

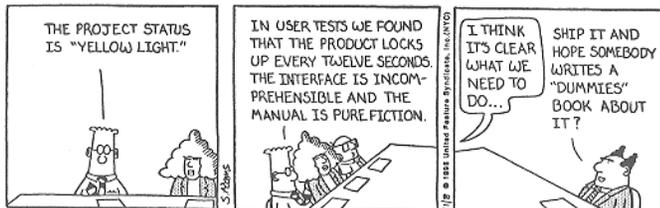
Avaliação heurística

Nielsen

<http://www.useit.com/papers/heuristic/>

Carlos Hitoshi Morimoto

Departamento de Ciência da Computação – IME/USP
2o Semestre de 2007



Introdução

- Para que serve avaliação?
 - Entender o ambiente
 - O sistema é apropriado ao ambiente de trabalho?
 - Comparar projetos
 - Com competidores e entre alternativas
 - Implementação dos requisitos
 - X% dos usuários são capazes de clicar na primeira tentativa
 - Conformação com o projeto
 - Onde está o botão cancel?

Objetivos

- Descrever a importância do processo de avaliação
- Definir o que são as técnicas de desconto
- Avaliação heurística
 - Descrição do processo
 - Aplicação do método

Quando e como avaliar?

- Nos primeiros estágios para:
 - Prever a usabilidade
 - Garantir que os projetistas entendem os requisitos de usabilidade
 - Testar idéias rapidamente e informalmente
- No final para:
 - Identificar as dificuldades do usuário (refinamento)
 - Melhorar o produto (upgrade)

Métodos de desconto



- São baratos
 - Não é necessário equipamentos ou laboratórios especiais
 - Quanto mais cuidadoso você é, melhor fica
- Rápidos
 - Leva menos de 1 dia, enquanto outros métodos podem levar semanas.
- Fáceis de usar
 - Pode ser aprendido em algumas horas.

Exemplos de técnicas



- Caminhada cognitiva usando cenários
 - Imagine-se no lugar do usuário
 - Semelhante a depuração de código
 - Protótipos de baixa fidelidade (sketches)
- Pensando em voz alta
 - Simplificação, sem o uso de psicólogos e câmeras de vídeo do método tradicional
- Avaliação heurística

Avaliação heurística (AH)

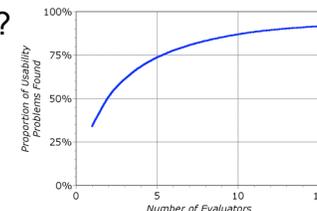


- Autor: Jakob Nielsen
- Ajuda a identificar problemas de usabilidade em um sistema interativo
- Método:
 - Usa um pequeno grupo de avaliadores (3-5)
 - Cada um avalia o sistema independentemente segundo regras heurísticas
 - Avaliadores encontrarão problemas distintos
 - Suas opiniões são juntadas mais tarde
- Pode ser utilizado em sketches

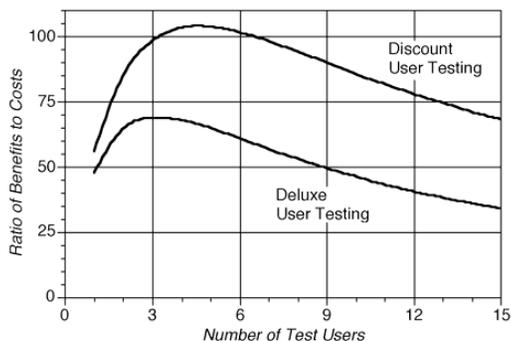
Porque vários avaliadores?



- Um único avaliador não é capaz de encontrar todos as falhas
 - Estudos indicam que um avaliador pode encontrar 35% dos problemas
- Avaliadores encontram problemas diferentes
 - 5 avaliadores encontram 75% dos problemas de usabilidade
- Por que não 10 ou + avaliadores?
 - Custo
 - Mais avaliadores não encontram muitos problemas novos



Retornos decrescentes



Heurísticas

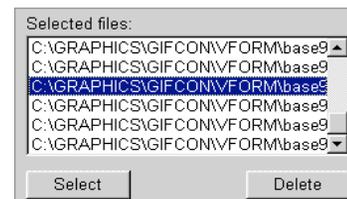
- 1: Visibilidade do estado do sistema
- 2: relação entre mundo real e sistema
- 3: controle e liberdade
- 4: Consistência
- 5: prevenção de erros
- 6: reconhecimento e não recordação
- 7: flexibilidade e eficiência de uso
- 8: design estético e minimalista
- 9: ajuda a reconhecer, diagnosticar e recuperar de erros
- 10: Ajuda e documentação

Descrição do Processo de AH

- Os avaliadores testam a interface várias vezes
 - Inspeccionando os vários elementos da interface
 - Comparando com uma lista de heurísticas
 - Considerando outros princípios/resultados de ações que o avaliador possa imaginar

1: visibilidade do estado do sistema

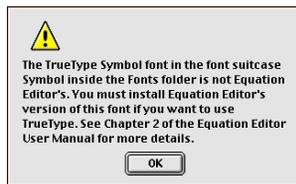
- Mantenho o usuário informado sobre o que está acontecendo
 - Exemplo sobre tempo de resposta:
 - 0.1s: não precisa fazer nada
 - 1.0s: o usuário sente descontinuidade na interação
 - 10s: máxima duração para manter o usuário focado na atividade. Use barras de progresso.
 - O feedback está Apropriado?



