

CENTRO ESTADUAL DE EDUCAÇÃO TECNOLÓGICA PAULA SOUZA

ARMANDO NAZARÉ DE OLIVEIRA

GUIA DE REFERÊNCIA PARA QUALIDADE DA USABILIDADE DE  
PROJETOS DE INTERFACES EM PROCESSOS DE  
DESENVOLVIMENTO DE SOFTWARE

SÃO PAULO  
JUNHO - 2012

ARMANDO NAZARÉ DE OLIVEIRA

GUIA DE REFERÊNCIA PARA QUALIDADE DA USABILIDADE DE  
PROJETOS DE INTERFACES EM PROCESSOS DE  
DESENVOLVIMENTO DE SOFTWARE

Dissertação apresentada como exigência parcial para obtenção do Título de Mestre em Tecnologia no Centro Estadual de Educação Tecnológica Paula Souza, no Programa de Mestrado em Tecnologia: Gestão, Desenvolvimento e Formação, sob orientação do Prof. Dr. Marcelo Duduchi Feitosa.

SÃO PAULO  
JUNHO - 2012

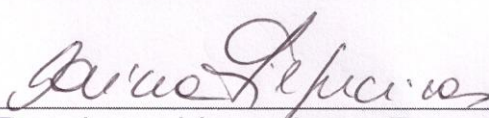
ARMANDO NAZARÉ DE OLIVEIRA

GUIA DE REFERÊNCIA PARA QUALIDADE DA USABILIDADE  
DE PROJETOS DE INTERFACES EM PROCESSOS DE  
DESENVOLVIMENTO DE SOFTWARE



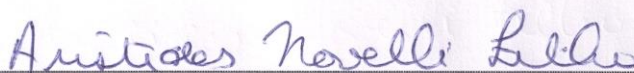
---

PROF. DR. MARCELO DUDUCHI FEITOSA



---

PROFA. DRA. LUCIA VILELA LEITE FILGUEIRAS



---

PROF. DR. ARISTIDES NOVELLI FILHO

São Paulo, 28 de junho de 2012

## **Dedicatória**

À minha família que, com certeza, foram os que mais sentiram o impacto deste meu empreendimento durante todo o tempo de dedicação como também os que mais me encorajaram a continuar até o fim.

## **Agradecimentos**

À minha família pelo total apoio e pela compreensão durante o meu projeto de pesquisa.

A todos os professores e demais funcionários do Programa de Mestrado do CEETEPS pelo carinho e profissionalismo durante esta jornada.

Aos professores Marcelo Duduchi, Aristides Novelli e Senira Fernandes, especialmente, por terem não apenas me apoiado como efetivamente contribuído do início ao fim para a elaboração deste trabalho.

*“O único lugar onde o sucesso vem antes do trabalho é no dicionário”*

*Albert Einstein*

## Resumo

OLIVEIRA, A. N. Guia de referência para qualidade da usabilidade de projetos de interfaces em processos de desenvolvimento de software. 2012. 175f. Dissertação (Mestrado), Centro Estadual de Educação Tecnológica Paula Souza.

O presente trabalho propõe um guia de referência para a verificação de qualidade do projeto de interfaces em processos de desenvolvimento de software estruturados e bem definidos, sob a perspectiva da usabilidade. Tal guia visa orientar o processo de construção de interfaces ao invés de validar apenas o produto final ou intermediário gerado, considerando que uma forma de trazer qualidade e eficácia na adoção de um *software* é por meio do foco no desenvolvimento de sua interface. Como boa parte dos conjuntos de melhores práticas foca em técnicas específicas de verificação e validação e não no planejamento de todo o processo, foi realizado um levantamento das melhores práticas de usabilidade, coletadas de normas, padrões, modelos, guias e referencial teórico. Além de apresentar o processo para elencar e consolidar as práticas para a elaboração do guia, o trabalho mostra os resultados da verificação da aplicabilidade do mesmo mediante análise de seu uso em um processo real de uma instituição que atua no desenvolvimento de sistemas. Tal análise foi feita a partir da avaliação, por especialistas da área de usabilidade, da efetividade e relevância das práticas mencionadas no guia, a coerência de sua estrutura e seu uso.

**Palavras-Chave:** usabilidade, desenvolvimento de *software*, IHC e avaliação de processos.

## **Abstract**

OLIVEIRA, A. N. Guia de referência para qualidade da usabilidade de projetos de interfaces em processos de desenvolvimento de software. 2012. 175f. Dissertação (Mestrado), Centro Estadual de Educação Tecnológica Paula Souza.

This work proposes a reference guide for quality verification of project interfaces on structured and well defined software development processes, from the perspective of usability. This guide may orient the process of interfaces development instead of validating only the final or intermediate generated product, considering that one way of bringing quality and efficacy in the adoption of software is focusing on its interface development. Considering that a great part of the sets of best practices focus on specific verification and validation techniques instead of the whole process planning, a survey of usability best practices was conducted, gathering these practices from norms, standards, models, guides and theoretical referential. Besides showing the process for listing and consolidate practices for preparing the guide, this works also shows the results of verification of the guide applicability by analyzing its use in a real process on a systems development organization. This analysis was based on the usability experts evaluation, about the effectiveness and relevance of the practices mentioned in the guide, the coherence of its structure and use.

**Keywords:** usability, software development, CHI, processes assessment.



## Lista de Figuras

|                                                                       |     |
|-----------------------------------------------------------------------|-----|
| Figura 1 - Modelo de qualidade para qualidade externa e interna ..... | 28  |
| Figura 2 - Estrutura de Usabilidade .....                             | 30  |
| Figura 3 - Atividades e respectivas saídas .....                      | 31  |
| Figura 4 - Processo de <i>design</i> centrado no usuário .....        | 33  |
| Figura 5 - Estrutura e elementos do modelo ISO/IEC 15504-5 .....      | 35  |
| Figura 6 - Um modelo simples de <i>design</i> de interação .....      | 44  |
| Figura 7 - Modelo de ciclo de vida Estrela .....                      | 45  |
| Figura 8 - Ciclo de Vida da Engenharia de Usabilidade .....           | 47  |
| Figura 9 - Guia passo a passo de avaliação de usabilidade .....       | 49  |
| Figura 10 - Processo de <i>Design</i> de Interação .....              | 50  |
| Figura 11 - Fase de Análise de Requisitos .....                       | 55  |
| Figura 12 - Fase de Projeto, Teste e Desenvolvimento .....            | 78  |
| Figura 13 - Fase de Instalação .....                                  | 113 |

## Lista de Tabelas

|                                                                              |     |
|------------------------------------------------------------------------------|-----|
| Tabela 1 - Passos do Modelo de <i>Design</i> Contextual Rápido .....         | 48  |
| Tabela 2 - Comparativo de técnicas de coleta de dados.....                   | 57  |
| Tabela 3 - Atributos de usabilidade .....                                    | 68  |
| Tabela 4 - Exemplo de medidas para propriedades desejáveis do produto .....  | 69  |
| Tabela 5 - Graus de severidade dos problemas encontrados .....               | 95  |
| Tabela 6 - Classificação dos métodos de avaliação de usabilidade.....        | 106 |
| Tabela 7 - Eficácia relativa de protótipos de baixa vs alta fidelidade ..... | 110 |
| Tabela 8 - Primeira consolidação de atividades para o guia .....             | 116 |
| Tabela 9 - Conjunto de atividades propostas para o guia .....                | 119 |
| Tabela 10 - Objetivos das atividades do guia.....                            | 121 |

## Lista de Abreviaturas e Siglas

|         |                                                                                      |
|---------|--------------------------------------------------------------------------------------|
| CARD    | Collaborative Analysis of Requirements and Design                                    |
| CHI     | Computer-Human Interaction                                                           |
| CMMI    | Capability Maturity Model Integration                                                |
| DDIU    | Design Detalhado da Interface com o Usuário                                          |
| DECIDE  | Determine, Explore, Choose, Identify, Decide, Evaluate                               |
| GOMS    | Goals, Operators, Methods, Selection Rules                                           |
| GUI     | Graphic User Interface                                                               |
| HCD     | Human-Centred Development                                                            |
| HCI     | Human-Computer Interaction                                                           |
| HMD     | Helmet/head Mounted Display                                                          |
| IHC     | Interação Humano-Computador                                                          |
| ISO     | International Organization for Standardization                                       |
| JAD     | Joint Application Development                                                        |
| JAI     | Jornadas de Atualização em Informática                                               |
| MBTI    | Myers-Briggs Type Indicator                                                          |
| MC      | Modelo Conceitual                                                                    |
| MoLIC   | Modeling Language for Interaction as Conversation                                    |
| MOT     | Metáforas do Pensamento Humano                                                       |
| OPM3    | Organizational Project Management Maturity Model                                     |
| PA      | Process Area                                                                         |
| PDT     | Padrões de <i>Design</i> de Telas                                                    |
| PICTIVE | Plastic Interface for Collaborative Technology Initiatives through Video Exploration |
| PMBOK   | Project Management Body of Knowledge                                                 |
| RAD     | Rapid Applications Development                                                       |
| SPICE   | Software Process Improvement and Capability Determination                            |
| UED     | User Environment Design                                                              |
| UMM     | Usability Maturity Model                                                             |
| WIMP    | Windows, Icons, Mouse e Pull-down menus or Pointers                                  |

## Sumário

|         |                                                                                 |     |
|---------|---------------------------------------------------------------------------------|-----|
| 1.      | Introdução .....                                                                | 14  |
| 1.1     | Metodologia de Pesquisa .....                                                   | 16  |
| 1.2     | Estrutura.....                                                                  | 17  |
| 2.      | IHC e Usabilidade .....                                                         | 17  |
| 2.1     | Interfaces e Interação.....                                                     | 17  |
| 2.1.1   | Interfaces.....                                                                 | 18  |
| 2.1.2   | Interação.....                                                                  | 19  |
| 2.2     | Interação Humano-Computador (IHC) .....                                         | 21  |
| 2.3     | Usabilidade .....                                                               | 23  |
| 3.      | Normas, Modelos e Ciclos de Vida .....                                          | 26  |
| 3.1     | Principais Referências Utilizadas .....                                         | 26  |
| 3.2     | Normas e Padrões .....                                                          | 27  |
| 3.2.1   | NBR ISO/IEC 9126-1.....                                                         | 28  |
| 3.2.2   | NBR 9241-11 .....                                                               | 29  |
| 3.2.3   | NBR ISO/IEC 12207 .....                                                         | 31  |
| 3.2.4   | ISO 13407 .....                                                                 | 33  |
| 3.2.5   | ISO/IEC 15504-5 .....                                                           | 34  |
| 3.2.6   | ISO/TR 18529.....                                                               | 36  |
| 3.3     | Modelos.....                                                                    | 38  |
| 3.3.1   | PMBOK.....                                                                      | 39  |
| 3.3.2   | CMMI .....                                                                      | 39  |
| 3.3.3   | UMM .....                                                                       | 41  |
| 3.4     | Ciclos de Vida .....                                                            | 42  |
| 4.      | Melhores Práticas de Usabilidade.....                                           | 53  |
| 4.1     | Atividades da Fase de Análise de Requisitos .....                               | 54  |
| 4.1.1   | Obtenção do Perfil do Usuário.....                                              | 57  |
| 4.1.2   | Análise de Contexto da Tarefa .....                                             | 61  |
| 4.1.3   | Estabelecimento dos Objetivos de Usabilidade .....                              | 67  |
| 4.1.4   | Levantamento das Capacidades e Restrições de Plataforma .....                   | 70  |
| 4.1.5   | Definição dos Princípios Gerais de Projeto .....                                | 70  |
| 4.2     | Atividades da Fase de Projeto, Teste e Desenvolvimento .....                    | 77  |
| 4.2.1   | Reengenharia do Trabalho .....                                                  | 80  |
| 4.2.2   | Projeto do Modelo Conceitual.....                                               | 80  |
| 4.2.3   | Elaboração de maquetes do Modelo Conceitual .....                               | 85  |
| 4.2.4   | Avaliação Iterativa do Modelo Conceitual.....                                   | 86  |
| 4.2.4.1 | Passos para a Avaliação Iterativa .....                                         | 86  |
| 4.2.4.2 | Escolha do Método de Avaliação da Interface .....                               | 87  |
| 4.2.4.3 | Método Analítico .....                                                          | 90  |
| 4.2.4.4 | Método Empírico .....                                                           | 97  |
| 4.2.4.5 | Outros métodos e abordagens de avaliação.....                                   | 103 |
| 4.2.5   | Definição dos Padrões de <i>Design</i> de Telas.....                            | 108 |
| 4.2.6   | Elaboração de Protótipos dos Padrões de <i>Design</i> de Telas.....             | 109 |
| 4.2.7   | Avaliação Iterativa dos Padrões de <i>Design</i> de Telas.....                  | 111 |
| 4.2.8   | Desenvolvimento do Guia de Estilo .....                                         | 111 |
| 4.2.9   | Elaboração do <i>Design</i> Detalhado da Interface com o Usuário .....          | 112 |
| 4.2.10  | Avaliação Iterativa do <i>Design</i> Detalhado da Interface com o Usuário ..... | 113 |
| 4.3     | Atividades da Fase de Instalação .....                                          | 113 |

|        |                                                        |     |
|--------|--------------------------------------------------------|-----|
| 4.3.1  | Obtenção do <i>Feedback</i> do Usuário .....           | 114 |
| 5.     | Construção do guia para avaliação de usabilidade ..... | 115 |
| 6.     | Guia Proposto .....                                    | 124 |
| 6.1    | Atividades do Guia .....                               | 124 |
| 6.1.1  | Atividade 1.....                                       | 124 |
| 6.1.2  | Atividade 2.....                                       | 125 |
| 6.1.3  | Atividade 3.....                                       | 127 |
| 6.1.4  | Atividade 4.....                                       | 128 |
| 6.1.5  | Atividade 5.....                                       | 135 |
| 6.1.6  | Atividade 6.....                                       | 136 |
| 6.1.7  | Atividade 7.....                                       | 138 |
| 6.1.8  | Atividade 8.....                                       | 139 |
| 6.1.9  | Atividade 9.....                                       | 140 |
| 6.1.10 | Atividade 10.....                                      | 142 |
| 6.1.11 | Atividade 11.....                                      | 143 |
| 6.1.12 | Atividade 12.....                                      | 144 |
| 6.1.13 | Atividade 13.....                                      | 145 |
| 6.1.14 | Atividade 14.....                                      | 146 |
| 6.1.15 | Atividade 15.....                                      | 147 |
| 6.1.16 | Atividade 16.....                                      | 151 |
| 6.1.17 | Atividade 17.....                                      | 158 |
| 6.2    | Método para verificação de aderência ao guia.....      | 160 |
| 6.3    | Uso e perspectivas de aplicabilidade do guia .....     | 164 |
| 6.3.1  | Ajustes no guia e no método de verificação .....       | 168 |
| 7.     | Conclusão .....                                        | 169 |
| 8.     | Referências .....                                      | 170 |

## 1. Introdução

É observado nos últimos anos um aumento da população com acesso a *softwares* (embarcados, corporativos, acadêmicos, de automação bancária, via telefonia móvel, internet e outros canais), seja para o uso doméstico, gestão de empresas, lazer, aprendizado ou outros fins. Para Cybis, Betiol e Faust (2007), no início da informatização os usuários não tinham dificuldade de operar sistemas pois eram seus próprios desenvolvedores mas, com o tempo, os *softwares* passaram a ser utilizados por não-desenvolvedores, para os quais eram aplicados intensos treinamentos. Porém, o que se restringia, há alguns anos, a ambientes corporativos, militares e acadêmicos, hoje está presente no cotidiano de diversos grupos sociais. Um desafio que se apresenta é de que os *softwares* sejam concebidos considerando a diversidade e especificidades dessa nova e crescente comunidade usuária.

Apesar dessa evolução, alguns mitos e crenças sobre a importância do foco na construção de interfaces elencadas por Mayhew (1999), ainda aparecem dominantes, como: “a qualidade da interface não importa”, “as tarefas relacionadas ao projeto de interface não ficam claras até a fase de desenho detalhado do projeto” ou “a usabilidade é subjetiva e não pode ser medida”. Algumas empresas têm a crença errônea quanto ao estudo e aplicação da usabilidade, por acreditarem que as atividades de avaliações de usabilidade vão retardar seus projetos (NIELSEN e LORANGER, 2007). Conforme Shneiderman e Plaisant (2010), interfaces eficazes podem mudar a vida de muitas pessoas. Médicos podem fazer diagnósticos mais precisos, pilotos podem pilotar aviões com mais segurança, crianças podem aprender de forma mais eficaz, portadores de deficiência podem levar vidas mais produtivas, entre outros. Segundo esses autores, algumas interfaces, no entanto, são perturbadoras, fazendo com que usuários tenham que lidar com frustração, medo e fracasso ao se deparar com menus excessivamente complexos, terminologia incompreensível e caminhos de navegação caóticos, por exemplo. A questão ganha importância ainda maior quando consideramos sistemas críticos, como os voltados às áreas de saúde, segurança e infra-estrutura pública. Após o ataque terrorista em 11/09/2001, por exemplo, membros do Congresso Norte Americano culpavam as inadequações da interface pela falha no sistema de detecção dos terroristas (SHNEIDERMAN e PLAISANT, 2010).

A fim de reverter esta situação, uma forma que pode trazer qualidade e eficácia na adoção de um *software* (maior utilização efetiva, menor curva de aprendizado e menor resistência) é por meio do foco no processo de criação da interface. Conforme a ISO 13407 (1999), a qualidade de processo contribui para melhorar a qualidade do produto e a qualidade em uso e, por isso, avaliar e melhorar o processo é um meio de melhorar a qualidade do produto. Entretanto, o referencial teórico atualmente disponível sobre usabilidade está disperso por vários formatos distintos, sendo que cada formato possui um objetivo diferente. As normas e padrões disponíveis, por exemplo, possuem enfoque destinado ao cumprimento integral de seus itens (com vistas a certificações por meio de aderência total à norma) e não como guias em que as instituições que desenvolvem *software* possam selecionar e utilizar apenas as práticas coerentes e aplicáveis ao seu tipo de aplicação. Já em relação ao referencial teórico (compêndios de melhores práticas), boa parte é focado na validação do produto e não do processo de criação dessa interface. Outro ponto é que esse referencial nem sempre aparece estruturado sob uma visão de processos ou de ciclo de vida ou ainda buscam abranger todo o processo de desenvolvimento de *software*, mas não possuem foco em usabilidade. Nesse contexto, sempre surgem dúvidas sobre qual referência entre manuais, padrões, normas e compêndios deve-se adotar para garantir a utilização de boas práticas de usabilidade nos projetos de interface e que impacte positivamente no produto final. Considerando todas essas fontes, é possível que um guia centralizado dessas práticas, orientado sob uma perspectiva diferente, auxilie os desenvolvedores a obter melhores resultados nos projetos, a partir da implementação dessas práticas no processo de criação das interfaces.

O presente trabalho propõe um guia orientado por essa perspectiva: a avaliação da qualidade do processo de desenvolvimento de interfaces com foco em usabilidade. O guia não possui objetivo de certificação ou de conformidade, portanto, não busca conferir uma pontuação ou um nível de maturidade dos processos a uma organização, mas sim um objetivo de orientação, proporcionando à organização uma visão da sua aderência às melhores práticas de usabilidade. O foco é na avaliação do processo e não do produto final ou intermediário gerado e o campo de estudos que servirá de base para a construção do guia é o da usabilidade, dentro da área de Interação Humano-Computador (IHC).

Outro ponto a destacar é que a presente pesquisa teve o foco na criação de um guia que possa ser, em um primeiro momento, principalmente adotado no cenário brasileiro, tipicamente por organizações de grande porte voltadas ao desenvolvimento de sistemas primordialmente corporativos e que já possuam metodologia e processos de desenvolvimento de software definidos e consolidados, onde as metodologias mais tradicionais são amplamente utilizadas e a usabilidade corre o risco de ser colocada em segundo plano durante o processo de desenvolvimento.

Com base nas considerações acima, a fim de se manter a aderência do trabalho ao contexto mencionado, não foi objeto desta pesquisa a construção de um guia que esteja amplamente aderente às metodologias ágeis e/ou às organizações que estejam muito distante do contexto, como empresas de porte diverso e/ou voltadas a outros segmentos (empresas de mídia, por exemplo).

Mesmo assim, abre-se com este trabalho espaço para estudos futuros sob outras perspectivas que venham a atender a adoção destas metodologias, visto que a adoção destas vêm crescendo, mesmo por parte das empresas enquadradas no contexto do presente trabalho (LACERDA, BARBOSA e RIBEIRO, 2011).

## **1.1 Metodologia de Pesquisa**

Os procedimentos que servem de base ao presente trabalho constituem-se dos passos elencados a seguir. Em uma primeira fase, foi realizado um levantamento das melhores práticas de usabilidade, coletadas de normas, padrões, modelos, guias e referencial teórico. A pesquisa bibliográfica envolveu publicações impressas e eletrônicas, principalmente do período compreendido entre 1980 e 2012. O objetivo foi identificar as atividades que poderiam compor o guia. Uma vez elencadas as melhores práticas a partir das diversas referências, o passo seguinte foi a elaboração do guia. Para isso, foram consolidadas as práticas identificadas como mais relevantes e/ou corroboradas, conforme critério adotado para a seleção destas atividades. A fase seguinte tem como objetivo a verificação da aplicabilidade do guia proposto, mediante a análise de seu uso em um processo real de uma instituição que atua no desenvolvimento de sistemas. Nesta fase, especialistas da área de usabilidade avaliaram a efetividade e relevância das práticas mencionadas



no guia, a coerência de sua estrutura e o método de avaliação propostos no presente trabalho, sugerindo correções e melhorias. O objetivo desta verificação não foi avaliar a qualidade do processo desta organização, mas sim a qualidade do guia e do método proposto pelo presente trabalho. A última fase compreende a análise dos resultados, onde são consideradas as observações levantadas na validação e feitos os eventuais ajustes no guia.

## **1.2 Estrutura**

O presente trabalho estrutura-se da seguinte forma a partir deste ponto: o capítulo dois apresenta uma visão geral de IHC e em especial de usabilidade, o eixo de IHC que é o objeto de estudo desse trabalho de pesquisa. O capítulo três apresenta o levantamento do conjunto de normas, padrões e modelos pesquisados, além de uma análise sobre o ciclo de vida a ser adotado para o presente trabalho a partir do capítulo seguinte. O capítulo quatro apresenta o levantamento do conjunto de melhores práticas de usabilidade, coletadas dos diversos compêndios, além de um maior detalhamento de algumas referências tratadas no capítulo anterior. O capítulo cinco demonstra o processo de elaboração do guia de usabilidade proposto. O capítulo seis sugere a proposta do guia de usabilidade (objeto de estudo desta pesquisa) e do método para a verificação de aderência de um processo de desenvolvimento de software às práticas do guia proposto. O capítulo sete apresenta a conclusão deste trabalho de pesquisa e considerações sobre trabalhos futuros correlacionados.

## **2. IHC e Usabilidade**

Para discutirmos a usabilidade em projetos de interface, é importante apresentar em que contexto consideramos os conceitos de interfaces e interação.

### **2.1 Interfaces e Interação**

Interfaces podem ser entendidas como os meios pelos quais possibilitamos a interligação ou comunicação entre sistemas ou entre sistemas e usuários. A interação pode ser entendida como uma forma de diálogo entre o computador e o usuário. A escolha do tipo de interface pode ter um efeito profundo na natureza desse diálogo (DIX et al., 2004). Sistemas interativos são aqueles onde os *softwares*

possuem interfaces com as quais os usuários possam interagir (PREECE et al., 1994).

### 2.1.1 Interfaces

No contexto deste trabalho, será considerada como interface toda a porção de um sistema com a qual um usuário mantém contato ao utilizá-lo, tanto ativa quanto passivamente. A interface engloba tanto *software* quanto *hardware* (dispositivos de entrada e saída, tais como teclados, *mouse*, *tablets*, monitores, impressoras entre vários outros).

Considerando a interação como um processo de comunicação, a interface pode ser vista como o sistema de comunicação utilizado neste processo. Segundo Moran (1981), a interface de usuário deve ser entendida como sendo a parte de um sistema computacional com a qual uma pessoa entra em contato física, perceptiva ou conceitualmente. A dimensão física inclui os elementos de interface que o usuário pode manipular. A dimensão perceptiva engloba aqueles que o usuário pode perceber. A dimensão conceitual resulta de processos de interpretação e raciocínio do usuário, desencadeados pela sua interação com o sistema, com base em suas características físicas e cognitivas, seus objetivos e seu ambiente de trabalho.

Prates e Barbosa (2003) mencionam que *“interfaces com baixa qualidade de uso trazem diversos problemas, dentre os quais: requerem treinamento excessivo, desmotivam a exploração, confundem os usuários, induzem os usuários ao erro, geram insatisfação, diminuem a produtividade e não trazem o retorno de investimento previsto”*.

Conforme Filho (2005), *“o desenvolvimento de interfaces de qualidade pode consumir algo perto de 50% do tempo e dos recursos alocados para o projeto”*. O grau de dificuldade para se implementar bons projetos de sistemas pode aumentar ou diminuir conforme a quantidade de formatos e elementos de interação envolvidos no projeto.

### 2.1.2 Interação

Neste trabalho, será considerada como interação o processo de comunicação entre pessoas e sistemas interativos (PREECE et al., 1994). Neste processo, usuário e sistema trocam turnos em que um “fala” e o outro “ouve”, interpreta e realiza uma ação. Esta ação pode ser tão simples quanto dar uma resposta imediata à fala do outro, ou consistir de operações complexas que alteram o estado do mundo.

A interação pode ser classificada em quatro tipos de tarefas executadas pelo usuário durante o uso de um sistema (PREECE, ROGERS e SHARP, 2005): instrução, conversação, manipulação e navegação e a exploração e pesquisa.

Shneiderman e Plaisant (2010) propõem uma classificação nos seguintes estilos de interação, que inclui entre outros a manipulação direta, seleção em menu, preenchimento de formulário, linguagem de comando e linguagem natural.

De acordo com Rebelo (2009), de forma geral podem ser observadas quatro atividades básicas que definem o processo do projeto de interação (PREECE, ROGERS e SHARP, 2005).

A primeira atividade básica é a de identificar necessidades e estabelecer requisitos. Trata-se da base dos requisitos do produto. Esta atividade sustenta o *design* e o desenvolvimento. O objetivo desta etapa é conhecer o usuário alvo e o tipo de suporte útil que o produto deve oferecer. Para isso é fundamental iniciar uma abordagem centrada no usuário.

A segunda atividade é a de desenvolver *designs* alternativos que preencham esses requisitos. Esta é a atividade central do projeto de interação, quando surgem as idéias que devem atender aos requisitos, as quais devem ser geradas com base em algum tipo de suporte. É composta por duas sub-atividades da geração de idéias: *design* (ou projeto) conceitual e *design* (ou projeto) físico. O *design* conceitual ou modelo conceitual do produto ganha forma juntamente com a descrição sobre o que o produto fará, como se comportará e parecerá. O *design* físico pode, então, ser iniciado com detalhes de interação e de interfaces, o que pode incluir o estudo de cores, sons, imagens, menus, animações, ícones, etc.

A terceira atividade trata de construir versões interativas dos *designs*, de maneira que possam ser analisadas. Busca-se com esta atividade fornecer meios de simular o processo de interação. As versões prototipadas são os meios mais conhecidos para mostrar ao usuário como um produto está sendo modelado e verificar a primeira reação de aceite. Entretanto, isso não significa que tal versão seja funcional. Protótipos em papel podem ser desenvolvidos e aplicados com rapidez, são baratos e eficazes na busca de problemas nas primeiras fases do projeto. O usuário tem uma noção real de como será a interação e do grau de atendimento às suas necessidades.

A quarta atividade trata de avaliar o que está sendo construído durante o processo, a fim de determinar a usabilidade e aceitabilidade do produto e utilizando vários critérios, tais como número de erros cometidos pelo usuário, atratividade, preenchimento dos requisitos, entre outros. Esta atividade não substitui os testes de qualidade que certificam o produto final, mas ajudam a verificar se todo ou parte do produto encontra-se em condição de uso.

Além das quatro atividades básicas, existem três características-chave quanto ao processo *design* de interação (PREECE, ROGERS e SHARP, 2005): foco no usuário, critérios de usabilidade específicos e iteração.

O foco no usuário faz com que estes estejam envolvidos no desenvolvimento do projeto. Utiliza-se a abordagem de desenvolvimento centrada no usuário, o que significa que as preocupações deste direcionam o desenvolvimento mais do que as preocupações técnicas.

Os critérios de usabilidade específicos são buscados pelo estabelecimento de metas claras, que devem ser identificadas, documentadas e acordadas no início do projeto. Isto auxilia na decisão pelas diferentes alternativas de *design*.

A terceira característica, iteração, é inevitável em todas as quatro atividades, pois os *designers* raramente conseguem encontrar a solução final na primeira vez (PREECE, ROGERS e SHARP, 2005). A iteração permite refinar o *design* por meio do *feedback*. Há, desta forma, uma transmissão de informações entre as atividades no ciclo de vida e a repetição dessas dentro do ciclo.

Como veremos a seguir, a área de IHC estuda o processo de interação, principalmente do ponto de vista do usuário: as ações que ele realiza usando a interface de um sistema, e suas interpretações das respostas transmitidas pelo sistema por meio da interface (PRATES e BARBOSA, 2003).

## **2.2 Interação Humano-Computador (IHC)**

Nos anos 80 foi cunhado o termo Interação Humano-Computador (IHC) como uma nova área de estudo cujo foco era não apenas o projeto de interface, mas todos os aspectos relacionados com a interação entre usuários e sistemas (PREECE et al., 1994).

Conforme Preece et al. (1994), IHC é uma área multidisciplinar que envolve investigação de modelos cognitivos ligados à transmissão de informação e aprendizagem, interação entre indivíduos e interfaces e análise das interfaces sob uma perspectiva ampla, considerando áreas como: ciência da computação, filosofia, psicologia cognitiva, psicologia social e organizacional, ergonomia ou fatores humanos, linguística, inteligência artificial, sociologia, antropologia, engenharia, *design*, entre outras.

O objetivo primordial da IHC, além de elaborar análises e prever fenômenos na relação entre usuário e sistema, é oferecer a projetistas (*designers*) de sistemas e pesquisadores os resultados práticos para o projeto da interface de usuário, pois com o estudo teórico dos fenômenos decorrentes dessa interação seria possível saber de antemão se o sistema a ser criado teria condições de satisfazer todas as necessidades do usuário em termos de utilização, aplicação e comunicação (OLIVEIRA NETTO, 2004).

Rebelo (2009) cita que o objetivo de um projeto de interação é tornar simples a vida de quem usa algo. Para isso, é preciso tratar os processos para comunicação e interação humana com os artefatos ou produtos interativos, o que inclui decisões sobre detalhes dos procedimentos da comunicação de ordem lógica ou física e comportamental. O projeto de interação é uma atividade prática e criativa que tem por objetivo final o desenvolvimento de um produto que ajude os usuários a atingirem suas metas.

