

Introdução aos Sistemas de Informação

Sistema e seus Requisitos

Aula 02
Profa. Rosângela

O que será visto?

- Processo de software - o que é?
- Qualidade de um software
- Requisitos
- Documento de Requisitos
- Técnicas para Obtenção de Requisitos

2

Processo de Software

- Um conjunto de atividades e resultados associados que geram um produto de software. Essas atividades são, em sua maioria, executadas por desenvolvedores de sistemas.

3

Atividades Fundamentais no Processo de Software

- **Especificação do Software** - definição de requisitos e análise de requisitos - a funcionalidade do software e as restrições em sua operação devem ser definidas.
- **Desenvolvimento do Software** - projeto e implementação - o software deve ser produzido de modo que atenda a suas especificações.
- **Validação do software** - integração e teste - o software tem de ser validado para garantir que ele faz o que o cliente deseja.
- **Evolução do software** - o software deve evoluir para atender às necessidades mutáveis do cliente.

4

Atributos para um bom software

- **Facilidade de Manutenção:**
 - Possibilidade do software evoluir para atender às necessidades mutáveis dos clientes.
- **Nível de Confiança:**
 - Confiabilidade, proteção e segurança. O software confiável não deve ocasionar danos físicos ou econômicos, no caso de defeitos no sistema.
- **Eficiência:**
 - Rapidez de resposta, o tempo de processamento, a utilização de memória, entre outros. (não deve desperdiçar os recursos do sistema, como memória e ciclos de processador).
- **Facilidade de Uso:**
 - Dispor de interface apropriada com o usuário e de documentação adequada. Deve ser utilizável para qualquer tipo de usuário.

5

Responsabilidade Profissional e Ética

- **Confidencialidade** - respeitar seus empregadores ou clientes, quer tenham ou não assinado um acordo formal, quanto ao sigilo de informações.
- **Competência** - não devem enganar quanto ao seu nível de competência, ou seja, não devem aceitar serviços que estejam fora do seu limite de competência.
- **Direitos de propriedade intelectual** - estar cientes das leis locais que regulam o uso da propriedade intelectual, como patentes e direitos autorais. Devem ser cuidadosos, a fim de assegurar que a propriedade intelectual de empregadores e clientes seja protegida.
- **Má utilização de computadores** - não empregar suas habilidades técnicas para o mau uso de computadores de outras pessoas. Abrange desde casos triviais - jogar no computador do cliente - até casos sérios - disseminação de vírus.

6

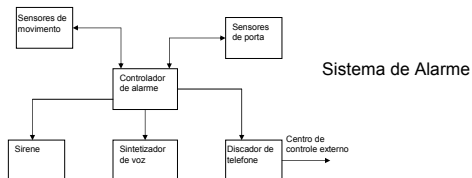
Engenharia de Sistemas

- É a atividade de especificar, projetar, implementar, validar, implantar e manter os sistemas como um todo. Os engenheiros de sistema não se ocupam apenas com o software, mas com as interações de software, hardware e sistema com os usuários e seu ambiente.

7

Modelagem de Sistemas

- O que é um modelo?
- Modelo de Arquitetura - mostra os subsistemas principais e suas inter-conexões.



8

Funcionalidade dos Subsistemas do Sistema de Alarme

Subsistema	Descrição
Sensor de movimento	Detecta movimento nos cômodos monitorados pelo sistema.
Sensor de porta	Detecta a abertura de porta nas portas externas do edifício.
Controlador de alarme	Controla a operação do sistema.
Sirene	Emite um aviso sonoro quando um intruso é detectado.
Sintetizador de voz	Sintetiza uma mensagem de voz dando a localização do possível intruso.
Discador de telefone	Faz as chamadas externas para avisar a segurança, a polícia, etc..

9

O que é um Requisito?

