



Avaliação em IHC

Parte 2 - Avaliação Heurística

Prof^ª. Dra. Marília S. Mendes
marilia.mendes@ufc.br

Onde encontrar a matéria?

- Capítulos 9 e 10



Aula passada...

O que é avaliação de IHC?

Por que avaliar?

Tipos de Avaliação

O que avaliar?

Testes com usuários

Avaliação em laboratório

Avaliação em ambiente real

Quando avaliar o uso de um sistema?



Métodos de Avaliação

Os métodos de avaliação de IHC podem ser classificados em: métodos de **investigação**, de **observação de uso** e de **inspeção**



investigação



observação de uso



Métodos de Inspeção

- Não envolvem a participação de usuários
- O avaliador tenta se colocar no lugar do usuário enquanto examina (ou inspeciona) uma solução de IHC
- Permite identificar problemas que os usuários podem vir a ter quando interagirem com o sistema, e quais formas de apoio o sistema oferece para ajudá-los a contornarem esses problemas
- Alguns métodos de inspeção em IHC são:
 - Avaliação heurística
 - Percurso cognitivo
 - Método de inspeção semiótica



Avaliação Heurística

- método de avaliação de IHC criado para encontrar problemas de **usabilidade** durante um processo de design iterativo
- método simples, rápido e de baixo custo para avaliar IHC, quando comparado aos métodos empíricos
- tem como base um conjunto de heurísticas de usabilidade, que descrevem características desejáveis da interação e da interface



Heurísticas

- ISO 9126 – Usabilidade
- Critérios ergonômicos – Bastien & Scapin
- Heurísticas semióticas
- 10 heurísticas de Nielsen



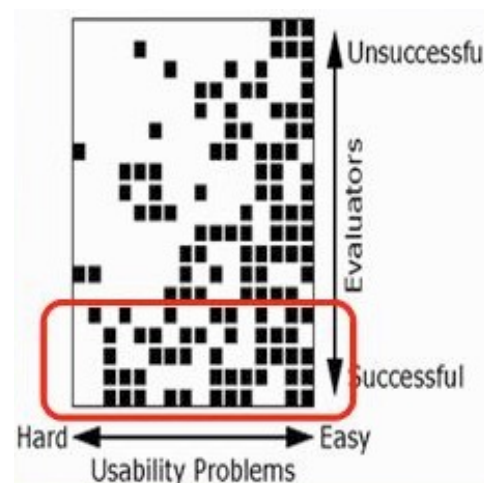
Heurísticas por especialistas

- Alguns especialistas acreditam que uma análise heurística só faz sentido quando é feita por um expert na área. Aliás, um expert em **duas** áreas: tanto em usabilidade, quanto na indústria/negócio que está relacionado ao produto, pois:
 - Um profissional com menos experiência pode acabar fazendo uma análise mais subjetiva do que a recomendada.
 - Para evitar esse problema de subjetividade, o ideal é que se utilize mais de um avaliador – uma avaliação feita por um profissional normalmente pega 20% dos problemas de usabilidade de um sistema. O número recomendado é de 3 a 5 profissionais.
 - O profissional deve estar também familiarizado com aquela indústria.
 - Caso contrário, há o risco de que as sugestões da análise heurística acabam sendo impossíveis de serem executadas, ou que elas não façam sentido do ponto de vista dos objetivos de negócio da empresa.



Por que vários avaliadores?

- Nº de problemas identificados depende da experiência dos avaliadores
- Nem todos os avaliadores detectam todos os problemas
- Bons avaliadores detectam problemas fáceis e difíceis



Avaliação Heurística



Quem avalia?

- ✓ 3 a 5 avaliadores
- ✓ pode ser necessário um especialista no domínio para esclarecer eventuais dúvidas dos avaliadores (cenários de uso típicos, listando passos que um usuário deve seguir para realizar uma amostra realista de tarefas, também podem ajudar)

Quando? Durante quanto tempo?

- ✓ nas etapas iniciais do ciclo de projeto e desenvolvimento (uso de maquetes).
- ✓ uma sessão deve durar de uma a duas horas. Caso a interface seja muito complexa, pode-se realizar mais de uma sessão.



[Barbosa, 2002; Nielsen, 1994 (a)]

Avaliação Heurística

Como?