A IHC tem sido caracterizada basicamente por abordagens de caráter cognitivo (PREECE et al., 1994), e estas abordagens têm pontos em comum com as áreas da psicologia cognitiva, da ciência cognitiva e da inteligência artificial, encarregada de estudarem o processo cognitivo, ou seja, o processo que torna possível a aquisição de conhecimento. Elas aplicam suas teorias na busca de compreensão das possibilidades e impossibilidades da mente dos usuários, e os resultados obtidos são quantitativamente muito mais numerosos do que os aferidos por qualquer outro tipo de abordagem (OLIVEIRA NETTO, 2004).

Os principais eixos de estudo da IHC são a: comunicabilidade, acessibilidade e usabilidade. A partir de Prates e Barbosa (2003), podemos entender que:

A comunicabilidade busca avaliar o processo implícito de comunicação *designer*–usuário, que se dá por meio da interface. Se refere à capacidade de os usuários entenderem o *design* tal como concebido pelos projetistas. A hipótese subjacente ao conceito de comunicabilidade é que, se um usuário entende as decisões que o projetista tomou ao construir a interface, aumentam suas chances de fazer um bom uso daquele sistema. Uma interface com boa comunicabilidade permite que o usuário formule um modelo mental compatível com o do projetista.

A acessibilidade está relacionada à possibilidade de acesso a sistemas por indivíduos portadores de alguma deficiência física, necessidade especial, idosos, bem como à inclusão digital e social.

A usabilidade, foco deste trabalho, é o conceito de qualidade de uso mais amplamente utilizado e está relacionado à facilidade e eficiência de aprendizado e de uso, bem como satisfação do usuário.

Além disso, questões de ética, inclusão social e digital, conforme mencionado por Baranauskas e Melo (2006), devem ser contempladas na acessibilidade e usabilidade da solução final e, portanto, podem ser incorporadas às práticas do guia a ser criado.

A seguir são apresentados os aspectos de usabilidade considerados neste trabalho.

## 2.3 Usabilidade

Segundo Nielsen (1993) a *“usabilidade é um atributo de qualidade que avalia quão fácil uma interface é de usar”*, ou *“a medida de qualidade da experiência de um usuário ao interagir com um produto ou um sistema”*. A usabilidade está associada à utilização de métodos durante o processo de criação do produto que contribuam com a facilidade de uso.

Sommerville (2007) aponta a usabilidade como um atributo essencial de um bom *software* e destaca: *“O software deve ser usável, sem esforço excessivo, pelo tipo de usuário para o qual ele foi projetado. Isso significa que ele deve apresentar uma interface com o usuário e documentação adequadas”*.

A NBR ISO/IEC 9126 (2003) define a usabilidade como a capacidade do produto de *software* de ser compreendido, aprendido, operado e atraente ao usuário, quando usado sob condições especificadas.

Segundo a NBR 9241-11 (2002), usabilidade é a medida pela qual um produto pode ser usado por usuários específicos para alcançar objetivos específicos com efetividade, eficiência e satisfação em um contexto de uso específico.

O conceito de usabilidade permite avaliar a qualidade de um sistema com relação a fatores que os projetistas definem como sendo prioritários ao sistema. Alguns fatores típicos envolvidos no conceito de usabilidade, conforme Nielsen (1993) e Preece, Rogers e Sharp (2005), são: facilidade de aprendizado, facilidade de uso, eficiência de uso e produtividade, satisfação do usuário, flexibilidade, utilidade e segurança no uso. Prates e Barbosa (2003) definem cada um desses fatores da forma abaixo.

A facilidade de aprendizado se refere ao tempo e esforço necessários para que os usuários aprendam a utilizar uma determinada porção do sistema com determinado nível de competência e desempenho. A facilidade de uso está relacionada não apenas com o esforço cognitivo para interagir com o sistema, mas também com o número de erros cometidos durante esta interação. A eficiência de uso serve para analisar se o sistema faz bem aquilo a que se destina. A produtividade serve para avaliar se o usuário consegue fazer o que precisa de forma

rápida e eficaz, geralmente considerando o tempo decorrido do início à conclusão de uma tarefa e o número de passos que o usuário precisou realizar. A satisfação do usuário enfatiza a avaliação subjetiva do sistema feita por seus usuários, incluindo emoções que possam surgir durante a interação, sejam elas positivas, como prazer e diversão, ou negativas, como tédio ou frustração. A flexibilidade considera o quanto um sistema é capaz de acomodar estas idiossincrasias. A utilidade se refere ao quanto um sistema oferece o conjunto de funcionalidades necessárias para os usuários realizarem suas tarefas. A segurança no uso se refere ao grau de proteção de um sistema contra condições desfavoráveis ou até mesmo perigosas para os usuários.

Considerando tais fatores, a usabilidade de uma aplicação pode ser medida e, para isso, dividida nas seguintes metas (PREECE, ROGERS e SHARP, 2005): eficácia, eficiência, segurança, utilidade, facilidade de aprendizado (*learnability*) e facilidade de se lembrar como se usa (*memorability*).

Uma outra abordagem relacionada à usabilidade é o *Design* de Interação, cujo foco é desenvolver sistemas que provoquem respostas positivas por parte dos usuários, como sentir-se à vontade, confortável, seguro e apreciar a experiência de usar tais sistemas. Pesquisas sugerem que a estética de uma interface pode ter um efeito positivo na percepção que as pessoas têm da usabilidade do sistema: quando sua aparência é agradável, os usuários provavelmente são mais tolerantes com a usabilidade (PREECE, ROGERS e SHARP, 2005).

Neste sentido, a usabilidade é incrementada por meio dos princípios de *design*, abstrações generalizáveis que podem orientar os *designers* a refletir sobre diferentes aspectos do *design* proposto (PREECE, ROGERS e SHARP, 2005). Os princípios mais comuns neste caso são: visibilidade, *feedback*, restrições, mapeamento, consistência e *affordance*.

A visibilidade está relacionada ao quanto a interface deixa visíveis aos usuários as suas funções, facilitando o uso. O *feedback* está relacionado ao retorno de informações ao usuário sobre o resultado das ações que ele está realizando. As restrições estão voltadas à limitação das ações que o usuário pode fazer, impedindo-o de selecionar opções incorretas em determinados momentos. Podem



ser físicas (a própria forma restringe o movimento), lógicas (desabilitando funções, por exemplo) ou culturais (convenções aceitas, como cores de alertas e ícones). O mapeamento refere-se à relação entre os controles e seus efeitos, como, por exemplo, a convenção universalmente aceita sobre as setas utilizadas para movimento de cursor para cima ou para baixo ou a sequência padrão das setas para se executar e controlar músicas (*rewind, play, forward*). A consistência está relacionada ao uso de elementos semelhantes em toda a interface para a realização de tarefas similares. O *affordance* é relacionado ao atributo de um objeto que permite aos usuários saber como utilizá-lo, ou seja, o quanto o objeto “dá uma pista” (NORMAN, 1988), ao usuário de sua utilidade. Por exemplo, o formato e posicionamento do botão de um mouse nos “convida” a pressioná-lo, bem como uma maçaneta a puxá-la. Um exemplo para interfaces seria o uso de botões em 3D (a fim de criar uma ilusão ainda maior de que tal controle da interface deve ser “pressionado”).

Em relação à importância de se adotar critérios de qualidade de uso no processo de desenvolvimento de sistemas interativos, Prates e Barbosa (2003) consideram que do ponto de vista do usuário, o sistema é a interface, e portanto, sua qualidade pode afetar, entre outros aspectos, a produtividade, a qualidade do produto final e o custo de suporte para atendimento ao usuário.

Uma vez apresentados os conceitos e perspectivas pelas quais a usabilidade será tratada neste trabalho, pode-se discutir a seguir as normas, padrões, modelos e melhores práticas em que se baseia o guia proposto.

### 3. Normas, Modelos e Ciclos de Vida

Há um número considerável de referências sobre o tema usabilidade. O foco do presente trabalho foi buscar primeiramente referências com maior profundidade na estruturação de processos e de ciclos de vida de usabilidade e em seguida incorporar em uma estrutura proposta de ciclo de vida as práticas mais amplamente difundidas nas bibliografias pesquisadas.

Para facilitar a organização deste trabalho, as referências pesquisadas foram agrupadas em três grupos: a) Normas e Padrões, b) Modelos e c) Melhores práticas (compêndios).

Este capítulo apresenta o levantamento dos dois primeiros grupos acima (normas, padrões e modelos) e uma análise sobre o ciclo de vida que será adotado para apresentar, no capítulo seguinte, o levantamento do terceiro grupo de referências: os compêndios de melhores práticas. Essa estruturação de seções e capítulos é detalhada na próxima seção.

#### 3.1 Principais Referências Utilizadas

O grupo de normas e padrões apresenta a NBR ISO/IEC 9126-1 (2003), NBR 9241-11 (2002), NBR ISO/IEC 12207 (2009), ISO 13407 (1999), ISO/IEC 15504-5 (2006) e ISO/TR 18529 (2000), que possuem relação direta ou indireta com o tema desta pesquisa. Esse grupo de referências será apresentado na próxima seção deste capítulo.

O grupo de modelos apresenta observações quanto ao PMBOK, CMMI e UMM, analisados sob a perspectiva da usabilidade e buscando-se extrair práticas relacionadas ao tema. Esse grupo de referências será apresentado ainda nesse capítulo, em uma seção posterior à seção de normas e padrões.

Alguns detalhes de algumas normas, padrões e modelos são explorados de forma mais específica no capítulo quatro, junto ao terceiro grupo de referências pesquisados (compêndios de melhores práticas). Antes de apresentar este terceiro grupo de referências, para finalizar esse capítulo é apresentada uma análise quanto aos ciclos de vida de *software* existentes e na adoção de um desses ciclos como

estrutura principal para o guia a ser elaborado pelo presente trabalho (“espinha dorsal” onde podem ser “encaixadas” as práticas pesquisadas).

Após essas seções, inicia-se então o capítulo quatro, onde será apresentado o terceiro e último grupo de referências: as melhores práticas extraídas de compêndios, organizadas e inseridas nos respectivos pontos do ciclo de vida de desenvolvimento de sistemas adotado para este trabalho. Esse grupo de referências finaliza a revisão bibliográfica, apresentando obras de autores de destaque como Jakob Nielsen, Ben Shneiderman, Catherine Plaisant, Deborah Mayhew, Hoa Loranger, Jennifer Preece, Yvonne Rogers e Helen Sharp, bem como pesquisadores que desenvolveram produções relevantes fortemente embasados nesses autores e em outros mencionados, por meio de livros, teses, dissertações e artigos científicos.

### **3.2 Normas e Padrões**

Há algumas normas com referência à usabilidade, qualidade e engenharia de *software* que podem ser utilizadas como balizadores para esse trabalho.

A NBR ISO/IEC 9126-1 (2003) trata do modelo de qualidade de *software* e foca processos, incluindo a usabilidade como uma das seis características principais dos atributos de qualidade. A NBR 9241 (2002) trata dos requisitos ergonômicos para trabalho de escritórios com computadores, reservando um capítulo para usabilidade e outros subsequentes que lidam com aspectos de interação e diálogo em sistemas. A NBR ISO/IEC 12207 (2009) traz contribuições em relação aos processos de ciclo de vida e desenvolvimento de *software*. A ISO 13407 (1999) trata o projeto centrado no usuário e também se aplica à usabilidade, atuando na questão dos ciclos de “análise/concepção/testes” com retroalimentação dos resultados entre um ciclo a outro, com objetivo de reduzir o risco de falhas conceituais do projeto e garantir que, a cada ciclo, o sistema responda cada vez melhor às expectativas e necessidades dos usuários (CYBIS, BETIOL e FAUST, 2007). A ISO 15504-5 (2006) trata de avaliações de processo para tecnologia da informação e representa um conjunto de melhores práticas para a engenharia de *software*. A ISO/TR 18529 (2000) trata da ergonomia da interação humano-sistema, abordando descrições de processos de ciclo de vida centrado no humano.

### 3.2.1 NBR ISO/IEC 9126-1

Embora esta norma esteja sendo substituída pela série NBR ISO/IEC 25000 (2008), diversas referências bibliográficas (inclusive outras normas) ainda fazem uso ou menção à NBR ISO/IEC 9126-1 (2003).

A NBR ISO/IEC 9126-1 - Engenharia de *software* - Qualidade de produto (2003) descreve um modelo de qualidade do produto de *software*, composto de duas partes: a) qualidade interna e qualidade externa e b) qualidade em uso.

A usabilidade tem destaque na norma nas características de qualidade externa e interna, onde os atributos de qualidade de *software* são categorizados em seis características: funcionalidade, confiabilidade, usabilidade, eficiência, manutenibilidade e portabilidade, conforme ilustrado na Figura 1.



**Figura 1** - Modelo de qualidade para qualidade externa e interna

**Fonte:** Adaptado de NBR ISO/IEC 9126 (2003)

De acordo com a norma, a usabilidade deve compreender as capacidades de inteligibilidade (possibilitar ao usuário compreender se o *software* é apropriado e como ele pode ser usado para tarefas e condições de uso específicas), apreensibilidade (possibilitar ao usuário aprender sua aplicação), operacionalidade (possibilitar ao usuário operá-lo e controlá-lo) e atratividade (ser atraente ao usuário). A conformidade é atingida pela aderência às normas, convenções, guias de estilo ou regulamentações relacionadas à usabilidade.

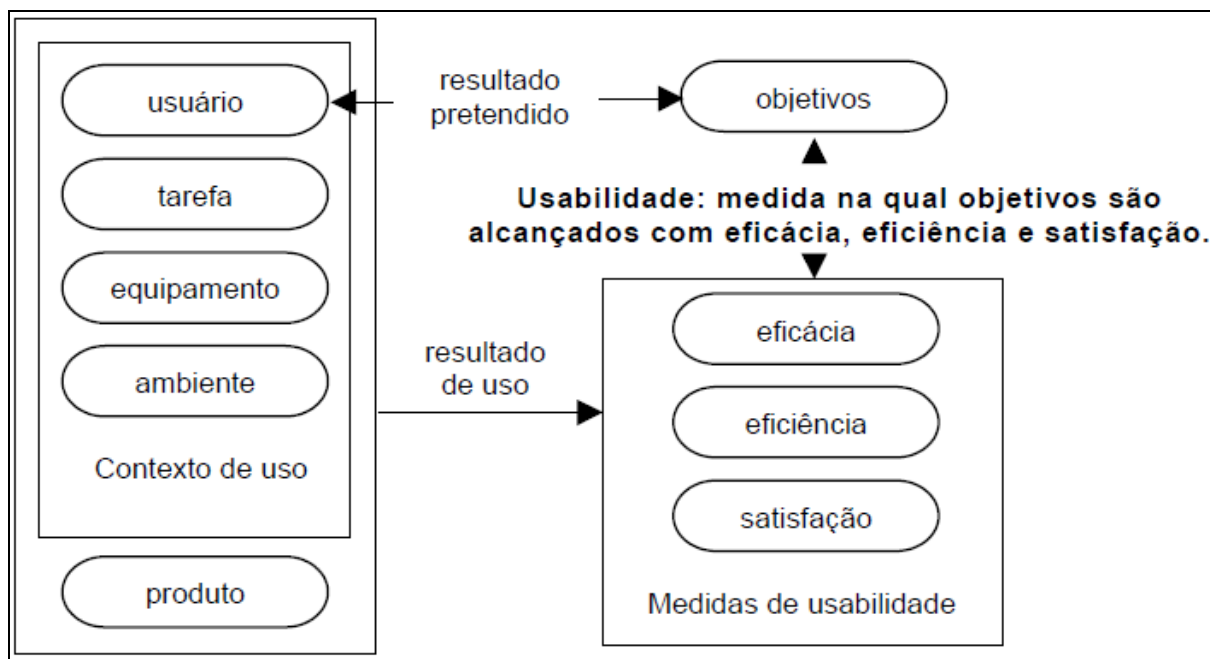
Em relação ao caráter de elaboração progressiva dos requisitos com base nas necessidades dos usuários, uma consideração mencionada nesta norma merece destaque:

*“As necessidades explicitadas pelo usuário nem sempre refletem suas reais necessidades porque: (1) frequentemente, o usuário não está consciente de suas necessidades reais; (2) as necessidades podem mudar após terem sido explicitadas; (3) usuários diferentes podem ter ambientes operacionais diferentes e (4) pode ser impossível consultar todos os tipos de usuários, particularmente para produtos de software de prateleira. Então, requisitos de qualidade não podem ser completamente definidos antes do início do projeto. Além disto, é necessário entender as necessidades reais do usuário tão detalhadamente quanto possível e representá-las nos requisitos.”*

Pode-se observar a importância de que o processo de desenvolvimento da interface contemple tarefas que busquem não apenas o envolvimento dos usuários e a identificação dos requisitos desde o início do projeto mas, principalmente, um processo de elaboração progressiva e revisão destes requisitos pela equipe do projeto, usuários e demais envolvidos. A usabilidade deve estar inserida nesse processo, por meio de requisitos e metas voltadas à usabilidade que devem ser identificados e acompanhados durante o ciclo de vida do projeto de interface. Isso corrobora a abordagem de engenharia de usabilidade, que será apresentada no presente trabalho, pela ênfase à inserção das atividades de usabilidade, de uma forma iterativa, ao longo do ciclo de vida de criação da interface.

### **3.2.2 NBR 9241-11**

A NBR 9241 - Requisitos ergonômicos para trabalho de escritório com computadores (2002) define a usabilidade como a medida na qual um produto pode ser usado para alcançar objetivos com eficácia, eficiência e satisfação em um contexto específico de uso, conforme ilustrado na Figura 2.



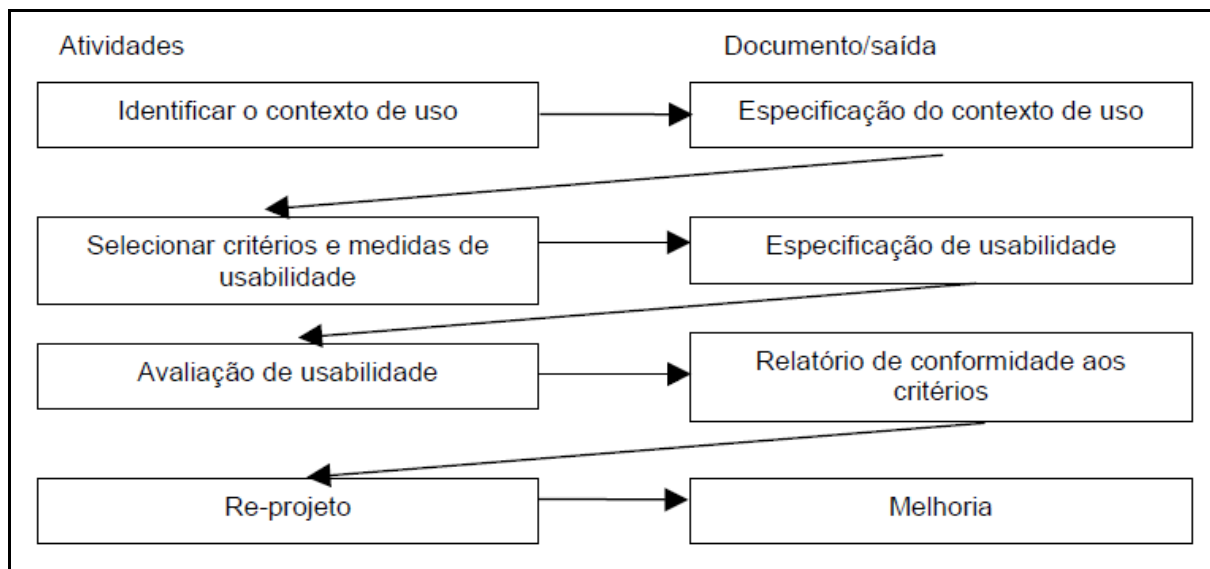
**Figura 2** - Estrutura de Usabilidade  
**Fonte:** NBR 9241-11 (2002)

A eficácia é medida pela acurácia e completude com as quais usuários alcançam objetivos específicos. A eficiência é medida pelos recursos gastos em relação à acurácia e a abrangência com as quais os usuários atingem os objetivos. A satisfação é medida pela ausência do desconforto e pela presença de atitudes positivas para com o uso de um produto. O contexto específico de uso envolve os usuários, tarefas, *hardware*, *software*, materiais e os ambientes físico e social no qual o produto é usado.

Quanto à estrutura desta norma, a parte onze é denominada NBR 9241-11 (2002) e trata de orientações específicas sobre usabilidade. Porém, as demais partes também oferecem diretrizes aplicáveis à usabilidade, já que tratam de requisitos (de tarefa, apresentação visual, cores e layout) e orientações sobre apresentação da informação e diálogos (por menu, linguagem de comando, manipulação direta e por preenchimento de formulário), aplicáveis às interfaces.

A norma orienta que exista uma descrição explícita do contexto de uso do produto e das medidas relevantes de usabilidade, na forma de princípios e técnicas gerais, em vez de requisitos para usar métodos específicos. Conforme Cybis, Betiol e Faust (2007) as medidas de desempenho e de satisfação dos usuários avaliam a qualidade do sistema de trabalho com todas as suas interligações e qualquer mudança como treinamento adicional ou melhoria de iluminação forcem uma

reavaliação da usabilidade do sistema. A norma menciona as atividades de usabilidade como parte de um planejamento da qualidade. A Figura 3 mostra o relacionamento entre as atividades envolvidas e as respectivas saídas geradas.



**Figura 3** - Atividades e respectivas saídas  
**Fonte:** Adaptado de NBR 9241-11 (2002)

É possível notar uma semelhança entre a NBR 9241-11 (2002) e a abordagem de ciclo de vida utilizada neste trabalho, na qual é dada ênfase em atividades iniciais de entendimento do usuário, tarefas e ambiente de trabalho. O projetista de *software* precisa identificar antecipadamente os usuários, tarefas e ambientes antes de projetar atributos apropriados de usabilidade, segundo as orientações e requisitos da parte dez desta norma e também das partes doze até dezessete. Além disso, a norma traz como referência de publicações relevantes algumas das referências utilizadas também neste presente trabalho, como Shneiderman e Plaisant (2010) e Nielsen (1993).

Para facilitar o entendimento da aplicação prática desta norma, os seus detalhes estão distribuídos ao longo deste trabalho, nos pontos específicos em que ela pode ser utilizada como referência.

### 3.2.3 NBR ISO/IEC 12207

A norma NBR ISO/IEC 12207 – Engenharia de sistemas e software - Processos de ciclo de vida de software (2009) estabelece uma estrutura comum para os processos de ciclo de vida de *software*, com terminologia bem definida, que

pode ser referenciada pela indústria de *software*. Esta versão da norma faz menção clara à usabilidade e menciona que quando a usabilidade é importante ao projeto, os requisitos de usabilidade devem ser planejados, especificados e implementados por meio de processos. A norma referencia para isso processos de outras normas como a NBR 9241-11 (2002), ISO 13407 (1999), NBR ISO/IEC 9126-1 (2003), NBR ISO/IEC 25000 (2008) e ISO/TR 18529 (2000).

Nesta norma, os processos de ciclo de vida são agrupados em dois tipos: processos contextuais de sistema e processos específicos de *software*. Os contextuais abrangem quatro grupos de processos (contratuais, organizacionais, de projeto e técnicos), enquanto os específicos de *software* contemplam três grupos de processos (de implementação, de apoio e de reuso de *software*).

O “Anexo E” da NBR ISO/IEC 12207 (2009) fornece uma visão específica de processo para usabilidade, demonstrando claramente os itens desta norma (processos, atividades e tarefas) que devem ser cobertos para uma implementação de processo com foco no atendimento à usabilidade. Dentre esses processos, podemos citar o de gestão de portfólio, gestão de infraestrutura, planejamento de projeto, controle e avaliação de projeto, definição de requisitos de *stakeholders*, análise de requisitos de sistema, arquitetura do sistema, integração de sistema, gestão da informação, medição, operação de *software* e manutenção de *software*.

A usabilidade é tratada de forma mais explícita nesta norma em alguns processos como o de definição de requisitos de *stakeholders* (item 6.4.1 da norma) e o de análise de requisitos de *software* (item 7.1.2 da norma), porém, podemos observar práticas que devem envolver a atenção à usabilidade em vários outros processos. A norma faz menção à NBR ISO/IEC 9126-1 (2003) e à NBR ISO/IEC 25000 (2008) como guias para auxiliar a especificação de características de qualidade e cita algumas destas características que estão relacionadas à usabilidade da interface como as especificações de testabilidade (definida como a “*extensão em que um teste objetivo e factível pode ser projetado para determinar se um requisito é atendido*”) e especificações de interação entre homem-máquina, dispostos por vários itens da norma.



### 3.2.4 ISO 13407

A ISO 13407 - Processos de *Design* Centrados no Humano (1999) fornece orientação para obtenção de qualidade de uso pela incorporação de atividades de *design* centradas no usuário em todo o ciclo de vida de sistemas iterativos. Ela descreve o *design* centrado no usuário como uma atividade multidisciplinar, que incorpora conhecimento e técnicas sobre fatores humanos e ergonomia com o objetivo de aumentar a eficácia e produtividade e melhorar as condições humanas de trabalho. O processo consiste de cinco etapas, conforme ilustrado na Figura 4.

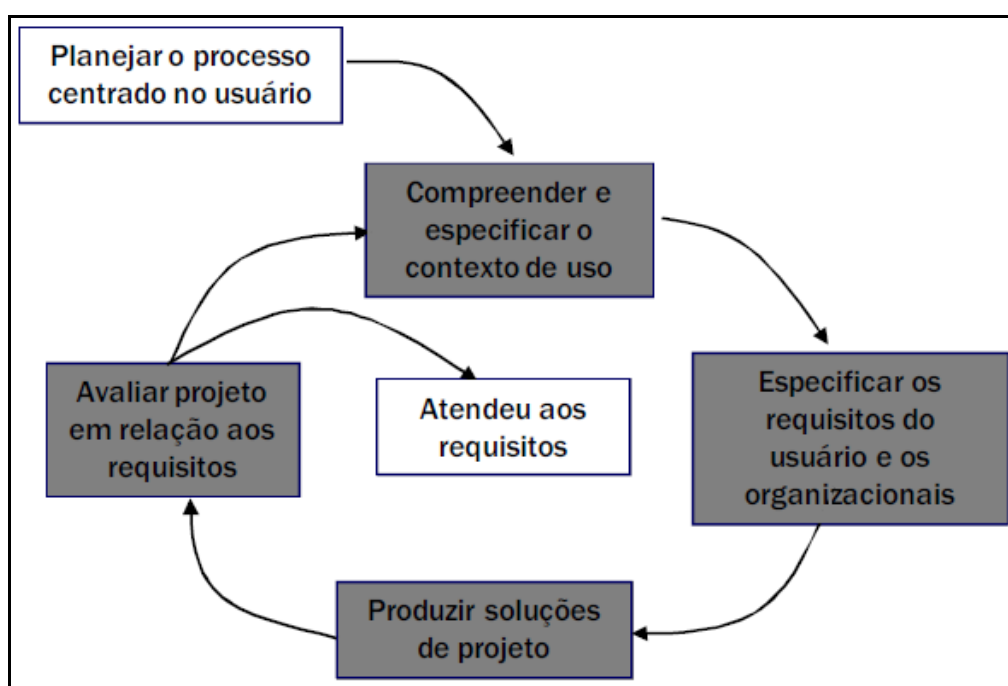


Figura 4 - Processo de *design* centrado no usuário

A primeira etapa, planejar o processo centrado no humano, é referente à elaboração de um plano de ação e dos parâmetros de avaliação de eficiência das metas estabelecidas para a qualidade e reforça a importância do envolvimento dos usuários desde as primeiras atividades da concepção do produto. A segunda etapa, especificar o contexto de uso, trata da análise sistemática do perfil do usuário, das tarefas e seu contexto, dos objetivos do uso do produto para cada grupo de usuários e da decisão quanto às métricas de avaliação do uso em contexto. A terceira etapa, especificar requisitos organizacionais e de usuário, diz respeito ao levantamento e análise dos requisitos, considerando, entre outros, requisitos não-funcionais de usabilidade que possam ser usados para medir a satisfação do usuário. A quarta etapa, produzir soluções de *design*, se refere ao uso de normas e guias de estilo,

além de métodos de modelagem para especificação do produto, prototipação, avaliação por usuários e melhoria da especificação após a avaliação. A quinta etapa, avaliar o *design* em relação aos requisitos de usuário, trata dos testes do produto e validação do atendimento aos requisitos.

Percebe-se nesta norma uma abordagem de *design* formativo centrado no usuário (BENINI, 2006), com ciclos de iteração para um detalhamento progressivo dos requisitos de usuário, o que corrobora com as abordagens mencionadas posteriormente neste trabalho de engenharia de usabilidade.

### 3.2.5 ISO/IEC 15504-5

A ISO/IEC 15504 – Avaliação de Processos de Tecnologia da Informação também é conhecida como SPICE (*Software Process Improvement and Capability dEtermination*). Na prática existem duas “15504”: uma é a norma ISO/IEC 15504, também conhecida como Framework 15504 ou SPICE e a outra é o modelo exemplo de avaliação de processo, que consta na parte cinco da norma, denominado de ISO/IEC 15504-5 (2006). A norma é baseada na NBR ISO/IEC 12207:1998 e sua estrutura e conjunto de práticas são muito aderentes ao modelo CMMI, embora possua nomenclatura diferente tanto nas práticas como nos níveis de maturidade/capacidade propostos.

Segundo a ISO/IEC 15504-5 (2006), o nível de capacidade de cada processo é consequência de uma pontuação dos atributos de processo, utilizando a seguinte escala: “N” (atributo não atingido pelo processo), “P” (atingido parcialmente), “L” (atingido largamente) ou “F” (atingido completamente, em inglês, *fully*). Para estar em um nível de capacidade, um processo tem que ter pontuações “L” ou “F” nos atributos do nível e “F” em todos os atributos dos níveis anteriores (SALVIANO, 2006). A escala desta norma e a escala adotada para o CMMI (CHRISSIS, KONRAD e SHRUM, 2003) podem servir de base para a definição da escala a ser utilizada no guia proposto pelo presente trabalho.

O modelo para avaliação de processo da ISO/IEC 15504-5 (2006) define quarenta e oito processos, para as atividades de aquisição, fornecimento, desenvolvimento, manutenção, apoio, gerência e melhoria de *software*. Esses

processos estão organizados em nove grupos que, por sua vez, estão organizados em três categorias: fundamentais, organizacionais e de apoio, conforme a Figura 5.



