



MANIFESTO ÁGIL, SCRUM E EXTREME PROGRAMMING

COMO CONSTRUIR SOFTWARE COM QUALIDADE E
QUE AGREGAM VALOR AO CLIENTE?

CAIO ROSÁRIO DIAS

- FORMADO EM TÉCNICO DE INFORMÁTICA – IFBA;
- QUINTO SEMESTRE DO CURSO DE ANÁLISE E DESENVOLVIMENTO DE SISTEMAS – IFBA;
- DESENVOLVIMENTO DE SOFTWARE PARA PLATAFORMAS WEB COM C#;

CONTATOS:

CAIODIAS12@GMAIL.COM

CAIODIAS@IFBA.EDU.BR

ENGENHARIA DE SOFTWARE

A ENGENHARIA DE SOFTWARE É UMA ÁREA DO CONHECIMENTO DA INFORMÁTICA VOLTADA A ESPECIFICAÇÃO, DESENVOLVIMENTO E MANUTENÇÃO DE SISTEMAS DE SOFTWARE APLICANDO TECNOLOGIAS E PRÁTICAS DE CIÊNCIA DA COMPUTAÇÃO, GERÊNCIA DE PROJETOS E OUTRAS DISCIPLINAS, OBJETIVANDO ORGANIZAÇÃO, PRODUTIVIDADE E QUALIDADE.



CENÁRIO DE DESENVOLVIMENTO DE SOFTWARE

MODERN RESOLUTION FOR ALL PROJECTS

	2011	2012	2013	2014	2015
SUCCESSFUL	29%	27%	31%	28%	29%
CHALLENGED	49%	56%	50%	55%	52%
FAILED	22%	17%	19%	17%	19%

The Modern Resolution (OnTime, OnBudget, with a satisfactory result) of all software projects from FY2011-2015 within the new CHAOS database. Please note that for the rest of this report CHAOS Resolution will refer to the Modern Resolution definition not the Traditional Resolution definition.

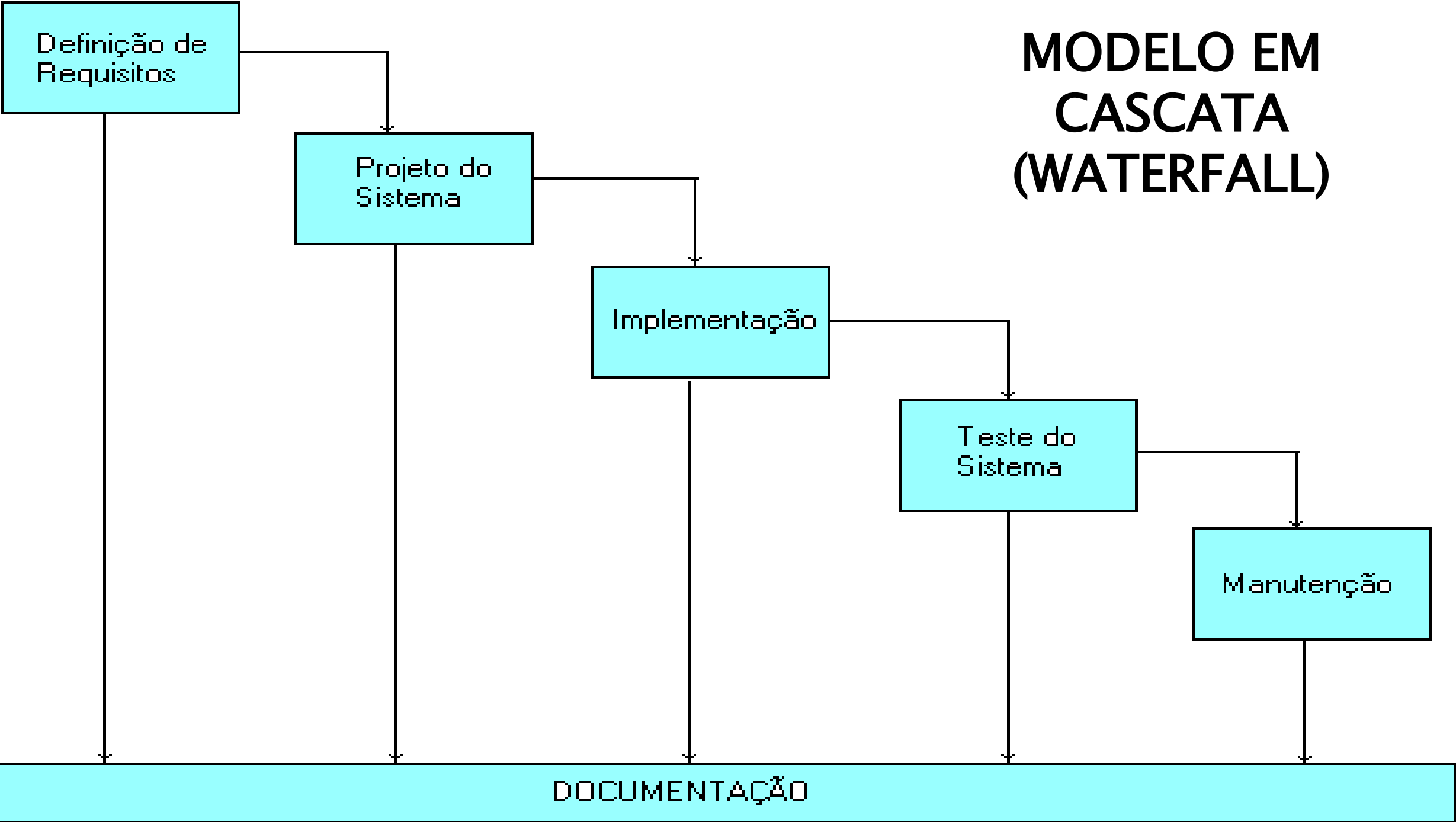
MOTIVOS ?

