**Análise do Kanban**

1. Iniciando a análise do quadro proposto no exercício podemos abordar cada coluna a seguir:
   1. **Backlog:** Podemos analisar que o backlog desse projeto é composto por 7 cards (tarefas/bugs) e que destes, temos 4 bugs, sinalizando que há algo de errado no sistema que está sendo desenvolvido. Nessa situação, eu como o profissional por estar a frente da qualidade preciso chamar atenção para o time(numa daily, por exemplo) sobre a importância de sempre revisarmos as funcionalidades antes de passar para coluna de testes as tarefas que estão sendo entregues, tomando os devidos cuidados com a qualidade e preciso rever a minha estratégia de testes para a continuidade do projeto. Obviamente que os pontos precisam ser anotados para serem abordados com mais calma em uma retrospective, mas não vejo como problema abordar esse ponto em uma daily;
   2. **Análise:** Nessa coluna, não há tarefas em análise. Um dos problemas seria devido a grande quantidade de backlog que já poderia ser iniciado a análise e o posterior desenvolvimento. Essa coluna pode indicar problemas de gestão de backlog pelo PO, falta de cuidado do time de desenvolvimento com os refinamentos, pouca mão de obra no time de dev ou simplesmente despriorização do backlog;
   3. **Desenvolvimento:** Na coluna de desenvolvimento, é possível visualizar 1 tarefa e 1 bug. Olhando para a quantidade de tarefas no backlog ou até mesmo para as tarefas em testes, fica claro a baixa operacional do time de desenvolvimento. Deve ser verificado o motivo do baixo número de tarefas da coluna, caso não seja por mão de obra, pode-se iniciar a análise de itens do backlog através da análise;
   4. **Testes:** Na coluna de testes, claramente vemos um gargalo. Partindo do princípio da priorização do Kanban seguindo fluxo da direita para esquerda, a coluna de testes é a que demanda a maior prioridade. Portanto, em situações que ocorram acumulo de itens, faz-se necessária a atenção de todo time e uma força tarefa para desafogar a coluna. Ou através de ajuda de outros analistas de qualidades vindos de outras equipes ou de os próprios desenvolvedores ajudarem nas validações com o apoio do QA;
   5. **Feito:** Ao olharmos a coluna e vermos apenas 1 item concluído, vemos que o projeto ainda entregou pouco e falando de entrega de valor, não é possível chegar a esse ponto por se tratar de apenas 1 item. Então cabe alinhar as expectativas de entrega com os interessados e focar nas demais etapas desenvolvimento para entrega dentro do prazo.
2. Soluções:
   1. Priorização de backlog pelo PO;
   2. Criação ou diminuir intervalos de agendas para refinamento de tarefas;
   3. Reduzir o gargalo na coluna de testes através de uma força tarefa entre desenvolvedores e QA’s;
   4. Se o caso, buscar apoio temporário de desenvolvedores de outras squads para desafogar atividades;
   5. Aumentar o foco na qualidade das entregas. Uma problemática que é comum dentro de um time de desenvolvimento, é de que nem todos desenvolvedores tem cuidado ao entregar suas atividades, não testando os mínimos cenários da funcionalidade em questão. Isso ocasiona muito retrabalho em um fluxo ágil, pois o QA precisa iniciar os testes, reportar bugs que muitas vezes são fáceis de se detectar até mesmo no fluxo de sucesso da funcionalidade. Que depois será recorrigido até ser entregue em produção.

**Papel do QA**

1. **Planning**: Nesse cerimônia, o QA pode participar estimando a entrega no momento da planning poker, levando em consideração a construção dos cenários de testes, além de avaliar a necessidade de automação de testes para cada feature. Vale citar que existem empresas que preferem não incluir o QA nessa estimativa por alguns motivos, porém creio que o processo de testes demanda um esforço considerável no ciclo ágil;
2. **Daily:** Pontuar impedimentos nos testes, como problemas de ambiente, cenários não passados, além de alinhar prioridades a serem testadas no fluxo ágil;
3. **Refinamento:** O QA participa no refinamento de negócios para compreender mais sobre o produto que será desenvolvido, através da apresentação da pessoa de negócios, tira dúvidas e levanta questões pertinentes ao funcionamento do software. Já no refinamento técnico, ele acompanha como será feito o desenvolvimento, tirando dúvidas sobre componentes, sobre retornos de backend, frontend além de auxiliar no fechamento de contrato entre backend e frontend que irão impactar nos testes integrados;
4. **Review:** Na Review, funciona como uma apresentação de entrega e o QA pode tomar a frente ou auxiliar na construção de slides ou vídeos para evidenciar as funcionalidades que agregam valor ao produto;
5. **Retrospectiva:** Na retro, o QA pode anotar todos os pontos de atenção/melhoria a respeito da qualidade e até mesmo dos processos que envolvem o time que foram observados durante o ciclo de sprint e buscar sempre que possível pontuar melhorias ou plano de ação. Vale também citar os pontos positivos e mencionar colegas que mais te auxiliaram no processo dos testes.