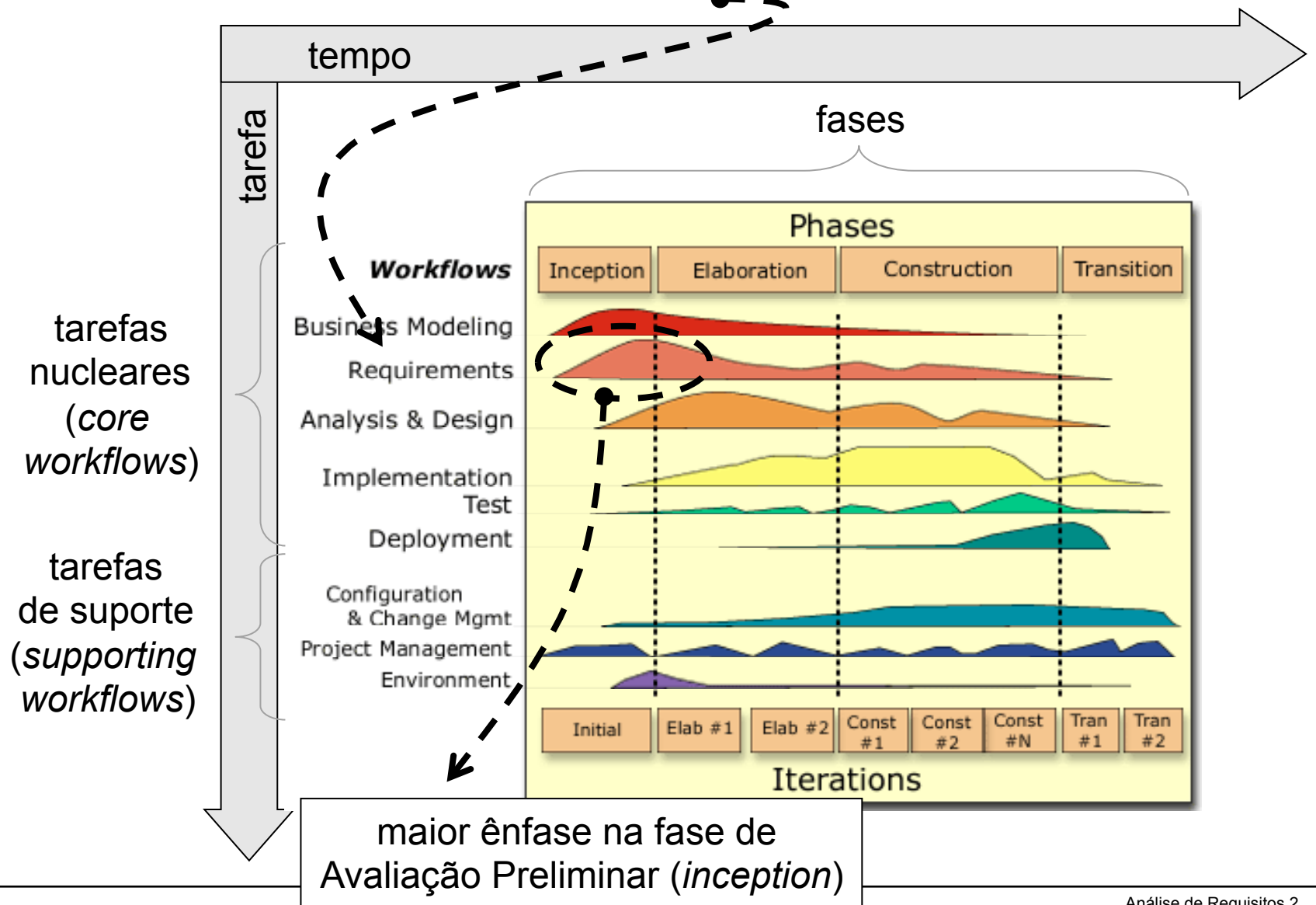


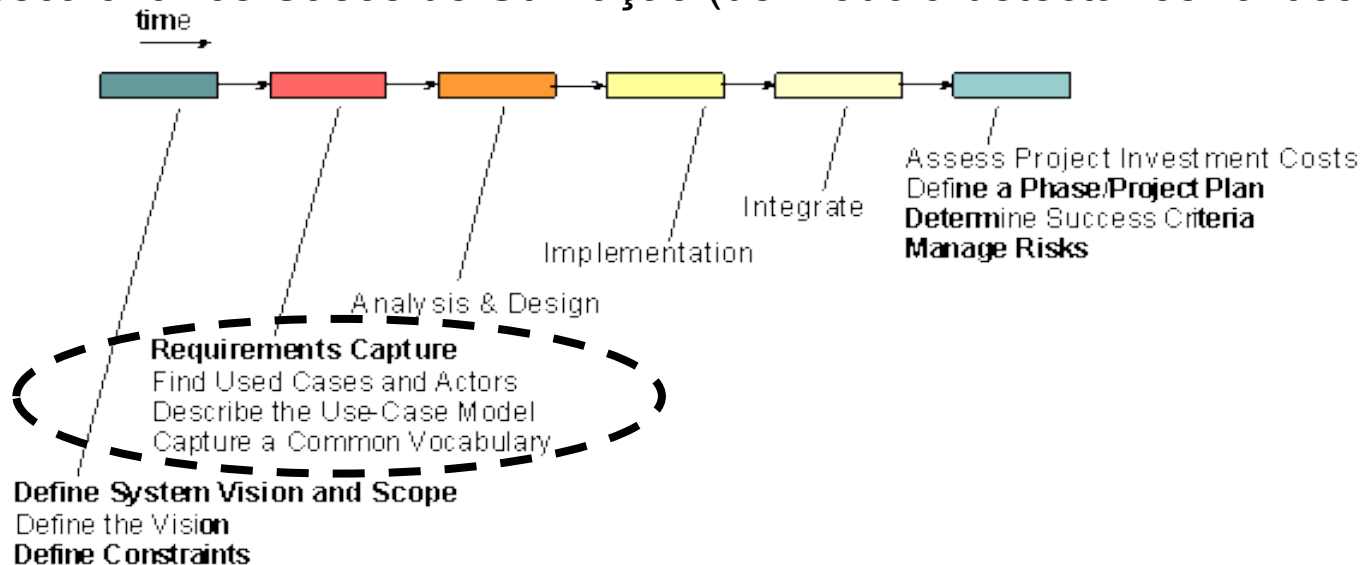
Análise de Requisitos

Análise de Requisitos – onde aparece ?



Alguns objectivos da fase de Avaliação Preliminar

- Partindo dos casos de utilização críticos (5 a 10% do total)
 - efectuar as tarefas – análise, desenho, implementação
 - com o objectivo de encontrar e avaliar viabilidade de uma arquitectura
- Para isso é necessário
 - capturar os requisitos do sistema
 - organizar esses requisitos em Casos de Utilização (“Use Cases”)
 - escalonar os Casos de Utilização (de modo a detectar os "críticos")



Requisitos

- O que são ?
 - São a descrição das necessidades ou propósitos de um produto
 - Quais os artefactos a construir ?
 - síntese de objectivos (*overview statement*)
 - clientes (*customers*)
 - metas a alcançar (*goals*)
 - funções do sistema (*system functions*)
 - atributos do sistema (*system attributes*)
- caracterização geral
- caracterização pormenor

in Criag Larman; Applying UML and Patterns

Um exemplo

- Considere-se o seguinte texto, que
 - exhibe os requisitos de uma aplicação para uma biblioteca
 - é escrito na perspectiva do cliente (quem irá "pagar" o sistema)

O departamento cultural da W-REAL (WeREadALot) – empresa de tradução, tem necessidade de uma aplicação informática que suporte alguns dos processos da biblioteca de que é responsável.

A biblioteca empresta livros e revistas aos requisitantes que estão registados na biblioteca. Os bibliotecários são responsáveis por esta tarefa.

A biblioteca (o seu gestor) é responsável pela aquisição de novos títulos. Dos títulos mais procurados são adquiridos diversos exemplares. Os exemplares antigos são removidos (pelo gestor) quando estão degradados.

Um requisitante pode reservar um título que não esteja no momento disponível. Assim que um exemplar desse título estiver disponível essa pessoa será notificada. A reserva é cancelada quando o requisitante leva o exemplar, ou se após decorrida 1 semana ele não for requisitado.