- Não cumprimento de prazos (72%);
- Problemas na **comunicação** (71%);
- Mudança de escopo (69%);
- Estimativa errada de prazo (66%).

The background is a dark blue gradient. In the corners, there are decorative white line-art patterns resembling circuit boards or neural networks, with lines connecting to small circles.

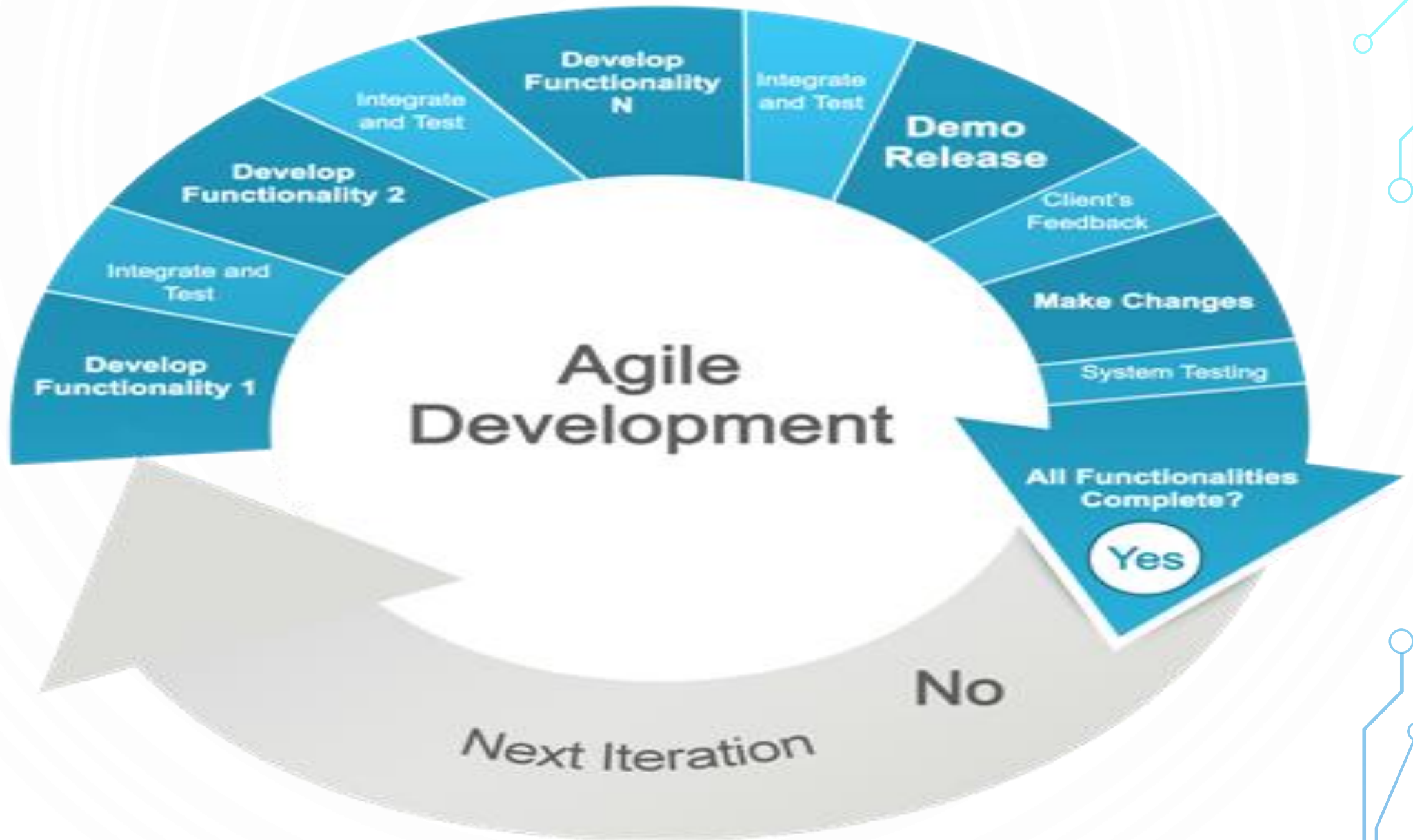
MODELOS DE DESENVOLVIMENTO DE SOFTWARE

MODELO EM CASCATA (WATERFALL)

