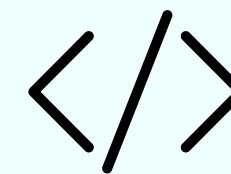
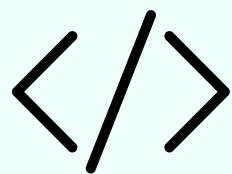


# 5 DESAFIOS NO DIA A DIA COMO QA



## 1. FALTA DE REQUISITOS PARA O SOFTWARE

Os requisitos orientam como o software deve funcionar, incluindo o que ele precisa realizar e as restrições que deve obedecer. Com a ausência, o processo de desenvolvimento e de teste fica prejudicado, pois as equipes não têm uma visão completa e padronizada do que deve ser entregue.

## 2. COMUNICAÇÃO COM O DEV TEAM

A comunicação eficaz entre Dev e QA é fundamental para o sucesso de um projeto. A ideia ultrapassada de que o QA serve apenas para apontar as falhas do Dev não se aplica hoje em dia. O QA trabalha em estreita colaboração com o Dev desde o início do projeto, fornecendo orientação e apoio contínuo. Juntos, eles podem entregar um projeto de alta qualidade, superando expectativas e garantindo excelência em todos os aspectos.

## 3. INSTABILIDADE NAS APLICAÇÕES

Quando a área de qualidade não é levada a sério por questões financeiras ou por não ter investimentos suficiente em ferramentas e ambientes de teste, frequentemente, ocorre instabilidade nas aplicações, impedindo que os testes sejam executados com qualidade.

## 4. FALTA DE PROCESSOS BEM DEFINIDOS

Processos bem estruturados ajudam a garantir transparência, facilitar tomadas de decisão e promovem colaboração entre todas as áreas no ciclo de desenvolvimento. Com a falta do mesmo, podem ocorrer desalinhamentos entre as equipes, atrasos e retrabalho, comprometendo a qualidade e previsibilidade do projeto.

## 5. PRAZOS APERTADOS E PRIORIZAÇÃO DE TAREFAS

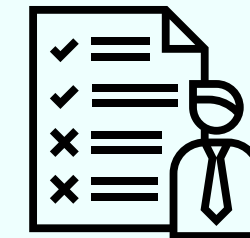
Equipes ágeis enfrentam prazos apertados e dificuldades na priorização de tarefas, comprometendo a qualidade das entregas. Para superar esses desafios, é essencial revisar o backlog, priorizar tarefas, analisar dados históricos e realizar reuniões frequentes de planejamento. Essas práticas melhoram a eficiência e atendem melhor às necessidades do cliente.

# DESAFIOS



# SOLUÇÃO

## 1. FALTA DE REQUISITOS PARA O SOFTWARE



Ao atuar em um ambiente sem requisitos formais, é essencial que o QA se comunique bem com os envolvidos no projeto, como desenvolvedores, POs e Analistas de Negócio, para facilitar o acesso a informações críticas do software. O uso e a análise de documentos auxiliares (como mapas mentais que deixam as regras conversadas em reuniões mais claras) ajuda a definir um escopo básico de testes.

Além disso, o uso de testes exploratórios permite identificar problemas com base na experiência e observação. A criação de checklists de funcionalidades básicas também oferece uma referência mínima para garantir a qualidade, cobrindo os pontos essenciais mesmo sem documentação detalhada.

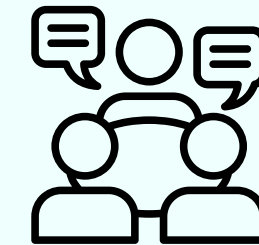
Porém, em um momento oportuno, explique os impactos negativos causados pela ausência de requisitos bem definidos, como:

- Aumento no tempo de execução do desenvolvimento e dos testes devido à falta de clareza;
- Retrabalho e correções após o desenvolvimento;
- Riscos de entregar um produto que não atende às necessidades do cliente ou usuário final.

Isso pode ajudar a equipe a ver o valor de uma documentação clara.

# SOLUÇÃO

## 2. COMUNICAÇÃO COM O DEV TEAM

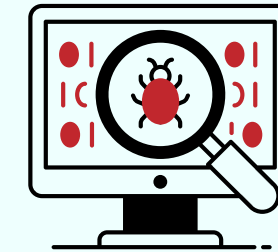


A comunicação fluida é essencial em uma squad para garantir que o produto seja entregue com o máximo nível de qualidade. Por isso, é fundamental que haja entendimento entre Q.A e Dev para alcançarem um objetivo comum.

- **Participação desde o início:** O Q.A deve estar envolvido desde o início do projeto (shift left) para entender os requisitos e identificar possíveis bugs.
- **Identificação de testes:** É importante definir o que precisa ser testado antes que o Dev comece a codificar (refinamento), ajudando assim nos testes unitários.
- **Comunicação empática:** Ao encontrar bugs, a comunicação deve ser empática e sem acusações, contribuindo para a melhoria contínua do produto.
- **Atividade nas reuniões:** Ser participativo e fazer questionamentos nas reuniões demonstra a importância da qualidade para o time e ajuda a ganhar a confiança de todos.
- **Métricas e documentação:** Levantar métricas e documentar o trabalho permite que os Devs vejam o impacto da qualidade, mostrando que o Q.A não está apenas testando, mas também contribuindo para o sucesso do projeto.

# SOLUÇÃO

## 3. INSTABILIDADE NAS APLICAÇÕES



Como a empresa não possui recursos financeiros para a área de qualidade de software, apresentamos abaixo um conjunto de soluções viáveis, com recursos limitados e de baixo custo financeiro.

