



Interação Humano-
Computador

Lucia Filgueiras
2018

Avaliação heurística



Agenda

- Métodos de Inspeção
- Avaliação Heurística
- Processo de Avaliação
- Heurísticas de Usabilidade

MÉTODOS DE INSPEÇÃO

O que são métodos de inspeção?

- São caracterizadas por obterem a avaliação pelo julgamento de especialistas;
- Não envolvem os usuários finais, diretamente;
- Avaliação Preditivas: avaliam o potencial de usabilidade do produto com base em protótipos ou especificações.

Métodos de Inspeção

- Avaliação heurística (Nielsen)
- Percurso cognitivo (Wharton)
- MIS: Método de inspeção semiótica (SERG-PUC Rio)
- Avaliação de Recuperabilidade (Grupo i- Poli USP)
- Inspeções de conformidade a padrão

Avaliação heurística é o mais popular dos métodos de inspeção de usabilidade.

AVALIAÇÃO HEURÍSTICA

Histórico

- Proposta por Nielsen como método de avaliação “descontado” (econômico) comparado ao esforço dos envolvidos nos testes de usabilidade.
- Fácil de ensinar;
- Rápida de aplicar;
- Produtiva(problemas/unidade de tempo).

Definição segundo Nielsen

“Análise Heurística é um método de engenharia de usabilidade para **encontrar os erros de usabilidade** em uma interface para que sejam corrigidos em um processo de desenvolvimento iterativo.

Envolve um pequeno grupo de avaliadores para examinar a interface e avaliá-la de acordo com princípios de usabilidade reconhecidos (as heurísticas)”

Utilidade

- Quando se tem poucos recursos para avaliar o projeto;
- Quando os protótipos são de baixa fidelidade e não podem ser levados a usuários finais;
- Não serve para produzir soluções aos problemas, mas indica as guidelines que devem ser usadas.

