

Metodologias Ágeis

QA Aula 7

**Tech Fixe
prof. Hérzio Pinto**

**Carmen Cabral
22/11/2023**

Índice

JIRA	3
Para que serve o JIRA?	3
Utilização do JIRA para os testers	3
Backlog	4
SCRUM	4
Conceitos-chave do Scrum	4
TIME SCRUM (<i>SCRUM TEAM/ST</i>)	4
PAPÉIS (<i>ROLES</i>)	4
<i>BACKLOG</i> DO PRODUTO	4
<i>BACKLOG</i> DO SPRINT	4
REUNIÕES DIÁRIAS (<i>DAILY MEETINGS</i>)	4
REVISÃO DE SPRINT	4
RETROSPECTIVA	4
KANBAN	5
Principais conceitos	5
QUADRO KANBAN	5
LIMITES DE TRABALHO EM PROGRESSO (<i>WIP/WORK IN PROGRESS</i>)	5
GESTÃO VISUAL	5
FLUXO CONTÍNUO	5
PRIORIZAÇÃO E FOCO NA QUALIDADE	5
FEEDBACK E MELHORIA CONTÍNUA	5
DIFERENÇAS ENTRE SCRUM E KANBAN	6
QUAL METODOLOGIA USAR: SCRUM, KANBAN OU HÍBRIDA?	6

JIRA

Ferramenta de software desenvolvida pela **Atlassian**, projetada para auxiliar no **gerenciamento de projetos, rastreamento de problemas e colaboração** em equipes de desenvolvimento de software.

JIRA se chamava “Gojira”, palavra japonesa para “Godzilla”, uma alusão à ideia de que a ferramenta pode lidar com problemas e projetos muito grandes de forma eficaz.

Para que serve o JIRA?

Rastreamento de problemas (*Issue Tracking*):

Podem incluir bugs, requisitos de software, tarefas de desenvolvimento, solicitações de recursos.

Gerenciamento de Projetos:

Fornece ferramentas para criar, gerenciar, acompanhar progresso, atribuir tarefas e definir prazos.

Metodologias Ágeis:

Suporta Scrum e Kanban, permite implementar quadros ágeis, gerenciar sprints e visualizar o progresso das tarefas.

Personalização:

Para atender às necessidades específicas de uma equipe ou projeto. Pode-se personalizar campos, fluxos de trabalho e relatórios.

Integração: com outras ferramentas de controle de versão, como o Git, ferramentas de automação de testes e de gestão de testes.

Colaboração: discussões, comentários e notificações para manter os membros da equipe atualizados sobre o progresso e as mudanças.

O **JIRA** é usado com outras ferramentas da **Atlassian**, como o **Confluence** – documentação e colaboração – e o **Bitbucket** – para controle de versão.

Utilização do JIRA para os testers

Criação de Casos de Teste:

Para documentar cenários de teste, requisitos de teste e critérios de aceitação.

Atribuição de Tarefas de Teste:

Pode atribuir a si mesmo ou a outros membros da equipe para execução.

Execução de Testes:

Atualizar o status dos casos de teste conforme sua execução, indicando se eles passaram, falharam ou estão em progresso.

Rastreamento de defeitos:

Registrar bugs e problemas com informações detalhadas, passos para reprodução e evidências.

Rastreamento de progresso:

Usar os quadros ágeis, como **Scrum** e **Kanban**, para acompanhar o progresso dos testes e tarefas de teste em uma **sprint** ou **ciclo de desenvolvimento**.

Associação de testes a *issues* (agrupamento de tarefas) de desenvolvimento:

Associar casos de teste a requisitos ou issues para rastrear a cobertura de testes.

Avaliação de qualidade – Como:

Usar relatórios, métricas, taxa de defeitos, cobertura de testes e progresso geral do teste.

Automatização de testes:

Integrar ferramentas de automação de testes, como o **Selenium** para rastrear testes e resultados.

Gestão de versões e Sprints:

Participar de reuniões de planejamento de Sprint e definir quais casos de teste serão incluídos em cada **Sprint**.

