

1. FLUXO DE TRABALHO (DEVELOPMENT + QA + SCRUM)

1.1. Fluxo Geral do Processo no Scrum

1. Backlog do Produto

- O Product Owner cria, prioriza e mantém os épicos e User Stories.
- No contexto deste projeto, as histórias relacionadas são:
 - US01 – Login com credenciais válidas
 - US02 – Login com credenciais inválidas

2. Refinamento (Refinement)

- A equipe discute a funcionalidade.
- QA valida a testabilidade.
- Dev estima complexidade.
- Critérios de aceite são analisados.
- Riscos são identificados.

3. Sprint Planning

- A equipe escolhe quais User Stories entrarão na Sprint.
- É definida a meta da Sprint.
- A equipe utiliza Planning Poker para estimar esforço.
- QA estima:
 - Tempo para criar cenários
 - Tempo para criar casos de teste (step e BDD)
 - Tempo para executar testes

4. Desenvolvimento (Development)

- O Desenvolvedor implementa a funcionalidade.
- O QA acompanha critérios de aceite.
- É criado um Pull Request (PR).
- Revisões são feitas por Dev e QA.
- Depois do merge, a tarefa é movida para “Ready for QA”.

5. Testes (Quality Assurance)

- QA executa testes funcionais e de sistema.
- São validados:
 - Critérios de aceite
 - Cenários de teste
 - Casos de teste
- Evidências são registradas (prints ou vídeos).
- Bugs são criados se necessário.

6. Encerramento da Sprint

- É feita a Sprint Review, onde a funcionalidade é demonstrada.
- QA confirma que todos os requisitos foram atendidos.
- A Sprint Retrospective identifica melhorias para o próximo ciclo.

1.2. Fluxo de Trabalho no Jira (Workflow)

Fluxo Completo de Status

```
To Do → In Progress → Code Review → Ready for QA → In QA → Done
```

Descrição dos Status

To Do

Item já priorizado pelo PO e pronto para iniciar.

In Progress

O time de desenvolvimento está trabalhando na implementação.

Code Review

O código foi enviado via Pull Request e aguarda revisão por outro desenvolvedor ou técnico.

Ready for QA

O desenvolvimento foi concluído e a funcionalidade está pronta para testes.

In QA

O QA está executando testes funcionais, cenários e casos de teste.

Done

A funcionalidade foi testada, validada e atende aos critérios de aceite.

1.3. Ciclo de Vida do Bug (Bug Workflow)

```
New → To Do → In Progress → Ready for Retest → Retesting → Closed
```

Descrição dos Status do Bug

New

Bug relatado pelo QA pela primeira vez. Ainda não foi triado.

To Do

O bug foi aceito, priorizado e está na fila para correção.

In Progress

O desenvolvedor está realizando a correção.

Ready for Retest

O desenvolvedor entregou a correção. QA pode iniciar reteste.

Retesting

QA está testando novamente o comportamento corrigido.

Closed

O QA confirmou que o bug foi completamente resolvido. O bug é fechado.

Reopen (opcional, mas recomendado)

Usado quando o problema persiste após correção.

1.4. Regras e Boas Práticas do Ciclo de Bug

- Todo bug deve conter:
 - Título claro
 - Passos detalhados para reproduzir
 - Resultado atual
 - Resultado esperado
 - Evidências (print ou vídeo)

- Severidade
- Prioridade
- QA define **severidade**
- PO define **prioridade**

1.5. Resumo Geral do Fluxo

Sequência para User Stories

```
Backlog → Refinamento → Planning → Desenvolvimento → Testes → Done
```

Sequência para Work Items no Jira

```
To Do → In Progress → Code Review → Ready for QA → In QA → Done
```

Sequência para Bugs

```
New → To Do → In Progress → Ready for Retest → Retesting → Closed
```