- ✓ cada avaliador deve examinar a interface **individualmente** (avaliações independentes e livres de tendências) e gerar um relatório por escrito de suas descobertas e comentários
- ✓ o avaliador **percorre a interface diversas vezes** (pelo menos **duas**: uma para ganhar uma visão de conjunto e outra para examinar cuidadosamente cada elemento), inspecionando os diversos elementos de interface e comparando-os com a **lista** de princípios (“**heurísticas**”) de usabilidade.
- ✓ depois das avaliações terem sido feitas, os **avaliadores devem se reunir** e gerar um relatório de suas descobertas combinadas. Este relatório irá conter uma **lista dos problemas** encontrados: para cada elemento de interface, cada problema encontrado deve ser detalhado e justificado, fazendo alusão às heurísticas que foram violadas, e ter sua severidade avaliada).



Heurísticas de Nielsen

- 1) Visibilidade do status do sistema
- 2) Equivalência entre o sistema e o mundo real
- 3) Controle do usuário e liberdade
- 4) Consistência e padrões
- 5) Prevenção de erro
- 6) Reconhecer ao invés de relembrar
- 7) Flexibilidade e eficiência de uso
- 8) Estética e design minimalista
- 9) Auxílio ao usuário para reconhecer, diagnosticar e recuperar-se de erros
- 10) Ajuda e documentação.



1) Visibilidade do status do sistema

- Informar aos usuários onde eles estão
 - De onde vêm e para onde podem ir
- Informar aos usuários o que se passa
 - De forma clara e sucinta
 - Em tempo útil:
 - 0.1 s: não são precisos indicadores
 - 1.0 s: o usuário tende a divagar
 - 10 s: indicador de duração máxima para fixar atenção do usuário
 - para tempos de resposta mais demorados: indicadores de progresso

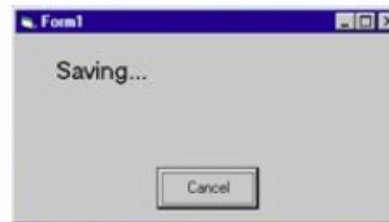


1) Visibilidade do status do sistema - exemplos

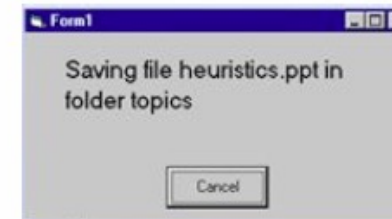
Feedback



BOM



MAU



BOM



BOM

O sistema deve sempre manter continuamente os usuários informados sobre o que está sendo feito, através de um feedback apropriado dentro de um tempo razoável.



2) Equivalência entre o sistema e o mundo real

Evite utilizar termos técnicos ou jargões desconhecidos do público alvo. A terminologia utilizada deve ser baseada na linguagem do usuário e não orientada ao sistema.

O sistema deve falar a "língua do usuário, com palavras, frases e conceitos familiares ao usuário, ao invés de termos orientados ao sistema.

As informações devem ser organizadas conforme o modelo mental do usuário.

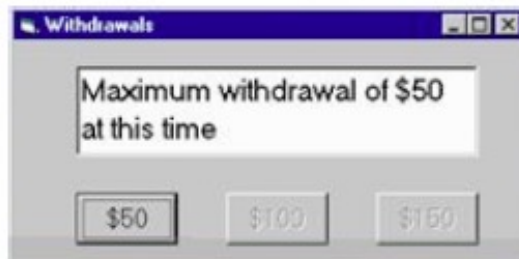
A seqüência da interação e o acesso aos objetos e operações devem ser compatíveis com o modo pelo qual o usuário realiza suas tarefas.

O sistema deve seguir convenções do ambiente real e fazer as informações aparecerem em uma ordem lógica e natural.

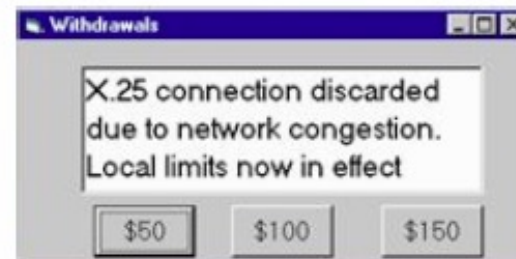


2) Equivalência entre o sistema e o mundo real - exemplos

- Usar terminologia familiar ao usuário
- Não usar termos orientados ao sistema



BOM



MAU



3) Controle do usuário e liberdade



Os usuários precisam ter a sensação de que controlam o sistema e que o mesmo responde às suas ações.

O sistema deve prover undo e redo.

As saídas devem ser claramente demarcadas.

