



Princípios, Diretrizes para Design e Métodos de Avaliação de IHM

Capítulo 8 e 10

Adaptado por Luciana Mara e Thiago Vilela



Barbosa e Silva 2010

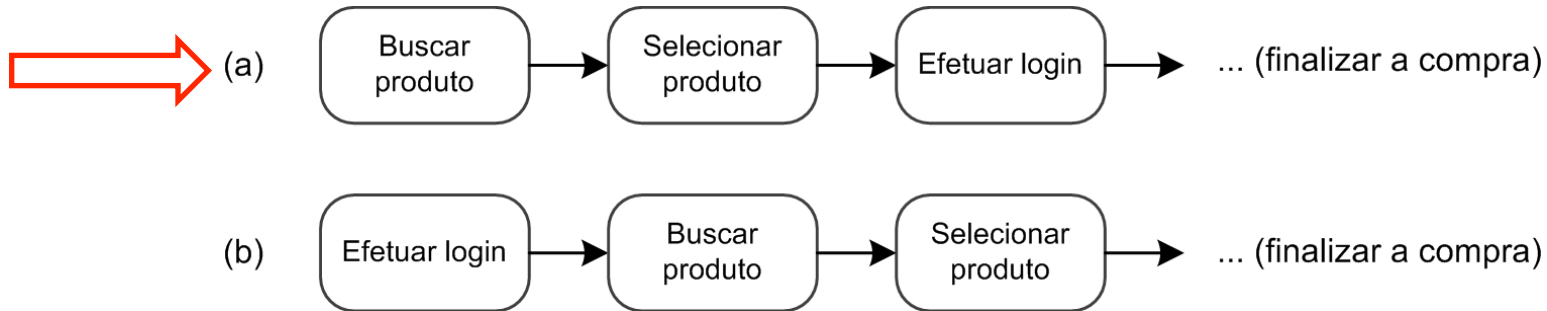
Princípios e Diretrizes de Design

- **podem auxiliar um projeto de IHC por chamarem atenção e apontarem soluções para problemas comuns na prática.**
- **entretanto, eles jamais devem substituir as atividades de análise, design e avaliação de IHC**, pois cada caso de design possui suas particularidades que não podem ser ignoradas.



Correspondência com as Expectativas dos Usuários

- explorar os **mapeamentos naturais**, seja entre as variáveis mentais e as físicas, seja entre as tarefas e os controles utilizados para manipular essas variáveis no mundo real e no sistema projetado



Por exemplo, o sistema deve permitir que o usuário encontre o produto desejado antes de se identificar, como ele costuma fazer em lojas físicas.

- estruturar a interação de forma a seguir uma **linha de raciocínio** e **fornecer um fechamento**
- seguir as convenções do mundo real, fazendo com que a informação apareça em uma **ordem natural e lógica**.



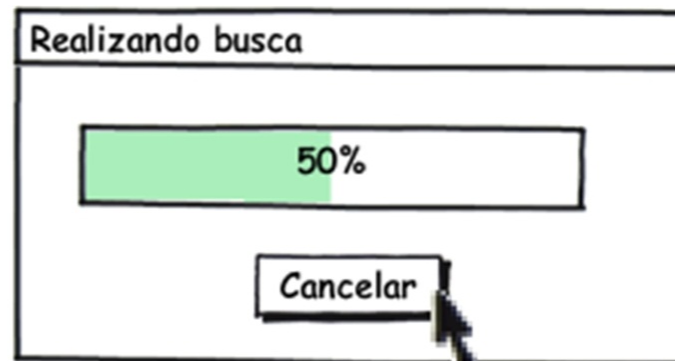
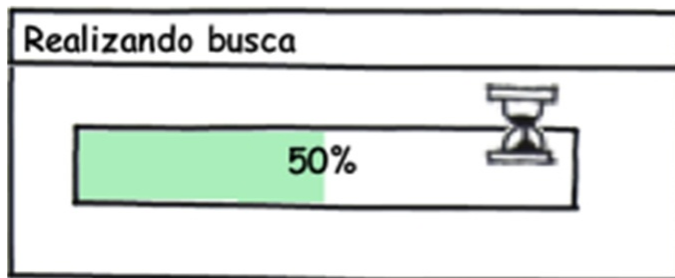
Equilíbrio entre Controle e Liberdade do Usuário

