



# **A aula interativa do Módulo 4 - Bootcamp Arquiteto Cloud Computing começará em breve!**

**Atenção:**

- 1) Você entrará na aula com o microfone e o vídeo DESABILITADOS.**
- 2) Apenas a nossa equipe poderá habilitar seu microfone e seu vídeo em momentos de interatividade, indicados pelo professor.**
- 3) Utilize o recurso Q&A para dúvidas técnicas. Nossos tutores e monitores estarão prontos para te responder e as perguntas não se perderão no chat.**
- 4) Para garantir a pontuação da aula, no momento em que o professor sinalizar, você deverá ir até o ambiente de aprendizagem e responder a enquete de presença. Não é necessário encerrar a reunião do Zoom, apenas minimize a janela.**

# Soluções para Desenvolvimento

Segunda Aula Interativa

Prof.<sup>a</sup> Analía Irigoyen

# Arquiteto de Soluções - Cloud


---

Segunda Aula Interativa – Módulo 4 - Soluções para Desenvolvimento

Prof.<sup>a</sup> Analia Irigoyen



## Princípios das 3 maneiras

Objetivo da Segunda Maneira	Princípios e Práticas
<p>Rápido feedback em todos os estágios do fluxo de valor (Direita para a Esquerda)</p> 	<ol style="list-style-type: none"><li>1. Ver problemas quando ocorrem (“ir ao gembu”)</li><li>2. Aglomerar quando problema aparece (Andon)</li><li>3. Qualidade próxima da fonte (menos aprovações)</li><li>4. Telemetria self service e irradiadores de informação disponível para todos</li><li>5. Desenvolvimento por hipóteses e Testes A/B</li><li>6. Equipes Dev e Ops compartilham o trabalho diário e plantões de suporte 24 x7</li><li>7. Revisão de código usando as técnicas: Programação em pares, sobre os ombros, divulgação por email, assistida por ferramentas</li></ol>

## Telemetria



- ✓ Coleta remota e automática de dados para monitorar a saúde das aplicações.
- ✓ Visão de todo o fluxo de valor permite otimizar e comunicar melhor.

- Fornece feedback efetivo:
- ✓ **Experiência do cliente;**
  - ✓ Erros no pipeline de implantação e após deploy;
  - ✓ Eventos proativos;
  - ✓ **Funcionalidades mais usadas pelos clientes.**

### Feedback

Visão clara do fluxo de valor em todos os estágios do ciclo de vida do serviço, produto, desenvolvimento e implementação, entrada em operação e retirada.

Dessa forma, todos os participantes do fluxo de valor tem a oportunidade de aprender desde os primeiros estágios do projeto.

## Opções para resolver problemas



### **Correção: Fix Forward**

É uma mudança no código e deve ser executada em ambientes com grande maturidade: Teste automatizado, implantação rápida e telemetria.

### **Reversão: Rollback**

Usuário deixa de receber a implantação e há 2 formas:

1. Implantar a versão anterior (Ex.: Canário);
2. Alternância de recursos: Mais fácil de reverter e menos arriscada.

## Suporte compartilhado



A implantação de um pipeline com testes automatizado não garante zero erro, em função da complexidade dos sistemas.

Erro de software depende que a equipe Dev priorize a solução para não impactar a operação por muito tempo.

Todos os participantes do Fluxo de Valor devem compartilhar as responsabilidades para resolver os incidentes em produção.



## Eficácia do Pull Request

Recomendação é que a revisão seja realizada pelo colega **antes** de efetivar o código no trunk.

Revisão deve ser em pequenos lotes. Shoup diz: “Quando aumentamos o código de 10 linhas para 100 linhas, a chance de erro aumenta 10 vezes”.

Mudanças críticas podem ter a revisão do especialista no assunto (segurança, BD, redes).

### Programação em pares



O código é criado por duas pessoas trabalhando juntas em um único computador.

A melhor maneira de parear o programa é sentar lado a lado na frente do monitor para revezar as atividades de codificação (piloto) e revisão (navegador).

Tempo gasto a mais é 15% e aumenta qualidade do software de 70% para 85%.