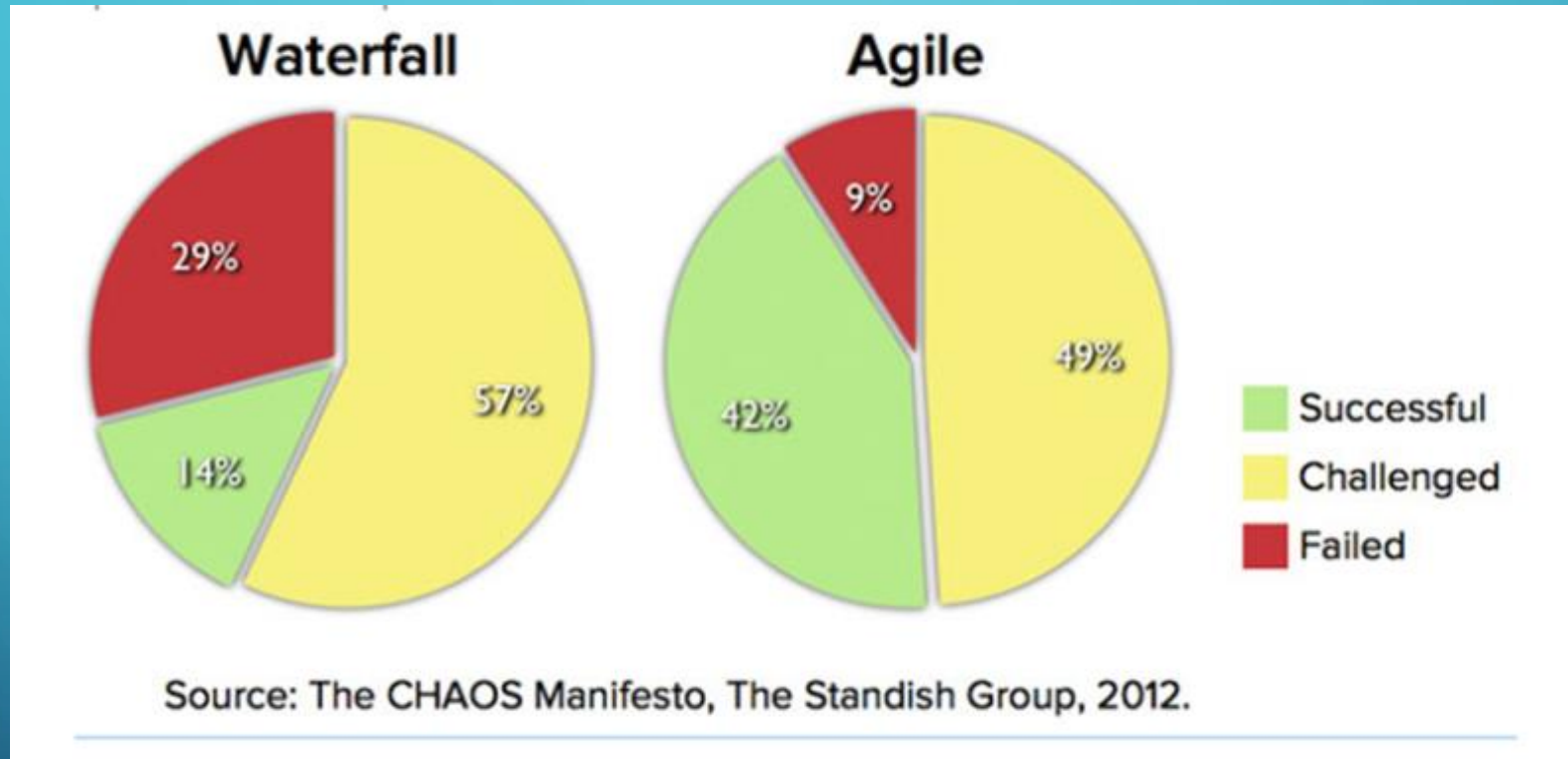


MAS PARECE UM MODELO IDEAL ... O SOFTWARE NÃO É ENTREGUE ?

- O cliente não está ciente de todo o processo de produção do software;
- Requisitos podem ser transmitidos de maneira incorreta e/ou superficial e desenvolvidos;
- O cliente pode estar pensando em uma nova funcionalidade que quebra as implementações já realizadas;
- Alguns clientes não possuem conhecimento sobre o **que um software pode oferecer**, e fazem falsas estimativas;



WATERFALL *VERSUS* AGILE



MAS QUAL É A DIFERENÇA?

No modelo ágil os planejamentos detalhados são feitos apenas para a fase **atual do projeto**, diferente dos modelos tradicionais que possuem o planejamento de todo projeto antes de sua implementação e testes. Para fases futuras, os planos são considerados apenas rascunhos que podem se adaptar a mudanças conforme o time aprende e passa a conhecer melhor o sistema e as tecnologias utilizadas.

QUESTÃO PARA ANÁLISE

Seu chefe solicita a realização de um trabalho na empresa para ser entregue em três meses. Para a realização da atividade os funcionários devem dividir-se em trios.

COMO FAZER ESSE TRABALHO?

- **Primeira Equipe: Responsabilidades Individuais;**
- **Segunda Equipe: Divisão em Etapas;**
- **Terceira Equipe: Entregas Cíclicas;**

FALHAS ?

PRIMEIRA EQUIPE:

1. **FALTA DE COMUNICAÇÃO:** A EQUIPE NÃO ESTÁ CIENTE DE TODO PROCESSO DE DESENVOLVIMENTO;
2. **COMPROMETIMENTO DO TRABALHO:** ALGUM MEMBRO PODE NÃO REALIZAR A ATIVIDADE;
3. **COOPERAÇÃO E APRENDIZADO:** A INDIVIDUALIZAÇÃO DO TRABALHO FAZ COM QUE OS MEMBROS NÃO APRENDAM UNS COM OUTROS E NEM AJUDEM-SE;
4. **DESCASO COM O SOLICITANTE:** O PROFESSOR NÃO SABE COMO ANDA O TRABALHO E NEM COMO PODE AJUDAR A EQUIPE;
5. **CRIAÇÃO DE TAREFAS DESNECESSÁRIAS;**

FALHAS ?

SEGUNDA EQUIPE:

1. **PROBLEMAS COM MODIFICAÇÕES:** A EQUIPE PODE ENCONTRAR UM PROBLEMA EM UMA FASE INICIAL DO SEU PROJETO, MODIFICAÇÕES EM ETAPAS INICIAIS PODEM IMPACTAR TODO O PROJETO;
2. **SEGUIR UM PLANO:** A EQUIPE CRIOU UM PLANO ESTATÍCO E NÃO FLEXÍVEL;
3. **DESCASO COM O SOLICITANTE:** O PROFESSOR NÃO SABE COMO ANDA O TRABALHO E NEM COMO PODE AJUDAR A EQUIPE;

TERCEIRA EQUIPE:

1. **ANALISAREMOS AO DECORRER DA PALESTRA;**

MANIFESTO ÁGIL

O MANIFESTO ÁGIL É UMA DECLARAÇÃO DOS VALORES E DOS PRINCÍPIOS QUE FUNDAMENTAM O DESENVOLVIMENTO ÁGIL DE SOFTWARE. ELE FOI ASSINADO EM 2001 POR DEZESSETE PROFISSIONAIS DE SOFTWARE QUE JÁ VINHAM EXPERIMENTANDO METODOLOGIAS E PRÁTICAS MAIS LEVES E ENXUTAS.

