

FIAP GRADUAÇÃO

I ENGENHARIA DE SOFTWARE

METODOLOGIA DE DESENVOLVIMENTO DE SOFTWARE

É UM CONJUNTO DE **BOAS PRÁTICAS** APLICADAS
NA **ESTRUTURAÇÃO E CONTROLE** DO
PROCESSO DE DESENVOLVIMENTO DE SOFTWARE

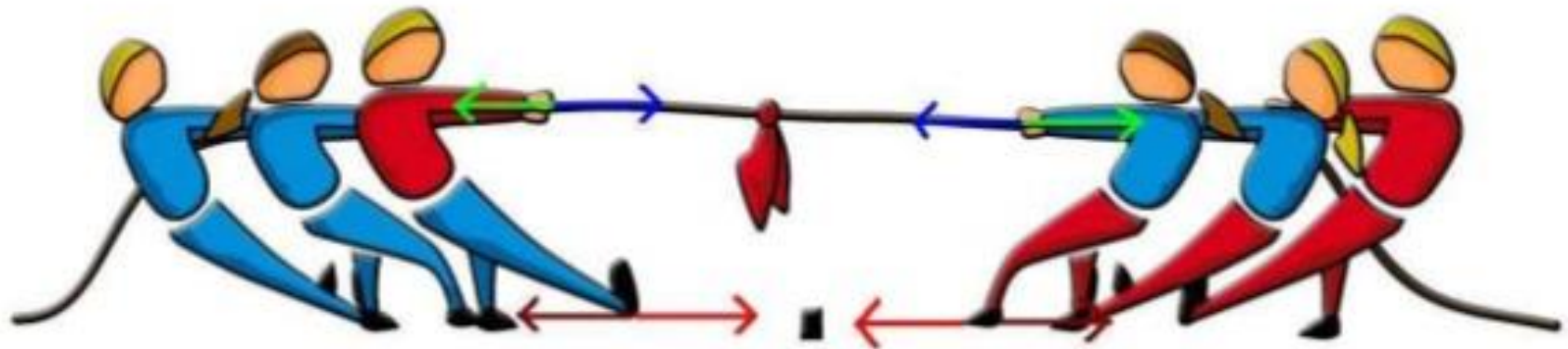
DIFERENTES ABORDAGENS E INTENÇÕES
X

DIFERENTES METODOLOGIAS

ENGENHARIA DE SOFTWARE

EM QUE SÃO DIFERENTES ?

TRADICIONAL



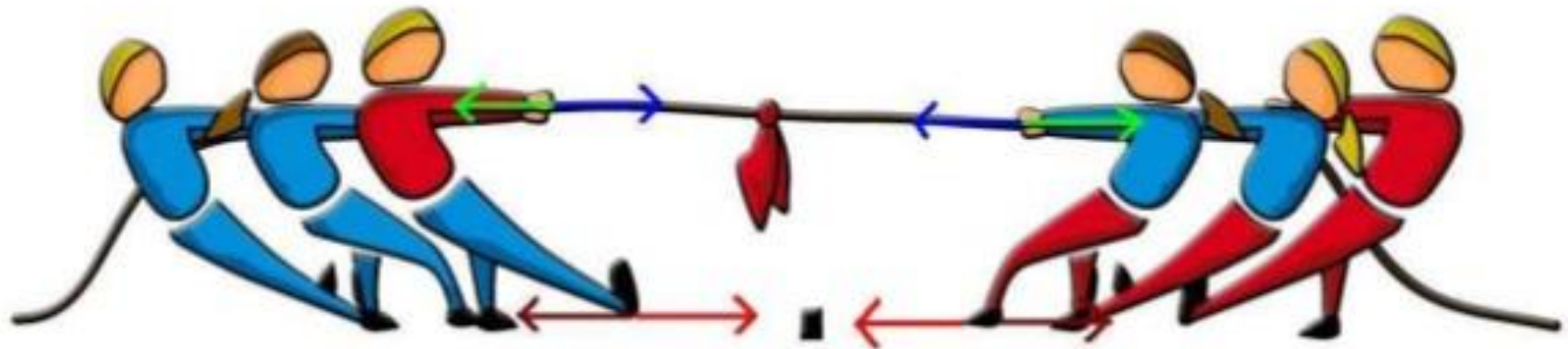
ÁGIL

DIFERENTES ABORDAGENS E INTENÇÕES
X
DIFERENTES METODOLOGIAS

ENGENHARIA DE SOFTWARE

QUAL É MELHOR ?

TRADICIONAL

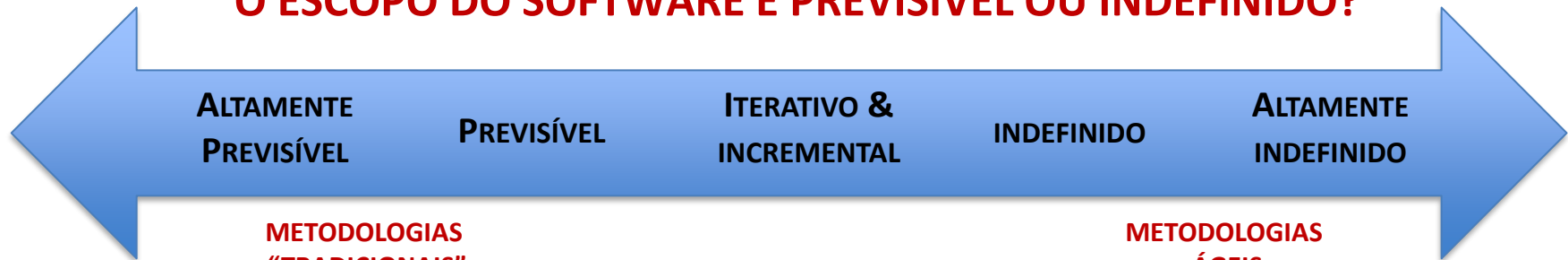


ÁGIL

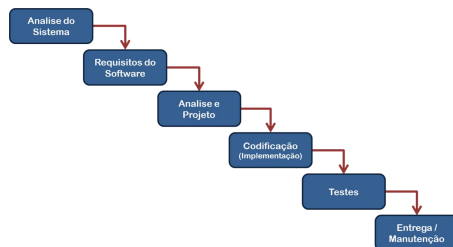
**DIFERENTES ABORDAGENS E INTENÇÕES
X
DIFERENTES METODOLOGIAS**

ENGENHARIA DE SOFTWARE

O ESCOPO DO SOFTWARE É PREVISÍVEL OU INDEFINIDO?



METODOLOGIAS "TRADICIONAIS"

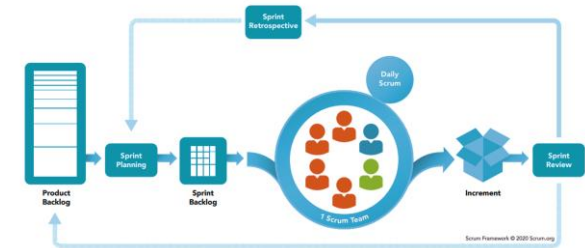


- FLUXO LINEAR
- ETAPAS LONGAS
- CONTROLE RIGOROSO DE REQUISITOS



- FLUXO ITERATIVO
- ETAPAS DE MÉDIA DURAÇÃO
- ALGUMA FLEXIBILIDADE NOS REQUISITOS
- MEIO PREDITIVO / MEIO ADAPTIVO

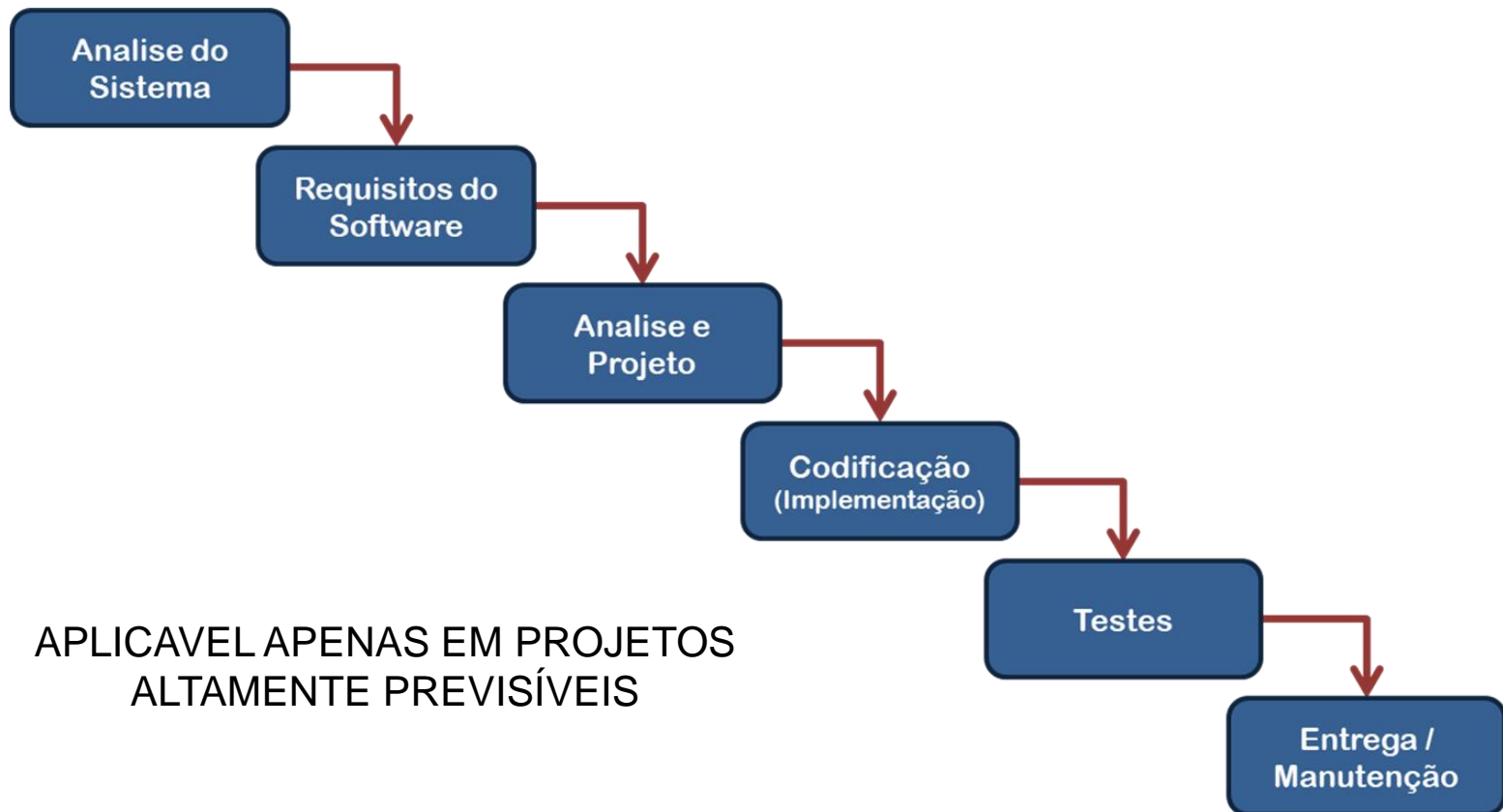
METODOLOGIAS ÁGEIS



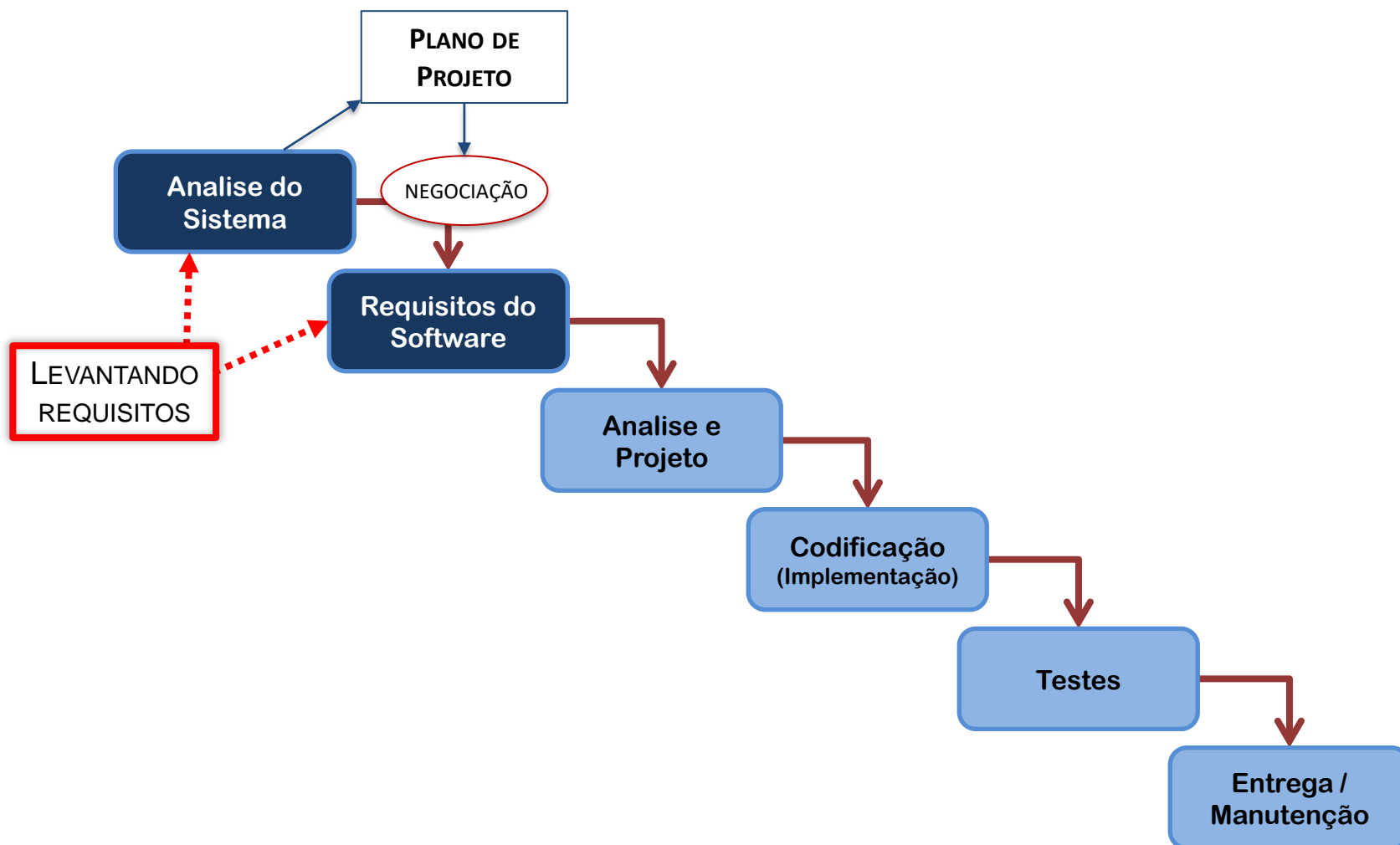
- FLUXO ALTAMENTE ITERATIVO
- ETAPAS CURTAS E FREQUENTES
- REQUISITOS EMERGENTES AO LONGO DO PROJETO