O catalogador é quem recepciona encomendas e cataloga títulos recepcionados.

Artefactos – caracterização geral

- Síntese de objectivos (*overview statement*)

Neste projecto serão automatizados os processo de empréstimo e reserva de livros e revistas. O sistema desenvolvido será utilizado numa biblioteca.

- Clientes (*customers*)

Departamento cultural da W-REAL (WeREadALot) – empresa de tradução.

- Metas a alcançar (*goals*)

Pretende-se que este sistema contribua para:

- responder "assim que possível" a quem solicita títulos não disponíveis
- obter informação que suporte as decisões de aquisição de cópias
- garantir aos utilizadores respostas imediatas e correctas

Artefactos – caracterização pormenor

- Funções do sistema (*system functions*)
 - representam aquilo que o sistema é suposto fazer
- Para verificar que determinado *X* é de facto uma função do sistema,
 - esse *X* deve aparecer com sentido na seguinte frase:

O sistema tem que fazer <X>

- "O sistema tem que fazer *actualização de existências após empréstimo*"
 - "O sistema tem que fazer *registo de reserva de títulos não disponíveis*"
 - "O sistema tem que fazer *notificações por e-mail*"
- Atributos do sistema (*system attributes*)
 - representam qualidades não-funcionais do sistema
 - ... pelo que essas qualidades não aparecem com sentido nas frases:
 - "O sistema tem que fazer *simples de utilizar*" !!
 - "O sistema tem que fazer *independente do sistema operativo*" !!

Funções do sistema – categorias

- As funções de sistema devem ser categorizadas, de modo a,
 - definir prioridades
 - identificar as que, sendo evidentes, consomem recursos (tempo, ...)

Categoria	Significado
Evidente	Tem que ser realizada. O utilizador tem que ter conhecimento da sua realização.
Invisível	Tem que ser realizada. Não é visível para os utilizadores.
Adorno	Opcional. Não afecta significativamente o custo ou outras funções.

- Observações: categoria Invisível
 - exemplo – "registar dados de modo persistente"
 - funções desta categoria são, por vezes, incorrectamente esquecidas ...

Funções do sistema – exemplo

As funções do sistema podem ser agrupadas.
O agrupamento de funções ajuda a manter as "ideias organizadas".

Funções básicas

Ref. #	Função	Categoria
R1.1	Registrar empréstimos (exemplar e utilizador)	Evidente
R1.2	Registrar reservas (título e utilizador)	Evidente
R1.3	Actualizar inventário após empréstimo	Invisível
R1.4	Listar títulos encomendados e ainda não recebidos	Evidente
R1.5	Recepcionar encomenda de exemplares	Evidente
R1.6	Actualizar inventário após recepção de encomenda	Invisível
R1.7	Eliminar registo de exemplares em mau estado	Evidente
R1.8	Cancelar reserva de títulos	Evidente
R1.9	Listar existências	Evidente
R1.10	Garantir persistência e integridade dos dados	Invisível
R1.11	Autenticar utilizador (nome e senha)	Invisível
R1.12	Catalogar existências	Evidente

Funções de notificação

Ref. #	Função	Categoria
R2.1	Notificar utilizador da disponibilidade de exemplar	Evidente
R2.2	Manter fila de prioridade de reservas	Invisível
R2.3	Notificar utilizador para devolução de exemplar	Evidente
R2.4	Configurar antecedência da notificação	Adorno
R2.5	Elaborar carta de notificação	Evidente

Atributos do sistema

- Os atributos do sistema
 - são características ou dimensões do sistema
 - ... não são funções
- Alguns atributos do sistema
 - facilidade de utilização (*ease of use*)
 - interação homem-máquina (*interface metaphor*)
 - plataformas (*platforms*)
 - tolerância a falhas (*fault tolerance*)
 - tempo de resposta (*response time*)
 - custo do projeto (*retail cost*)

