

Especificação



- Objetivo da aula
 - esboçar algumas técnicas de especificação (quatro), dando ênfase a casos de uso
 - assume-se que os alunos já tenham visto, ou irão ver, isto em mais detalhe em alguma outra disciplina
- Justificativa
 - através da adoção de alguns padrões de redação, pode-se reduzir significativamente os defeitos em especificações
 - casos de uso podem dirigir o desenvolvimento dos testes
 - desenvolvimento dirigido por comportamento (BDD behavior driven development)
 - casos de uso podem ser aplicados a
 - interação do usuário (pessoa) com o artefato
 - interação entre artefatos

Leitura complementar: [Cockburn, 2005]

Mar 2017

Arndt von Staa © LES/DI/PUC-Ri

Historieta (extreme programming)



- Historieta (user story) pequena estória, algo que caiba em uma ficha catalográfica
 - historieta sempre tem um nome: o objetivo principal
 - historieta é sempre uma ação
- Estrutura da frase de uma historieta (sintaxe simples):
 - quem ou papel desempenhado?
 - faz o que?
 - por que faz?

Mar 2017

rndt von Staa © LES/DI/PUC-Ric

Historieta (extreme programming)



- Como poderia ser escrita a historieta "Registrar esforço de desenvolvimento"?
- quem faz o que por que
- Como desenvolvedor quero registrar em folhas de tempo o tempo gasto por tarefa realizada em determinado dia, para saber onde despendi o meu esforço naquele dia.
- · Quais são os elementos léxicos dessa historieta?
 - desenvolvedor
 - registrar
 - folha de tempo
 - tempo gasto
 - tarefa realizada
- dia
- saber
- despender
- esforço

Quais são os termos que precisam de mais explicação?

Mar 2017

Arndt von Staa © LES/DI/PUC-R

Historieta



Historietas podem utilizar elementos que necessitam de uma especificação mais detalhada. Ex.

- o que é uma folha de tempo?

Mar 2017

rndt von Staa © LES/DI/PUC-Ric

Historieta



O item do dicionário de dados (termos) a seguir explica suficientemente bem?

- Uma folha de tempo de um determinado funcionário registra o tempo por ele gasto (esforço) em todas as tarefas e atividades realizadas em um determinado dia
 - Definição: uma atividade leva a um artefato de qualidade controlada.
 - Definição: uma tarefa é um passo de trabalho na direção de concluir uma atividade.
 - Exemplo:
 - atividade: desenvolver um módulo
 - tarefas: projetar o módulo, redigir o módulo de definição, redigir o módulo específico de teste, redigir o script de teste inicial, redigir o módulo de teste, redigir o código do módulo, verificar e validar estaticamente todos artefatos do módulo, testar o módulo, redigir o script de teste final, assegurar total conformidade com todos os padrões requeridos, realizar o controle de aceitação

Mar 2017

Arndt von Staa © LES/DI/PUC-Ri

Historieta



- O que falta para poder entender corretamente?
 - Regras de negócio, ex.
 - atividades são realizadas por uma ou mais tarefas
 - as tarefas de uma mesma atividade podem ser realizadas por diferentes pessoas
 - as tarefas de uma mesma atividade podem ocorrer em diferentes dias.
 - uma mesma tarefa pode ocorrer repetidas vezes em um mesmo dia e/ou em dias diferentes.
 - uma atividade estará concluída somente quando o artefato a ser desenvolvido tiver sido aceito
 - ou seja, passou por todos os controles de qualidade.
 - » verificação
 - » validação
 - » aprovação (possivelmente só pela equipe de desenvolvimento)

Mar 2017

Arndt von Staa © LES/DI/PUC-Rid

7

Crítica à estrutura da frase



- A estrutura da frase usada até agora é sempre suficiente?
 - Como especificar coisas repetidas?
 - ex. para cada tarefa devem ser fornecidos ...
 - Como especificar inicializações?
 - ex. ao entrar no sistema, o desenvolvedor quer ver a folha de tempo do dia corrente com todos os dados preenchidos até o momento, inclusive os pré preenchidos
 - o que é um dado pré preenchido?
 - Mais alguma coisa?
 - ex. a lista de atividades pendentes, i.e. as que já deveriam estar concluídas e as que deveriam terminar no dia mais próximo do dia corrente
 - deu para entender isso?
 - Como registrar os requisitos não funcionais e inversos?

Mar 2017

Arndt von Staa © LES/DI/PUC-R

Historieta mais elaborada



Estilo mais elaborado:

- [<contexto: pré condições>]<ator: quem, ou o que></ação: quer, ou faz><objetos: o que objetos diretos e indiretos conforme requeridos pelo verbo><resultado: pós condições>[<restrições>]
- Pode ser fornecida uma lista complementar de dados e/ou de regras de negócio necessários para que o programador saiba o que deve ser programado.

Mar 2017

Arndt von Staa © LES/DI/PUC-Ri

Exemplo de historietas mais elaboradas



- Uma forma "impessoal" de redigir
 - O desenvolvedor deseja saber o esforço gasto por tarefa para que possa aprimorar o seu modo de trabalhar.
- Uma forma de redigir focada no papel do interessado
 - Como líder de projeto (scrum master) desejo saber o esforço gasto por todas tarefas de uma dada atividade para que eu possa aprimorar o processo de desenvolvimento que utilizo.
- Como é que se "mede" aprimorar?
- Quando é projetado o banco de dados?
- Nomes de tarefas e de atividade precisam ser padronizados?