O usuário controla o sistema, ele pode, a qualquer momento, abortar uma tarefa, ou desfazer uma operação e retornar ao estado anterior.

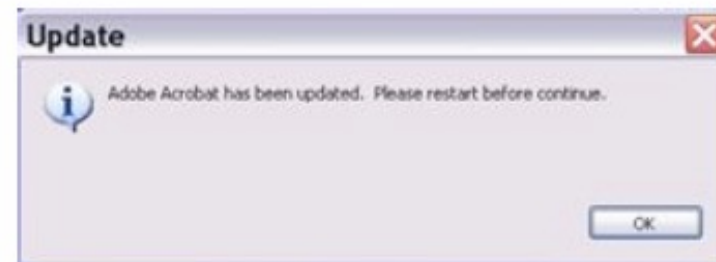
Os usuários freqüentemente escolhem funções erradas e necessitam de clareza nas opções de "saídas de emergência" sem ter que atravessar um extenso diálogo.

3) Controle do usuário e liberdade - exemplos

- Oferecer meios para sair de situações inesperadas (erros)



BOM



MAU

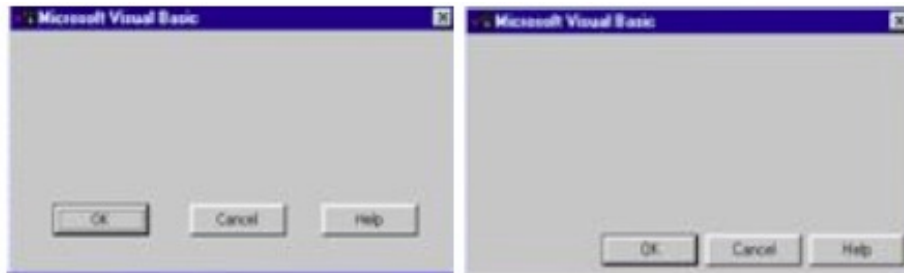


4) Consistência e padrões

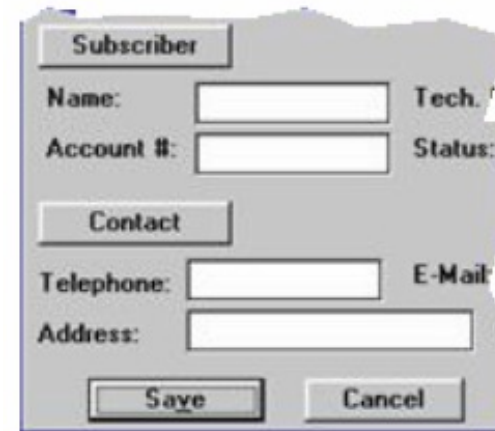
- Os usuários não se devem preocupar quando é que diferentes palavras, situações ou ações significam a mesma coisa. Sigam as convenções da plataforma
 - WWW -Elos (*links*) de cor azul (“toda” a gente usa)
 - Diálogos do Windows
 - O mesmo elemento da interface com significado diferente (botão de Cancel)
- Efeitos consistentes
 - Comandos e ações com o mesmo efeito em situações equivalentes - sistema previsível
- Definir terminologia, cores, localização de elementos, etc...
- Respeitar esta definição em toda a IU



4) Consistência e padrões - exemplos



MAU



MAU



MAU



Ícones de alguns canais e serviços do
provedor AOL

BOM



5) Prevenção de erro

- O sistema deve ser projetado de modo a evitar que o usuário cometa erros.
- As informações devem ser bem organizadas e os elementos “clicáveis” devem ser bem descritos para que o usuário não cometa o erro de acessar uma página, não obtendo a informação esperada e forçando-o a voltar para a página anterior para efetuar uma nova tentativa.
- Mesmo assim, caso usuário cometa algum erro, o sistema deve alertar ao usuário a incorreção, oferecendo-lhe uma forma simples e construtiva de corrigi-lo.
- A dica é conhecer as situações que mais provocam erros e modificar a interface para que estes erros não ocorram!



5) Prevenção de erro - exemplos

- “Melhor que uma boa mensagem de erro é evitar o erro”
- Minimizar o uso do teclado
 - Número de erros ~ quantidade de teclas digitadas
- Verificar valores introduzidos
 - Comprar 500 livros de IHC?