Atributos do sistema – domínios de variação

- Podem variar num conjunto de valores de detalhe (*attribute details*)
 - usualmente valores discretos e simbólicos
- Exemplo de "valores de detalhe"
 - tempo de resposta $\in \{ \text{apropriado, longo} \}$
 - interacção homem-máquina $\in \{ \text{gráfica, colorida, forms-based} \}$
- Podem ter de restrições de fronteira (*attribute boundary constraints*)
 - usualmente um intervalo de valores numéricos
- Exemplo de "restrições de fronteira"
 - tempo de resposta $\subseteq [\text{máximo de 5 segundos}]$
 - custo do projecto $\subseteq [5000 \text{ .. } 5500] \text{ Euros}$

Atributos do sistema – categorias e exemplo

- Duas categorias se podem definir para os atributos do sistema

Categoria	Significado
Obrigatório	Tem que ser contemplado. É usual ser uma Restrição de Fronteira
Desejável	Deve estar preparado para alcançar.

- Exemplo

Atributo	Detalhe / Restrição de Fronteira	Categoria
interacção Homem-Máquina	Detalhe Formulários (<i>forms</i>) e caixas de diálogo	Obrigatório
	Detalhe Ênfase na utilização de teclado (em vez do rato)	Desejável
plataformas	Detalhe Windows NT e Linux	Obrigatório

Atributos e Funções do sistema

- É importante evidenciar
 - as relações entre Funções do Sistema e Atributos do Sistema
- Descrever atributos relacionados com funções específicas

Ref. #	Função	Categoria	Atributo	Detalhe / Restrição de Fronteira	Categoria
R1.5	Recepcionar encomenda de exemplares	Evidente	interacção Homem-Máquina	Detalhe Ênfase na utilização de teclado (em vez do rato)	Obrigatório

Casos de Utilização

- O que são ?
 - *"... narrative documents that describes the sequence of events of an actor (an external agent) using a system to complete a process"*
 - *"... they are stories of cases of using a system"*
- O que não são ?
 - *"... they are not exactly requirements or functional specifications,*
 - *but they illustrate and imply requirements in the stories they tell"*

in Ivar Jacobson; Object Oriented Software Engineering. A Use Case Driven Approach

Casos de Utilização e Processos do Domínio

- Um Caso de Utilização descreve um processo
 - como por exemplo um "processo de negócio"
- Um processo descreve, do início até ao fim, a
 - sequência de eventos, ações, ou transações
 - ... necessárias para produzir ou completar algo de valor para
 - ... um ator ou organização
- Exemplos de processos
 - encomenda de um produto
 - levantamento de dinheiro numa máquina ATM
 - matrícula de alunos em cursos numa escola
 - correção ortográfica de um documento num processador de texto
 - atendimento de uma chamada telefónica

