

UNIP

EDUCAÇÃO, POR SUA ESCOLHA

INTERFACE HUMANO-COMPUTADOR

Avaliação de Interfaces

Prof. Me. William Tenório

william.tenorio1@docente.unip.br

Campus São José do Rio Preto - SP

Objetivo da aula



Apresentar os conceitos básicos de avaliação de interfaces e mostrar como identificar problemas de usabilidade em sistemas interativos.

Como saber se um sistema é realmente fácil de usar?

Um sistema pode:

- funcionar tecnicamente;
- executar todas as operações corretamente;
- não apresentar erros de programação;

...e ainda assim ser difícil de usar.

Perguntas importantes

- O usuário entende o que deve fazer?
- A navegação é intuitiva?
- Os elementos estão organizados corretamente?
- O sistema evita erros?
- As informações são claras?

O que é avaliação de interface?



Avaliação de interface é o processo de analisar se um sistema é:

- fácil de usar;
- eficiente;
- compreensível;
- agradável para o usuário.

A ideia é descobrir problemas antes que eles prejudiquem a experiência do usuário.

Por que avaliar uma interface?



Porque um sistema pode:

- confundir usuários;
- gerar erros;
- dificultar tarefas simples;
- causar frustração;
- diminuir produtividade.

Uma boa avaliação ajuda a melhorar:

- usabilidade;
- navegação;
- acessibilidade;
- experiência do usuário (UX).

O que normalmente é avaliado?



Facilidade de aprendizado

O usuário consegue entender rapidamente como usar?

Eficiência

As tarefas são executadas rapidamente?

Erros

O sistema evita erros? Ajuda a corrigi-los?

Satisfação

O usuário gosta de usar o sistema?

Consistência

Botões, menus e ações seguem um padrão?

Avaliação com usuários

Usuários reais utilizam o sistema enquanto observamos dificuldades.

Exemplo:

- pedir para alguém cadastrar um produto;
- observar onde a pessoa trava;
- anotar problemas.

Vantagem: Mostra problemas reais de uso.

Avaliação heurística

Especialistas analisam a interface usando regras de usabilidade. As mais famosas são as heurísticas de Jakob Nielsen.

Heurísticas de Nielsen

1



Visibilidade do status do sistema

2



Correspondência entre o sistema e o mundo real

3



Controle e liberdade para o usuário

4



Consistência e Padrões

5



Prevenção de erros

6



Reconhecimento ao invés de memória

7



Flexibilidade e eficiência de uso

8



Estética e design minimalista

9



Recuperação diante de erros

10



Ajuda e Documentação

1. Visibilidade do status

O sistema deve informar o que está acontecendo.

Exemplo:

- barra de carregamento;
- mensagem “salvando”.

2. Compatibilidade com o mundo real

Usar linguagem simples e familiar ao usuário.

Evitar termos muito técnicos.

3. Controle do usuário

Permitir desfazer ações e cancelar operações.

4. Consistência

Elementos semelhantes devem funcionar da mesma forma.

5. Prevenção de erros

Evitar que o usuário cometa erros.

Exemplo: confirmação antes de excluir.

6. Reconhecimento em vez de memorização

O usuário não deve precisar decorar informações.

Menus e opções visíveis ajudam nisso.

Exemplos simples de problemas de interface



Exemplo 1 - Botão escondido

Usuário não encontra onde salvar.

→ Problema de visibilidade.

Exemplo 2 - Mensagem técnica

“Erro SQL 500”.

→ Linguagem inadequada para o usuário.

Exemplo 3 - Menu desorganizado

Itens importantes escondidos.

→ Problema de arquitetura da informação.

Avaliação Heurística	Avaliação com Usuários
Feita por especialistas	Feita com usuários reais
Baseada em regras	Baseada em comportamento real
Mais rápida	Mais detalhada
Menor custo	Maior custo
Detecta problemas gerais	Detecta problemas reais de uso

Diferença entre testar e avaliar



Testar

Verifica se o sistema funciona tecnicamente.

Exemplo:

- botão envia dados corretamente.

Avaliar

Verifica se o usuário consegue usar facilmente.

Exemplo:

- usuário encontra o botão?

Fluxo simplificado de desenvolvimento de interface



- 1. Levantamento de requisitos** -Entender: quem é o usuário; o que ele precisa; quais tarefas realizará.
- 2. Organização da informação** - Definir: menus; navegação; estrutura; hierarquia.
- 3. Protótipos e design** - Criar: wireframes; telas; fluxo de interação.
- 4. Avaliação da interface ← HEURÍSTICAS**

É nessa etapa que analisamos:

- a interface está clara?
- os botões fazem sentido?
- o usuário entende os feedbacks?
- há risco de erro?
- os elementos são consistentes?

As heurísticas funcionam como um checklist de qualidade da interface.

Avaliar interfaces é essencial para criar sistemas:

- mais fáceis;
- eficientes;
- intuitivos;
- agradáveis.

Mesmo avaliações simples já ajudam muito na melhoria da experiência do usuário.

Uma interface pode estar funcionando tecnicamente e ainda assim ser ruim para o usuário. As heurísticas ajudam justamente a encontrar esses problemas antes que o sistema seja entregue.

UNIP

EDUCAÇÃO, POR SUA ESCOLHA

Obrigado!

Prof. Me. William Tenório
william.tenorio1@docente.unip.br
Campus São José do Rio Preto - SP