Valores

Interação entre Indivíduos

mais do que

Processos e Ferramentas

Produto funcionando

mais do que

Documentação Extensa

Colaboração com o cliente

mais do que

Termos Negociados (contrato)

Resposta às mudanças

mais do que

Cumprimento de planos

Ágil

Tradicional

OS 12 PRINCÍPIOS ÁGEIS

- Nossa maior prioridade é satisfazer o cliente, através da entrega adiantada e contínua de software de valor.
- Aceitar mudanças de requisitos, mesmo no fim do desenvolvimento.
- Entregar software funcionando com frequência, na escala de semanas até meses, com preferência aos períodos mais curtos.
- Pessoas relacionadas à negócios e desenvolvedores devem trabalhar em conjunto e diariamente, durante todo o curso do projeto.

OS 12 PRINCÍPIOS ÁGEIS

- Em intervalos regulares, o time reflete em como ficar mais efetivo, então, se ajustam e otimizam seu comportamento de acordo.
- Simplicidade: a arte de maximizar a quantidade de trabalho que não precisou ser feito.
- O Método mais eficiente e eficaz de transmitir informações para, e por dentro de um time de desenvolvimento, é através de uma conversa cara a cara.

SCRUM

É UM FRAMEWORK DE PROCESSO ÁGIL UTILIZADO PARA GERENCIAR E CONTROLAR O DESENVOLVIMENTO DE UM PRODUTO DE SOFTWARE ATRAVÉS DE PRÁTICAS ITERATIVAS E INCREMENTAIS.



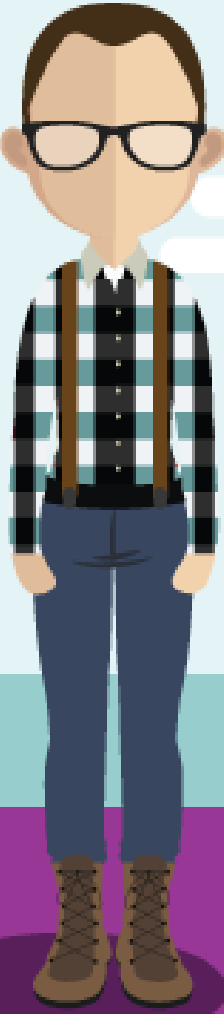
E O TRABALHO QUE SEU CHEFE PEDIU?

- Primeira Equipe: Responsabilidades Individuais;
- Segunda Equipe: Divisão em Etapas;
- Terceira Equipe: Entregas Cíclicas;