**Figura 5 - Estrutura e elementos do modelo ISO/IEC 15504-5**

**Fonte: SALVIANO (2006)**

Duas menções explícitas à usabilidade são feitas nesta norma: no processo ENG.3 (projeto da arquitetura de sistema) e no processo ENG.5 (projeto de software). O propósito do processo ENG.3 é “*identificar quais requisitos do sistema devem ser alocados a quais elementos do sistema*” (SALVIANO, 2006) e nesse caso a menção é feita na prática básica ENG.3.BP5, que trata da avaliação de arquiteturas alternativas. De acordo com a norma, os critérios de avaliação para o projeto de arquitetura devem ser definidos e a racional pela escolha da alternativa deve ser documentada. A usabilidade é apontada como uma das características de qualidade que podem fazer parte desse critério, além de outras características como modularidade, manutenibilidade, extensibilidade, escalabilidade, confiabilidade e segurança. A prática ENG.3.BP6 também é importante para a usabilidade já que trata da consistência entre a análise de requisitos e o projeto de arquitetura proposto ao sistema. O propósito do processo ENG.5 é “*prover um projeto (design) para o*

*software que implemente e possa ser verificado em relação aos requisitos*” (SALVIANO, 2006) ) e nesse caso a menção é feita na prática básica ENG.5.BP1, que trata da “transformação” de requisitos de *software* em um projeto de arquitetura que descreva de forma condizente a estrutura e identifique os principais elementos de *software*. De acordo com a norma, nesse processo de “transformação” deve haver a avaliação de arquiteturas alternativas, o que remete à mesma questão da usabilidade apontada anteriormente para o processo ENG.3. Já a prática ENG.5.BP5 também é importante para a usabilidade pois visa garantir que haja consistência entre a análise de requisitos e o *design* do *software*.

Embora o termo usabilidade não seja tratado de forma explícita em outros processos da norma, as atividades que compõem o ciclo de vida da construção de um sistema, e que estão relacionadas a prover e manter uma boa usabilidade do sistema, aparecem em diversos outros processos da ISO/IEC 15504-5 (2006). Tais processos devem ser considerados na elaboração do guia proposto pela presente dissertação. Como exemplo, os processos MAN.3.PB4 e PB5 consideram a questão de dimensionamento de esforço e isso deve ser contemplado para as atividades voltadas à usabilidade. Para facilitar o entendimento da aplicação prática desta norma, os detalhes sobre esses processos estão distribuídos ao longo deste trabalho, nos pontos específicos em que ela pode ser utilizada como referência.

### **3.2.6 ISO/TR 18529**

A ISO/TR 18529 – Ergonomia da interação humano-sistema - descrições de processos de ciclo de vida centrado no humano (2000) surgiu como uma evolução da ISO 13407 (1999) e sua estrutura aparece no modelo UMM (EARTHY, 2000), que será tratado posteriormente neste trabalho. Esta norma define sete processos: garantir conteúdo de HCD na estratégia do sistema, planejar e gerenciar o processo de HCD, especificar os requisitos dos *stakeholders* e da organização, entender e especificar o contexto de uso, produzir soluções de *design*, avaliar *designs* em relação aos requisitos e, por fim, introduzir e operar o sistema.

O primeiro processo, HCD.1 - Garantir conteúdo de HCD na estratégia do sistema, tem como propósito estabelecer e manter um foco nas necessidades dos *stakeholders* e do usuário nas partes da organização que lidam com marketing,

conceitos, desenvolvimento e suporte de sistemas. Esse processo possui cinco práticas (HCD.1.1 a HCD.1.5): representar o usuário final, coletar a inteligência de mercados (*marketing*), definir e planejar uma estratégia de sistema, coletar o *feedback* do marketing e analisar as tendências em usuários.

O segundo processo, HCD.2 - Planejar e gerenciar o processo de HCD, tem como propósito especificar como as atividades centradas no humano se encaixam no processo de ciclo de vida do sistema e na empresa. Esse processo possui oito práticas (HCD.2.1 a HCD.2.8): consultar *stakeholders*, identificar e planejar o envolvimento do usuário, selecionar métodos e técnicas centrados no usuário, assegurar que o time de projeto siga uma abordagem centrada no humano, planejar as atividades centradas no humano, gerenciar as atividades centradas no humano, disseminar a abordagem centrada no humano pela empresa e prover suporte ao projeto centrado no humano.

O terceiro processo, HCD.3 – Especificar os requisitos dos *stakeholders* e da organização, tem como propósito estabelecer os requisitos da organização e outras partes interessadas no sistema. Esse processo leva em conta as necessidades, competências e ambiente de trabalho de cada *stakeholder* relevante do sistema. Esse processo possui seis práticas (HCD.3.1 a HCD.3.6): esclarecer e documentar as metas do sistema, definir os *stakeholders*, avaliar risco aos *stakeholders*, definir o sistema, gerar os requisitos dos *stakeholders* e da organização e definir os objetivos de qualidade no uso.

O quarto processo, HCD.4 - Entender e especificar o contexto de uso, tem como propósito identificar, esclarecer e armazenar as características dos *stakeholders*, suas tarefas e o ambiente físico e organizacional no qual o sistema irá operar. Esse processo possui cinco práticas (HCD.4.1 a HCD.4.5): identificar e documentar as tarefas do usuário, identificar e documentar atributos significativos do usuário, identificar e documentar o ambiente organizacional, identificar e documentar o ambiente técnico e identificar e documentar o ambiente físico.

O quinto processo, HCD.5 - Produzir soluções de *design*, tem como propósito criar potenciais soluções de projeto utilizando como fonte de suporte práticas já estabelecidas do estado da arte, experiências e conhecimento dos participantes e os

resultados do contexto da análise do uso. Esse processo possui oito práticas (HCD.5.1 a HCD.5.8): alocar funções, produzir um modelo de tarefas composto, explorar o *design* do sistema, usar conhecimento existente para desenvolver soluções de *design*, especificar o sistema, desenvolver protótipos, desenvolver o treinamento do usuário e desenvolver o suporte ao usuário.

O sexto processo, HCD.6 – Avaliar *designs* em relação aos requisitos, tem como propósito coletar o *feedback* do projeto em desenvolvimento. Esse *feedback* será coletado de usuários finais e outras fontes representativas. Esse processo possui seis práticas (HCD.6.1 a HCD.6.6): especificar e validar o contexto da avaliação, avaliar protótipos a fim de definir os requisitos para o sistema, avaliar protótipos a fim de melhorar o *design*, avaliar o sistema a fim de checar se os requisitos do sistema foram atendidos, avaliar o sistema a fim de checar se as práticas requeridas foram seguidas e avaliar o sistema em uso a fim de assegurar que ele continue a atender as necessidades do usuário e da organização.

O sétimo processo, HCD.7 - Introduzir e operar o sistema, tem como propósito estabelecer os aspectos de humano-sistema do suporte da implementação do sistema. Esse processo possui seis práticas (HCD.7.1 a HCD.7.6): gerenciar as mudanças, determinar o impacto na organização e nos *stakeholders*, tratar *design* local e customização, entregar o treinamento do usuário, fornecer suporte aos usuários nas atividades planejadas e garantir conformidade com a legislação sobre ergonomia do ambiente de trabalho.

Tais processos devem ser considerados na elaboração do guia proposto pela presente dissertação. Para facilitar o entendimento da aplicação prática desta norma, os detalhes sobre esses processos estão distribuídos ao longo deste trabalho, nos pontos específicos em que eles podem ser utilizados como referência.

### 3.3 Modelos

Estudos preliminares mostram que é dada pouca ênfase a questões como usabilidade e projetos de interface nos modelos de processos atuais. Podemos observar isso ao analisarmos, por exemplo, dois dos mais conceituados e utilizados guias de processos: o PMBOK e o CMMI. Há também um modelo voltado à

usabilidade: o UMM. Essas três referências serão exploradas a seguir, pela perspectiva da usabilidade.

### **3.3.1 PMBOK**

O PMBOK (Project Management Body of Knowledge) - Guia do Conhecimento em Gerenciamento de Projetos (PMBOK, 2008) foi desenvolvido pelo PMI - Project Management Institute nos EUA e é hoje um dos modelos de gestão de projetos mais reconhecidos mundialmente, com processos que reúnem as melhores práticas desse assunto e ainda tem uma aderência significativa no segmento de Tecnologia da Informação. Já o modelo de maturidade em gestão de projetos do PMI é o OPM3 (Organizational Project Management Maturity Model).

O PMBOK, assim como o OPM3, não apresenta maior foco ou profundidade em práticas de usabilidade ou de IHC, embora exista ênfase na elaboração progressiva e detalhada dos requisitos de escopo (processo chamado de decomposição) bem como a verificação da aderência desses requisitos às necessidades do usuário, cobertos pelas áreas de conhecimento de Gerenciamento de Escopo e Gerenciamento de Qualidade, por exemplo. Além disso, todas as atividades voltadas à usabilidade devem ser definidas, dimensionadas e seu esforço deve ser inserido nas estimativas do projeto, conforme as áreas de conhecimento de Gerenciamento de Tempo e de Escopo. Lições aprendidas dos projetos também devem ser registradas a fim de se manter atualizada a base de ativos de processos.

### **3.3.2 CMMI**

O CMMI - Capability Maturity Model Integration, desenvolvido pelo Software Engineering Institute (SEI) da Universidade de Carnegie Mellon nos EUA, é atualmente um dos modelos de maturidade para processos de desenvolvimento de *software* mais reconhecidos mundialmente. Seu objetivo é reunir as melhores práticas de desenvolvimento e integração de sistemas e propor um modelo para a avaliação de aderência dos processos de uma organização (CHRISSIS, KONRAD e SHRUM, 2003). O CMMI também não entra em profundidade na questão de usabilidade, tratada pela abordagem de IHC, embora seja possível encontrar nas diversas práticas os pontos específicos em que a busca pela usabilidade possa ser explorada. Para efeito de ilustração, podemos considerar como exemplo um

processo de desenvolvimento de *software* aderente às Áreas de Processo (PA's) dos níveis dois e três de maturidade do modelo CMMI. Nesse caso, as práticas destacadas a seguir poderiam estar presente no guia proposto por esse trabalho de pesquisa, a fim de cobrir cada uma das quatro atividades básicas propostas por Preece, Rogers e Sharp (2005) para o projeto de interação.

Na primeira atividade básica de identificação de necessidades e requisitos, uma das práticas seria que o processo possua uma atividade formal de elicitación de requisitos (a fim de cumprir a PA de Gestão de Requisitos), e que onde houver requisitos atendidos sob a forma de interface tal atividade seja conduzida sob a ótica da interação do usuário, ou seja, que nesses casos as especificidades de interação para atender as necessidades e cobrir os requisitos sejam levantadas nessa etapa do processo. Técnicas e artefatos de IHC como prototipações, cenários, modelos de tarefa, *storyboards*, dentre outros poderiam ser usados (BARBOSA, FURTADO e GOMES, 2006) para atendimento ao foco em usabilidade.

Na segunda atividade básica, relacionada a desenvolver projetos alternativos que vão ao encontro dos requisitos, uma das etapas do processo aderente ao CMMI deve contemplar a análise da solução técnica (PA de Solução Técnica), que visa buscar possíveis alternativas de solução (tanto à arquitetura como à implementação) para atendimento aos requisitos do usuário. Com foco em usabilidade, esta etapa poderia incorporar uma validação das alternativas de solução técnica em relação à qualidade das interações propostas em cada alternativa.

Na terceira atividade básica de construir versões interativas de maneira que possam ser comunicadas e analisadas, um processo aderente ao nível três do CMMI deve contemplar as PA's de Verificação, Validação e Integração de Produto. Os processos relacionados à essas PA's deveriam incorporar atividades que enfoquem a qualidade da interação sob a perspectiva de usabilidade. Nesse caso, técnicas típicas nesses processos como protótipos, testes e revisões por pares, por exemplo, devem refletir passos que busquem atendimento não apenas ao cumprimento de requisitos técnicos e funcionais como também aos aspectos referentes às necessidades de usabilidade.



Na quarta atividade básica de avaliar o que está sendo construído e medir sua aceitabilidade, os processos orientados a cumprir PA's como as de Validação, Verificação, Gestão de Requisitos e Garantia de Qualidade do Processo e Produto devem ser verificados a fim de que contemplem atividades com foco na usabilidade. Já os processos voltados à PA de Medição e Análise são de grande importância já que devem conter métricas e indicadores que permitam refletir a visibilidade de qualidade da usabilidade do processo, informando, por exemplo, indicadores relacionados ao número de erros e dificuldade no uso do protótipo, tentativas frustradas ou impossibilidade de acesso a funcionalidades do sistema, taxa de abandono, tempo e quantidade de passos para se alcançar determinada funcionalidade, dentre outros. Algumas organizações utilizam o modelo CMMI em conjunto com a norma NBR ISO/IEC 9001 – Sistemas de gestão da Qualidade (2008) que também contempla a medição e acompanhamento de objetivos (cobertos por itens como o 5.4, 5.6 e 8.2 da norma, voltados a objetivos de qualidade, análise crítica e monitoramento e medição).

Além disso, de modo geral, os processos orientados ao planejamento e controle do projeto (PA's de Planejamento do Projeto e Monitoramento e Controle do CMMI) devem contemplar o eventual esforço adicional necessário para tais atividades focadas em usabilidade, bem como as consequentes atividades de acompanhamento necessárias, de modo que o plano do projeto seja realista em relação às expectativas e aos custos decorrentes dessa abordagem.

### **3.3.3 UMM**

Há ainda um modelo voltado especificamente para a usabilidade, o UMM - Modelo de Maturidade de Usabilidade ou Usability Maturity Model (EARTHY, 2000). Esse é um modelo para processos de Desenvolvimento Centrado em Humanos (HCD - *Human-Centred Development*) e fornece um conjunto de processos que sejam aderentes à NBR ISO/IEC 12207 (2009), ISO/IEC 15504-5 (2006) e modelo CMMI (CHRISSIS, KONRAD e SHRUM, 2003). Tais processos devem ser considerados na elaboração do guia proposto pela presente dissertação. Porém, como os processos desse modelo são similares aos da ISO/TR 18529 (2000), para simplificar o presente trabalho estaremos utilizando como referência a ISO/TR 18529 (2000) e não o UMM (EARTHY, 2000).

Para facilitar o entendimento, os detalhes sobre esses processos estão distribuídos ao longo deste trabalho, nos pontos específicos em que eles podem ser utilizados como referência.

### 3.4 Ciclos de Vida

Nesta seção é definido o ciclo de vida que servirá como estrutura básica de organização do guia a ser criado. Primeiramente, são apresentados os modelos pesquisados de ciclo de vida baseados na engenharia de *software* e em IHC. Em seguida é detalhado o modelo de ciclo de vida definido como o mais aderente ao objetivo deste trabalho de pesquisa: o da engenharia de usabilidade. A partir desta definição, no capítulo a seguir são apresentadas, dentro do ciclo de vida da engenharia de usabilidade, as práticas obtidas das diversas referências e que foram consideradas para a elaboração do guia de avaliação de processos a ser construído.

O termo *modelo de ciclo de vida* é utilizado para representar um modelo que capta um conjunto de atividades e a maneira como elas se relacionam (PREECE, ROGERS e SHARP, 2005). Sommerville (2007) utiliza o termo *modelo de processo* para denominar o que chamamos de ciclo de vida. A ISO/IEC 15504-5 (2006), por meio da prática básica MAN.3.PB2, estabelece que os projetos devem ter definido qual o seu ciclo de vida. Modelos de ciclo de vida foram desenvolvidos em campos relacionados ao *design* de interação, como em engenharia de *software* e em IHC. A seguir serão descritos três modelos da engenharia de *software* e três da IHC que provaram ser bem-sucedidos e que mostram como a ênfase no desenvolvimento de *software* mudou gradualmente a fim de incluir uma visão mais iterativa, centrada no usuário. Alguns modelos de ciclo de vida para projetos de interface sob a perspectiva de engenharia de *software* foram propostos, como o modelo cascata (*waterfall*), o modelo espiral e o modelo RAD (*Rapid Applications Development*) - Desenvolvimento Rápido de Aplicações.

O modelo cascata foi proposto em 1970 e até então não havia uma abordagem com que todos concordassem no que diz respeito ao desenvolvimento de *software*. É um modelo linear em que cada passo deve ser completado antes que o próximo possa ser dado. Uma desvantagem é que a análise de requisitos é feita

no início e, mesmo com certo nível de iteração, os requisitos não são revisados e avaliados com o usuário ao longo do ciclo.

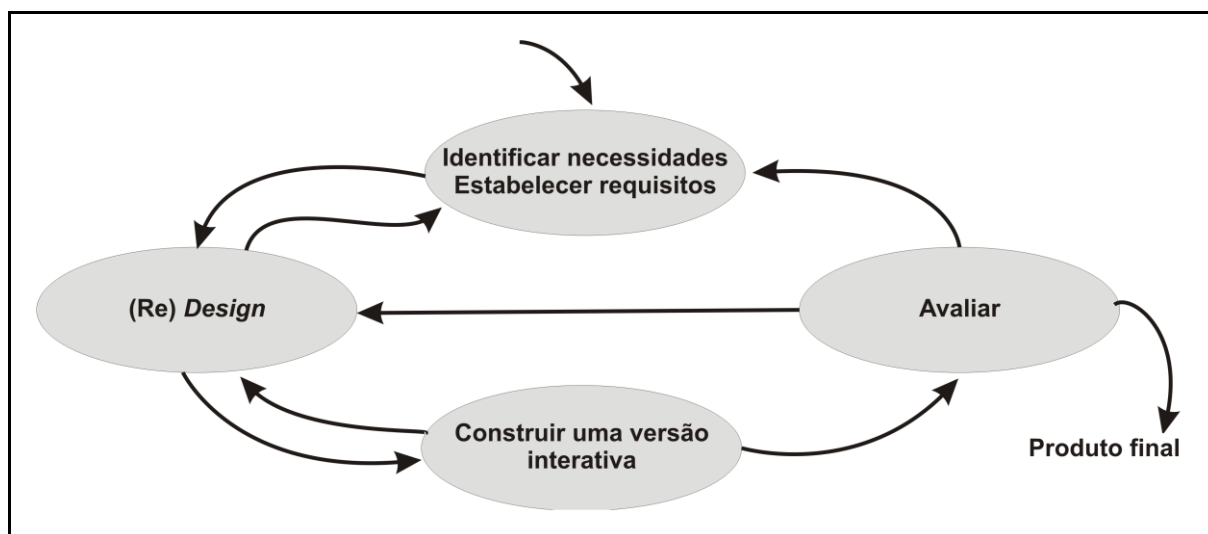
O modelo espiral foi sugerido por Barry Boehm (BOEHM, 1988) e propõe um *framework* iterativo que permite que as idéias e o progresso sejam repetidamente verificados e validados. Cada iteração pode ser baseada em um modelo de ciclo de vida diferente, com atividades diferentes. O que guia o desenvolvimento nesse modelo são o plano de desenvolvimento e as especificações focadas em riscos. Um segundo modelo proposto por Boehm em 1998 foi denominado de Espiral WinWin e incorpora identificação e negociação com *stakeholders*.

O modelo RAD é centrado no usuário e busca minimizar o risco causado por requisitos que se alterem durante o projeto. Possui duas características-chave: ciclos com tempo limitado de cerca de seis meses (*time-boxing*) com entregas (finais ou parciais do sistema) ao final desse período e oficinas (*workshops*) JAD (*Joint Application Development*) nos quais usuários e desenvolvedores se reúnem para discutir os requisitos.

Alguns modelos de ciclo de vida para projetos de interface analisados sob a perspectiva de IHC foram propostos, como o Modelo Simplificado (PREECE, ROGERS e SHARP, 2005), o Modelo Estrela (HIX e HARTSON, 1993) e o Modelo da Engenharia de Usabilidade (MAYHEW, 1999, 2008).

O modelo Simplificado (PREECE, ROGERS e SHARP, 2005) do *design* de interação possibilita um número ilimitado de repetição do ciclo, desde que a última atividade sempre seja um teste. Ele possui o foco nos usuários e determina e possibilita que um produto evolua da forma que for compreendida como melhor pela equipe. Conforme ilustrado na Figura 6, esse modelo propõe que a maioria dos projetos se inicia com a identificação de necessidades e estabelecimento de requisitos. Em seguida, alguns *designs* alternativos são gerados para atendimento a tais necessidades e requisitos. Versões interativas desses *designs* são desenvolvidas e avaliadas. Com base no *feedback* dessas avaliações, a equipe pode precisar retornar à identificação de necessidades ou refinamento de requisitos. A sequência das atividades nos ciclos iterativos depende da dinâmica estabelecida

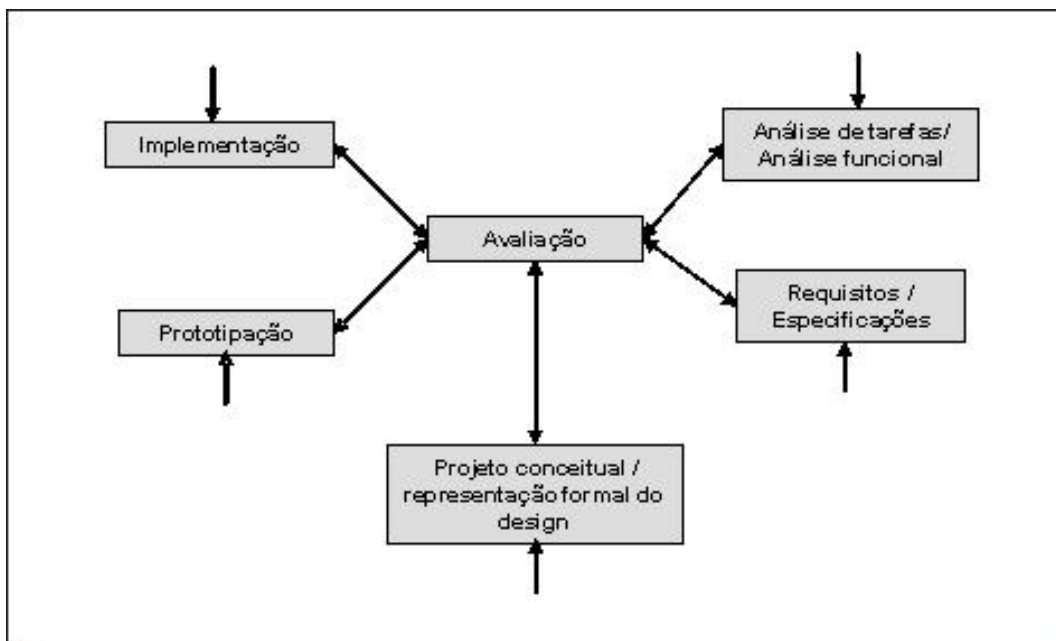
pela equipe e dos problemas encontrados nas avaliações. O produto final emerge da evolução da ideia inicial “bruta” até o produto acabado.



**Figura 6** - Um modelo simples de *design* de interação

**Fonte:** Preece, Rogers e Sharp (2005)

O modelo Estrela não especifica um ordenamento das atividades, mas sua flexibilidade exige que uma avaliação sempre seja feita antes de iniciar uma nova atividade. Os *designers* realizam o *design* alterando entre duas visões: a visão do sistema (trabalho analítico, visão *top-down*, mais organizada e formal) e a visão do usuário (trabalho sintético, visão *bottom-up*, mais criativa e *ad-hoc*). Conforme ilustrado na Figura 7, o modelo apresenta no centro a atividade de avaliação, interligada a quatro atividades: implementação, análise de tarefas/funcional, a prototipação e análise de requisitos/especificação. A avaliação está interligada às versões alternativas (projeto conceitual/representação formal do *design*).



**Figura 7** - Modelo de ciclo de vida Estrela  
**Fonte:** Preece, Rogers e Sharp (2005)

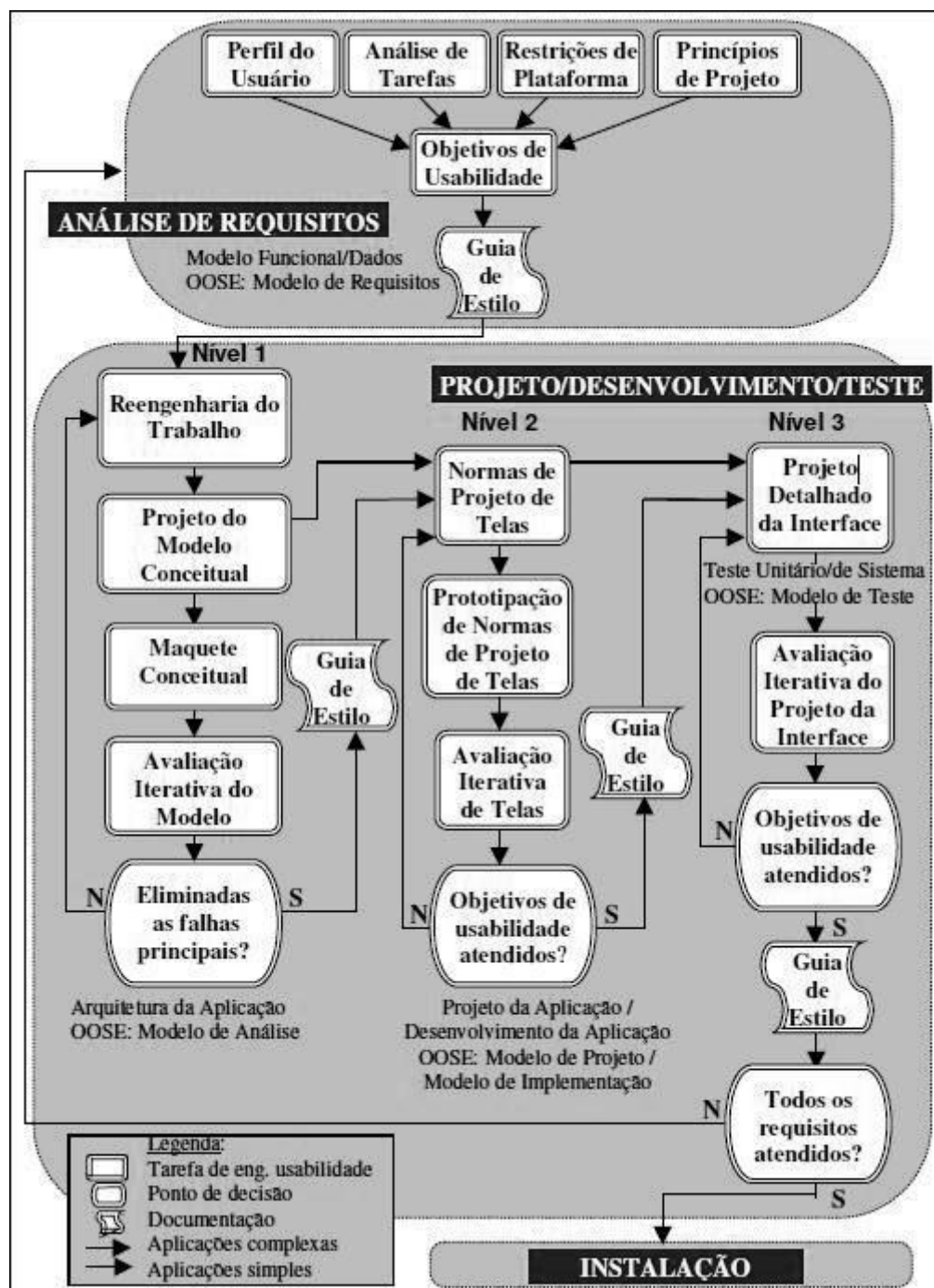
O modelo da Engenharia de Usabilidade originou-se de iniciativas de cientistas como Card, Moran e Newell (1983) e Norman (1983), que produziram os primeiros estudos de estruturas e processos cognitivos, com o objetivo de favorecer a concepção de interfaces humano-computador mais adaptadas. Conforme Mayhew (1999), o modelo é baseado em diversas disciplinas, como a psicologia cognitiva (que estuda a percepção e cognição humanas para a orientação do projeto das interfaces homem-computador), psicologia experimental (que utiliza métodos empíricos para a medida do desempenho do usuário e de sua satisfação), antropologia (que, por meio de técnicas como a etnografia, estuda aspectos culturais para a descrição das características dos usuários), ergonomia (que estuda os esforços físicos para maximizar o desempenho e reduzir a fadiga e o desconforto na utilização de sistemas) e engenharia de *software* (que orienta a integração da engenharia de usabilidade ao ciclo de vida dos sistemas). Essa abordagem propõe uma série de atividades que buscam o atendimento aos requisitos de usabilidade.

Mayhew (1999, 2008) apresenta um modelo de engenharia de usabilidade com estrutura aderente à proposta da ISO 13407 (1999) e de Preece, Rogers e Sharp (2005), com um ciclo de vida baseado em três grandes grupos de tarefas essenciais: tarefas de análise de requisitos, tarefas de projeto, desenvolvimento e teste e tarefas de instalação, conforme ilustrado na Figura 8.

O primeiro grupo de tarefas considera a análise de requisitos, que contempla as atividades de caracterização do perfil do usuário, análise do contexto de tarefas, capacidades/restrições da plataforma, princípios gerais do projeto e especificação dos objetivos de usabilidade.

O segundo grupo de tarefas considera projeto, teste e desenvolvimento, que contempla as atividades de reengenharia do trabalho, projeto do modelo conceitual, maquetes do modelo conceitual, avaliação iterativa do modelo conceitual, padrões de *design* de telas, protótipos dos padrões de *design* de tela, avaliação iterativa dos padrões de *design* de telas, *design* detalhado da interface com o usuário e avaliação iterativa do *design* detalhado da interface com o usuário.

O terceiro grupo considera a instalação, contemplando a atividade de *feedback* do usuário para verificação da satisfação e resolução das pendências.



**Figura 8 - Ciclo de Vida da Engenharia de Usabilidade**

**Fonte:** Avelino (2005)

Para Cybis, Betiol e Faust (2007), o ciclo da engenharia de usabilidade descreve as atividades e as relações e tem foco no usuário, permitindo a realização de sucessivos ciclos de análise, concepção e testes, com o necessário *feedback* dos resultados dos testes, de um ciclo a outro, identificando e refinando continuamente o conhecimento sobre o contexto de uso e as exigências em termos de usabilidade. A utilização de ciclos contínuos possibilita a construção de versões intermediárias da interface do sistema, que são submetidas a testes de uso pelos representantes dos

usuários. Nesse processo, tem-se, inicialmente, versões “grosseiras” de interfaces mas, com o avanço do desenvolvimento, concebem-se protótipos e versões mais acabadas do sistema, em simulações mais fidedignas a cada ciclo.

Outro modelo de processo aderente à engenharia de usabilidade é o método de *design* contextual rápido (HOLTZBLATT et al., 2005), que envolve os passos mencionados na Tabela 1. Podemos observar a semelhança entre esse método e o ciclo de vida da engenharia da usabilidade por vários fatores como a abordagem centrada no usuário, o uso de entrevistas e observações etnográficas para mapeamento do perfil do usuário e das tarefas envolvidas e o uso de *storyboards*, cenários, protótipos e maquetes para facilitar a compreensão das interações e a evolução da construção da interface.

