**Aluna: Mayara Rosa Omito**

**Professor: Fábio Araújo**

**Escola Britânica de Artes Criativas & Tecnologia**

**Curso: Engenharia de Qualidade de Software**

**Módulo 7: Fluxo de Trabalho Ágil**

**Exercício 01 – Análise do Kanban**

**Backlog - Problema**

Nesta primeira coluna, constata-se uma quantidade considerável de bugs, o que pode indicar um problema em relação à qualidade do código ou falta de testes adequados. Essa situação pode impactar negativamente a produtividade da equipe, que pode gastar mais tempo corrigindo bugs do que implementando novos recursos. Além disso, pode afetar a qualidade do produto final, pois erros podem aparecer e afetar a experiência do usuário.

**Backlog - Soluções**

**1.** Um passo que a equipe pode tomar é priorizar os bugs mais críticos e corrigi-los primeiro. Isso ajudará a reduzir o número de erros na sessão de retorno e permitirá que a equipe se concentre em outro trabalho a ser feito no futuro.

**2**. A equipe deve se certificar de que os bugs sejam descritos com clareza e detalhes suficientes para que que seja possível entender completamente o problema e iniciar-se o trabalho de correção. Se necessário, a equipe deve solicitar mais informações ou mais detalhes ao relator do bug.

**3.** É interessante que a equipe adote revisões periódicas para garantir que todos estejam na mesma página e que problemas importantes sejam resolvidos.

**4**. É importante que equipe se comunique regularmente sobre o status do bug, quem está trabalhando no quê e se houver alguma alteração nos cronogramas ou prioridades.

**5.** A equipe deve evitar se sobrecarregar e trabalhar com um número gerenciável de tarefas ao mesmo tempo.

**Análise – Problema**

O problema da coluna “Análise”, está no fato de que não há mais detalhes sobre a tarefa, o que pode gerar dúvidas na equipe e tornar o processo inconclusivo, causando reuniões extras para solucionar as dúvidas.

**Análise – Soluções**

**1.**Adotar descrições claras e conversas detalhadas entre os membros da equipe para discutir e esclarecer dúvidas e confirmar os detalhes antes de adicioná-los ao quadro Kanban pode garantir que as tarefas sejam bem definidas e compreendidas por todos os membros da equipe, evitando mal-entendidos e confusões. É importante que equipe identifique e documente todas as tarefas antes de incluí-las no quadro de Kanban, para evitar que detalhes importantes fiquem de fora.

**Desenvolvimento - Problema**

Bugs e tarefas aqui dividem a mesma sessão, o primeiro trata-se de um problema a ser corrigido, enquanto o segundo é um novo recurso ou melhoria que precisa ser implementada. Ambos têm objetivos diferentes e exigem habilidades e recursos diferentes. Esse quadro pode deixar a equipe confusa, que por sua vez, pode acabar priorizando uma ação em detrimento de outra, levando ao aumento de bugs.

**Desenvolvimento – Soluções**

**1.** É importante dividir a tarefa e o bug em diferentes sessões, removendo-os para os seus respectivos lugares. Dessa forma, é garantido que cada elemento seja tratado como uma entidade separada e tenha sua própria sessão de desenvolvimento.

**2**.Se o bug for urgente e precisar ser corrigido imediatamente, é possível reagendar e priorizar a tarefa. Dessa forma, a equipe pode ter certeza de que o erro será corrigido rapidamente e evitará possíveis problemas.

**3.** Caso o erro for grande o suficiente para ser corrigido em vários partes, é possível criar uma tarefa separada para cada um destes passos. Dessa forma, a equipe pode acompanhar o andamento de cada etapa e evitar confusão com a tarefa original.

**4.** Priorizar a etapa de anotação pode ser uma solução imprescindível para conhecer mais a respeito do processo de desenvolvimento, e consequentemente, a equipe terá mais discernimento ao distribuir os elementos nas colunas corretas em projetos futuros.

**Testes – Problema**

Na coluna de testes, é possível perceber que o problema apontado no item anteriormente citado (sessão de desenvolvimento), aqui, é ampliado, sendo assim, e equipe levará mais tempo para separar bugs de testes, sendo provável que gastem de duas a três reuniões extras (a depender do tempo disponível e estabelecido) para tratar do mesmo problema.

**Testes – Solução**

Todas as soluções sugeridas para a coluna de desenvolvimento podem ser aplicadas no caso da sessão de testes.

**Feito - Problema**

A sessão “Feito” no quadro de Kanban é a etapa final, portanto, é imprescindível que a tarefa esteja destrinchada de modo claro, permitindo o vislumbre de todo o esforço que a equipe empregou no projeto, já que esta falta de clareza pode impedir a identificação de possíveis problemas no processo de desenvolvimento.

**Feito – Solução**

**1.**Mais uma vez, registrar todas as tarefas no processo de desenvolvimento é importante para que nenhum detalhe seja esquecido, e para que se evite a falta de clareza quanto ao item apresentado, é interessante que a informação seja destrinchada em vários post-its, sendo cada tarefa individual representada por um único card.

**Exercício 02 – O Papel do QA (Quality Assurance) em cada evento ágil**

**Planning:** O QA pode ajudar a equipe de desenvolvimento a determinar os critérios de aceitação para cada história de usuário e definir os testes necessários para verificar o trabalho realizado.

**Daily:** Na Daily, o QA pode informar a equipe sobre o status dos testes em andamento, compartilhar os resultados dos testes concluídos e relatar problemas ou impedimentos que afetam a qualidade do software.

**Review:** Pode apresentar os resultados dos testes realizados durante a sprint e destacar quaisquer problemas ou problemas de qualidade encontrados.

**Retrospectiva:** Pode compartilhar ideias de melhorias para os processos de teste e qualidade, identificar áreas que precisam de upgrade e trabalhar com a equipe para desenvolver um plano de ação para a próxima sprint.