2: Relação entre sistema e mundo real



- Coerência entre mundo real e o sistema
 - Use a linguagem do usuário
 - Siga convenções do mundo real
 - Exemplo: Desktop do Macintosh
 - Arrastar o disquete para a cesta de lixo
 - ruim



4. Consistência e padronização



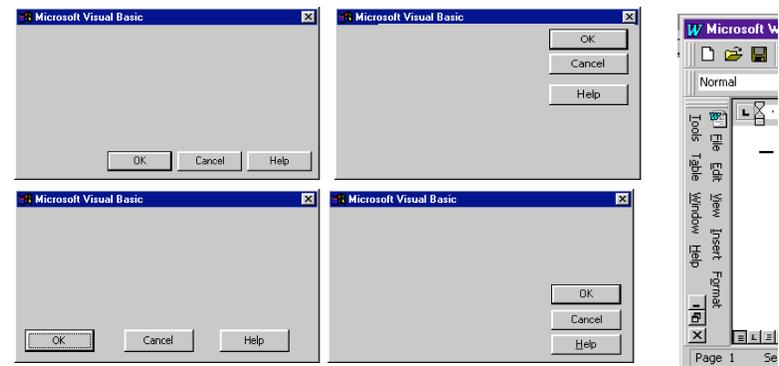
- Os usuário não podem ficar pensando se palavras, situações ou ações tem outro significado em situações diferentes
 - Exemplos:
 - botão +/- em controle remoto
 - file not found
 - Search for files or folders

3: usuário tem liberdade e controle



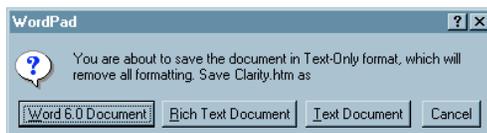
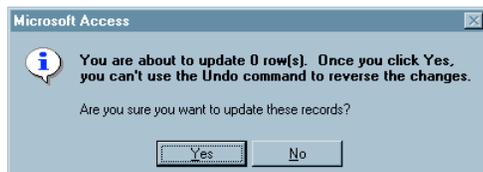
- O sistema oferece formas fáceis de escapar de situações indesejadas?
 - Correção fácil para escolhas (ações incorretas), undo, redo
- Wizards
 - O usuário deve responder a uma pergunta antes de prosseguir
 - Adequado para atividades infrequentes
 - Não para atividades frequentes
 - Bom para aprendizes ou usuários infrequentes

4: consistência e padronização



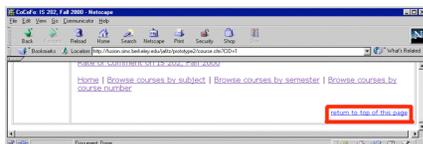
5: Prevenção de erros

- MS Access 95 após uma query sem efeito nos dados
- MS Wordpad, sempre querendo salvar no formato word, um arquivo txt por exemplo



6: prefira reconhecimento à recordação

- Faça os objetos, ações e opções visíveis e acessíveis



5. Prevenção de erros

- Faça sua interface difícil de cometer erros. Melhor que uma mensagem de erro é um design que impede que o erro aconteça!

- Exemplo:

enter date DD/MM/YY

01/11 2004

'please enter date in correct format'

alternativa:

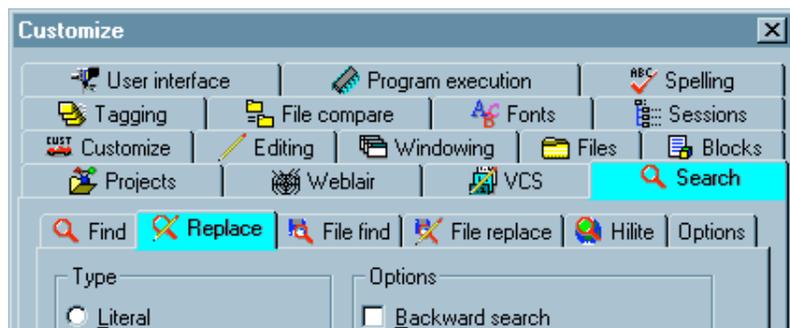
7: Flexibilidade e eficiência

- Aceleradores para experts
- Permita que usuários criem comandos para ações frequentes (macros)

Edit	Block	Help
Cut		Ctrl+X (Shift+Del)
Copy		Ctrl+C (Ctrl+Ins)
Paste		Ctrl+V (Shift+Ins)
Banner		
Transition		
Wide Palette GIF		
LED Sign		

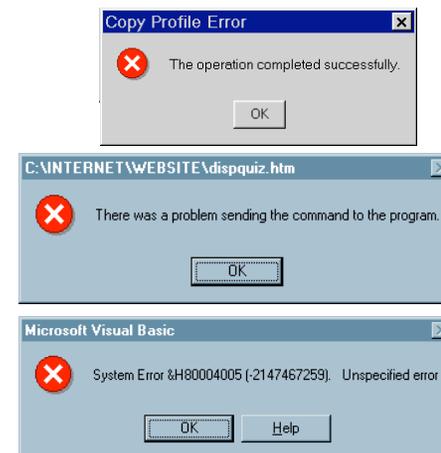
8: Estética e minimalista

- Evite informação irrelevante em diálogos. Organize a informação de forma clara e elegante.