- **mantenha o usuário no controle:** o computador, o ambiente de trabalho e a interface pertencem ao usuário. Com o usuário “no comando”, ele aprende rapidamente e ganha um sentimento de **maestria (contrário de frustração)**.
- **usuários não devem ficar presos num caminho de interação único para realizar uma atividade.** O caminho mais rápido ou preferencial pode ser o de “menor resistência”, mas usuários que queiram explorar diferentes alternativas e cenários devem conseguir fazê-lo **(acionar um comando por cliques ou atalhos)**.
- **Usuários costumam querer sentir que controlam o sistema e o sistema responde às suas ações, e não o contrário.**



Equilíbrio entre Controle e Liberdade do Usuário

- permita que o usuário cancele, desfça (CTRL+Z) e refaça suas ações. Isso facilita o **aprendizado por exploração**, pois **reduz a ansiedade e o medo de errar**. Induz o usuário a explorar funcionalidades sem perder o controle...



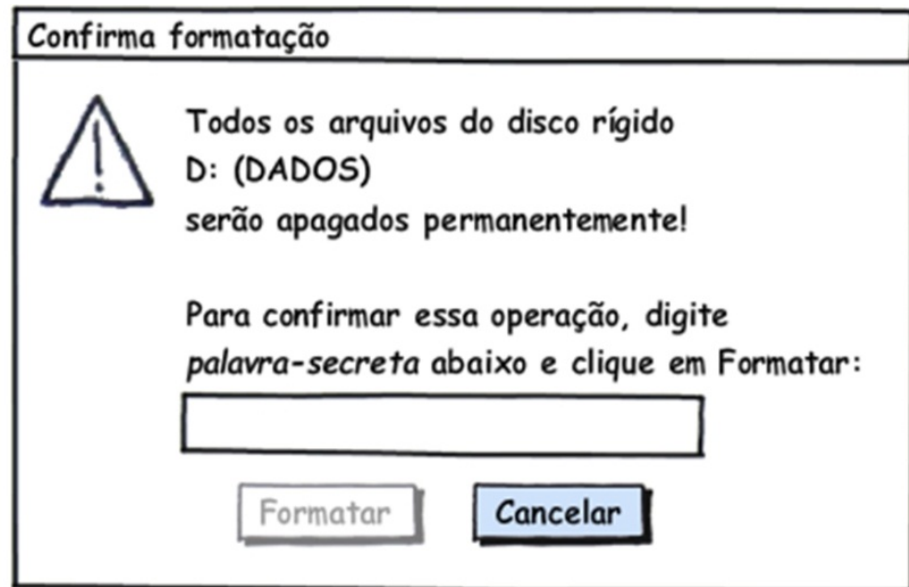
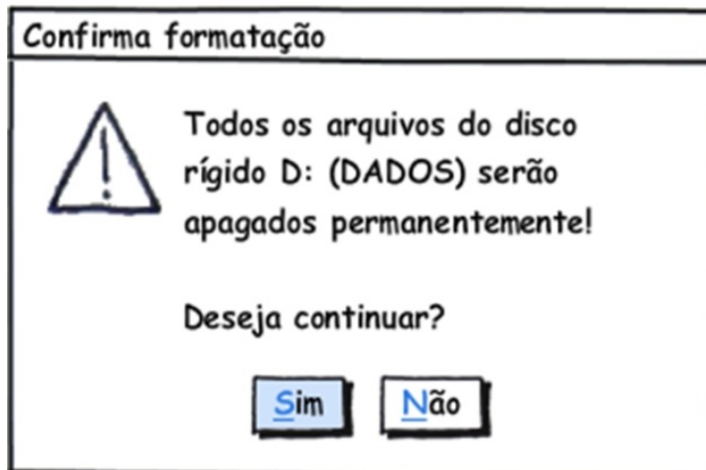
Equilíbrio entre Controle e Liberdade do Usuário