• E

A screenshot of a software window titled 'Form1'. It contains a 'Date:' label followed by a single text input field. Below this, there are three separate input fields for 'Month', 'Day', and 'Year'. The 'Month' field is set to 'May', 'Day' to '22', and 'Year' to '1997'. Below these, there are three dropdown menus for 'Month', 'Day', and 'Year', with 'May', '22', and '1997' selected respectively.

MAU

A screenshot of a software window titled 'Appointment'. It has tabs for 'General', 'Attendees', 'Notes', and 'Planner'. The 'When' section has 'Start' and 'End' time pickers set to '8:30AM' and '4:30PM' respectively, and a date picker set to 'Wed 5 /14 /97'. There is an 'All day' checkbox. The 'Description' field contains 'Smart Technology Sem'. A calendar pop-up is visible, showing the month of May 1997, with the 14th highlighted. The 'Where' field is at the bottom.

BOM



6) Reconhecer ao invés de relembrar

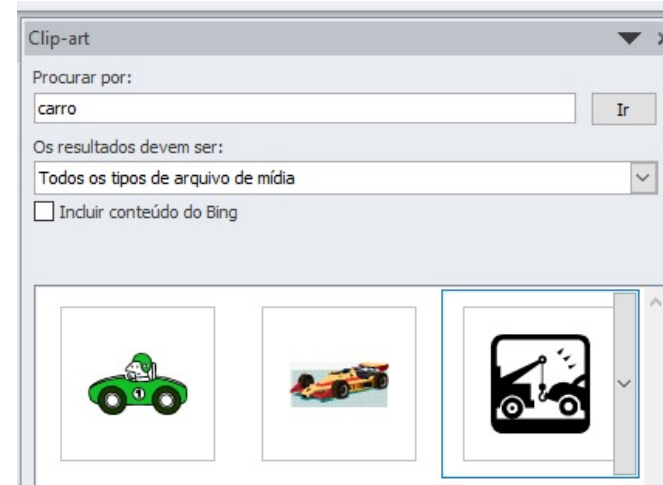
- Tornar objetos, ações e indicações
 - Visíveis
 - Fáceis de identificar e recuperar
 - Mesmo sem nunca os ter visto antes entende-se o seu significado
- Maus exemplos:
 - Ícones sem significado
 - Nomes mal escolhidos
 - Indicações insuficientes
 - Ações mal identificadas



6) Reconhecer ao invés de lembrar - exemplos

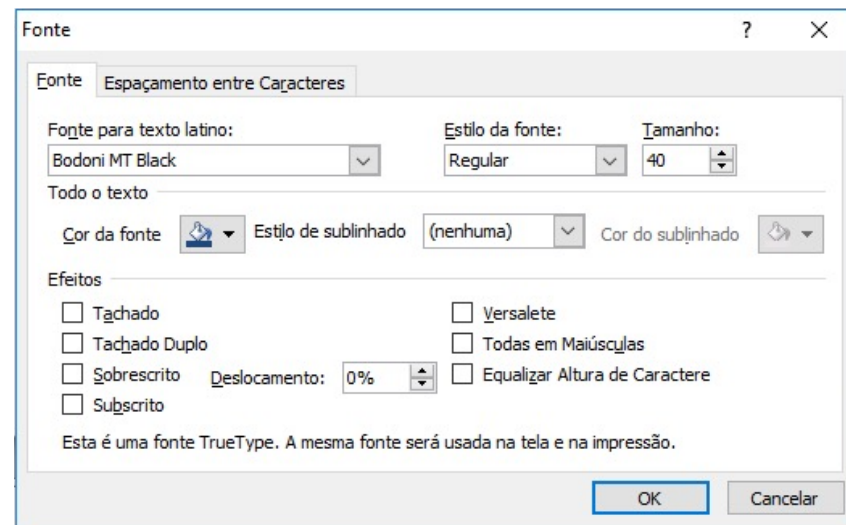


BOM



BOM

COMO É MESMO
ESTA FONTE?