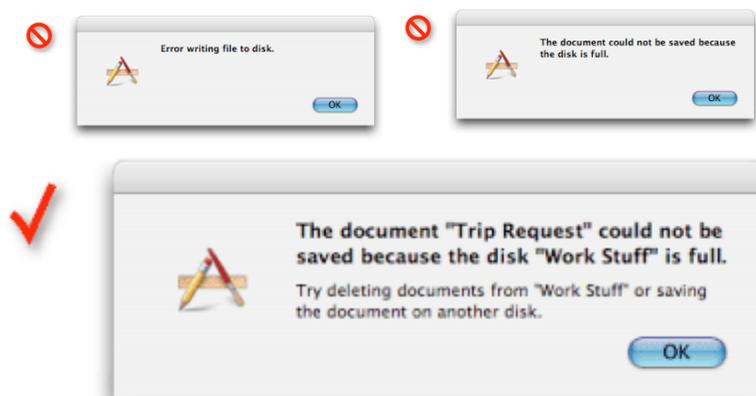


9: Recuperação de erros

- Ajude o usuário a reconhecer, diagnosticar e recuperar de erros
- Mensagens simples na linguagem do usuário
- Indicar o problema claramente
- Sugerir uma solução, ou como evitar o erro



9: Recuperação de erros



10: Documentação e help

- Fácil acesso e procura
- Focalizado na tarefa do usuário
- Clareza nos passos a serem seguidos
- Sucinto

Fases da AH



- 1) treinamento antes da avaliação
 - Forneça aos avaliadores conhecimento sobre o domínio do sistema e cenários
- 2) Avaliação
 - Cada avaliador avalia e os resultados são agrupados
- 3) Análise de severidade
 - Determina quão severo é cada problema (prioridades)
 - Pode ser feito individualmente e depois em grupo
- 4) Discussão em grupo com os projetistas

Descrição dos problemas



- Cada problema deve ser descrito separadamente
 - Se há 3 problemas com um comando, descrever todos eles
 - Assim se evita repetir problemas, na reimplementação
 - Pode não ser possível consertar todos os problemas

Processo de avaliação



- Ao menos duas vezes para cada avaliador:
 - Primeiro, para "sentir" o sistema
 - Segundo para focar em elementos específicos
- Usar cenários, se necessário
- Cada avaliador deve fazer uma lista dos problemas
 - Descrever por que é um problema, segundo uma heurística ou outra informação
 - Ser específico e descrever cada problema separadamente

Onde encontrar problemas



- Problemas graves são mais fáceis de encontrar
 - Alguns problemas simples são mais fáceis de encontrar usando AH
- Problemas podem acontecer:
 - Em um único lugar na interface
 - Em + de um lugar, mas deve-se comparar os locais para se encontrar o problema
 - Na estrutura de toda a interface
 - Algo faltando

Análise de severidades



- A análise de severidades serve para:
 - alocar recursos para uma nova solução
 - Estimar a necessidade de maiores esforços na busca de usabilidade
- Severidade é uma combinação de:
 - Frequência: comum ou raro?
 - Impacto: é fácil ou difícil de ser ignorado pelo usuário
 - Persistência: uma vez, ou continuamente?
- Deve ser calculada ao juntar as avaliações
- Deve ser feita independentemente por cada avaliador.

Fase de discussão



- Conduzida entre os avaliadores e projetistas
- Discute-se as características gerais da interface
- Sugere-se melhoramentos potenciais para solucionar os problemas principais
- Os projetistas avaliam o esforço para concertar os problemas
- Brainstorming de sugestões, soluções, alternativas, etc.

Níveis de severidade



- 0 – não concordo que seja um problema
- 1 – problema cosmético
- 2 – problema pequeno de usabilidade
- 3 – problema grande, importante concertar
- 4 – problema catastrófico, imperativo concertar

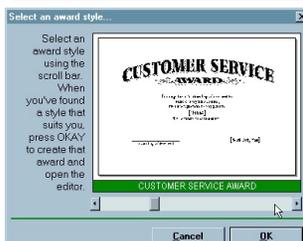
O Que você de saber:



- Definir o método de avaliação heurística.
- Quando utilizar o método de AH.
- Como utilizar o método de AH.
- Descrever cada uma das heurísticas, e identificar casos bons e ruins.

Exemplo: avaliação heurística

Ferramenta para
Impressão de diplomas



Alternativa