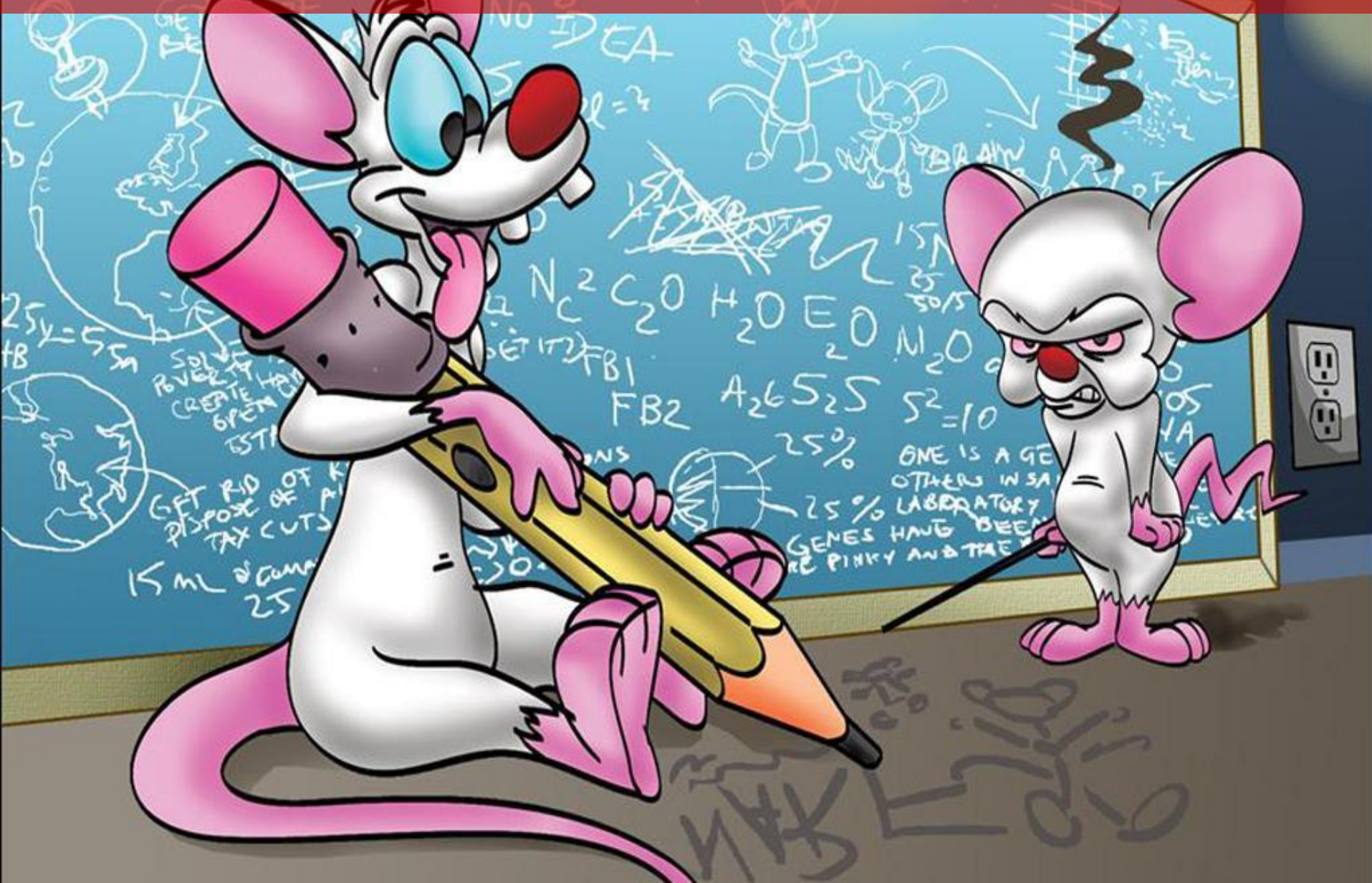
Descrição geral do processo

- Um pequeno grupo de avaliadores examina a interface e julgam o atendimento a um conjunto de princípios de usabilidade (heurísticas).
- Os resultados são compilados.
- Os resultados são analisados quanto à gravidade
- Resultado: **problemas de usabilidade**

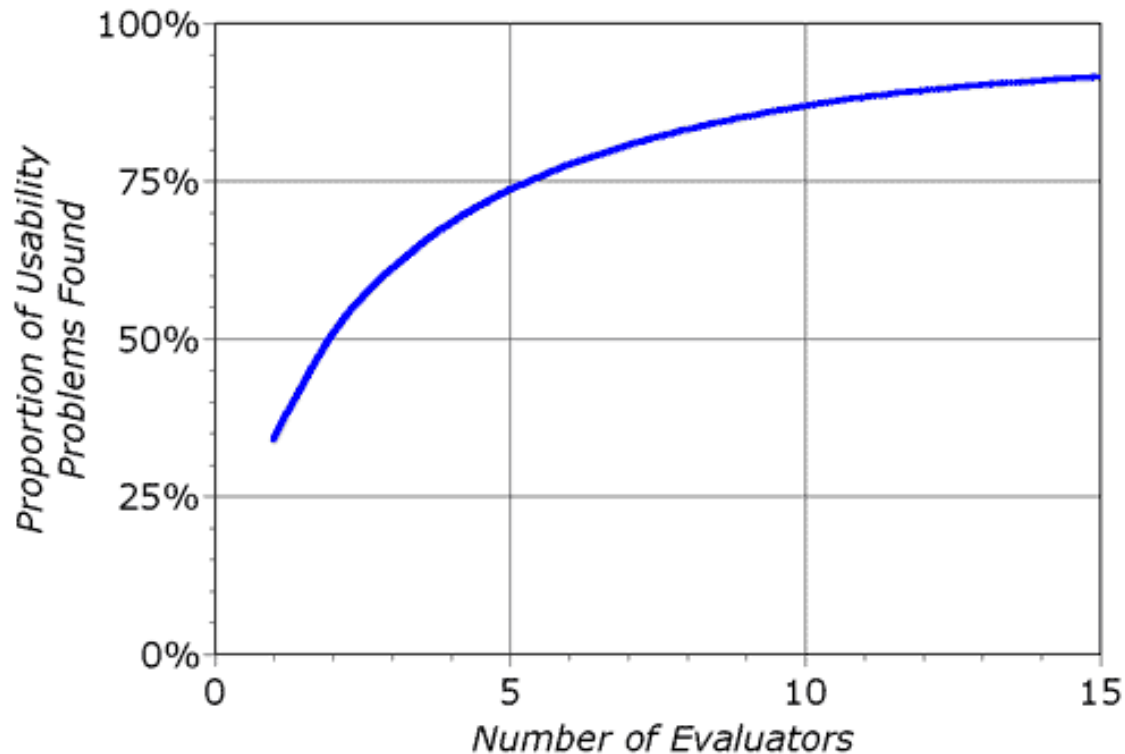
Avaliadores

- Os avaliadores devem ser capazes de pensar como usuários enfrentariam as barreiras, portanto:
 - Devem ter conhecimento no domínio da aplicação
 - Devem ter conhecimento sobre o usuário e suas reações
 - Devem ter conhecimento sobre as heurísticas de usabilidade
- **Pessoas diferentes encontram problemas diferentes!!!**

As técnicas de inspeção são fortemente dependentes das qualidades dos inspetores!



