

# ROTEIRO RODA ÁGIL

Time:

Data: \_\_/\_\_/\_\_

## 1. Pessoas sensacionais

### 1.1 - Colaboração e comunicação

O time percebe que a comunicação é frequente e transparente?

O time percebe que há interação e entrosamento entre todos, inclusive com o PO?

Todos conseguem dizer o que todos estão fazendo?

Todos se ajudam na resolução dos problemas e na remoção dos impedimentos?

Há evidências de que o time utiliza alguma ferramenta de feedback?

### 1.2 - Motivação e confiança

O time sabe responder qual é o propósito do seu trabalho/entrega?

Não há necessidade do time conferir constantemente o trabalho um do outro?

A taxa de ausência por afastamento/atestado é baixa?

Há evidências/depoimentos que comprovem o sentimento de trabalho em equipe?

A taxa de turnover é baixa dentro do time. Deve ser abaixo de 30%?

### 1.3 - Autonomia e auto-organização

Os membros do time se auto-organizam para determinar a melhor maneira de conseguir atingir o propósito do seu trabalho?

Há autonomia dentro do time para utilizar ferramentas para executar o seu trabalho da melhor forma?

O time consegue resolver os seus problemas ou direcionar as soluções? (comportamental)

O time tem autonomia sobre as soluções de arquitetura? (técnico)

A maioria dos impedimentos são removidos pelo próprio DevTeam e não somente pelo ScrumMaster?

### 1.4 - Kaizen

As reuniões de retrospectiva ocorrem regularmente?

O time consegue identificar oportunidades de melhoria sem dependência externa?

Problemas ou oportunidades de melhoria são encaminhadas ou resolvidas?

Há tempo para manter o fluxo de melhoria contínua?

INDICADOR: Quantos itens foram melhorados (p. ex.: processo, comunicação, arquitetura) no último semestre?

Escala: Não houve = 1; até 2 melhorias = 2; até 3 melhorias = 3; até 4 melhorias = 4; 5 ou mais melhorias = 5

### 1.5 - Interdisciplinaridade (conceito T-Shaped)

INDICADOR: Qual a taxa de interdisciplinaridade do time? (planilha aba Tx Interdisciplinaridade)

## 2. Experimente e aprenda rápido

### 2.1 - Práticas Lean-Agile

Possui gestão visual acessível?

Realizam todas as cerimônias ágeis que fazem sentido para o time?

O time limita e respeita o WIP?

O sistema é puxado? Os membros do time puxam as tasks conforme vão terminando ou é direcionado/nomeado?

O time atua prontamente nas falhas externas? (stop-the-line)

O time se considera completo para fazer todas as entregas necessárias com qualidade? (se falta algum papel de atuação)

### 2.2 - Métricas e ferramentas

Possui métricas (lead time, throughput, cycle time)?

Todos os membros do time conhecem as métricas?

Utiliza ferramentas para apurar as métricas?

O time tem tempo para acompanhar as métricas?

As decisões e o planejamento são com base nas métricas?

### 2.3 - Comprometimento com o produto (entregas)

O Product Owner (ou quem o representa) percebe que há qualidade nas entregas do time?

O Product Owner (ou quem o representa) percebe valor nas entregas do time?

O Scrum Master percebe que o time está comprometido com os prazos?

### 2.4 - Compartilhamento de conhecimento

INDICADOR: Quantidade de treinamentos realizados por membro do time. (planilha aba Tx Conhecimento)

O time tem tempo para promover momentos de troca de conhecimento entre si?

### 2.5 - Ritmo das Entregas

INDICADOR: Quantidade de entregas das versões no último semestre?

Escala: Não há padrão=1/Trimestral=2/Mensal=3/Semanal=4/Diário=5

## 3. Segurança um pré-requisito

### 3.1 - Código sustentável (refactoring)

Qual o percentual de melhoria da Fat, Tangle e McCabe?

Escala: 1 = Raramente (<= 5%); 2 = Esporádico (<=10%); 3 = Regular (<=20%); 4 = Bom (<= 30%); 5 = Ótimo (> 30%)

### 3.2 - Automações (pipeline)

Pipeline de build e Delivery (One Click Deploy não analógico e com Qualidade)

Há coluna de compilação no Pipeline?

Há coluna de testes unitários no pipeline?

Há coluna de métricas Understand/srccheck no Pipeline, gerando Histogramas e scatterplots?

