

# Uma estratégia para testes funcionais de aplicações Web apoiada na ConDado

Daniele Constant Guimarães

Profa. Dra. Eliane Martins  
Orientadora

Dezembro - 2003

## Introdução

- Aplicações Cliente-Servidor:
  - Centralizadas → Distribuídas
- Extensão: computação baseada na Internet
- Aplicações web
- Metodologias eficientes para análise, projeto, construção, teste e manutenção.

## Dificuldades de testar aplicações web

---

- Implementação em múltiplas linguagens de programação
- Interação com usuários e outras aplicações web

22/11/2005

Instituto de Computação -  
UNICAMP

3

## Estratégia de Teste em desenvolvimento

---

- Objetivo:
  - Funcionalidade
  - Falhas de comportamento
- Se aplica a testes de sistema
- Baseado nos testes de caixa preta

22/11/2005

Instituto de Computação -  
UNICAMP

4

## Trabalhos Relacionados

- ETACS (Estratégia de Teste de Software para Ambiente Cliente-Servidor): Avaliação dos riscos e definição de prioridades
- SCENT (A method for SCENario-Based Validation and Test of Software): Propõe casos de uso para os testes de sistema
- TOTEM (Testing of Object-Oriented software sysTEms with the UML): Obtenção dos requisitos de teste com base nos resultados da fase de análise

22/11/2005

Instituto de Computação -  
UNICAMP

5

## Objetivos do trabalho

- Estratégia:

Especificação → Casos de uso → Cenários  
→ Modelos de estado → Casos de teste

- Será baseada em riscos para evitar a geração de um grande número de casos de testes.
- Será utilizada em estudos de caso: aplicação da área espacial

22/11/2005

Instituto de Computação -  
UNICAMP

6

## Estratégia Proposta - Passos Principais

- P1: atribuir prioridades aos casos de uso do sistema
- P2: Gerar as seqüências de casos de uso
- P3: Derivar os cenários para os casos de uso
- P4: Derivar a máquina de estados clássica
- P5: Gerar as seqüências de testes

22/11/2005

Instituto de Computação -  
UNICAMP

7

## Ex: Imead

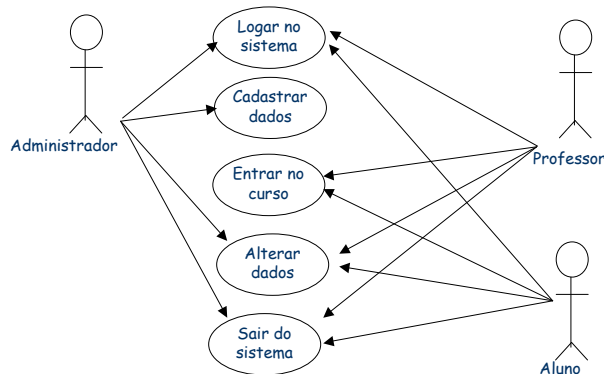
- Objetivo: auxiliar professores no ensino à distância via internet.
- Os professores e alunos serão cadastrados por um administrador.
- Deve permitir a alteração, exclusão e visualização dos dados cadastrados.
- Restrições:
  - O aluno e o professor não podem se logar várias vezes simultaneamente no sistema
  - O aluno tem um número limitado de acessos ao sistema.

22/11/2005

Instituto de Computação -  
UNICAMP

8

# Casos de uso - diagrama



22/11/2005

Instituto de Computação - UNICAMP

9

# Casos de uso - descrição

## Caso de uso: Alterar dados

**Fluxo normal; ator: administrador**  
**Cenário 20:**

1. O usuário escolhe a opção alterar dados.
2. O sistema pede a matrícula cujos dados serão alterados.
3. O usuário digita a matrícula.
4. A página para alteração de dados é exibida.
5. O usuário altera os dados
6. O usuário confirma a alteração.
7. Os dados são alterados com sucesso .

**Fluxo normal; atores: professor, aluno**

**Cenário 29:**

1. O usuário escolhe a opção alterar dados.
2. A página para alteração de dados é exibida.
3. O usuário altera os dados
4. O usuário confirma a alteração.
5. Os dados são alterados com sucesso

[Voltar para cenários .](#)