- Um requisito é uma afirmação do que o sistema **deve fazer** ou quais as **características que ele deve ter**.
- É escrito sob a perspectiva da pessoa do negócio.
- Enfoque da Análise: **O QUÊ o sistema deve fazer**
- **Requisitos de Negócio** - necessidades dos usuários do negócio (Requisitos do Usuário, algumas vezes)
- Enfoque do Projeto: **COMO** o sistema faz alguma coisa (implementação). São escritos sob a perspectiva do desenvolvedor (**Requisitos do Sistema**)
- Não há uma linha dividindo o "quê" e o "como".

10

Requisitos funcionais

- São declarações de funções que o sistema deve fornecer, como o sistema deve reagir a entradas específicas e como deve se comportar em determinadas situações. Em alguns casos, os requisitos funcionais, podem também explicitamente declarar o que o sistema não deve fazer. A especificação deve ser completa e consistente.
- **Completa** - que todas as funções requeridas pelo usuário devem estar definidas.
- **Consistente** - requisitos não devem ter informações contraditórias

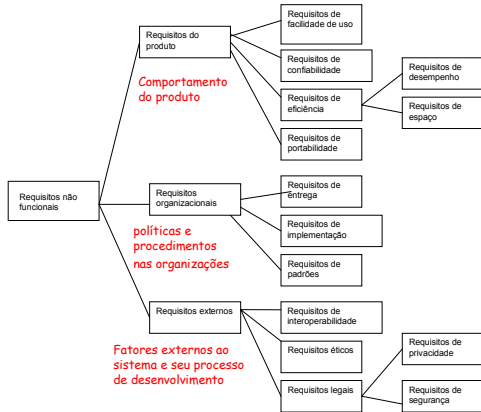
11

Requisitos não funcionais

- São aqueles que não dizem respeito diretamente às funções específicas fornecidas pelo sistema. Eles podem estar relacionados a propriedades de sistemas como : confiabilidade, tempo de resposta e espaço em disco. Como alternativa, eles podem definir restrições para o sistema como a capacidade dos dispositivos de E/S e as representações de dados utilizadas nas interfaces de sistema.
- **Exemplo:**
 - Se um sistema de aviação não atender aos seus requisitos de confiabilidade, ele não será atestado como seguro para operação;
 - Se um sistema de tempo real falhar em cumprir com seus requisitos de desempenho, as funções de controle não operarão corretamente

12

Políticas e Procedimentos nas Organizações



13

Exemplos de Requisitos não Funcionais

- **Requisito de produto:** Deve ser possível que toda a comunicação necessária entre o ambiente X e o usuário seja expressa no conjunto-padrão de caracteres YY.
- **Requisito organizacional:** O processo de desenvolvimento de sistema e os documentos a ser entregue deverá estar de acordo com o processo e os produtos a serem entregues, definidos em STAN-95.
- **Requisito Externo:** O sistema não deverá revelar aos operadores nenhuma informação pessoal sobre os clientes, além de seus nomes e o número de referência.

14

Exemplo de Requisito não Funcional

- **Confiabilidade de hardware**
 - Qual a probabilidade de um componente de hardware falhar e quanto tempo leva para reparar esse componente?
- **Confiabilidade de software**
 - Qual é a probabilidade de que um componente de software venha a produzir uma saída incorreta? Diferentemente da falha de hardware, ele pode prosseguir em operação mesmo depois que um resultado incorreto tenha sido produzido.
- **Confiabilidade de operador**
 - qual a probabilidade de que o operador de um sistema cometa um erro?

15

Se eu ignorar os Requisitos Não Funcionais, o que acontece?

- Desenvolvedor de sistema há vários anos
- Projeto vendido ao cliente com base na definição de requisitos
- Custo do sistema \$5.000,00
- Custo informado \$250,00
- Causas:
 - Desenvolvedor entende o sistema como se fosse só para um usuário;
 - Cliente quer sistema para ser usado por vários usuários em 3 diferentes departamentos e que possam trabalhar concorrentemente.

16

Especificação de Requisitos (cont.)

- A atividade a ser desenvolvida é captar os requisitos sob uma perspectiva dos usuários, isto é, os modelos gerados procuram definir a funcionalidade e restrições que devem ser consideradas para atender às necessidades dos usuários.
- A etapa de Especificação de Requisitos é independente do modelo de processo escolhido, uma vez que trata os requisitos do sistema sob uma perspectiva externa.

