

A modelação com casos de uso é uma técnica de modelação textual e visual, que hoje está associada à linguagem UML. Esta tornou-se num método muito útil e muito importante nas primeiras etapas dos processos de desenvolvimento de sistemas, tanto nos processos mais clássicos como nos mais atuais. Estes modelos valerão uma preciosa ajuda ao responsável por desenvolver o sistema, no sentido em que esse desenvolvimento será guiado precisamente por eles, já que descrevem todas as possibilidades de interação que o sistema terá com o utilizador final, conseguindo assim identificar os requisitos funcionais que serão necessários garantir para o correto funcionamento do sistema.

Numa sociedade em constante mudança, processos clássicos de desenvolvimento de sistemas, como o waterfall, demonstraram ser processos de desenvolvimento arriscados, já que uma mudança no sistema ou uma correção de erros terá custos monetários tanto mais elevados quanto mais longe das primeiras etapas estes forem descobertos. Assim, é imperativo utilizar processos de desenvolvimento ágeis e modernos em que caso necessitem de mudanças a meio do processo o custo dessa mudança será bastante reduzido, comparativamente aos processos clássicos.

Foi no sentido de demonstrar que é possível aplicar casos de uso comuns a métodos de desenvolvimento ágeis e evolutivos que surgiu o Use-Case 2.0. O Use-Case 2.0 é precisamente uma técnica escalonada de utilização de casos de uso para o desenvolvimento ágil de requisitos. Assim, ao invés de ser um novo processo, é apenas uma técnica ágil, que pode ser implementada em qualquer processo.

O Use-Case 2.0 rege-se por 6 princípios que objetivamente pretendem manter toda a informação o mais simples possível sem perder em vista a ideia principal e sem perder o foco na utilidade de cada parte do sistema. Pretendem também dividir os casos de uso em fatias compostas por histórias de casos de uso, realizar entregas funcionais recorrentes ao promotor para que este possa acompanhar o desenvolvimento e possa apontar “erros” o mais cedo possível, e fazer sempre tudo isto no ambiente e da forma que for mais confortável à equipa.

Relativamente às fatias mencionadas, estas devem ser tão “pequenas” que possam ser implementadas, por exemplo, num sprint da metodologia Scrum. Cada fatia deverá conter uma história de caso de uso, ou mais caso descrevam situações familiares, e tem de ter no mínimo um caso para teste. Se existirem fatias com as mesmas histórias só serão válidas se tiverem casos de teste diferentes. As histórias também têm de ser claras para garantir que podem ser usadas em planeamento ágil.

É este o conceito geral, cortar em fatias. Se uma fatia se demonstrar demasiado grande então é cortada novamente. O objetivo será sempre trabalhar com peças o mais simples possíveis,

para que a sua ligação final seja mais fácil de realizar, sem perder pormenores, garantindo que todos os requisitos funcionais são identificados a tempo e que são devidamente implementados.

O facto de um caso de uso demasiado grande para ser abordado em apenas um sprint de Scrum poder ser dividido em fatias para agilizar os sprints e de, como já referido, o conceito conseguir encaixar perfeitamente tanto em processos ágeis como clássicos são vantagens deste método de aplicação de casos de uso.

Concluindo o Use-Case 2.0 é uma técnica bastante apropriada de aplicar os casos de uso comuns aos processos de desenvolvimento evolutivo, uma vez que simplificam o caso de uso, permite aplicar as suas partes progressivamente, mantendo as entregas funcionais recorrentes que se exigem num desenvolvimento evolutivo, e, graças a isso, permite identificar os problemas que possam existir a tempo de minimizar os custos ao promotor.