



# Universidade de Brasília

Universidade de Brasília – UnB

Faculdade do Gama UnB – FGA

Curso de Engenharia de Software

## **Exercício 1 SWEBOK – Interação Humano Computador**

Professor

André Barros de Sales

Aluno

João Pedro Sconetto – 14/0145940

## **1) O que é o SWEBOK? Por quem e para que ele foi escrito?**

R.1) O SWEBOK é um livro guia para a base de conhecimento acerca da engenharia de software, apresentado de forma a ser uma bibliografia que visa auxiliar e formalizar toda a base de conhecimento obtido e dos padrões acordados entre a comunidade de engenharia, assim como ser uma ferramenta adjunta para instrução sobre os temas abordados e documentados neste guia. O SWEBOK é um projeto da *IEEE Computer Society (Institute of Electrical and Electronics Engineers)* com edição de *Pierre Bourque* e *Richard E. Fairly*.

Página 17 (xvii - *Foreword*)

## **2) Baseado no SWEBOK v. 3.0, descreva o que é Projeto de Interface do Usuário (User Interface Design). Essa descrição deve conter:**

- Qual área do conhecimento (KA) ele está contido;
- Os princípios gerais;
- Os problemas principais;
- As Modalidades;
- A apresentação da informação;
- Processo;
- Localização e Internacionalização;
- Modelos conceituais e metáforas.

R.2) O Projeto de Interface de Usuário (PIU ou *UID*) é uma parte essencial do processo de projeto de software. Ele deve garantir que a interação entre humano e máquina forneça uma operação efetiva e controle da máquina. Para que o software atinja todo o seu potencial, a interface do usuário deve ser projetada para combinar as habilidades, a experiência e as expectativas de seus usuários antecipados.

O *UID* está contido na área de conhecimento de Design de Software (*Software Design*).

Os princípios gerais são:

- “Aprendabilidade” (*Learnability*);
- Familiaridade ao Usuário (*User familiarity*);
- Consistência (*Consistency*);
- Pouca Surpresa (*Minimal Surprise*);
- Recuperabilidade (*Recoverability*);
- Guiar o Usuário (*User guidance*);

- Diversidade de Usuários (*User diversity*).

Os principais problemas relacionados com o *UID* se relacionam com duas perguntas a qual o *UID* deve que resolver:

Como o usuário deve interagir com o software?

Como as informações do software devem ser apresentadas ao usuário?

Enquanto as modalidades de interação do software com o usuário, estas podem ser classificadas nos seguintes estilos primários:

- Resposta-Pergunta (*Question-answer*);
- Manipulação direta (*Direct manipulation*);
- Menu de seleção (*Menu selection*);
- Preenchimento de formulário (*Form fill-in*);
- Linguagem de comandos (*Command language*);
- Linguagem natural (*Natural language*).

A informação do software pode ser apresentada em natureza textual ou gráfica, um bom design mantém a apresentação de informação separada da informação em si. O modelo de arquitetura MVC (*Model-View-Controller*) é uma maneira efetiva de se manter a apresentação das informações separadas da informação em si.

A *UID* é um processo iterativo, protótipos de interface são usados várias vezes para determinar *features*, a organização e a aparência da interface de usuário do software. Esse processo envolve três atividades chave:

- Análise do usuário (*User analysis*);
- Prototipagem do software (*Software prototyping*);
- Avaliação da interface (*Interface evaluation*);

A *UID* frequentemente precisa considerar a internacionalização e a localização do software, o que significa adaptar o software para diferentes linguagens, diferenças regionais e requisitos técnicos do mercado alvo. A Internacionalização é o processo de desenhar uma aplicação de software para que esta possa ser adaptada para várias línguas e regiões sem mudanças drásticas de engenharia.

Localização é o processo de adaptar softwares internacionalizados para uma região específica ou língua adicionando componentes específicos da localização e traduzindo os textos.

Os designers de interface de usuário podem usar metáforas e modelos conceituais para configurar mapeamentos entre o software e algum sistema de referência conhecido pelos usuários no mundo real, o que pode ajudar os usuários a aprender e usar a interface com mais facilidade. Por exemplo,

a operação "excluir arquivo" pode ser transformada em uma metáfora usando o ícone de uma lixeira.

Página 54~56 (*Chapter 2 – Software Design - 4. User interface Desing*)

**3) Baseado no SWEBOK v. 3.0, descreva o que é Teste de Software. Essa descrição deve conter a definição e o objetivo dos Testes de Usabilidade e Interação Humano Computador.**

R.3) O teste de software consiste na verificação dinâmica de que um programa fornece comportamentos esperados em um conjunto finito de casos de teste, adequadamente selecionados a partir do domínio de execução geralmente infinito.

A principal tarefa da usabilidade e do teste de interação do computador humano é avaliar o quão fácil é para os usuários finais aprender e usar o software. Em geral, pode envolver o teste das funções do software que suportam as tarefas do usuário, a documentação que auxilia os usuários e a capacidade do sistema de se recuperar dos erros dos usuários.

Página 82~88 (*Chapter 4 – Software Testing – 2.2 Objectives of Testing*)

## Referência Bibliográfica

[1] *Guide to the Software Engineering Body of Knowledge, Version 3.0 – SWEBOK. IEEE Press Piscataway, NJ, USA 2004. ISBN:0769510000. Editors: P. Borque and R. E. Fairley, eds.*