Product Owner

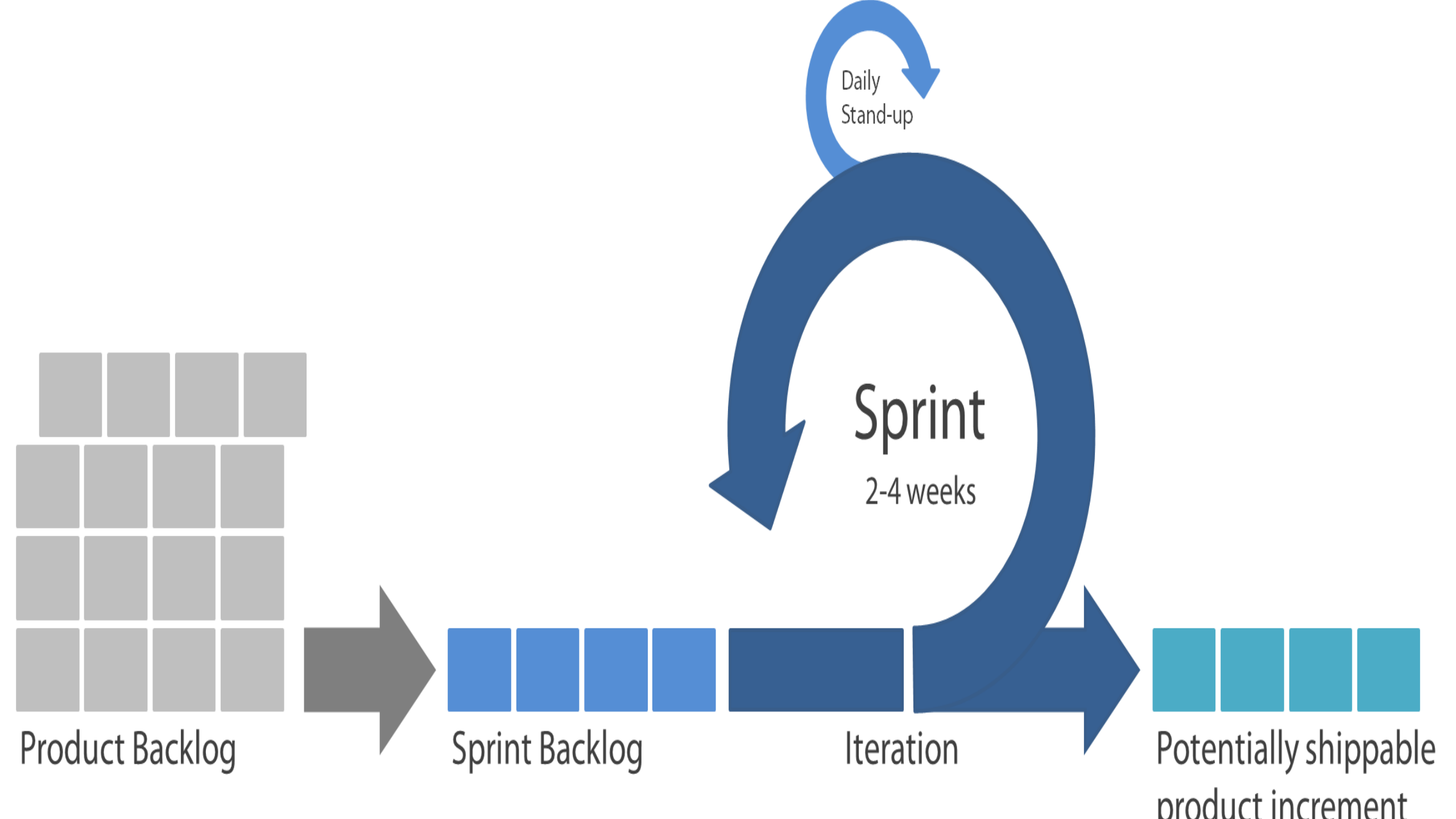


Scrum Master



Delivery Team






Backlog

Fila

Fazendo

Concluído



Cadastrar
Users




Validar
CPF user



Confirmar
senha




Captcha



Exibir
formulário

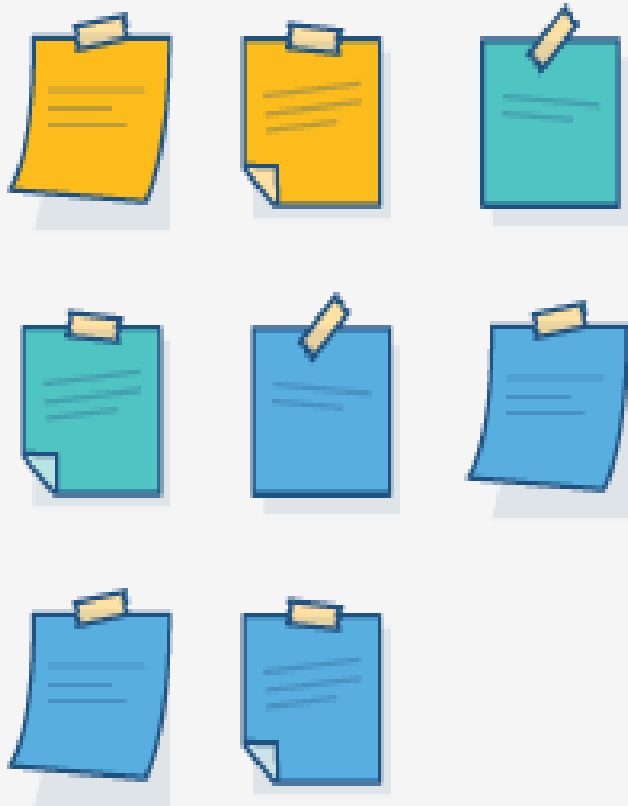


Campos
Obrigatórios



Exibir
resultado

To do



Doing



Done



Ticket ID: #42

The week view of the new
calendar feature

Chris

Alex

3

O QUE É A EXTREME PROGRAMMING?

É UMA METODOLOGIA ÁGIL DE DESENVOLVIMENTO DE SOFTWARE VOLTADA PARA TIMES DE PEQUENO A MÉDIO PORTE, NO QUAL OS REQUISITOS SÃO VAGOS E MUDAM FREQUENTEMENTE. O XP VALORIZA A ENTREGA CONTÍNUA DE SOFTWARE FUNCIONAL, COMUNICAÇÃO DA EQUIPE, TESTES AUTOMATIZADOS E A REDUÇÃO DE CUSTOS. É FLEXÍVEL PARA A MUDANÇAS DE REQUISITOS, VALORIZANDO O FEEDBACK COM O USUÁRIO E A QUALIDADE DO CÓDIGO-FONTE FINAL.





VALORES DO XP

- COMUNICAÇÃO;
- FEEDBACK;
- CORAGEM;
- RESPEITO;
- SIMPLICIDADE;

COMUNICAÇÃO

TER FOCO NA COMUNICAÇÃO É ESSENCIAL EM PROJETOS DE SOFTWARE, POIS É A PRINCIPAL FORMA DE TRANSMITIR E TROCAR INFORMAÇÕES E CONHECIMENTOS. POR ESSA RAZÃO, É IMPORTANTE INCENTIVAR MEIOS EFICAZES DE COMUNICAÇÃO.