17

Definição de Requisitos

- Texto que relata
 - Requisitos Funcionais
 - Requisitos Não funcionais
- Numerar requisitos: agrupar os requisitos de acordo com a funcionalidade
- Exemplo:
 - A. Requisitos Funcionais
 - 1. Impressão
 - 1.1 O usuário deseja imprimir um relatório com informações cadastrais de todos os clientes.
 - B. Requisitos Não Funcionais
 - 1. Requisitos de Desempenho
 - 1.1 O tempo de resposta deve ser menor que 7 segundos.

18

Analogia

- Construção de uma casa com a construção de um Documento de Requisitos
 - Os futuros moradores sabem quantos eles são, o que eles esperam da casa em termos de dependências, mas não têm habilidade de projeto e conhecimentos de engenharia para construí-la.
 - Por outro lado, se o arquiteto atuar sozinho, não conseguirá projetar algo como os moradores idealizaram.

19

Problemas que normalmente ocorrem...

- **Falta de clareza** - é difícil utilizar a linguagem natural de maneira precisa e sem ambiguidade, sem produzir um documento de difícil leitura.
- **Confusão de requisitos** - quando os requisitos funcionais e não funcionais, os objetivos do sistema e as informações sobre o projeto não estão claramente definidos.
- **Fusão de requisitos** - quando vários requisitos diferentes são expressos junto como um único requisito.

20



Como o cliente
explica...



Como o cliente
explica...



Como o líder de projeto
entende...



Como o cliente
explica...



Como o líder de projeto
entende...



Como o analista
projeta...



Como o cliente
explica...



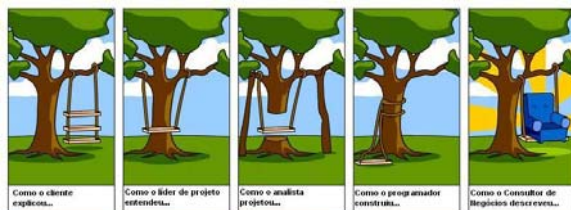
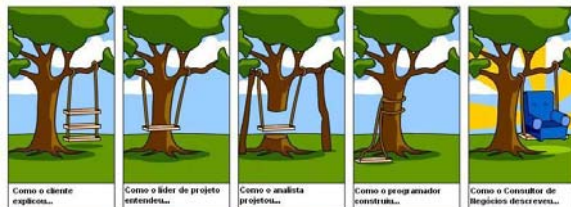
Como o líder de projeto
entende...

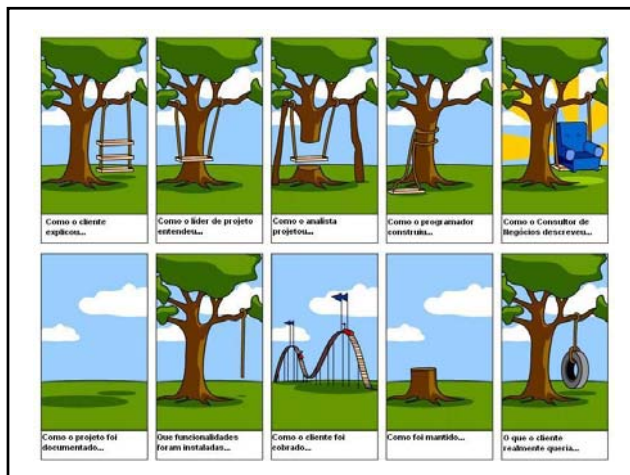
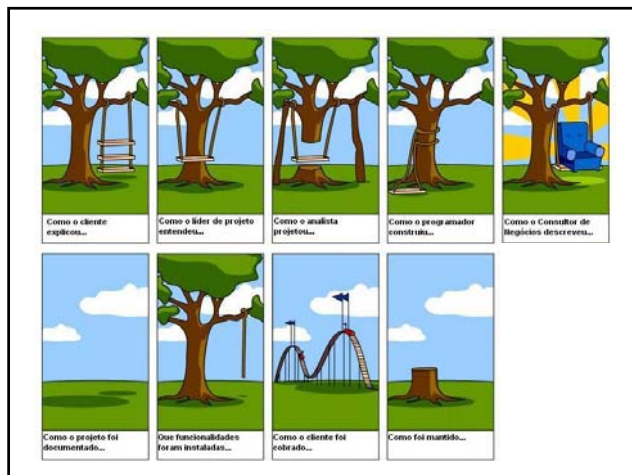


Como o analista
projeta...