Número de Avaliadores



Um único avaliador acha apenas 35% dos problemas.

5 avaliadores encontram 75% dos problemas.

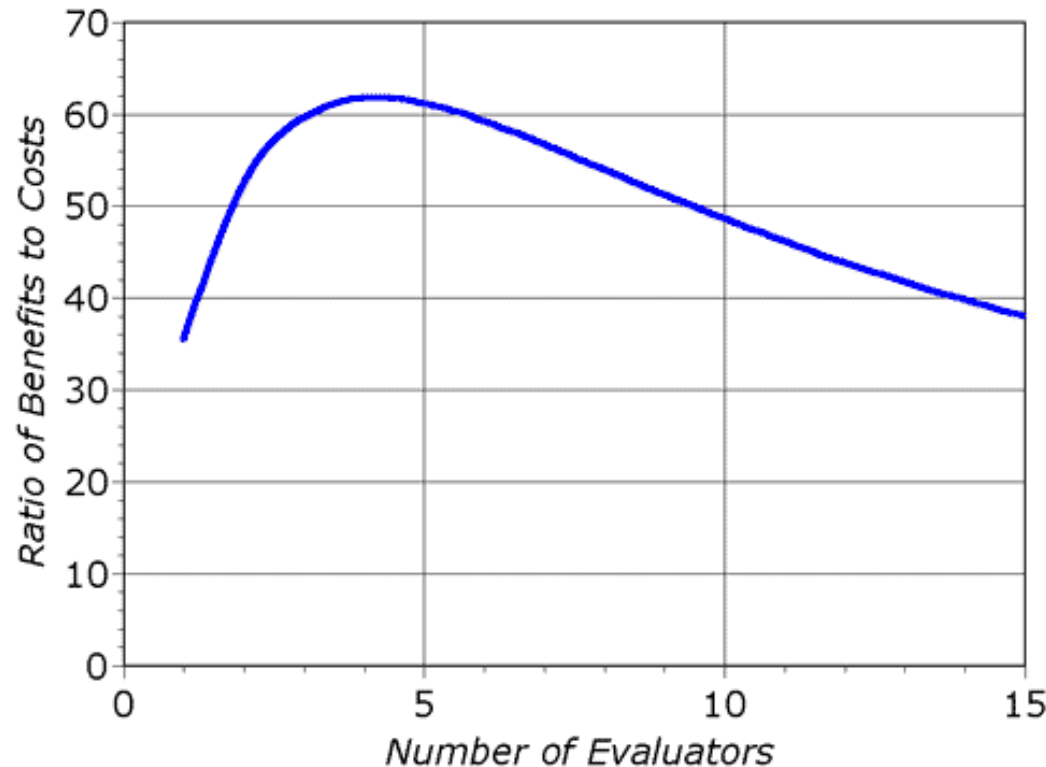
$$N(i) = N (1-(1-l)^i)$$

onde:

i: número de avaliadores;

l: proporção que um avaliador encontra

Análise de custo-benefício para identificar número de avaliadores



o número excessivo de avaliadores traz pouco retorno de investimento

PROCESSO DE AVALIAÇÃO

Treinamento pré-avaliação

- Explicação sobre o processo de avaliação heurística;
- Sessão de treinamento para explicar o domínio para os avaliadores;
- Apresentação do cenário específico sobre o qual eles vão trabalhar, identificando o que os usuários deveriam fazer e por quê.

Avaliação

- Cada avaliador examina a interface separadamente, justificando seus comentários.
- Os avaliadores recebem um tempo determinado para avaliação e percorrem a interface em dois passos:
 - no primeiro, para ter noção do fluxo da interação;
 - no segundo, para focar em elementos específicos da interface.
- Um observador pode assistir os avaliadores, fornecendo as informações e explicações necessárias.
- Os problemas de usabilidade são comunicados ao observador ou anotados pelo próprio avaliador, identificando qual princípio de usabilidade foi violado.


Debriefing

- Reunir os observadores, desenvolvedores e avaliadores para:
 - discutir os problemas encontrados;
 - as possíveis soluções.
- A discussão deve ser livre e as ideias não devem ser rebatidas (“brainstorm”).
- Classificação subjetiva de severidade pelos avaliadores.

