**Casos de Uso x User Stories**

Para descrição das funcionalidade para descrever todas as regras e o que se espera dos PBI’s pode ser utilizado os métodos Casos de Uso e User Stories.

**Casos de uso (use case)**

Descrevem as funcionalidades do sistema de software a partir da perspectiva do usuário. Eles representam as interações entre o usuário e o sistema.

**Desafios**

* Podem ser complexos e difíceis de entender.
  + São mais formais.
* Podem ser difíceis de manter atualizados.
* Podem não ser representativos das necessidades reais dos usuários.

**Características**

* Possuem maior rastreabilidade
* Descrevem o que o sistema faz, não como ele faz.
* São focados nas funcionalidades do sistema.
* São utilizados para documentar os requisitos funcionais do sistema.

**Objetivos**

* Descrever as funcionalidades do sistema de forma clara e concisa.
* Facilitar a comunicação entre os stakeholders.
* Servir como base para o desenvolvimento do sistema.

A UML (Unified Modeling Language) pode ser utilizada na descrição de casos de uso. A UML fornece uma notação gráfica para representar os elementos de um caso de uso, como ator principal, fluxo principal, fluxos alternativos e pós-condições.

**Vantagens de usar UML na descrição de casos de uso**

* Facilita a visualização e compreensão das funcionalidades do sistema.
* Permite a identificação de pontos de falha e oportunidades de melhoria.
* Facilita a comunicação entre os stakeholders.

**Elementos da UML utilizados na descrição de casos de uso**

* **Diagramas de Caso de Uso:** Representam os casos de uso e as interações entre os atores e o sistema.
* **Diagramas de Sequência:** Representam a ordem das mensagens trocadas entre os objetos do sistema durante a execução de um caso de uso.
* **Diagramas de Atividade:** Representam o fluxo de atividades dentro de um caso de uso.

**User Stories**

Descrevem as necessidades do usuário de forma breve e concisa. Elas são escritas na perspectiva do usuário e utilizam linguagem natural.

Descrição contando uma história do que se precisa da funcionalidade execute.

**Desafios**

* Podem ser ambíguas e difíceis de interpretar.
* Podem não conter detalhes suficientes para a implementação.
* Podem ser difíceis de priorizar.

**Características**

* Simples
* Fácil entendimento
* Incentivam conversa
* Descrevem o que o sistema faz, não como ele faz.
* São focados nas funcionalidades do sistema.
* São utilizados para documentar os requisitos funcionais do sistema.

**Objetivos**

* Capturar as necessidades dos usuários de forma clara e concisa.
* Facilitar a comunicação entre a equipe de desenvolvimento e os usuários.
* Servir como base para o planejamento e desenvolvimento do sistema.

**Qual dos dois utilizar?**

Casos de Uso e User Stories são duas técnicas importantes para documentar os requisitos de um sistema de software. A escolha da técnica mais adequada dependerá do contexto do projeto e das necessidades dos stakeholders.

A combinação de Casos de Uso e User Stories pode ser uma estratégia eficaz para documentar os requisitos de um sistema de software. Os Casos de Uso podem ser utilizados para descrever as funcionalidades do sistema de forma geral, enquanto as User Stories podem ser utilizadas para detalhar as necessidades dos usuários.

Porém user stories são melhores para entendimento devido a forma natural e de conto da história.

**Como especificar os itens PBI’s**

* Protótipos navegáveis.
* Criar documento de requisitos com users stories com descrição de como a funcionalidade deve funcionar
* alguns diagramas para descrever por exemplo uma lista de status e as variações

**Como deve ser criado as user stories**

Uma boa User Story deve ser clara, concisa e completa, e deve conter os seguintes pontos primordiais:

**Estrutura**

* Formato: "Como [usuário], eu quero [funcionalidade] para [benefício]."
* Exemplo: "Como cliente, eu quero poder pesquisar produtos por nome para encontrar o que procuro rapidamente."

**Foco no Usuário**

A User Story deve ser escrita na perspectiva do usuário. Use linguagem clara e concisa que o usuário possa entender.

**Valor para o Usuário**

A User Story deve descrever o que o usuário precisa e por que isso é importante para ele. O benefício deve ser claro e tangível para o usuário.

**Independência**

Cada User Story deve ser independente e autocontida. Evite dependências entre as User Stories.

**Estimação**

Forneça uma estimativa de esforço ou tamanho para cada User Story. Isso ajudará a priorizar e planejar o desenvolvimento.

**Pontos Adicionais**

* Critérios de Aceitação: Defina as condições que devem ser atendidas para que a User Story seja considerada concluída.
* Prioridade: Determine a prioridade da User Story em relação às outras User Stories.
* Detalhes: Adicione detalhes adicionais, como requisitos funcionais e não funcionais, se necessário.

**Recomendações**

Inicie com uma frase simples e clara que descreva o que o usuário precisa. Evite jargões técnicos e linguagem complexa. Envolva os usuários na criação das User Stories. Revise e atualize as User Stories regularmente.

**Exemplos de User Stories**

**Exemplo:**

Como cliente, eu quero poder pesquisar produtos por nome para encontrar o que procuro rapidamente.

**Critérios de Aceitação**

O sistema deve permitir que o cliente insira um nome na barra de pesquisa.

O sistema deve exibir uma lista de produtos que atendem ao critério de pesquisa.

A lista de produtos deve ser ordenada por relevância.

**Estimativa**

3 dias

**Prioridade**

Alta

**Detalhes**

O sistema deve ter uma barra de pesquisa na página inicial.

A barra de pesquisa deve ser autocompletar com sugestões de produtos.

A lista de produtos deve mostrar o nome, a imagem e o preço do produto.

**Benefício**

O cliente poderá encontrar o que procura de forma rápida e fácil.

**Qual deve ser o nível de detalhamento do product backlog**

Para os PBI’s que estejam na próxima sprint para execução as user stories devem estar extremamente detalhadas, para melhor entendimento de execução da equipe de desenvolvimento.

As próximas sprints podem ter algumas sprints porém não podem constar muito detalhadas pois pode ser que as demandas das sprints futuras podem sofrem alterações.

**Quando podemos dizer que um PBI está “Ready”?**

Quer dizer que o item da PBI está pronto para ser apresentado para a equipe de desenvolvimento, que a especificação (user stories) está concluída e completa.

**Boa prática do PO no detalhamento**

O PO deve estar 2 sprints à frente do time:

* Enquanto o time trabalha na sprint atual (01), o PO está franularizando os requerimentos da próxima sprint (02)
* E se sobrar tempo, detalhar a outra sprint (03) em requerimentos mais generalista (menos granularidades)