# Importância da Terceira Maneira



A diagram on the left side of the slide. It features a central blue circle with three teal-colored, rounded rectangular shapes pointing towards it from the right. These shapes are arranged in a vertical column, corresponding to the three horizontal teal bars on the right. The top and bottom teal shapes have a small white triangle pointing towards the central blue circle, while the middle dark teal shape has a small white triangle pointing away from it.

Correr Riscos (Cultura Justa – Fatos e Dados).

Loop de feedback contínuo.

Melhoria contínua até a perfeição.

# Aprendizado x Segurança psicológica

Segurança Psicológica



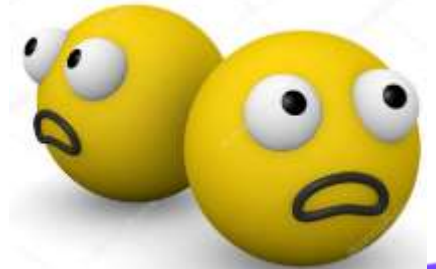
Zona de Conforto

Zona de Aprendizado



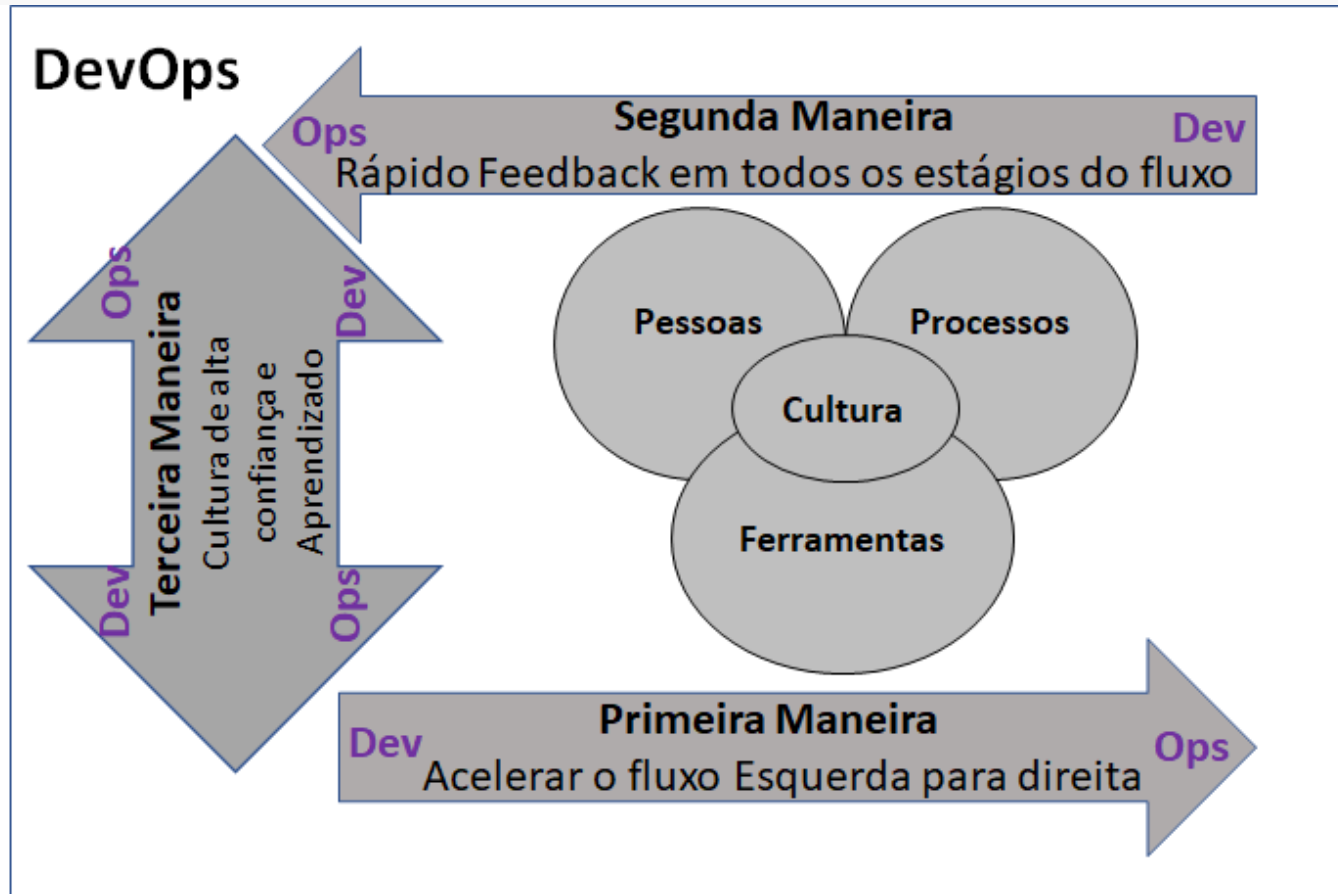
Zona da Apatia

Zona da Ansiedade



## Motivação e Responsabilidade

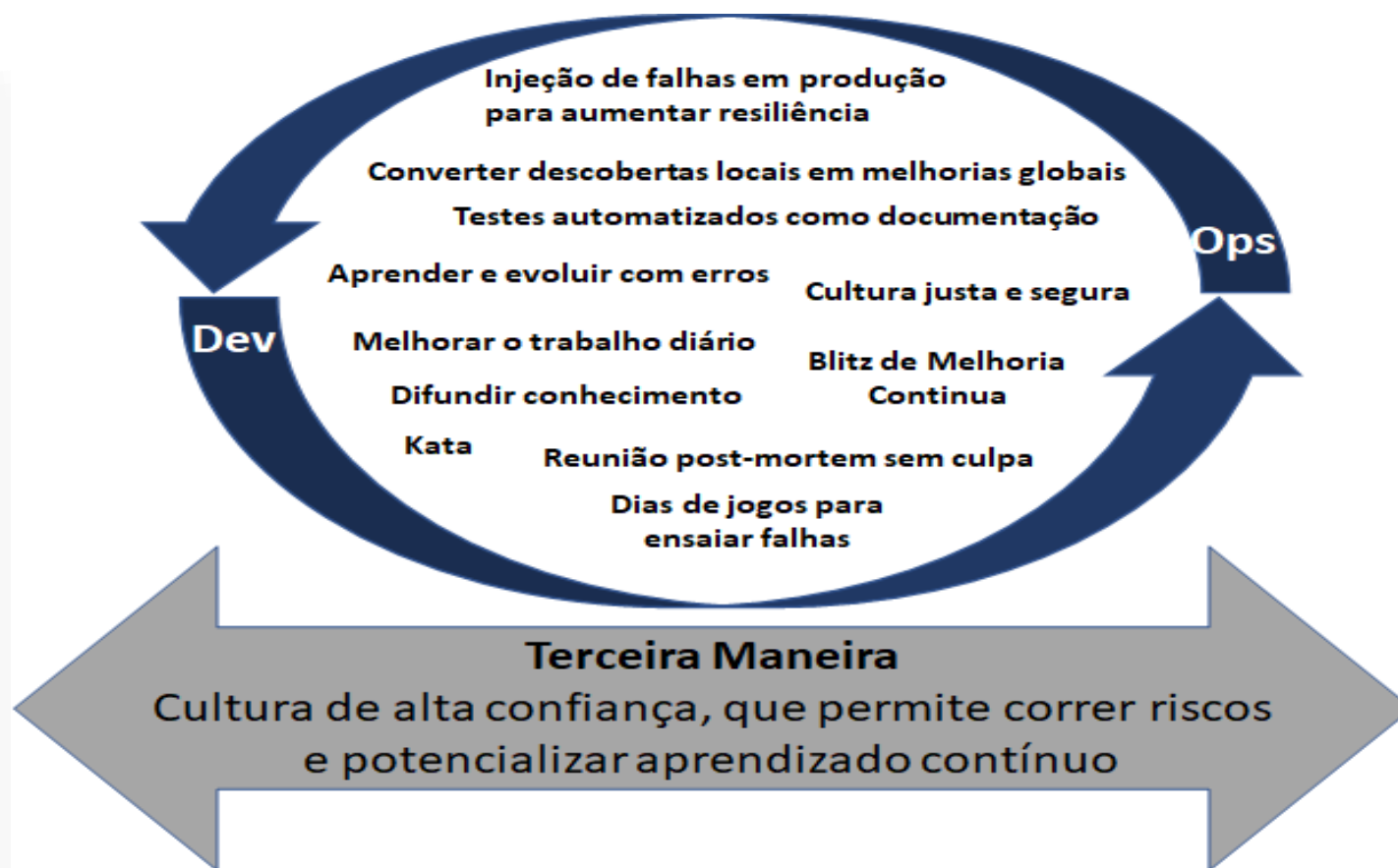
# Importância da 3ª Maneira



Overview do DevOps. Fonte: Elaborado pelo autor.



# Introdução da Terceira Maneira



Segunda Maneira. Fonte: adaptado de (MUNIZ; SANTOS; IRIGOYEN; MOUTINHO. 2019).

## Terceira Maneira – Principais Pontos



- ☐ Tipos de Macaco do exército simiano (Case Netflix).
- ☐ Reunião post-mortem livre de culpa.
- ☐ Dias de jogos.
- ☐ Descobertas.