Mar 2017

Arndt von Staa © LES/DI/PUC-Rio

.0

Exemplos de historietas mais elaboradas



- Para poupar tempo ao preencher,
 - ao ativar o sistema, o desenvolvedor quer ver a folha de tempo do dia corrente.
 - cria a folha do dia ao acessá-la pela primeira vez?
 - se não for dia seguido, cria os dias intermediários?
- Para organizar o seu trabalho diário,
 - o desenvolvedor deseja que as tarefas pendentes já venham pré preenchidas na folha correspondente ao dia corrente.
- Para poder relembrar ou mesmo corrigir dados incorretos,
 - o desenvolvedor deseja poder selecionar para ler ou editar qualquer uma das folhas de tempo de dias passados até o dia corrente inclusive.

Mar 2017

Arndt von Staa © LES/DI/PUC-Ric

4.4

Avaliação desta técnica de especificar



- Com uma especificação completa redigida neste estilo você sabe o que deverá ser desenvolvido?
 - você sabe quais são as finalidades (serviço) do programa?
 - você consegue sempre identificar as características (features)
 relevantes que constituem o programa/sistema?
 - cada historieta corresponde a exatamente uma característica?
 - você sabe quais as interações previstas com outros sistemas?
 - você sabe que requisitos interessam a outros interessados?
 - você sabe quais os requisitos não funcionais ou inversos a serem satisfeitos?
 - você sabe como interagir com o sistema?
 - você sabe, ou é capaz de imaginar, que erros de uso podem ocorrer?
 - você sabe como erros de uso devem ser tratados?

2017

Arndt von Staa © LES/DI/PUC-Rio

12

Mar 2017

Algumas recomendações



- Historietas são frequentemente incompletas
 - o desenvolvedor não deve completar da forma que acha, mas sim, deve procurar o autor, ou interessado (product owner na terminologia SCRUM) para tirar dúvidas
 - problema, o autor estará sempre presente?
 - se não estiver presente, um representante (*proxy*) tem suficiente conhecimento para poder responder?
 - o desenvolvedor é disciplinado e segue essa recomendação?
- Historietas interdependem
 - o tratamento isolado e, possivelmente em tempos não coincidentes, consegue assegurar que a solução dada para uma historieta não conflita com a solução dada para outra?

Stellman, A.; Greene, J.; 2014. *Learning Agile: Understanding Scrum, XP, Lean, and Kanban*. O'Reilly Media. Kindle Edition.

• É uma excelente referência para quem quer aprender processos ágeis

Mar 2017

Arndt von Staa © LES/DI/PUC-Rio

13

Especificação usando exemplos



- Uma especificação pode ser criada usando exemplos de uso
 - uma suíte verbal de casos de teste de aceitação
 - desenvolvimento baseado em testes de aceitação (acceptance test driven development ATDD)
 - uma variante de desenvolvimento baseado em testes (test driven development TDD)
- Cada caso de teste é um exemplo verbal de uso do sistema
- O conjunto deve ser abrangente para tratar cada aspecto do sistema
- Historietas podem em grande parte ser escritas no formato de um caso de teste de aceitação

Adzic, G.; Bridging the Communication Gap: Specification by Example and Agile Acceptance Testing; London, UK: Neuri, Kindle edition; 2009

Mar 2017

Arndt von Staa © LES/DI/PUC-Rio

Exemplo de caso de teste de aceitação



- A base de dados já contém usuário "jose.silva@..."
- Forneço:
 - nome: José Silva; telefone: (21)12345678; apelido: Zezé;
 e-mail: jose.silva@serviçoMail; senha: x1t3Z10; x1t3Z10
- Clico:

Alternativamente:

cadastrar

- chamo a função "cadastrar(dados fornecidos)"
- Sistema responde:
 - "Erro: o usuário 'jose.silva' já está cadastrado"
 - Do ponto de vista teste, qual a diferença entre "clicar" e "chamar"?
 - No momento não interessa se é ou não boa interface humana
 - no caso não é, por que?

Mar 2017

Arndt von Staa © LES/DI/PUC-Rid

15

Componente Controle de Acesso 1 / 3



- O Controle de Acesso é um componente genérico a ser utilizado por um sistema qualquer, digamos sis, para identificar usuários registrados que, ao realizarem o login no sistema sis, serão autorizados com seus respectivos direitos de uso
- Um usuário autorizado identifica-se por meio de um par <idUsuario, Senha>
- Cada par <idUsuario, Senha> vincula-se a exatamente um conjunto de direitos de uso
 - O conjunto de todos os possíveis direitos de uso é específico para cada sistema sis que venha a utilizar o componente Controle de Acesso

Sis é uma "variável" que receberá o nome do sistema específico

Mar 2017

Arndt von Staa © LES/DI/PUC-Ri

Ex: Componente Controle de Acesso 2 / 3



- O cadastro de usuários autorizados é mantido pelo gerente do *Controle de Acesso* do sistema sis.
- O gerente associa o conjunto de direitos de uso específico a cada par <idUsuario, Senha> contido no cadastro.
 - o usuário não pode alterar os direitos de uso
 - o usuário pode alterar a senha, porém não pode utilizar o mesmo nome para diferentes conjuntos de direitos de uso
 - caso o usuário tenha esquecido a senha, poderá solicitar o envio de nova senha por mensagem eletrônica para o endereço eletrônico cadastrado inicialmente pelo gerente do Controle de Acesso