Há coluna de deployar em um ambiente de testes automatizados?

Há coluna de testes via GUI do sistema nesse ambiente de testes?

Há coluna de testes de integração com diversas partes da solução?

Há coluna deployando os artefatos todos em produção ou pelo menos prontos para serem implantados em produção?

### 3.3 - Práticas BDD/TDD e Clean Code

Há evidência de Code Review com críticas de Clean Code?

Há evidências de testes BDD (GIVEN-WHEN-THEN = GWT) contra a camada de GUI?

Há evidências que o GWT é criado ANTES da codificação?

Há evidências de testes unitários escritos contra o código?

Há evidências que o teste unitário é criado ANTES da codificação?

### 3.4 - Qualidade E2E

INDICADOR: Volume de defeitos (evolução do backlog entre um semestre e outro)

Escala: 1 = >60% da capacidade; 2 = >40%; 3 = >25%; 4 = >10%; 5 = abaixo de 10% da capacidade)

### 3.5 - Práticas DevOps

Controle de versões

Ferramentas de build

Deployment das aplicações

Job de liberação de versões

Escala: 1 = Desconhece; 2 = Conhece; 3 = Pratica; 4 = Domina; 5 = Ensina

Checklist:

Há evidência de know-how de automatizações de rotinas? (Python, PowerShell ou similar)

O deploy é 100% digital, sem necessidade de intervenção humana?

A execução de testes automatizados é 100% digital, sem necessidade de intervenção humana?

A liberação de versão é 100% digital, sem necessidade de intervenção humana?

Há evidência de know-how de tecnologia de container como Docker/Linux ou Docker/Windows?

Há evidência de know-how Ansible ou similar?

Há know-how de setup de pipeline Jenkins ou GitLab?

Há know-how de análise de logs de build para identificar problemas ocorridos?

Há know-how local de rodar slaver Jenkins ou GitLab (runners)?

Há know-how local de setup e execução de testes via UI? (ex. SeleniumGrid/Hub no caso Web).

## 4. Valor a todo instante

### 4.1 - Quebra das Stories, Features e Epics

INDICADOR: Cycle time de histórias.

Escala: Não tem controle = 1; <= 10 = 2; <= 8 = 3; <= 5 = 4; <= 2 = 5)

### 4.2 - User eXperience

INDICADOR: Quantidade de salts abertas por tipo "orientação/esclarecimento".

Escala: Até 5% = 5; <= 10% = 4; <= 15% = 3; <= 20% = 2; > 20% = 1

### 4.3 - Entrega de valor ao cliente

INDICADOR: Quantidade de MVPs realizados no semestre.

Escala: 1= Não há ou <=20%; 2 = Baixo (<=40%); 3 = Regular (<=60%); 4 = Bom (<=80%); 5 = Ótimo (>80%)

#### 4.4 - Relacionamento com negócio

Há um PO dedicado para direcionar e garantir que a solução correta está sendo desenvolvida e entregue?

O Product Owner participa de todo o planejamento e todas as cerimônias que fazem sentido a sua presença?

O Product Backlog está organizado e priorizado? (grooming)

Há critérios de aceite (DoD) e estes são conferidos pelo Product Owner?

Há critérios de preparado (DoR) e estes são respeitados pelo Product Owner?

O time de desenvolvimento consegue sanar suas dúvidas quanto a solução (negócio), não comprometendo a entrega?

O time de desenvolvimento percebe valor nas entregas direcionadas pelo PO (ou por quem o representa)?

#### 4.5 - Satisfação do cliente

INDICADOR: NPS (Escala: <0 = 1; >0 e <=15 = 2; >15 e <= 30 = 3; >30 e <= 45 = 4; >45 = 5)

Checklist:

O time conhece a NPS dos principais clientes ou do segmento em que estão atuando?

O time consegue afirmar que conhece as necessidades dos diferentes perfis de usuários que utilizam o sistema? (personas)

O time conhece e está trabalhando nas principais reclamações e problemas enfrentados pelos clientes?

O time possui conhecimento de negócio suficiente para desenvolver soluções aderentes às necessidades dos clientes?

O time considera que o desempenho do produto está atendendo as expectativas do cliente?

O time conhece a percepção do cliente sobre o produto?

O time se relaciona eventualmente com o cliente? Já esteve no cliente?