22/11/2005

Instituto de Computação - UNICAMP

10

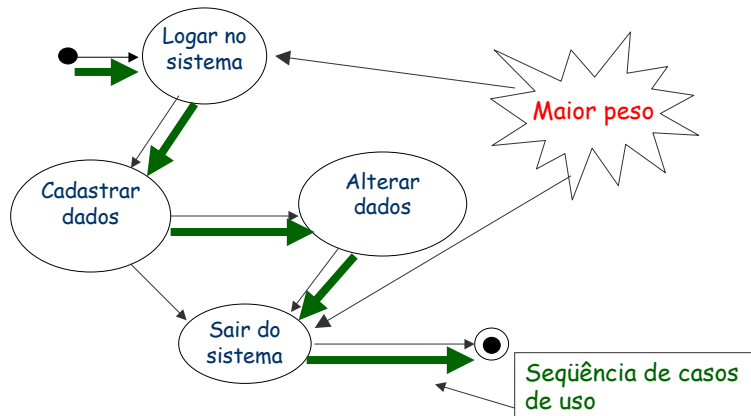
## Casos de uso: Cálculo da prioridade

- Escolher itens para avaliação de riscos
- Livro: "Effective Methods for Software Testing" de William Perry
- Para o exemplo: foram escolhidos 6 itens.

## Casos de uso: Cálculo da prioridade

Ator: administrador	
Casos de uso	Total
Logar no sistema	40,66
Cadastrar dados	27
Alterar dados	27,66
Sair do sistema	40,66

## Diagrama de atividade e Sequências de casos de uso



22/11/2005

Instituto de Computação -  
UNICAMP

13

## Cenários para os casos de uso

- Obtidos a partir dos fluxos do caso de uso.
- Calcular a prioridade dos cenários: mesmos critérios usados no cálculo da prioridade dos casos de uso.
- Modelo: diagrama de seqüência

22/11/2005

Instituto de Computação -  
UNICAMP

14

## Cenários: cálculo da prioridade

Matrícula  
inexistente

Caso de uso: Alterar dados

Fluxo normal

Cenários	Ator	Fluxo	Total
20	Administrador	Normal	27
21	Administrador	Alternativo	25
27	Administrador	Exceção	27
29	Professor, aluno	Normal	31
30	Professor, aluno	Exceção	29

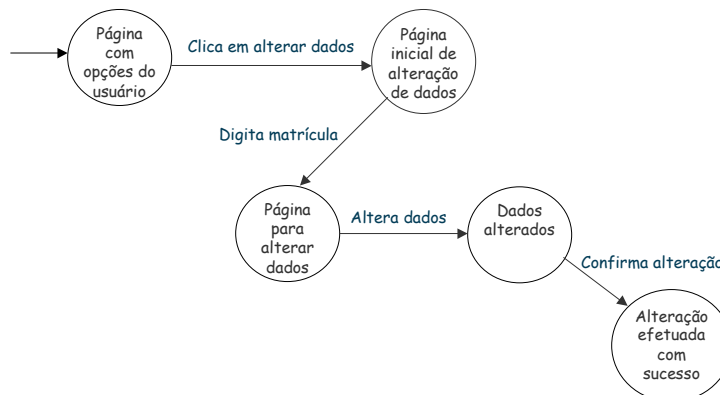
22/11/2005

Instituto de Computação -  
UNICAMP

15

## Máquina de estados

Ator: administrador - Cenário 20



22/11/2005

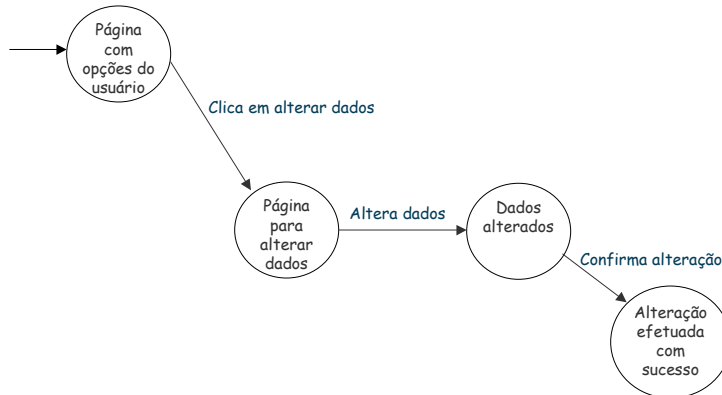
Instituto de Computação -  
UNICAMP

16



# Máquina de estados

Atores: professor, aluno - Cenário 21



22/11/2005

Instituto de Computação -  
UNICAMP

17

# Geração dos casos de testes

- Ferramenta: ConDado
- 1ª versão:
  - Um único modelo com todos os cenários
  - + de 72000 casos de teste
- 2ª versão
  - Um modelo para cada cenário
  - Um caso de teste para cada cenário (total: 14)

22/11/2005

Instituto de Computação -  
UNICAMP

18

## Considerações finais

---

- Objetivo: garantir a qualidade das aplicações web testadas
- Próximo passo:
  - Procurar um algoritmo que percorra o diagrama de atividades passando pelos casos de uso de maior peso.