PARADIGMAS “TRADICIONAIS”

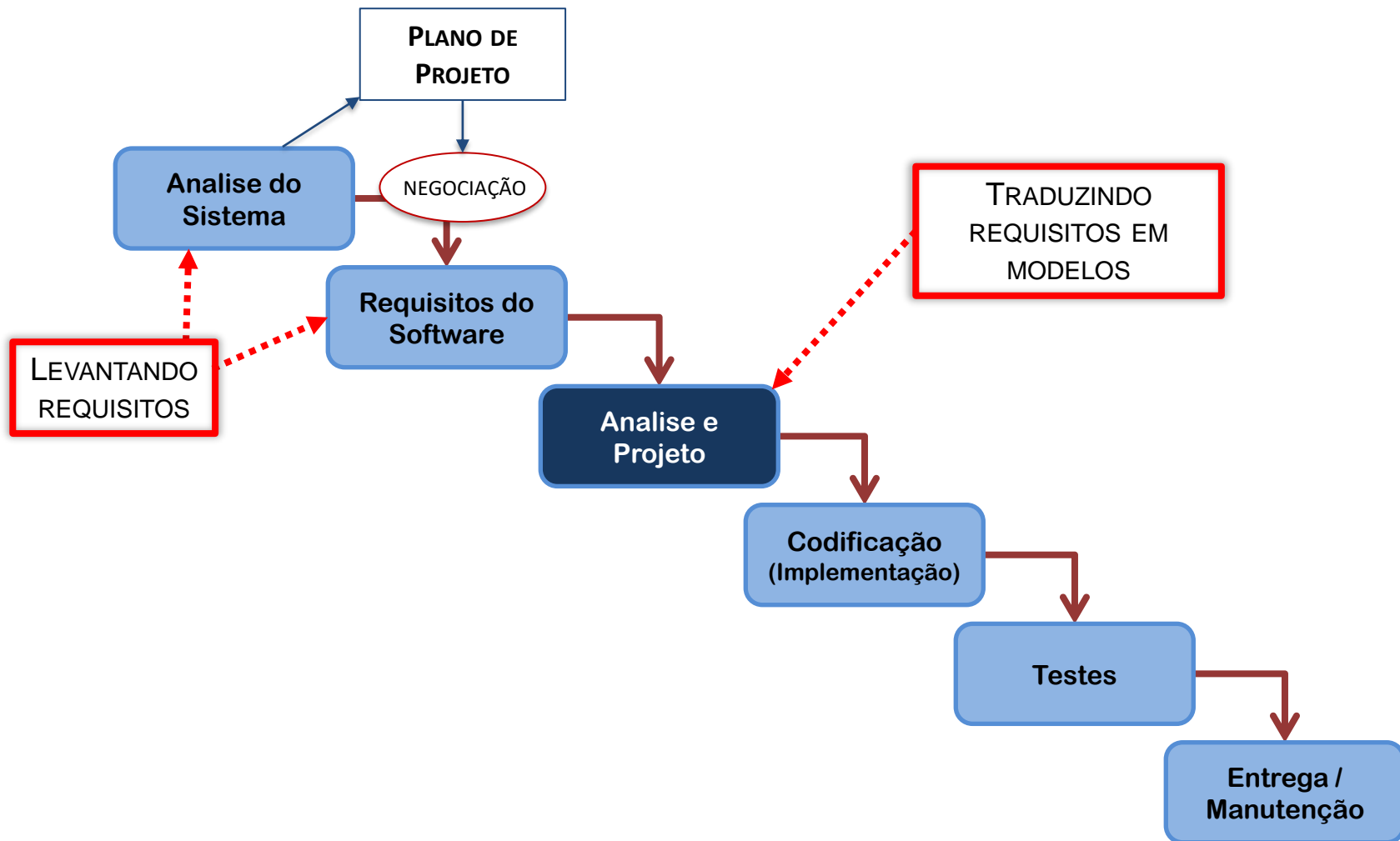
METODOLOGIA CASCATA OU WATERFALL



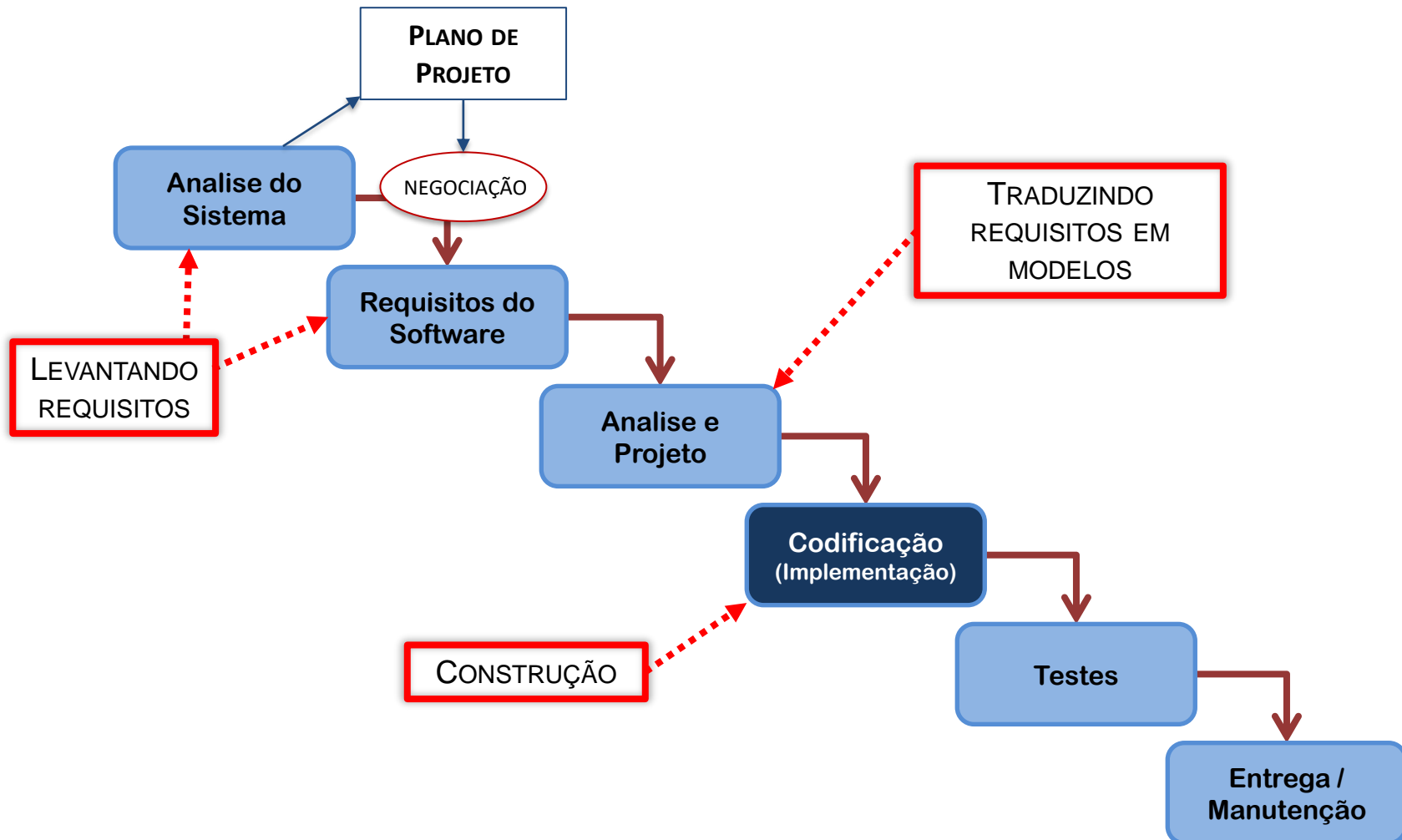
DOS REQUISITOS À CONSTRUÇÃO



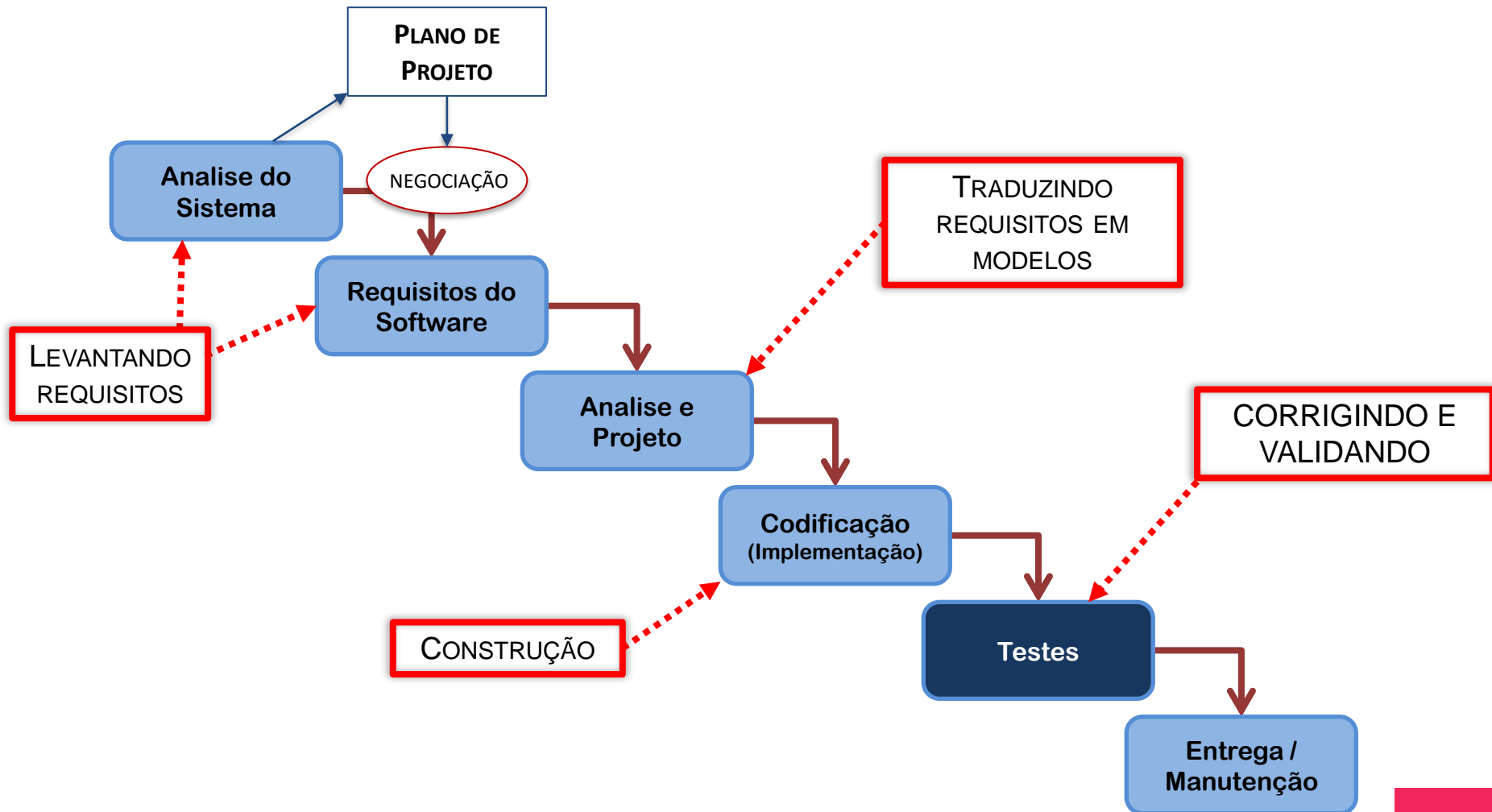
DOS REQUISITOS À CONSTRUÇÃO



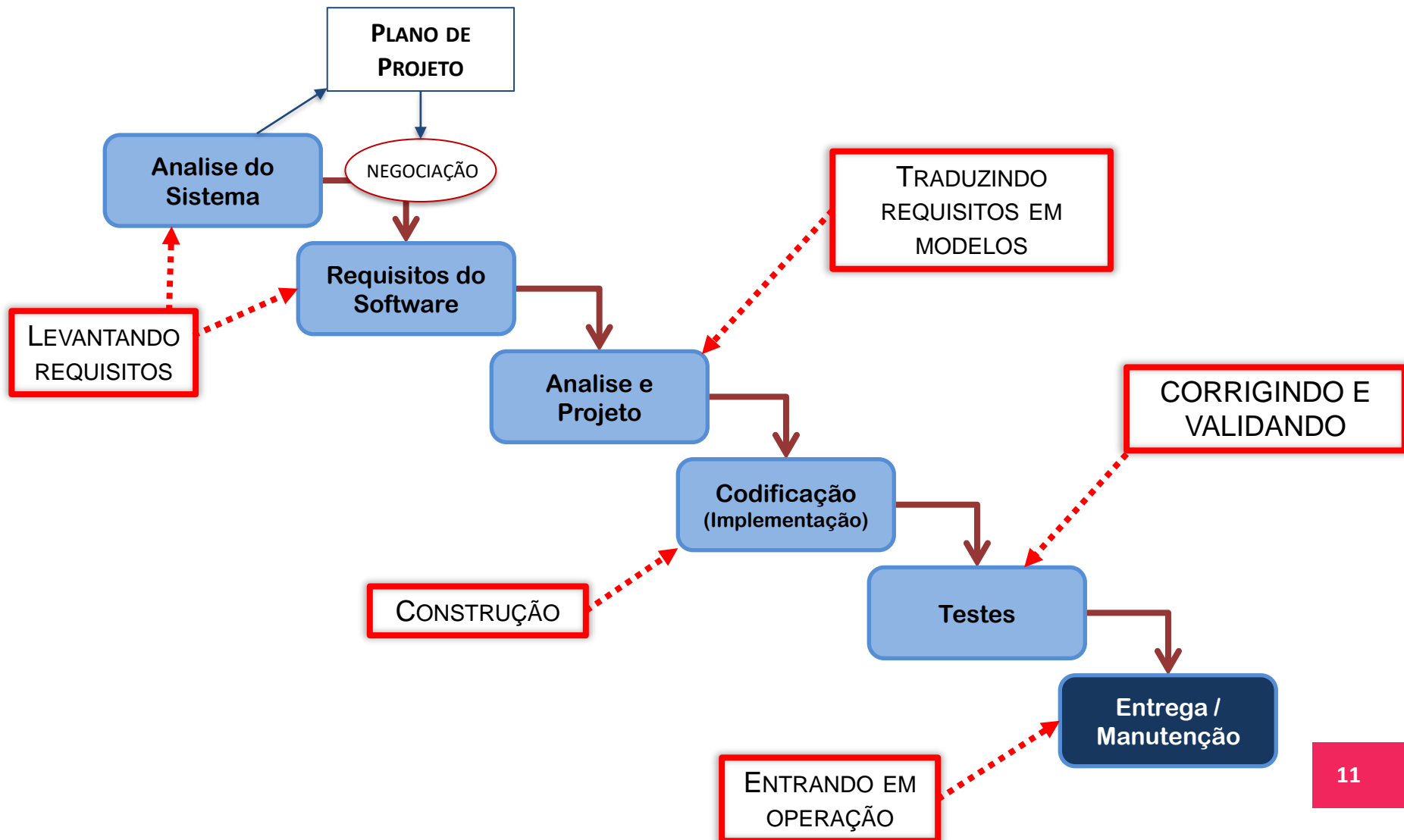
DOS REQUISITOS À CONSTRUÇÃO



DOS REQUISITOS À CONSTRUÇÃO

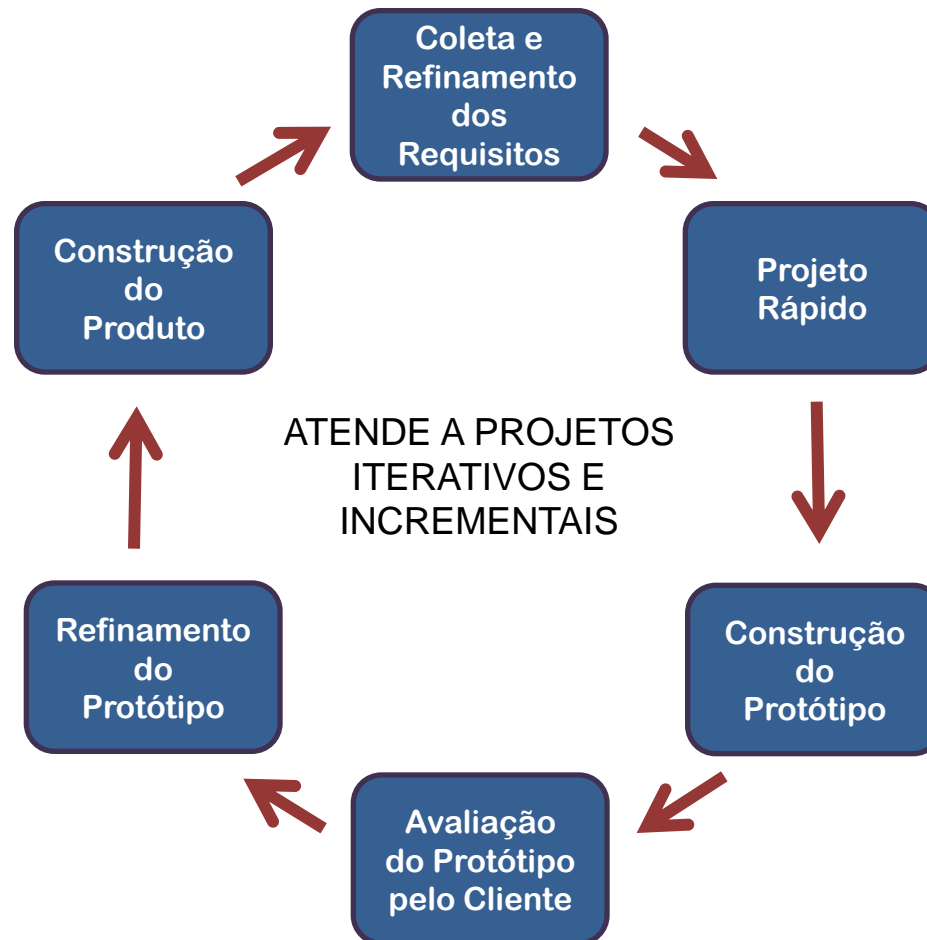


DOS REQUISITOS À CONSTRUÇÃO



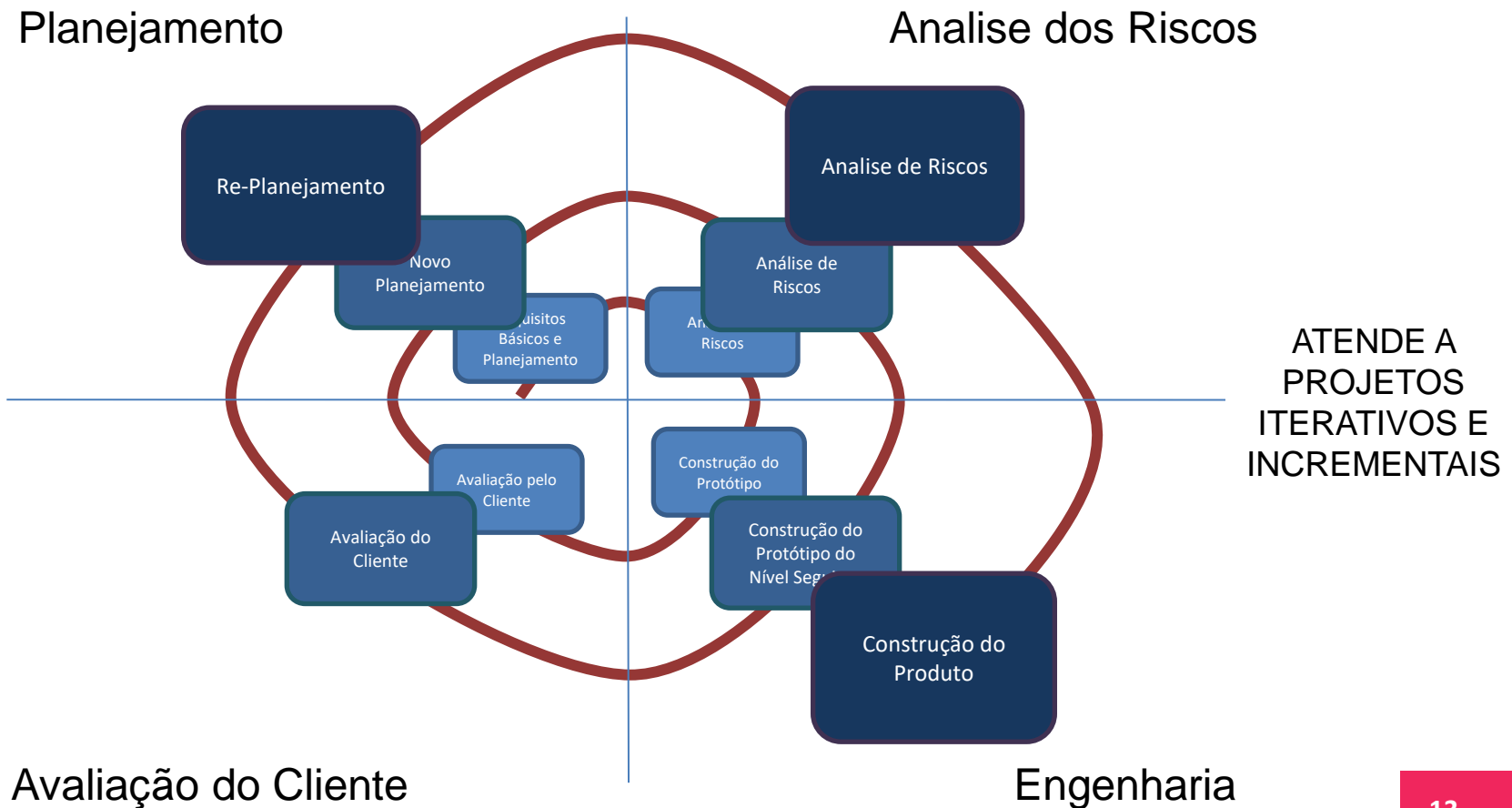
PARADIGMAS “TRADICIONAIS”

METODO DE PROTOTIPAÇÃO



PARADIGMAS “TRADICIONAIS”

METODO ESPIRAL



MANIFESTO ÁGIL

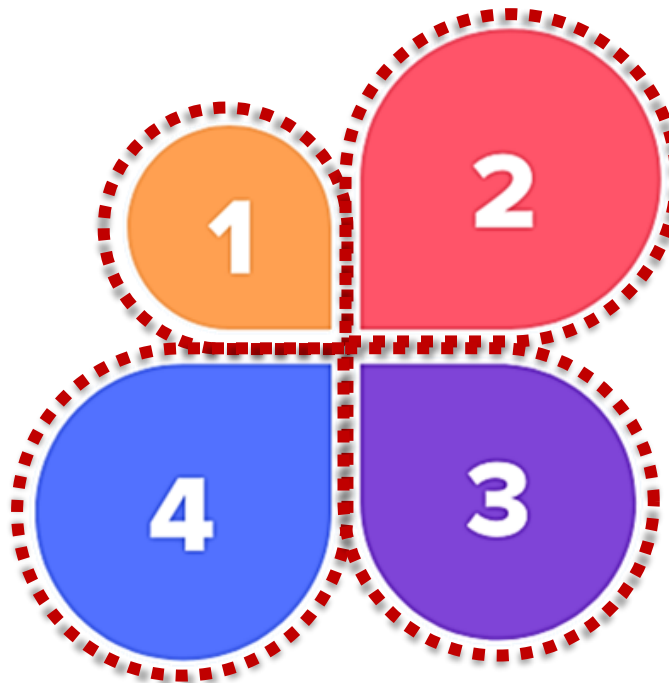
Manifesto Ágil

**Indivíduos
e interações**

mais que
processos e
ferramentas

**Responder
a mudanças**

mais que
seguir um
plano



**Software em
funcionamento**

mais que
documentação
abrangente

**Colaboração
com o cliente**

mais que
negociação
de contratos

MANIFESTO ÁGIL

Os 12 princípios ágeis

-  **SATISFAÇA O CONSUMIDOR**
-  **ACEITE BEM AS MUDANÇAS**
-  **ENTREGAS FREQUENTES**
-  **TRABALHE EM CONJUNTO**
-  **CONFIE E APOIE**
-  **CONVERSAS FACE A FACE**
-  **SOFTWARE FUNCIONANDO**
-  **DESENVOLVIMENTO SUSTENTÁVEL**
-  **ATENÇÃO CONTÍNUA**
-  **MANTENHA A SIMPLICIDADE**
-  **TIMES AUTO-ORGANIZADOS**
-  **REFLETIR E AJUSTAR**