## Exército de Símios Netflix

- ✓ O Chaos Monkey evoluiu para novos macacos chamado “Exército Símio”.
- ✓ Essa técnica ficou famosa quando houve uma queda de zona da AWS em 2011 e a Netflix não foi afetada porque tinha se preparado proativamente.

IGTi



# Exército de Símios Netflix



✓ **Chaos Gorilla:** Simula a falha de **uma zona inteira** de disponibilidade AWS (Serviço Web Amazon).

✓ **Chaos Kong:** Simula indisponibilidade em **regiões inteiras** da AWS (Ex.: Europa, América do Norte, África, etc.).

# Exército de Símios Netflix



## Macaco de Latência

- ✓ Causa atrasos ou paralisações artificiais.
- ✓ Simula a **degradação de serviço** para garantir que serviços dependentes respondam de forma adequada.

## Macaco Janitor

- ✓ Responsável em garantir que o ambiente esteja livre de desperdício e desorganização.
- ✓ **Procura recursos não utilizados e desativa.**



# Exército de Símios Netflix



## Macaco de Conformidade

- ✓ Localiza e desliga instâncias AWS que **não seguem as melhores práticas** (Ex.: falta de e-mail para alerta).
- ✓ O Macaco de Segurança **é uma extensão para vulnerabilidades.**

## Macaco Doutor

- ✓ Verifica a integridade de cada instância.
- ✓ **Desliga instâncias não íntegras** quando o responsável não resolve a causa raiz no tempo combinado.

