

Padrões de desenvolvimento: a equipa

Padrão	Problema	Solução
SmallWritingTeam	Usar muitas pessoas para escrever um caso de uso é ineficiente, e os compromissos feitos para ajustar os diferentes pontos de vista, podem resultar num sistema menos satisfatório.	Restringir o número de pessoas que trabalham num produto para duas ou três. Usar o processo do padrão TwoTierReview para incluir mais pessoas.
ParticipatingAudience	Não se pode satisfazer as necessidades dos <i>stakeholders</i> sem a sua participação e contribuição.	Sempre que possível envolver activamente clientes e <i>stakeholders</i> internos no processo de desenvolvimento de casos de uso.
BalancedTeam	Equipas com indivíduos que têm a mesma maneira de pensar na forma de desenvolver casos de uso, pode resultar num conjunto de casos de uso estritamente limitados que não satisfazem as necessidades.	Formar a equipa com pessoas de diferentes especialidades de forma a incentivar os interesses dos <i>stakeholders</i> no processo de desenvolvimento. A equipa deve ter programadores e utilizadores finais.

Padrões de Desenvolvimento: O Processo

Padrão	Problema	Solução
BreadthBeforeDepth	Não haverá progressos no tempo, nem serão criados de casos de uso coerentes se for gasta a energia a escrever casos de uso muito detalhados.	Direccionar as energias do desenvolvimento de casos de uso para uma visão global e adicionar progressivamente mais detalhes.
SpiralDevelopment	Desenvolver casos de uso num único passo é difícil, impede a incorporação de novos dados, podendo atrasar a descoberta de factores de risco.	Desenvolver os casos de uso num processo iterativo e em cada iteração aumentar progressivamente a precisão e a objectividade dos casos de uso.
MultipleForms	Projectos diferentes necessitam de diferentes formalidades durante a sua execução. Pessoas diferentes têm preferências diferentes para o desenvolvimento. Assim, é contra-producente tentar que todas as pessoas usem o mesmo método de trabalho.	Seleccionar um formato de desenvolvimento baseado nos riscos do projecto e nas preferências das pessoas envolvidas.
TwoTierReview	Muitas pessoas podem necessitar de rever os casos de uso, tornando este processo caro e demorado.	Assegurar dois tipos de revisões: a primeira efectuada pela equipa a nível interno e repetida várias vezes. A segunda deverá ser realizada pelo grupo completo, podendo fazer só uma única revisão.
QuittingTime	Desenvolver o modelo de casos de uso para além das necessidades dos <i>stakeholders</i> , consome recursos e atrasa o projecto.	Desenvolver os casos de uso até ficarem completos e satisfazerem adequadamente as necessidades da audiência.
WritersLicence	Uma ênfase excessiva no estilo dificulta a escrita dos casos de uso.	Pequenas diferenças no estilo de escrita são inevitáveis, mas cada caso de uso deve passar nos seguintes testes: deve seguir o modelo e o estilo básico da organização; estar logicamente correcto; ser legível para os avaliadores finais; ser suficientemente preciso para ser implementado.

Padrões de Estrutura: Conjunto de casos de uso

Padrão	Problema	Solução
SharedClearVision	A falta de uma visão clara sobre o sistema pode conduzir a indecisões, a opiniões contraditórias entre os <i>stakeholders</i> . Pode ainda paralisar o projecto.	Identificar claramente os objectivos do sistema de acordo com a missão da organização. Assegurar que uma visão clara dos objectivos seja partilhada por todas as pessoas envolvidas no projecto. Incluir também nesta visão os problemas que o sistema poderá resolver e os problemas que o sistema poderá não resolver; quem são os <i>stakeholders</i> ; e como o sistema beneficiará os <i>stakeholders</i> .
VisibleBoundary	A extensão de um sistema crescerá de uma maneira incontrolável se não se souber quais são as suas fronteiras.	Estabelecer uma fronteira visível entre o sistema e o seu ambiente, enumerando quem e o que pode interagir com o sistema.
ClearCastOfCharacters	Analizando os utilizadores do sistema sem se atender aos seus papéis, poder-se-á perder comportamento importante do sistema ou introduzir comportamento redundante.	Identificar os actores que interagem com o sistema e o papel que cada um deles representa. Descrever claramente os actores.
UserValuedTransactions	Um sistema é imperfeito se não puder entregar os serviços que são importantes aos seus utilizadores e se não apoiar os propósitos e os objectivos específicos do sistema.	Identificar os serviços importantes que o sistema deve entregar aos actores para satisfazer os objectivos do negócio. Cada caso de uso deve descrever os serviços essenciais que são importantes para pelo menos um utilizador.
EverUnfoldingStory	O número de passos necessários para a descrição do comportamento do sistema excede tanto a memória como o interesse dos vários tipos de leitores.	Organizar o conjunto de casos de uso como uma história hierárquica de maneira que possa ser desdobrada para adquirir mais detalhe ou compactada para adquirir mais contexto.