- **Usar diálogos de confirmação em excesso** não apenas aumenta o tempo de realização das tarefas, mas também pode tornar a comunicação ineficiente, pois **muitos usuários acabam prosseguindo a interação sem mesmo ler o conteúdo desses diálogos.**




Equilíbrio entre Controle e Liberdade do Usuário

- quando uma **operação considerada perigosa não puder ser desfeita**, devemos **projetar medidas de segurança** para que ela não seja acionada incidentalmente



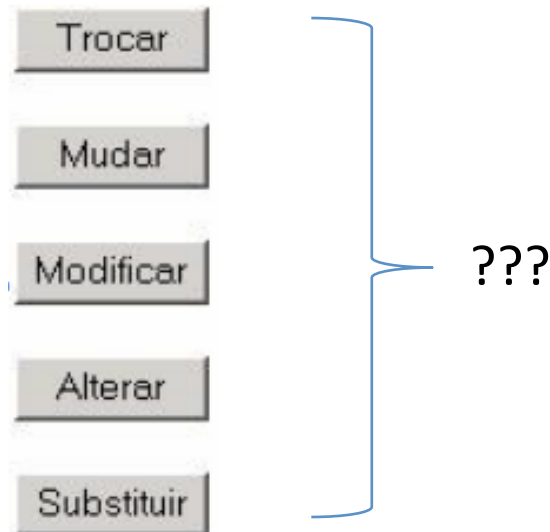
Consistência e Padronização

- **Padronize** as ações, os resultados das ações, o *layout* dos diálogos e as visualizações de informação.
- **Exemplo:** ações semelhantes devem funcionar de modo semelhante. O botão  não deve ser utilizado para fechar uma aplicação em algumas situações e para confirmar ações em outras.



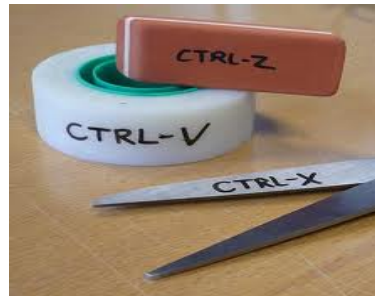
Consistência e Padronização

- O sistema deve utilizar uma **padronização de suas terminologias**.
- **Exemplo:** Utilizar rótulos distintos para o mesmo procedimento em um mesmo sistema pode confundir o usuário.



Promovendo a Eficiência do Usuário

- **Mantenha o usuário ocupado:** processamentos demorados não devem impedir o usuário de realizar outras atividades no sistema. Tais processamentos devem ser executados em *background*.
- **Proteja o trabalho dos usuários,** evitando sua perda em casos de falha no fornecimento de energia ou na transmissão da rede. **IDEAL: salvamento automático.**
- **Forneça atalhos e aceleradores.** À medida que a frequência de uso do sistema aumenta, o usuário geralmente deseja acelerar o passo da interação. Para usuários experientes estes recursos são bastante úteis.



Antecipação

- **O software deve tomar iniciativa e fornecer informações adicionais úteis**, em vez de apenas responder precisamente a pergunta que o usuário tiver feito.
- **Para isso, escolha bons valores padrão (*defaults*)**.
Geralmente campos com valores *default* já devem vir selecionados.

As pessoas tendem a aceitar os valores marcados para aquilo que elas não entendem ou que julgam ser a resposta “certa”.



Antecipação

- Considere cada alternativa. Ela é eficiente? É neutra? Ou induz a uma determinada opção?

☐ Não quero receber a newsletter semanal da Empresa

☒ Não quero receber a newsletter semanal da Empresa

☐ Quero receber a newsletter semanal da Empresa

☒ Quero receber a newsletter semanal da Empresa

Quer receber a newsletter semanal da Empresa?

☒ sim

☐ não



Quer receber a newsletter semanal da Empresa?