MAU

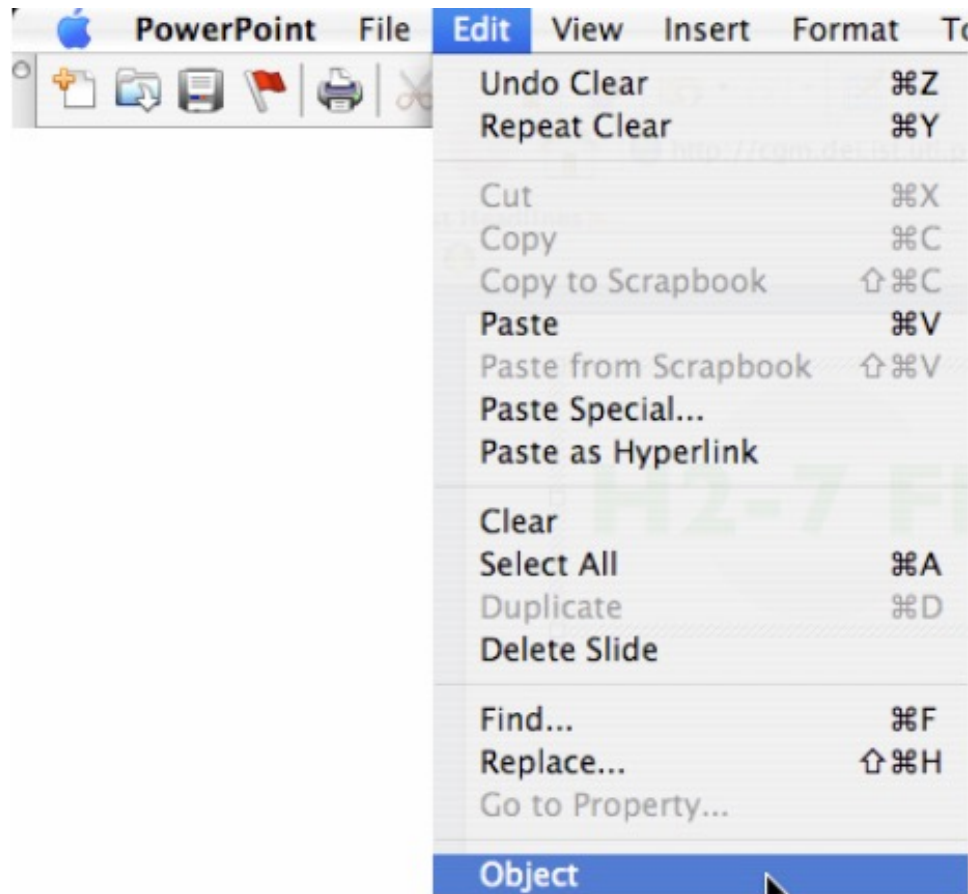


7) Flexibilidade e eficiência de uso

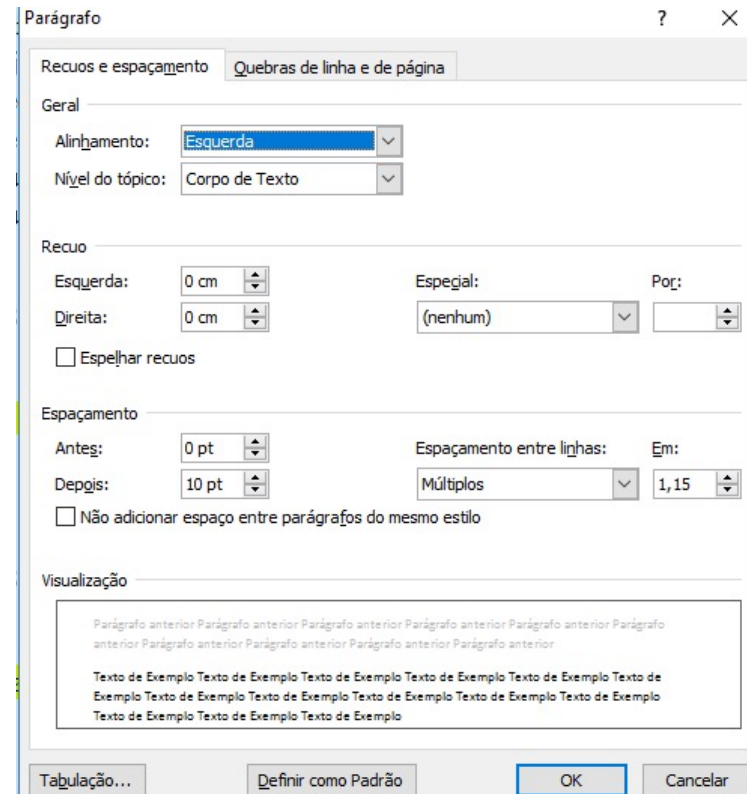
- O sistema pode atender as necessidades de ambos usuários inexperientes e experientes.
Para que os usuários mais experientes acessem a informação desejada com um menor número de cliques.
- O sistema deve permitir ao usuário ajustar suas ações freqüentes (personalização).
- Atalhos (Aceleradores)
Atalhos servem para usuários experientes agilizar a interação, executando as operações mais rapidamente.
Abreviações, teclas de função, duplo clique no mouse, função de volta em sistemas hipertexto.
Atalhos também servem para recuperar informações que estão numa profundidade na árvore navegacional a partir da interface principal.