**Tabela 1** - Passos do Modelo de *Design* Contextual Rápido

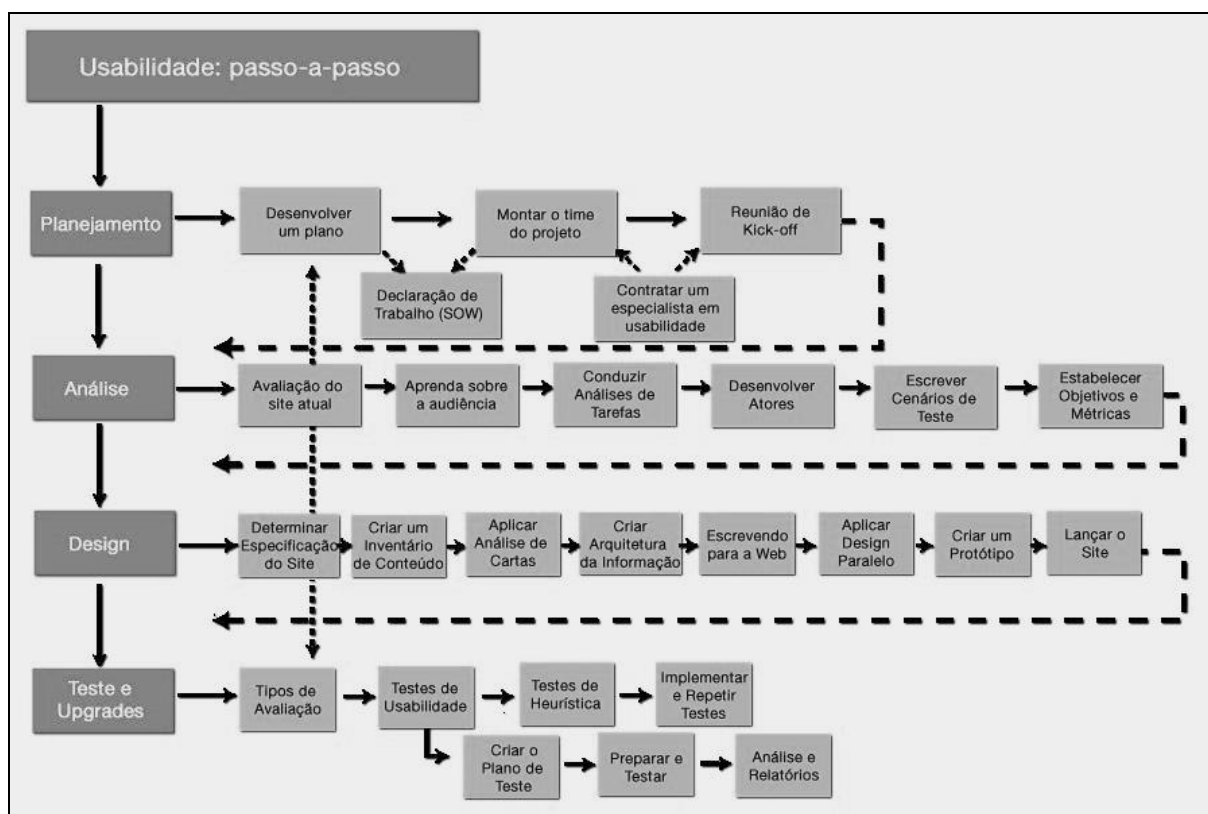
| <b>Passo</b>                                                        | <b>Objetivo</b>                                                                                                     |
|---------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------|
| Pesquisa contextual                                                 | Observar e entender as tarefas                                                                                      |
| Sessões de interpretação e modelagem do trabalho                    | Entender o fluxo do processo, os impactos políticos e culturais, anotação dos pontos-chave e afinidades (consensos) |
| Consolidação do modelo e construção de diagrama de afinidades       | Apresentar o que foi feito a uma população maior, buscando <i>insights</i> e concordâncias                          |
| Desenvolvimento de perfis                                           | Caracterizar o público-alvo (categorias de usuários)                                                                |
| Visualização ( <i>visioning</i> )                                   | Revisar e percorrer os dados consolidados                                                                           |
| <i>Storyboarding</i>                                                | Definir e ilustrar premissas                                                                                        |
| <i>Design</i> de ambiente de usuário (UED)                          | Representar coerentemente o ambiente. Construído a partir dos <i>storyboards</i> .                                  |
| Entrevistas e avaliações com protótipos de papel e modelos/maquetes | Garantir que o sistema atenderá os requisitos dos usuários finais.                                                  |

**Fonte:** Adaptado de HOLTZBLATT et al. (2005)

Shneiderman e Plaisant (2010) apresentam um processo “passo-a-passo” focado em usabilidade, conforme a Figura 9, que se assemelha em vários aspectos ao ciclo da engenharia de usabilidade. Esse processo contempla inicialmente uma etapa de Planejamento, que busca o detalhamento do escopo e da declaração do trabalho a ser criado e a identificação dos usuários. Nessa etapa também é planejada a composição da equipe de projeto, que podem contar com especialistas em usabilidade. Na etapa seguinte, de Análise, busca-se primeiramente avaliar a situação atual (processos, sistemas e interfaces), o que pode demandar avaliações



heurísticas, análise de *logs* do sistema e outras formas de se avaliar a usabilidade do atual sistema. Em seguida, é feita uma análise detalhada das tarefas, criação de cenários e atores e, finalizando a etapa de análise, o estabelecimento dos objetivos de usabilidade para o projeto. Os passos desta etapa são bem semelhantes aos primeiros passos da engenharia de usabilidade, embora em uma sequência um pouco diferente. Na etapa de *Design*, podemos observar técnicas semelhantes como o *design* paralelo (semelhante ao uso da técnica de *design* participativo na engenharia da usabilidade), além da abordagem de casos de uso e técnica de prototipação. Finalizando o processo, a etapa de Testes e Upgrades considera, assim como a engenharia da usabilidade, aspectos como o foco no planejamento, a importância dos ciclos de iteração e a abordagem de Nielsen (1993) nas avaliações heurísticas.



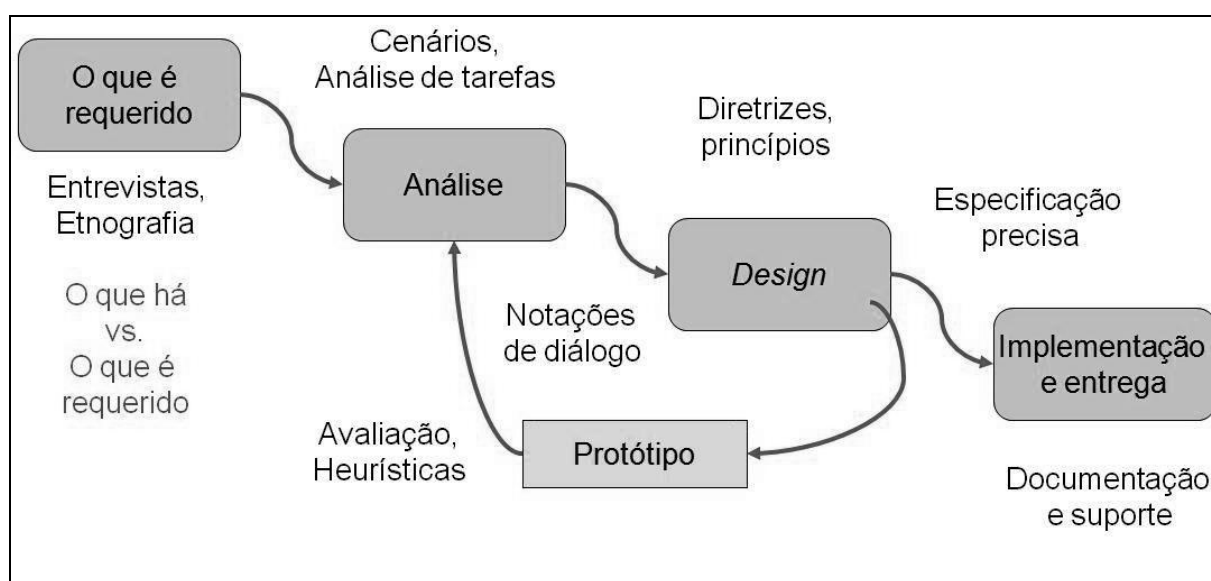
**Figura 9** - Guia passo a passo de avaliação de usabilidade

**Fonte:** Adaptado de <http://www.usability.gov/methods/process.html>, acesso em: 22 de Maio de 2012

DIX et al. (2004) apresenta uma visão de processo de *design* de interação, conforme a Figura 10, e afirma que: “Em algumas empresas os projetistas reclamam que são chamados tarde demais para participar do design da solução. Outras empresas entendem a usabilidade como equivalente a testes (checar se os usuários

*conseguem utilizar o sistema e consertar os erros). No entanto, nas melhores empresas a usabilidade é considerada desde o início do projeto”.*

Esse processo se baseia em quatro etapas e um ciclo de interação. Na primeira etapa identifica-se o que é realmente requerido para o sistema e qual a situação atual. Isso pode ser feito por meio de técnicas como entrevistas, gravações, e observações etnográficas. A etapa seguinte é a de Análise, onde os resultados da etapa anterior são organizados de modo que seja possível a identificação de elementos chave e a comunicação com os estágios seguintes no processo de *design*. São realizadas análises de tarefas e cenários para se documentar e ilustrar como os usuários envolvidos desempenham as tarefas e como se dão os processos de interação, tanto na situação atual como na situação desejada. A etapa de *Design* marca o momento de quando se move do “que se quer” para “como fazê-lo”. Nesta etapa, as diretrizes e princípios estabelecidos devem ser adotados e respeitados, notações de diálogo são utilizadas e as decisões de *design* são documentadas. Nesse momento há um ciclo de interação entre as etapas de Análise e *Design*, já que dificilmente se atinge da primeira vez o resultado esperado com o *design*. São realizadas avaliações do *design*, o que tipicamente envolve prototipação, para se avaliar a qualidade da solução e os pontos de melhoria. A última etapa é a criação e implementação do *design* finalizado, o que envolve a codificação, documentação e manuais.



**Figura 10** - Processo de *Design* de Interação  
**Fonte:** Adaptado de DIX et al. (2004 p.195)

Conforme mencionado anteriormente, considerando que a presente pesquisa teve o foco na criação de um guia que possa ser principalmente adotado no cenário brasileiro, tipicamente por organizações de grande porte voltadas ao desenvolvimento de sistemas primordialmente corporativos, que já possuam metodologia de desenvolvimento de *software* e processos definidos e consolidados, onde as metodologias tradicionais são amplamente utilizadas, foi adotado como base para este trabalho o ciclo de vida da engenharia de usabilidade.

A adoção de um ciclo de vida como uma espécie de “espinha dorsal” na qual serão inseridas as práticas de usabilidade das outras diversas fontes pesquisadas, facilita o entendimento, a corroboração e a sumarização destas para a confecção do guia de referência sem impedir que as organizações que adotem os demais ciclos de vida mencionados anteriormente (ou qualquer outro que contemple as práticas de usabilidade ao longo do desenvolvimento do sistema) possam adotar o guia proposto.

O ciclo de vida da engenharia de usabilidade considera e consolida o resultado da extensa pesquisa de diversos autores reconhecidos como Beyer, Dumas, Hackos, Jacobson, Mack, Mayhew, Nielsen, Norman, Ramey, Redish, Redish, Shneiderman e Wixon, entre outros (MAYHEW, 1999).

Ele contempla tarefas estruturadas para análise de requisitos de usabilidade, tarefas de definição de metas explícitas para usabilidade (orientadas diretamente pelos dados da análise de requisitos), tarefas que dão suporte a uma abordagem estruturada do desenho da interface (orientadas diretamente pelas metas de usabilidade e dados dos requisitos) e tarefas de avaliação da usabilidade para o projeto de interação (orientadas às metas de usabilidade definidas).

Novamente é notada a importância de se incorporar as atividades relacionadas à usabilidade no planejamento do projeto do sistema. O guia de avaliação de usabilidade do processo deve conter uma atividade que permita verificar se na etapa de planejamento do projeto está contemplado o planejamento de usabilidade. Um planejamento de usabilidade deve prover informações como o dimensionamento do nível de esforço e recursos necessários para implementar as tarefas de usabilidade mencionadas, a definição de datas e seqüenciamento das

atividades de usabilidade, a incorporação das atividades de usabilidade no cronograma do projeto, uma eventual análise de custo-benefício destas ações e a integração do plano de usabilidade ao plano do projeto (MAYHEW, 1999).

Mayhew (1999) menciona ainda a dificuldade de se implementar o ciclo de vida de engenharia de usabilidade por completo em uma organização e defende que em alguns casos pode ser indicada uma implementação gradativa por atividades e em uma amostragem reduzida de projetos. Pode-se começar, por exemplo, pela implementação das tarefas de teste relacionadas ao desenho detalhado da interface de usuário.

Desta forma, o guia proposto neste trabalho poderia avaliar se há ao menos uma adaptação das melhores práticas de usabilidade à maturidade dos processos da organização. Essa pode ser uma abordagem interessante, em que a implementação gradual e por partes de processo acompanha a evolução da maturidade em usabilidade da organização, a exemplo do que é feito na implementação de modelos como o CMMI.

## 4. Melhores Práticas de Usabilidade

Serão elencados, neste capítulo, vários exemplos das atividades (e respectivas técnicas) com o foco em usabilidade, que compõem os processos do ciclo de vida de projetos de interfaces, propostas pelos autores de referência pesquisados.

O objetivo é que o guia proposto pelo presente trabalho incorpore estas práticas como sugeridas e contemple o atendimento a tais práticas na avaliação do processo quanto à sua qualidade de usabilidade.

As práticas e técnicas são apresentadas considerando-se os respectivos grupos de processos aos quais elas pertencem, não apenas para facilitar o entendimento, mas também como sugestão de momento do ciclo de vida de desenvolvimento do *software* onde sua execução é geralmente mais apropriada ou recomendada pelos autores pesquisados.

Um ponto importante que merece ser lembrado é de que nem todas as atividades do ciclo de vida de usabilidade devem ser realizadas para todos os projetos de interfaces. No caso de sistemas que permitem interação por meio de linguagem de comando ou natural, por exemplo, algumas tarefas relacionadas ao detalhamento do modelo conceitual podem ser adaptadas para um nível de detalhes maior enquanto outras relacionados aos padrões de *design* de telas e *design* detalhado podem não se aplicar a essa forma de interação.

Outro exemplo é o caso de projetos que irão desenvolver uma nova interface (do “zero”) ou atender a um novo grupo de usuários. Neste caso, faz sentido executar todas as atividades. No caso de um projeto de manutenção ou *upgrade* de uma interface já existente (em que tais atividades já tenham sido executadas em um projeto anterior) talvez não faça sentido executar novamente alguns dos passos (tais como as atividades relacionadas à obtenção do perfil do usuário, dos objetivos de usabilidade, entre outros).

Por outro lado, outras atividades podem ser necessárias para projetos deste tipo (que não se aplicam a projetos de interfaces novas) como a comparação do

desempenho e de outras atividades do usuário com as versões anteriores do sistema (PREECE, ROGERS e SHARP, 2005).

Portanto, a fim de garantir o uso coerente da usabilidade em todos os projetos, pode-se criar um guia de adaptação de usabilidade, de forma semelhante ao que é indicado no nível três do CMMI (CHRISSIS, KONRAD e SHRUM, 2003), para as adaptações das tarefas aplicáveis a um projeto específico, no qual se documente quais atividades de usabilidade deverão ser necessárias para este projeto.

Este guia de adaptação pode fazer parte do Guia de Estilo do projeto (que será comentado mais adiante nessa dissertação) e deve ser elaborado logo no início do projeto. Deve haver também um processo que garanta que as atividades de usabilidade definidas estão realmente sendo cumpridas (por exemplo, por meio de auditorias de processo e produto, ao longo do projeto). O Guia de Estilo também deve ser objeto de atenção no processo de gestão de mudanças, ou seja, o guia pode impactar uma solicitação de mudança do projeto ou ser impactado por esta e, por isso, a usabilidade deve ser pauta da análise de impacto das mudanças. Pode-se utilizar, por exemplo, um processo organizacional definido e aderente ao processo SUP.10 (gestão de mudança) da ISO/IEC 15504-5 (2006) ou aos processos HCD.7.1 (gerenciar as mudanças) e HCD.7.2 (determinar o impacto na organização e nos *stakeholders*) da ISO/TR 18529 (2000).

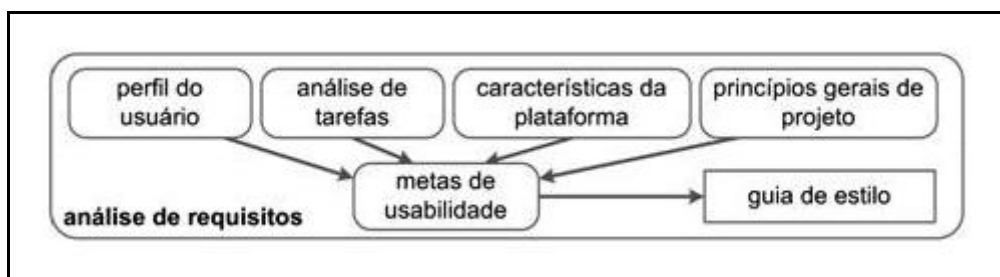
Conforme mencionado, as próximas seções apresentam uma visão detalhada sobre as três grandes fases e respectivas atividades propostas para o ciclo de vida: fase de Análise de Requisitos, fase de Projeto, Teste e Desenvolvimento e fase de Instalação.

#### **4.1 Atividades da Fase de Análise de Requisitos**

Esta fase contempla uma série de atividades, conforme ilustrado na Figura 11, que visam o detalhamento das características do usuário, das tarefas envolvidas no sistema e da arquitetura da solução.

Também são estabelecidas nesta fase as diretrizes de usabilidade que devem ser respeitadas ao longo do projeto, bem como os objetivos de usabilidade que

devem ser perseguidos. A principal documentação produzida ao final desta fase é um guia de estilo a ser utilizado para o projeto de interface.



**Figura 11** - Fase de Análise de Requisitos  
**Fonte:** Adaptado de BARBOSA e SILVA (2010)

Na obtenção do perfil do usuário, obtém-se uma descrição da população de usuários em termos de características relevantes ao projeto de interface (frequência de uso, nível de conhecimentos e habilidades com interfaces, por exemplo). Questionários e entrevistas podem ser utilizados para a obtenção dessas informações.

Na análise do contexto de tarefas é feito um mapeamento do processo de trabalho do usuário que será apoiado pelo *software* a ser desenvolvido, bem como as atividades envolvidas, suas descrições e o fluxo entre elas. Observações, entrevistas, análises de cenários e do ambiente de trabalho, *brainstormings* e elaboração de diagramas de afinidade podem ser utilizados para a obtenção e/ou documentação dessas informações.

Durante o levantamento das capacidades e restrições de plataforma são identificadas características da tecnologia escolhida como sistema operacional, manipulação direta, cores entre outras. Análise de documentações do ambiente operacional e entrevistas com especialistas podem ser utilizados para a obtenção dessas informações.

Na definição dos princípios gerais do projeto são levantadas diretrizes de engenharia de usabilidade que possam ser relevantes ao produto a ser desenvolvido. Consulta à literatura disponível (*guidelines*, *journals* e outras publicações relevantes) e a comunidades específicas e especialistas da área de usabilidade podem ser utilizados para a obtenção dessas informações.

Todas as informações obtidas por meio das quatro tarefas acima mencionadas são documentadas em um produto chamado guia de estilo.

Com o estabelecimento dos objetivos de usabilidade, são definidas metas qualitativas (que reflitam os requisitos de usabilidade extraídos das análises de perfil do usuário e do contexto da tarefa) e quantitativas (que definam critérios mínimos de desempenho esperado e da satisfação do usuário em termos de eficiência e eficácia) que devem ser cumpridas como critério de aceitação da interface produzida. Metas gerais de negócios da organização e *benchmarks* também podem ser considerados como fonte para a obtenção dessas informações.

As atividades da fase de Análise de Requisitos são essenciais para que sejam reunidas informações suficientes, relevantes e apropriadas de forma que um conjunto de requisitos estável possa ser produzido.

Dentre as principais técnicas podemos citar os questionários, entrevistas, grupos de foco (*focus groups*) e oficinas (*workshops*), observação natural e estudo da documentação.

Vale aqui ressaltarmos uma diferença de abordagens entre alguns autores pesquisados. Preece, Rogers e Sharp (2005) mencionam essas técnicas como forma de estabelecer requisitos enquanto Mayhew (1999) documenta um conjunto de técnicas semelhantes de modo mais detalhado na fase de Instalação (conforme seção “Obtenção do *Feedback* do Usuário” do presente trabalho) mas as elenca desde a fase de Análise de Requisitos (principalmente os questionários, observações e entrevistas) como forma de se obter o perfil do usuário, os requisitos e as tarefas.

O critério para escolha da técnica a ser utilizada para a análise de requisitos deve considerar fatores como o número de usuários que se pretende atingir, o volume e o tipo de dados com os quais se pretende atuar (qualitativo ou quantitativo) e a disponibilidade de tempo para a coleta de informações.

Preece, Rogers e Sharp (2005) apresentam um comparativo entre as diversas técnicas, conforme a Tabela 2.



**Tabela 2** - Comparativo de técnicas de coleta de dados

| <b>Técnica</b>                    | <b>Boa para</b>                                         | <b>Tipo de dados</b>                         | <b>Vantagens</b>                                                                                     | <b>Desvantagens</b>                                                                                          |
|-----------------------------------|---------------------------------------------------------|----------------------------------------------|------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------|
| Questionários                     | Responder a questões específicas                        | Dados qualitativos e quantitativos           | Pode atingir várias pessoas com poucos recursos                                                      | O <i>design</i> é crucial. O índice de resposta pode ser baixo. As respostas podem não ser o que você deseja |
| Entrevistas                       | Explorar questões                                       | Mais dados qualitativos do que quantitativos | Entrevistador guia o entrevistado se necessário. Encoraja o contato entre desenvolvedores e usuários | Requer tempo. Ambientes artificiais podem intimidar o usuário                                                |
| Grupos de foco e <i>workshops</i> | Coletar vários pontos de vista                          | Mais dados quantitativos do que qualitativos | Ressalta áreas de consenso e conflito. Encoraja o contato entre desenvolvedores e usuários           | Possibilidade de dominarem certos tipos de personalidade                                                     |
| Observação natural                | Entender o contexto da atividade do usuário             | Qualitativo                                  | Observar o trabalho real oferece percepções que outras técnicas não podem oferecer                   | Requer muito tempo. Envolve grande quantidades de dados                                                      |
| Estudo de documentação            | Aprender sobre procedimentos, regulamentações e padrões | Quantitativo                                 | Não compromete o tempo dos usuários                                                                  | O trabalho diário será diferente dos procedimentos documentados                                              |

**Fonte:** Adaptado de Preece, Rogers e Sharp (2005, p. 235)

Serão apresentadas a seguir as atividades que pertencem a esta fase de Análise de Requisitos: obtenção do perfil do usuário, análise do contexto de tarefas, levantamento das capacidades e restrições da plataforma, definição dos princípios gerais do projeto e estabelecimento dos objetivos de usabilidade, resultando na elaboração de um guia de estilo a ser utilizado para o projeto de interface.

#### **4.1.1 Obtenção do Perfil do Usuário**

Há um grande conjunto de indivíduos com alguma participação (*stake*) no desenvolvimento de um produto: os chamados *stakeholders*. Kotonia e Sommerville (1998) definem os *stakeholders* como “indivíduos ou organizações que serão afetados pelo sistema e que têm influência direta ou indireta nas necessidades desse sistema”.

Dix et al. (2004) observam ser muito pertinente, em uma abordagem centrada no usuário, lembrar que “*geralmente o ‘cliente’ formal que encomenda o sistema é*

*um dos últimos da lista dos que serão afetados por ele. Tenha muito cuidado com mudanças que diminuam o poder, a influência ou o controle de alguns stakeholders sem colocar nada tangível no lugar”.*

O envolvimento do usuário no processo de desenvolvimento de sistemas é muito importante para buscar a usabilidade da interface. Dois aspectos importantes envolvidos nessa participação do usuário, conforme Preece, Rogers e Sharp (2005), são o gerenciamento da expectativa (certificar-se de que as expectativas do usuário sejam realistas e atingíveis e que não haja surpresas na sua satisfação após a entrega do produto) e o sentimento de apropriação (aumentando seu comprometimento com o sistema e diminuindo sua resistência, uma vez que sintam que seu envolvimento contribuiu significativamente para o sucesso da interface gerada).

De acordo com Mayhew (1999) há duas principais maneiras de se obter o perfil do usuário: por meio de questionários aplicados diretamente aos usuários reais ou de entrevistas com indivíduos com conhecimento sobre a população de usuários. A autora salienta que embora a técnica de entrevistas seja mais rápida e barata, os questionários produzem resultados mais precisos e confiáveis.

Para tanto, a seguir serão elencados alguns passos para a obtenção de um perfil de usuário por meio de questionários.

Uma vez levantados os usuários, deve-se determinar as suas categorias, como por exemplo, papel, cargo ou função no sistema, além de outras características relevantes dos usuários. Os usuários também podem ser categorizados conforme seu nível de habilidade no uso da interface. Uma possível classificação, conforme Shneiderman e Plaisant (2010), pode ser: novatos (primeira vez), conhecedores ocasionais ou especialistas frequentes. Tal categorização permite que se considere para cada categoria aspectos da interface mais relevantes e que exigem maior atenção por parte da equipe responsável pelo projeto do sistema. Para usuários novatos ou que utilizam a interface pela primeira vez, requer-se atenção maior às instruções, caixas de diálogos e ajuda *online*. A interface deve ter poucas ações, vocabulário restrito e prover maior *feedback*. Já no caso de conhecedores ocasionais a atenção deve ser quanto ao uso de menus estruturados,

sequências consistentes, reconhecimento e proteção a danos. Para especialistas frequentes as respostas devem ser rápidas, o *feedback* breve e o uso de atalhos e aceleradores é recomendado.

Com a categorização finalizada, deve-se desenvolver um esboço de questionário, adaptado ao projeto específico e com informações ao usuário como o propósito e benefício deste questionário. Uma boa prática é coletar *feedback* sobre esse esboço e revisar o questionário nesse momento.

A partir da primeira versão do questionário, um piloto deve ser realizado, entrevistando um grupo (dois ou três usuários por categoria, por exemplo) a respeito da qualidade do questionário (checando clareza, exclusividade mútua e múltiplas escolhas, dentre outros fatores) e revisar o questionário, incorporando os eventuais ajustes necessários.

Preece, Rogers e Sharp (2005) identificam algumas recomendações gerais em relação ao uso de questionários, mencionadas a seguir. Deve-se fazer perguntas claras e específicas e, sempre que possível, fechadas, com várias possibilidades de resposta. Pode-se considerar incluir uma opção “não tenho opinião” para questões que busquem opinião. A ordem das perguntas é importante, de forma que uma não influencie a outra. Perguntas gerais devem preceder perguntas específicas. Perguntas múltiplas e complexas devem ser evitadas. Em relação a escalas, a ordem deve ser intuitiva e consistente entre as questões. Deve-se considerar se a escala irá utilizar um número par ou ímpar de pontos. Um número ímpar apresenta um ponto central claro enquanto o par compele o participante a tomar uma decisão. Pode-se usar, por exemplo, escala Likert ou de diferencial semântico. Diferentes populações de usuários podem requerer diferentes questionários. As instruções sobre a forma de preenchimento devem ser claras (atribuir notas, marcar com um “X”, marcar “Sim” ou “Não”, quais são as escalas, etc). Por último, os autores recomendam que se evite questionários longos, já que custam mais e inibem a participação.

Uma vez concluído o piloto, inicia-se o planejamento para o uso efetivo dos questionários revisados. Uma amostra de usuários deve ser selecionada. A amostra deve cobrir o mesmo número de usuários em cada categoria (MAYHEW, 1999).

Como é comum, dependendo da motivação dos usuários, obter-se resposta de no máximo 30% dos questionários enviados no caso de usuários internos e de no máximo 10% no caso de usuários externos, esta seleção da amostra é um passo muito importante. Algumas formas de se aumentar a taxa de respostas, conforme Preece, Rogers e Sharp (2005) são assegurar-se de que houve um bom planejamento e que o questionário foi bem projetado. Se o questionário for baseado em papel, deve-se facilitar o envio das respostas (envelope, selo, etc) e se for *online* (*email*, *web*), deve-se facilitar o acesso (compatibilidade com sistema operacional, navegadores, configurações de vídeo, *fonts*, etc). O motivo e a importância do questionário devem ser explicados, o anonimato deve ser assegurado e incentivos realmente atrativos para o preenchimento podem ser utilizados. Um cuidado com viéses é importante como, por exemplo, solicitar envio por email quando apenas parte da população usa email com rotina. O meio de resposta deve ser facilitado e o prazo final para resposta deve estar claro, mas mantendo a possibilidade de recebimento após tal data.

Os dados recebidos devem ser registrados por meio de planilhas ou sistemas e deve-se considerar quaisquer comentários extras dos usuários. Ao final, um resumo dos dados deve ser produzido a fim de possibilitar a análise e interpretação. Um engenheiro de usabilidade pode auxiliar na interpretação, bem como para sumarizar as necessidades e variações de cada categoria.

O passo final é a apresentação dos resultados às partes interessadas, com as conclusões e eventuais implicações ao projeto.

Shneiderman e Plaisant (2010) consideram um aspecto importante em relação à determinação do perfil de usuários: a questão de Usabilidade Universal. Quando há uma diversidade muito grande de usuários, compreendendo diferentes estilos (introvertido ou extrovertido, tolerante ou avesso a riscos, adotante prematuro ou tardio, sistemático ou oportunista), habilidades, conhecimentos, motivações, culturas, personalidades, gênero e idade, por exemplo, ou então uma alta diversidade de *hardware*, *browsers*, monitores, plataformas e redes (por exemplo, para produtos eletrônicos, serviços na Web ou de *e-government*) podem ser necessárias algumas análises para que o projeto da interface considere tais aspectos. Ferramentas como o MBTI (*Myers-Briggs Type Indicator*) ou o *Big Five*

*Test* (modelo OCEAN) podem auxiliar a avaliar os diferentes perfis ou tipos psicológicos dos usuários para que o projetista do sistema considere na concepção da interface tais diferenças e preferências. Sistemas que serão utilizados internacionalmente também devem considerar aspectos de diversidade cultural como: linguagem, caracteres e algarismos, formatos de data, notações de peso e medida, padrões para números de telefone e endereço, nomenclaturas de documentos e pronomes de tratamento, etiquetas, padrões morais, legislação local e grau de formalidade, entre vários outros itens.

A NBR ISO/IEC 12207 (2009) destaca um processo (item 6.4.1 da norma) para a definição dos requisitos de *stakeholders* e reforça a questão da atenção à usabilidade nas tarefas envolvidas nesse processo (item 6.4.1.3.2.4 da norma).

A NBR 9241-11 (2002) também destaca uma atividade para o detalhamento do perfil do usuário, com a qual características relevantes dos usuários precisam ser levantadas, como: conhecimento, habilidade, experiência, educação, treinamento, atributos físicos e capacidades sensoriais e motoras. Essa norma menciona também que pode ser necessário definir os tipos de usuários (níveis de experiência e diferentes funções desempenhadas). Observamos que tais características destacadas pela norma são semelhantes às citadas pelos demais autores referenciados no presente trabalho.

A ISO/TR 18529 (2000) trata questões relacionadas a obtenção do perfil do usuário principalmente em práticas como a HCD.1.1, HCD.1.2 e HCD.1.5 (voltadas à garantia da abordagem centrada em humano na estratégia) e HCD.3.2, HCD.3.3, HCD.3.4 e HCD.4.1 (voltadas aos requisitos organizacionais e de *stakeholders*).