☐ sim

☐ não

Quer receber a newsletter semanal da Empresa? *

☐ sim

☐ não

* indica campo obrigatório



Visibilidade e Reconhecimento

- o usuário não deve ter de se lembrar de informações de uma parte da aplicação quando tiver passado para uma outra parte da aplicação; **deixar tudo que seja referenciado na mesma parte;**
- quando o usuário realiza uma ação, o sistema deve mantê-lo informado sobre o que ocorreu ou está ocorrendo, através de *feedback* (resposta do sistema) adequado e no tempo certo.



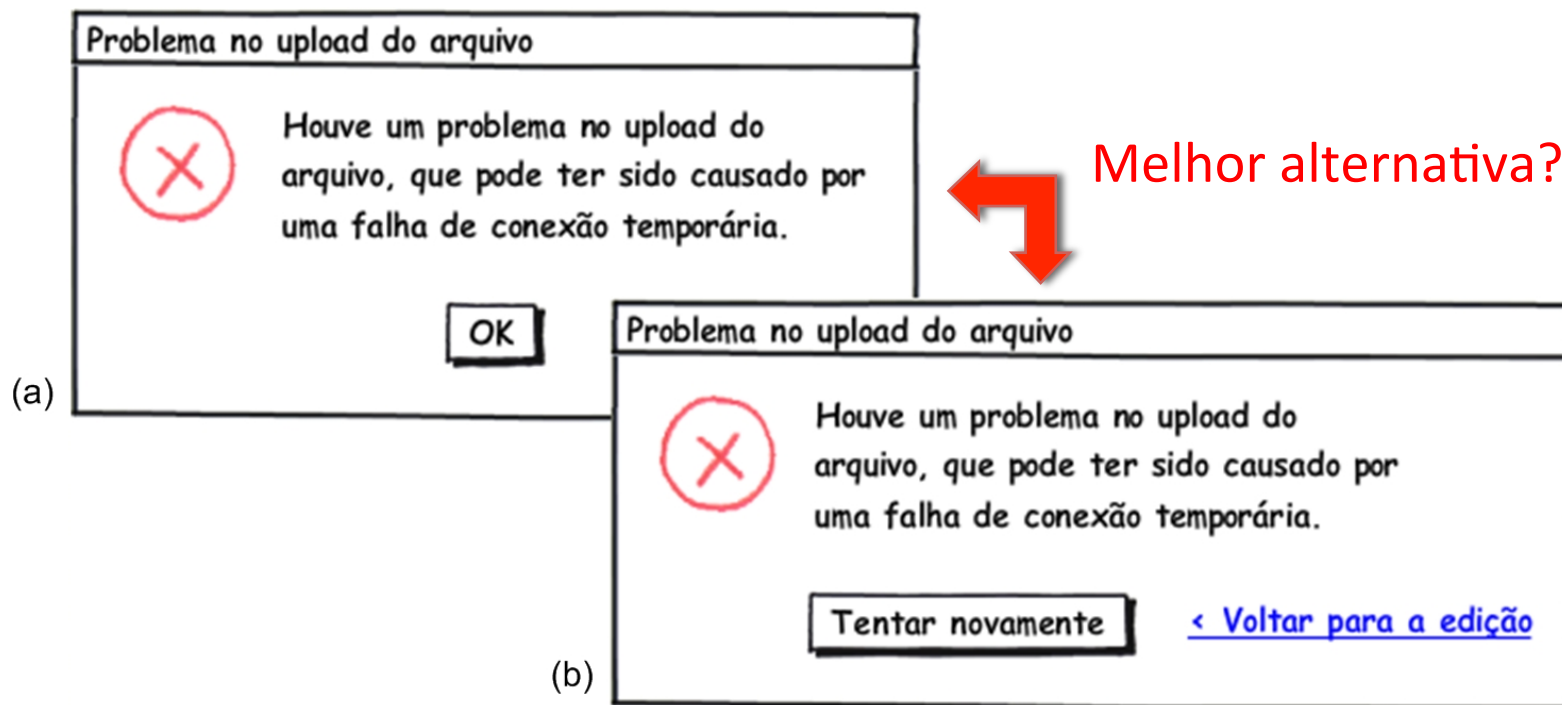
Conteúdo Relevante e Expressão Adequada

- **projeto estético e minimalista:** a informação das caixas de diálogo devem ser simples e relevantes.
- **as mensagens de instrução e ajuda** devem ser concisas e informativas.
- **os rótulos (descrições) de menus e botões** devem ser claros e livres de ambiguidade;
- **ao utilizar cores é necessário informações secundárias** (rótulos) para garantir a acessibilidade.