Mar 2017

Arndt von Staa © LES/DI/PUC-Ric

17

Ex: Componente Controle de Acesso 3 / 3



Esboço da interface gráfica do componente Controle de Acesso

- O captcha tem por objetivo reduzir a chance de um robô tentar e conseguir interagir com o sistema
- A cada vez que o usuário puder entrar com dados,
 - os campos de dados deverão estar vazios
 - deverá ser fornecido um novo captcha gerado randomicamente

Sistema sis	
UsuárioSenha	
Digite os caracteres Captcha	
Login Cancelar Mudar senha Esqueci senha	

Obs. Para diminuir ações erradas por parte do usuário, os nomes do widgets devem dar uma boa ideia do seu propósito.

Mar 201

Arndt von Staa © LES/DI/PUC-Ri

Diagramas de fluxo de dados



- Diagramas de fluxo de dados (DFD) podem especificar a arquitetura identificando a interdependência entre
 - processos (programas, componentes, ...)
 - entidades externas
 - e.g. pessoas, outros sistemas, programas ou componentes
 - depósitos de dados
 - e.g. arquivos, bases de dados, tabelas
 - comunicação assíncrona
 - dados que fluem entre os elementos acima
 - comunicação síncrona
- Permitem particionar um sistema complexo em partes (processos) estanques com interfaces bem definidas
 - possível instrumento para projetar boas arquiteturas

Ao invés de DFDs poderiam ser utilizados diagramas de atividades da UML 2

Mar 201

Arndt von Staa © LES/DI/PUC-Rio

Arquitetura do sistema Controle de Acesso LES sistema de gerência de usuários Tabela de direitos de uso disponíveis para Sis Direitos de uso selecionáveis por Sis Dados de usuário Manter o cadastro de autorizado **Processo** usuários autorizados Configurador externo * Gerente Sistema ou agente Dados de usuário responsável autorizado Fluxos de **Entidades** Cadastro de dados Depósito externas usuários autorizados de dados Direitos de uso de usuário identificado Solicitação de Identificação, Obter direitos de uso direitos de uso Sistema de determinado Senha e Ação usuário autorizado Direitos de uso Componente da aplicação Usuário componente

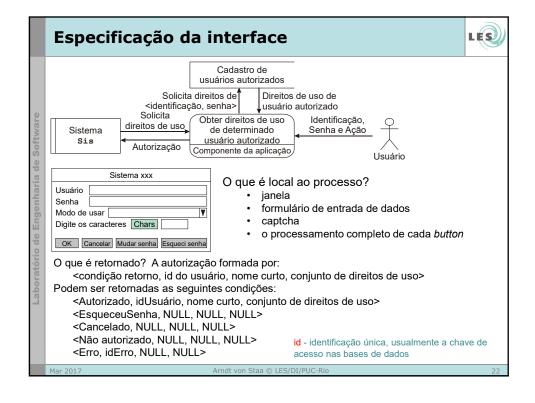
Critérios básicos de um processo



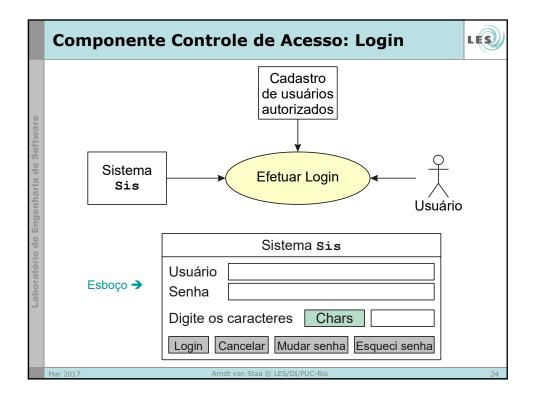
- Cada processo do DFD
 - possui um objetivo
 - usualmente o nome
 - redigido na forma de uma ação
 - requer a especificação da interface com outros elementos do diagrama
 - produz resultados condizentes com o objetivo
 - devem estar definidas as propriedades de todos os resultados (assertivas de saída, pós-condições)
 - requer dados que permitam produzir os resultados
 - devem estar definidas as propriedades de todos os dados (assertivas de entrada, pré-condições)
 - tudo que flui para dentro deve ser necessário
 - tudo que flui para fora deve ter sido inserido ou gerado

Mar 201

Arndt von Staa © LES/DI/PUC-Rio



Caso de uso por característica Cada caso de uso deve descrever exatamente uma característica exatamente: Goldilocks principle: just right, ou seja: nem mais, nem menos A implementação de um caso de uso pode envolver diversos programas, por exemplo cliente (browser) e servidor Exemplo ao fornecer os dados para um sistema evidentemente não estamos interessados somente nos dados digitados pelo cliente estamos interessados também na validação desses dados diversas propriedades podem ser validadas no browser, ex. campo numérico, número de caracteres permitidos mas a validação possivelmente requer acesso a bases de dados, o que deverá ser realizado no servidor, ex. usuário está cadastrado



Crítica aos diagramas de caso de uso



- O que significa o diagrama?
 - qual a finalidade ou propósito?
- Como verificar se a especificação está completa e correta?
 - quais são os dados?
 - quais são os resultados?
- Como verificar se a implementação está correta?
 - quais são os casos de teste?
 - quais são os critérios de aceitação?
 - quais são as regras de negócio?
- Como gerar os casos de teste?
- Qual o conjunto de todos os casos de uso do componente ou sistema?