Padrões de Estrutura: Casos de uso

Padrão	Problema	Solução
CompleteSingleGoal	Objectivos inadequados provocarão incerteza nos escritores acerca de onde um caso de uso termina e o outro começa.	Escrever cada caso de uso com um objectivo completo e bem definido. Seleccionar e atribuir um nome ao objectivo do actor principal.
VerbPhraseName	Nomes genéricos sem significado não vão de encontro às expectativas do leitor nem fornecem um ponto de referência adequado.	Atribuir um nome ao caso de uso com um verbo forte numa frase representando o objectivo do actor principal.
ScenarioPlusFragments	O leitor deve ser capaz de seguir com facilidade o caminho pelo cenário ou história, de outra forma pode ser um trabalho inútil ou mesmo perder informação.	Escrever a história como um cenário simples sem considerar possíveis falhas. Depois, colocar fragmentos de história que mostrem as condições alternativas que poderão ocorrer.
ExhaustiveAlternatives	Um caso de uso pode ter várias alternativas. A falta de algumas vai provocar enganos quanto ao comportamento do sistema por parte de quem está a desenvolver.	Identificar todas as alternativas e falhas que devem ser geridas pelo caso de uso.
Adornments	A inclusão de requisitos não funcionais num caso de uso pode rapidamente desordenar e obscurecer esse mesmo caso de uso.	Criar campos adicionais no <i>template</i> dos casos de uso, fora do texto que descreve o cenário, contendo informação suplementar que é útil associar aos casos de uso.
PreciseAndReadable	Casos de uso demasiado complicados para leitores não especializados, ou demasiado imprecisos para quem desenvolve, resultam em sistemas inadequados e mal construídos.	Escrever o caso de uso de forma a ser acessível à audiência menos técnica, com poder de decisão, para que esta o possa avaliar. Também deve ser suficientemente preciso para que quem o vai desenvolver compreenda o que vai construir.

Padrões de Estrutura: Cenários e Passos

Padrão	Problema	Solução
DetectableConditions	Os escritores debatem-se com o problema de saber quantas e quais as condições a incluir no caso de uso.	Incluir somente as soluções detectáveis. Juntar as condições que têm o mesmo efeito no sistema.
LeveledSteps	Caso de uso com passos excessivamente grandes ou excessivamente pequenos não ajudam a atingir o objectivo final tornando o caso de uso difícil de ler e de compreender.	Cada cenário deve ter de três a nove passos; Os passos desses cenários devem situar-se em níveis semelhantes e a um nível de abstracção justamente abaixo do objectivo do caso de uso.
ActorIntentAccomplished	Tanto o leitor como o escritor podem ficar confusos acerca do comportamento de um sistema se não estiver claro qual é o actor que tem a responsabilidade para executar determinado passo.	Escrever cada passo para mostrar claramente, qual a acção que o actor vai desempenhar.
ForwardProgress	Os escritores têm que decidir o comportamento a colocar num único passo. Podem-no fazer escrevendo muito detalhadamente, tornando os casos de uso muito longos e cansativos de ler.	Eliminar ou juntar os passos que não deixam o actor avançar. Simplificar passagens que possam distrair os leitores deste progresso.
TechnologyNeutral	A inclusão de restrições tecnológicas e de detalhes de implementação numa descrição de um caso de uso aumenta a sua complexidade e obscurece o seu objectivo.	Escrever cada caso de uso de uma forma que seja independente da tecnologia.