#### **4.1.2 Análise de Contexto da Tarefa**

A NBR 9241-11 (2002) alerta que um produto pode ter níveis significativamente diferentes de usabilidade quando usado em diferentes contextos e ressalta que convém que qualquer descrição das atividades e passos envolvidos no desempenho da tarefa estejam relacionados aos objetivos a serem alcançados.

A seguir serão elencados alguns passos que podem ser seguidos para se obter uma boa análise de contexto das tarefas dos usuários.

O primeiro passo é a revisão da especificação de requisitos. Quando disponíveis nesse estágio do ciclo de vida, a documentação sobre os requisitos auxilia no entendimento das fronteiras da aplicação e na visão geral da automação que será implementada. Ao longo do ciclo de vida (já iniciando nesta fase de Análise de Requisitos) os requisitos funcionais e não-funcionais serão coletados, elicitados e capturados. Exemplos de diferentes tipos de requisitos são mostrados por Preece, Rogers e Sharp (2005) como requisitos funcionais (o que o produto deveria fazer), requisitos de dados (tipo, volatilidade, tamanho/quantidade, persistência, precisão e valor das quantidades de dados exigidos), requisitos ambientais (circunstâncias em que se espera que o sistema opere, e que podem ser referentes a quatro aspectos: físicos, sociais, organizacional e técnico), requisitos do usuário (características do grupo de usuários pretendido, como novato/especialista, frequente/casual) e requisitos de usabilidade (voltados às metas de usabilidade). Para a revisão desses requisitos, deve-se reunir os membros da equipe de projeto (a fim de obter maior comprometimento e buscar um equilíbrio entre a visão centrada no sistema e a visão centrada no usuário) e com usuários representativos. É elaborada uma lista dos principais artefatos envolvidos nas tarefas e como estes são utilizados. Deve-se observar também aspectos do ambiente físico e sócio-cultural que não tenham sido levantados até o momento. O intuito é que o sistema se encaixe corretamente com os demais processos, ferramentas e ambiente ao qual ele será inserido.

Um aspecto importante a ser considerado nesta atividade é a questão da revisão dos requisitos de segurança e seu impacto na usabilidade do sistema. Avelino (2005) conduziu um trabalho de pesquisa (também baseado no ciclo de vida da engenharia de usabilidade para a criação de uma metodologia de especificação de requisitos) que aborda a relação entre a usabilidade e os requisitos de segurança de um sistema. No caso de sistemas onde há grande atenção nos aspectos de segurança (por exemplo, sistemas críticos), técnicas de análise de erros humanos podem ser utilizadas para identificar causas desses erros e, em alguns casos, sua predição, permitindo a busca por um balanceamento entre o nível de automação e o nível de risco do sistema. Neste caso, erros que, por exemplo, iniciem, prolonguem ou agravem um incidente devem ser cuidadosamente considerados na avaliação da interface.

Alguns modelos de análise de erros humanos permitem uma estimativa de probabilidade de cada um dos erros e um cálculo do risco do sistema. Isso pode ser utilizado para a comparação com os níveis aceitáveis pela organização e uma tomada de decisão quanto a conseqüentes ações para redução dos riscos. Tais ações podem ter relação direta com a usabilidade do sistema como, por exemplo, mudança de *hardware*, aumento na tolerância a falhas e melhoria na recuperação de erros (AVELINO, 2005). Portanto, dependendo da natureza do sistema, pode ser importante haver atividades durante o ciclo de vida do projeto da interface (não apenas nas atividades da fase de análise de requisitos mas também nas demais fases) que garantam que haja uma aderência e coerência entre os aspectos de segurança e de usabilidade.

O processo de Elicitação de Requisitos (ENG.1) da ISO/IEC 15504-5 (2006) reforça, por meio da prática básica ENG.1.BP2, a importância de técnicas como a observação, protótipos, simulações, cenários e outras como forma de revisão dos requisitos e solicitações junto aos clientes.

Uma vez revisados os requisitos do sistema, passa-se à identificação e documentação dos atores-chave e casos de uso. O objetivo é ter um foco nas tarefas-chave (casos de uso), tipos de usuários (atores) e artefatos/objetos associados entre eles. Se há muitos casos de uso e atores, pode-se escolher uma amostragem relevante em relação ao escopo do produto. Quanto à abordagem de caso de uso, pode-se utilizar a tradicional (JACOBSON et al., 1992) ou a essencial (CONSTANTINE e LOCKWOOD, 1999). Uma técnica alternativa pode ser o uso de Análise Hierárquica de Tarefas (AHT) onde parte-se de um objetivo do usuário e a seguir se desmembram (em ordem hierárquica) todas as tarefas e sub-tarefas.

O passo seguinte é a realização de entrevistas e observações contextuais. Esse é um passo mais amplo e complexo e um aprofundamento foge do objetivo do presente trabalho, portanto, apenas uma visão geral dos principais passos será apresentada, embora Mayhew (1999) e Preece, Rogers e Sharp (2005) apresentem um detalhamento bem maior sobre a condução efetiva de tais atividades. Esse passo pode ser repetido posteriormente após a liberação de uma primeira versão do sistema. O objetivo desse passo é captar o trabalho real dos usuários (inclusive o

conhecimento tácito e soluções de contorno e não apenas o explícito em manuais, políticas e procedimentos) no ambiente real de trabalho dos mesmos.

Em relação a entrevistas, estas podem ser abertas/não estruturadas, estruturadas ou semi-estruturadas. Nas entrevistas abertas/não-estruturadas (*open-ended*) o formato e conteúdo das respostas não é pré-determinado, o que permite com que os próprios usuários identifiquem os pontos relevantes. Requer atenção ao foco e a questões relevantes e um tempo maior para análise (PREECE, ROGERS e SHARP, 2005). Nas entrevistas estruturadas as perguntas são fechadas ou pré-determinadas (como em questionários), direcionando-se os usuários aos itens que se deseja *feedback* e coletando as respostas em uma escala Likert, por exemplo. As entrevistas semi-estruturadas combinam características dos dois tipos mencionados, alternando-se questões fechadas com comentários e questões abertas.

Há ainda as entrevistas em grupo, também denominadas grupos de foco (*focus groups*), grupos focais ou grupos de estudos específicos. Nesse caso, novas questões e suporte a essas questões vão surgindo por meio da participação do grupo. Apresentam uma alternativa de baixo custo, mas exigem itens como um grupo que represente uma amostra representativa do grupo de usuários e um facilitador experiente que mantenha o foco nas questões relevantes. Cuidados especiais devem ser considerados para a escolha do facilitador e do ambiente adequado (espaço e infra-estrutura para a realização já que o número de participantes é maior), conforme destacam Preece, Rogers e Sharp (2005). Outras formas são as entrevistas por telefone ou *online* (síncronas ou assíncronas).

Em relação às observações, um estudo etnográfico junto ao ambiente do usuário pode ser de grande valia. A etnografia consiste em um método oriundo da antropologia e significa literalmente “descrever a cultura” (HAMMERSLEY e ATKINSON, 1983). Objetiva a observação ao invés da interpretação. Os observadores mergulham no ambiente de trabalho dos usuários e participam de todas as atividades de seu cotidiano. Essa experiência deve ser documentada e compartilhada com o time envolvido no projeto do sistema, a fim de que se valide se até mesmo pequenos detalhes das soluções propostas pelos *designers* vão ao encontro da dinâmica de trabalho real dos usuários.



Uma pesquisa conduzida por Lucy Suchman (SUCHMAN, 1987), revelou uma análise detalhada de como um sistema de ajuda não auxiliava os usuários em muitas ações e ressaltou a inadequação de se basear o *design* de um sistema interativo puramente em um modelo de usuário abstrato. Ela defende que os *designers* devem projetar sistemas interativos focando os detalhes singulares da situação particular (e real) do usuário ao invés de se basear em modelos preconcebidos de como as pessoas deveriam agir.

Entender o comportamento do usuário permite que se ressaltem prioridades, preferências e intenções implícitas (PREECE, ROGERS e SHARP, 2005). Existem técnicas específicas para lidar com a coleta e interpretação de dados do trabalho de campo (etnografia) e interpretá-los no *design* do sistema, como o Método da Coerência e o *Design* Contextual (PREECE, ROGERS e SHARP, 2005).

A seguir serão citadas algumas práticas recomendadas para entrevistas e observações, úteis para várias atividades do ciclo de vida como análise de contexto das tarefas, obtenção do perfil de usuário, avaliações iterativas e obtenção do *feedback* do usuário, conforme Mayhew (1999). É altamente recomendável que os participantes (condutor e entrevistados) recebam algum treinamento sobre etnografia e análise contextual de tarefas para entender os objetivos e técnicas desse passo. Ao selecionar os usuários, deve-se iniciar com aqueles que possuam domínio das tarefas para depois incluir os que não tenham tanta familiaridade (experiência e/ou frequência de uso) com o trabalho. Quando há tarefas específicas (baixa frequência, alta complexidade, condições especiais, por exemplo) é importante o agendamento com antecedência. Para a condução das entrevistas e observações, pode ser interessante criar um *script* ou manual a ser seguido pelo responsável pela execução, a fim de se orientar as instruções que devem ser transmitidas, o comportamento adequado e esperado e evitar falhas de comunicação e de interpretação. A captura dos dados pode ser feita por anotação livre, planilhas, fotos, gravação de vídeo ou áudio. A filmagem permite uma análise posterior mais detalhada mas acrescenta tempo ao processo e, portanto, deve-se considerar se haverá tempo para assistir e anotar as informações posteriormente. Devem ser registradas as características físicas do ambiente (iluminação, calor, nível de ruído e distrações) socioculturais (moral e motivação do grupo, como trabalham em equipe), o contexto do trabalho (frequência, importância da tarefa no conjunto

todo, entre outras) além de detalhes observados nas análises como, por exemplo, volume e frequência de transações, taxa de erro, nível de treinamento exigido, redundância na entrada de dados, gargalos de processo, grau de interrupção durante as tarefas, entre outros.

Outras práticas sobre o mesmo tema são citadas por Preece, Rogers e Sharp (2005) como evitar perguntas longas (dificultam a lembrança) e sentenças compostas por mais de uma pergunta (dificultam o registro e a análise), evitar jargões e perguntas tendenciosas (mesmo que inconscientes). As respostas devem ser registradas exatamente como foram dadas, sem ajustes estéticos, correções e alterações de forma alguma.

O passo seguinte, uma vez que já foram registradas todas essas informações a partir das entrevistas e/ou observações, é a elaboração de cenários de tarefas. Isso ajudará no desenvolvimento e teste da aplicação.

Um cenário consiste em uma “*descrição narrativa informal*” (CARROLL, 2000), geralmente na forma de uma história, que, ao descrever atividades ou tarefas humanas, permite a exploração e discussão de contextos, necessidades e requisitos (PREECE, ROGERS e SHARP, 2005). De acordo com Shneiderman e Plaisant (2010), cenários podem ser especialmente úteis quando há cooperação entre múltiplos usuários na interface (por exemplo, salas de controle, *cockpits* ou mesas de operação financeiras) ou múltiplos dispositivos físicos envolvidos (por exemplo, laboratórios ou processo de *check-in* de um hotel).

Os passos acima mencionados, desde as entrevistas até a documentação das tarefas e cenários, devem ser executados de forma iterativa (em ciclos) até que se tenha atingido um nível de informações adequado.

Para finalizar, deve-se assegurar que as tarefas básicas estão identificadas e que estas representam os casos de uso ou blocos de casos de uso. O usuário deve participar ou ao menos validar o resultado desse passo. Em seguida, deve-se obter um modelo organizacional das tarefas que identifique como as tarefas e artefatos se organizam (logicamente e em fluxo) em uma visão de hierarquia. Esse modelo criado será especialmente útil em uma tarefa posterior, denominada de “Reengenharia do Trabalho”.

### 4.1.3 Estabelecimento dos Objetivos de Usabilidade

Essa atividade requer a participação do máximo de envolvidos no projeto, como usuários, desenvolvedores, gestão, marketing, qualidade, suporte técnico, dentre outros (MAYHEW, 1999). Os dados obtidos do perfil do usuário podem ajudar a identificar a importância relativa entre os objetivos de desempenho e de preferência/satisfação do usuário. Os resultados documentados da análise contextual das tarefas também auxiliam já que demonstram características do contexto e ambiente de trabalho (local com barulho ou interrupções, alto volume de informações, por exemplo) que podem requerer cuidados especiais na construção das interfaces e, conseqüentemente, objetivos específicos para averiguar se a qualidade da usabilidade do sistema foi atingida.

Primeiramente, os objetivos gerais de negócios (como aumento de vendas ou de produtividade) devem ser levantados e decompostos em objetivos de usabilidade (como maior facilidade de uso ou desempenho da interface).

A partir disso, objetivos qualitativos de usabilidade já podem ser esboçados e priorizados por meio de classificações como imprescindíveis (seu atingimento é obrigatório), importantes (dependem do custo e tempo necessários) ou desejáveis (apenas se o custo for baixo).

Em seguida, deve-se formular objetivos quantitativos de usabilidade, que envolvem, tipicamente, medidas tangíveis como tempo para executar uma tarefa, número de erros encontrados durante o uso, número de tentativas para aprender a tarefa, índice de satisfação do usuário, entre outros, sempre considerando o tipo de usuário (especialista ou novato).

O tempo de resposta também é um fator crítico e pode ser um dos objetivos de usabilidade. Shneiderman e Plaisant (2010) ressaltam que o tempo de resposta deve ser apropriado à tarefa e fornece como diretrizes alguns parâmetros como o tempo de digitação, movimento de cursor e seleção de mouse (de cinquenta a cento e cinquenta milissegundos) e o tempo conforme a complexidade da tarefa (um segundo para tarefas simples ou freqüentes, de dois a quatro segundos para tarefas comuns e de oito a doze segundos para tarefas complexas). Um *benchmark* pode

ser realizado para se refinar as metas quantitativas, comparando as metas definidas com a dos concorrentes, por exemplo.

Os objetivos devem ser então priorizados e em seguida revisados com os usuários e gerência, a fim de se obter consenso e comprometimento sobre a sua importância para o sucesso do projeto.

Sommerville (2007) aponta que as avaliações de usabilidade devem ser conduzidas em relação a uma especificação baseada em atributos de usabilidade, conforme mostrado na Tabela 3.

**Tabela 3** - Atributos de usabilidade

| <b>Atributo</b>           | <b>Descrição</b>                                                              |
|---------------------------|-------------------------------------------------------------------------------|
| Facilidade de aprendizado | Quanto tempo leva para que um novo usuário torne-se produtivo com o sistema?  |
| Velocidade de operação    | Qual a adequação da resposta do sistema com a prática de trabalho do usuário? |
| Robustez                  | Quão tolerante é o sistema em relação aos erros do usuário?                   |
| Facilidade de recuperação | Quão bom é o sistema quanto à recuperação de erros do usuário?                |
| Facilidade de adaptação   | Quão fielmente o sistema está ligado a um único modelo de trabalho?           |

**Fonte:** Adaptado de Sommerville (2007)

A ISO/IEC 15504-5 (2006) destaca no processo de gestão da qualidade (MAN), respectivamente por meio das práticas MAN.4.PB1, PB4 e PB5, que os objetivos de qualidade devem ser estabelecidos, controlados por um sistema de gestão da qualidade e monitorados quanto ao seu progresso e atingimento. Desta forma, se a usabilidade for considerada um atributo da qualidade, os objetivos de usabilidade devem ser considerados em tais processos.

A ISO/TR 18529 (2000) também reforça a importância do estabelecimento de objetivos em práticas como a HCD 3.1.e 4.2 (metas e objetivos de qualidade).

Conforme mencionado anteriormente, pela norma NBR 9241-11 (2002), os objetivos de usabilidade devem considerar três atributos: eficiência, eficácia e satisfação do usuário. A Tabela 4 apresenta exemplos de medidas de usabilidade para propriedades desejáveis do produto, sugeridos por esta norma.

**Tabela 4 - Exemplo de medidas para propriedades desejáveis do produto**

| <b>Objetivos de usabilidade</b>                                 | <b>Medidas de eficácia</b>                                                                                       | <b>Medidas de eficiência</b>                                                                                                                        | <b>Medidas de satisfação</b>                           |
|-----------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------|
| Adequado às necessidades de usuários treinados                  | Número de tarefas importantes realizadas;<br>Porcentagem de funções relevantes usadas                            | Eficiência relativa comparada com um usuário experiente                                                                                             | Escala para satisfação com características importantes |
| Adequado às necessidades para usar facilmente                   | Porcentagem de tarefas completadas com sucesso na primeira tentativa                                             | Tempo gasto na primeira tentativa <sup>1)</sup> ;<br>Eficiência relativa na primeira tentativa                                                      | Taxa de uso voluntário                                 |
| Adequado às necessidades para uso não freqüente ou intermitente | Porcentagem de tarefas completadas com sucesso depois de um período específico sem uso                           | Tempo gasto reaprendendo funções <sup>1)</sup> ;<br>Número de erros persistentes                                                                    | Frequência de reuso                                    |
| Redução de necessidade de suporte                               | Número de referências para documentação;<br>Número de chamadas ao suporte;<br>Número de acessos para obter ajuda | Tempo produtivo <sup>1)</sup> ;<br>Tempo para aprender por critério <sup>1)</sup>                                                                   | Escala para satisfação com recursos de apoio           |
| Facilidade de Aprender                                          | Número de funções aprendidas;<br>Porcentagem de usuários que conseguem aprender por critério                     | Tempo para aprender por critério <sup>1)</sup> ;<br>Tempo para reaprender por critério <sup>1)</sup> ;<br>Eficiência relativa durante o aprendizado | Escala para facilidade de aprendizado                  |
| Tolerância a erros                                              | Porcentagem de erros corrigidos ou apresentados pelo sistema;<br>Número tolerado de erros do usuário             | Tempo gasto na correção de erros                                                                                                                    | Escala para tratamento de erros                        |
| Legibilidade                                                    | Porcentagem de palavras lidas corretamente em uma distância normal de visualização                               | Tempo para ler corretamente um número especificado de caracteres                                                                                    | Escala para desconforto visual                         |

<sup>1)</sup> Convém que nesses exemplos os recursos sejam medidos em relação a um nível especificado de eficácia.

**Fonte:** NBR 9241-11 (2002, pag. 11)

Dix et al. (2004) apresentam uma série de métricas de usabilidade que podem ser consideradas para o estabelecimento de objetivos, como: tempo para concluir uma tarefa, percentual de tarefa concluída, percentual de tarefa concluída por unidade de tempo, proporção de sucessos para falhas, tempo gasto em erros, percentual ou número de erros, percentual ou número de competidores melhor do que o sistema, número de comandos usados, frequência de uso da ajuda e da documentação, percentual de comentários de usuários favoráveis/desfavoráveis, número de repetições de comandos com falha, número de execuções com sucesso e fracasso, número de vezes em que a interface confunde o usuário ou induz a erro, número de características boas e más recordadas por usuários, número de comandos disponíveis não utilizados, número de comportamentos regressivos, número de usuários preferindo o seu sistema, número de vezes em que os usuários precisam contornar um problema, número de vezes em que o usuário é interrompido em uma tarefa, número de vezes em que o usuário perde o controle do sistema e número de vezes em que o usuário expressa frustração ou satisfação.

#### 4.1.4 Levantamento das Capacidades e Restrições de Plataforma

Esta tarefa visa identificar todos os aspectos relevantes de *hardware* e *software* da(s) plataforma(s), buscando itens como estações de trabalho, dispositivos, ferramentas de desenvolvimento e sistema operacional envolvidos.

Uma atenção especial deve ser dada a interfaces que utilizam dispositivos de entrada ou saída não usuais. No caso de dispositivos de saída como displays e monitores, características importantes devem ser consideradas como a dimensão, resolução (número de *pixels*), número de cores, luminosidade, taxas de utilização (no caso de permitir animações e vídeos), portabilidade e privacidade, entre outras (SHNEIDERMAN e PLAISANT, 2010). Dispositivos de saída não convencionais como telas grandes (*walls* ou monitores múltiplos) ou pequenas (óculos ou lentes) ou *displays* do tipo *heads-up* ou *helmet/head mounted display* (HMD) podem requerer cuidados específicos para que a usabilidade não seja comprometida. O mesmo se aplica a dispositivos de entrada diferenciados ou menos usuais como *trackball*, *trackpoint*, controles para mãos (luvas) ou pés, *joysticks*, sensores, papel digital, teclados diferenciados (coletores ou *smartphones*) e mesas (*surface*), entre outros.

Caso mais de uma plataforma for identificada e/ou se os projetistas não estão familiarizados com as capacidades e restrições da plataforma, deve-se revisar a documentação das plataformas, que podem conter informações sobre padrões de interface de alguns componentes, por exemplo, e entrevistar o pessoal técnico de posse das informações já levantadas. Quando há vários projetistas e/ou plataformas envolvidos, é importante que se documente as capacidades e restrições de plataforma identificados (MAYHEW, 1999, 2008).

Organizações que utilizam a ISO/IEC 15504-5 (2006) podem utilizar o processo organizacional definido e aderente principalmente ao processo RIN.4 (infra-estrutura).

#### 4.1.5 Definição dos Princípios Gerais de Projeto

Os princípios gerais são as diretrizes de usabilidade estabelecidas pela organização e que devem ser seguidas ao longo de todo o projeto.

Esse passo não precisa ser realizado a cada projeto de desenvolvimento de sistemas caso os princípios já tenham sido definidos anteriormente e incorporados de alguma forma à metodologia da organização. Entretanto, a organização deve possuir processos que garantam que tais princípios sejam seguidos pelos projetistas em todos os projetos. Uma boa prática é que tais princípios sejam ainda periodicamente revistos já que evoluções na tecnologia e na própria maturidade da organização no desenvolvimento de *softwares* podem requerer alterações nos princípios definidos.

Serão apresentadas a seguir uma série de diretrizes que podem ser incorporadas aos princípios de uma organização.

Um primeiro passo para a definição dos princípios é revisar quaisquer outros guias de estilo de alto nível relevantes como guias e fontes de lições aprendidas desenvolvidas previamente (MAYHEW, 1999).

Guias corporativos ou de família de produtos, por exemplo, podem ter diretrizes como de “*look and feel*” como logotipos a serem usados, cores padronizadas, ícones que devem ser usados para representar determinadas funções, posicionamento padrão de determinadas barras e menus, itens e/ou menus que devem usar o princípio de agrupamento e se há uma nomenclatura padrão (para que o usuário encontre rapidamente sua opção sem ficar confuso), uso de cores e fundos, *font* padrão a ser usada em menus, em caixas e em mensagens, padrão de terminologia e abreviações, padrão sobre o uso de *fonts* maiúsculas e minúsculas, espessura de linhas, entre outras.

Shneiderman e Plaisant (2010) citam outros exemplos de diretrizes como o formato de perguntas, comentários e de mensagens de erro, justificção, espaços em branco e margens e adaptação à exibição em telas pequenas e grandes. Tais diretrizes devem ser seguidas para garantir melhor usabilidade e uma identidade visual da interface.

Publicações específicas como livros, *journals*, fóruns e matérias de conferências, por exemplo, podem trazer boas contribuições. Conjuntos de diretrizes (*guidelines*) como as Heurísticas de Nielsen (1993), descritas neste trabalho na seção de “Avaliação Heurística”, e as Oito Regras de Ouro de Shneiderman e

Plaisant (2010), descritas neste trabalho na seção de “Definição dos Princípios Gerais de Projeto” podem fazer parte dos princípios.

Há inúmeras diretrizes, por exemplo, voltadas a *web design*, sendo que algumas referências estão disponíveis na própria *web*. Nielsen e Loranger (2007) propuseram um conjunto de diretrizes de usabilidade baseadas em evidências empíricas, provenientes de testes de 716 *websites* com 2.163 usuários espalhados pelo mundo e de uma fonte mais específica que contou com 69 usuários que testaram 25 *websites* de vários gêneros como indústria, serviços, entretenimento, medicina e culturais (WATANABE e DUDUCHI, 2009).

A seguir serão citadas algumas destas diretrizes propostas, relacionadas à navegação e a arquitetura da informação, textos, apresentação dos elementos das páginas e da página principal.

Quanto às diretrizes propostas para navegação e arquitetura de informação, um *website* não deve ser estruturado de modo a refletir a estrutura hierárquica da empresa exceto quando esse *website* for dirigido ao público interno, pois essas estruturas podem confundir o usuário já que não fazem sentido aos mesmos. A estrutura do site deve se manter consistente de uma página a outra, algo particularmente difícil em grandes sites, que normalmente são compostos por sites menores desenvolvidos por equipes diferentes. Também deve-se evitar um *design* rebuscado, extravagante, exótico pois pode sobrecarregar a página e dificultar o uso. Deve-se evitar a redundância, provocada por exemplo por *links* e categorias duplicados. *Links* e *labels* devem ser curtos e bem especificados, pois chamam a atenção do usuário em meio a grandes quantidades de texto. O usuário deve ter a informação de onde está e como prosseguir ou retroceder para outras partes da tarefa e do site. Para os chamados *links* profundos com arquitetura hierárquica de informações, pode-se utilizar técnicas como o *breadcrumb trail* (“trilha de migalhas de pão”) que indica a localização atual do usuário no contexto hierárquico do site. Deve-se evitar a cor azul para textos que não são *links*, pois o padrão para identificação de *links* é o sublinhado na cor azul. Também devem ser evitados os menus com mais do que dois níveis pois encobrem a página, são difíceis de controlar e dificultam a navegação e busca da opção desejada (WATANABE e DUDUCHI, 2009).



Quanto às diretrizes propostas para textos, a tipografia não deve ser baseada apenas na marca ou logotipo da empresa ou ainda em preferências pessoais. O esquema de cores escolhido deve comunicar algo, ser legível, refletir a personalidade e o tom do site aos usuários para que estes encontrem o que procurem e consigam realizar suas tarefas. Quanto às *fonts*, deve-se optar pelas que normalmente estarão disponíveis para a maioria dos usuários pois caso esteja indisponível os sistemas dos usuários talvez utilizem uma *font* alternativa inadequada. São preferíveis as *fonts sans serif* (sem serifa), ou seja, aquelas mais simples e sem detalhes decorativos. As *fonts* com serifas (que possuem arremates nas extremidades de cada letra ou outros enfeites e variações sutis) são preferíveis para material impresso mas os estudos mostram que para a leitura em computadores as *fonts* sem serifas são as mais indicadas já que os monitores não oferecerem uma qualidade tipográfica adequada (WATANABE e DUDUCHI, 2009). Quanto ao tamanho, as *fonts* não devem ser menores do que dez pontos e a decisão deve considerar a idade dos usuários. O número de *fonts* e cores deve ser limitado e consistente ao longo das páginas de modo a refletir a identidade visual do *website*.

Quanto às diretrizes propostas para apresentação dos elementos de uma página, deve-se estruturar a página em ordem de prioridade, agrupar as áreas relacionadas, alinhar os elementos adequadamente para criar ordem, posicionar os elementos no lugar em que os usuários esperam encontrar e evitar a poluição com elementos desnecessários. Ao se oferecer páginas longas, deve-se atentar para que o conteúdo mais importante esteja na parte central e fornecer dicas visuais para que os usuários percebam o valor da rolagem, já que as pesquisas de Nielsen e Loranger (2007) demonstram que a tendência dos usuários é olhar exatamente para o meio da página e raramente olhar para a margem direita a fim de verificarem a barra de rolagem pois pressupõem que os elementos abaixo da tela não têm importância. Outra estratégia bem sucedida é criar *links* (entre três e cinco) diretos na página principal para as tarefas de maior prioridade e acesso pelos usuários.

Ainda em relação a diretrizes para aplicações *web*, Shneiderman e Plaisant (2010) destacam algumas outras referências como o *Sun's Web Design Guide* (SUN, 2008), *The National Cancer Institute's Research-Based Web Design & Usability Guidelines* (NCI, 2008), *The World Wide Web Consortium's Web*

*Accessibility Initiative* (WAI, 2008) e *The Web Style Guide* (LYNCH e HORTON, 2008).

As 388 diretrizes do Instituto Nacional do Câncer nos EUA (NCI, 2008), por exemplo, são voltadas à navegação pela interface e possuem um foco para páginas *web* informativas mas podem ser amplamente aplicadas a outros tipos de interface. Dentre essas diretrizes, podemos citar: padronize sequências de tarefas, assegure-se de que os *links* são descritivos, use títulos únicos e descritivos, use botões para escolhas mutuamente excludentes (binárias), desenvolva páginas que possam ser impressas corretamente e use miniaturas de imagens para pré-visualização de imagens maiores.

Há ainda diretrizes associadas aos objetivos de organização de layout, como as de Smith e Mosier (1986): consistência de transações de entrada de dados (usar sequências similares e abreviações), mínimo de entrada de dados pelo usuário (diminuir a chance de erros, usando listas ao invés de digitação e evitando redundância de digitação ou seleção de dados), mínima carga de memória para o usuário, compatibilidade entre dados exibidos e entrada de dados e flexibilidade ao usuário para controle da entrada de dados.

Em relação a diretrizes voltadas a busca e visualização das informações no sistema, Shneiderman e Plaisant (2010) elencam algumas orientações como as mencionadas a seguir. Para cada atividade, deve-se avaliar o melhor tipo de pesquisa, como por exemplo, busca em texto, banco de dados, imagens, mapas, diagramas, sons, vídeos ou outro. Para cada pesquisa, deve-se avaliar o melhor método, como por exemplo, consulta em linguagem natural, formulário de preenchimento, manipulação direta, consulta dinâmica ou automática concomitante à entrada de dados, consulta por exemplos, palavra-chave, busca por filtros simples, avançados ou com consulta booleana ou outro. Para cada pesquisa, deve-se também avaliar o melhor tipo de visualização e suas particularidades, como por exemplo exibição da informação automática ou após ação de confirmação, uso de *previews*, tipo de saída de dados (lista, tabela, gráfico, relatório, arquivo para download), necessidade de disponibilizar ordenação, classificação ou agrupamento pelo próprio usuário, visualização na janela atual da busca ou em nova janela, necessidade de visualização em *zoom* e/ou 3D, entre outros.

Há diretrizes voltadas a obter a atenção do usuário, mencionadas por Wickens e Hollands (2000), como, por exemplo, voltadas a: intensidade da *font* (dois níveis são recomendados ), marca (uso de *underline*, “\*”, *bullets*, *pointers*, “X”), tamanho de *font* (até quatro diferentes tamanhos no máximo), quantidade de *fonts* (até três no máximo), recurso de vídeo inverso (cor inversa, apenas quando necessário), recurso de “pisca” (usar com cuidado e moderação), cor da *font* (usar até quatro no máximo) e relacionadas aos recursos de áudio (como, por exemplo, haver distinção entre sons para *feedback* positivo e negativo).

Outro aspecto relevante é a questão das cores utilizadas na interface. Há um campo vasto de estudo nesse sentido já que as cores podem despertar atenção ou desinteresse do usuário, facilitar ou dificultar a organização lógica da informação exibida, evocar diferentes reações emocionais como encantamento, excitação, medo, raiva dentre outras. Shneiderman e Plaisant (2010) elencam também várias diretrizes associadas a este tema, como por exemplo: limite o número de cores e as use de forma conservadora, reconheça o poder de uma cor para acelerar ou retardar uma tarefa, esteja alerta a expectativas comuns sobre cores, deixe a codificação das cores sob controle do usuário, considere restrições de cores por usuários devido a deficiências visuais, seja consistente no uso das cores e esteja atento a problemas técnicos.