Pode ser interessante para representar o fluxo de trabalho, neste caso é melhor usar BPMN (business process modeling notation)

Mar 2017

Arndt von Staa © LES/DI/PUC-Ri

25

Crítica aos diagramas de caso de uso



- Evidentemente a linguagem gráfica proposta é excessivamente abstrata
 - solução: utilizar um formulário para especificar os detalhes de cada caso de uso
 - o formulário a seguir foi inspirado em [Cockburn, 2001]

Mar 2017

Arndt von Staa © LES/DI/PUC-Rio

	Exemplo: Cas	so de uso Efetuar Login	LES
	Caso de uso		
Ф	Resumo		
Software	Escopo		
50	Ator principal		
Laboratório de Engenharia de	Interessados		
oratói	Invariante		
Lab	Pré condição		
	Acionamento		
	Mar 2017	Arndt von Staa © LES/DI/PUC-Rio	27

	Exemplo:	Caso de uso Efetuar Login	3
	Caso de uso	Efetuar login no sistema sis	
ıre	Resumo		
Software	Escopo		
	Ator principal		
rio de Engenharia de	Interessados		
aboratório	Invariante		
Lab	Pré condição		
	Acionamento		
	Mar 2017	Arndt von Staa © LES/DI/PUC-Rio	28

Caso de uso	Efetuar login no sistema Sis
Resumo	O sistema sis solicita que o componente Controle de Acesso forneça o direitos de uso correspondente ao par digitado pelo usuário, composto por identificação e senha previamente cadastrados
Escopo	
Ator principal	
Tourish	
Invariante	
Pré condição	
Acionamento	

Caso de uso	Efetuar login no sistema sis
Resumo	O sistema sis solicita que o componente Controle de Acesso forneça os direitos de uso correspondente ao par digitado pelo usuário, composto por identificação e senha previamente cadastrados
Escopo	Componente que visa obter autorização de uso
Ator principal	
Invariante	
Pré condição	
Acionamento	

Caso de uso	Efetuar login no sistema Sis
Resumo	O sistema sis solicita que o componente Controle de Acesso forneça o direitos de uso correspondente ao par digitado pelo usuário, composto por identificação e senha previamente cadastrados
Escopo	Componente que visa obter autorização de uso
Ator principal	Sistema Sis
Invariante	
Pré condição	
Acionamento	

Caso de uso	Efetuar login no sistema Sis
Resumo	O sistema sis solicita que o componente Controle de Acesso forneça os direitos de uso correspondente ao par digitado pelo usuário, composto por identificação e senha previamente cadastrados
Escopo	Componente que visa obter autorização de uso
Ator principal	Sistema Sis
Interessados	Sistema sis Objetivo: permitir o uso somente a usuários autorizados, restrito a um determinado conjunto de direitos de uso Usuário Objetivo: obter a autorização de uso segundo os direitos de uso com os quais foi registrado Cadastro de usuários autorizados Objetivo: registrar cada par <usuário, senha=""> autorizado e respectivos direito de uso</usuário,>
Invariante	
Pré condição	
Acionamento	

Caso de uso	Efetuar login no sistema Sis
Resumo	O sistema sis solicita que o componente Controle de Acesso forneça os direitos de uso correspondente ao par digitado pelo usuário, composto por identificação e senha previamente cadastrados
Escopo	Componente que visa obter autorização de uso
Ator principal	Sistema Sis
Interessados	Sistema sis Objetivo: permitir o uso somente a usuários autorizados, restrito a um determinado conjunto de direitos de uso Usuário Objetivo: obter a autorização de uso segundo os direitos de uso com os quai foi registrado Cadastro de usuários autorizados Objetivo: registrar cada par <usuário, senha=""> autorizado e respectivos direit de uso</usuário,>
Invariante	
Pré condição	
Acionamento	

Caso de uso	Efetuar login no sistema Sis
Resumo	O sistema sis solicita que o componente Controle de Acesso forneça os direitos de uso correspondente ao par digitado pelo usuário, composto por identificação e senha previamente cadastrados
Escopo	Componente que visa obter autorização de uso
Ator principal	Sistema sis
Interessados	Sistema sis Objetivo: permitir o uso somente a usuários autorizados, restrito a um determinado conjunto de direitos de uso Usuário Objetivo: obter a autorização de uso segundo os direitos de uso com os quais foi registrado Cadastro de usuários autorizados Objetivo: registrar cada par <usuário, senha=""> autorizado e respectivos direito de uso</usuário,>
Invariante	
Pré condição	O cadastro de usuários autorizados está atualizado, disponível e criptografado segundo uma chave interna
Acionamento	