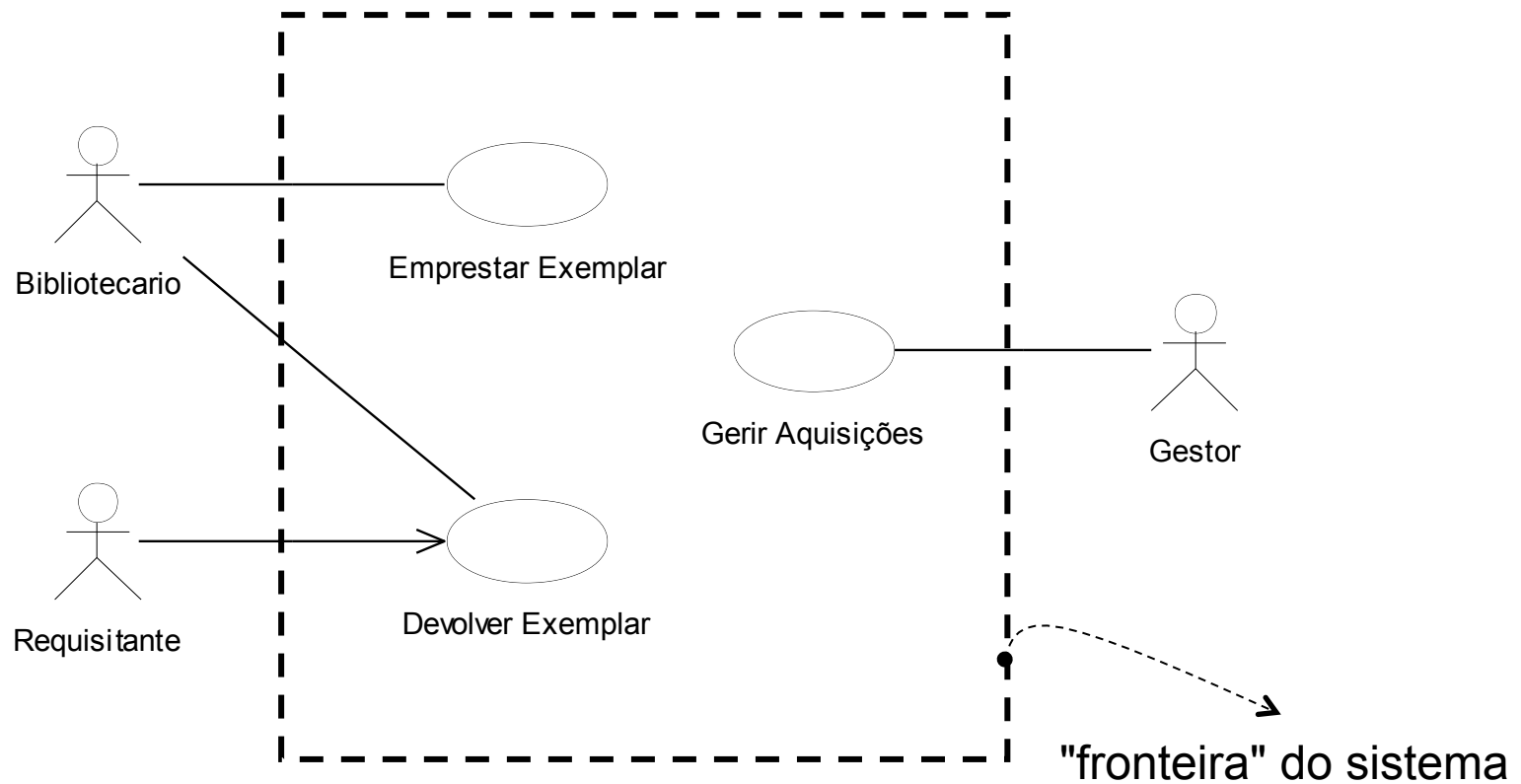
Casos de Utilização – formas de os identificar

- Focado nos actores (*actor-based*)
 - 1. Identificar os actores relacionados com o sistema (ou organização)
 - 2. Por actor, identificar os processos que ele inicia ou em que participa
- Focado nos eventos (*event-based*)
 - 1. Identificar os eventos externos a que o sistema tem que responder
 - 2. Relacionar os eventos com actores e com casos de utilização
- Exemplo
 - focado nos Actores

Actor	Processo
Bibliotecário	Emprestar exemplar Efectuar reserva Administrar requisitantes
Catalogador	Recepcionar encomenda Catalogar títulos
Gestor	Gerir aquisições Remover exemplares degradados
Requisitante	Devolver exemplar Enviar notificação

Casos de Utilização e Actores – exemplo

- A "fronteira" do sistema marca separação entre ambiente
 - interno – responsabilidades do sistema
 - externo – atores do sistema



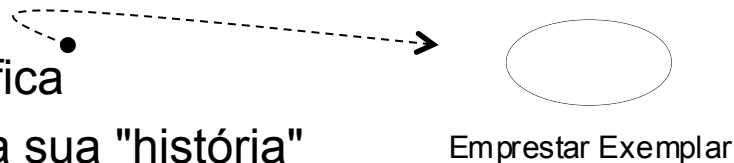
Casos de Utilização – a sua representação completa

- A representação completa tem que incluir

- uma notação gráfica
- a "história" que descreve

- Linguagem UML

- define a sua notação gráfica
- não define formato para a sua "história"



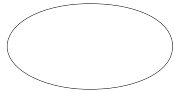
- Vamos adoptar o seguinte formato para a sua "história"

Cabeçalho	
Nome:	<i>nome do caso de utilização</i>
Resumo:	<i>síntese da "história"</i>
Referências:	<i>funções de sistema imputadas</i>

Cenário Principal (fluxo típico de eventos)	
Acção do Actor	Resposta do Sistema
<i>numeração única e por sequência para as acções e respostas</i>	
Cenários Alternativos (1, 2, 3, ...)	
Número de Sequência	Alternativa
<i>alternativa que ocorre em número(s) de sequência do Cenário Principal</i>	