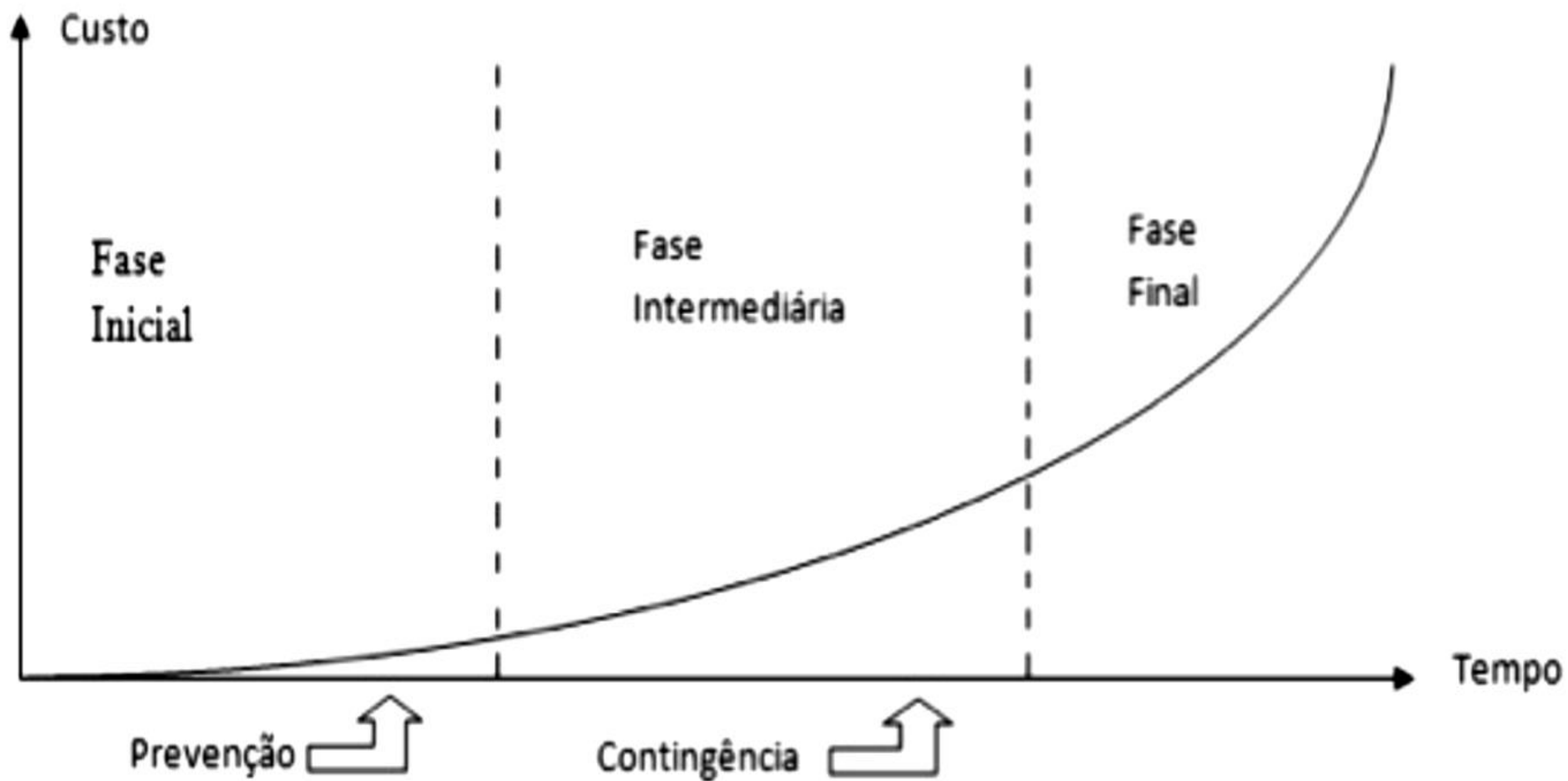
PARA QUE OS DESENVOLVEDORES COMPREENDAM O QUE O CLIENTE DESEJA E ESTE ÚLTIMO ENTENDA OS DESAFIOS TÉCNICOS QUE PRECISAM SER VENCIDOS, É PRECISO QUE HAJA COMUNICAÇÃO ENTRE AS PARTES.

FEEDBACK

“NORMALMENTE, QUANTO MAIS CEDO
DESCOBRIMOS UM PROBLEMA, MENOS PREJUÍZOS
ELE PODE CAUSAR E MAIORES SÃO AS CHANCES DE
RESOLVÊ-LO DE FORMA BARATA.”

FALHOU ? NÃO ATENDEU ? ÓTIMO!

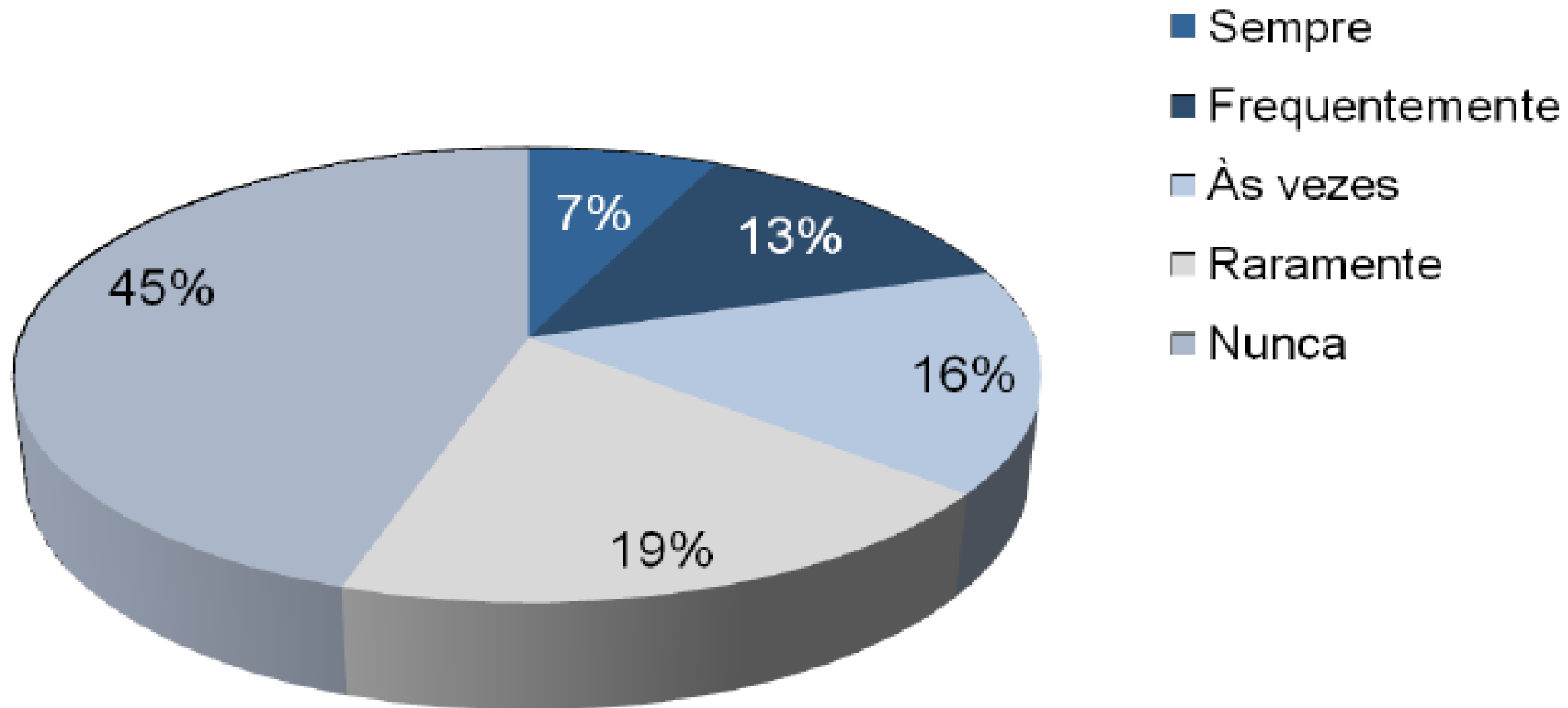
VAMOS REFAZER DA MANEIRA CORRETA!