## Dias de Jogos – Games Days



Esse conceito vem da engenharia da resiliência, cujo objetivo é criar exercícios programados para aumentar a resiliência através da injeção de falhas de grande escala em sistemas críticos.

Foi popularizado por Jesse Robbins pelo trabalho que fez na Amazon para garantir a disponibilidade do site. Ele ficou conhecido como o “Mestre do Desastre” e defende que **“Um serviço não está realmente testado até o estragarmos em produção”**.

# Dias de Jogos – Etapas



## 1. Planejar a interrupção

Ex.: Simular perda completa de datacenter

## 2. Adotar medidas

Ex.: Criar site de contingência

## 3. Testar medidas

Ex.: Usar site de contingência

## 4. Executar interrupção

Ex.: Avaliar resultados da contingência

## 5. Seguir plano e aprender

Ex.: Avaliar situações não previstas

## Post – Mortem - Aprendizado com falhas **IGTI**

Para o verdadeiro aprendizado, é necessário combater a teoria da maçã podre, que busca “eliminar as pessoas que causaram os erros”.

✓ Segundo Dekker, o erro humano é a consequência do projeto de ferramentas que as pessoas recebem para trabalhar: **Deve-se buscar a causa sistêmica dos erros.**

Um engenheiro do Google confessou: "Eu estraguei uma linha de código e isso nos custou um milhão de dólares em receita" Não foi demitido...