Simplificação do formato sugerido em
Applying UML and Patterns, por Craig Larman

Casos de Utilização – Cenário Principal (exemplo)



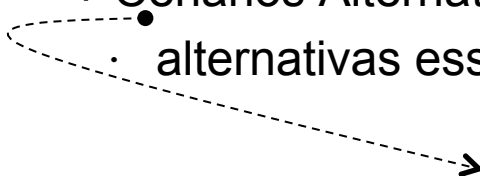
Emprestar Exemplar

Cabeçalho	
Nome:	<i>Emprestar Exemplar</i>
Resumo:	<i>Um exemplar, de determinado título, que esteja disponível, ou esteja reservado, é identificado e emprestado.</i>
Referências:	<i>R1.1, R1.3, R1.8, R1.10, R2.2</i>

Cenário Principal (fluxo típico de eventos)	
Acção do Actor	Resposta do Sistema
1 O Caso de Utilização inicia quando o requisitante indica pretender que lhe seja emprestado determinado título	
2 Requisitante indica qual o título	
3 Requisitante indica se existe reserva	
	4 Código do exemplar disponível
	5 Localização do exemplar
6 Identificação do requisitante	
	7 Autenticação do requisitante
	8 Autorização de empréstimo
	9 Emissão de talão de empréstimo

Casos de Utilização – Cenário Alternativo (exemplo)

- ... Notação Gráfica + Formato
- Formato =
 - Cabeçalho + Cenário Principal
 - + Cenários Alternativos
 - alternativas essenciais ao Cenário Principal



Cenário Alternativo 1	
Número de Sequência	Alternativa
1	Requisitante já apresenta exemplar
2, 3, 4, 5 sem efeito	
Cenário Alternativo 2	
Número de Sequência	Alternativa
4	Não existe exemplar disponível
Código de erro. 5, 6, 7, 8, 9 sem efeito	
Cenário Alternativo 3	
Número de Sequência	Alternativa
7	Requisitante não é autenticado
Apresentar informação sobre o requisitante. 8, 9 sem efeito	
8	Autorização não é concedida
Apresentar motivo. 9 sem efeito	

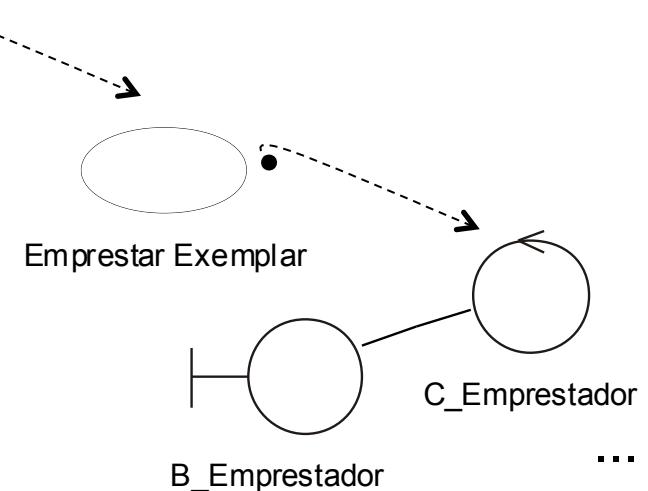
Como usar o formato proposto?

- Na prática o formato pode ser usado de dois modos
 - Resumido – apenas inclui Cabeçalho
 - Expandido – inclui Cenário Principal e Cenários Alternativos
- Formato Resumido – nas fases iniciais usar para descrever
 - casos de utilização “menos críticos”
 - ... ajuda a compreender o âmbito do projeto
 - ... numa fase posterior ele poderá vir a ser expandido
- Formato Expandido – nas fases iniciais usar para descrever
 - casos de utilização mais "importantes" e "decisivos" para o sucesso
 - ... contribui para afinar sensibilidade sobre complexidade do sistema
 - ... é candidato a incursões de implementação