| DIVERSIDADE DE METODOLOGIAS ÁGEIS

KANBAN

LEAN

SCRUM

XP

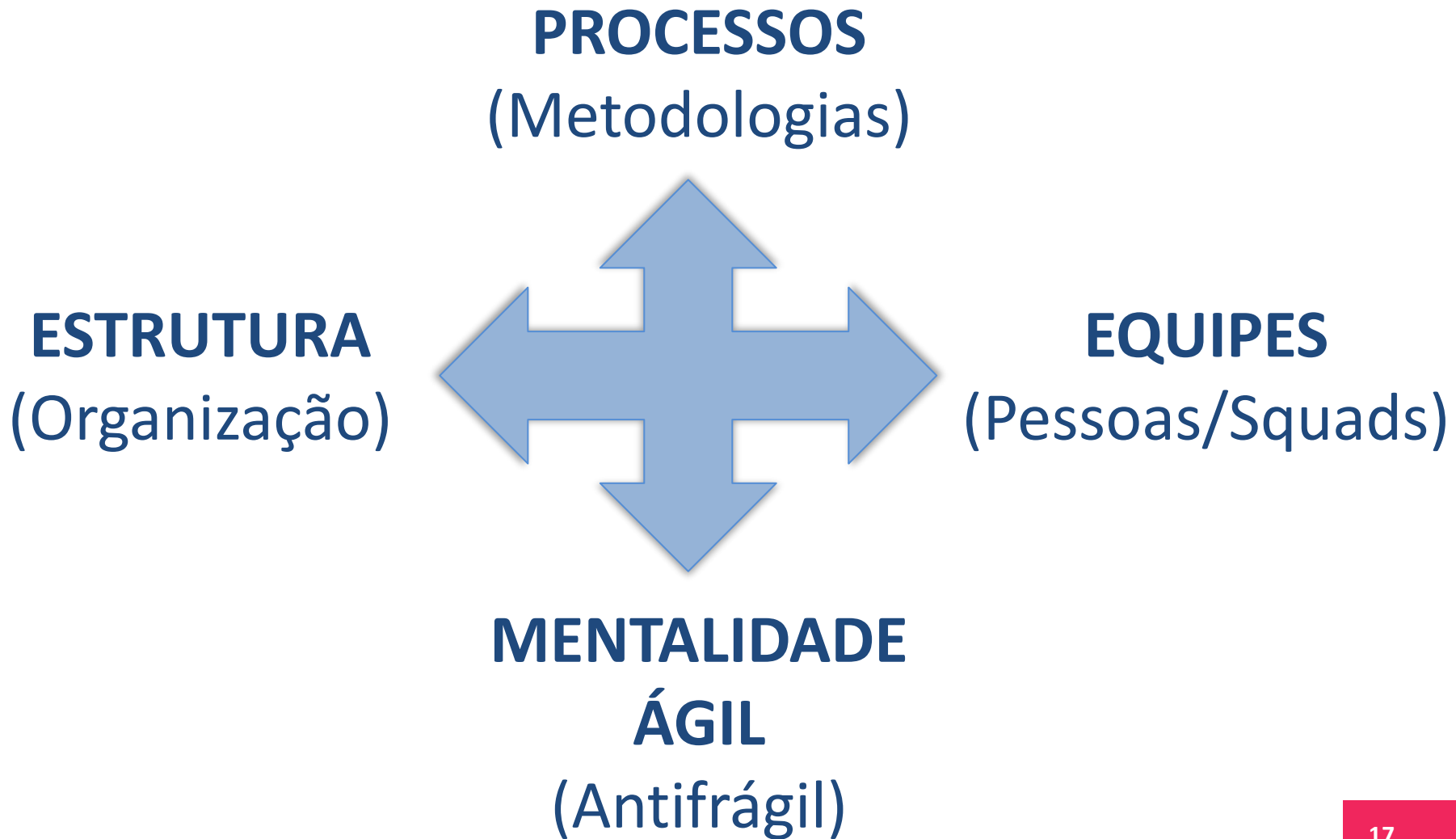
FDD

TDD

DSDM

CRYSTAL

| GESTÃO ÁGIL



Antifrágil : Coisas que se beneficiam com o caos

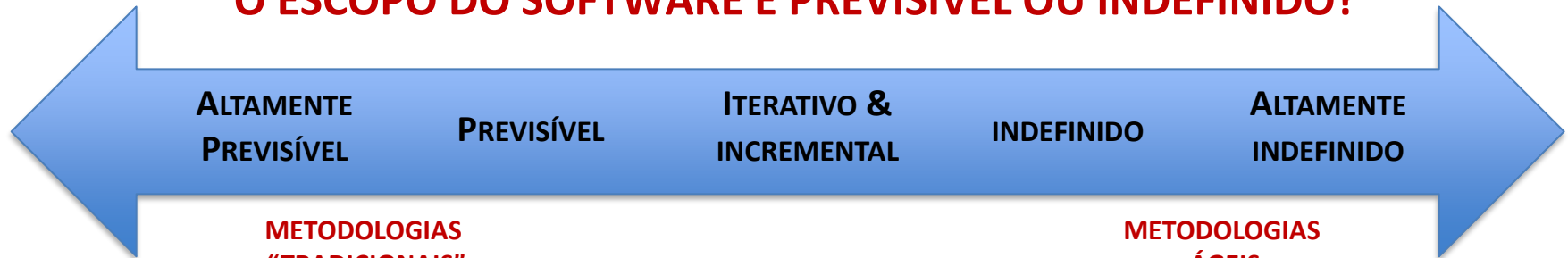
Nassim Nicholas Taleb



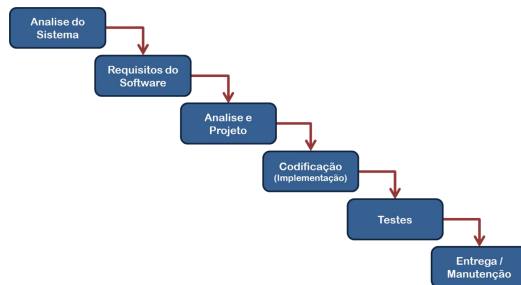
Assim como ficamos fisicamente mais fortes quando submetidos à tensão, muitas coisas se beneficiam do estresse, da desordem e da volatilidade. O que Taleb identificou e chama de antifrágil não só tira proveito do caos, como precisa dele para sobreviver e florescer. O antifrágil está além do resiliente e do robusto. Estes resistem a choques e permanecem os mesmos; o antifrágil, por sua vez, se torna cada vez melhor. Além disso, ele é imune a erros de previsão e está protegido de eventos adversos

ENGENHARIA DE SOFTWARE

O ESCOPO DO SOFTWARE É PREVISÍVEL OU INDEFINIDO?

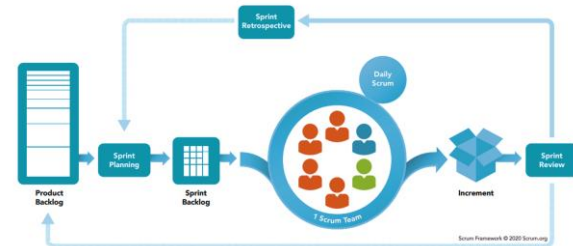


**METODOLOGIAS
“TRADICIONAIS”**



- FLUXO LINEAR
- ETAPAS LONGAS
- CONTROLE RIGOROSO DE REQUISITOS

**METODOLOGIAS
ÁGEIS**



- FLUXO ALTAMENTE ITERATIVO
- ETAPAS CURTAS E FREQUENTES
- REQUISITOS EMERGENTES AO LONGO DO PROJETO

DIFERENÇAS ENTRE TRADICIONAIS X ÁGEIS

TRADICIONAL



Maior comunicação com o cliente no início e final do processo



Geralmente possuem etapas de implementação mais longas



Os requisitos são definidos antecipadamente



Escopo definido no início do projeto



Interação entre as equipes durante etapas específicas



Os testes acontecem no final

Comunicação contínua com o cliente durante todas as etapas



Etapas de implementação iterativas e mais curtas



Os requisitos evoluem ao longo do projeto



Escopo definido ao longo do projeto com possibilidades de ajustes contínuos



Maior interação entre as equipes durante todas as etapas

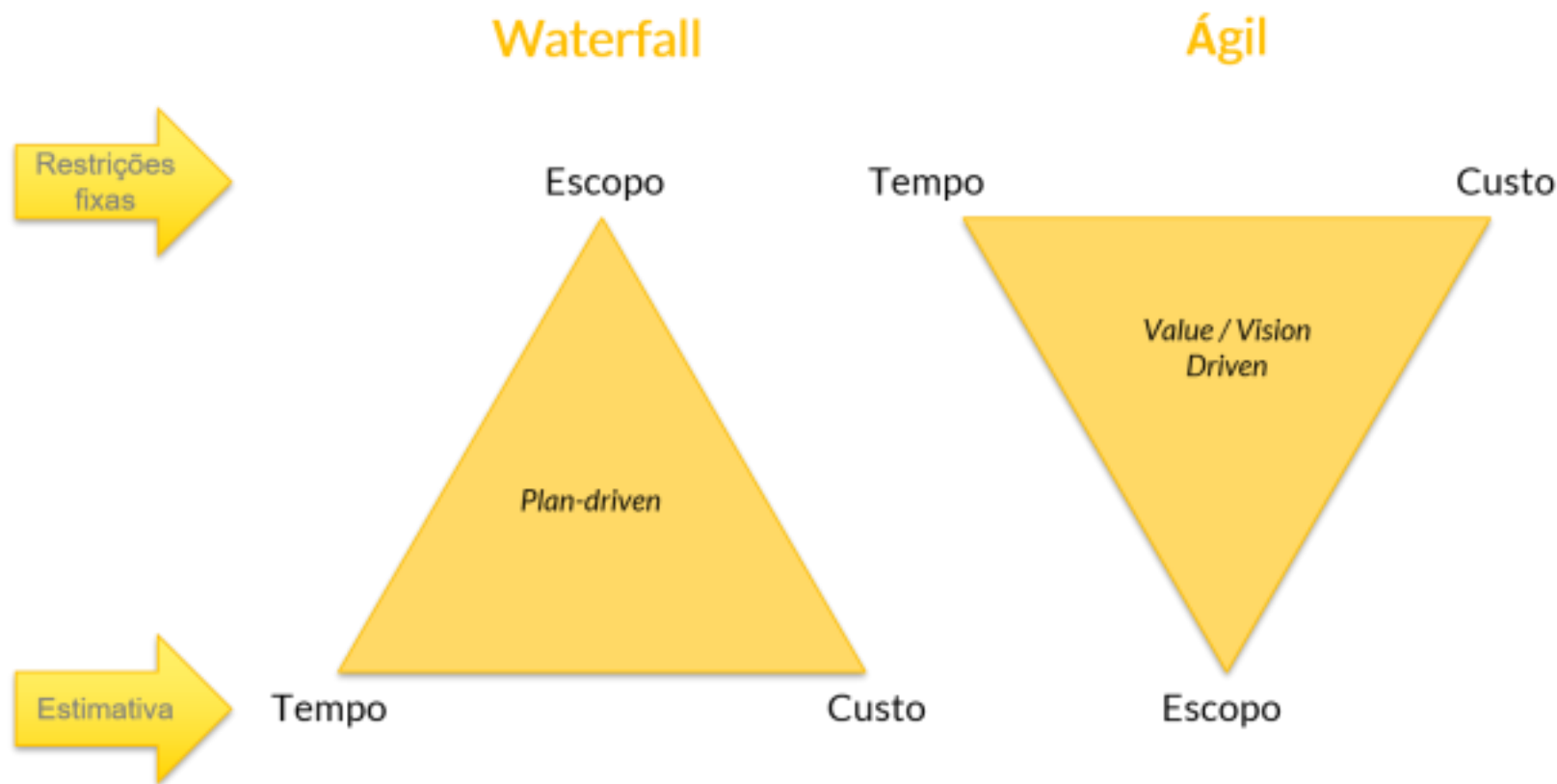


Os testes ocorrem a cada iteração*

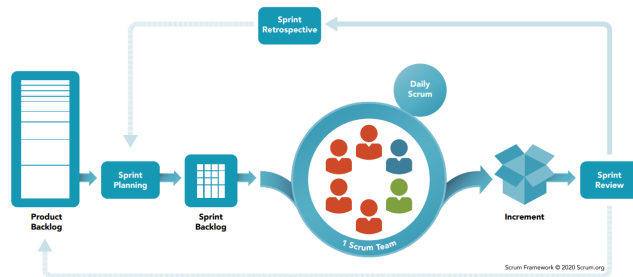


ÁGIL

DIFERENÇAS ENTRE TRADICIONAIS X ÁGEIS



SCRUM COMO FRAMEWORK



METODOLOGIA OU FRAMEWORK ?

NÃO É UM PROCESSO OU UMA TÉCNICA PARA O DESENVOLVIMENTO DE PRODUTOS.

É UM **FRAMEWORK** DENTRO DO QUAL SE PODE EMPREGAR DIVERSOS
PROCESSOS E TÉCNICAS

FUNDAMENTADO NA **TEORIA DE CONTROLE DE PROCESSOS EMPÍRICOS**

ABORDAGEM ITERATIVA E INCREMENTAL PARA OTIMIZAR A PREVISIBILIDADE E
CONTROLAR RISCOS

SCRUM – OS PILARES

TRANSPARÊNCIA

COMPENSANDO A AUSÊNCIA DE
“DOCUMENTAÇÃO”

TODOS OS ASPECTOS DO PROCESSO QUE AFETAM O RESULTADO
DEVEM SER VISÍVEIS PARA AQUELES QUE GERENCIAM OS RESULTADOS.

INSPEÇÃO

GARANTINDO QUALIDADE
E DIRECIONAMENTO

OS DIVERSOS ASPECTOS DO PROCESSO DEVEM SER INSPECIONADOS COM UMA FREQUÊNCIA SUFICIENTE PARA
QUE VARIAÇÕES INACEITÁVEIS NO PROCESSO POSSAM SER DETECTADAS E CORRIGIDAS IMEDIATAMENTE.

ADAPTAÇÃO

FLEXIBILIDADE PARA REAGIR A
MUDANÇAS

SE O INSPETOR DETERMINAR, A PARTIR DA INSPEÇÃO, QUE UM OU MAIS ASPECTOS DO PROCESSO ESTÃO FORA
DOS LIMITES ACEITÁVEIS E QUE O PRODUTO RESULTANTE SERÁ INACEITÁVEL, ELE DEVERÁ AJUSTAR O PROCESSO
OU O MATERIAL SENDO PROCESSADO. ESSE AJUSTE DEVE SER FEITO O MAIS RÁPIDO POSSÍVEL PARA MINIMIZAR
DESVIOS POSTERIORES.