Como o programador
constrói...





Documento de Requisitos(DR): formalização

- É a declaração oficial do que é exigido dos desenvolvedores de sistema.
- Deve incluir os requisitos de usuário para um sistema e uma especificação detalhada dos requisitos de sistema. Algumas vezes, esses dois tipos de requisitos podem ser integrados em uma única descrição. Em outros casos, os requisitos de usuários são definidos em uma introdução à especificação dos requisitos de sistema. Podem ser apresentados em documentos separados se possuírem um número grande de requisitos.

Seis "Requisitos" de um DR (Heninger, 1980)

1. Especificar somente o comportamento externo do sistema.
2. Especificar as restrições à implementação
3. Ser fácil de ser modificado
4. Servir como uma ferramenta de referência para os responsáveis pela manutenção do sistema
5. Registrar a estratégia sobre o ciclo de vida do sistema
6. Caracterizar respostas aceitáveis para eventos indesejáveis

Padrão IEEE

1. Introdução

a. **Propósito**

Deve delinear o propósito do documento de requisitos.

b. **Escopo do produto**

Deve identificar, pelo nome, o software a ser produzido; deve explicar, o que o software fará e, se necessário, o que ele não fará.

c. **Definições, Acrônimos e Abreviações**

Deve fornecer as definições de todos os termos, acrônimos e abreviações utilizados no documento, para que se possa interpretá-lo adequadamente.

d. **Visão Geral do restante do documento**

Deve descrever o restante do documento de requisitos e explicar como esse documento está organizado.

33

Padrão IEEE (cont.)

• 2. Descrição Geral

- Descreve os fatores gerais que afetam o produto e seus requisitos. Essa seção não declara requisitos específicos. Ao invés disso, ela fornece um *background* para os requisitos específicos, os quais são definidos em detalhes na Seção 3.

• 2.1 Perspectiva do produto

- Deve colocar o produto em perspectiva com outros produtos relacionados. Se o produto é independente e totalmente auto-contido, esse fato deve ser declarado aqui. Se o Documento de Requisitos define um produto que é um componente de um sistema maior, como frequentemente ocorre, então essa subseção deve relatar os requisitos daquele sistema maior que têm alguma relação com a funcionalidade do software identificando as interfaces entre aquele sistema e o software em questão.

• 2.2 Funções do Produto

- Deve fornecer um resumo das maiores funcionalidades que o software deve realizar.

34

Padrão IEEE (cont.)

• 2.3 Características do Usuário

- Deve descrever as características gerais dos usuários do produto, incluindo nível de educação, experiência e conhecimento técnico.

• 2.4 Restrições Gerais

- Deve descrever as restrições que o sistema tem.

• 2.5 Suposições e Dependências

- Deve listar cada um dos fatores que afetam os requisitos declarados na Seção 3.

35

Padrão IEEE (cont.)

3. Requisitos

Deve conter todos os requisitos do software (funcionais, não funcionais e de interface) em um nível de detalhe suficiente para permitir que os projetistas possam elaborar um projeto que satisfaça esses requisitos. Os requisitos podem ser divididos em Requisitos Funcionais e Requisitos não funcionais. Cada requisito declarado deve possuir uma identificação única.

3.1 Requisitos funcionais

Descrição, entrada exigida, processamento, saída gerada

3.2 Requisitos não funcionais

do produto, organizacionais, externos

3.3 Atributos

Conter todos os atributos necessários para o software. Exs: de confiabilidade, segurança, portabilidade, manutenção

36

Padrão IEEE (cont.)

