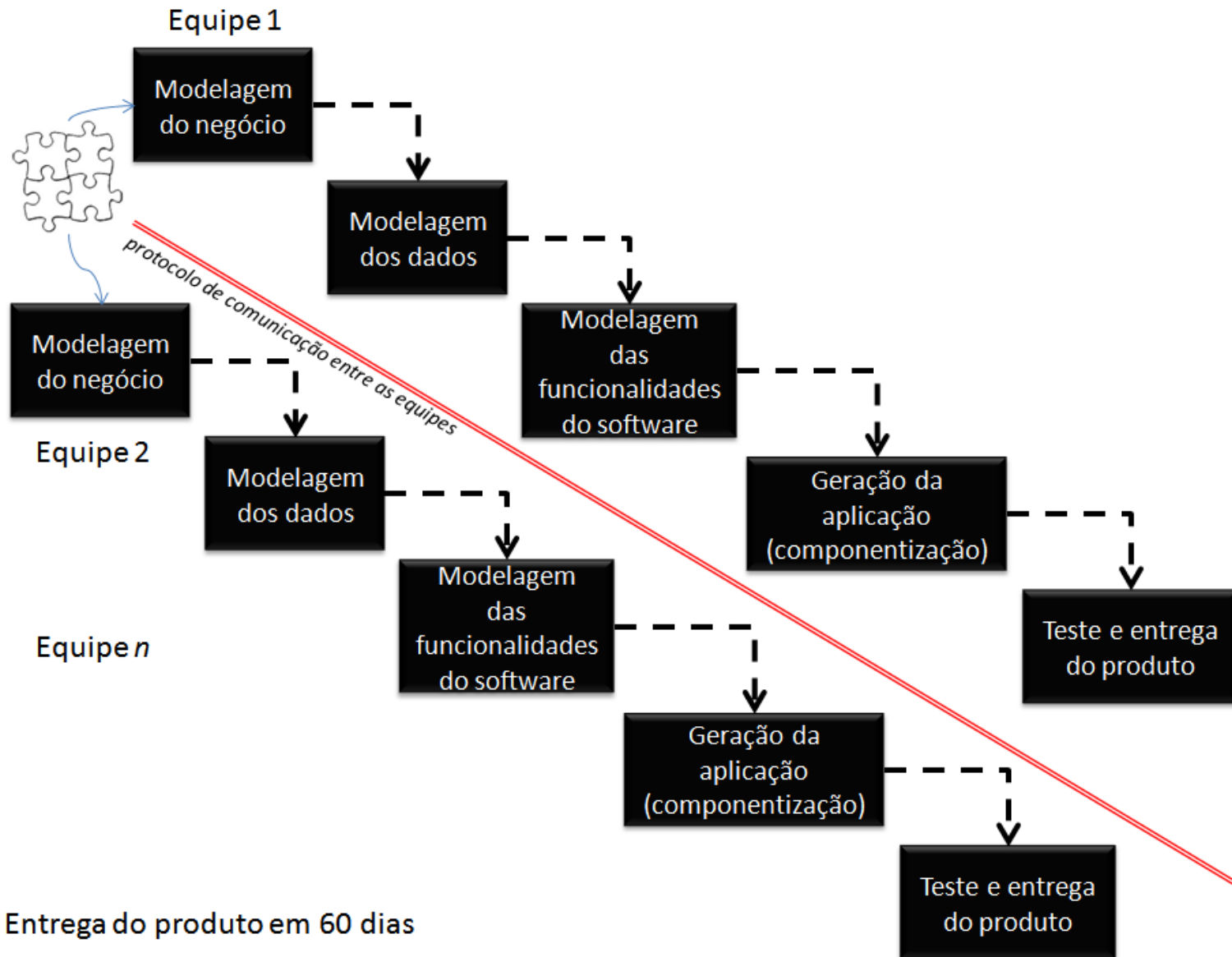
## ***Vantagens:***

- Perene - Não termina quando o SW é entregue;
- Suporta mecanismos de redução de risco;
- Mais versátil para lidar com mudanças;

## ***Desvantagens:***

- Descarte do protótipo;
- Pode haver muitos ajustes no protótipo final, para aumento da qualidade;
- O desenvolvedor pode esquecer estruturas inapropriadas no protótipo.