SCRUM E SEUS VALORES



CORAGEM

Membros do Scrum Team tem coragem de fazer o que é certo e enfrentar problemas difíceis

FOCO

Todos focam no trabalho do Sprint e nos objetivos do Scrum Team

COMPROMETIMENTO

As pessoas se comprometem em atingir os objetivos do Scrum Team

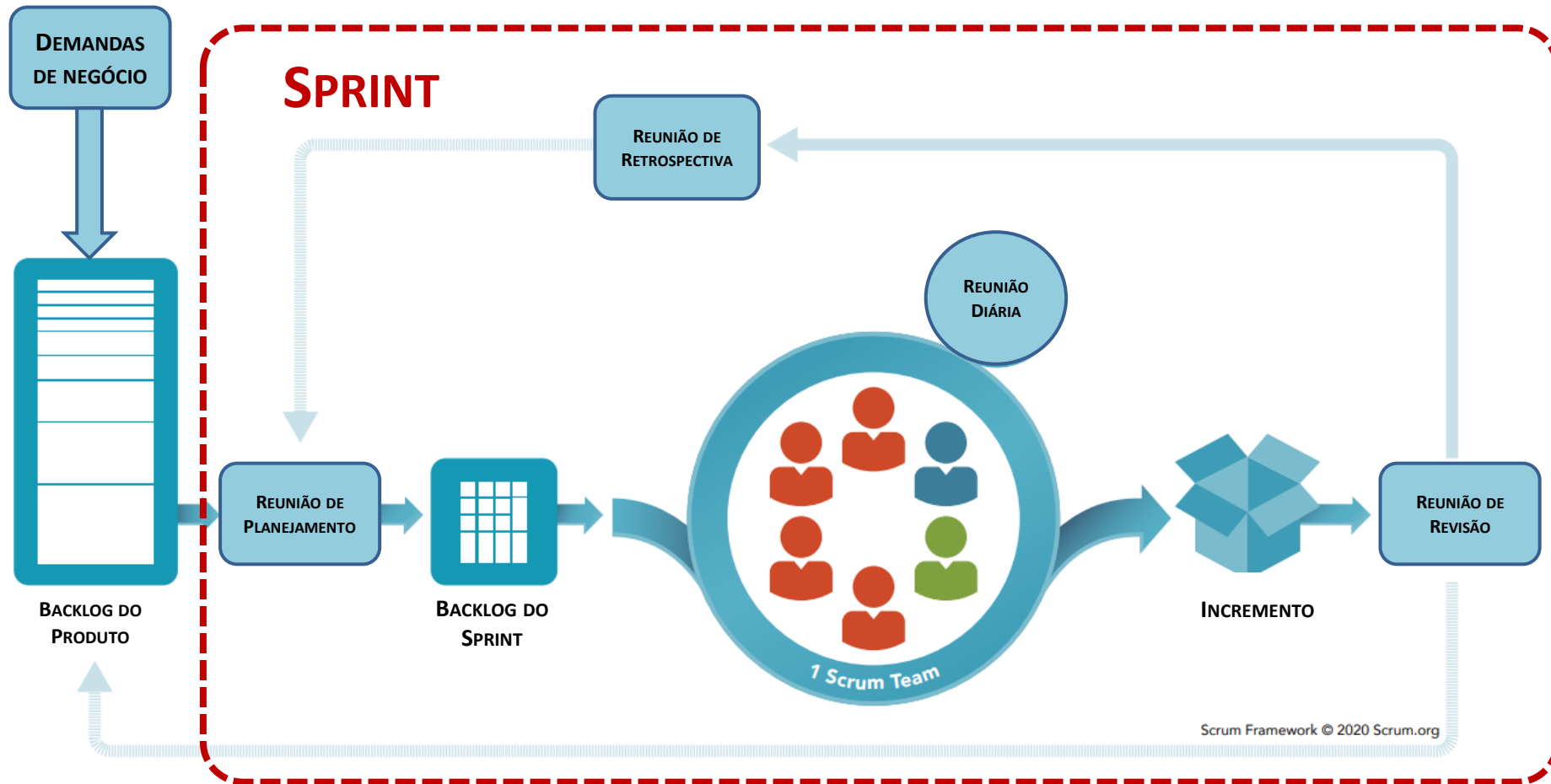
RESPEITO

Os Membros do Scrum Team respeitam a competência e a independência dos demais

ABERTURA

O Scrum Team e seus Stakeholders concordam em ser francos e abertos sobre o trabalho e os desafios de sua execução

SCRUM E SUAS CERIMÔNIAS



METODOLOGIA OU FRAMEWORK DE GESTÃO ?

CERIMÔNIAS DO SCRUM

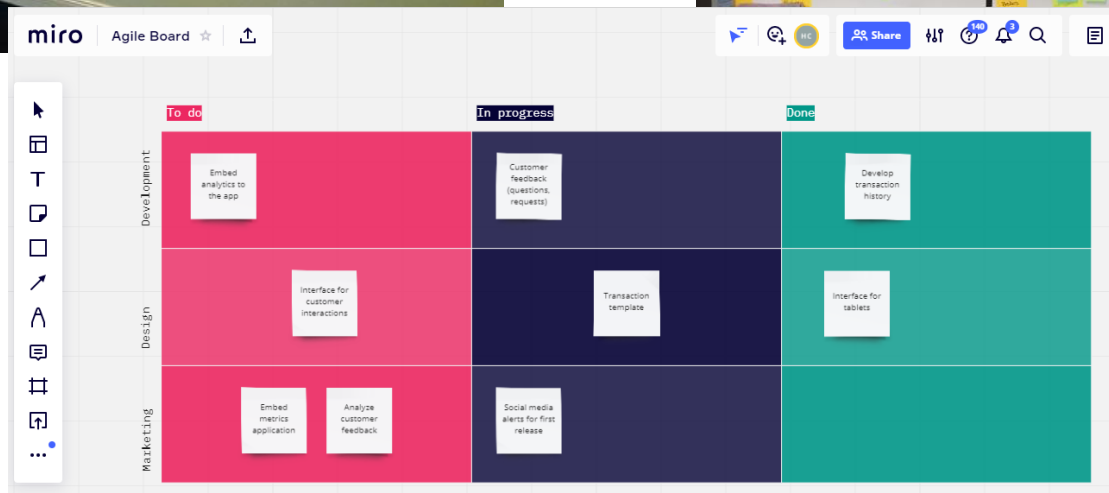
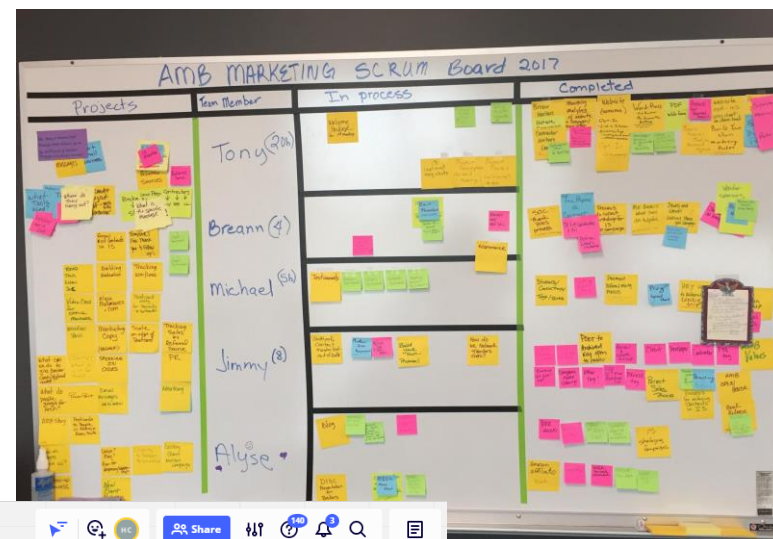
SCRUM BOARD



PRINCIPAL MECANISMO DE CONTROLE

CERIMÔNIAS DO SCRUM

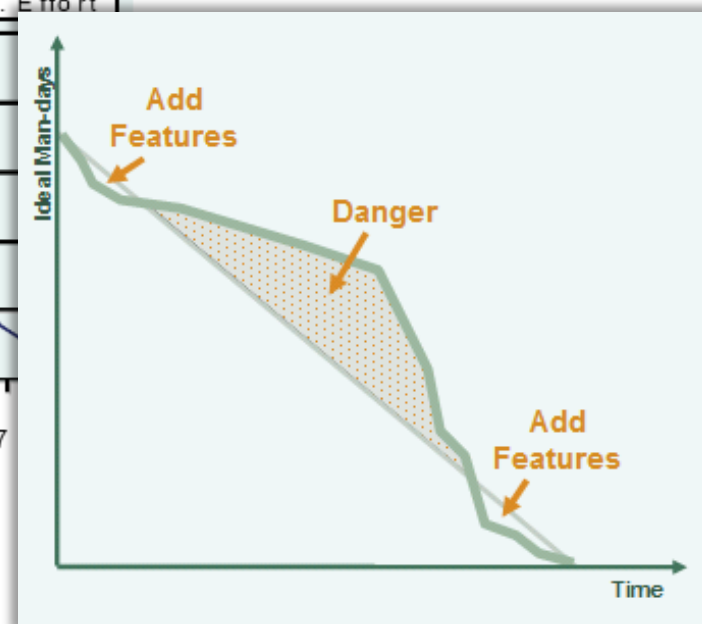
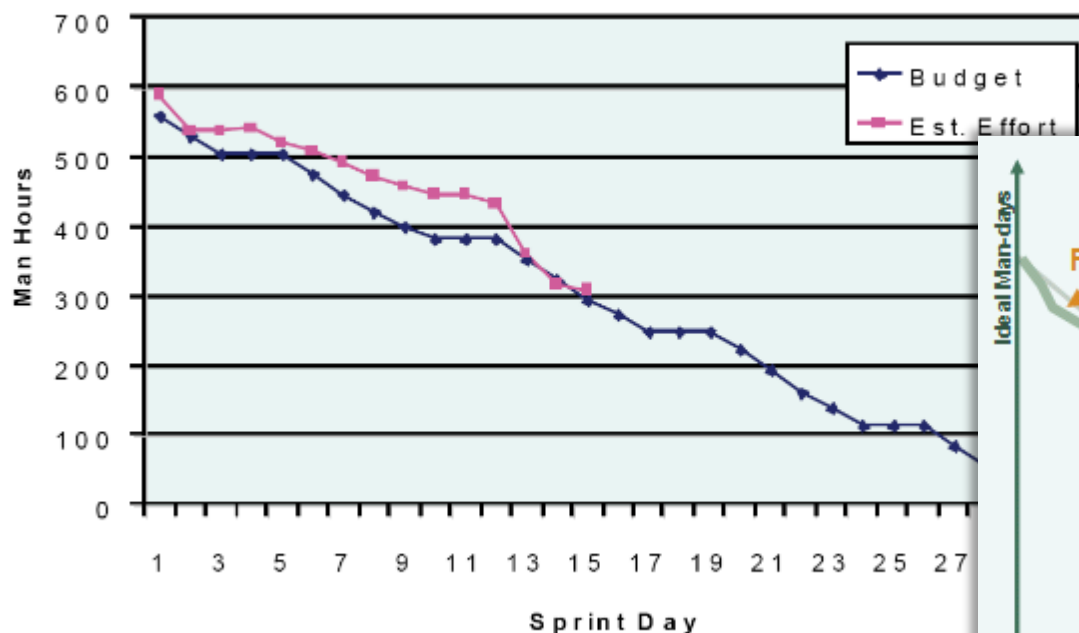
SCRUM BOARD