Projeto para Erros

- ajude os usuários a reconhecerem, diagnosticarem e se recuperarem de erros, informando-lhe sobre o que ocorreu, as consequências disso e como reverter os resultados indesejados



Métodos de Avaliação de IHC

- **Avaliação por inspeção**
 - Avaliação heurística



- **Avaliação por observação**
 - Teste de usabilidade
 - Prototipação em papel



Avaliação de IHC através de Inspeção

- Permitem ao avaliador examinar ou inspecionar uma solução de IHC para tentar **antever** as possíveis consequências de certas decisões de design;
- **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 e quais formas de apoio o sistema oferece para ajudá-los a contornarem esses problemas;
- alguns métodos de inspeção em IHC são:
 - 1. avaliação heurística
 - 2. percurso cognitivo
 - 3. 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;
- tem como base um conjunto de heurísticas de usabilidade (**diretrizes**), que descrevem características desejáveis da interação e da interface;
- Nielsen propõe um conjunto de inicial de **10 heurísticas**, que pode ser complementado conforme o avaliador julgar necessário.



Heurísticas de Nielsen (1/4)

- **1. visibilidade do estado do sistema:** o sistema deve sempre manter os usuários informados sobre o que está acontecendo através de **feedback (resposta às ações do usuário) adequado e no tempo certo.**
- **2. correspondência entre o sistema e o mundo real:** o sistema deve **utilizar palavras, expressões e conceitos que são familiares aos usuários,** em vez de utilizar termos orientados ao sistema ou jargão dos desenvolvedores. O designer deve seguir as convenções do mundo real, fazendo com que a informação apareça em uma ordem natural e lógica, conforme esperado pelos usuários.



Heurísticas de Nielsen (2/4)

- **3. controle e liberdade do usuário:** os usuários frequentemente realizam ações equivocadas no sistema e precisam de uma “saída de emergência” claramente marcada para sair do estado indesejado sem ter de percorrer um diálogo extenso. A interface deve **permitir que o usuário desfaça e refaça suas ações.**
- **4. consistência e padronização:** os usuários não devem ter de se perguntar se palavras, situações ou ações diferentes significam a mesma coisa. O designer deve **seguir as convenções** da plataforma ou do ambiente computacional. Ex.: Alterar; Editar; Modificar.
- **5. reconhecimento em vez de memorização:** o designer deve tornar os objetos, as ações e opções visíveis. As **instruções de uso do sistema devem estar visíveis ou facilmente acessíveis sempre que necessário.**



Heurísticas de Nielsen (3/4)

- **6. flexibilidade e eficiência de uso:** **aceleradores podem tornar a interação do usuário mais rápida e eficiente**, permitindo que o sistema consiga servir igualmente bem os usuários experientes e inexperientes. Ex. atalhos – CTRL+Z, CTRL+C.
- **7. projeto estético e minimalista:** a interface **não deve conter informação que seja irrelevante ou raramente necessária**. Cada unidade extra de informação em uma interface reduz sua visibilidade relativa, pois compete com as demais unidades de informação pela atenção do usuário. Ex.: caixas de diálogo excessivas.
- **8. prevenção de erros:** melhor do que uma boa mensagem de erro é um **projeto cuidadoso que evite que um problema ocorra**, caso isso seja possível. Ex.: (prevenção passiva, prevenção ativa, alerta).