# Modelo RAD



# ***Modelo RAD***

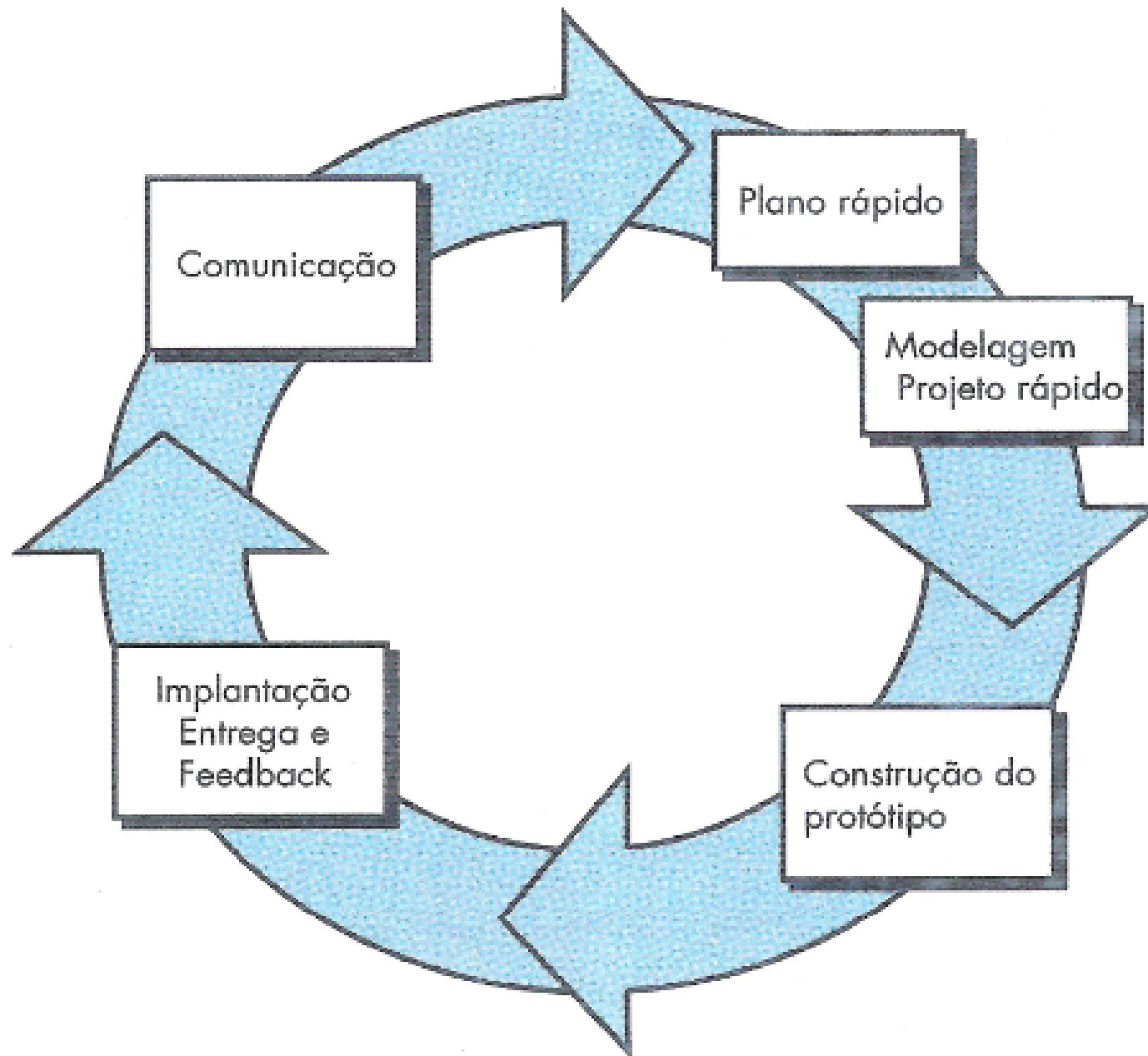
## ***Vantagens:***

- Linhas paralelas de desenvolvimento;
- Entrega rápida entre 60 e 90 dias;
- Trabalha com equipes;
- Desenvolvido com componentes de SW.

## ***Desvantagens:***

- SW grande. Número de pessoas cresce;
- Mais caro para desenvolver;
- Sinergia entre clientes desenvolvedores.