CERIMÔNIAS DO SCRUM

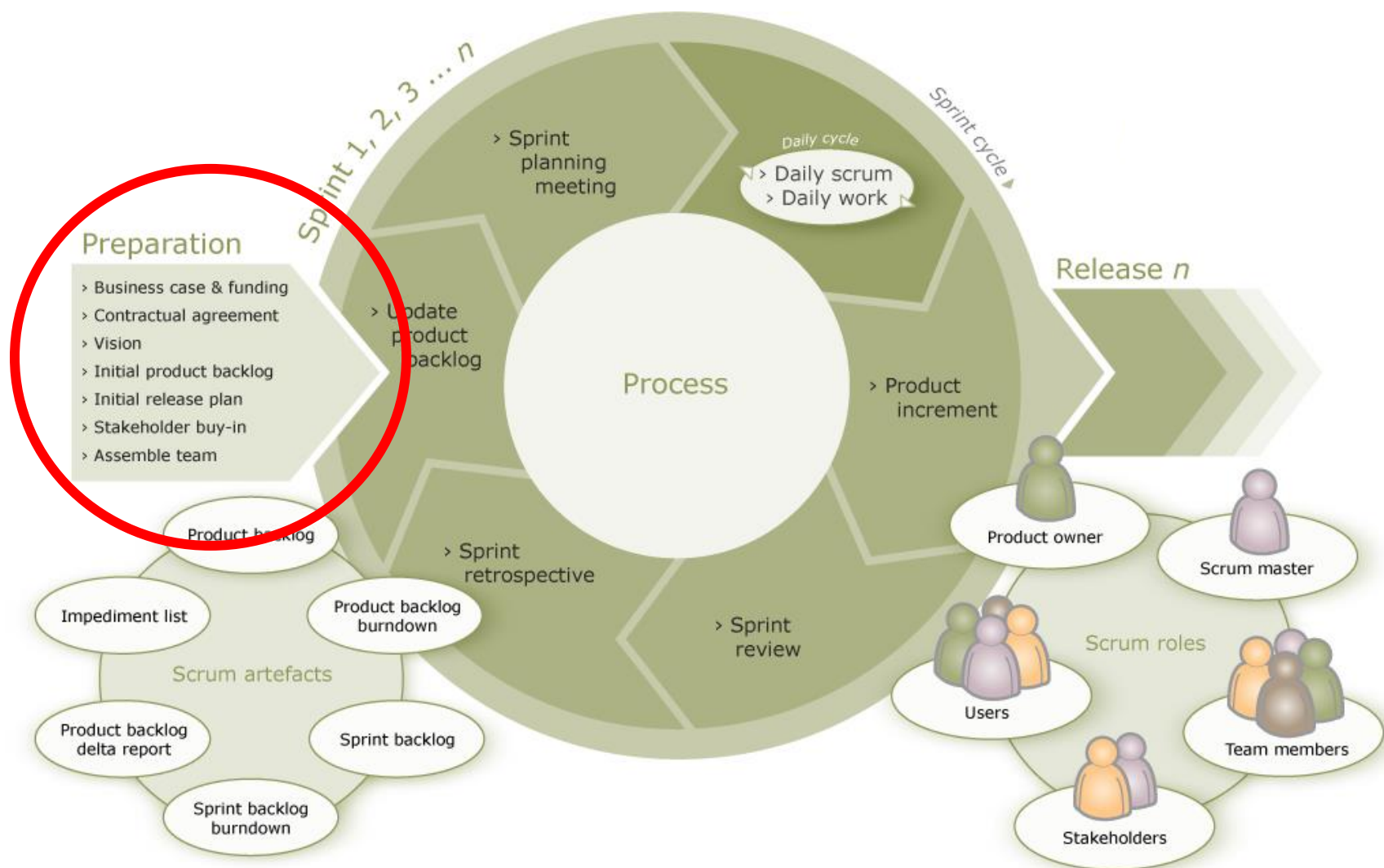
SCRUM BURNDOWN CHART

QUADRO DE “QUEIMA” DE ESFORÇOS



ACOMPANHAMENTO DIÁRIO DO ESFORÇO/PRODUÇÃO

SCRUM – O PROCESSO EXPANDIDO



SCRUM TEAM

- EQUIPE PEQUENA (6/7 MEMBROS)
- AUTOGERENCIADA
- CO-LOCALIZADA

- DESENVOLVEDORES + CLIENTE
- FLEXIBILIDADE DE PAPÉIS
- ALTA COLABORAÇÃO



PRODUCT OWNER



SCRUM MASTER



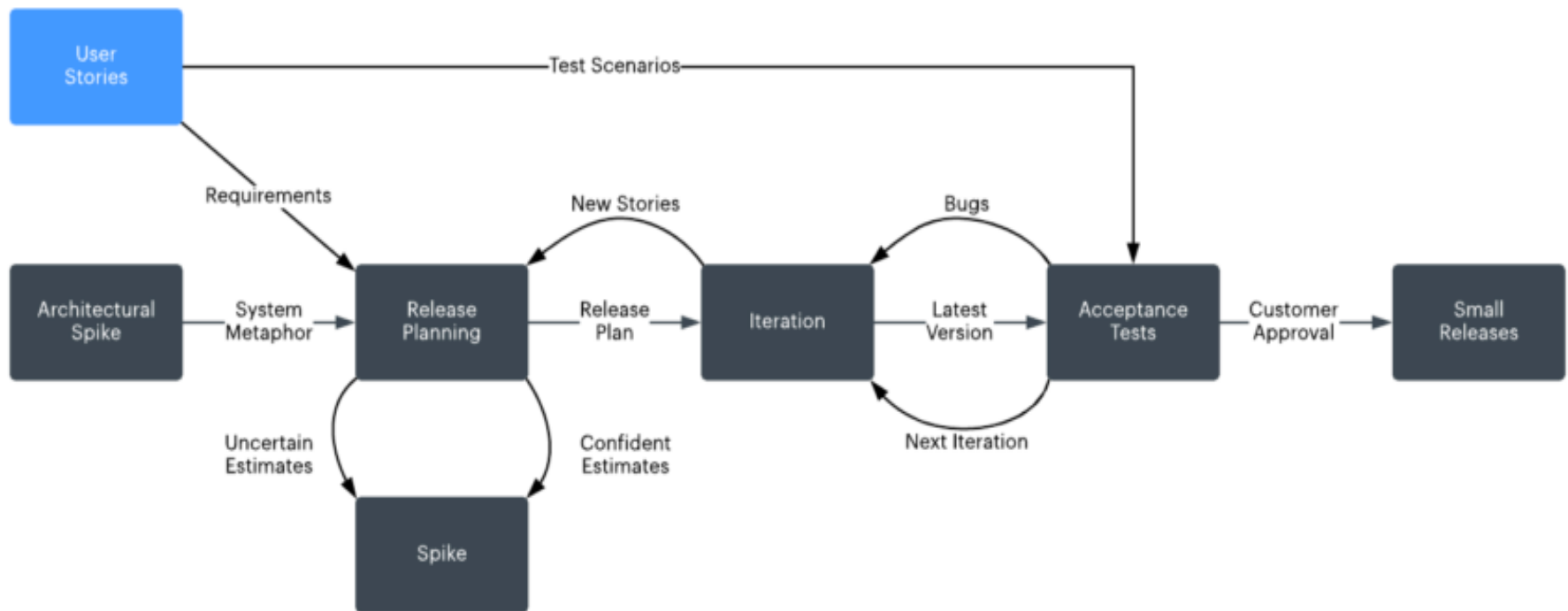
DEVELOPER TEAM

DECIDE O QUE
FAZER E PORQUÊ
(RESPONSÁVEL PELO BACKLOG
DO PRODUTO)

FOCA EM COMO
TRABALHAR
MELHOR
(FACILITADOR E REMOVEDOR
DE EMPECILHOS)

REALIZA O TRABALHO
DE
DESENVOLVIMENTO
(COLABORAÇÃO E FLEXIBILIDADE)

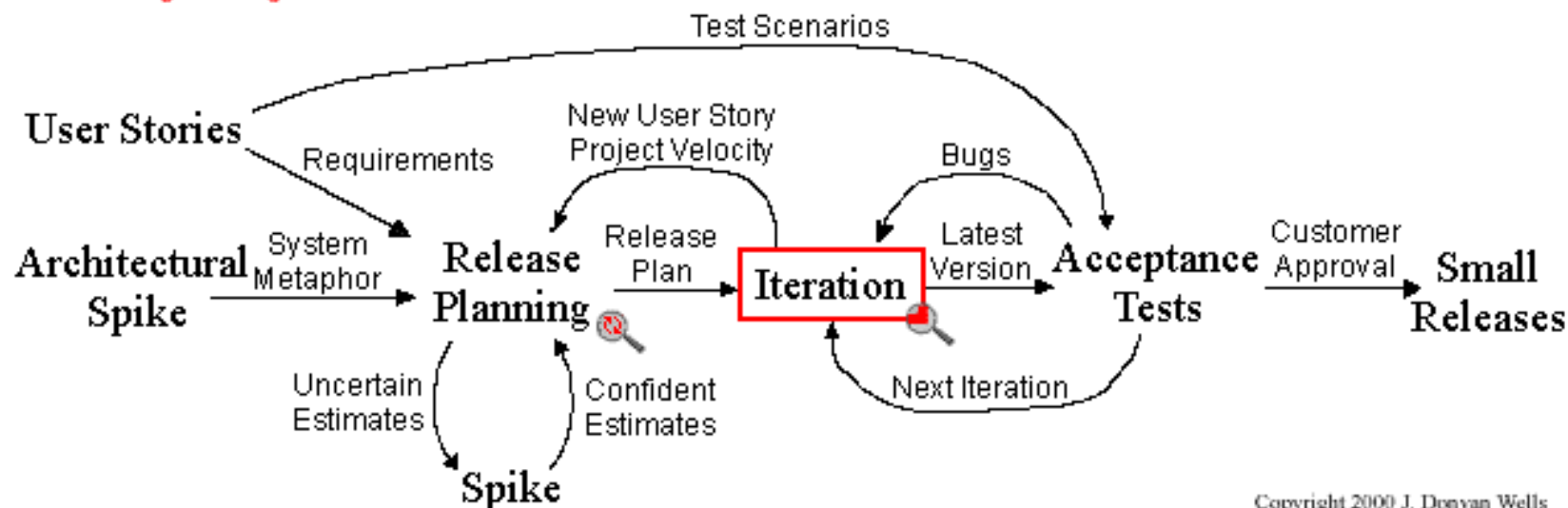
XP – EXTREME PROGRAMMING



XP – EXTREME PROGRAMMING



Extreme Programming Project

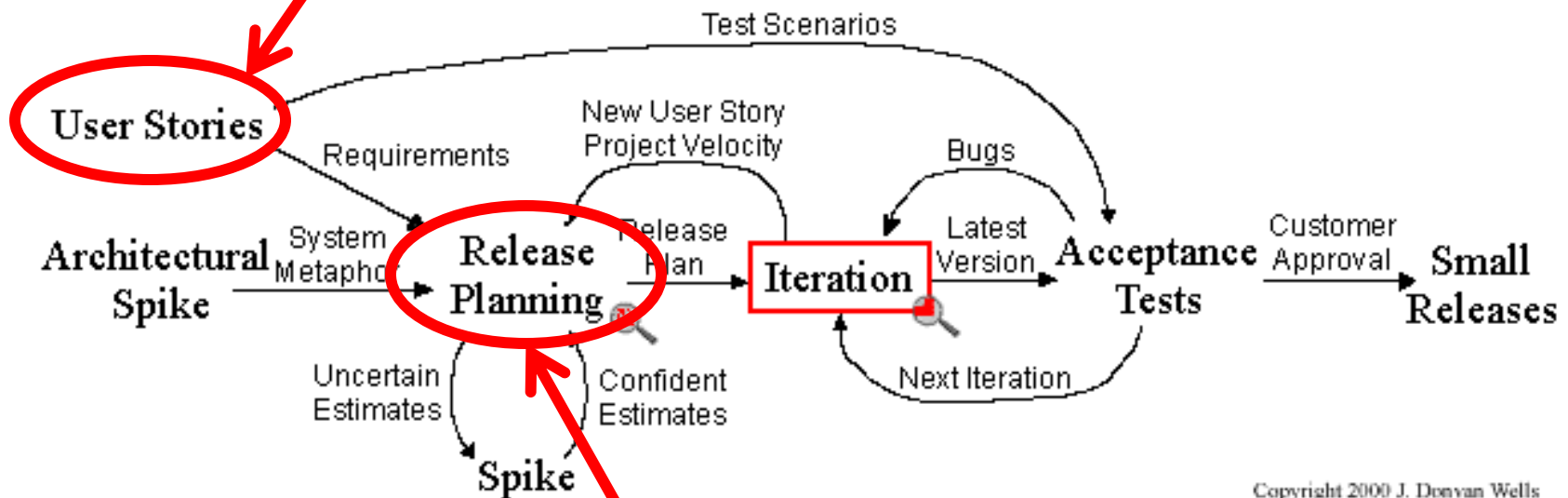


Copyright 2000 J. Donovan Wells

TÉCNICAS E METODOLOGIA PARA DESENVOLVIMENTO ÁGIL
(POSSUI MUITA SIMILARIDADE DE CONCEITOS COM SCRUM)

XP – EXTREME PROGRAMMING

ASSIM COMO NO SCRUM OS REQUISITOS SÃO DOCUMENTADOS DE FORMA SIMPLES, COMO “ESTÓRIAS DO USUÁRIO”.

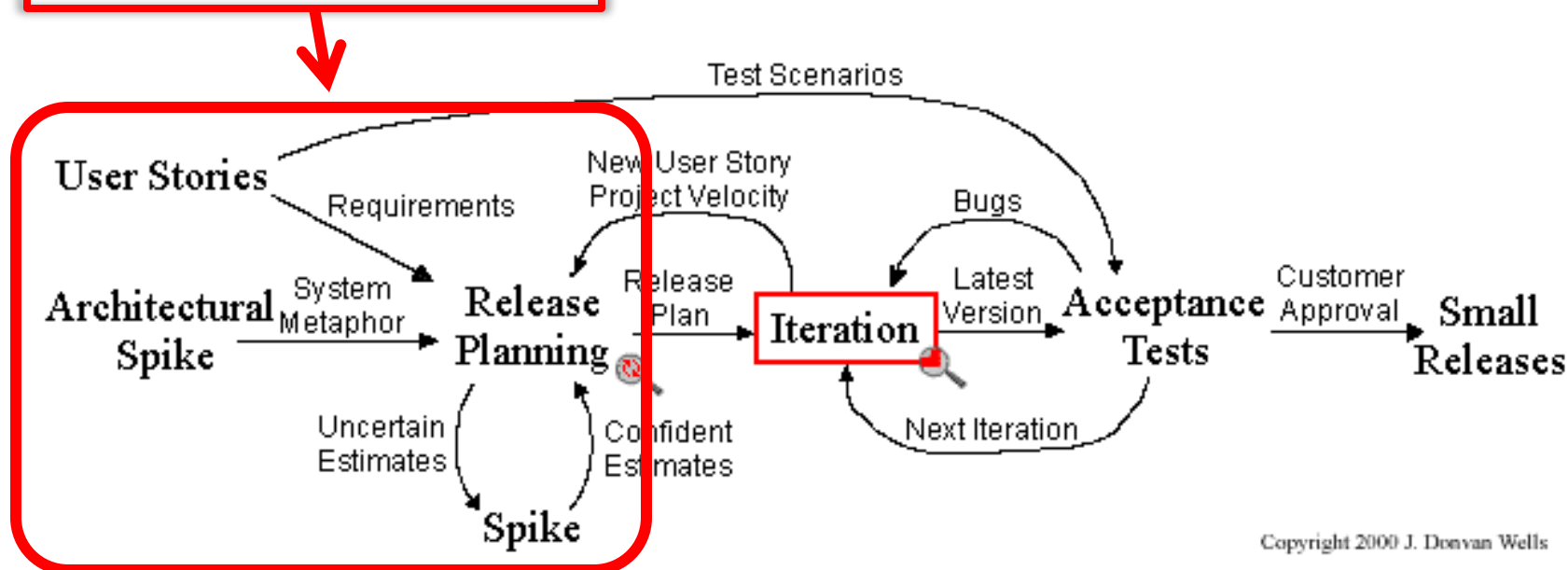


Copyright 2000 J. Donovan Wells

A LISTA DE “ESTÓRIAS DO USUÁRIO” É O PONTO DE PARTIDA PARA O PLANEJAMENTO DA VERSÃO (OU RELEASE)