SIMPLICIDADE

ELA VISA MANTER O TRABALHO O MAIS SIMPLES E FOCADO POSSÍVEL, ENTREGANDO SOMENTE O QUE REALMENTE AGREGA VALOR AO CLIENTE. ACRESCENTAR SUPORTE PARA FUTURAS FUNCIONALIDADES DE FORMA DESNECESSÁRIA COMPLICA O *DESIGN* E ELEVA O ESFORÇO PARA DESENVOLVER INCREMENTOS SUBSEQUENTES.

Uso de Funcionalidades pelo Cliente



CORAGEM

MUDANÇAS EM PROJETOS SÃO INEVITÁVEIS E ADAPTAR-SE É A MELHOR SOLUÇÃO. PARA ISSO, A EQUIPE BUSCA CONFIANÇA EM SEUS MECANISMOS DE PROTEÇÃO, TAIS COMO DESENVOLVIMENTO ORIENTADO A TESTES, PROGRAMAÇÃO EM PAR E INTEGRAÇÃO CONTÍNUA.

RESPEITO

RESPEITO É UM VALOR QUE DÁ SUSTENTAÇÃO A TODOS OS DEMAIS. MEMBROS DE UMA EQUIPE SÓ IRÃO SE PREOCUPAR EM COMUNICAR-SE MELHOR, POR EXEMPLO, SE IMPORTAREM UNS COM OS OUTROS. RESPEITO É O MAIS BÁSICO DE TODOS OS VALORES. SE ELE NÃO EXISTIR EM UM PROJETO, NÃO HÁ NADA QUE POSSA SALVÁ-LO. SABER OUVIR, SABER COMPREENDER E RESPEITAR O PONTO DE VISTA DO OUTRO É ESSENCIAL PARA QUE UM PROJETO DE SOFTWARE SEJA BEM SUCEDIDO.

O QUE HÁ DE DIFERENTE NOS VALORES DO XP ?

A ABORDAGEM DO XP ALÉM DE PROMOVER TÉCNICAS PARA PRODUÇÃO DE SOFTWARE COM QUALIDADE PROMOVE TAMBÉM A VALORIZAÇÃO HUMANA NOS PROCESSOS DE DESENVOLVIMENTO SOFTWARE. QUANTO MELHOR A COMUNICAÇÃO, RESPEITO E CONFIANÇA ENTRE OS ENVOLVIDOS NA CONSTRUÇÃO DO SOFTWARE, MAIORES SÃO A CHANCE DO SUCESSO. SOFTWARES SÃO PRODUZIDOS POR PESSOAS QUE PRECISAM ENTENDER, APRENDER E GOSTAR DO SEU TRABALHO.

A decorative graphic on the left side of the slide, consisting of a network of white lines and small circles on a blue background, resembling a circuit board or a neural network structure.

PRÁTICAS DO XP

Ambiente de Trabalho

- **Sentar Junto:** A equipe deve trabalhar em um ambiente amplo e aberto, onde a comunicação seja fortalecida;
- **Time Multidisciplinar:** Uma equipe não deve ser formada apenas por desenvolvedores ou profissionais de apenas uma área. O espírito de contribuição para o projeto deve prevalecer;
- **Área de Trabalho Interativa:** Transformar o Ambiente de trabalho num reflexo do Projeto;

AMBIENTE DE TRABALHO

ABERTO

- Ambiente livre para interação entre os desenvolvedores;
- A troca de informações é constante e apoiada.

FECHADO

- As pessoas ficam limitadas apenas ao seu local de trabalho;
- Comunicação entre os indivíduos não são bem vistas pelos gestores.

DESENVOLVIMENTO DO SOFTWARE

- Refatoração;
- Programação em Pares;
- Histórias de Usuário;
- Código Compartilhado;
- Desenvolvimento Dirigido por Testes;
- Menor documentação possível;

STORY USERS

“ HISTÓRIAS NÃO SÃO REQUISITOS, ELAS SÃO DISCUSSÕES SOBRE RESOLVER PROBLEMAS PARA A ORGANIZAÇÃO.”

AS STORY USERS TEM A FUNÇÃO DE DESCREVER FUNCIONALIDADES DO SISTEMA SOLICITADAS PELOS CLIENTES DE FORMA ÁGIL E MENOS

<Título>

Como um <papel de usuário>

Eu quero <objetivo>

Para que <razão de negócio>

Reposição de estoque

Como um repositor de estoque

Eu quero localizar quais prateleiras do supermercado que contêm menos de 50% de sua capacidade de produtos

Para que eu possa fazer a reposição dos produtos e não faltar itens para os clientes em compras.

NOSSA STORY USER!