- Apêndices
- Índice

37

Padrão IEEE com Alterações

- **Índice:** Alfabético, índice de diagramas, de funções, etc..
- **Prefácio:** Definir o público alvo a que se destina o documento e descrever seu histórico da versão, incluindo lógica para a criação de uma nova versão e um sumário das mudanças feitas em cada versão.
- **Introdução:** Deve descrever: a necessidade do sistema, brevemente suas funções e explicar como ele deverá operar com outros sistemas; como o sistema se ajusta aos negócios em geral ou aos objetivos estratégicos da organização que está solicitando o software.
- **Glossário:** Deve definir os termos técnicos utilizados no documento. Não deve fazer suposições sobre a experiência ou o conhecimento do leitor.
- **Definição de requisitos de usuário:** Descrever os serviços fornecidos para o usuário e os requisitos não funcionais do sistema. Podem ser utilizadas: linguagem natural, diagramas ou outras notações que sejam compreensíveis pelo cliente.

38

Padrão IEEE com Alterações

- **Arquitetura de sistemas:** Apresentar uma visão geral de alto nível da arquitetura prevista de sistema, mostrando a distribuição de funções por meio de módulos de sistemas. Se houver reuso, os componentes devem ser destacados.
- **Especificação de requisitos do sistema:** Descrever requisitos funcionais e não funcionais com mais detalhes.
- **Modelos de Sistema:** Devem mostrar o relacionamento entre os componentes de sistema e o sistema e seu ambiente. Podem ser modelos de fluxos de dados, de objetos, etc.
- **Evolução do sistema:** Devem ser descritas as suposições fundamentais nas quais o sistema se baseia e as mudanças previstas, devidas à evolução do hardware, mudanças de usuário, etc.
- **Apêndices:** Detalhes e informações específicas relacionadas à aplicação que está sendo desenvolvida (Exs: descrições de hardware, banco de dados)

39

Usuários de um DR

- **Clientes de sistema** - Especificam os requisitos e os têm para verificar se eles atendem a suas necessidades. Especificam mudanças nos requisitos.
- **Gerentes** - Utilizam o documento para planejar um pedido de proposta para o sistema e para planejar o processo de desenvolvimento do sistema.
- **Engenheiros de sistema** - Utilizam os requisitos para compreender que sistema deve ser desenvolvido.
- **Engenheiros de teste de sistema** - Utilizam os requisitos para desenvolver testes de validação para o sistema.
- **Engenheiros de manutenção de sistema** - Utilizam os requisitos para ajudar a compreender o sistema e as relações entre suas partes.

40

Técnicas para obter requisitos

- Técnicas existentes
 - Entrevista
 - Questionário
 - Observação direta
 - Fluxo de processos
 - Sessões brainstorming
 - Prototipação
 - Casos de Uso

41

Entrevista

- Usa-se essa técnica quando poucas pessoas conhecem as informações necessárias para o desenvolvimento do sistema
- Preparar antecipadamente
- Perguntas objetivas
- Evitar constrangimento dos participantes

42

Procedimentos para Entrevista

- **Antes:**
 - Planejar; identificar a posição e responsabilidades do entrevistado; marcar horário; escolher local tranquilo.
- **Durante**
 - Apresente-se informando a finalidade da entrevista, explique as anotações que fizer, não demore mais que 2 horas; encerre com observação positiva; agradecendo a contribuição do entrevistado.
- **Depois**
 - Documente os pontos relevantes; envie a documentação ao entrevistado (aprovação final); se for necessário esclarecimento posterior, contate o entrevistado para marcar outra reunião; envie os resultados para os usuários e seus gerentes

43

Questionário

- Usa-se quando muitas pessoas conhecem as informações necessárias para o desenvolvimento do sistema.
- Preparar antecipadamente com questões objetivas.
- Desvantagem:
 - Comunicação com usuários restrita, não há troca de informação face a face.
 - Preparação exige tempo !!!