Heurísticas de Nielsen (4/4)

- **9. ajude os usuários a reconhecerem, diagnosticarem e se recuperarem de erros:** as **mensagens de erro devem ser expressas em linguagem simples, sem jargões** (sem códigos indecifráveis), indicar precisamente o problema e sugerir uma solução de forma construtiva.
- **10. ajuda e documentação:** **é necessário oferecer ajuda e documentação de alta qualidade.** Tais informações devem ser facilmente encontradas, focadas na tarefa do usuário, enumerar passos concretos a serem realizados e não ser muito extensas.



Avaliação Heurística

- Resultados

→ Para cada problema identificado, o avaliador deve anotar:

- qual diretriz foi violada,
- em que local o problema foi encontrado (em que tela e envolvendo quais elementos de interface),
- qual a gravidade do problema e
- uma justificativa de por que aquilo é um problema
- também pode anotar ideias de soluções



Severidade de Problemas na Avaliação Heurística

A severidade de um problema envolve três fatores:

- a **frequência** com que o problema ocorre: é um problema comum ou raro?
- o **impacto** do problema, se ocorrer: será fácil ou difícil para os usuários superarem o problema?
- a **persistência** do problema: o problema ocorre apenas uma vez e será superado pelos usuários, ou atrapalhará os usuários repetidas vezes?

Nielsen sugere a seguinte escala de severidade:

- **problema cosmético**: não precisa ser consertado a menos que haja tempo no cronograma do projeto
- **problema pequeno**: o conserto deste problema pode receber baixa prioridade
- **problema grande**: importante de ser consertado e deve receber alta prioridade. Esse tipo de problema prejudica fatores de usabilidade tidos como importantes para o projeto
- **problema catastrófico**: é extremamente importante consertá-lo antes de se lançar o produto, pois provavelmente impedirá que o usuário realize suas tarefas e alcance seus objetivos



Avaliação Heurística

- Exemplo:

Login
Para fazer o login digite seu email ou seu CPF /CNPJ (caso tenha se cadastrado como Pessoa Física ou Jurídica, respectivamente) depois digite sua Senha e clique em Confirmar.

Email:

ou CPF/CNPJ:

Senha:

- Heurística violada: *Controle e liberdade do usuário*. Os usuários não têm a opção, através do website, de voltar à página anterior. Para isso, precisam utilizar o botão de voltar do navegador.
 - Local: ausência de botão de volta no formulário de login;
 - Severidade: 2 (problema pequeno). O usuário está acostumado a utilizar o botão voltar do navegador, e ele funciona corretamente no site.
 - Recomendação: incluir um botão *Voltar* no formulário.



Métodos de Avaliação de IHC

- **Avaliação por inspeção**

- Avaliação heurística



- **Avaliação por observação**

- Teste de usabilidade
- Prototipação em papel



Avaliação de IHC através de Observação

- permitem coletar dados sobre **problemas reais de uso** (e não apenas problemas potenciais previstos pelo avaliador através de uma inspeção), **para identificar problemas que os usuários enfrentaram.**
- Alguns métodos são:
 - teste de usabilidade
 - prototipação em papel



Teste de Usabilidade

- avalia a **usabilidade** a partir de observações de experiências de uso dos usuários-alvo.
- os objetivos de avaliação determinam quais **critérios de usabilidade** (facilidade de aprendizado, facilidade de recordação, satisfação do usuário, eficiência, segurança no uso) devem ser **medidos**.
- **Exemplo:** pode-se avaliar a **facilidade de aprendizado** medindo (através de indicadores):
 - **Quantos erros** os usuários cometem nas primeiras sessões de uso?
 - **Quantos usuários conseguiram completar com sucesso** determinadas tarefas?
 - **Quantas vezes** os usuários consultaram a **ajuda on-line ou o manual** de usuário?