## Post-Mortem - Dicas para criar cultura justa



1. Nunca nomear ou envergonhar quem é o culpado da falha.

2. Incentivar quem compartilha problemas do sistema.

3. Criar confiança para que a equipe aprenda com problemas.

4. Lições aprendidas sem culpa e injeção controlada de falhas em produção.



# Reunião post-mortem livre de culpa **IGTI**

**Após a solução de incidentes, objetivo é entender e divulgar:**

1. Ações que deram certo;
2. Ações que podem ser aprimoradas;
3. Erros e medidas para evitar recorrência.

**Etapas importantes:**

1. Agendar a reunião com as pessoas envolvidas;
2. Realizar a reunião;
3. Publicar o resultado da reunião.

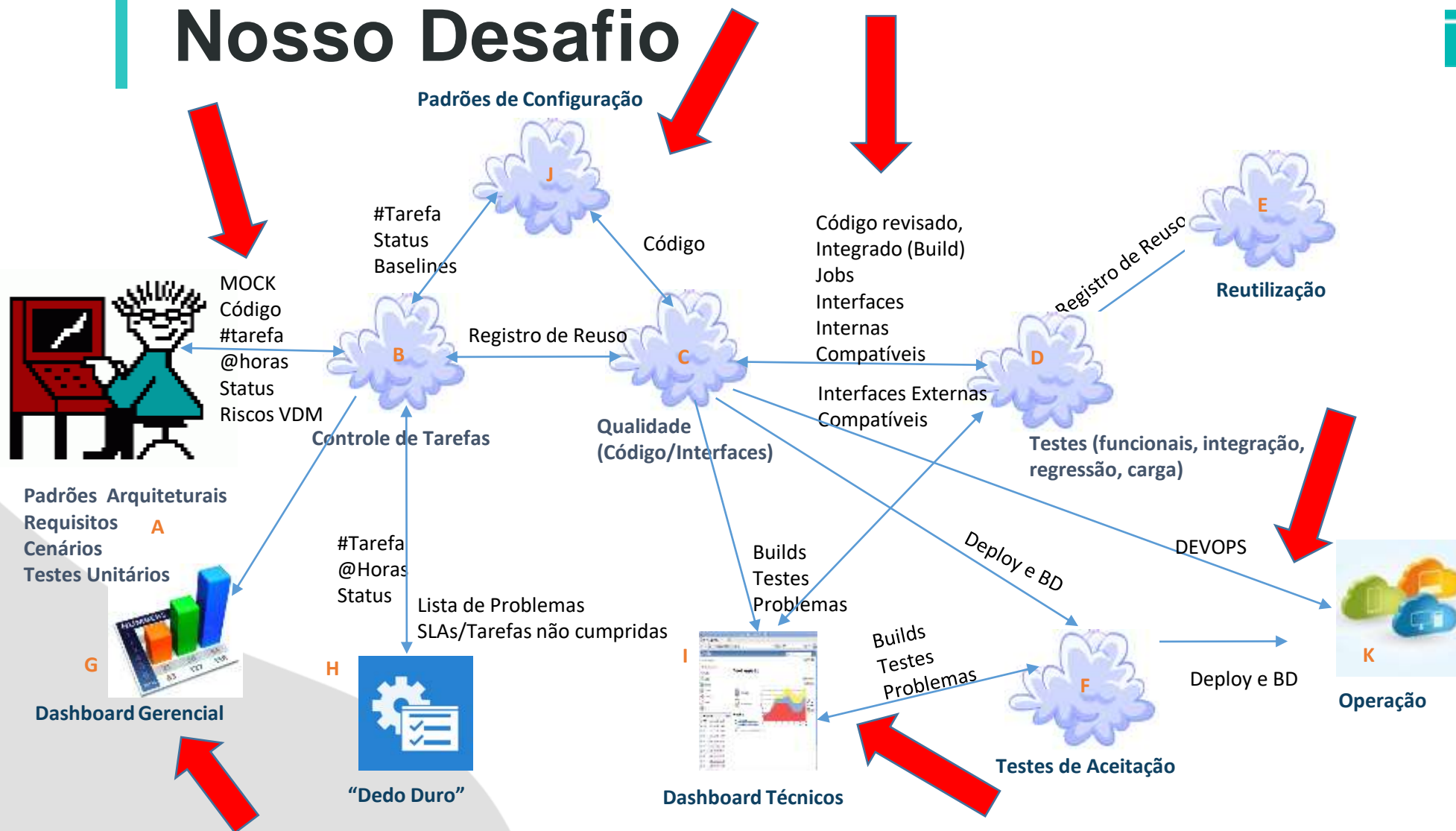
# Como transformar melhorias Locais em Globais