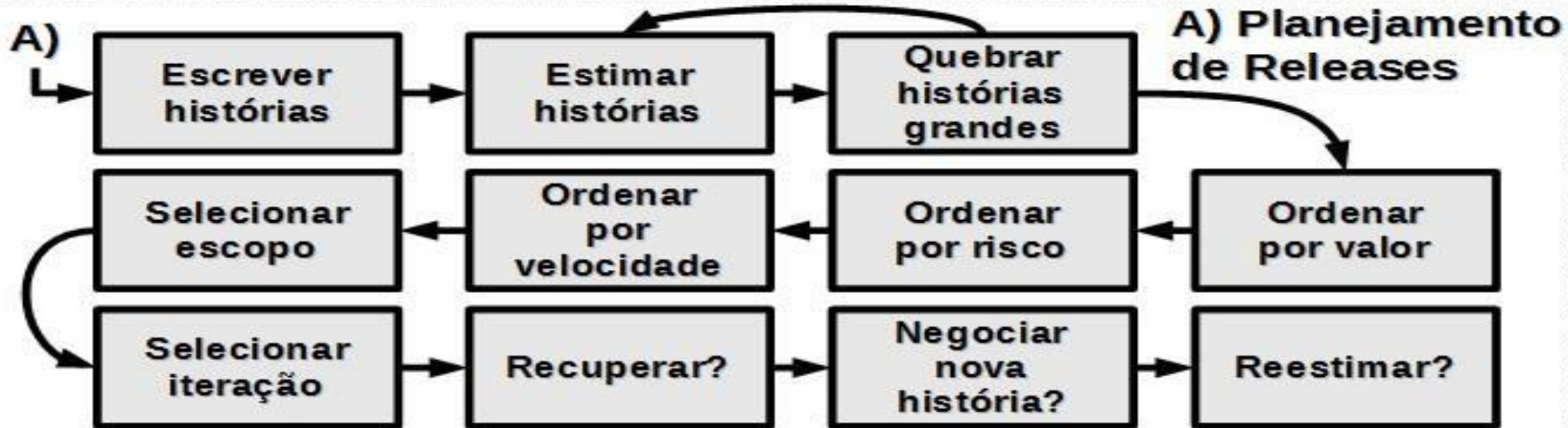
- COMO UM ...
- EU QUERO ...
- PARA QUE ...

JOGO DO PLANEJAMENTO

O JOGO DO PLANEJAMENTO ENVOLVE OS CLIENTES E OS DESENVOLVEDORES PARA PLANEJAREM AS ENTREGAS DE UMA FORMA COLABORATIVA. TAMBÉM CHAMADO DE *RELEASE PLANNING*, O NOME *JOGO* MOSTRA A ESSÊNCIA DA DINÂMICA: TRATAR O PLANEJAMENTO COMO UM JOGO QUE CONTÉM UM OBJETIVO, JOGADORES, PEÇAS E REGRAS.

O DESENVOLVIMENTO ITERATIVO E INCREMENTAL É UTILIZADO NO JOGO DO PLANEJAMENTO. OS INCREMENTOS SÃO DEFINIDOS NO PLANEJAMENTO DE RELEASES; E AS ITERAÇÕES, NO PLANEJAMENTO DE ITERAÇÕES.

Jogo do Planejamento



ENTREGAS FREQUÊNTES

POR QUE DEIXAR O CLIENTE ESPERANDO UM LONGO TEMPO PARA ENTREGAR UM REQUISITO IMPORTANTE?

- Entrega de valor adiantado e contínuo;
- O processo é aprimorado rapidamente por falhar mais cedo (conceito *fail fast*);
- Feedback do cliente mais cedo para confirmação ou adaptação dos requisitos;
- Satisfação dos usuários por ter respostas rápidas às suas necessidades;
- Reduz a taxa de defeitos por precisar realizar testes completos em ciclos menores;
- O código é atualizado e integrado com maior frequência;
- O software não fica ultrapassado;
- Facilita enxergar os diversos desperdícios ocultos nas grandes entregas;
- Evita a procrastinação de prazos;
- Transparência;

REUNIÃO DIÁRIA DE PÉ

Você já participou de reuniões longas e improdutivas? Reuniões estão no topo da lista de coisas entediadas que desperdiçam o tempo da maioria dos desenvolvedores.

Reuniões Ágeis !

- O que eu fiz ontem?
- O que eu farei hoje?
- O que há de problemas no meu caminho?



PROGRAMAÇÃO EM PARES

Duas pessoas trabalhando em um único computador produzirá mais que duas pessoas trabalhando separadas, já que isso aumenta o foco, e também gera qualidade e troca de conhecimento.



$$1 + 1 > 2$$

VANTAGENS

- DISSEMINAÇÃO DE PADRONIZAÇÃO DO CÓDIGO PARA EQUIPE;
- POSSE COLETIVA DO CÓDIGO E DAS REGRAS DE NEGÓCIO DA EMPRESA;
- APROXIMAÇÃO DO TIME;
- DISSEMINAÇÃO DE CONHECIMENTO;
- REVISÃO CONTÍNUA DO CÓDIGO;
- NÃO HÁ O MONOPÓLIO DE CONHECIMENTO;
- MENOR CHANCE DE FALHAS EM IMPLEMENTAÇÕES;
- PRESSÃO DO PAR;
- NIVELAMENTO;

OBRIGADO!

eXtreme Programming – Práticas para o dia a dia no desenvolvimento ágil de software – Casa do Código;

Agile – Desenvolvimento de Software com entregas frequentes e com foco no valor de negócio – Casa do Código;

Um Estudo de Caso da Adoção das Práticas e valores do Extreme Programming – Dissertação de Mestrado – Vinícius Manhães Teles;

Desenvolvimento Ágil – <http://www.desenvolvimentoagil.com.br/xp/>