

# Spiral Development

O desenvolvimento em espiral é um método de desenvolvimento iterativo que tem como o objetivo minimizar problemas e fazer o desenvolvimento do projeto da melhor maneira possível.

Desenvolver casos de uso de uma vez é difícil e torna mais caro a incorporação de nova informação adquirida posteriormente neles. Desenvolver um caso de uso desta maneira causa também demoras na detecção de fatores de risco no projeto.

Isto acontece devido ao tempo necessário para compreender e entender o funcionamento de um sistema. Pode levar algum tempo até que o developer entenda o que o cliente quer com toda a clareza.

Atrasos são caros e ao examinar os requisitos é provável que algo tenha que ser alterado pelo que ao criar o caso de uso inteiramente de uma vez é mais comum existirem erros nos requisitos. Ao existirem erros estes vão criar atrasos no desenvolvimento e ser uma despesa adicional que poderia ter sido evitada.

No spiral development o método utilizado é criando iterações com cada iteração sendo progressivamente mais precisa e exacta com o que se quer obter do caso de uso.

Ao desenvolver iterativamente é possível obter um melhor entendimento do projeto pois a grande quantidade de informação que é necessário utilizar para criar o caso de uma vez toda ao mesmo tempo pode causar confusão. Com este melhor entendimento e desenvolvimento gradual são minimizados os erros nos requisitos.

O spiral development funciona em conjunto com BreadthBeforeDepth.

Utilizado o BreadthBeforeDepth deve listar-se os atores e os objetivos e trabalhar com essa lista inicialmente. Com a lista deve criar-se um plano de projeto, uma estimativa do trabalho a realizar e priorizar os casos de uso mais importantes usando isto para organizar as equipas de desenvolvimento.

Continuando com o BreadthBeforeDepth deve escolher-se uma parte específica do projeto como por exemplo um caso de uso e aprofundar e expandir essa área. Depois de trabalhada a parte específica deve ser revisto o projeto em geral e os outros casos de uso.

Se for necessário utilizar os métodos MergeDroplets e CleanHouse ou outros para melhorar a estrutura e fazer as modificações necessárias às listas de atores e objetivos.

Deste modo pode ir-se do geral ao pormenor e com o que se aprendeu a trabalhar nos detalhes fazer modificações ao geral.

Devemos parar o desenvolvimento quando tivermos a certeza que os casos de uso são o que os necessários e satisfazem as necessidades dos clientes.