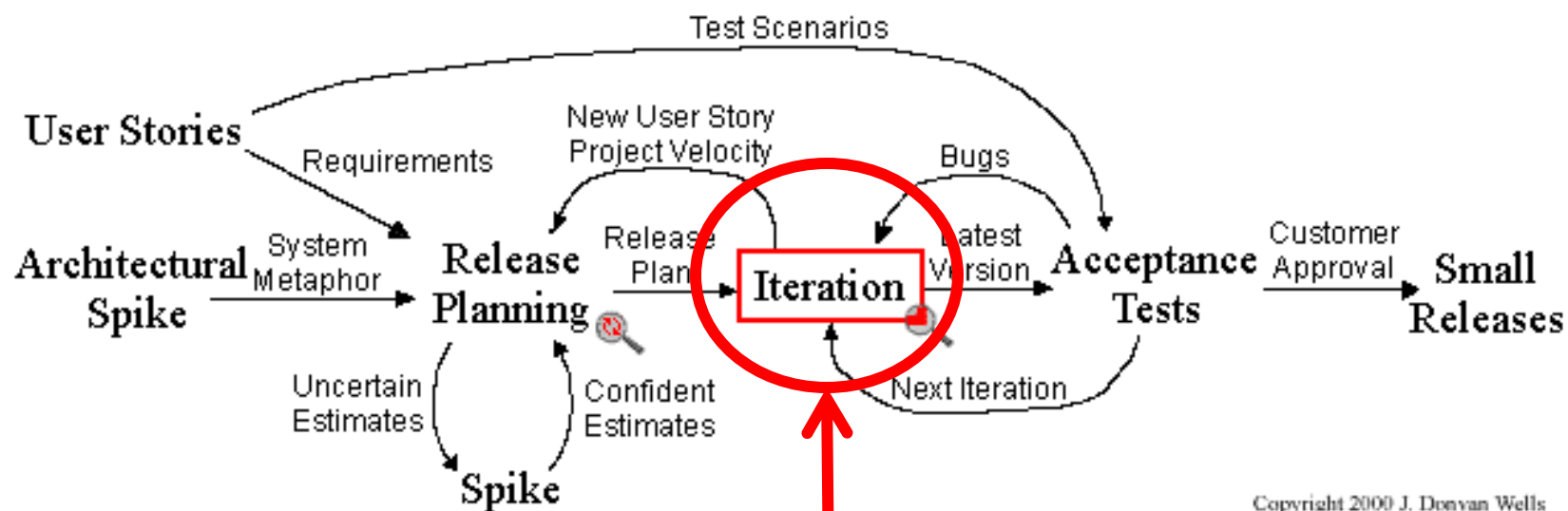
XP – EXTREME PROGRAMMING

EQUIVALENTE À CRIAÇÃO DO
BACKLOG DO PRODUTO
NO **SCRUM**



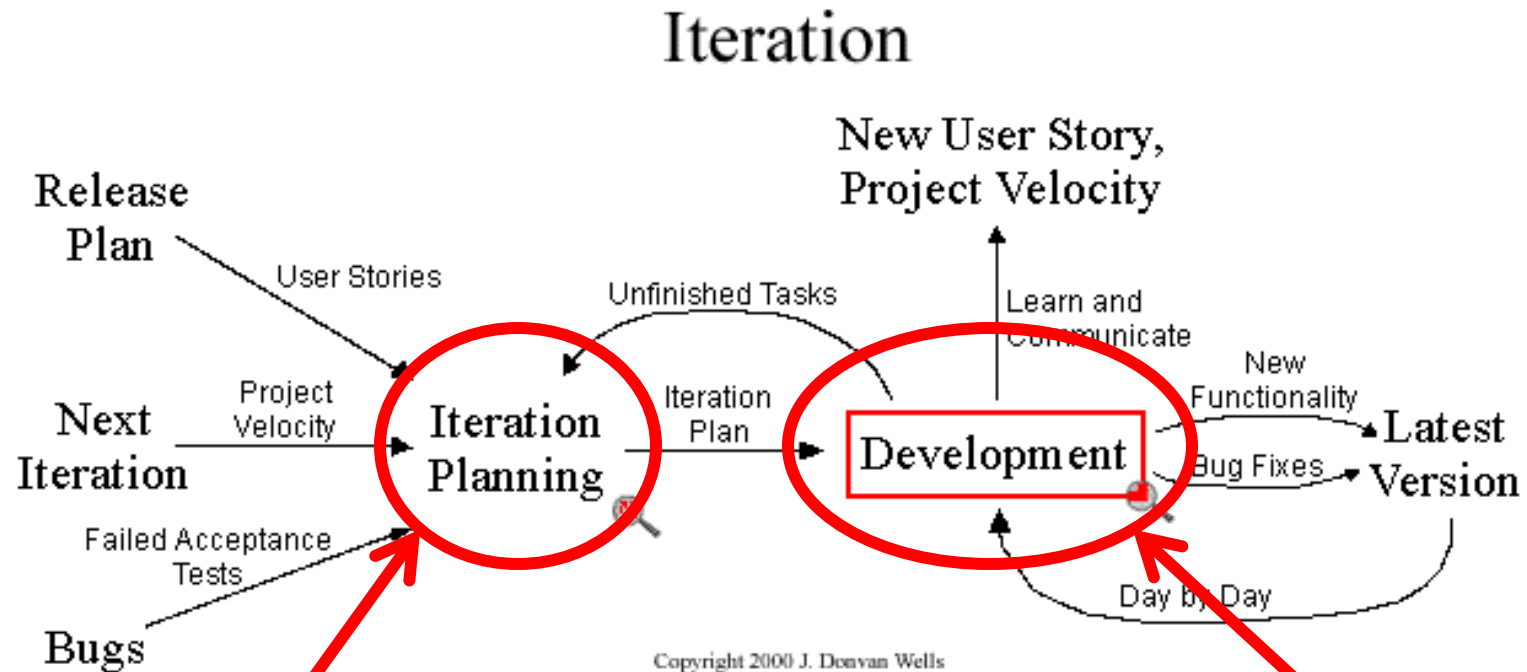
Copyright 2000 J. Donovan Wells

XP – EXTREME PROGRAMMING



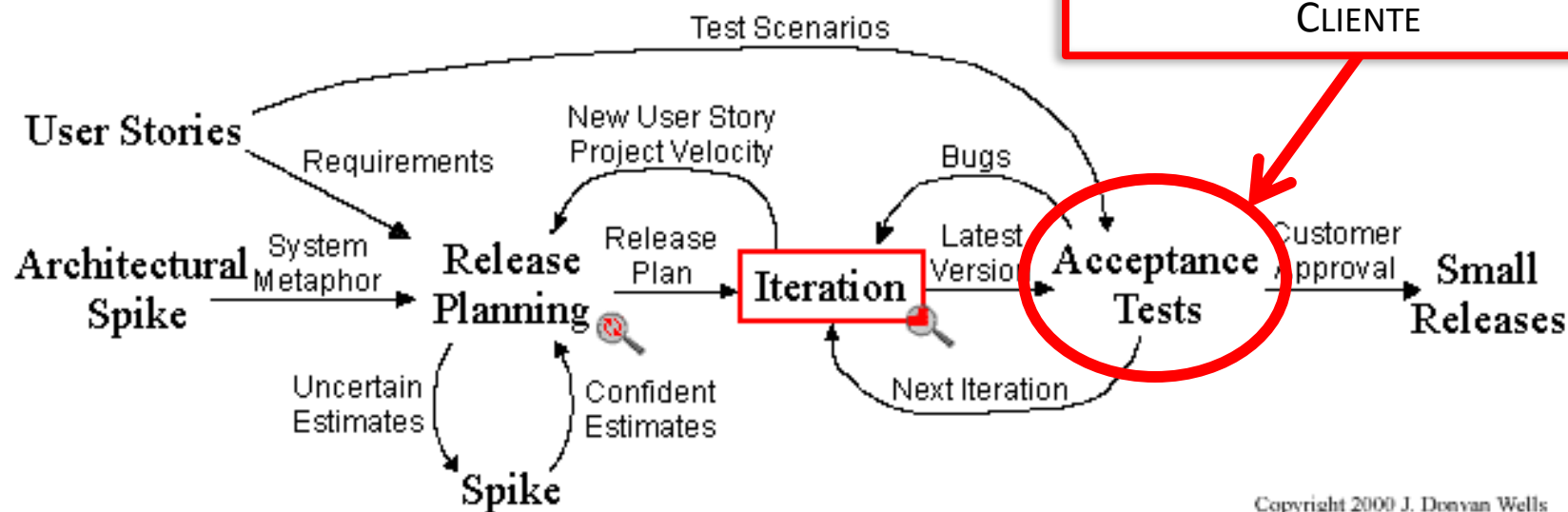
AS **ITERAÇÕES** NA **XP** EQUIVALEM
AOS
SPRINTS NO **SCRUM**

XP – EXTREME PROGRAMMING



XP – EXTREME PROGRAMMING

UMA DAS GRANDES DIFERENÇAS DO CICLO DA **XP** ESTÁ NA IMPORTÂNCIA DADA AOS TESTES DE ACEITAÇÃO POR PARTE DO CLIENTE



Copyright 2000 J. Donovan Wells

VALORES DO XP - EXTREME PROGRAMMING

SIMPLICIDADE

“FAREMOS O QUE FOR PEDIDO, E NADA A MAIS”

COMUNICAÇÃO (FACE-A-FACE)

“TRABALHAREMOS JUNTOS EM TUDO, DOS REQUISITOS AO CÓDIGO”

FEEDBACK

“APRESENTAREMOS O SW O MAIS BREVE POSSÍVEL PARA AVALIAÇÃO E FAREMOS QUAISQUER MUDANÇAS NECESSÁRIAS”

RESPEITO

“DESENVOLVEDORES RESPEITAM O EXPERTISE DOS CLIENTES E VICE-VERSA”

CORAGEM

“DIREMOS SEMPRE A VERDADE SOBRE O PROGRESSO E AS ESTIMATIVAS”

PRÁTICAS DO XP - EXTREME PROGRAMMING



PRÁTICAS DO XP - EXTREME PROGRAMMING

JOGO DE PLANEJAMENTO

AS HISTÓRIAS DO CLIENTE SÃO DESCRITAS EM PEQUENOS CARTÕES

CADA HISTÓRIA TEM SEU TEMPO ESTIMADO EM *PONTOS*
(1 *PONTO* EQUIVALE A 1 *DIA DE TRABALHO IDEAL*)

AS ESTIMATIVAS SÃO REALIZADAS EM EQUIPE, JUNTO COM O CLIENTE

O GRUPO UTILIZA OS CARTÕES PARA DEFINIR JÁ NO INÍCIO DOS TRABALHOS COM SERÁ
A DIVISÃO DO SOFTWARE EM RELEASES

PRÁTICAS DO XP - EXTREME PROGRAMMING

REUNIÕES EM PÉ (STAND-UP MEETING)

REUNIÃO DIÁRIA, EM PÉ, COM DURAÇÃO MÁXIMA DE 20 MINUTOS PARA AVALIAÇÃO DO ANDAMENTO DO TRABALHO E DEFINIÇÃO DAS HISTÓRIAS QUE SERÃO DESENVOLVIDAS NO DIA

PROGRAMAÇÃO EM PAR

A XP EXIGE QUE TODA A CODIFICAÇÃO SEJA DESENVOLVIDA POR UMA DUPLA DE PROGRAMADORES.

UM RESPONSÁVEL PELA CODIFICAÇÃO (O CONDUTOR), E OUTRO PELA VERIFICAÇÃO E ACOMPANHAMENTO (O NAVEGADOR)

| PRÁTICAS DO XP - EXTREME PROGRAMMING

CÓDIGO COLETIVO

TODOS PROGRAMADORES TEM ACESSO E PODEM ALTERAR QUALQUER PEDAÇO DE CÓDIGO PARA MELHORÁ-LO OU COMPLETÁ-LO

REFACTORING

TODOS PROGRAMADORES TEM A OBRIGAÇÃO DE ALTERAR QUALQUER CÓDIGO QUE FOR ENCONTRADO ESCRITO DE FORMA CONFUSA, PARA DAR MELHOR CLAREZA E QUALIDADE.

PADRÕES DE DESENVOLVIMENTO

PRÁTICAS PADRONIZADAS DE GERAÇÃO DE CÓDIGO PARA FACILITAR A MANUTENÇÃO POSTERIOR

| PRÁTICAS DO XP - EXTREME PROGRAMMING

DESENVOLVIMENTO GUIADO POR TESTES

DESENVOLVIMENTO PRIMEIRAMENTE DOS TESTES, PARA DEPOIS SE
DESENVOLVER O CÓDIGO, VISANDO PASSAR NOS TESTES

TESTES DE UNIDADE — TESTAM CLASSES

TESTES DE ACEITAÇÃO — TESTAM A INTERAÇÃO ENTRE CLASSE PARA O
ATENDIMENTO DA ESTÓRIA

EQUIPES DO XP - EXTREME PROGRAMMING

GERENTE DE PROJETO

RESPONSÁVEL PELOS ASSUNTOS ADMINISTRATIVOS DO PROJETO

COACH

RESPONSÁVEL PELAS QUESTÕES TÉCNICAS DO PROJETO.

ANALISTA DE TESTE

RESPONSÁVEL EM GARANTIR A QUALIDADE ATRAVÉS DE TESTES ESCRITOS.

REDATOR TÉCNICO

RESPONSÁVEL EM DOCUMENTAR O SISTEMA.

DESENVOLVEDOR

RESPONSÁVEL EM ANALISAR, PROJETAR E CODIFICAR O SISTEMA.

CICLOS DO XP - EXTREME PROGRAMMING/



CICLOS DE PLANEJAMENTO E FEEDBACK

| METODOLOGIAS ÁGEIS – ESCOLHA A SUA

SCRUM
X
EXTREME PROGRAMMING
??

