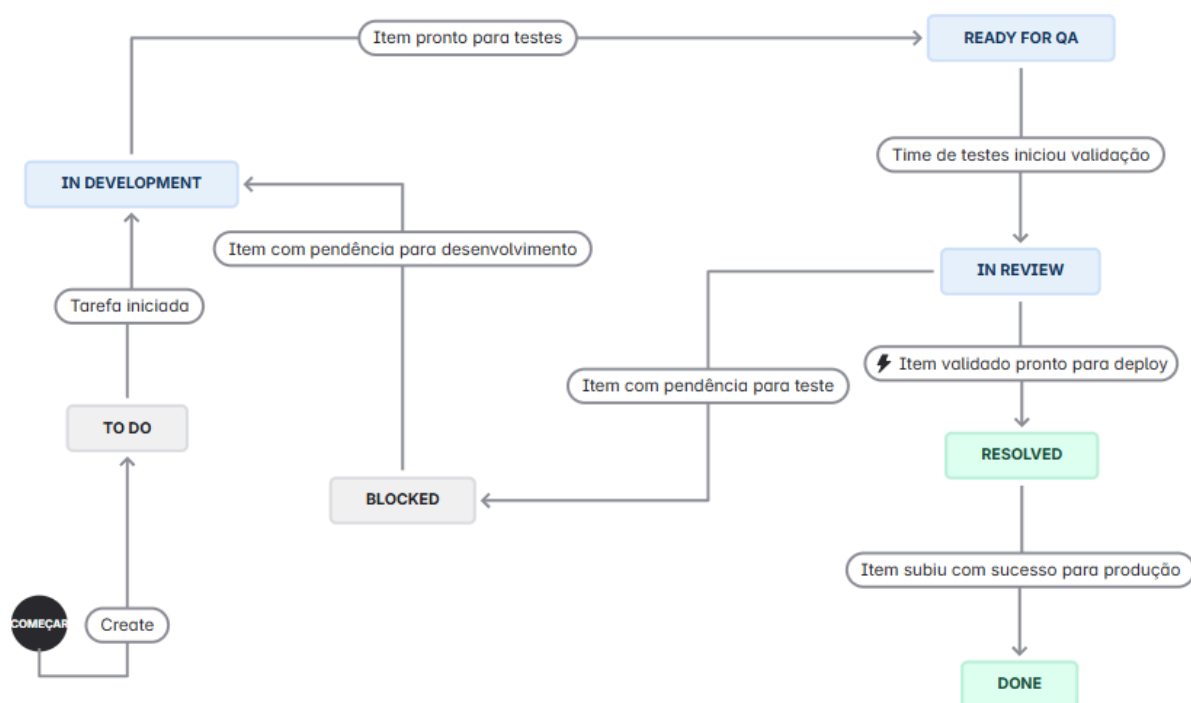


## Fluxo de Trabalho – Projeto Dia Dia do QA

O presente documento tem a finalidade de definir os fluxos de trabalho e ciclo de vida de bug em um projeto de desenvolvimento de software com processos de garantia de qualidade.

Os status serão aplicáveis a todos os tipos de itens disponíveis (tarefas, subtarefas, bugs e etc.) de modo a simplificar o fluxo.

### Gráfico do Fluxo



### Definições

#### 1. TODO

Tarefa inserida na sprint. Ainda não começou. Aguardando atribuição e início do desenvolvimento.

#### 2. BLOCKED

A tarefa não pode progredir devido a impedimentos (como dependências, falhas técnicas ou decisões pendentes). Deve ser priorizada na daily para remoção do bloqueio.

### 3. **IN DEVELOPMENT**

A tarefa foi assumida por um desenvolvedor e está em processo ativo de implementação. Inclui codificação, testes locais e commits parciais.

### 4. **READY FOR QA**

Desenvolvimento concluído. A tarefa está disponível para testes de QA. Deve cumprir os critérios de aceite definidos pelo time e documentação adequada.

### 5. **IN REVIEW**

A tarefa está sendo revisada — pode incluir revisão de código entre pares e validação de testes QA. Pode envolver testes manuais, testes exploratórios ou revisões automatizadas.

### 6. **RESOLVED**

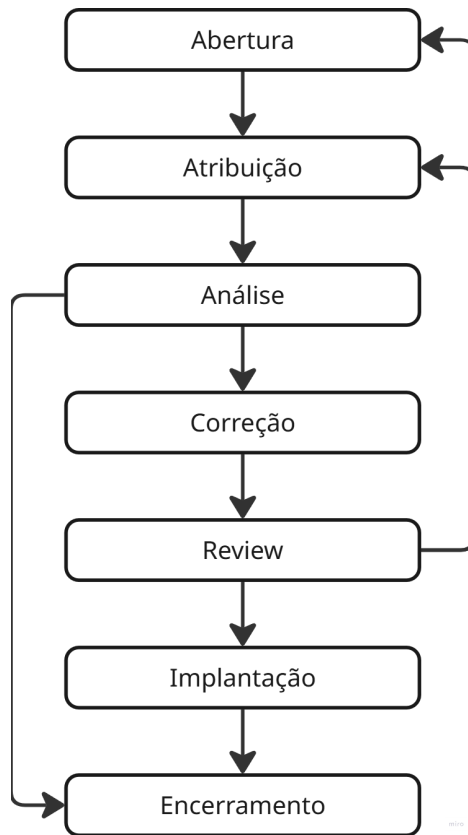
Item foi aprovado por QA. A tarefa está pronta para ser integrada à base final, aguardando deploy. Nenhum ajuste adicional foi necessário.

### 7. **DONE**

A entrega foi validada e subiu para ambiente de produção com sucesso. A tarefa está formalmente concluída e arquivada no ciclo da sprint.

# Ciclo de Vida de um Bug

## Gráfico:



## 1. Abertura

O problema é identificado e registrado em uma ferramenta de rastreamento de erros (como Jira, Azure DevOps, TestLink), incluindo detalhes como: descrição, passos para reprodução, evidências, ambiente afetado e possível impacto.

## 2. Atribuição

O bug é designado a um desenvolvedor ou equipe técnica para que possa ser investigado e corrigido. Também pode ser movido para uma sprint ou fila de correções, conforme processo interno.

### **3. Análise**

O responsável investiga a origem do erro e busca reproduzir o problema. O objetivo é entender a causa raiz antes de modificar qualquer parte do código. Caso se constate que não se trata necessariamente de um erro o bug pode ser encerrado.

### **6. Correção**

Com base na análise, o desenvolvedor realiza as alterações no código ou configurações do sistema para eliminar o bug, assegurando que a solução não afete outras funcionalidades.

### **7. Review**

A equipe de QA (ou o próprio desenvolvedor, dependendo do processo) testa a correção em ambientes de homologação para validar se o problema foi resolvido e se não surgiram efeitos colaterais.

### **8. Implantação**

Após validação, a correção é liberada para produção por meio do pipeline de deployment definido pela equipe, tornando a solução efetiva para todos os usuários.

### **9. Encerramento**

Com a correção verificada e implantada, o bug é oficialmente encerrado no sistema de rastreamento, podendo conter observações finais ou relatório de execução.