Revisões de Casos de teste:

Colaborar com outros membros para revisar e melhorar os casos de teste.

Documentação de teste:

Para documentar resultados de testes, observações e detalhes relacionados ao processo de teste.

Integração com ferramentas de gerenciamento de testes:

Como o Zephyr.

Backlog

Tem vários Epic e cada Epic tem várias User Stories.

SCRUM

Metodologia ágil para gerenciar projetos.

Abordagem **iterativa** e **incremental** para a gestão de projetos com foco na **colaboração, adaptabilidade** e **entrega contínua de valor** aos clientes.

Conceitos-chave do Scrum**TIME SCRUM (SCRUM TEAM/ST)**

autogerenciável, trabalha em ciclos de desenvolvimento curtos (**Sprints**) – com duração de 2 a 4 semanas.

PAPÉIS (ROLES)

Scrum Master/SM: responsável pelo prosseguimento do projeto; resolver os **impedimentos**; auxiliar a equipe.

Product Owner/PO: representa os interesses do **cliente**; determina as **prioridades do backlog do produto**.

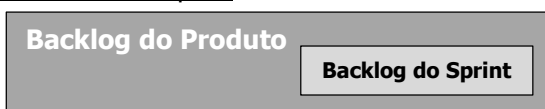
Equipe de desenvolvimento: responsável pelas **entregas** do trabalho executado durante a sprint.

BACKLOG DO PRODUTO

Lista **priorizada** dos **recursos, funcionalidades** e **melhorias** para o produto.

BACKLOG DO SPRINT

Antes de iniciar a **Sprint**, a equipe (ST) seleciona um conjunto de itens do **backlog do produto** que serão trabalhados durante o sprint.

**REUNIÕES DIÁRIAS (DAILY MEETINGS)**

Durante um **sprint**, a equipe realiza reuniões diárias curtas (*stand-up meetings*) para acompanhar o progresso, discutir impedimentos e planejar o trabalho para o próximo dia.

REVISÃO DE SPRINT

Após concluir um Sprint, a **equipe** realiza um **teste estático** para **detectar defeitos** ou para fornecer **melhorias**.

RETROSPECTIVA

Após concluir um Sprint, os **clientes** avaliam o processo e identificam melhorias.

Retrospectiva x Revisão

***Retrospectiva Sprint:** só o time scrum

***Sprint Review:** time scrum + **cliente** para apresentar o que foi feito na sprint.

https://geekbot.com/blog/sprint-review-vs-sprint-retrospective-the-critical-difference/?k_id=&adgroup_id=&campaign_name_ad=18393461182&qclid=CjwKCAiAvJarBhA1EiwAGgZl0DdjaoHhMA-PU4RmBbEcHn72ynsGaYw1mUTUg8otqeRCUjQuqbarnxoCAU8QAvD_BwE

Mesmos princípios: flexibilidade, colaboração e entrega incremental

Scrum: enfatiza a transparência, a inspeção e a adaptação contínua. Permite que as equipes respondam a mudanças nos requisitos do projeto e nas necessidades dos clientes de forma ágil.

Outras metodologias ágeis: Kanban, Extreme Programming (XP) e Lean.

SCRUM PROCESS

Product Owner/PO: 1 User stories; 2 Product Backlog;
3 Sprint Planning.

Scrum Master/SM: 4 Sprint Backlog + Equipe/ Team.

5 Implementation (*Sprint/Daily*) >> 6 Review >> 7 Retrospect

8 Finished work

KANBAN

Palavra de origem japonesa que significa “**cartão visual**” ou “**sinal**”. Originalmente desenvolvida pela Toyota no Japão como parte do sistema de produção **Lean**, mais foi adaptada em outros contextos, como o **desenvolvimento de software** e **gerenciamento de projetos**.

Principais conceitos**QUADRO KANBAN**

Quadro dividido em colunas que representam os estágios do fluxo de trabalho: ‘**a fazer/ to do**’, ‘**em andamento/ doing**’ e ‘**concluído/ done**’.