Classificação de severidade

- Cada problema deve ser avaliado pelos especialistas em usabilidade de acordo com os fatores de gravidade:
 - O impedimento de continuidade da tarefa dá o **impacto** sobre a experiência de uso.
 - A(s) técnica(s) que aponta(m) o problema estabelece a **confiança**.
 - O número de usuários que percebem o problema dá a **visibilidade**.
 - O tempo que o problema afeta o uso dá a **persistência**.
 - A localização do problema define sua **invasividade**.
- O **custo** de cada problema deve ser analisado pelos especialistas de desenvolvimento.

Níveis de Severidade do Nielsen

- 
- 0** = não concordo que este é um problema de usabilidade.
 - 1** = **Cosmético**: não precisa ser corrigido, a menos que haja tempo extra disponível no projeto.
 - 2** = **Menor**: corrigir, mas com baixa prioridade.
 - 3** = **Maior**: importante corrigir e alta prioridade.
 - 4** = **Catástrofe**: corrigir urgente isso antes que produto seja usado.

Avaliação de Impacto



Lombada

problema que atormenta o usuário mas não o impede de atingir seu propósito.



Obstáculo

problema que demanda tempo do usuário para ser superado, mas que pode ser vencido.



Barreira

problema que impede o usuário de continuar a interação.

HEURÍSTICAS DE USABILIDADE

Regras de projeto

- As regras de projeto orientam no sentido de se produzir uma “boa” interface, em relação aos atributos de usabilidade;
- Não há provas científicas de que estas regras realmente levem a uma interface melhor, mas são resultado da experiência: **HEURÍSTICAS.**
- Podem ser gerais, específicas da categoria e específicas do produto.

Origem (1990)

As heurísticas originais surgiram da análise de **249 problemas**, identificados a partir de avaliações realizadas por especialistas em 11 projetos distintos.



As primeiras

1. Diálogo simples e natural
2. Usar a linguagem do usuário
3. Minimizar carga de memória
4. Ser consistente
5. Fornecer realimentação
6. Apresentar saídas claras
7. Fornecer atalhos e defaults
8. Dar mensagens de erro claras e construtivas
9. Evitar erros
10. Fornecer ajuda e documentação



As 10 mais famosas heurísticas de usabilidade (Nielsen)

1. O status do sistema deve ser visível (feedback)
2. Sintonia entre o sistema e o mundo real (affordance)
3. Controle e liberdade do usuário
4. Consistência e padronização
5. Prevenção de erros
6. Reconhecimento mais que memória
7. Flexibilidade e eficiência de uso
8. Design estético e minimalista
9. Ajuda aos usuários para reconhecer, diagnosticar e se recuperar dos erros
10. Ajuda e documentação



O status do sistema deve ser visível



Eu sei o que está acontecendo aqui!

O que houve?

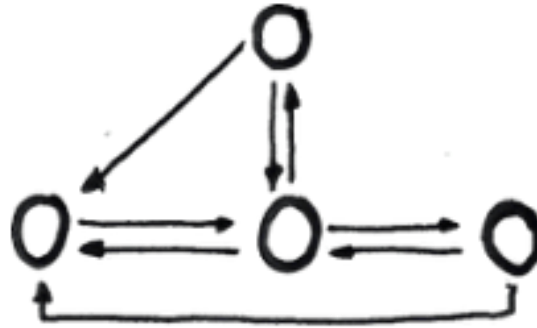
Sintonia entre o sistema e o mundo real



Eu entendo o que o sistema está dizendo!

O que significa isso?

Controle e liberdade do usuário



É assim que eu faço isso!

Como é que eu faço isso?

Consistência e padronização



Já vi isso antes!

Pra onde eu vou?

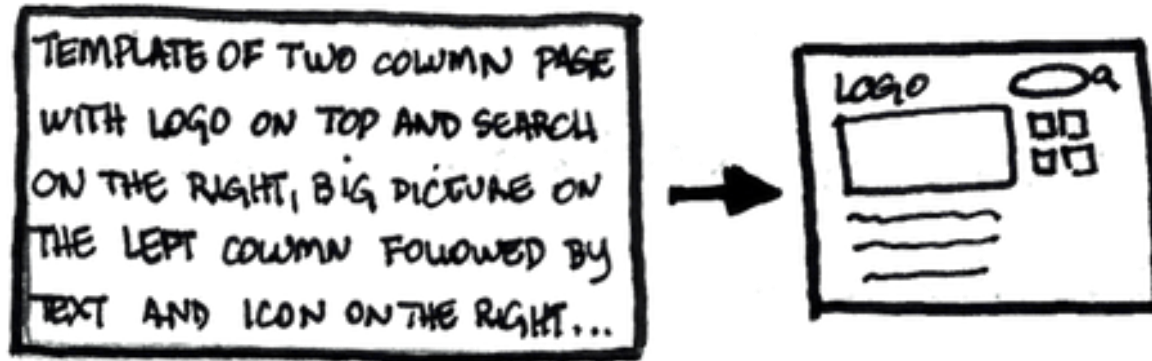
Prevenção de erros



Ufa! Quase errei!