Outras diretrizes são voltadas a facilitação na entrada de dados, conforme elencam Smith e Mosier (1986): consistência na exibição de dados (padronização e controle), assimilação de informação eficiente pelo usuário (familiaridade e diagramação), mínima carga de memória para o usuário (informação entre uma tela e outra), compatibilidade entre dados exibidos / entrada de dados e flexibilidade ao usuário para a exibição dos dados (classificação de linhas e colunas, por exemplo).

Shneiderman e Plaisant (2010) propõem também as Oito Regras de Ouro para o *design* de interfaces:

- a) Esforce-se para consistência: na sequência de ações, na terminologia comum em *menus*, *prompts* e *helps*, cores, *layout* e *fonts*.
- b) Atenda a usabilidade universal: considerando a diversidade, a plasticidade e as categorias de usuários.

- c) Ofereça *feedback* informativo: adequado a cada ação (de simples a complexas).
- d) Projete diálogos para produzir fechamento: preveja sequências de ações divididas em grupos, com início, meio, fim e *feedback* ao final.
- e) Previna erros: desabilite opções inapropriadas e dê instruções de recuperação em casos de erro.
- f) Permita fácil reversão de ações: encoraja a exploração da interface.
- g) Suporte *locus* interno de controle: peritos desejam se sentir no comando da interface. Não querem surpresas. Evite entradas de dados entediante. Outras diretrizes a esse respeito podem ser observadas na heurística “Controle e liberdade do usuário” de Nielsen (1993), mencionada na seção “Avaliação Heurística” do presente trabalho.
- h) Reduza a carga de memória de curto prazo: as pessoas normalmente conseguem se lembrar de sete mais/menos dois pedaços de informação. Evite a necessidade de memória do usuário entre telas.

Um ponto importante observado por Mayhew (2008) é de que o *design* ideal não pode ser alcançado pela simples aplicação sistemática de diretrizes (*guidelines*) pois cada produto e grupo de usuários são únicos e, portanto, tais diretrizes devem ser adaptadas aos requisitos do produto e validadas contra estes.

Shneiderman e Plaisant (2010) propõem também os quatro “E”s para manter as diretrizes ativas nos processos: *Education* (treinamento e discussão das diretrizes), *Enforcement* (verificação de aderência das interfaces às diretrizes), *Exemption* (liberação rápida de idéias criativas ou novas tecnologias) e *Enhancement* (revisão periódica mantém diretrizes atualizadas).

Com a conclusão dos passos de todas as cinco atividades previamente descritas (obtenção do perfil do usuário, análise de tarefas, levantamento das capacidades e restrições, estabelecimento dos objetivos de usabilidade e definição dos princípios gerais) encerra-se a fase de Análise de Requisitos e obtém-se como principal saída desta fase a primeira versão do Guia de Estilo.

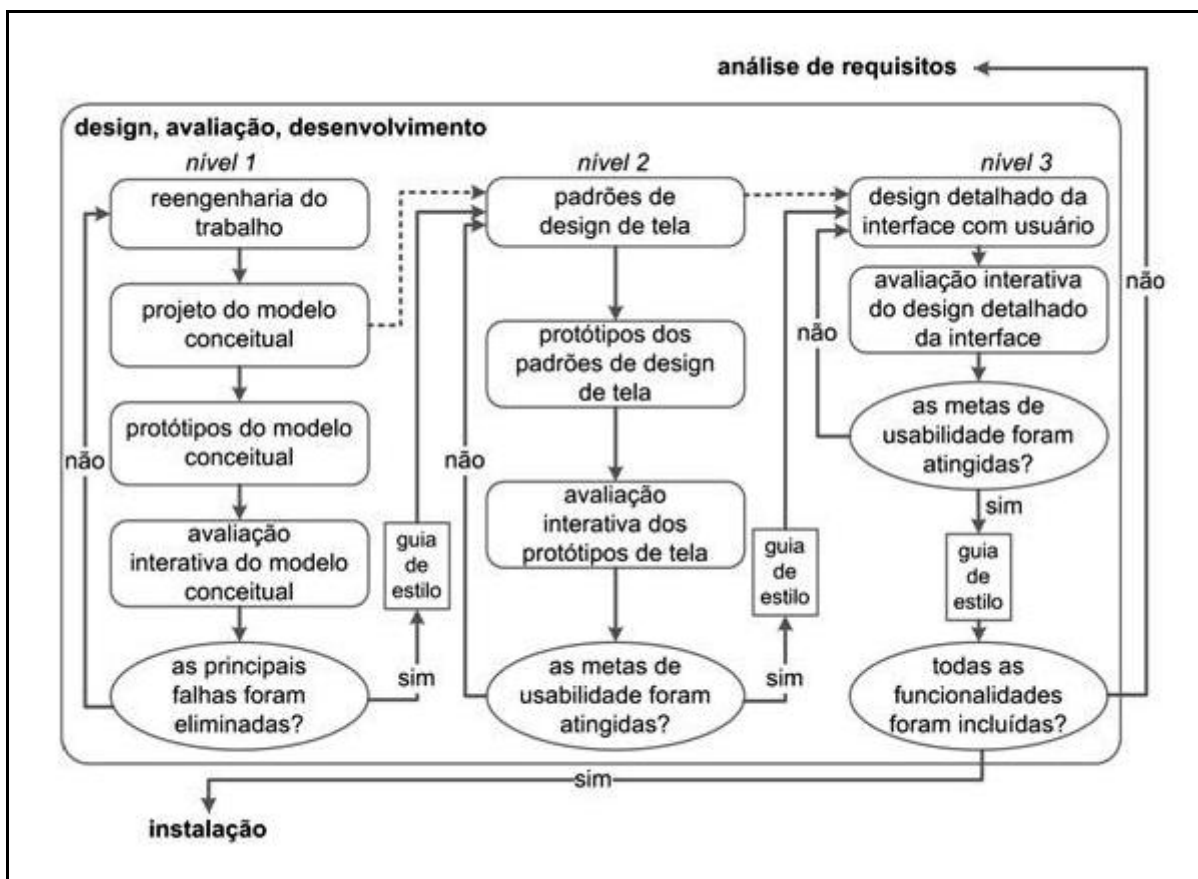
O Guia de Estilo é um documento que incorpora todas as informações obtidas até o momento e que será não somente utilizado como também progressivamente detalhado e revisado na próxima fase do ciclo de vida, descrita a seguir.

## **4.2 Atividades da Fase de Projeto, Teste e Desenvolvimento**

Esta fase contempla as atividades centrais voltadas ao desenvolvimento do sistema, de uma forma gradativa e progressiva, conforme ilustrado na Figura 12.

Mayhew (1999) apresenta uma divisão desta fase em três níveis, sendo que o primeiro nível endereça questões de projeto de alto nível, o segundo trata do estabelecimento de padrões e o terceiro finaliza o projeto de interface, baseado na conclusão dos resultados dos dois níveis anteriores. O documento de guia de estilo (progressivamente elaborado) é atualizado a cada nível e finalizado no último nível. Para a conclusão de cada nível, as respectivas atividades são repetidas (ciclo iterativo) até que as maiores falhas de projeto tenham sido identificadas e eliminadas, possibilitando que a qualidade da interface seja progressivamente avaliada e sujeita a ajustes e melhorias.

Corroborando a importância dessa abordagem de ciclos de iteração com contínuo aprofundamento e progresso da interface, Preece, Rogers e Sharp (2005) mencionam que *“toda iteração deveria envolver um progresso no design com cada vez mais profundidade”*. O progresso é atingido mediante ciclos de *design*, avaliação e novo *design* (*redesign*).



**Figura 12** - Fase de Projeto, Teste e Desenvolvimento

Fonte: Adaptado de BARBOSA e SILVA (2010)

A seguir veremos um breve resumo das atividades desta fase e posteriormente o seu detalhamento.

Na reengenharia do trabalho o modelo e fluxo de trabalho do usuário são redesenhados, baseado nos resultados extraídos da fase de análise de requisitos, de forma que se considere os objetivos de negócio e de usabilidade estabelecidos, buscando maior agilidade e produtividade. Análise da automação, percurso por cenários e revisão das novas tarefas com os usuários podem ser utilizados para a obtenção do novo processo.

Durante o projeto do Modelo Conceitual (MC) são geradas alternativas para o projeto de interface (ainda em um alto nível) especificando caminhos de navegação, interfaces principais e regras para seus componentes essenciais.

A elaboração de maquetes do MC visa a criação de produtos que permitam uma visualização das idéias obtidas e incorporadas ao projeto de alto nível concebido na tarefa anterior (ainda sem um detalhamento maior) e possibilitem a

validação e refinamento do MC. Protótipos e/ou maquetes no papel são técnicas que permitem ilustrar, por exemplo, desenhos, janelas e caixas de diálogo, e possibilitam a validação do MC criado.

Na avaliação iterativa do MC as maquetes produzidas são utilizadas pelos usuários, com o mínimo possível de treinamento prévio e intervenção por parte dos projetistas, simulando o uso da interface que está sendo criada e permitindo que se avalie, refine e valide o MC. Testes formais de usabilidade e métodos de inspeção podem ser utilizados para concluir esta atividade.

A definição dos Padrões de *Design* de Telas (PDT) visa buscar coerência e consistência (pilares da usabilidade) da interface. Os pilares são observados, por exemplo, por meio da padronização obtida na navegação e durante as interações do usuário entre as telas do sistema, garantindo aderência a padrões mandatórios da tecnologia utilizada (Microsoft Windows ou Apple Macintosh, por exemplo) e aos requisitos e modelo conceitual elaborados no primeiro nível.

Na elaboração dos protótipos dos PDT são apresentados produtos de baixa fidelidade, sem necessariamente conexão com bancos de dados, que permitam simulação com a interface, bem como sua validação e refinamento.

Durante a avaliação iterativa dos PDT são realizadas simulações mais realistas do uso das interfaces permitindo um re-desenho e re-validação das interações, gerando um conjunto robusto de padrões. Assim como no MC, testes formais de usabilidade e métodos de inspeção podem ser utilizados para concluir esta atividade.

A elaboração do *Design* Detalhado da Interface com o Usuário (DDIU) permite a entrega de um detalhamento completo da interface do sistema para que se possa prosseguir ao desenvolvimento do sistema.

Na avaliação iterativa do DDIU, uma validação ainda mais detalhada é conduzida, permitindo o acesso a funcionalidades e categorias de usuários até então não explorados devido à limitação da interface produzida nas etapas anteriores. A aderência aos objetivos de usabilidade e o seu atingimento são novamente verificados.

Serão apresentadas a seguir as atividades que pertencem a esta fase de Projeto, Teste e Desenvolvimento: reengenharia do trabalho, projeto do Modelo Conceitual (MC), elaboração de maquetes do MC, avaliação iterativa do MC, definição dos Padrões de *Design* de Telas (PDT), elaboração de protótipos dos PDT, avaliação iterativa dos PDT, elaboração do *Design* Detalhado da Interface com o Usuário (DDIU) e avaliação iterativa do DDIU.

#### **4.2.1 Reengenharia do Trabalho**

De acordo com Mayhew (1999), essa atividade visa cumprir as seguintes metas: perceber o poder e eficiência da automação (garantindo, porém, o controle humano), reorganizar o trabalho para suportar as metas de negócio, minimizar o esforço de re-treinamento para a nova forma de trabalho e maximizar a eficiência e eficácia considerando as restrições e capacidades cognitivas humanas.

Na fase de Análise de Requisitos, foram levantados cenários de tarefa e o modelo organizacional da tarefa do usuário e estes podem ter necessidade de uma reengenharia. A aplicação não deve simplesmente traduzir o processo atual em uma interface mas não necessariamente precisa forçar os usuários a aprender uma forma totalmente diferente de fazer as tarefas atuais. O objetivo é buscar formas de reduzir e simplificar o trabalho, atingindo os objetivos de negócio e de usabilidade definidos.

Se ocorrer reengenharia, o novo modelo organizacional e de sequência de tarefas deve ser refinado e validado com usuários reais e representativos de cada tipo principal (atores) e posteriormente documentado. Tal documentação será utilizada no projeto do Modelo Conceitual (próxima tarefa a ser descrita nesse trabalho) e no *Design* Detalhado de Interface com o Usuário (DDIU).

#### **4.2.2 Projeto do Modelo Conceitual**

Um modelo conceitual é uma *“descrição do sistema proposto (...) a respeito do que ele deve fazer, de como deve se comportar e com que deve se parecer (...) que seria compreensível pelos usuários da maneira pretendida”* (PREECE, ROGERS e SHARP, 2005). Para isso deve-se saber quais tarefas o usuário deverá desempenhar e considerar qual seria o melhor modo de interação para dar suporte à



execução dessa tarefa. Uma forma de categorizar os tipos de modelos conceituais seria em modelos baseados em atividades e baseados em objetos.

Os modelos baseados em atividades, conforme descrito previamente na seção “Interação” do capítulo dois do presente trabalho, consideram os tipos mais comuns de atividades em que os usuários estarão envolvidos (instrução, conversação, manipulação e navegação, exploração e pesquisa) e os respectivos modelos de interação adequados a tais atividades.

Já os modelos baseados em objetos focam a maneira como certo objeto é utilizado em determinado contexto, geralmente baseados em uma analogia com algo do mundo físico, como por exemplo os ambientes operacionais (áreas de trabalho do Windows e Mac) e portais *web*, que oferecem ao usuário um ambiente familiar ao iniciarem a aplicação.

Outra maneira de se descrever modelos conceituais é por meio de metáforas de interface, que podem ser baseadas em atividades, objetos ou ambas. Um exemplo é a denominação conhecida por “mecanismo de busca”, que remete tanto a uma comparação com um objeto físico (motor, mecanismo), quanto a um conjunto de atividades rotineiras (busca de informações em diversos arquivos e locais).

Para a elaboração do projeto do Modelo Conceitual (MC) não há uma sequência obrigatória de passos, mas as atividades descritas a seguir devem ser consideradas (MAYHEW, 1999).

Primeiramente deve-se definir se o MC será orientado ao produto (quando há produtos claros e identificados a serem criados, denominados e salvos pelo usuário) ou ao processo. Todos os produtos ou processos devem estar claramente identificados. Em um modelo orientado a produtos, isso significa identificar produtos e ferramentas (por exemplo, em um sistema de inventário, o produto pode ser o formulário de aquisição e a ferramenta a uma lista de contato dos vendedores) enquanto em um modelo orientado a processo o detalhamento é obtido por meio do modelo organizacional ajustado após a reengenharia.

Em seguida, devem ser definidas as regras de apresentação para os produtos ou processos.

Em um modelo orientado a produto, com uma interface gráfica de usuário (GUI), podemos normalmente classificar o que é apresentado ao usuário em produtos (primários, aqueles que são os pontos principais do produto e secundários, usados como ferramentas para gerar e modificar os primários), ferramentas (objetos do sistema oferecidos para que o usuário tome ações) e ações (desempenhadas pelo usuário para gerar e modificar os produtos). Uma vez realizada essa classificação, há uma série de diretrizes básicas que devem ser consideradas para a construção de uma boa interface. Por exemplo, produtos primários devem ser exibidos no primeiro nível de interação, agilizando desta forma a ação e reduzindo a ocorrência de efeitos indesejáveis como confusão ao entender os primeiros passos a serem feitos, as principais atividades, a sequência de tarefas e uma navegação entediante. Isso também simplifica a interação, uma vez que o usuário não precisa deixar o produto principal para realizar todas as principais ações e tarefas. Já produtos secundários devem ser exibidos em listas de diálogo ou em menus posteriores, deixando a navegação menos carregada e focando a atenção do usuário nas tarefas-chave.

Já em um modelo orientado a processo essa priorização seguirá a hierarquia de processos definida no Modelo Organizacional de Tarefa ajustado após a reengenharia, com os produtos primários nos primeiros níveis e secundários nos níveis subseqüentes.

Outra atividade importante é definir regras para as janelas da interface, de forma a facilitar a orientação para um comportamento adequado do usuário. Neste caso, por exemplo, produtos primários podem ser representados por ícones e a interface deve permitir com que sejam “minimizadas” e que se abra outra janela em paralelo. Já os produtos secundários podem ser representados por caixas de diálogo e, neste caso, não pode ser minimizados, ocultados ou fechados até que a ação seja concluída. Isso faz com que o usuário não fique perdido em uma série de caixas de diálogo e perca o rastreamento e sequência de sua operação (MAYHEW, 1999).

Shneiderman e Plaisant (2010) apontam a importância de uma boa otimização das janelas para buscar ganho de produtividade, redução da confusão, distração e erros por parte do usuário e exemplifica como alguns usuários precisam acessar informações de diversas fontes: em uma típica operação, um agente de

viagens pode precisar alternar entre várias janelas para concluir sua tarefa, como a janela do email de solicitação do seu cliente, itinerário proposto em um mapa, calendário da viagem, reserva de voo, reserva de hotéis e o cálculo de toda a proposta. Uma técnica que pode ser utilizada em casos como esse é a de janelas múltiplas coordenadas (*coordinated windows*) com a qual as diferentes janelas aparecem (conforme a ação disparada pelo usuário), mudam seus conteúdos (por meio de preenchimento automático a partir das informações fornecidas pelo usuário entre uma tela e outra) e se fecham (tão logo a operação referente a esta seja concluída) de forma automática, ou seja, sem comandos adicionais por parte do usuário.

Para a organização de janelas, podem ser usadas técnicas como as de “salas” de janelas (*room approach*) onde conjuntos de telas são agrupadas conforme finalidades comuns, facilitando o acesso e a orientação do usuário na interface. Outra técnica é a de salvar estados de janelas: assim como se salva, por exemplo, um documento em um processador de texto, permitindo sua recuperação da mesma forma como foi visualizado no último acesso pelo usuário, o usuário “salva” a configuração de uma tela (o que faz com que a interface crie um ícone correspondente dentro da aplicação) permitindo que ele retorne àquele estado da informação posteriormente (MAYHEW, 1999).

Uma vez definidas as regras para as janelas, passa-se à identificação das principais telas e exibições, especificando nesse momento apenas o conteúdo dessas principais telas e menus.

Em seguida deve-se definir e projetar os principais caminhos de navegação, identificando os diferentes caminhos que o usuário pode seguir e as interações em cada um dos caminhos, a fim de projetar a estrutura de menus e caixas de diálogo apropriadamente e respeitando as hierarquias da aplicação.

Mayhew (1999) salienta que, como ainda deverão ocorrer mudanças no projeto, o ideal nesta fase do projeto é documentar os caminhos e menus “à mão”, ou seja, em papel (sem a necessidade ou formalização de ferramentas para isso) para que não haja esforço de retrabalho posterior desnecessário para a atualização de documentação.

Alternativas surgem observando-se outros *designs* semelhantes e o processo de inspiração e criatividade pode ser melhorado aproveitando-se a própria experiência do *designer* e olhando-se para outras idéias e soluções. A escolha entre as alternativas irá depender de uma observação da interação e da experiência dos usuários e demais *stakeholders* com estas alternativas, suas preferências e sugestões de melhoria, bem como no atendimento aos critérios de qualidade definidos (PREECE, ROGERS e SHARP, 2005).

Uma outra forma de obtenção do MC é por meio do *design* participativo. Essa técnica faz com que os usuários estejam ativamente envolvidos no desenvolvimento do *design*. Para isso, pode-se utilizar ferramentas de *software* específicas que permitam a interação com o usuário durante o desenvolvimento da interface. Desta forma, a ferramenta alerta quando princípios gerais (*guidelines*) ou específicos da aplicação (nomenclatura, menus, navegação, etc) estão sendo violados, solicita uma justificativa documentada caso o princípio realmente não for seguido e transmite essa informação (redesenho e justificativa) ao projetista para que este tome as devidas providências. No entanto, há também técnicas para *design* participativo baseadas em papel como o uso de maquetes, PICTIVE (*Plastic Interface for Collaborative Technology Initiatives through Video Exploration*) ou CARD (*Collaborative Analysis of Requirements and Design*).

Segundo Shneiderman e Plaisant (2010), o *design* participativo apresenta vantagens, como a criação de um maior potencial de aceitação do *design* pelo usuário já que o mesmo possui oportunidade de influenciar as decisões de *design*. Porém, há também desvantagens, como as possibilidades de tornar a implementação mais cara e longa, de construir antagonismo com os usuários que não foram envolvidos ou cujas sugestões foram rejeitadas e de forçar o *designer* a comprometer o seu *design* para satisfazer participantes incompetentes.

Outra forma de se obter o MC é por meio de paradigmas de Interação. Trata-se de uma outra fonte de inspiração para instruir o *design* de um modelo conceitual, utilizando-se uma filosofia ou maneira particular de pensar o *design* (PREECE, ROGERS e SHARP, 2005). Por muitos anos, o paradigma que prevaleceu foi o desenvolvimento de aplicações para *desktops*, em que predominava um projeto de *software* utilizando uma interface GUI ou WIMP (*Windows, Icons, Mouse e Pull-down*

*menus* ou *Pointers*). Com o advento de novas tecnologias sem fio, móveis, portáteis e cada vez mais interativas, outros paradigmas alternativos foram propostos por pesquisadores, dentre os quais podemos citar a computação ubíqua ou *ubicmp* (tecnologia inserida no ambiente), computação pervasiva (integração total de tecnologias), computação vestível (*wearables*), realidade aumentada e 3D (PREECE, ROGERS e SHARP, 2005).

### 4.2.3 Elaboração de maquetes do Modelo Conceitual

O primeiro passo é selecionar as funcionalidades que realmente necessitam de maquetes como, por exemplo, aquelas que todos os usuários utilizarão, as novas funcionalidades, aquelas com alta visibilidade, antigas funcionalidades que foram atualizadas, funcionalidades que envolvam aspectos de segurança e confiabilidade e aquelas de missão crítica e/ou relacionadas a esforços de marketing.

Para cada funcionalidade selecionada, deve-se esboçar o seu projeto de interface (sendo que detalhes de telas ainda não são o foco nesse momento) e em seguida construir as maquetes, baseadas em papel ou em protótipo. No caso de papel, pode se usar diferentes pedaços de papel para cada uma das telas, janelas, menus e caixas de diálogo.

Vale aqui apenas uma consideração sobre nomenclatura utilizada para “maquetes”. O que Mayhew (1999, 2008) define como maquete é denominado para alguns outros autores de forma mais genérica como protótipos. Preece, Rogers e Sharp (2005) definem protótipo como “*uma representação limitada de um design que permite aos usuários interagir com ele e explorar a sua conveniência*” e cita ainda os seguintes exemplos que remetem a essa definição abrangendo maquetes e protótipos:

*“Quando você ouve o termo protótipo, pode imaginar algo como um modelo em escala menor de um prédio ou de uma ponte, ou talvez uma parte de um software ainda com muitas falhas. No entanto, um protótipo pode ser também um esboço de papel de uma tela ou conjunto de telas, uma “fotografia” eletrônica, uma simulação em vídeo de uma tarefa, uma maquete tridimensional, de papel ou cartolina, de uma estação de trabalho completa, ou uma simples “pilha” de telas vinculadas por hyperlinks, entre outros”.*

A única observação é que ao utilizarmos a nomenclatura sugerida por Mayhew (1999, 2008) as atividades estão descritas em pontos diferentes do ciclo de vida, sendo as maquetes abordadas no primeiro grupo de tarefas desta fase (MC) e os protótipos no segundo grupo (PDT).

#### **4.2.4 Avaliação Iterativa do Modelo Conceitual**

Dada a importância e abrangência deste tópico em específico, esta seção está dividida em cinco subseções: passos para a avaliação iterativa, escolha do método de avaliação da interface, método analítico, método empírico e, por fim, outros métodos e abordagens de avaliação iterativa.

##### **4.2.4.1 Passos para a Avaliação Iterativa**

No processo de concepção de interfaces, as avaliações têm um papel fundamental e devem ser executadas durante todo o ciclo de desenvolvimento, a fim de que seus resultados sejam utilizados para a melhoria gradual da interface. Isso significa que as avaliações não constituem uma fase única no desenvolvimento e muito menos como uma atividade a ser executada apenas no final do processo (ROCHA e BARANAUSKAS, 2003).

Portanto, as técnicas mencionadas nesta seção de “Avaliação Iterativa do Modelo Conceitual” devem ser aplicadas não apenas nessa atividade como também nas outras duas atividades de avaliação iterativa, ou seja, na avaliação iterativa dos Padrões de *Design* de Telas (PDT) e do *Design* Detalhado da Interface com o Usuário (DDIU).

Os passos de uma avaliação variam de acordo com o método de avaliação definido para o projeto. Desta forma, a primeira decisão a ser tomada é qual método deve ser utilizado para a avaliação da usabilidade da interface. Existem diversos métodos que variam entre si por vários aspectos como, por exemplo, a participação ou não do usuário nos testes, a infra-estrutura e tempo dedicados à avaliação, a etapa do ciclo de vida em que pode ser executado, a disponibilidade de um especialista com conhecimento específico em determinadas técnicas, entre outros.

A próxima seção se destina a explicar de forma detalhada os diferentes métodos e suas características.

#### **4.2.4.2 Escolha do Método de Avaliação da Interface**

Conforme comentado na seção anterior, um primeiro passo importante é a escolha do método de avaliação de interface a ser utilizado. Preece et al. (1994) destacam que os métodos diferem entre si em vários aspectos. É preciso entender as diferentes características de cada método para se definir qual deles é o mais apropriado para se avaliar a interface de um *software* em um determinado contexto. Os métodos podem ser analíticos, onde apenas os especialistas ou avaliadores examinam a interface, ou empíricos, onde os usuários são envolvidos e realizam testes com a interface. As próximas seções do presente trabalho tratarão das abordagens desses dois métodos. Prates e Barbosa (2003) consideram que as principais diferenças entre os métodos se encontram nas seguintes dimensões: etapa do ciclo em que as avaliações devem ou podem ser aplicadas, técnica de coleta de dados, tipos de dados coletados e tipo de análise feita. Essas dimensões serão exploradas nos parágrafos seguintes.

Em relação à etapa do ciclo em que as avaliações devem ou podem ser aplicadas, podemos classificar as avaliações em dois tipos: formativas ou somativas (PREECE, ROGERS E SHARP, 2005). As avaliações formativas (ou construtivas) são realizadas durante o processo de *design*. Pode utilizar-se de avaliações rápidas (reuniões informais entre usuários e desenvolvedores) cenários, *storyboards*, modelo conceitual ou protótipos. Problemas de interação são identificados e consertados antes de a aplicação ser terminada e liberada para uso. Com isso, a qualidade é maior e o custo das alterações é menor. Já as avaliações somativas (ou conclusivas) são realizadas em produtos já terminados e buscam verificar a existência de aspectos como a sua conformidade aos padrões estabelecidos.

Quanto à técnica de coleta de dados, esta também pode ser feita em diferentes etapas do ciclo. Pode ser utilizada a coleta de opinião de usuários (por meio de questionários e entrevistas) ou a observação direta de usuários, por meio de anotações do observador, gravação de vídeo, áudio ou da interação em laboratórios.

É importante lembrar que as práticas recomendadas para questionários, entrevistas, etnografia e observação participativa (mencionadas anteriormente na fase de Análise de Requisitos) também se aplicam a esse passo de avaliação da interação do usuário com o sistema, conforme Mayhew (1999).

Quanto ao uso de questionários do tipo “*user survey*” como forma de avaliação da usabilidade, existem modelos prontos e padronizados, como, por exemplo: *QUIS – Questionnaire for User Interface Satisfaction*, *PUEU – Perceived Usefulness and Ease Use*, *NAU – Nielsen’s Attributes of Usability*, *NHE – Nielsen’s Heuristic Evaluation*, *CSUQ – Computer System Usability Questionnaire*, *ASQ – After-Scenario Questionnaire*, *PHUE – Practical Heuristics for Usability Evaluation* e *PUTQ – Purdue Usability Testing Questionnaire* (AVELINO, 2005).

O recurso da filmagem é interessante por permitir de forma simples não só a captura como o registro claro (e a possibilidade de análise posterior mais aprofundada) de aspectos como comunicação não-verbal e paralinguística dos usuários durante os testes. No caso de laboratórios pode-se coletar dados qualitativos ou quantitativos sobre o uso, como o tipo e frequência das dificuldades enfrentadas pelos usuários, caminhos preferenciais, o tempo gasto para executar a tarefa ou o número de erros cometidos. Outra técnica é a de registro de uso (observação indireta) com a qual podem ser usados diários (usuários registram suas ações), *logs* que armazenam as ações executadas pelos usuários ou gravação da interação do usuário com o sistema. O registro (*log*) de dados contínuo do desempenho do usuário pode auxiliar, por exemplo, a otimizar desempenho, reduzir custos, servir de guia/alerta para necessidades de aquisição de *hardware*, melhorias em treinamento e planos de expansão do sistema. A arquitetura do *software* deve tornar mais fácil para os gestores do sistema coletar dados sobre padrões de uso do sistema, velocidade de desempenho do usuário, taxa de erros, frequência de solicitação de assistência on-line, entre outros. Em alguns casos, assistência *online*, por telefone, email e caixa de sugestões *online* podem auxiliar, já que alguns usuários se sentem mais seguros simplesmente por saber que há um atendimento humano disponível. Em alguns sistemas o suporte pode monitorar o computador do usuário e ver exatamente o que ele vê (SHNEIDERMAN e PLAISANT, 2010).



Já a técnica de coleta da opinião de especialistas é mais indicada quando os usuários não estão disponíveis ou o seu envolvimento implica em um custo elevado. Neste caso, especialistas examinam a interface buscando identificar possíveis dificuldades que os usuários podem vir a ter ao utilizar o *software*.

Shneiderman e Plaisant (2010) apontam também o uso de *wikis* e *newsgroups* para a coleta de informações dos usuários a respeito da interface. Tais ferramentas permitem postagens de mensagens e perguntas e contam ainda com moderadores. Ao usar uma abordagem de redes sociais, comentários e sugestões de melhoria devem ser incentivados pela gerência.

Em relação aos tipos de dados coletados, estes podem ser quantitativos (número de erros ocorridos ou o tempo gasto para concluir uma tarefa, por exemplo) ou qualitativos (lista de problemas que os usuários tiveram na interação, sugestões de melhoria, por exemplo).

A última dimensão destacada por Prates e Barbosa (2003) é em relação ao tipo de análise feita. Há três tipos de análise: preditiva, interpretativa e experimental.

Na análise preditiva os avaliadores verificam os dados coletados de especialistas e tentam prever que tipo de problemas os usuários enfrentarão (por meio de inspeção da interface ou técnicas de modelagem). Os modelos preditivos permitem que se obtenham várias medidas de desempenho dos usuários sem realmente ter de testá-los. A técnica de modelagem preditiva mais conhecida é a GOMS (*Goals, Operators, Methods, Selection Rules*). A técnica consiste na decomposição de metas em ações e estas em métodos. Os usuários aplicam seleção de regras para escolher entre os métodos a fim de alcançar as metas (SHNEIDERMAN e PLAISANT, 2010). Desta técnica derivam os modelos GOMS e o *keystroke level*, que buscam modelar o conhecimento e os processos cognitivos envolvidos quando usuários interagem com sistemas. O *keystroke level* fornece previsões numéricas reais do desempenho do usuário.