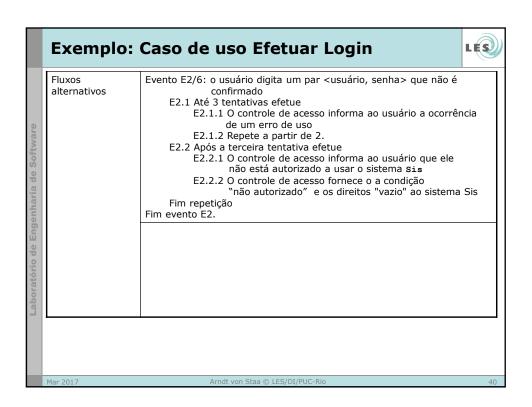
Caso de uso	Efetuar login no sistema sis
Resumo	O sistema sis solicita que o componente Controle de Acesso forneça os direitos de uso correspondente ao par digitado pelo usuário, composto por identificação e senha previamente cadastrados
Escopo	Componente que visa obter autorização de uso
Ator principal	Sistema sis
Interessados	Sistema sis Objetivo: permitir o uso somente a usuários autorizados, restrito a um determinado conjunto de direitos de uso Usuário Objetivo: obter a autorização de uso segundo os direitos de uso com os quais foi registrado Cadastro de usuários autorizados Objetivo: registrar cada par <usuário, senha=""> autorizado e respectivos direito de uso</usuário,>
Invariante	Observação: Uma invariante é uma assertiva que vale antes e após o processamento E cujos termos podem ser alterados durante o processamento
Pré condição	O cadastro de usuários autorizados está atualizado, disponível e criptografado segundo uma chave interna
Acionamento	Efetuar login inicia quando o sistema sis solicitar o fornecimento de um autorização de uso

	Fluxo principal		
Engenharia d	Fluxos alternativos		
Laboratório de Engenharia de Software	4ar 2017	Arndt von Staa © LES/DI/PUC-Rio	36

Fluxos alternativos	 O componente abre a janela e exibe o formulário O componente limpa os campos e gera o captcha O usuário digita sua identificação, senha e o captcha O usuário clica "Login" O componente verifica se os dados estão lexicamente corretos O componente busca os dados de <usuário, senha=""></usuário,> O componente cria a autorização: Autorizado com os direitos de uso correspondentes a <usuário, senha=""></usuário,> O componente fecha a janela O componente retorna a autorização ao sistema Sis
------------------------	--

	Exemplo:	Caso de uso Efetuar Login	LES
de Software	Fluxo principal	 O componente abre a janela e exibe o formulário O componente limpa os campos e gera o captcha O usuário digita sua identificação, senha e o captcha O usuário clica "Login" O componente verifica se os dados estão lexicamente corretos O componente busca os dados de <usuário, senha=""></usuário,> O componente cria a autorização: Autorizado com os direitos de correspondentes a <usuário, senha=""></usuário,> O componente fecha a janela O componente retorna a autorização ao sistema Sis 	e uso
Laboratório de Engenharia	Fluxos alternativos	Evento E1/5: Um ou mais campos estão lexicamente incorretos E1.1 Até 3 tentativas efetue E1.1.1 O controle de acesso informa ao usuário a ocorrên de um erro de uso E1.1.2 repete a partir de 2. E1.2 Após a terceira tentativa efetue E1.2.1 O controle de acesso informa ao usuário que ele não está autorizado a usar o sistema Sis E1.2.2 O controle de acesso fornece a condição "não autorizado" e os direitos "vazio" ao sistema Fim repetição Fim evento E1.	
	Mar 2017	Arndt von Staa © LES/DI/PUC-Rio	38

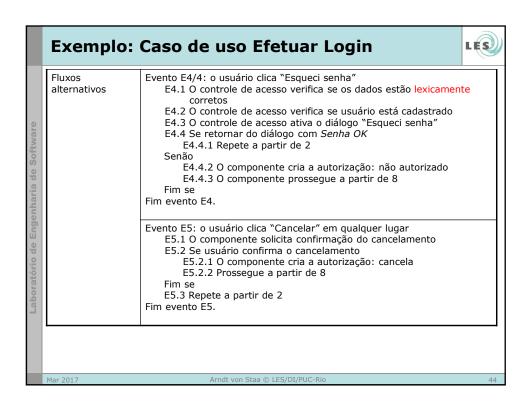
	Exemplo:	Caso de uso Efetuar Login	LES
Engenharia de Software	Fluxos alternativos		
Laboratório de E			
	Mar 2017	Arndt von Staa © LES/DI/PUC-Rio	39



Fluxos alternativos	Evento E2/6: o usuário digita um par <usuário, senha=""> que não é confirmado E2.1 Até 3 tentativas efetue E2.1.1 O controle de acesso informa ao usuário a ocorrência de um erro de uso E2.1.2 Repete a partir de 2. E2.2 Após a terceira tentativa efetue E2.2.1 O controle de acesso informa ao usuário que ele não está autorizado a usar o sistema sis E2.2.2 O controle de acesso fornece o conjunto "não autorizado" ao sistema sis Fim repetição Fim evento E2. Evento E3/4: o usuário clica "Trocar senha" E3.1 O controle de acesso verifica se os dados estão lexicamente corretos</usuário,>
	E3.2 O controle de acesso verifica se <usuário, senha=""> está autorizado E3.3 O controle de acesso ativa o diálogo "Trocar senha" E3.4 Repete a partir de 2 Fim evento E3.</usuário,>

	Exemplo:	Caso de uso Efetuar Login	LES
Laboratório de Engenharia de Software	Fluxos alternativos		
	Mar 2017	Arndt von Staa © LES/DI/PUC-Rio	42