Xii! Errei 😞 E agora?

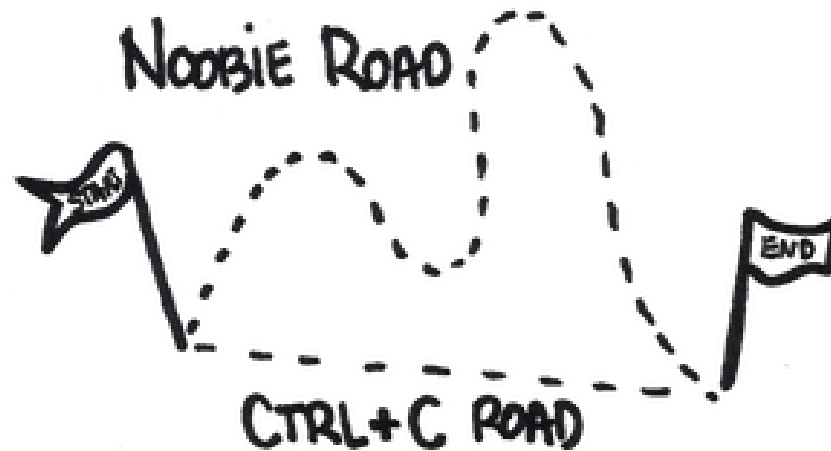
Reconhecimento mais que memória



Eu sei o que tenho que fazer agora!

Não consigo me lembrar do que tenho que
fazer...

Flexibilidade e eficiência de uso



Vou fazer de outro jeito agora!

Sou obrigado a fazer assim?

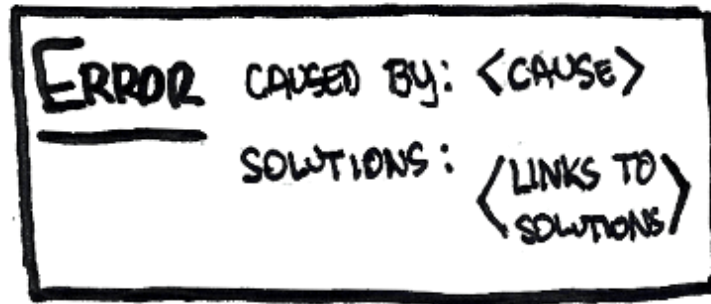
Design estético e minimalista



Uau!

Argh!

Ajuda aos usuários para reconhecer, diagnosticar e se recuperar dos erros



Sei o que errei e sei como consertar.

Estou perdido...

Ajuda e documentação



Preciso de ajuda!

Chega...

Heurísticas específicas

- Para aplicações e arquiteturas específicas:
 - Páginas Web
 - Home-pages
 - Governo eletrônico
 - Interfaces de dispositivos portáteis (mobile interfaces)
- Para documentos
- Para acessibilidade (idosos, deficientes, etc)

Checklists

- Listas de verificação, baseadas em heurísticas, orientam o trabalho do avaliador.
- ErgoList - <http://www.labiutil.inf.ufsc.br/ergolist/>

Heuristic Evaluation - A System Checklist			
1. Visibility of System Status			
The system should always keep user informed about what is going on, through appropriate feedback within reasonable time.			
#	Review Checklist	Yes No N/A	Comments
1.1	Does every display begin with a title or header that describes screen contents?	<input type="radio"/> <input type="radio"/> <input type="radio"/>	
1.2	Is there a consistent icon design scheme and stylistic treatment across the system?	<input type="radio"/> <input type="radio"/> <input type="radio"/>	
1.3	Is a single, selected icon clearly visible when surrounded by unselected icons?	<input type="radio"/> <input type="radio"/> <input type="radio"/>	
1.4	Do menu instructions, prompts, and error messages appear in the same place(s) on each menu?	<input type="radio"/> <input type="radio"/> <input type="radio"/>	
1.5	In multipage data entry screens, is each page labeled to show its relation to others?	<input type="radio"/> <input type="radio"/> <input type="radio"/>	

<http://www.stcsig.org/usability/topics/articles/he-checklist.html>