### **Adotar ferramentas gratuitas de código aberto para:**

- Automação de Testes: Utilizar Selenium WebDriver ou Cypress para testes em aplicações web; Appium com TestNG para testes mobile; e Rest Assured para testes de APIs.
- Testes de APIs: Usar ferramentas como Postman para testar as APIs.
- Testes de Performance: Utilizar o JMeter para simulações de carga e desempenho.

**Criação e configuração de ambiente local com Docker:** Utilizar Docker para criar um ambiente de testes em máquina virtual e/ou local. Manter uma documentação detalhada e atualizada para que todos os QAs e desenvolvedores saibam configurar e executar esses ambientes.

**Monitoramento e alertas:** Configurar alertas para as aplicações com notificações em canais de comunicação para sinalizar falhas e erros nos sistemas.

**Cultura de Smoke Test para APIs críticas:** Implementar smoke tests nas APIs mais críticas e configurar alertas em canais de comunicação para sinalizar qualquer indisponibilidade.

**Integração com CI/CD utilizando ferramentas Open Source:** Implementar pipelines automatizadas com GitLab CI ou GitHub Actions para execução dos testes mais críticos para promover detecção rápida de falhas e maior estabilidade nas entregas.

**Promover uma cultura de qualidade:** Demonstrar e evidenciar como a instabilidade impacta negativamente a produtividade dos QAs, pois cenários básicos levam muito tempo para serem executados. Buscar o apoio da liderança para viabilizar a implementação das iniciativas de qualidade.

# SOLUÇÃO

## 4. FALTA DE PROCESSOS BEM DEFINIDOS



A falta de processos definidos pode afetar diretamente a eficiência dos trabalhos realizados pelas equipes, gerar atrasos, desperdícios, aumento de custos, retrabalhos, entre outros aspectos cruciais para um bom desenvolvimento. De modo geral, o QA enfrenta essa dificuldade muito pelo fato de não haver um processo consolidado, seja ele referente a qualidade ou no contexto geral.

Uma forma de mitigar esse tipo de problema é realizar a elaboração de um processo, caso não haja, ou disseminar a cultura de boa prática do processo existente.

- No primeiro caso, o ideal seria realizar um levantamento de etapas e afins que virão a compor o processo em si, definir cada etapa, papel e as responsabilidades e ações a serem efetuadas em cada um destes itens, criar um guia descritivo e intuitivo a respeito do processo de modo que todos os envolvidos possam ter acesso e entendê-lo e realizar treinamento e capacitação do processo à todos os envolvidos. Após isto, efetivamente adotar o processo elaborado, disseminando a boa prática, comunicação contínua e ativa e realizar retrospectivas para avaliar a aplicação do processo em cada sprint, ajustando o que não estiver funcionando.
- No caso de já existir um processo, podemos ser adotadas práticas similares ao passo anterior como: disseminar a boa prática, revisar se o processo ainda faz sentido no contexto atual, treinar e capacitar os envolvidos, além de também poder realizar levantamentos de pontos que podem ser melhorados.
- Em ambos os contextos o ideal é envolver todas as pessoas interessadas, construir algo que seja efetivo, consistente e valioso, sem ter exageros em etapas, responsabilidades ou afins. Desta forma, será possível criar ou seguir um processo capaz de transformar significativamente os padrões de produção e operação de uma empresa, assegurando a entrega de produtos e serviços de alta qualidade que atendam ou superem as expectativas dos clientes.



# SOLUÇÃO

## 5. PRAZOS APERTADOS E PRIORIZAÇÃO DE TAREFAS



Equipes ágeis enfrentam desafios com prazos apertados e priorização de tarefas, o que pode afetar a qualidade das entregas. Para contornar esses problemas, é fundamental adotar práticas que aumentem a eficiência e mantenham o foco nas funcionalidades de maior valor.

Estratégias:

- **Revisão do Backlog:**
  - Realizar revisões regulares do backlog para assegurar que as tarefas estejam atualizadas e alinhadas com os objetivos do negócio. Isso permite que a equipe se concentre nas funcionalidades mais críticas, refletindo o que realmente agrega valor ao cliente.
- **Priorização por Valor e Risco:**
  - Aplicar métodos de priorização, como a Matriz GUT e o MoSCoW, facilita a classificação das tarefas de acordo com seu impacto e urgência. Isso garante que as atividades mais importantes recebam atenção imediata, evitando a sobrecarga no backlog.
- **Ajuste com Dados Históricos:**
  - Analisar o desempenho anterior da equipe ajuda a estimar a capacidade de entrega e prever potenciais gargalos. Técnicas de análise preditiva transformam esses dados em insights valiosos, permitindo ajustes proativos no planejamento.
- **Reuniões de Refinamento:**
  - Agendar reuniões frequentes de planejamento e refinamento é vital para reavaliar prioridades e assegurar que o time esteja alinhado com os objetivos atuais do projeto. Isso ajuda a evitar o acúmulo de tarefas no backlog e garante que a equipe se concentre nas funcionalidades de maior impacto.

## GRUPO 7

CLEBER SOUZA - [LINKEDIN.COM/IN/CLEBER-AUGUSTO8/](https://www.linkedin.com/in/cleber-augusto8/)

JÚLIA ORSI - [LINKEDIN.COM/IN/JÚLIA-ORSI-93BB54235/](https://www.linkedin.com/in/júlia-orsi-93bb54235/)

LUCAS SANTANA - [LINKEDIN.COM/IN/LUCASANTANAS/](https://www.linkedin.com/in/lucasantanas/)

RAYRA BULCÃO - [LINKEDIN.COM/IN/RAYRABULCAO/](https://www.linkedin.com/in/rayrabulcao/)

RUAN OSÓRIO - [LINKEDIN.COM/IN/RUANOSORIO/](https://www.linkedin.com/in/ruanosorio/)