Teste de Usabilidade

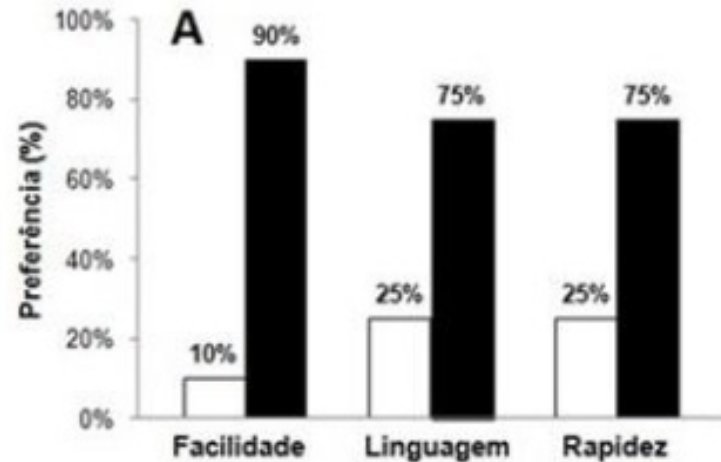
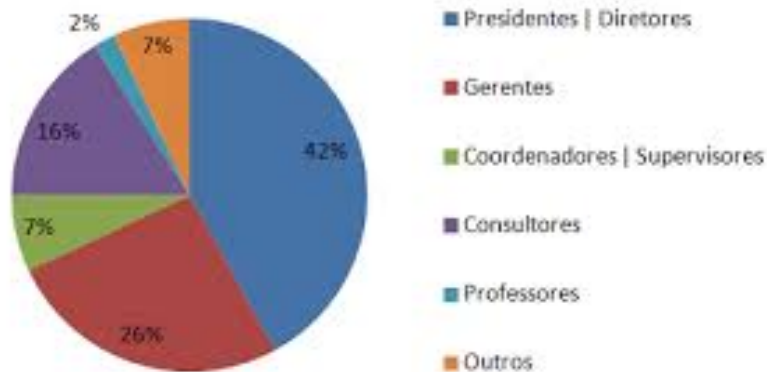
- para cada tarefa, realizada por cada participante, é possível medir:
 - o grau de sucesso da execução
 - o total de erros cometidos
 - quantos erros de cada tipo ocorreram
 - quanto tempo foi necessário para concluí-la
 - o grau de satisfação do usuário, etc.
- ênfase na avaliação do **desempenho** dos participantes na realização das tarefas e de **suas opiniões e sentimentos** decorrentes de suas experiências de uso



Exemplo

- Resultados de um teste de usabilidade devem conter:

Perfil de Participantes por Função

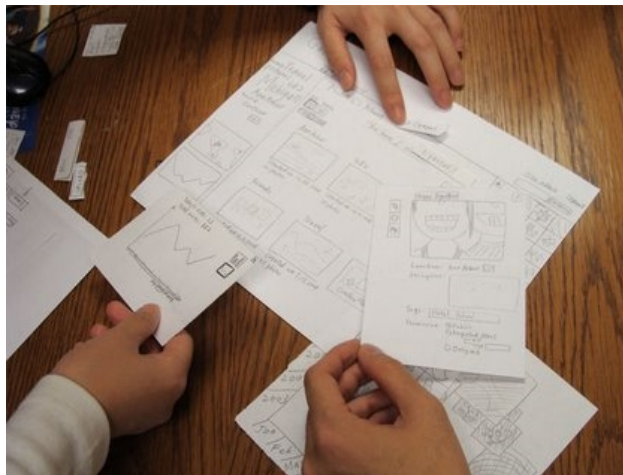


- Objetivos da avaliação,
- Número e perfil dos avaliadores e participantes,
- As tarefas executadas pelos participantes,
- Tabelas e gráficos com as medições realizadas,
- Lista de problemas encontrados com possíveis sugestões de soluções.



Prototipação em Papel

- método que avalia a **usabilidade** de um design de IHC **representado em papel**, através de simulações de uso com a participação de potenciais usuários
- é um modo rápido e barato de identificar problemas de usabilidade com a participação dos usuários, antes mesmo de construir uma solução executável



Prototipação em Papel



- os usuários simulam a execução de tarefas num protótipo em papel, falando, fazendo gestos ou escrevendo suas intenções de ação sobre o sistema
- um avaliador atua como “computador” para simular em papel a execução do sistema e expressar suas reações em resposta às ações do usuário
- outro avaliador observa e registra a experiência de uso simulada



Prototipação em Papel