Fluxos alternativos	Evento E4/4: o usuário clica "Esqueci senha" E4.1 O controle de acesso verifica se os dados estão lexicamente corretos E4.2 O controle de acesso verifica se usuário está cadastrado E4.3 O controle de acesso ativa o diálogo "Esqueci senha" E4.4 Se retornar do diálogo com Senha OK E4.4.1 Repete a partir de 2 Senão E4.4.2 O componente cria a autorização: não autorizado E4.4.3 O componente prossegue a partir de 8 Fim se Fim evento E4.
------------------------	--



	Exemplo	: Caso de uso Efetuar Login	LES
	Pós condições		
9.	Garantia mínima		
Engenharia de Software	Requisitos		
Laboratório de En	Regras de negócio		
Labora	Casos de uso correlatos		
	Mar 2017	Arndt von Staa © LES/DI/PUC-Rio	45

	Exemplo	o: Caso de uso Efetuar Login	
1	Pós condições	O sistema sis recebeu a autorização <i>Autorizado</i> correspondente ao par <usuário, senha=""> fornecido</usuário,>	
0	Garantia mínima		
Engenharia de Software	Requisitos		
aboratório de Ei	Regras de negócio		
Labor	Casos de uso correlatos		
	Mar 2017	Arndt von Staa © LES/DI/PUC-Rio	46

Pós condições	O sistema sis recebeu a autorização <i>Autorizado</i> correspondente ao par <usuário, senha=""> fornecido</usuário,>
Garantia mínima	O componente retorna uma das autorizações: <i>Não autorizado, Cancela</i> , ou <i>Erro</i>
Requisitos	
Regras de negócio	
Casos de uso correlatos	

Pós condições	O sistema sis recebeu a autorização <i>Autorizado</i> correspondente ao par <usuário, senha=""> fornecido</usuário,>
Garantia mínima	O componente retorna uma das autorizações: <i>Não autorizado, Cancela</i> , ou <i>Erro</i>
Requisitos	 Segurança: todos os espaços que contenham dados do cadastro e/ou do componente devem ter sido obliterados antes de retornar Segurança: a senha nunca deve ser exibida Segurança: a cada ativação da entrada de dados, novo captcha deve ser fornecido Segurança: a cada ativação da entrada de dados os campos são fornecido em branco
Regras de negócio	
Casos de uso correlatos	

Pós condições	O sistema sis recebeu a autorização <i>Autorizado</i> correspondente ao par <usuário, senha=""> fornecido</usuário,>
Garantia mínima	O componente retorna uma das autorizações: <i>Não autorizado, Cancela,</i> ou <i>Erro</i>
Requisitos	 Segurança: todos os espaços que contenham dados do cadastro e/ou do componente devem ter sido obliterados antes de retornar Segurança: a senha nunca deve ser exibida Segurança: a cada ativação da entrada de dados, novo captcha deve ser fornecido Segurança: a cada ativação da entrada de dados os campos são fornecido em branco
Regras de negócio	 A identificação de usuário e a senha não devem conter caracteres diacríticos, espaços em branco, tabulações, nem os caracteres '%', '\', '/', '?', '*', '@' Os campos são sensíveis às caixa alta e baixa
Casos de uso correlatos	

Exemple	o: Caso de uso Efetuar Login
Pós condições	O sistema sis recebeu a autorização <i>Autorizado</i> correspondente ao par <usuário, senha=""> fornecido</usuário,>
Garantia mínima	O componente retorna uma das autorizações: <i>Não autorizado, Cancela,</i> ou <i>Erro</i>
Requisitos	 Segurança: todos os espaços que contenham dados do cadastro e/ou do componente devem ter sido obliterados antes de retornar Segurança: a senha nunca deve ser exibida Segurança: a cada ativação da entrada de dados, novo captcha deve ser fornecido Segurança: a cada ativação da entrada de dados os campos são fornecidos em branco
Regras de negócio	 A identificação de usuário e a senha não devem conter caracteres diacríticos, espaços em branco, tabulações, nem os caracteres '%', '\', '?', '*', '@' Os campos são sensíveis às caixa alta e baixa
Casos de uso correlatos	Reconfirmar usuário corrente durante o uso de sis Solicitar nova senha Mudar a senha Registrar dados de usuário Alterar dados de usuário Excluir dados de usuário
Mar 2017	Arndt von Staa © LES/DI/PUC-Rio

Como redigir as ações



- As ações devem ser legíveis e compreensíveis por todos os interessados
- Solução: redigir em português restrito

```
{<pré-condição> ":"} (<ator>|<entidade>) <atividade> {pós-condição}
  {<restrições>}
    pré-condição – condição que deve estar satisfeita para que a
         atividade possa ser realizada
                 - papel (não a pessoa) que realiza ou dispara a
    ator
         atividade
    entidade
                - equipamento, sistema ou componente que realiza ou
         dispara a atividade
                - ação a realizar, no formato:
    atividade
           - <verbo> <objetos>
           - verbo na terceira pessoa do presente do indicativo
    pós-condição – o que deve valer ao terminar a atividade
               - condições a serem respeitadas ao realizar a atividade
```