Na análise interpretativa os avaliadores analisam os dados coletados a partir da interação do usuário com o sistema e procuram explicar os fenômenos que ocorreram durante esta interação (normalmente com dados coletados em ambientes sem interferência de observadores).

Por fim, na análise experimental os dados são coletados em ambientes controlados, como laboratórios (se diferencia da interpretativa porque as variáveis manipuladas são conhecidas).

Há ainda abordagens baseadas em teoria explanatória, que prevê a descrição da sequência de eventos, identificação de causas e efeitos e de eventuais intervenções necessárias. Norman (1988) oferece uma abordagem baseada em um modelo cíclico composto de sete estágios de ação: formando o objetivo, formando a intenção, especificando a ação, executando a ação, percebendo o estado do sistema, interpretando o estado do sistema e avaliando o resultado. O objetivo é a identificação de um golfo de execução (incompatibilidade entre as intenções do usuário e as ações permitidas) e de um golfo de avaliação (incompatibilidade entre a representação do sistema e as expectativas dos usuários)

Uma vez entendidas as diferentes características de cada método de avaliação, vamos detalhar a seguir os dois métodos com maior foco para o presente trabalho: o método analítico e o empírico.

#### **4.2.4.3 Método Analítico**

Segundo Nielsen e Mack (1994) é o método no qual avaliadores inspecionam ou examinam aspectos de uma interface relacionados a usabilidade. Existem três tipos de conhecimento envolvidos em uma avaliação analítica: conhecimento sobre o domínio, conhecimento e experiência no projeto e avaliação de interfaces de usuário e, por último, a experiência em se realizar um tipo específico de avaliação (LYNCH e PALMITER, 2002). Existem os seguintes tipos de avaliação analítica: avaliação heurística, percurso cognitivo (*cognitive walkthrough*), percurso pluralista ou pluralístico, conformidade com diretrizes e padrões e, por fim, inspeções de consistência, padrões e formais.

A seguir serão detalhados os tipos de avaliação heurística e percurso cognitivo do método analítico e, em seguida, o método empírico. Os demais tipos serão elencados de forma mais breve na seção “Outros métodos e abordagens de avaliação”. Não são considerados aqui os métodos de avaliação de comunicabilidade, por não fazerem parte do escopo deste trabalho.

#### 4.2.4.3.1 Avaliação heurística

Conforme Nielsen e Mack (1994), a avaliação heurística é um método de inspeção que visa identificar problemas de usabilidade a partir de um conjunto de princípios de usabilidade, também chamados de heurísticas ou diretrizes (guidelines), baseadas em melhores práticas definidas por profissionais experientes e especialistas em IHC. Este método não envolve usuários e deve ser realizado por avaliadores especialistas. Nielsen (2000) sugere que, a fim de se alcançar uma melhor relação custo-benefício, o ideal é que a equipe seja formada por um número de três a cinco avaliadores, mas o número pode ser maior no caso de sistemas críticos, por exemplo. É um método rápido e de menor custo que a maior parte dos métodos de avaliação amplamente difundidos.

Uma avaliação heurística é normalmente composta de três estágios. O primeiro estágio é composto por uma sessão breve e preliminar, na qual se diz aos especialistas o que fazer. Em seguida, há o período de avaliação, no qual cada especialista inspeciona independentemente o produto, utilizando as heurísticas como guia, e registra os problemas encontrados. Ao final da avaliação, há a sessão de resultados, na qual os especialistas se reúnem para discutir as descobertas, priorizar os problemas e sugerir soluções. Em geral, um avaliador demora em média de uma a duas horas para realizar a avaliação. Sessões de avaliação muito longas devem ser divididas em sessões menores, com foco em determinadas partes do sistema (NIELSEN e MACK, 1994).

As heurísticas não devem ser observadas apenas nas etapas de testes. O processo de desenvolvimento do *software* deve contemplar atividades que busquem assegurar a observação e o atendimento a esses elementos durante todo o ciclo de vida do desenvolvimento do *software*, desde a sua concepção e planejamento do projeto até o seu encerramento. Prates e Barbosa (2003) reforçam que é possível realizar uma avaliação heurística nas etapas iniciais do ciclo de projeto e desenvolvimento. Esta avaliação pode ser feita sobre interfaces que ainda não tenham sido implementadas, representadas em papel.

A avaliação com foco em usabilidade deve rastrear o processo de desenvolvimento da interface identificando em quais pontos há atividades que

busquem o atendimento às heurísticas, bem como levantar quais as possíveis evidências a serem geradas, coletadas e analisadas. Além disso, esses pontos devem fazer parte do dimensionamento de esforço para a construção da interface e alguns deles devem ser monitorados por meio de indicadores de desempenho. O processo de desenvolvimento do *software* deve contemplar esses itens no seu processo de estimativa de esforço, por exemplo.

Nielsen (1993) propôs um conjunto básico de dez heurísticas. Cada elemento de interface (ou conjunto de elementos) deve ser analisado para verificar sua conformidade com cada uma das seguintes heurísticas:

- Visibilidade do estado do sistema
- Compatibilidade entre o sistema e o mundo real
- Controle e liberdade do usuário
- Consistência e padronização
- Prevenção de erros
- Reconhecimento ao invés de lembrança
- Flexibilidade e eficiência de uso
- Estética e *design* minimalista
- Ajudar ao usuário para reconhecer, diagnosticar e se recuperar de erros
- Ajuda e documentação

A seguir estarão detalhadas cada uma das heurísticas pela abordagem de Nielsen (1993), bem como as respectivas corroborações de outros autores em algumas das heurísticas.

A visibilidade do estado do sistema consiste em manter o usuário informado sobre o que está acontecendo, por meio de *feedback* adequado e no tempo certo.

A compatibilidade entre o sistema e o mundo real busca a utilização de conceitos, vocabulário e processos familiares e coerentes ao modelo mental dos usuários. Como modelo mental do usuário podemos entender o conjunto dos dois conhecimentos por parte do usuário sobre a interface: como utilizar a interface e como ela funciona. Quanto mais ele sabe como usá-la e como ela funciona, mais desenvolvido será o modelo mental do usuário (PREECE, ROGERS e SHARP,

2005) e, conseqüentemente, melhor este resolverá os obstáculos ao utilizar o sistema.

Para o controle e liberdade do usuário deve-se fornecer alternativas e “saídas de emergência”, como as possibilidades de desfazer e refazer ações e voltar ao ponto anterior. A importância de tal heurística pode ser corroborada pela regra de suportar locus interno de controle, conforme mencionado previamente na seção “Definição dos Princípios Gerais de Projeto” deste trabalho, que trata das Oito Regras de Ouro de Shneiderman e Plaisant (2010). O objetivo é garantir o controle humano à medida que se aumenta a automação. Há atividades que geralmente são melhor desempenhadas por máquinas como armazenamento, monitoramento, recuperação e processamento de grandes quantidades de informação, execução de várias atividades simultâneas e a preservação de um mesmo nível de desempenho após longos períodos de tempo. Entretanto, há atividades que normalmente são melhor desempenhadas por humanos como o reconhecimento de padrões, adaptação de decisões conforme a situação, seleção de alternativas quando a abordagem original falha, antecipação de falhas, aplicação de princípios para resolver problemas e desenvolvimento de novas soluções. O sistema deve, portanto, preservar ao usuário o poder de tomada de decisão e definição de prioridades em situações críticas e imprevistos (SHNEIDERMAN e PLAISANT, 2010).

Para consistência e padronização, deve-se considerar que palavras, situações e ações semelhantes devem significar conceitos ou operações semelhantes durante todo o percurso pelo sistema. O sistema deve obedecer eventuais padrões e convenções do ambiente, plataforma ou ferramenta em uso.

A prevenção de erros visa evitar que o erro aconteça, informando o usuário sobre as conseqüências de suas ações ou, se possível, impedindo ações que levariam a situações de erro. Conforme Nielsen (1993), *“Ainda melhor que uma boa mensagem de erro é um design cuidadoso que possa prevenir esses erros”*. Norman (1983) propõe para a prevenção de erros as seguintes diretrizes: ações corretas e sequências completas. Para as ações corretas (por exemplo, o motor do avião não pode ir para o reverso sem o trem de pouso baixado) deve-se buscar desabilitar opções que possam causar erros, priorizar a seleção ao invés de digitação e possibilitar preenchimento automático sempre que possível. Para as sequências

completas a partir de uma única ação do usuário (por exemplo, quando o trem de pouso do avião baixa, centenas de passos são disparados automaticamente) buscase o uso de comandos únicos e macros.

Para o reconhecimento ao invés vez de lembrança deve-se tornar objetos, ações e opções visíveis e compreensíveis, bem como orientar as ações do usuário para que ele não tenha que acionar constantemente sua memória para tomar uma ação.

A flexibilidade e eficiência de uso busca proporcionar uma interface fácil para usuários leigos mas flexível o bastante para se tornar ágil aos usuários avançados, com o uso de aceleradores e caminhos alternativos para uma mesma tarefa. Deve-se permitir que os usuários customizem ações freqüentes (uso de teclas de atalhos, máscaras e navegação com “tab” em formulários, dentre outras).

Uma estética e *design* minimalista da interface evita porções de informação irrelevantes. Busca-se exibir apenas os textos e *design* que o usuário precisa ter acesso. Cada unidade extra de informação em um diálogo compete com as unidades de informação relevantes e reduz sua visibilidade relativa.

Ajuda aos usuários para reconhecerem, diagnosticarem e se recuperarem de erros deve ser considerada. O sistema deve emitir mensagens de erro em linguagem simples, sem códigos, indicando precisamente o problema e sugerindo ao usuário o que ele precisa fazer como possível solução de contorno. Shneiderman e Plaisant (2010) ressaltam nesse caso a importância de tornar as mensagens de erro específicas (e não genéricas ou obscuras), em tom positivo, cordial (e não hostil ou violento), claro e construtivo (com foco na resolução do problema). O estilo deve ser centrado no usuário e na orientação clara e compreensível de sua ação após o erro. Outros cuidados devem ser considerados como a padronização e consistência nas abreviações utilizadas e no melhor posicionamento da mensagem na tela e a atenção aos alertas sonoros (deixá-los sob controle do usuário) pois podem causar embaraços.

Ajuda e documentação deve estar acessível, com mecanismo de busca, focados no domínio e na tarefa do usuário e apresentar passos concretos para que ele possa atingir seus objetivos. Sempre que possível a interface deve possibilitar a

coleta de dados que permitam a identificação de quais pontos do sistema necessitam de maior ajuda ao usuário.

Shneiderman e Plaisant (2010) apresentam uma série de vantagens no uso de documentação ou ajuda (*help*) *online* como a facilidade de busca, localização direta e navegação, a possibilidade de se usar recursos como tutoriais, demonstrações, animações, recursos combinados de texto, áudio e vídeo, *links* e *hiperlinks* (inclusive externos à interface) para se navegar entre a documentação, chats e mensagens instantâneas, a vantagem econômica (custo menor do que papel para se duplicar e distribuir), a rapidez na atualização e disponibilização de novas versões, entre outras. Contudo, há desvantagens que devem ser consideradas e tratadas sempre que possível na construção da interface como a fadiga causada pelo uso extenso, a maior frequência de rolagem e troca de páginas, dificuldade de leitura ou até mesmo incompatibilidade de uso em dispositivos com displays pequenos (como celulares), entre outros. Um bom sistema de ajuda e documentação, no entanto, não deve ser usado para compensar um mau *design* da interface e uma usabilidade deficiente.

Para encerrar a abordagem de avaliação heurística acima descrita (NIELSEN, 1993) é importante lembrar que para cada problema encontrado, ou seja, para cada heurística violada, deve-se definir ainda a localização do problema, ou seja, onde ele ocorre na interface, e sua gravidade.

Para a gravidade, pode-se atribuir um grau de severidade a cada um dos problemas, conforme a Tabela 5, de forma a priorizar seu tratamento e avaliar sugestões de ajustes no *design* na versão atual ou posterior.

**Tabela 5** - Graus de severidade dos problemas encontrados

| Severidade | Descrição                                                                      |
|------------|--------------------------------------------------------------------------------|
| 1          | Não é um problema de usabilidade                                               |
| 2          | Problema cosmético - precisa ser corrigido apenas se sobrar tempo no projeto   |
| 3          | Problema de usabilidade menor - prioridade baixa para correção                 |
| 4          | Problema de usabilidade grave - prioridade alta para correção                  |
| 5          | Catástrofe de usabilidade - correção obrigatória antes da liberação do produto |

**Fonte:** Adaptado de Nielsen e Mack (1994, p. 103)

Existem ainda estudos focados para heurísticas em ambientes específicos como sistemas *web* e comunidades *online*. Preece (2000) compilou, por exemplo, a partir de diversas fontes uma lista de recomendações para avaliação de websites, baseada em heurísticas como evitar páginas órfãs (não vinculadas à homepage), páginas muito longas ou com muito espaço em branco (que forcem o uso da barra de rolagem), menus hierárquicos estreitos e profundos, cores não-padronizadas para os *links*, URLs complexas e downloads muito demorados que aborreçam os usuários. Deve-se buscar fornecer suporte à navegação, como um bom mapa do site e uma navegação confortável, com *design* da informação agradável.

Outras diretrizes como estas estão declaradas na seção “Definição dos Princípios Gerais de Projeto” do presente trabalho ou nas seções específicas pelas atividades correlacionadas a cada um dos itens.

#### **4.2.4.3.2 Percurso Cognitivo**

*“Os percursos cognitivos envolvem simular um processo de solução de problemas a cada passo do diálogo homem-computador, verificando se é possível assumir que os objetivos do usuário e sua memória para as ações conduzam a uma próxima ação correta”* (NIELSEN e MACK, 1994). É outro tipo de avaliação analítica, que avalia uma proposta de projeto de IHC no contexto de tarefas específicas do usuário (WHARTON et al. 1994). Ele visa avaliar principalmente a facilidade de aprendizado do sistema, enfoque motivado por observações que os usuários aprendem mediante exploração. Não envolve usuários.

Antes de realizar a avaliação, deve-se passar por uma fase de preparação, na qual se definem as hipóteses sobre os usuários e sobre o conhecimento que eles têm a respeito da tarefa e da interface proposta. Também são definidos os cenários de tarefas (construídos a partir de uma seleção de tarefas importantes e de tarefas freqüentes), a seqüência “correta” de ações para concluir cada tarefa (tal como definida pelo projetista) e a proposta de *design* em papel ou protótipo (ilustrando cada passo e indicando o estado da interface antes/depois de cada um).

O procedimento de execução da avaliação compreende alguns passos. Primeiramente o projetista apresenta uma proposta de *design*. Em seguida os



avaliadores constroem histórias plausíveis sobre a interação de um usuário típico com a interface, com base nos cenários de tarefas selecionados. Os avaliadores, então, simulam a execução da tarefa, efetuando uma série de perguntas sobre cada passo. Ao final, os avaliadores anotam pontos-chave, como o que o usuário precisa saber antes de realizar a tarefa e o que o usuário deve aprender ao realizar a tarefa. Segundo Prates e Barbosa (2003), a cada um desses quatro passos acima, os avaliadores devem se fazer uma série de perguntas, buscando descobrir problemas em potencial, ou seja, que poderiam ocorrer durante a interação de usuários reais com o produto final implementado conforme aquela proposta de *design*.

#### **4.2.4.4 Método Empírico**

Esse método envolve usuários para a coleta de dados, que são posteriormente analisados pelo especialista para identificar os problemas da interface. Normalmente, os testes são realizados em ambientes controlados, como laboratórios, e requerem algumas práticas, por parte do responsável pela avaliação, para o seu correto planejamento e execução.

A seguir serão apresentadas algumas dessas práticas referentes ao planejamento e em seguida práticas relacionadas à execução dos testes.

O planejamento consiste de uma série de atividades que buscam garantir que o avaliador tenha controle sobre as condições de teste.

Critérios relevantes e pontos críticos devem ser determinados. Os critérios normalmente envolvem aspectos da qualidade de uso que se pretende que o sistema possua, como por exemplo priorizar o desempenho (se desejar que o usuário execute a tarefa em um determinado período de tempo) ou que os usuários estejam simplesmente satisfeitos com o sistema. Os pontos críticos normalmente estão relacionados com decisões de *design* e tarefas estratégicas para o uso da aplicação de uso muito freqüente (PRATES e BARBOSA, 2003);

Primeiramente, deve-se revisar as condições de testes, certificando-se de que estas são as mesmas para todos os participantes.

Em seguida, deve-se selecionar as tarefas que terão foco no teste. O critério pode ser o mesmo já mencionado anteriormente para selecionar funcionalidades, na seção “Elaboração de maquetes do Modelo Conceitual”. Estas devem ser típicas e tão realistas quanto se possa prever sobre o uso a ser feito do sistema. O banco de dados deve estar corretamente populado, ou seja, com dados apropriados e próximos da realidade, para a execução das tarefas. O avaliador deve definir também as medidas a serem observadas para cada aspecto que se deseja apreciar. Por exemplo, para se avaliar o critério de produtividade, possivelmente será desejável medir o tempo gasto no desempenho de cada tarefa e o número de erros cometidos por tarefa, enquanto para se avaliar a usabilidade pode-se medir o percentual de usuários que conseguirem se recuperar de um erro, que se disserem satisfeitos com a aplicação ou que prefiram um outro sistema.

O próximo passo é a elaboração de um plano com os eventos de teste. Os cenários gerados no passo de análise de contexto das tarefas podem ser usados e adaptados aos testes, com a ajuda de um usuário final experiente, de modo a criar descrições de tarefas que possam ser facilmente lidas e entendidas pelos usuários de teste e que sejam o mais realista possível. Deve-se planejar a sequência exata de eventos de teste.

O passo seguinte diz respeito à seleção dos usuários que realizarão os testes. Uma decisão a ser tomada é se o foco do teste será na facilidade de aprendizado ou de uso. Para isso, as informações obtidas do perfil do usuário e dos objetivos de usabilidade definidos podem auxiliar. Para um teste de facilidade de aprendizado deve-se recrutar novatos e investir apenas no mínimo de instrução aos mesmos. Já para o teste de facilidade de uso, especialistas devem ser recrutados e treinados para a execução dos testes.

Deve haver um cuidado especial ao se definir o perfil dos participantes. O objetivo é ter usuários que representem usuários típicos do sistema, o que normalmente pode ser diagnosticado por meio de um questionário breve. Deve-se buscar identificar o tipo de usuário desejado (uma categoria dos altamente prioritários ou uma amostra de todas as potenciais categorias, por exemplo), listar as características e habilidades requeridas para os usuários e o número necessário de participantes. Conforme Dumas e Redish (1999), tipicamente envolve-se em uma

avaliação de cinco a doze usuários. Nielsen (2000) recomenda que cinco usuários participem da avaliação já que desta forma se encontram aproximadamente 85% dos problemas da aplicação e o benefício dos novos erros encontrados vale o custo do teste executado. Nos casos em que a aplicação se destina a usuários de perfis distintos, o autor recomenda ainda que para cada perfil identificado se faça a avaliação com três usuários, pois muitas vezes usuários de perfis distintos identificam os mesmos problemas e ressalta que, embora um teste com quinze usuários permita potencialmente que se encontre todos os problemas de uma aplicação, vale mais a pena fazer três ciclos de testes (incremental) com cinco usuários do que um com quinze pois, com um teste com quinze usuários todos os problemas poderão ser encontrados mas a solução a ser desenvolvida após o teste não será avaliada e pode conter novos problemas (PRATES e BARBOSA, 2003).

Outro ponto importante é avaliar o tipo de motivação do usuário. Ele pode ser motivado por ser um potencial usuário ou por alguma compensação financeira (por exemplo) pela participação no teste e isso deve ser considerado no planejamento do projeto.

Uma prática importante é garantir o atendimento às questões éticas junto aos usuários. Preece, Rogers e Sharp (2005) apontam que deve-se explicar aos participantes os objetivos do estudo sendo feito e exatamente como deverá ser a participação deles. Deve-se deixar claro o processo de teste, o seu tempo aproximado, o tipo de dado que será coletado e ainda como os dados serão analisados. Deve-se garantir as expectativas quanto ao anonimato dos usuários, deixando claro que dados particulares identificados durante o teste não serão divulgados. Sempre que trechos de depoimentos dos usuários são utilizados eles devem ser anônimos e deve-se retirar descrições ou trechos que permitam a identificação do usuário. Nestes casos, deve-se requisitar autorização prévia do usuário e de preferência mostrar-lhe o relato a ser divulgado. O usuário deve consentir por escrito na execução do teste e o documento de consentimento deve especificar as condições acordadas do teste e deve ser assinado tanto pelo participante do teste, quanto pelo avaliador. Além disso, o formulário deve permitir ao usuário acrescentar novas condições ao acordo, caso o deseje. Por fim, os usuários devem entender que a qualquer momento podem interromper o teste, caso desejem.

Ainda em relação ao planejamento e preparação, um ponto importante é providenciar com antecedência todos os materiais necessários e aplicáveis como, por exemplo, instruções aos observadores, instruções aos usuários (agradecimento pela participação, formulário para permissão de gravação em vídeo e explicações gerais), questionário pré-teste (para identificar características, categoria e perfil do usuário), material de treinamento (conforme a decisão entre facilidade de aprendizado ou de uso), plano de testes para os usuários (descrição detalhada das tarefas a serem executadas) e planilha para armazenamento dos dados coletados (*logs*, observações e comentários). Todo o material gerado nos testes deverá ser identificado para cada usuário. Questionários pós-teste (captam as reações subjetivas, impressões e sugestões de melhoria) podem ser preparados, bem como um modelo para resumo de análise dos dados, sendo que este último tem foco maior em identificar eventuais falhas no nível de modelo conceitual (e gerar uma análise de problemas e soluções propostas) do que em refinar as interfaces para atingimento dos objetivos de usabilidade. Quanto às instruções ao observador ou aplicador do teste, pode-se elaborar um roteiro que especifique a sequência correta das suas atividades e seu comportamento esperado durante os testes, como por exemplo, não interferir no teste do usuário, não responder perguntas relacionadas ao uso da aplicação e a permissão para responder perguntas que não estão relacionadas com o que se deseja observar.

Em seguida, o local (ambiente de teste) deve ser planejado, projetado e montado, de forma a retratar o mais fielmente possível o ambiente real do usuário (silêncio ou barulho, móveis, utensílios, etc). No caso de ambientes mais comuns, laboratórios podem ser usados, mas no caso de ambientes de trabalho incomuns (como um posto policial, uma sala de cirurgia, etc) o ideal é que o teste seja feito no próprio ambiente real. Shneiderman e Plaisant (2010) apontam algumas considerações relevantes quanto aos ambientes de testes. Os autores lembram que os testes de usabilidade duram apenas horas ou dias e algumas interfaces precisam de mais tempo para ser exploradas nos detalhes (curva de aprendizado e de desempenho). Além disso, condições de alta pressão e criticidade dificilmente podem ser reproduzidas em laboratórios convencionais. Testes em dispositivos móveis como celulares, *smartphones* e *tablets*, por exemplo devem considerar uma atenção durante a simulação em itens que podem alterar os resultados, como a

disponibilidade de baterias extras e/ou carregadores, a influência da potência/força do sinal no local de teste e até mesmo que os dedos podem bloquear parte do visor ou *display*.

Se a opção for o uso de laboratórios, podem ser usadas duas salas ou apenas uma. No caso de duas salas, uma é destinada à execução do teste e outra à observação. As salas são separadas por um vidro espelhado, de forma que o participante não enxergue quem se encontra do outro lado do vidro, mas os observadores possam ver o participante e suas ações (SHNEIDERMAN e PLAISANT, 2010). A sala de teste costuma ser equipada com um computador e espaço para o participante e um avaliador. A sala de observação costuma ter um monitor que replica o que está sendo visto no monitor do usuário e um outro computador para anotações. Câmeras de vídeo e gravadores podem estar presentes em uma sala ou na outra, ou em ambas dependendo do teste a ser executado. Para as filmagens, duas câmeras são o ideal (uma para capturar a tela do aplicativo e outra voltada para o rosto do usuário). O observador deve estar posicionado de forma a captar ambas imagens. Há ainda dispositivos como o *eye-tracking*, que monitora e registra quais as áreas da tela que são vistas, por quanto tempo e quais são ignoradas (SHNEIDERMAN e PLAISANT, 2010). Várias técnicas podem ser utilizadas para se obter o melhor resultado em laboratórios, como a de pensar em “voz alta” (*think-aloud*) na qual o usuário deve falar em voz alta exatamente o que está pensando, para que o observador possa entender melhor e registrar as dificuldades, dúvidas e outros sentimentos que o usuário está experimentando ao utilizar a interface. Já com o *Retrospective think aloud*, os comentários são feitos e registrados logo após a conclusão de uma ação ou tarefa. Ao contrário do *think aloud*, esta técnica não aumenta o tempo da tarefa pois não há aumento de carga cognitiva durante a execução da ação pelo usuário. Cabe um alerta que se usarmos essa técnica concomitantemente ao *eye-tracking* podemos ter resultados falsos, já que os olhos podem desviar-se ou perder-se enquanto o participante fala (SHNEIDERMAN e PLAISANT, 2010).

Quando não se tem um laboratório disponível pode-se considerar o uso de equipamento móvel (como câmeras de vídeo ou sistemas de registro de interação) em alguma sala disponível. Quando não existir a figura do observador nos testes,

deve-se avaliar outra forma de captar as informações acima (PRATES e BARBOSA, 2003).

Ao final desse planejamento, espera-se garantir que o ambiente seja adequado e os usuários possam agir da forma mais natural possível e semelhante ao ambiente real. A seguir serão descritas algumas práticas coletadas a respeito da execução dos testes.

Durante a execução, o plano de avaliação deve ser seguido. Esse plano deve ser disponibilizado aos responsáveis pela condução dos testes, servindo de guia e atualizado caso necessário. Uma boa prática para a execução dos testes é iniciar com um teste-piloto, que permite avaliar a qualidade do material gerado. Esta é uma forma de depurar e validar os procedimentos e materiais desenvolvidos até então e identificar eventuais ajustes. Nesse caso, os usuários devem representar significativamente a população-foco dos testes. Deve-se observar se os participantes conseguiram entender corretamente todo o material apresentado, se o tempo de execução do teste está dentro do previsto e é viável, se por meio das tarefas propostas se consegue obter as medidas especificadas e avaliar o critério desejado (PRATES e BARBOSA, 2003). Caso tenham sido identificados quaisquer necessidades de ajustes, deve-se revisar os procedimentos e materiais de teste.

Após o piloto e eventuais ajustes, parte-se para o teste definitivo. Os usuários são convocados (com o devido recrutamento e agendamento), o teste é realizado e os dados coletados e sumarizados, conforme o planejamento. Exemplos de formas de sumarizar os dados: número de vezes que cada problema ocorreu, tempo gasto em erros *versus* tempo em trabalho produtivo, número de usuários que experimentaram determinado erro e número total de erros em determinada tarefa (MAYHEW, 1999).

De posse dessa informação, os projetistas da interface devem analisar e interpretar os dados, focando em áreas que apontem para potenciais problemas (alta frequência de erros ou alta média de tempo para executar uma tarefa, por exemplo). Os problemas observados devem ser registrados e classificados. O avaliador primeiramente classifica os problemas pela sua gravidade (NIELSEN, 1998). Um exemplo de classificação dos problemas pode ser em: catastróficos

(impedem que o usuário termine sua tarefa), sérios (atrapalham a execução da sua tarefa) ou cosméticos (atrasam a execução e/ou irritam usuários). Além disso, para cada medida observada, o avaliador verifica se o critério está em um patamar desejável ou não, comparando-o aos objetivos pré-estabelecidos.

Com isso, conclusões podem ser formuladas e mudanças recomendadas. Os problemas encontrados podem ser priorizados conforme seu impacto na usabilidade e no negócio e as respectivas soluções priorizadas considerando seu custo de implementação.

#### **4.2.4.5 Outros métodos e abordagens de avaliação**

Além dos métodos detalhados nas seções anteriores, serão elencados a seguir uma série de métodos alternativos para a condução de avaliações.

A técnica de avaliação remota possui vantagens como eliminação de tempo e custos de deslocamento dos usuários e permitir que a aplicação seja testada em seu ambiente de trabalho natural (MAYHEW, 1999). Dentre os métodos podemos citar: avaliação remota com acompanhamento por meio de *softwares* de emulação, vídeo-conferência, uso de *softwares* de monitoramento com captura de dados e imagens durante o uso da interface pelo usuário (avaliação remota instrumentada), *logs* do uso do sistema ou aplicativos que permitem com que o usuário identifique e comente (com *tags* e descrições) os pontos da interface que apresentaram problemas (avaliação remota semi-instrumentada). A avaliação pode ser síncrona ou assíncrona (SHNEIDERMAN e PLAISANT, 2010).

Outra técnica é a de verificação da conformidade a diretrizes e padrões, que propõe que uma interface seja inspecionada para se verificar o grau de aderência desta com alguma lista de diretrizes (*guidelines*) gerais.

Por meio da técnica de percurso pluralista ou pluralístico, usuários, desenvolvedores e especialistas em usabilidade percorrem uma interface juntos discutindo questões de usabilidade associadas a elementos de diálogo envolvidos nos passos de cenário.

Já a técnica de inspeção de consistência é voltada à verificação de consistência entre uma nova interface de uma família de produtos com as demais interfaces dessa família. Verificam-se terminologia, *fonts*, cores, layout, formatos de entrada e saída de dados, bem como documentação e ajuda *online* (SHNEIDERMAN e PLAISANT, 2010).

Na técnica de inspeção padrão, um especialista em determinado padrão relevante da interface (por exemplo, a plataforma ou sistema operacional utilizado) checa a aderência da interface a esse padrão.

Outra técnica é a inspeção formal de usabilidade, semelhante a uma inspeção de código mas voltada a problemas de usabilidade. Shneiderman e Plaisant (2010) definem como reuniões no estilo de “sala de tribunal”, com um moderador/juiz, buscando pontos fortes e fracos da interface.

A técnica de Metáforas do Pensamento Humano (MOT) foca em como os usuários pensam ao interagir com a interface (SHNEIDERMAN e PLAISANT, 2010), abordando itens como hábito, fluxo de pensamento, conscientização e associações, relação entre enunciados e pensamentos e conhecimento.

Existem ainda outros métodos como a avaliação e testes de comunicabilidade, baseados na engenharia semiótica, que são extremamente úteis. Tais métodos possuem um foco maior na avaliação do produto final ou prototipado, conforme Souza, Prates e Barbosa (1999) e se baseiam em uma interação definida com o usuário.

Outra técnica é a de *Bird’s-eye view* (vista aérea), onde se coloca a impressão de todas as telas no chão ou parede para apurar inconsistências e, no caso do uso de múltiplos desenvolvedores, verificar se um padrão foi seguido (SHNEIDERMAN e PLAISANT, 2010).

Testes chamados de *Can-you-break-this* possuem uma abordagem destrutiva, com a qual os usuários percorrem a interface em busca de falhas fatais e vulnerabilidades. Tal abordagem teve muito sucesso em aplicações como jogos e gradativamente tem sido usada em outras aplicações (SHNEIDERMAN e PLAISANT, 2010).



