

Sobre as Heurísticas de Nielsen

Nos últimos vídeos foram apresentadas as Heurísticas de Nielsen. Jakob Nielsen é um cientista da computação que estuda a interação homem-máquina e elaborou um conjunto de heurísticas, ou seja, de regras de ouro para se projetar para ambientes digitais. Abaixo há uma lista com cada um dos tópicos:

1 - Visibilidade do Status do Sistema

O ser humano é muito dependente da visão em situações do dia a dia. No ambiente digital é mais dependente ainda já que as outras formas de interação são mais precárias. Por conta disso é primordial que a interface projetada ajude o usuário em relação à sua posição dentro de um sistema digital. Por exemplo quando o usuário acessa uma página específica de um sistema e há nela uma indicação de título assinalando a posição ao usuário. O designer deve sempre deixar com que o usuário tenha sensação de controle e não fique perdido na plataforma.

2 - Compatibilidade entre o sistema e o mundo real

É importante que um sistema digital fale a mesma linguagem do usuário. Ou seja, deve-se levar em conta a linguagem que o usuário está acostumado. Para ele conseguir se comunicar com a interface de maneira eficiente. Uma maneira de atingir esse objetivo com eficiência é utilizando ícones, que transmitem uma linguagem universal de fácil reconhecimento.

3 - Controle e liberdade para o usuário

Quando o usuário realiza uma ação por engano, deve-se apontar uma direção e uma maneira de solucionar o erro facilmente. Isso dá mais controle para o usuário apresentando uma “saída de emergência” do estado no qual ele entrou para o estado seguro. Uma maneira de simbolizar essa heurística seria por exemplo um usuário que deletou um e-mail importante por engano, mas conseguiu encontrar em sua lixeira, antes de excluir definitivamente.

4 - Consistência e Padronização

É muito importante manter a consistência das telas ao longo da interface. Dessa maneira não fica necessário que o usuário entenda diversos padrões e formas de interações diferentes para cada tela de um sistema. Com a consistência e padronização é criado um padrão replicado em todo o sistema no qual o usuário apenas precisa memorizar uma vez e isso se repetirá ao longo da plataforma inteira, criando uma melhor experiência de navegação.

5 - Prevenção de erros

O usuário diversas vezes comete erros na navegação de uma interface. Ela deve ser projetada para que auxilie o usuário a se recuperar caso tenha feito alguma ação errada. Um exemplo é quando em um sistema, o usuário comete um erro e usa a ação Control + Z do teclado.

6 - Reconhecimento em vez de memorização

É mais simples para o usuário reconhecer padrões ao invés de memorizá-los. Isso acontece pois o cérebro é ótimo em reconhecer padrões e na medida em que objetos são expostos ao usuários, mais as ações ficam familiarizadas e mais fácil ele reconhecerá tal elemento. Para isso é importante ter coesão visual na interface, para criar padrões reconhecíveis para os usuários.

7 - Eficiência e flexibilidade de uso

A interface deve ser criada tanto para usuário leigos quanto para experientes. Os leigos precisam de mais tempo e informações para executar tarefas e realizar ações. Conforme esses usuários vão conhecendo mais a interface cresce a necessidade de executar as mesmas tarefas por caminhos mais rápidos. É o caso de alguns atalhos em programas, onde por exemplo há a opção do menu de desfazer uma ação e um atalho (Control + Z) para desfazer.

8 - Estética e design minimalista

Um design minimalista garante que o usuário encontre as informações mais rapidamente sem se perder em uma plataforma. Quanto mais elementos estiverem na interface, maior a chance de se desconcentrar e deixar de realizar a ação desejada. Assim o design minimalista se torna muito eficiente em transmitir as informações de um sistema.

9 - Ajude os usuários a reconhecerem, diagnosticarem e recuperarem-se de erros

É importante mostrar para o usuário toda vez que ele comete um erro, uma maneira de se recuperar. Por exemplo em um campo de formulário onde há uma informação obrigatória e se o usuário tenta passar por ela é impedido. É interessante porque ele foi impedido de executar a ação e mostrar uma maneira de se recuperar (no caso de um campo de telefone, exibindo por exemplo: esse telefone está sem o DDD, insira por favor).

10 - Ajuda e documentação

Dentro de um sistema é importante que haja um local com ajuda. Isso faz com que os usuários se sintam seguros caso surja algum problema