As Funções do Sistema e os Casos de Utilização

- O formato proposto inclui uma secção "Referências"
 - onde se indica quais as Funções de Sistema que lhe estão imputadas
- Todas as Funções do Sistema devem estar
 - imputadas a Casos de Utilização
- A secção de "Referências" é um elemento importante
 - na "manutenção de rasto" (*traceability*) entre artefactos
 - ... navegar de cada função do sistema à sua implementação e teste

R1.1	Registar empréstimos (exemplar e utilizador)	Evidente
R1.3	Actualizar inventário após empréstimo	Invisível
R1.8	Cancelar reserva de títulos	Evidente
R1.10	Garantir persistência e integridade dos dados	Invisível
R2.2	Manter fila de prioridade de reservas	Invisível



Algumas regras úteis (Nome e Início)

- Nome do Caso de Utilização
 - deve começar com um verbo para evidenciar que é um processo
- exemplos:
 - Comprar Produto, Submeter Pedido, ...
- A seguinte forma de começar a descrição do Cenário Principal

1. O Caso de Utilização inicia quando <Ator> <Inicia Evento>

- encoraja uma clara identificação do ator e evento de iniciação

- exemplos:

1	O Caso de Utilização inicia quando o requisitante indica pretender que lhe seja emprestado determinado título
---	---

1	O Caso de Utilização inicia quando o cliente chega à caixa registadora com produtos para adquirir
---	---

Algumas regras úteis (Ponto de Decisão e Alternativas)

- Um cenário pode ter pontos de decisão
 - num ponto de decisão diversos caminhos alternativos podem existir
- Na maioria das situações existe um "caminho típico"
 - sendo os restantes pouco usuais ou excepcionais
 - e.g. de exceção: "não existe nenhum exemplar do título pretendido"
 - ... esses são descritos nos Cenários Alternativos
- Em algumas situações pode não existir um "caminho típico"
 - todas as hipóteses são "igualmente prováveis"
 - e.g. "o cliente pode pagar com cartão, dinheiro ou cheque"
 - ... pode-se então "partir" o Cenário Principal em diversas Secções

Quando não existe um caminho típico ... (exemplo)

Cenário Principal - Secção "Principal"	
Acção do Actor	Resposta do Sistema
1 O Caso de Utilização inicia quando o cliente chega à caixa com produtos para adquirir	
2 (passos intermédios estão omitidos)	
3 Cliente indica modo de pagamento	
a dinheiro - ver "Pagamento Dinheiro"	
b cartão - ver "Pagamento Cartão"	
c cheque - ver "Pagamento Cheque"	
	4 Emite factura e recibo
5 Vendedor entrega recibo	

Cenário Principal - Secção "Pagamento Dinheiro"	
Acção do Actor	Resposta do Sistema
1 Cliente entrega dinheiro - possivelmente em valor superior ao da compra	
2 Vendedor regista montante	
	3 Cálculo de troco
4 Vendedor deposita dinheiro do cliente	
5 Vendedor entrega demasia ao cliente	

Escalonar Casos de Utilização

- É necessário definir prioridades entre Casos de Utilização
 - os mais prioritários devem ser detalhados primeiro
- Consideremos as seguintes características
 - que contribuem para o aumento da prioridade de um Caso de Utilização
 - ... esta lista não pretende ser exaustiva; é uma base de partida

a	envolve esforço de investigação em tecnologias novas ou arriscadas
b	inclui funções de alta complexidade ou de tempo de resposta crítico
c	tem alto contributo na adição das classes que descrevem os conceitos do domínio, ou requer serviços especiais de suporte à persistência
d	exige grandes volume de informação ou detalhado conhecimento de negócio
e	representa processo essencial na linha-de-negócio
f	suporta directamente o retorno do investimento ou tem alta contribuição na redução de custos

Escalonar Casos de Utilização – exemplo

- Atribuir peso a cada contributo
 - considerou-se que o peso varia entre
 - ... 1 (não afecta); 2 (peso médio); 3 (peso alto)
- Para cada Caso de Utilização, atribuir pontos aos contributos
 - considerou-se que os pontos variam entre
 - ... 0 (não contribui) e 5 (contributo alto)

Caso de Utilização	a	b	c	d	e	f	Soma
Peso	3	1	2	1	1	1	
Emprestar exemplar	0	0	3	0	4	4	14
Devolver exemplar	0	0	3	0	3	4	13
Recepcionar encomenda	0	0	2	3	3	4	14
Enviar notificação	2	0	1	0	3	1	12
Efectuar reserva	2	0	3	0	1	0	13
Administrar requisitante	0	0	4	0	0	0	8
Catalogar títulos	0	0	4	4	4	4	20
Gerir aquisições	0	0	4	4	3	4	19

Escalonar Casos de Utilização – a escolha ...