# ***Modelo de prototipação***



# ***Modelo prototipação***

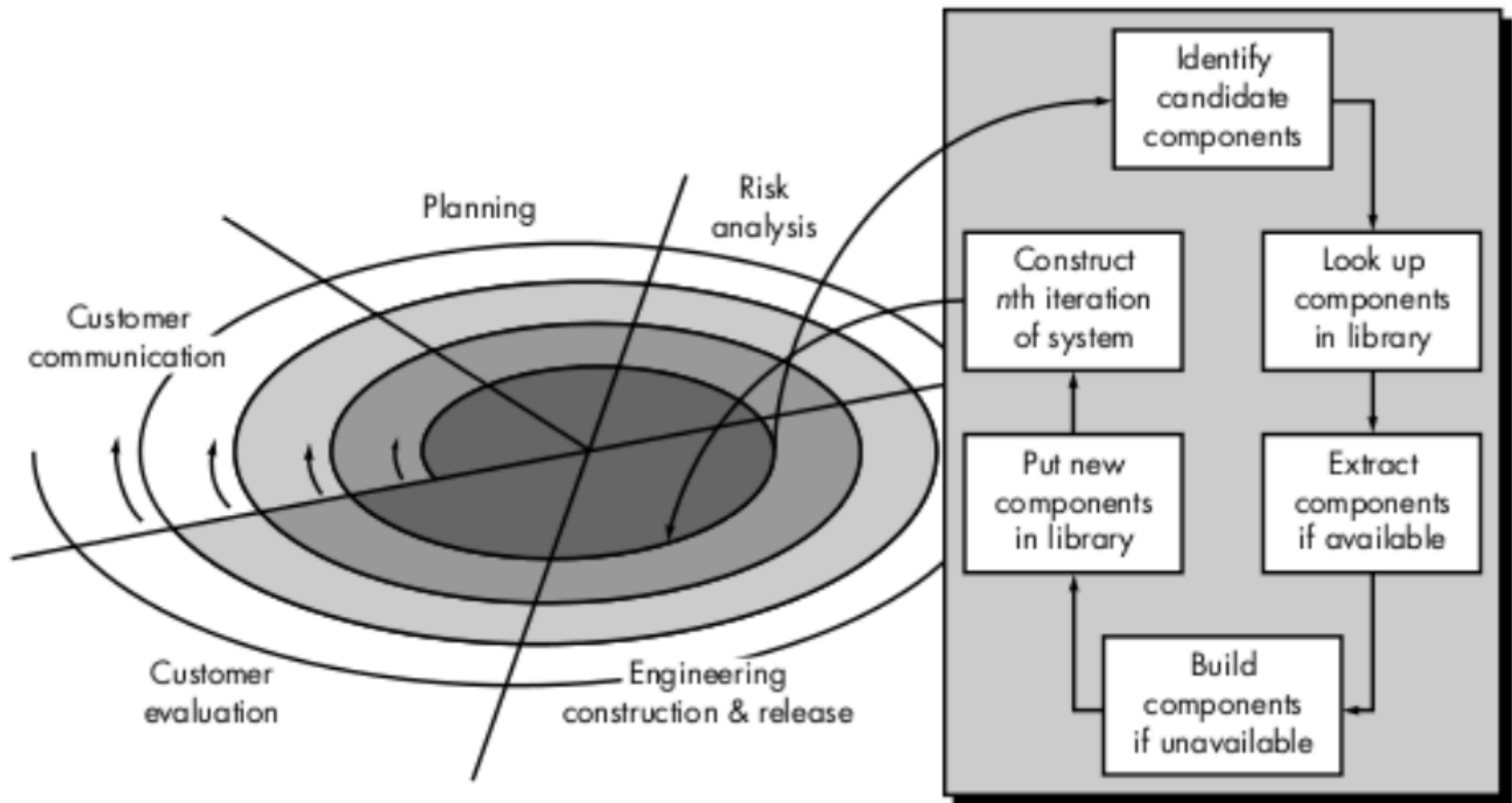
## ***Vantagens:***

- Sistema atual melhora a percepção do usuário em relação software;
- O desenvolvedor constrói algo imediatamente;

## ***Desvantagens:***

- Descarte do protótipo;
- Pode haver muitos ajustes no protótipo final, para aumento da qualidade;
- O desenvolvedor pode esquecer estruturas inapropriadas no protótipo.

# ***Desenvolvimento baseado em componentes***



# ***Desenvolvimento baseado em componentes***

## ***Vantagens:***

- Utiliza e desenvolve com base no reuso;
- Melhoramento do Espiral;
- Reuso melhor o custo final do projeto.

(Ex.: content management system)

## ***Desvantagens:***

- Há necessidade de codificação genérica - Reuso;
- Dura manutenção e versionamento das Bibliotecas;



# ***Outros modelos***

## ***Espiral ganha-ganha***

- Alta negociação entre clientes e desenvolvedores

## ***Modelo Formal***

- Baseados em formalismos matemáticos para a especificação, desenvolvimento e verificação dos sistemas de softwares e hardwares.

## ***Modelo desenvolvimento concorrente***

- Atividades de requisito, análise, projeto, codificação... etc... ocorrem ao mesmo tempo.
- Usado para sistemas de componentes em cliente/servidor

# ***Outros modelos***

## ***Técnicas de 4ª Geração***

- Baseada em ferramentas de geração automática de software;
- O Desenvolvedor especifica em alto nível e a ferramenta gera o código.
- Exemplos são os editores WYSIWYG que geram HTML

(what you see is what you get)