Os itens de trabalho são representados por cartões (ou **post-its**) que são movidos pelo quadro à medida que progredem de uma coluna para outra.

LIMITES DE TRABALHO EM PROGRESSO (WIP/WORK IN PROGRESS)

Limitar a quantidade de trabalho que pode estar em **andamento** simultaneamente em cada etapa do progresso. Evita sobrecarga e gargalos.

GESTÃO VISUAL

Para **acompanhar** o trabalho, identificar **gargalos** e garantir uma **compreensão** clara do que está acontecendo no projeto.

FLUXO CONTÍNUO

Retirada dos itens de trabalho do backlog de acordo com a capacidade, evitando pressões excessivas e atrasos.

PRIORIZAÇÃO E FOCO NA QUALIDADE

Entrega de itens de maior prioridade para evitar sobrecargas.

FEEDBACK E MELHORIA CONTÍNUA

Incentivo para coletar dados sobre o **desempenho** do processo. Alinhado com o princípio **Lean** de reduzir o desperdício e aumentar a eficiência.

Kanban:

*Altamente flexível **e pode ser adaptado a várias situações.**

***Mais liberdade para as equipes personalizarem seu processo de acordo com suas necessidades específicas.**

***Utilizado em situações em que os requisitos e prioridades mudam com frequência ou em equipes que já possuem um fluxo de trabalho estabelecido que desejam otimizar.**

Scrum:

***Prescritivo:** papéis e cerimônias definidos.

A FAZER / TO DO	FAZENDO / DOING	FEITO / DONE
Task 3	Task 2	Task 1

Stories	To do	In progress	Testing	Done
Story 1	Task 4	Task 3	Task 2	Task 1

To do	Up Next	In progress	In Review	Done
Task 5	Task 4	Task 3	Task 2	Task 1

DIFERENÇAS ENTRE SCRUM E KANBAN		
	Scrum	Kanban
Estrutura de processo	Mais rígida, com papéis bem definidos – Scrum Master, Product Owner, Equipe; cerimônias – reuniões diárias, revisões de sprint, retrospectivas; artefatos – backlog do produto, backlog do sprint.	Mais flexível, com papéis menos definidos e menos cerimônias prescritas. Foco principal: gestão visual e do fluxo de trabalho. Visual presencial e visual remoto (ferramentas tecnológicas, como o Trello, o Jira)
Limites de trabalho em progresso (WIP)	Não possui explicitamente, mas a equipe não deve pegar mais trabalho do que pode concluir durante um sprint.	Possui explicitamente.
Cerimônias	Possui várias: Reuniões diárias (dailys); Revisões; Retrospectivas.	Possui algumas. As reuniões de revisão e retrospectiva não são obrigatórias.
Papéis	Define papéis específicos: Scrum Master, Product Owner e Scrum Team	Não define papéis específicos, mas há funções necessárias, como líder de equipe ou proprietário do processo.
Priorização	Utiliza o backlog do sprint .	Individual de cada item do backlog do produto, conforme retirada para o trabalho.

QUAL METODOLOGIA USAR: SCRUM, KANBAN OU HÍBRIDA?

ABORDAGENS	SCRUM	KANBAN
Requisitos do projeto	Mais estáveis e podem ser planejados em sprints.	Variáveis e mudam com frequência.
Estrutura	Mais prescritiva com papéis definidos e cerimônias.	Mais flexível para lidar com interrupções e prioridades em constante mudança.
Equipe	Disposta a se comprometer com iterações fixas (sprints).	Já possui um fluxo de trabalho estabelecido que deseja otimizar.

ABORDAGEM HÍBRIDA

Usar elementos de ambas as metodologias para atender às suas necessidades.

Enfim, a escolha da abordagem deve ser baseada nas **características do projeto**, nas **preferências** da equipe e na **cultura organizacional**.

Tarefas

Assistir ao [vídeo](#). **Scrum** na prática 2022: Moxie – Especialistas em Jira

Assistir ao [minicurso](#). **Jira**: Canal Valor, com Marcelo Neves