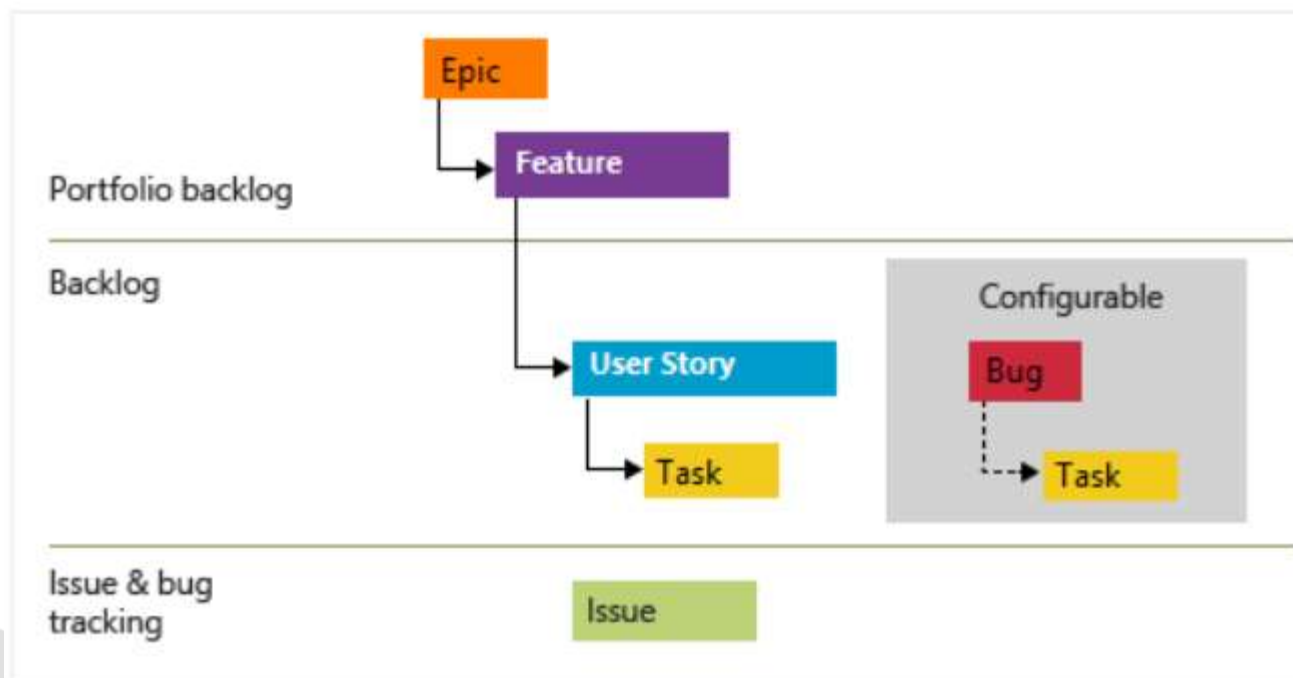
Transformar processos que estão documentados em editores de texto em *workflows* e/ou *scripts* automatizados é uma outra forma de transformar um conhecimento local em organizacional, permitindo a reutilização e a sua ampla utilização, fornecendo valor agregado para todos os que usam.