44

Procedimentos para Aplicar Questionário

- **Preparação:** identificar o tipo de informação que deseja obter. Escolher formato adequado, deixar espaço suficiente para respostas de questões descritivas. Montar questões simples, claras e concisas. Enviar carta acompanhando o questionário enfatizando sua importância.
- **Identificar os respondente:** nome, função e localização. Controlar para que todos recebam (monitorar)
- **Distribuição:** deve ser distribuído juntamente com as instruções detalhadas de como preenchê-lo e o prazo de devolução.
- **Análise das respostas:** analisar e consolidar as informações recebidas. Documentar as principais descobertas. Enviá-las juntamente com cópia do relatório para todos os respondentes.

45

Observação Direta

- Utilizada como processamento e confirmação de outros resultados (entrevista, questionário).
- Identificar documentos que devem ser coletados para posterior análise.
- Esclarecer o que será feito.
- Observar diretamente quem desenvolve o trabalho.
- Deve ter aprovação antecipada das gerências.

46

Procedimentos Observação Direta

- **Antes:** identificar as áreas, de usuários a serem observados; obter aprovação das gerências, nomes e funções das pessoas observadas; explicar a finalidade.
- **Durante:** familiarizar-se com o local que será observado. Observar facilidades manuais e não em uso; os agrupamentos organizacionais atuais. Coletar amostras de documentos e procedimentos escritos. Agradecer ao término.
- **Após:** documentar as descobertas feitas, consolidar os resultados, rever os resultados consolidados com as pessoas observadas e/ou com seus superiores.

47

Sessões Tipo JAD

- JAD - Joint Application Design - Projeto de Desenvolvimento de Aplicação em Conjunto
- Características
 - Brainstorming
 - Desenvolvido pela IBM
 - Certificação IBM para realizá-la

48

Definições JAD

- **Executivo Patrocinador:** gerente de nível mais alto comprometido com JAD, patrocina o processo do início ao fim.
 - Fornece diretrizes sobre objetivos e metas de um projeto
- **Gerentes funcionais e usuários finais:** peritos em sessões JAD.
- **Representantes do SI:** poucas pessoas convidadas a participar das sessões.
- **Líder de sessão JAD:** coração do processo JAD.
 - Seu papel é conduzir as entrevistas preparatórias antes da sessão JAD real com o executivo patrocinador e com gerentes funcionais para a definição do escopo básico do processo.

49

Processo JAD - 4 etapas

- Orientação Inicial (4 dias): finalidade do projeto, escopo e áreas funcionais envolvidas, objetivos que devem ser alcançados no final do workshop, suposições técnicas e de negócio que afetam o projeto, objetivos do workshop, fatores críticos de sucesso.
- Familiarização com a área: finalidade da tarefa, dados de entrada/saída, descrição do processo feito, problemas e oportunidades
- Preparação do material para o workshop; relatórios para simular as idéias
- Conduzir o workshop - 3 a 5 dias.

50

Prototipação

- **Permite:**
 - Reações iniciais do usuário
 - Sugestões do usuário
 - Inovações
 - Informações para revisão de planos

51

Tipos de Protótipos

- Não operacional: apenas interfaces de entrada e saída são implementadas
- Arranjado às Pressas: toda a funcionalidade do sistema. Não foi construído com o devido cuidado. Qualidade e desempenho são deficientes.
- Primeiro de uma série: útil quando o sistema será implantando em diversos locais.
- Características selecionadas: só parte será implementada.

52

Casos de Uso

- Cada caso de uso descreve um possível cenário de como a entidade externa interage com o sistema.
- São úteis na comunicação com os clientes, projetistas do sistema e responsáveis pelos testes.
- Para o Cliente: pode verificar se todas as funções desejadas estão especificadas.
- Para os Projetistas: representam os objetos a serem manipulados/armazenados pelo sistema.
- Para os Testadores: base para os testes do sistema.

53

Casos de Uso (cont.)

- Modelagem de diálogo que um usuário, um sistema externo ou outra entidade terá com o sistema a ser desenvolvido.
- Ator = entidade que interage com o sistema, pode ser um usuário, um dispositivo ou outro sistema.
- É escrito na voz ativa e no tempo presente.