- De acordo com o critério de escalonamento
 - é estabelecida uma relação de ordem entre Casos de Utilização
- No exemplo, temos
 - "Catalogar títulos" com o maior valor (de acordo com critério adoptado)
 - ... este deve ser tratado no primeiro ciclo de desenvolvimento
- Outro Caso de Utilização a ser tratado no primeiro ciclo
 - é o que contribui para a "iniciação" do sistema
 - ... o "ponto de entrada" no sistema
- No exemplo, temos
 - "Administrar requisitantes", que embora não tenha valor máximo
 - ... é um "ponto de entrada" no sistema
 - ... o sistema não pode funcionar sem registo de requisitantes

... primeiro ciclo de desenvolvimento

Core Workflows

Requirements

Analysis

Design

Implementation

Test

Inception

Elaboration

Construction

Transition

Preliminary
Iteration(s)

iter.
#1

iter.
#2

iter.
#n

iter.
#n+1

iter.
#n+2

iter.
#m

iter.
#m+1

Iterations

Uma iteração na fase de
avaliação preliminar (*inception*)

Múltiplas versões de Casos de Utilização

- Após escolha do(s) Casos de Utilização é necessário
 - estimar se todo ele será tratado no prazo de um ciclo
 - ... esse prazo pode ser definido, por exemplo, em 2 semanas
- Assume-se a estratégia de "prazo de um ciclo" (*time-box of a cycle*)
 - a cada ciclo de desenvolvimento é atribuído um limite fixo de tempo
- Se o Caso de Utilização é demasiado complexo
 - pode requerer vários ciclos de desenvolvimento
- Se o Caso de Utilização requer vários ciclos de desenvolvimento
 - deve ser redefinido em termos de versões
 - ... cada versão deve estar confinada ao prazo de um ciclo

Múltiplas versões de Casos de Utilização – exemplo

- No exemplo temos dois Casos de Utilização a tratar no 1º ciclo
 - Catalogar títulos
 - Administrar requisitantes
- Catalogar títulos
 - versão 1: catalogar por edição de dados; apenas pesquisas básicas
 - versão 2: pesquisas elaboradas que permitam validar a catalogação
 - versão 3: catalogar gerando código de barras a colocar nos exemplares
- Administrar requisitantes
 - versão 1: inserir, remover e pesquisar requisitantes
 - versão 2: definir grupos de requisitantes interesses idênticos
- Estas versões são distribuídas pelos ciclos de desenvolvimento
 - em conjunto com os restantes Casos de Utilização

Múltiplas versões de Casos de Utilização – simplificações

- Após decidir simplificar um Caso de Utilização
 - diferentes versões de complexidade crescente têm que ser definidas
- Para cada versão têm que ser indicadas
 - todas as simplificações assumidas
- Exemplo de algumas simplificações típicas
 - Não há manutenção do inventário
 - Utilizador não precisa de ser autenticado
 - Edição manual (não há leitor de código de barras)
 - Não há registo histórico de hábitos (não detecta interesses comuns)
 - Não há cálculo de taxas
 - Não há impressão de talões, cartões, ...
 - Sistema não está integrado com os restantes da organização

Processo de Desenvolvimento e Casos de Utilização

- Identificar Funções do Sistema e Atributos do Sistema
 - o que o sistema faz e com que características
- Identificar Casos de Utilização e Atores
 - encontrar a "fronteira" do sistema
- Escrever em formato Resumido todos os Casos de Utilização
 - melhorar ideia daquilo que se pretende
- Escrever em formato Expandido os Casos de Utilização
 - mais críticos, influentes, de maior risco ...
 - de modo a compreender melhor a natureza e dimensão do problema
- Escalonar por ordem de prioridade os diversos Casos de Utilização
 - e organizar em versões que "caibam" num ciclo de desenvolvimento