# Nosso Desafio

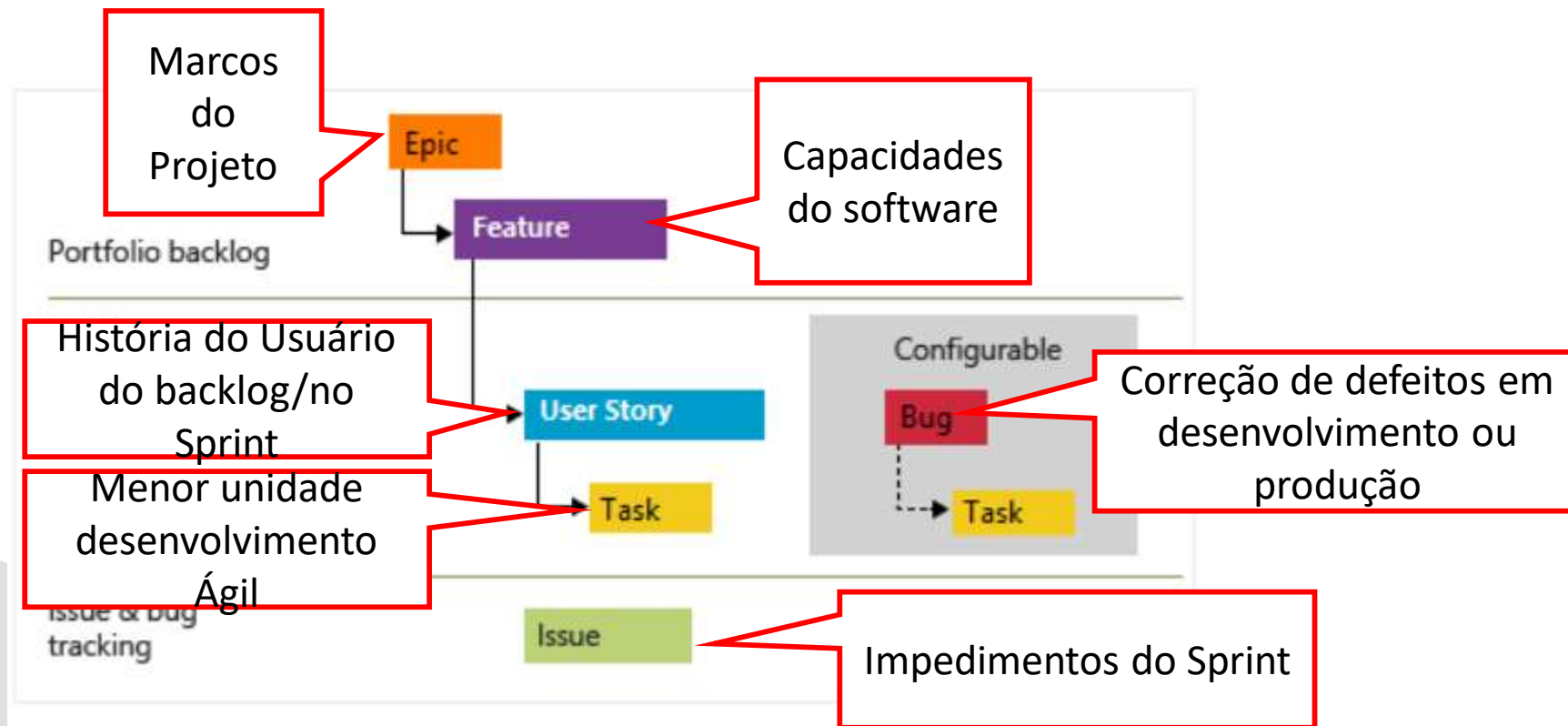
IGTI



# Azure DevOps - Agile

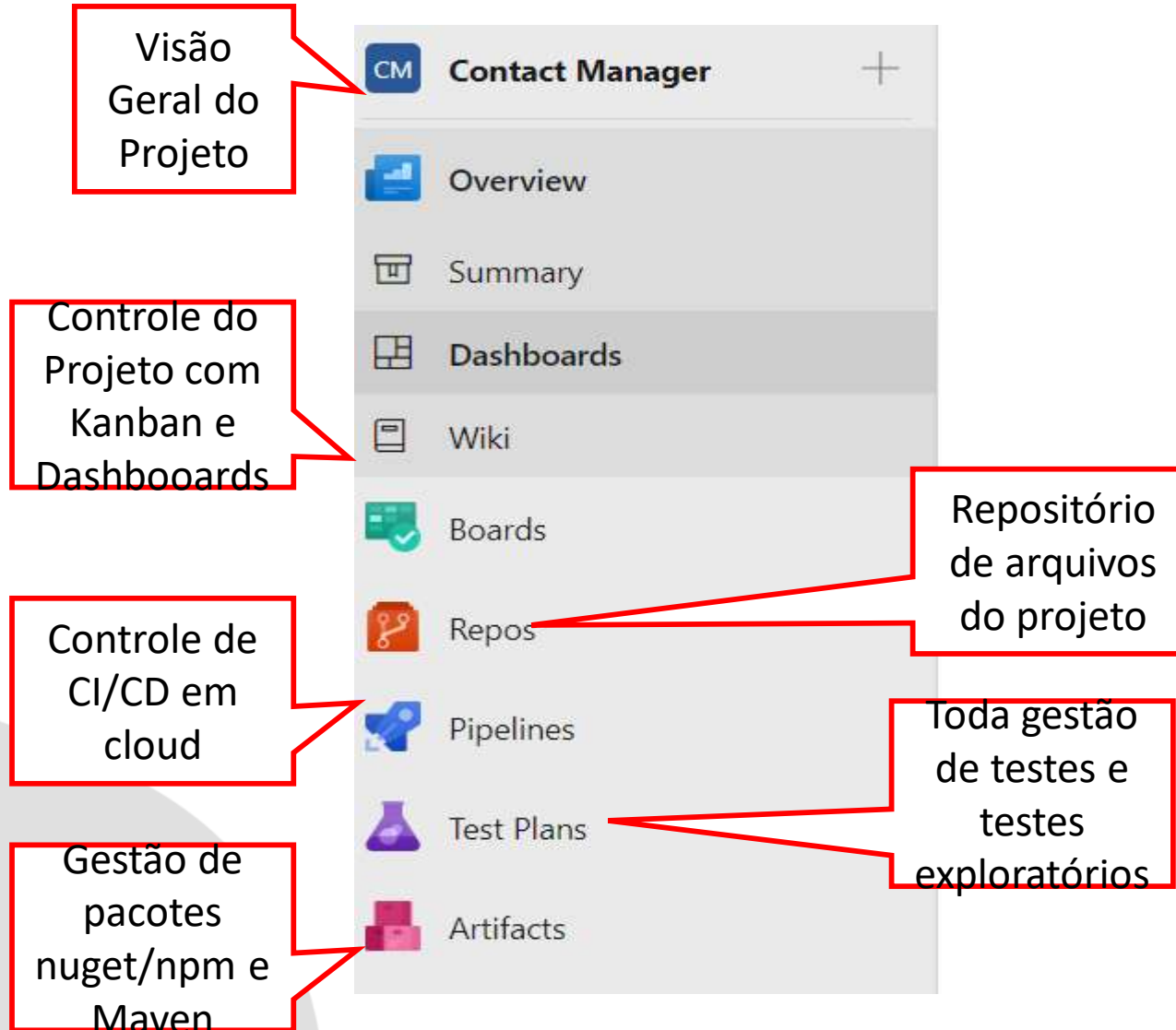


# Azure DevOps - Agile





# Azure DevOps



# Azure DevOps

IGTI

**promove**

Projects My work items My pull requests

Criação de  
Projetos

+ Create project

Filter projects

CM

**Contact Manager**

Projeto exemplo de gerenciamento de contatos.



PS

**ProMove Servicos**

Serviços da ProMove.



# Azure DevOps




Create new project

×

Project name \*


Description

Visibility



Public ⓘ

Anyone on the internet can view the project. Certain features like TFVC are not supported.



Private ⓘ

Only people you give access to will be able to view this project.

Public projects are disabled for your organization. You can turn on public visibility with [organization policies](#).

^ Advanced

Version control ⓘ

Git

▼

Work item process ⓘ

Agile

▼

# Azure DevOps – Pontos Importantes



- Processo, customização de campos e regras;
- Boards Configuração;
- Pipeline e Release;
- Dashboards, Capacidade (work details);
- Testes e SONAR;
- JOBS, rastreabilidade, PR, Work item;
- Retrospectiva (Market place).