Como redigir as ações



- Exemplos
 - O usuário digita a identificação, a senha e o captcha
 - O usuário seleciona a ação
 - Errado: o aluno foi para casa
 - Certo: o aluno vai para casa
- As atividades não interligadas não devem ser compostas
 - Errado: o aluno vai para casa e estuda teste de software
 - Certo:

não vale conector de expressão lógica se as ações não forem interligadas

- o aluno vai para casa
- o aluno estuda teste de software

Mar 2017

Arndt von Staa © LES/DI/PUC-Rio

Como redigir as ações



Exemplo: sistema de controle de um elevador

• O <módulo controlador> <informa> ao <módulo motor de determinado poço> o <andar para o qual o elevador deve se deslocar>

ação sujeito verbo objeto indireto objeto direto

- O <módulo motor> calcula o <plano de movimentação>
 - O <plano de movimentação> é formado por: <direção> e
 <sequência de velocidades> que levam da <posição atual> ao
 <andar destino>
 - <posição atual> é o lugar em que o elevador se encontra no momento da consulta, pode ser entre andares
- O <módulo motor> executa o <plano de movimentação> regra de negócio

informar – verbo bitransitivo, i.e. requer objeto direto e indireto

4ar 2017

Arndt von Staa © LES/DI/PUC-Rio

53

Como redigir os fluxos



- Redação do fluxo (path) principal
 - 1. {pré :} (ator | entidade) atividade {pós} {restrições}
 - 2. {pré :} (ator | entidade) atividade {pós} {restrições}

...

- n. {pré :} (ator | entidade) atividade {pós} {restrições}
- Redação do fluxo alternativo

Evento i/j: <condição identificadora do evento>

Ei.1. {pré :} (ator | entidade) atividade {pós} {restrições}

Ei.2. {pré :} (ator | entidade) atividade {pós} {restrições}

...

Ei.k. {pré :} (ator | entidade) atividade {pós} {restrições}

/j é a linha em que ocorre o evento. Pode ser uma lista de linhas.

- j será um número simples se o evento ocorrer no fluxo principal,
- e será Ei.j se ocorrer em fluxo alternativo
- eventos externos (ex. time out ou cancela) não têm a parte /j

Mar 2017

Arndt von Staa © LES/DI/PUC-Ric

```
Como redigir controles

( Se | Quando | Dado ) < condição > então

x.1. {pré-condição :} (ator | entidade) atividade {pós-condição} {restrições}

...

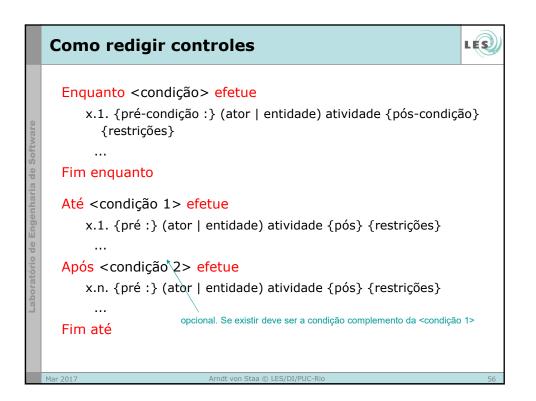
Fim ( se | quando | dado )

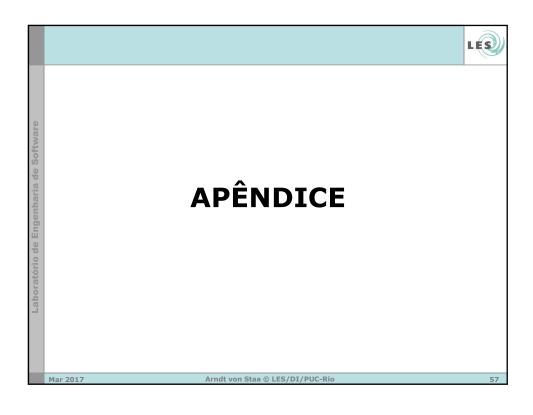
Para cada < elemento > ( contido em | de ) < conjunto > efetue

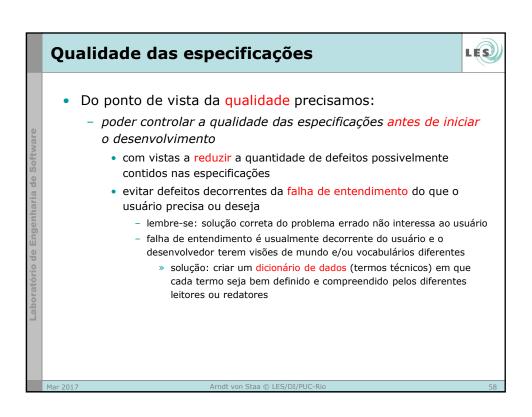
x.1. {pré-condição :} (ator | entidade) atividade {pós-condição} {restrições}

...

Fim para cada
```







Qualidade das especificações



- Do ponto de vista da qualidade precisamos:
 - poder controlar após desenvolver de modo a determinar se o que foi implementado corresponde ao especificado
 - poder criar suítes de teste a partir da especificação antes mesmo de desenvolver
 - assegura a verificabilidade da especificação
 - contribui para o desenvolvimento dirigido por testes
 - a suíte de teste existe antes de desenvolver e passa a ser uma forma de especificar baseada em exemplos
 - a suíte de teste visando teste automatizado viabiliza o controle a baixo custo durante o desenvolvimento incremental