7) Flexibilidade e eficiência de uso - exemplos



BOM



BOM



8) Estética e design minimalista

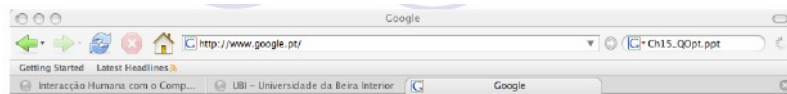
- Apresentar apenas a informação que o usuário necessita
- Menos é mais
 - Menos para aprender, para perceber mal, para distrair, etc.
- Informação deve aparecer numa ordem natural
 - Informação relacionada deve estar graficamente agrupada
 - A ordem de acesso à informação deve estar de acordo com as expectativas do usuário
- Eliminar ou esconder informação irrelevante ou raramente necessária
 - Vai competir com informação importante



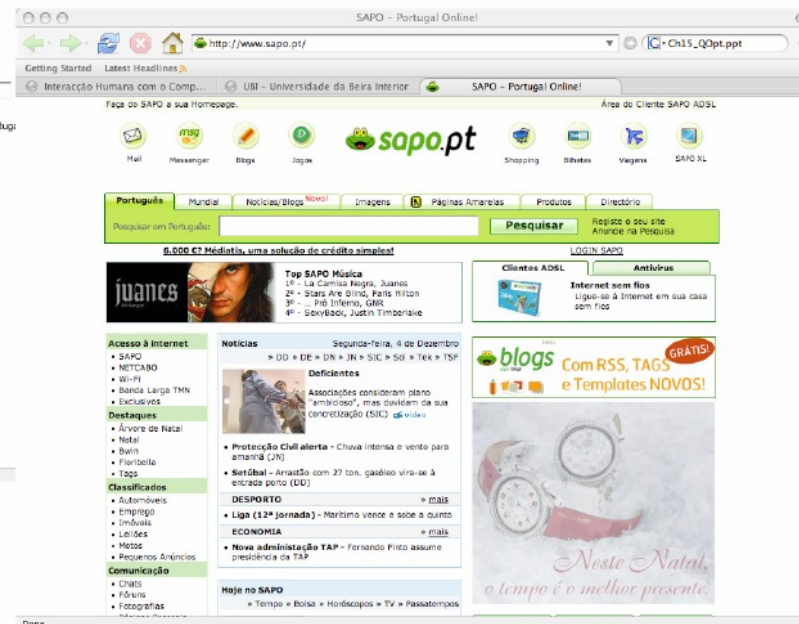
8) Estética e design minimalista - exemplos



MAU



BOM



MAU



9) Auxílio ao usuário para reconhecer, diagnosticar e recuperar-se de erros

Prover boas mensagens de erro.

A presença de mensagens explicativas deve ser um cuidado de design que previne a ocorrência de problemas.

O sistema deve ajudar os usuários a reconhecer, diagnosticar e recuperar erros.

Mensagens de erros devem ser expressas em linguagem plana (Linguagem clara), sem códigos, indicando o problema precisamente (fazendo com que o usuário entenda o que aconteceu) e sugerindo uma solução construtivamente.



9) Auxílio ao usuário para reconhecer, diagnosticar e recuperar-se de erros - exemplos



The screenshot shows a Google search interface from 2006. The search bar contains 'reatauantes Covilhã'. Below the search bar, the text 'Será que quis dizer: [restaurantes Covilhã](#)' is displayed. The main search results area states: 'Não foram encontradas páginas padrão na Web contendo todos os termos da sua consulta. A sua pesquisa - **reatauantes Covilhã** - não encontrou nenhum documento.' Below this, under the heading 'Sugestões:', there is a bulleted list of suggestions: 'Verifique que nenhuma palavra contém erros ortográficos.', 'Tente utilizar outras palavras-chave.', 'Tente palavras-chave mais gerais.', and 'Tente com menos palavras-chave.'. On the right side, under 'Links Patrocinados', there is an advertisement for 'Restaurantes' with the text 'Encontre nas Páginas Amarelas 20.000 restaurantes à sua escolha' and the link 'restaurantes.pai.pt'. The footer of the page includes 'Página Inicial do Google - Soluções de publicidade - Tudo sobre o Google' and '©2006 Google'.

reatauantes Covilhã - Pesquisa Google

http://www.google.pt/search?hl=pt-PT&q=reatauantes+Covilhã&btnG=Pesc...

