**História de usuário** **(*User Stories*)**

Curso Capacitação em Engenharia de Software

Sexta feira 03.03.2023

Em desenvolvimento de software e gerenciamento de produto, uma **História de usuário** (*User Stories*) é uma especificação de uma ou mais sentenças na linguagem de negócio ou cotidiana do usuário final ou usuário do sistema que captura o que um usuário faz ou necessita fazer como parte de sua função de trabalho. Histórias de usuário são usadas com metodologias ágeis de desenvolvimento de software como a base para definir o escopo de um projeto de software. É uma técnica de análise de requisitos. Ela captura o "quem", "o quê" e "por quê" de um requisito em uma forma concisa e simples, geralmente limitada em detalhes, de forma que possa ser escrita à mão em um pequeno cartão de notas de papel.

**História**

As Histórias de usuário originaram-se com a Extreme Programming (XP), cuja primeira descrição por escrito, em 1998, apenas alegou que os clientes definiam o escopo do projeto "com Histórias de usuários, que são como os casos de uso". Ao invés de ser oferecidas como uma prática distinta, elas foram descritas como uma das "peças do jogo" usadas no jogo de planejamento. No entanto, a maior parte da literatura impulsiona em torno de todas as maneiras, argumentando que as Histórias do usuário são "diferentes" dos casos de uso, na tentativa de responder de uma forma mais prática "como os requisitos são tratados" em XP e mais geralmente projetos ágeis. Isso conduz o surgimento, ao longo dos anos, de uma explicação mais sofisticada de Histórias de usuários.

Em 2001, Ron Jeffries propôs a fórmula bem conhecida dos Três Cês, ou seja, *Card* (cartão), *Conversation* (conversa), *Confirmation* (confirmação), para verificar a qualidade de uma Histórias de usuário:[[4]](https://pt.wikipedia.org/wiki/Hist%C3%B3ria_de_usu%C3%A1rio#cite_note-4)

* Um *Cartão* (ou muitas vezes uma nota de Post-it) é um símbolo físico dando forma tangível e durável para o que, de outra forma, seria apenas uma abstração. A ideia do cartão é garantir que a descrição da História seja sucinta, objetiva, direta.
* Uma *Conversa* está ocorrendo em tempo e lugares diferentes durante um projeto entre as várias partes interessadas relacionadas pelo recurso dado (clientes, usuários, desenvolvedores, testadores, etc.), o que é em grande parte verbal, mas na maioria das vezes complementado pela documentação;
* A *Confirmação*, o mais formal o melhor, garante que os objetivos que giravam em torno da conversa foram finalmente alcançados.

**Criando Histórias de usuário**

Histórias de usuário são escritas por ou para usuários ou clientes de negócio como uma maneira primária para influenciar a funcionalidade do sistema sendo desenvolvido. Histórias de usuário podem também ser escritas por desenvolvedores para expressar requisitos não funcionais (segurança, desempenho, qualidade, etc.), embora primariamente seja a tarefa de um gerente de produto garantir que as Histórias de usuário sejam capturadas.

Quando chega a hora de criar Histórias de usuário, um dos desenvolvedores se reúne com um representante do cliente, e.g. um gerente de produto (ou dono do produto em Scrum), que possui a responsabilidade de formular as Histórias de usuário. O desenvolvedor pode usar uma série de perguntas para obter a meta do representante do cliente, como perguntar sobre a conveniência de alguma funcionalidade em particular, mas deve tomar cuidado para não dominar o processo de criação da ideia.

Quando o representante do cliente concebe uma Histórias de usuário, ela é escrita em um cartão de nota (e.g. de 8x13 cm) com um nome e uma breve descrição. Se o desenvolvedor e o representante do cliente encontrar uma Histórias de usuário deficiente de alguma forma (muito grande, complicada, ou imprecisa), ela é reescrita até ficar satisfatória - muitas vezes utilizando as diretrizes INVEST. Comumente, as Histórias de usuário não são definitivas, uma vez que elas foram escritas e que os requisitos tendem a mudar ao longo do ciclo de desenvolvimento, os quais os processos ágeis manipulam não os tornando imutáveis.