Exercício 1:

1. Quais os problemas você enxerga neste quadro?

Liste pelo menos um problema por coluna (backlog, análise, desenvolvimento, testes, feito)

Backlog: Muitas tarefas estão nesta coluna e também bugs estão misturados com a tarefa, o que pode confundir ou dificultar mensurar a entrega das sprints por serem itens diferentes.

Análise: Nenhuma tarefa está em análise, ou seja, as que estão em backlog ainda não foram analisadas ou retornadas a esta coluna.

Desenvolvimento: Somente uma tarefa e bug está em desenvolvimento, o que pode ser falta de planejamento ou as tarefas estão sendo desenvolvidas muito rápidas e indo para teste diretamente sem antes ter sido realizado testes simples ou testes unitários.

Testes: De todas tarefas e bugs, somente um foi movido para Feito, o que pode significar que o restante ainda está passando por testes ou poderá voltar para desenvolvimento.

Feito: Se estiver no final ou metade da sprint, somente uma tarefa realizada sinaliza que algo de errado está acontecendo na sprint, podendo ser desde o planejamento até a execução.

2. Quais as possíveis soluções? Escreva pelo menos 5 soluções.

A primeira a ser feita seria separar em quadros separados os bugs e as tarefas, pois assim não ficará misturado itens que têm propósitos diferentes, um corrigir eventuais problemas/falhas e outros entregar valor com novas melhorias ou projetos. Deixando tudo num quadro só ao fazer uma avaliação da sprint não haverá exatidão no quanto que foi entregue de valor e correção de melhorias.

A segunda seria deixar itens de backlog em um projeto separado, pois neste projeto podemos criar mais colunas como documentação em desenvolvimento, análise técnica (para saber se é viável fazer tecnicamente), em espera, caso esta item dependa de mais informações de terceiros e uma de descartado, para ter-se controle sobre itens que foram descartados e seus motivos, assim todos envolvidos neste projeto podem ter acesso fácil a esta informação.

Se esta sprint estiver no início ou metade, podemos ver que há poucas tarefas/bugs a serem desenvolvidos, portanto após uma análise do motivo de somente dois itens estarem nesta etapa, pode-se pensar em fazer mais análises dos itens do backlog para iniciar a sprint com mais itens, pois a equipe pode estar mais madura e entregando com mais rapidez suas tarefas.

Adicionar mais uma coluna para fazer code review, ou seja, os desenvolvedores mais experientes avaliam as tarefas as de ir para testes, com o intuito de avaliar se o código manteve o padrão do sistema, se não acarretará em problemas quando subir em produção e o que pode ser melhorado como performance de execução.

Como podemos ver, a coluna de testes contém muitos itens, esta fila pode ser diminuída ou agilizada se for automatizado tarefas frequentes executadas pelos usuários, como login no sistema, criação/edição/exclusão de alguma informação etc.

Exercício 2: Papel do QA

1. Qual o papel do QA em cada evento ágil ( Planning, Daily, Review, Retrospectiva, Refinamento)?

Descreva como o QA pode ajudar em cada evento deste.

Planning: divisão das suas tarefas, planejamento dos testes, criação de cenários de testes, baseado nos critérios de aceitação, fazer o levantamento de dados para os testes (massa de dados) e divisão de testes automatizados.

Daily: Falar sobre os cenários de teste que elaborou para as tarefas da sprint, quais bugs validou e quais são mais prioritários, falar sobre as tarefas a nível de teste de aceitação e usabilidade que testou e se já tiver feito, quais testes automatizados já fez.

Review: Provar ao PO que todos os critérios de aceitação das histórias foram concluídos com sucesso

Retrospectiva: O QA teve tempo de testar todas tarefas ou ao menos as mais importantes e mais críticas, assim sendo essenciais de testes? Os testes automatizados funcionaram? Todos critérios de aceitação foram concluídos? Se não, quais não foram e por quê? Os testes de regressão foram realizados com sucesso?

Refinamento: Definir o que deve acontecer em ambos caminhos, o feliz e o alternativo. O que deve acontecer se tal informação não for informada. Quais campos nas tarefas devem ser obrigatórios ou qual fluxo deve acontecer para chegar-se ao caminho feliz. Se chegar no caminho alternativo, o que deve acontecer?

Qualquer usuário terá acesso para realizar as ações das tarefas definidas? Ou apenas algumas?

Se o sistema estiver fora ou feature for interrompida em alguma etapa, o que deve acontecer?

A integração já está terminada e entregue para iniciarmos esta tarefa e consequentemente os testes?