Getting Started Latest Headlines

Interacção Humana com o Comp... UBI - Universidade da Beira Interior reatauantes Covilhã - Pesquisa G...

Web Imagens Grupos Notícias

Google reatauantes Covilhã Pesquisar Pesquisa Avançada Preferências

Pesquisar: a web páginas escritas em Português páginas de Portugal

Web

Será que quis dizer: [restaurantes Covilhã](#)

Não foram encontradas páginas padrão na Web contendo todos os termos da sua consulta.

A sua pesquisa - **reatauantes Covilhã** - não encontrou nenhum documento.

Sugestões:

- Verifique que nenhuma palavra contém erros ortográficos.
- Tente utilizar outras palavras-chave.
- Tente palavras-chave mais gerais.
- Tente com menos palavras-chave.

Links Patrocinados

[Restaurantes](#)
Encontre nas Páginas Amarelas
20.000 restaurantes à sua escolha
[restaurantes.pai.pt](#)

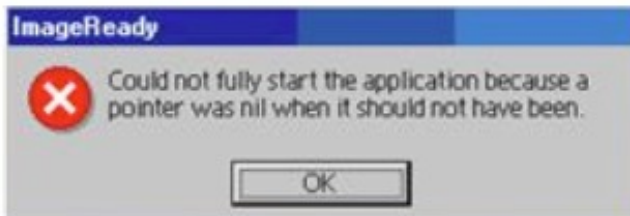
Página Inicial do Google - Soluções de publicidade - Tudo sobre o Google

©2006 Google

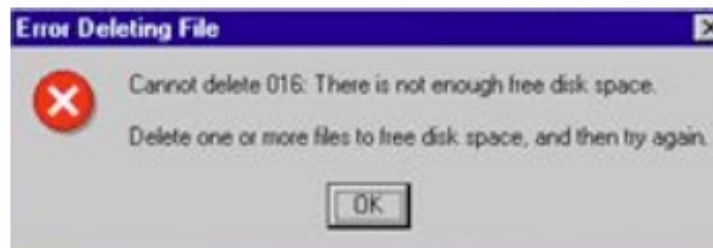
BOM



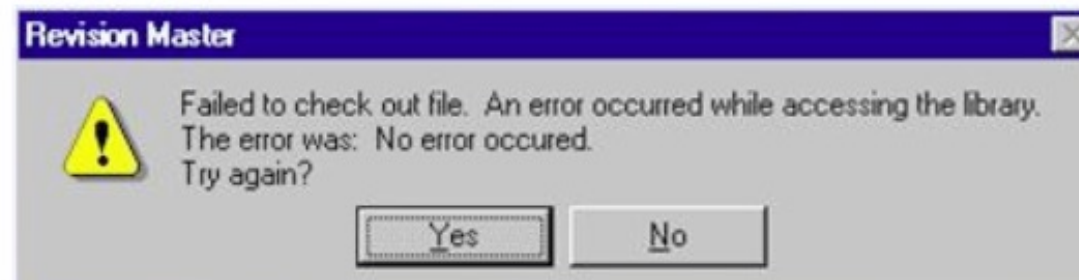
9) Auxílio ao usuário para reconhecer, diagnosticar e recuperar-se de erros - exemplos



MAU



MAU



MAU



10) Ajuda e documentação

- A ajuda não é substituto de um mau design da IU!
- A ajuda
 - Fácil de pesquisar
 - Centrada na tarefa do usuário
 - Listar passos concretos para concretizar a tarefa
 - Não demasiado extensa
 - No contexto
- O sistema não deve depender dela para ser utilizado
 - O usuário realiza as tarefas sem necessidade de a consultar



10) Ajuda e documentação - exemplos

O ideal é que um software seja tão fácil de usar (intuitivo) que não necessite de ajuda ou documentação.



BOM



BOM



Fases da avaliação heurística

1. Treino pré-avaliação

- Dar conhecimento aos avaliadores da funcionalidade
- Informação sobre cenários de interação

2. Avaliação

- Individual, seguida de consolidação de resultados

3. Classificação de severidade

- Determinar a gravidade de cada problema (prioridade)
- Pode fazer-se 1º individualmente e depois em grupo

4. Relatar

- Discutir resultados com equipe do projeto



Como conduzir a avaliação?