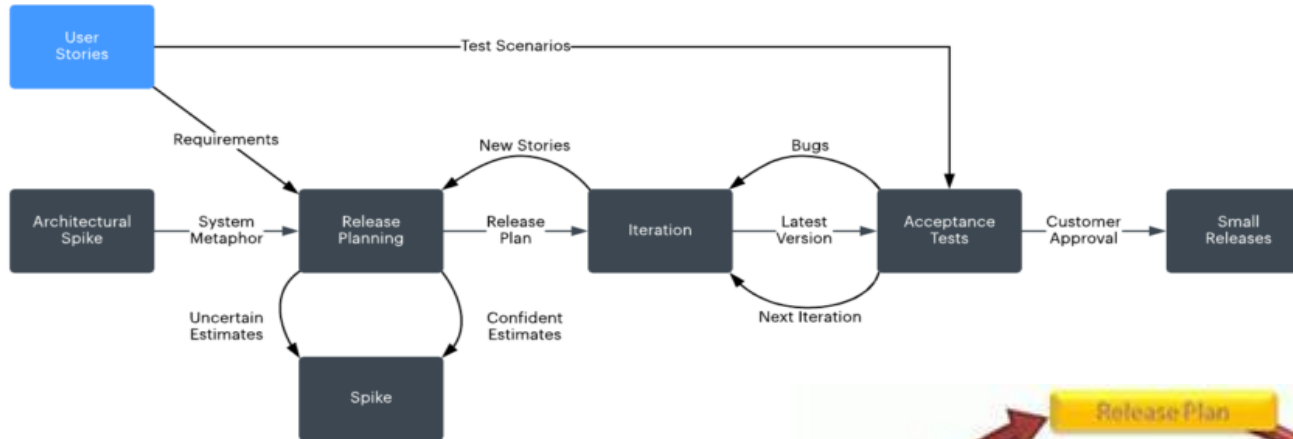
| METODOLOGIAS ÁGEIS – ESCOLHA A SUA

SCRUM & EXTREME PROGRAMMING

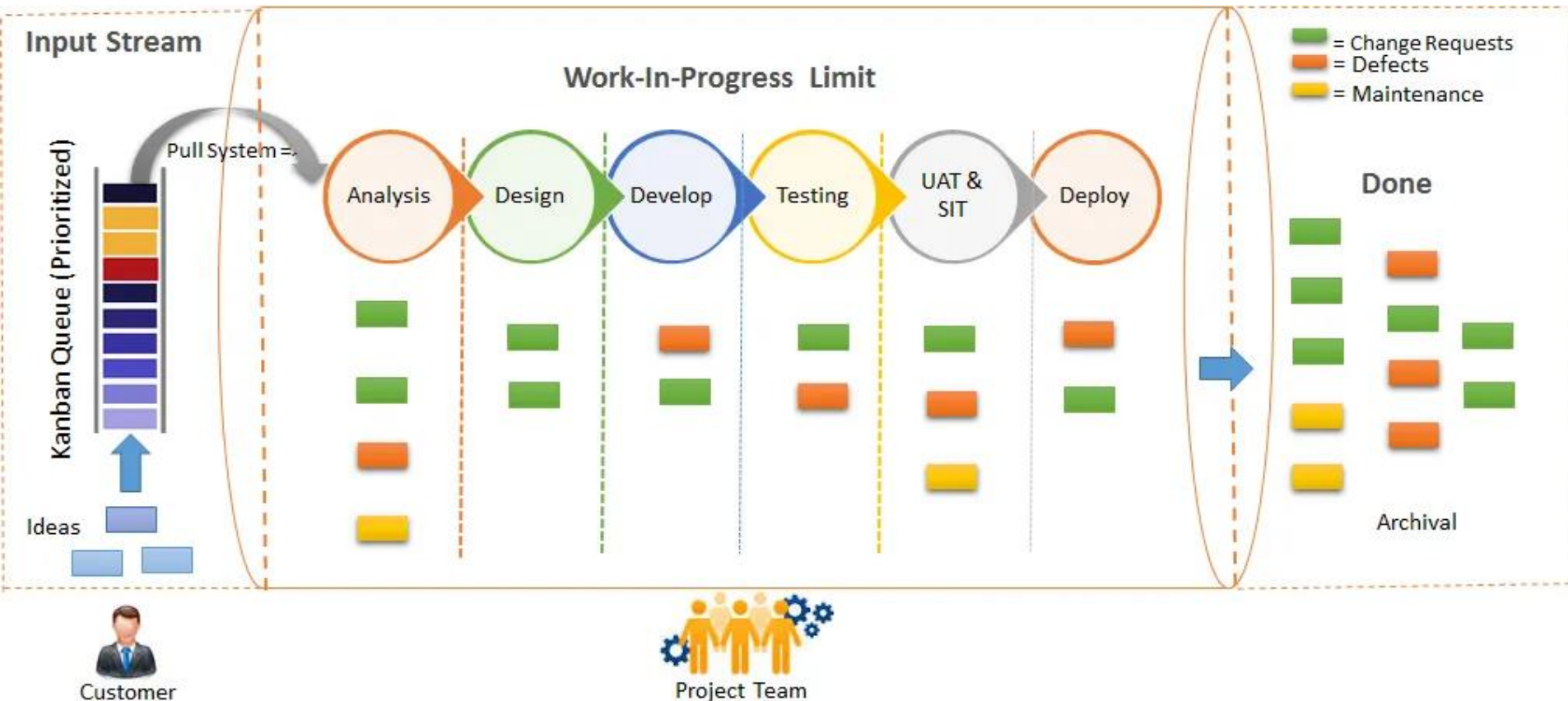
AMBAS RESPEITAM AO
AGILE MANIFESTO E SEUS 12 PRINCÍPIOS BÁSICOS

DESENVOLVEDORES TEM ADOTADO OCASIONALMENTE O SCRUM, MELHORADO
PELAS PRÁTICAS DE PROGRAMAÇÃO E DE TESTES DO XP

XP – EXTREME PROGRAMMING



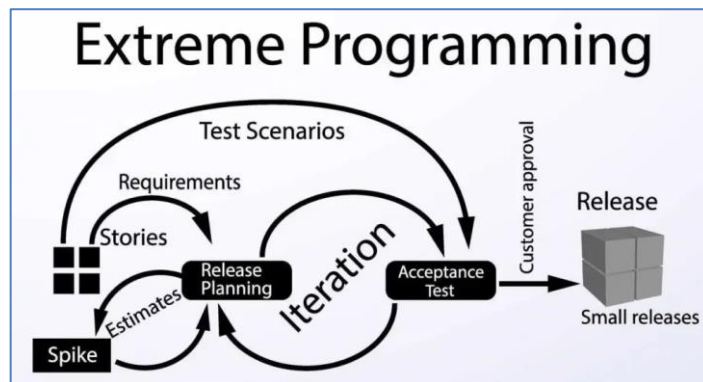
KANBAN NO DESENVOLVIMENTO DE SW



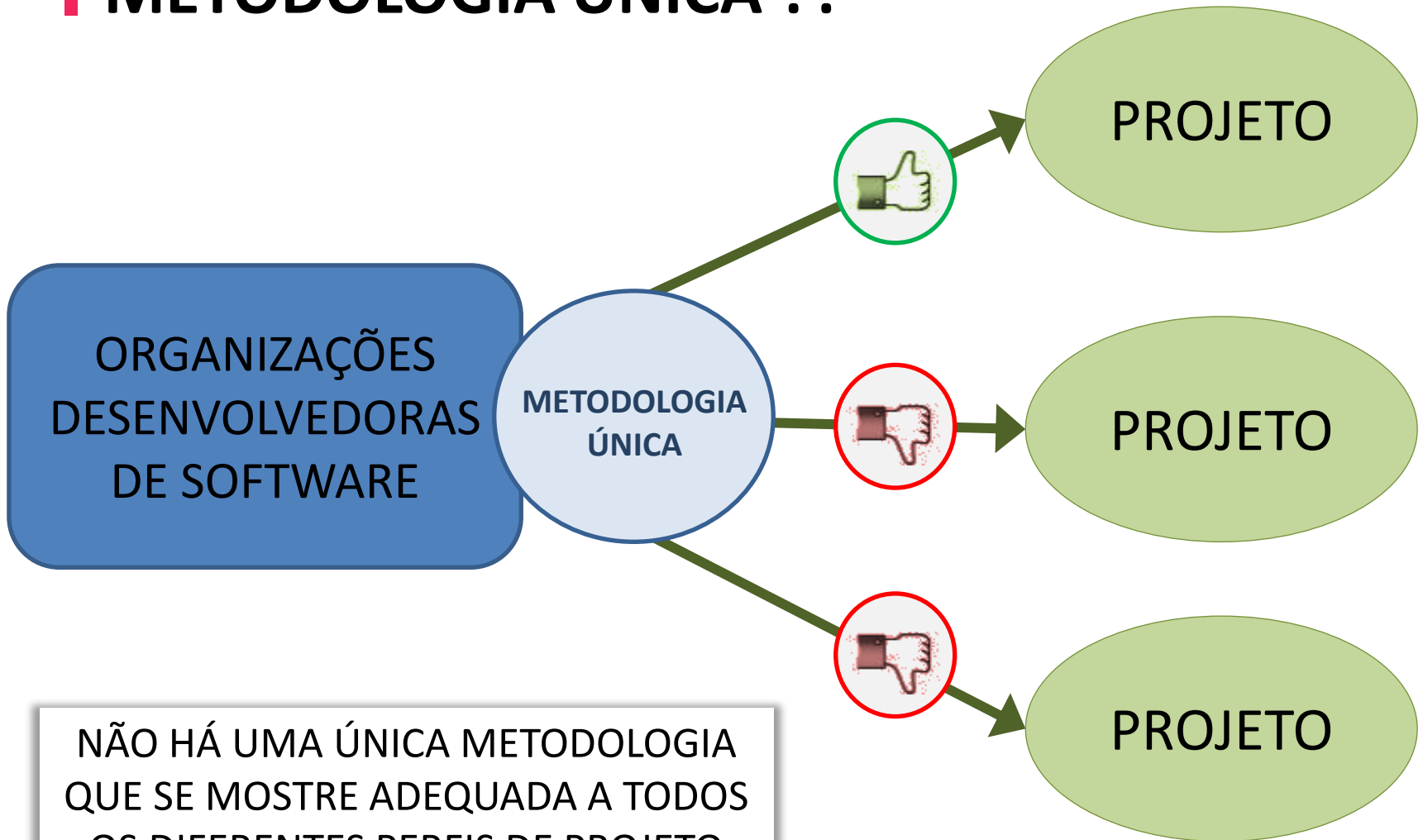
Fluxo “puxado” e contínuo (não há delimitação de Sprints)

Ritmo limitado pela capacidade da equipe

Incrementos variados, de acordo com a avaliação da equipe

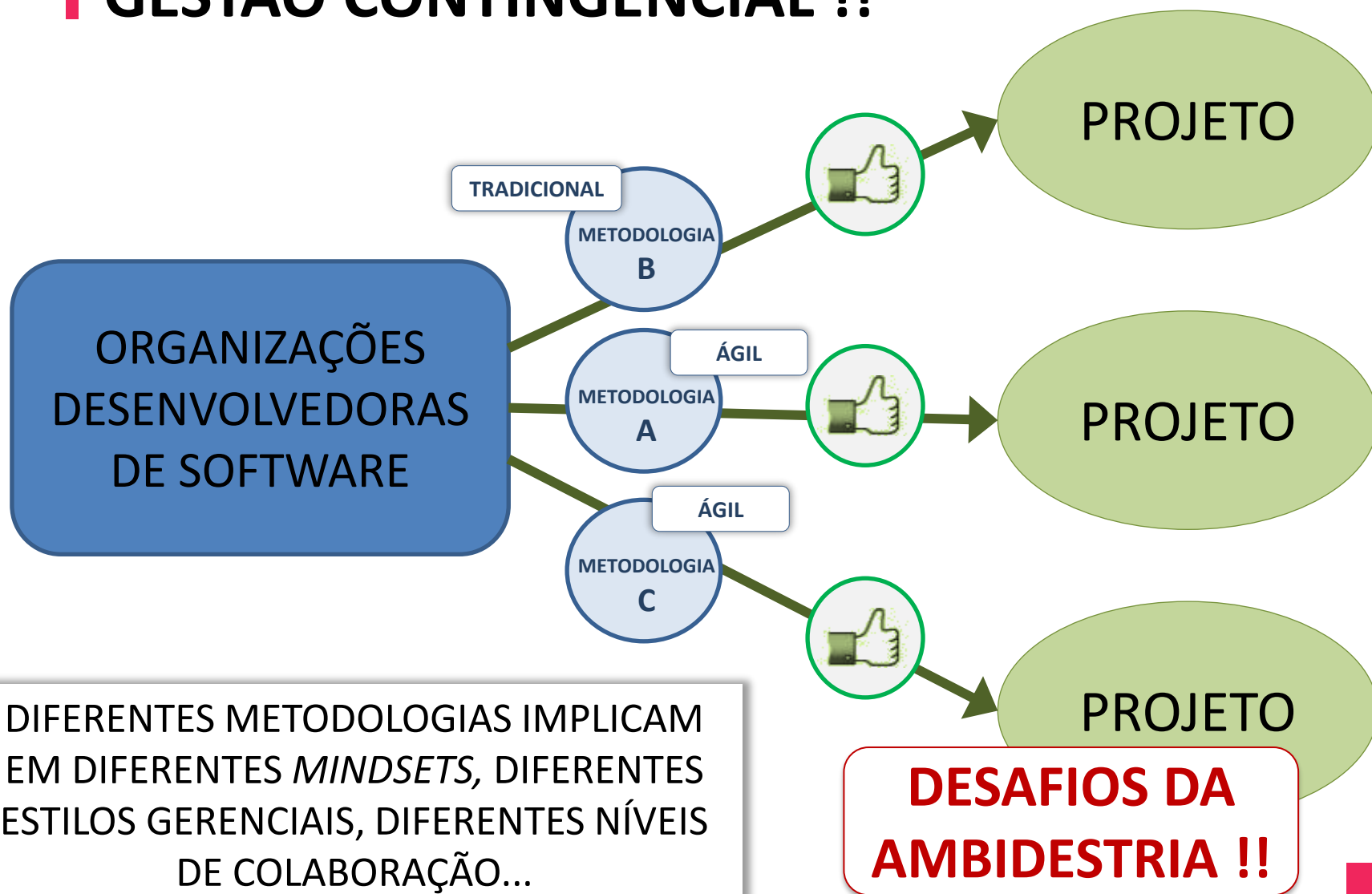


METODOLOGIA ÚNICA ??



NÃO HÁ UMA ÚNICA METODOLOGIA
QUE SE MOSTRE ADEQUADA A TODOS
OS DIFERENTES PERFIS DE PROJETO
(Cockburn – 2000; Boehm – 2004)

GESTÃO CONTINGENCIAL !!



DIFERENTES METODOLOGIAS IMPLICAM EM DIFERENTES *MINDSETS*, DIFERENTES ESTILOS GERENCIAIS, DIFERENTES NÍVEIS DE COLABORAÇÃO...
(Nerur et al – 2005)