54

Requisitos e Casos de Uso

- *Eles realmente são requisitos?* Especificam o que o sistema faz?
- *Eles não são todos os requisitos.* Eles não especificam interfaces externas, formatos de dados, regras de negócio e formulas complexas.

55

Exemplos de Casos de Uso

Caso de Uso: Cadastrar Cliente

Ator: Cliente

Fluxo Normal:

1. Cliente informa seu CPF ao Sistema
2. Sistema informa **que** o cliente não está cadastrado
3. Cliente informa demais dados
4. Sistema gerar senha e exibe para o cliente
5. Sistema cadastra o cliente
6. Sistema informa que o cliente foi cadastrado e também a senha gerada
7. Encerrar caso de uso

56

Exemplos de Casos de Uso

Curso Alternativo

2a. Cliente já cadastrado

2a.1 Sistema exibe mensagem "Cliente já consta do cadastro"

2a.2 Abandonar caso de uso

57

Caso de uso para Iniciar Sessão de uma Livraria na Internet

•Fluxo principal de eventos:

O cliente clica no botão Iniciar Sessão na Página Inicial. O sistema mostra a página de Iniciar Sessão. O Cliente digita seu nome de usuário e sua senha, então clica no botão OK. O sistema valida a informação de entrada contra os dados persistentes de Conta e então remete o cliente à Pagina Inicial.

•Fluxo alternativo de eventos:

Cliente clica no botão Conta Nova e é mostrada a página Criar Conta Nova

58

Continuação de Exemplo de Caso de Uso

• Fluxo alternativo de eventos:

Cliente clica no botão Lembrar Senha. Sua senha armazenada é exibida em uma caixa de diálogo separada. Quando o Cliente clicar OK, o sistema retornará à Pagina Inicial.

• Fluxo alternativo de eventos:

Cliente digita código de usuário não reconhecido pelo sistema, a página Criar Conta Nova é exibida.

• Fluxo alternativo de eventos:

Cliente digita senha incorreta, nova senha é solicitada.

• Fluxo alternativo de eventos:

Cliente digita senha incorreta por três vezes. É enviada uma mensagem para que ele entre em contato com o serviço de atendimento ao cliente.

59

Modelo de Caso de uso a ser utilizado

ID: UC-1 Nome do Caso de Uso: Exemplo

Criado por: Grupo 2 ou aluno1, aluno2

Atualizado por:

Data da criação: 02/04/2009

Atores: Cliente, Fornecedor

Descrição: Descrição da função desse caso de uso

Pré-condições: quais as pré-condições para que esse caso de uso possa ser realizado

Pós-condições: o que será obtido após a realização deste caso de uso

60

Modelo de Caso de uso a ser utilizado (cont.)

Curso Normal: passos que ocorrem em sequência

Cursos Alternativos: numerados de acordo com a alternativa do curso normal, representam exceções que podem acontecer.

Incluem: UC-22 Nome

Prioridade: Alta ou baixa para realização deste caso de uso

Frequência de Uso: aproximadamente duas vezes/semana.

Observações: se existir alguma observação pertinente.

61

Tarefa 1

- A capa do trabalho deve conter:
 - Grupo Nro
 - Nome e RA dos Integrantes do grupo
 - Nome do sistema em desenvolvimento
- Elaborar um documento:
 - Especificando o tipo de levantamento de requisitos que foi utilizado
 - Questões elaboradas para os clientes responderem
 - Respostas obtidas dos clientes
- Documento de requisitos, em linguagem natural contendo:
 - Título do projeto
 - Breve resumo - visão geral do sistema
 - Requisitos funcionais e não funcionais (numerados)

62

Tarefa 2 - Avaliação 1

- Elaborar um documento de requisitos com base em casos de uso.
- Modelo de Casos de Uso
- Descrição de casos de uso no formato apresentado.
- A capa do trabalho deve conter:
 - Grupo Nro
 - Nome e RA dos Integrantes do grupo
 - Nome do sistema em desenvolvimento

63