- Pelo menos dois passos por avaliador
 - Primeiro para familiarizar com aplicação
 - segundo para focar em elementos específicos
- Cada avaliador produz lista de problemas
 - Explicar com referência à heurística relevante ou outra informação
 - Ser específico
 - Listar cada problema em separado
 - Sugerir solução



Exemplo de problemas

- Problema: Campo da data não indica formato
 - Viola “H2-5: Evitar erros”
 - Correção: Substituir campo por um calendário
- Problema: Tipografia mistura letra maiúscula e minúscula e tipos
 - viola “H2-4: Consistência e Adesão a Normas”
 - atrapalha usuários
 - Correção: usar um só tipo em toda a interface



Graus de severidade

- Permitem atribuir recursos à solução de problemas
- Estimativas de esforço a investir em usabilidade
- Combinam
 - frequência do problema (quantas vezes)
 - impacto do problema (quão grave cada ocorrência)
 - persistência do problema
 - Isolado - só acontecendo uma vez
 - Repetitivo
- Calculados depois de consolidar avaliações
- Efetuados independentemente por todos os avaliadores



Graus de severidade

- 0 -não há consenso quanto a problema de usabilidade
- 1 -problema cosmético
- 2 -problema menor
- 3 -problema importante de usabilidade -corrigir
- 4 -CATÁSTROFE de usabilidade – imperativo corrigir!



Como relatar?

- Sessão com avaliadores, observadores e equipe de projeto
- Discutir características gerais da IU
- Sugerir possíveis melhoramentos para resolver principais problemas de usabilidade
- Equipe de projeto avalia custos de corrigir cada problema
- Sessão de *brainstorming*
 - minimizar críticas negativas durante o exercício



Ferramentas

- Exemplo de uma ferramenta de avaliação:
<http://uireports.herokuapp.com/>





Exemplos de avaliações



Atividade - Avaliação Heurística

2,0 pontos na prova prática

1. Avaliação da usabilidade de **um sistema** usando as 10 heurísticas de Nielsen. Deverá ser utilizado o template disponibilizado pela professora.
2. Sistemas a serem avaliados: Sites ou Aplicativos mobile de saúde / fitness
3. Equipes de até 3 pessoas.
4. Data de envio pelo sigaa: 29/06



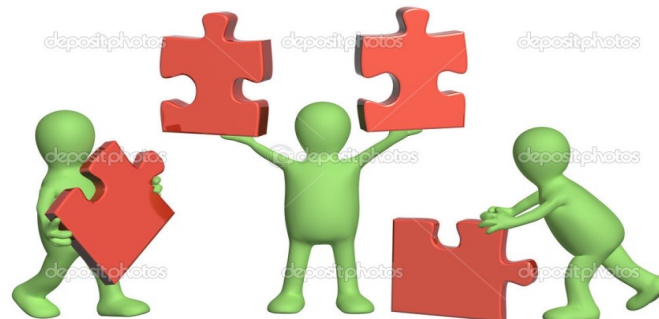
Atividade 10 - Avaliação Heurística

procedimentos

- a) Escolha um sistema dos listados
- b) Cada participante deverá realizar sua avaliação heurística usando o template disponibilizado
- c) Depois, os três componentes se reunirão e finalizarão o relatório conjuntamente.

OBS: Deve ter figuras das telas onde os problemas foram identificados.

- Cada equipe só deverá enviar um relatório.



Referências usadas nesta aula

- Estes slides foram parcialmente adaptados do Livro Interação Humano-Computador (Barbosa e Silva, 2010) – IHCCampuslides
- E do material da professora Maria Elizabeth Sucupira Furtado – coordenadora do laboratório de pesquisa LUQS – Laboratório de estudos do Usuários e da Qualidade de uso de Sistemas - <http://luqs.unifor.br>
- Livros utilizados:
 - BARBOSA, S.D.J.; SILVA, B.S. Interação Humano-Computador. Série SBC, Editora Campus-Elsevier, 2010.
 - PREECE, J.; ROGERS, Y.; SHARP, H. “Design de interação: além da interação homem computador”. Viviane Possamai (Tradução). Porto Alegre: Bookman, 2005.
 - NIELSEN, Jakob. Projetando Websites. 1957. Tradução de: GIBSON, Ana. Rio de Janeiro: Campus, 2000.