Mar 2017

Arndt von Staa © LES/DI/PUC-Rid

F0

Casos de uso: como criá-los



- Processo coerente com processos ágeis de desenvolvimento,
 ex: Scrum e eXtreme Programming
 - identificam-se as características (de baixo nível de abstração), possivelmente a partir de historietas
 - ordenam-se as características segundo a relevância para o usuário e, se necessário, observando as precedências
 - da característica mais relevante para a menos relevante
 - desenvolve-se incrementalmente o sistema adicionando características aceitas
 - ao término do desenvolvimento de uma característica, o usuário avalia o sistema parcialmente desenvolvido (construto, build) e verifica se atende aos seus anseios atuais
 - possivelmente adicionam-se novas e/ou eliminam-se características definidas e ainda não implementadas
 - reavaliam-se a relevância das características ainda a implementar e repete até concluir a liberação final

Mar 2017

Arndt von Staa © LES/DI/PUC-Rio

Caso de	e uso	nome (identificação) do caso de uso		
Resumo)	descrição resumida do objetivo principal		
Escopo		o que é abrangido pelo caso de uso		
Ator pri	ncipal	nome do ator que ativa (aciona) o caso de uso		
Interess	sados	nome (identificação) do interessado	descrição do interesse / objetivo	
Invariar	nte	condições que deverão estar satisfeitas antes e após efetuar o caso de uso devem envolver somente dados, arquivos, recursos e estados que poderão ser alterados pelo caso de uso		
Pré con	dições	 condições que precisam estar satisfeitas antes de iniciar o caso de uso Invariante + Pré condições devem envolver todos os dados, arquivos, recursos e estados necessários para poder realizar todo o caso de uso 		
Acionan	nento	condição ou evento que inicia o caso	de uso, disparado pelo ator principal	

	Formulári	io de especificação de casos de uso 2/3		
aboratório de Engenharia de Software				
	Fluxo principal	sequência de ações a serem executadas no caso normal		
	Fluxos alternativos	sequências de ações disparados por eventos previstos durante a execução ações do fluxo principal, ou de fluxos alternativos já descrito • fluxos alternativos devem indicar onde iniciam e onde terminam	0	
	Pós condições	condições que devem estar satisfeitas ao terminar a execução normal do caso de uso invariante + pós-condições deve envolver tudo o que resulta ao final do fluxo normal pós-condições devem poder ser utilizadas como o oráculo do teste		
	Garantia mínima	condições que devem ser satisfeitas sempre que o caso de uso não termir de forma normal. Possivelmente a invariante não estará totalmente assegurada ao terminar de forma não normal, portanto a garantia mínima deve relatar o todo o conjunto de condições asseguradas.		
Lal				
	Mar 2017	Arndt von Staa © LES/DI/PUC-Rio	62	

LES Formulário de especificação de casos de uso 2/3 requisitos adicionais, tais como requisitos não funcionais e características Requisitos desejáveis da interface humano computador (IHC) Regras de especificação das regras (assertivas) a serem satisfeitas nos diferentes negócio nassos relação dos casos de uso correlacionados com o presente caso de uso Casos de uso • ex. os demais casos de uso que complementam o conjunto de correlatos características que implementam um determinado propósito **Artefatos** relação de documentos, artigos, mensagens, fontes de informação, etc. que correlatos contêm alguma informação relevante para o caso de uso

Referências bibliográficas



Adzic, G.; Bridging the Communication Gap: Specification by Example and Agile Acceptance Testing; London, UK: Neuri, Kindle edition; 2009

Cockburn, A.; Writing Effective Use Cases; Reading, Massachusetts: Addison-Wesley; 2001

Cockburn, A.; Escrevendo Casos de Uso Eficazes - Um Guia para Desenvolvedores de Software; São Paulo, SP: Bookman; 2005

Díaz, I.; Losavio, F.; Matteo, A.; Pastor, O.; "A Specification Pattern for Use Cases"; Information & Management 41; New York, NY: Elsevier; 2004; pags 961-975

Dijkstra, E.W.; "Guarded commands, non-determinacy and formal derivation of programs"; *Communications of the ACM* 18(8); New York, NY: ACM Association for Computing Machinery; 1975; pags pp 453-457

Hall, A.; Chapman, R.; "Correctness by Construction: Developing a Commercial Secure System"; *IEEE Software* 19(1); Los Alamitos, CA: IEEE Computer Society; 2002; pags 18-25

Mar 201

Arndt von Staa © LES/DI/PUC-Ric

Referências bibliográficas



Heumann, J.; "Generating Test Cases from Use Cases"; The Rational Edge e-zine; New York, NY: International Business Machines; 2001; Buscado em: 22/jan/2009; URL: www.ibm.com/developerworks/rational/library/content/RationalEdge/jun01/GeneratingTestCasesFromUseCasesJune01.pdf

Nebut, C.; Fleurey, F.; Jézéquel, J-M.; Traon, Y.L.; "Automatic Test Generation: A Use Case Driven Approach"; *IEEE Transactions on Software Engineering* 32(3); Los Alamitos, CA: IEEE Computer Society; 2006; pags 140-155

Pinto, T.D.; *Uma Ferramenta para Geração e Execução Automática de Testes Funcionais Baseados na Descrição Textual de Casos de Uso*; Dissertação de Mestrado, DI/PUC-Rio; 2013

o site da ferramenta é http://funtester.org/download/

Stellman, A.; Greene, J.; 2014. *Learning Agile: Understanding Scrum, XP, Lean, and Kanban*. O'Reilly Media. Kindle Edition

Mar 2017

Arndt von Staa © LES/DI/PUC-Rio