Ferramentas para avaliação automatizada também podem ser usadas por especialistas quando há um grande número de interfaces a ser verificado. Uma delas é o Webtango, que pode auxiliar a acelerar esta análise (SHNEIDERMAN e PLAISANT, 2010).

Algumas ferramentas captam o comportamento do usuário, fazem análises e dão subsídio a tomadas de decisão em relação à melhorias e ajustes na interface. Permitem identificar, por exemplo, se uma árvore de menu está muito profunda ou contém redundâncias, se *widget labels* foram usados consistentemente, se todos os botões tem funções associadas; se o alinhamento e simetria estão corretos, se o *layout* está limpo, sofisticado ou criativo e se a velocidade está rápida demais mas há um conflito entre metas do projeto de interface (no caso de *e-commerce* e entretenimento a atratividade da interface pode ser mais importante que a sua velocidade).

Já o método de inspeção semiótica utiliza uma abordagem antecipativa por parte da análise de um especialista percorrendo a interface e buscando potenciais pontos de ruptura e fragilidade na interação, conforme Prates e Barbosa (2003).

Trabalhos como o de Junior et al. (2004), apresentam a análise das três técnicas mencionadas anteriormente (percurso cognitivo, testes de usabilidade e métodos de inspeção como a avaliação heurística), para o uso no caso de processos para a reengenharia de *software*.

O uso de técnicas como o MoLIC (*Modeling Language for Interaction as Conversation*), proposta por Paula (2003), pode ser utilizado como uma linguagem para a modelagem da interação humano-computador. A técnica se apresenta como uma solução de apoio a um processo definido de planejamento de interações e pressupõe simulações de interação com um usuário não totalmente definido.

Avelino (2005) propõe um quadro comparativo entre diversos métodos de avaliação de usabilidade, conforme ilustrado na Tabela 6.

**Tabela 6** - Classificação dos métodos de avaliação de usabilidade

|                                                                                       | Método de Avaliação                      | Fase do Ciclo de Vida |         |                     |          | Envolvimento do Usuário |                 |                   |
|---------------------------------------------------------------------------------------|------------------------------------------|-----------------------|---------|---------------------|----------|-------------------------|-----------------|-------------------|
|                                                                                       |                                          | Análise de Requisitos | Projeto | Codificação e Teste | Operação | Método Especialista     | Método Empírico | Método Contextual |
| TESTE                                                                                 | Protocolo pensando alto                  |                       | X       | X                   | X        |                         | X               |                   |
|                                                                                       | Protocolo perguntando-questões           |                       | X       | X                   | X        |                         | X               |                   |
|                                                                                       | Aprendizado de co-descoberta             |                       | X       | X                   | X        |                         | X               |                   |
|                                                                                       | Método de ensino                         |                       | X       | X                   | X        |                         | X               |                   |
|                                                                                       | Método do treinador                      |                       | X       | X                   | X        |                         | X               |                   |
|                                                                                       | Teste retrospectivo                      |                       | X       | X                   | X        |                         | X               | X                 |
| INSPEÇÃO                                                                              | Medida de desempenho                     |                       |         | X                   | X        | F                       | X               |                   |
|                                                                                       | Walkthrough pluralista                   |                       | X       |                     |          | H                       |                 |                   |
|                                                                                       | Walkthrough cognitivo                    |                       | X       | X                   | X        | H/F                     |                 |                   |
|                                                                                       | Avaliação heurística                     |                       | X       | X                   | X        | H                       |                 |                   |
|                                                                                       | Lista de verificação baseada em cenários |                       | X       | X                   | X        | H                       |                 |                   |
| QUESTÃO                                                                               | Inspeção de características              |                       |         | X                   | X        | H                       |                 |                   |
|                                                                                       | Estudo de campo pró-ativo                | X                     | X       |                     |          |                         |                 | X                 |
|                                                                                       | Observação de campo                      | X                     | X       | X                   | X        |                         |                 | X                 |
|                                                                                       | Entrevista                               | X                     | X       | X                   | X        |                         | X               | X                 |
|                                                                                       | Entrevista contextual                    | X                     | X       | X                   | X        |                         |                 | X                 |
|                                                                                       | Questionário / <i>user survey</i>        | X                     | X       | X                   | X        |                         | X               | X                 |
|                                                                                       | Grupo de foco                            | X                     | X       | X                   | X        |                         | X               | X                 |
|                                                                                       | Tempestade Cerebral                      | X                     | X       | X                   | X        |                         | X               | X                 |
|                                                                                       | Análise de registro de uso               |                       |         | X                   | X        | F                       | X               | X                 |
| Legenda: os métodos especialistas são classificados em Formais - F ou Heurísticos - H |                                          |                       |         |                     |          |                         |                 |                   |

Fonte: Avelino (2005)

Uma outra abordagem para orientar avaliações é o *framework* DECIDE (*determine, explore, choose, identify, decide, evaluate*), composto pela seguinte lista de checagem (PREECE, ROGERS e SHARP, 2005):

- Determinar as metas: entender quais são as metas, quem as quer e por quê. As metas escolhidas influenciam a abordagem de avaliação a ser utilizada;
- Explorar as questões: identificar as questões cujas respostas satisfaçam as metas e decompor em subquestões mais aprimoradas e específicas;

- Escolher o paradigma de avaliação e as técnicas: os paradigmas podem ser avaliações “rápidas e sujas”, testes de usabilidade, estudos de campo ou avaliação preditiva;
- Identificar as questões práticas a serem abordadas: como a seleção das tarefas típicas a serem testadas, a seleção apropriada dos usuários (respeitando perfil, categoria, representatividade, disponibilidade, entre outros aspectos já mencionados), equipamentos necessários (computadores, câmeras e demais equipamentos para gravação de áudio e vídeo, laboratório e sala de observação, etc), restrições de cronograma e/ou orçamento e necessidades de conhecimento especializado por parte da equipe de avaliação;
- Decidir como lidar com as questões éticas: diz respeito ao cumprimento dos códigos de conduta existentes, aspectos de privacidade e confidencialidade das informações, não-atribuição de usuários, direitos e consentimento por parte do usuário, apropriação de relatos e histórias pessoais, eventual recompensa ou pagamento ao usuário participante, entre outras;
- Avaliar, interpretar e apresentar os dados: geralmente, o paradigma de avaliação e técnica adotados determinam os tipos de dados coletados, como analisá-los e apresentar as descobertas, mas deve-se considerar questões como confiabilidade e validade dos resultados obtidos e a possibilidade de desvios (distorção dos resultados) como, por exemplo, por falha de aptidão/experiência do avaliador, do observador ou de outro membro da equipe ou distorção do resultado devido a validade ecológica (efeito *Hawthorne*, por exemplo), onde o ambiente em que a avaliação é conduzida influencie os resultados.

Com essa apresentação dos métodos alternativos de avaliação, encerra-se essa seção sobre avaliação iterativa do Modelo Conceitual. A seguir, será apresentada a próxima atividade do ciclo de vida da fase de Projeto, Teste e Desenvolvimento: a definição dos padrões de *design* de telas.

#### 4.2.5 Definição dos Padrões de *Design* de Telas

Para o caso de aplicativos baseados em plataforma de interface gráfica de usuário (GUI), os passos abaixo são recomendados. A partir destes, são obtidos uma série de esboços de padrões que serão posteriormente detalhados e validados.

Primeiramente é feito um esboço da padronização do uso de controles na interface (como *option button*, *check box*, *list box*, *spin box* e *drop-down box*, por exemplo) especificando as condições em que cada um deve ser utilizado, a fim de eliminar inconsistências, confusões e erros.

Em seguida, são tratados os padrões para janelas, formulários, planilhas e outros, dependendo do modelo conceitual adotado (orientado a produto ou a processo). Também devem ser esboçados padrões para caixas de diálogo, estabelecendo o *design* para campos obrigatórios *versus* opcionais, editáveis ou apenas de leitura, entre outros, bem como os padrões para caixas de mensagens, tratando os diferentes tipos de caixas como de erro, alerta, *status*, bem como questões de posicionamento, sintaxe, formato, *label* de botões, entre outros aspectos. Também devem ser definidos padrões de interação com os dispositivos de entrada, ou seja, as regras de como será dado o *feedback* ao usuário na interface, por meio de *mouse buttons*, *mouse clicks* e *shortcut/accelerator keys*, por exemplo. Há ainda os padrões de *feedback* que estabelecem como a interface exibirá certos tipos de *feedback* como, por exemplo, detalhes de cor, destaque, piscar, forma, tamanho e outros para demonstrar diferentes ações ou *status* como seleção, tarefa em andamento, tarefa concluída, processo ativo, etc. Shneiderman e Plaisant (2010) alertam para que se busque utilizar indicadores de progresso dinâmicos e gráficos, com informações numéricas e em destaque (ao invés de mensagens estáticas como “*por favor aguarde*”) desde que estas sejam confiáveis (a exemplo de uma barra de progresso que demonstra uma atividade em andamento e um percentual a ser concluído quando na verdade houve um problema de conexão e a atividade está parada ou foi cancelada).

Como uma técnica alternativa para se obter os PDT, pode-se usar o *design* participativo, descrita ao final da seção “Projeto do Modelo Conceitual” do presente trabalho.

Os padrões devem ser documentados. Pelos mesmos motivos expostos anteriormente na seção “Projeto do Modelo Conceitual”, Mayhew (1999) sugere que também não se invista tempo em documentar tais padrões por meio de ferramentas gráficas.

#### **4.2.6 Elaboração de Protótipos dos Padrões de *Design* de Telas**

Assim como discutido previamente na seção “Elaboração de maquetes do Modelo Conceitual”, o primeiro passo nesse momento é selecionar as funcionalidades a serem prototipadas, baseando-se nos pontos de maior preocupação e interesse da interface. Pode-se selecionar as funções-chave do sistema, as funcionalidades com o maior número de telas envolvidas, as funcionalidades mais frequentemente usadas ou mais representativas dentro de uma funcionalidade maior, as funções consideradas como potencialmente problemáticas ou funções executadas em sequência.

Em seguida, deve-se preparar uma especificação informal (com papel e lápis) para as funcionalidades escolhidas. Pode ser feito um esboço do *design* detalhado das telas reais incluindo janelas, controles de ações, caixas de diálogo, mensagens, menus, interações e caminhos, bem como rótulos, disposição e organização dos mesmos (MAYHEW, 1999).

Com base nessa especificação são criados os protótipos, que podem ser de baixa ou alta fidelidade.

Os protótipos de baixa fidelidade não se assemelham muito ao produto final, são simples, baratos e de rápida produção. Por serem facilmente modificáveis, oferecem excelente suporte à exploração de *designs* e idéias alternativas e são particularmente indicados nos primeiros estágios do desenvolvimento (PREECE, ROGERS e SHARP, 2005). Normalmente, utiliza-se papel por meio de métodos como *storyboards*, esboços, maquetes ou fichas. *Storyboards* ou narrativas gráficas são protótipos de baixa fidelidade utilizados para representar as interações entre o usuário e o sistema. A representação é feita por uma sequência de desenhos de esboços de tela e elementos do contexto de uso (CYBIS, BETIOL e FAUST, 2007).

Já os protótipos de alta fidelidade oferecem componentes de interface, aparência e comportamento bem parecidos com o futuro sistema, pois são desenvolvidos por meio da utilização de ferramentas de *software*. O conteúdo de informação é normalmente mais elaborado e possibilita a obtenção de medidas de usabilidade (CYBIS, BETIOL e FAUST, 2007). Utiliza-se aplicativos para geração dos protótipos, como o Macromedia Director, Visual Basic ou Smaltalk, por exemplo.

A Tabela 7 mostra algumas vantagens e desvantagens de cada tipo de protótipo.

**Tabela 7** - Eficácia relativa de protótipos de baixa vs alta fidelidade

| Tipo             | Vantagens                                                                                                                                                                                                                                                                                                                                                                                                                        | Desvantagens                                                                                                                                                                                                                                                                                                                                                                 |
|------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Baixa fidelidade | <ul style="list-style-type: none"> <li>• Possui custo mais baixo de desenvolvimento</li> <li>• Avalia múltiplos conceitos de <i>design</i></li> <li>• É um instrumento de comunicação útil</li> <li>• Aborda questões de leiaute de tela</li> <li>• É útil para identificação de requisitos de mercado</li> <li>• Demonstra que o conceito funciona</li> <li>• Permite prova de conceito (<i>proof-of-concept</i>)</li> </ul>    | <ul style="list-style-type: none"> <li>• Verificação de erros é limitada</li> <li>• Especificação é pobre em detalhes para codificação</li> <li>• “Uso” é conduzido pelo facilitador</li> <li>• Utilidade é limitada após estabelecimento dos requisitos</li> <li>• Utilidade é limitada para testes de usabilidade</li> <li>• Há limitações de fluxo e navegação</li> </ul> |
| Alta fidelidade  | <ul style="list-style-type: none"> <li>• Demonstra funcionalidades completas</li> <li>• É totalmente interativo</li> <li>• O uso é conduzido pelo usuário</li> <li>• Define claramente o esquema de navegação</li> <li>• Uso permite exploração e teste</li> <li>• Possui o mesmo <i>look and feel</i> do sistema final</li> <li>• Serve como uma especificação viva</li> <li>• É uma ferramenta de venda e marketing</li> </ul> | <ul style="list-style-type: none"> <li>• Desenvolvimento é mais caro</li> <li>• Criação demanda tempo</li> <li>• Ineficiente para prova de conceito (<i>proof-of-concept</i>)</li> <li>• Não serve para coleta de requisito</li> </ul>                                                                                                                                       |

**Fonte:** Adaptado de Preece, Rogers e Sharp (2005, p. 266)

Em relação a evolução ou não do protótipo nas fases seguintes do desenvolvimento, pode-se optar por uma das duas abordagens: prototipação evolutiva ou descartável. Na prototipação evolutiva o protótipo evolui para o produto final e, portanto, não será perdido. Entretanto, deve ser submetido a testes rigorosos para que seja incorporado ao sistema. Na prototipação descartável o protótipo é jogado fora e, portanto, não precisa ser testado. No entanto, desta forma a interface final é construída a partir do zero.

### **4.2.7 Avaliação Iterativa dos Padrões de *Design* de Telas**

Para essa atividade, cabem as mesmas técnicas anteriormente descritas na seção “Avaliação Iterativa do Modelo Conceitual”. A diferença é que nesse momento já há uma evolução maior no detalhamento da especificação da interface. Portanto, as técnicas e passos são os mesmos utilizados anteriormente, mas o foco aqui passa a ser os PDT e não mais o MC.

Uma prática recomendada é evitar utilizar como testadores os mesmos usuários selecionados para a avaliação dos MC pois esses já podem estar acostumados com a interface anterior e distorcer a medição da facilidade de aprendizado da interface, ou então simplesmente se confundir comparando com a versão anterior. Além disso, perde-se a oportunidade de aumentar o número de diferentes indivíduos a testar a interface. Os testes de PDT devem ser mais detalhados do que os de MC, a fim de validar transações completas em telas individuais e não mais apenas a navegação correta entre telas (MAYHEW, 1999). Na condução dos testes, os avaliadores e/ou observadores não devem influenciar a tomada de ações do usuário em qualquer sentido e nem fornecer informações sobre o funcionamento da interface, a fim de se ter um resultado isento.

### **4.2.8 Desenvolvimento do Guia de Estilo**

O Guia de Estilo começou a ser elaborado na fase de Análise de Requisitos e é normalmente concluído neste estágio do projeto, consolidando-se em um único documento (utilizado pelos projetistas e desenvolvedores) todos os resultados e padrões relevantes obtidos até o momento.

O Guia de Estilo deve considerar os produtos de trabalho de todas as tarefas da Análise de Requisitos, incluindo perfil do usuário, análise contextual das tarefas, capacidades e restrições de plataforma, objetivos de usabilidade, modelos ajustados após a reengenharia do trabalho, MC validado (que fará referência às características como a plataforma adequada, à família de produtos a que pertence o aplicativo gerado e a integração com outros guias de estilo corporativos) e PDT validados.

Os projetistas, desenvolvedores, usuários e demais envolvidos precisam não apenas ter acesso a essa documentação como também utilizá-la efetivamente,

segundo Mayhew (1999). O apoio da gestão nesse passo é fundamental, reforçando a obrigatoriedade do uso desses padrões nos projetos de interfaces.

A nomenclatura “guia de estilo” muitas vezes não é adotada pelas organizações. Nesse caso, um artefato alternativo pode ser utilizado, desde que contemple as considerações feitas nessa seção.

#### **4.2.9 Elaboração do *Design* Detalhado da Interface com o Usuário**

Nesta atividade, todo o conjunto detalhado de *widgets* (caixas, menus, ícones, barras de ferramentas, etc) devem ser concluídos, usando-se sempre como referência o Guia de Estilo. Pode-se eventualmente utilizar para esta atividade como técnica alternativa o *design* participativo, descrito anteriormente no presente trabalho na seção “Projeto do Modelo Conceitual”.

Deve-se concluir nesta atividade a identificação dos principais caminhos entre janelas, caixas de diálogo e caixas de mensagens (atividade iniciada na elaboração do MC), além do detalhamento completo do *design* de barras de menus e demais controles (atividade iniciada na elaboração do MC e PDT).

Definições relacionadas aos menus devem ser concluídas, como os seus tipos (único ou múltiplo, *pull-down/toolbar/ribbon*), suas características (lista pequena ou longa, a ser exibida em *display* pequeno ou grande, menu em áudio e/ou vídeo), sua exibição (explícita como os anteriormente mencionados ou implícita como os *embedded* menus e *hotlinks*), complexidade (estáticos, dinâmicos, simultâneos ou organizados em árvores) e sua organização e acesso a partir de outros locais, como em mapas de *websites*, por exemplo (SHNEIDERMAN e PLAISANT, 2010). O *design* do conteúdo de todas as telas, janelas, caixas de diálogo e caixas de mensagens deve ser concluído (atividade iniciada na elaboração do MC, prototipação e avaliações de usabilidade, onde as principais telas foram identificadas e testadas mas não havia até então um detalhamento do conteúdo dessas telas).

Além disso, deve-se concluir o *design* de todas as interações com dispositivos de entrada, incluindo detalhes de todos os meios pelos quais os usuários podem interagir com a interface, abrangendo desde dispositivos mais convencionais como teclado (considerando teclas de atalho e aceleradores), mouse (detalhando o uso de



*single-clicks, double-clicks, drag-and-drops*), tela sensível ao toque (*touchscreen* ou *touchpad*), voz, caneta (*light pen* ou *smart pen*) até dispositivos menos usuais como *trackball*, *trackpoint*, controles para mãos (luvas) ou pés, joystick, sensores, papel digital, teclados diferenciados (coletores ou *smartphones*), mesas (*surface*) e/ou outros dispositivos compatíveis com o aplicativo criado. Outras informações sobre esse item foram mencionadas anteriormente na seção “Levantamento das Capacidades e Restrições de Plataforma” do presente trabalho.

#### 4.2.10 Avaliação Iterativa do *Design* Detalhado da Interface com o Usuário

Para essa atividade cabem os mesmos passos e técnicas anteriormente descritas na seção “Avaliação Iterativa do Modelo Conceitual” e que valem também para a avaliação iterativa dos PDT, contudo o foco aqui passa a ser o *Design* Detalhado da Interface com o Usuário (DDIU).

### 4.3 Atividades da Fase de Instalação

Nesta última fase, a atividade de obtenção do *feedback* do usuário é realizada, conforme a Figura 13. Uma vez que o usuário participou desde o início da elaboração do projeto de interface, seu conhecimento sobre o sistema permite um *feedback* que pode auxiliar para melhorias, projetos de novas versões e até mesmo novos produtos correlacionados.



**Figura 13** - Fase de Instalação  
**Fonte:** Adaptado de BARBOSA e SILVA (2010)

Será apresentada a seguir a única atividade desta fase: a obtenção do *feedback* do usuário.

### 4.3.1 Obtenção do *Feedback* do Usuário

Mayhew (1999) aponta seis alternativas de técnicas possíveis para essa atividade: questionários, testes de usabilidade, entrevistas, grupos focais, abordagem automatizada e estudos de utilização. Conforme mencionado anteriormente, outros autores, como Preece, Rogers e Sharp (2005) dão maior ênfase a essas técnicas para atividades como levantamento de requisitos.

Em relação aos questionários e entrevistas, os passos são semelhantes aos descritos na seção “Obtenção do Perfil do Usuário” do presente trabalho. Quanto aos testes de usabilidade, os passos são os mesmos descritos nas três seções sobre avaliação iterativa (do MC, dos PDT e do DDIU). Sobre as entrevistas e os grupos focais (grupos de foco ou *focus groups*), os passos e técnicas são os mesmos mencionados anteriormente na seção “Análise de Contexto da Tarefa”. Para a abordagem automatizada, pode-se utilizar o *design* participativo, descrito anteriormente na seção “Projeto do Modelo Conceitual”.

As demais técnicas e diretrizes para questionários, entrevistas, etnografia e observação participativa mencionadas anteriormente no presente trabalho (na fase de análise de requisitos e nos passos de avaliação iterativa da interface) também se aplicam a essa última fase do projeto.

Quanto aos estudos de utilização, primeiramente define-se a técnica de coleta de dados, que pode ser por observação randômica (presencial) ou por meio de monitoramento remoto por *software*. Em ambas, o usuário é avisado previamente de que em algum momento em que estiver utilizando a interface será monitorado para que se colem as informações. Os estudos são executados em diferentes horários, pois os resultados e o trabalho podem variar. Os dados são coletados e analisados, buscando-se perceber as funções que são usadas com baixa frequência ou por pequenos períodos e tentando-se apurar os motivos. As conclusões são documentadas, relatando-se os eventuais problemas na usabilidade que causaram o baixo uso das funções identificadas.

## 5. Construção do guia para avaliação de usabilidade

Neste capítulo serão apresentadas as etapas e atividades realizadas para a elaboração do guia, objeto desta dissertação. No capítulo seguinte será apresentado o guia proposto.

A primeira etapa foi uma análise e consolidação de todas as práticas de usabilidade levantadas, sob uma perspectiva de processos. O objetivo foi analisar as diversas práticas distribuídas pelas normas, modelos e demais referências pesquisadas e consolidar as mais relevantes e corroboradas em um conjunto mínimo que serviria de base para a elaboração do guia.

Foi decidido iniciar-se pelo compêndio de melhores práticas, por possuir a maior quantidade de informações e corroborações e em seguida verificar as normas e modelos. Todo o capítulo “Melhores Práticas de Usabilidade” foi reavaliado e, em seguida, foram eliminadas as redundâncias (devido às corroborações) a fim de se chegar a um conjunto conciso de práticas. Os critérios adotados para a seleção das atividades a serem incluídas no guia foram os seguintes: atividades que dêem suporte aos processos das três fases do ciclo de vida da engenharia de usabilidade (análise de requisitos, projeto/teste/desenvolvimento e instalação) e que estejam mencionadas em no mínimo três referências pesquisadas e aplicáveis a estas fases. Esse referencial é mostrado em detalhes no item “justificativa” de cada uma das atividades do guia.

O próximo passo foi de estabelecer uma estrutura e nomenclatura para o guia. Nesse estágio, foi observada uma questão de divergência quanto às nomenclaturas. Foram pesquisados diversos tipos de práticas, dentre elas atividades, passos, técnicas, práticas, diretrizes, recomendações e observações. No entanto, para a elaboração do guia, o ideal é que haja um padrão. Como a maioria das referências voltadas ao foco de avaliação de processos utiliza a abordagem de “atividades”, foi adotada, da mesma forma, a nomenclatura de “atividade” para consolidar as práticas. Desta forma, dentro de cada atividade, podem ser inseridas as sub-atividades, técnicas, diretrizes e demais recomendações. Foi necessário, no entanto, uma releitura das práticas consolidadas a fim de transcrevê-las para a estrutura e nomenclatura adotados para este trabalho. Algumas práticas foram

transcritas para um formato de atividades ou então organizadas de forma que pudessem ser incorporadas a uma atividade principal. Nesse estágio, chegou-se a um conjunto consolidado de atividades principais, ilustrado na Tabela 8.

**Tabela 8** - Primeira consolidação de atividades para o guia

| <b>Atividade</b> | <b>Descrição</b>                                                                                                               |
|------------------|--------------------------------------------------------------------------------------------------------------------------------|
| 1                | Conscientizar o usuário e a gestão sobre a importância da usabilidade                                                          |
| 2                | Estabelecer objetivos claros de usabilidade                                                                                    |
| 3                | Estabelecer diretrizes de usabilidade                                                                                          |
| 4                | Estabelecer os processos da metodologia de desenvolvimento de <i>software</i> da organização, sob a perspectiva da usabilidade |
| 5                | Dimensionar o esforço destinado às atividades voltadas à usabilidade                                                           |
| 6                | Identificar o perfil do usuário, sob a perspectiva da usabilidade                                                              |
| 7                | Identificar as tarefas e contexto de uso, sob a perspectiva da usabilidade                                                     |
| 8                | Estabelecer um guia de estilo a ser usado pelo projeto                                                                         |
| 9                | Identificar necessidades e requisitos de usabilidade                                                                           |
| 10               | Construir versões iterativas, intermediárias e iterativas da interface                                                         |
| 11               | Executar avaliações iterativas da interface ao longo de sua construção                                                         |
| 12               | Realizar sessão para <i>feedback</i> do usuário e lições aprendidas                                                            |

Uma vez encerrado esse levantamento no compêndio de melhores práticas, o passo seguinte foi examinar as normas e modelos. Foram buscados os itens das normas que possuíam maior relevância e/ou corroboração nos compêndios de melhores práticas e verificar se poderiam ser cobertos pelas atividades até então elencadas. Os mesmos critérios mencionados anteriormente foram adotados para a seleção das atividades a serem incluídas no guia. Verificou-se nesse momento uma necessidade de se incorporar atividades de acompanhamento, monitoramento e controle sobre as atividades já definidas, de modo a garantir que estas sejam tratadas ao longo de todo o ciclo de vida do projeto. O próximo passo foi examinar cuidadosamente cada uma das atividades, sub-atividades, técnicas e demais itens que fariam parte do guia, a fim de observar eventuais pontos que pudessem ser questionados e consequentemente tratados de forma diferente no guia proposto. Foram elencados alguns pontos de melhoria que serão apresentados a seguir.

Um ponto observado é que não há um processo claro de gestão de mudanças no ciclo de vida de engenharia da usabilidade, que atue entre a fase de análise de requisitos e a fase iterativa de projeto/teste/desenvolvimento. Como exemplo, se durante a fase de projeto/teste/desenvolvimento houver uma solicitação de mudança no projeto, que impacte em resultados da fase anterior (perfil de usuário, tarefas, diretrizes, restrições de plataforma ou objetivos de usabilidade), o ciclo de vida não

mostra claramente que a análise de impacto de tal mudança irá considerar o impacto sobre tais resultados e, no caso da solicitação ser aprovada, que seja providenciada a atualização do guia de estilo que já está sendo utilizado pelo projeto. Podemos observar que outras referências incluem processos de gestão de mudanças, como o processo SUP.10 (gestão de mudança) da ISO/IEC 15504-5 (2006), os processos HCD.7.1 (gerenciar as mudanças) e HCD.7.2 (determinar o impacto na organização e nos *stakeholders*) da ISO/TR 18529 (2000) e os processos da área de conhecimento de Integração no PMBOK (2008).

Para tratar esse ponto, foi definida uma atividade de revisão da metodologia de desenvolvimento de *software* da organização. Esta atividade possui vários objetivos, entre eles a verificação do processo de gestão de mudanças previsto nessa metodologia, a fim de checar se o processo contempla a verificação do impacto à usabilidade na análise de impacto de uma solicitação de mudança.

Outro ponto importante é que o ciclo não faz distinção explícita entre atividades organizacionais e atividades de projetos. Certas atividades não precisam ser necessariamente executadas no ciclo de vida de cada projeto pois uma vez definidas e estabelecidas para a organização podem valer para um conjunto de projetos, de sistemas, programas, portfólios ou até mesmo para todos os projetos da organização. Esse ponto é tratado em referências como a NBR ISO/IEC 12207 (2009) no item 5.1.4 (adoção no nível de projeto ou organizacional) e nos processos de gerenciamento de portfólio desta norma, entre outros. No caso do guia, pode ser aplicável, por exemplo, ao estabelecimento das diretrizes e dos objetivos de usabilidade, à obtenção do perfil do usuário e à adequação da metodologia aos processos voltados à usabilidade. A organização deve, no entanto, ter processos que garantam a execução de tais atividades bem como sua manutenção e revisão.

Para tratar esse ponto, essas atividades foram classificadas no guia proposto como organizacionais e não mais pertencentes a um grupo de processos específico. Desta forma, há uma distinção clara entre as atividades que devem ser executadas no âmbito da gestão do projeto e as que devem ser tratadas por um processo organizacional (permeando todos os projetos).

O ciclo de vida não faz menção explícita a atividades de auditoria de processos (e não de produtos) que garantam que os processos voltados à busca da usabilidade, estabelecidos para a organização e para os seus projetos, sejam devidamente seguidos. Com isso, pode haver problemas para garantir que, por exemplo, as diretrizes e o guia de estilo estabelecidos (fundamentais para a garantia da usabilidade) sejam acompanhados e mantidos. Consequentemente, os objetivos de usabilidade correm risco de serem comprometidos.

Para tratar essa questão, foi aproveitada a distinção proposta entre atividades organizacionais e por projeto, de forma a facilitar o entendimento quanto às atividades e nomenclatura para a questão de revisões e auditorias necessárias. Para os projetos, o guia propõe que sejam realizadas auditorias de processo, que verificam se os processos de usabilidade definidos estão sendo seguidos, e auditorias de produto, que verificam a aderência dos produtos gerados às diretrizes e aos objetivos de usabilidade. Já para a organização, o guia propõe revisões periódicas da metodologia de desenvolvimento de *software*, com foco na manutenção dos processos, atividades, objetivos e diretrizes de usabilidade definidos.

Outra questão referente ao ciclo de vida de engenharia de usabilidade é a nomenclatura utilizada. A análise de requisitos, por exemplo, em algumas abordagens não é tratada como uma fase mas sim como uma atividade dentro de uma fase de análise do sistema, projeto lógico, planejamento do projeto, entre outras nomenclaturas. Para a fase de projeto/testes/desenvolvimento são utilizadas outras nomenclaturas como projeto físico, construção, execução, entre outras. A fase de instalação é também conhecida como implantação, implementação, finalização, encerramento, entre outras.

Para tratar esse ponto, no guia proposto foi usada a nomenclatura de “grupos de processos”, aderente ao PMBOK (2008) que agrupa as atividades em cinco grupos de processos que podem ser executados mais de uma vez no ciclo de vida do projeto: iniciação, planejamento, execução, monitoramento e controle e encerramento. Todas as atividades do guia estão associadas a um desses cinco grupos, exceto as atividades classificadas como organizacionais.

Ao final desses passos, chegou-se a um novo conjunto revisado de atividades principais que podem formar o guia, ilustrado na Tabela 9.

**Tabela 9** - Conjunto de atividades propostas para o guia

| Atividade | Descrição                                                                                                                       | Grupo de Processo        |
|-----------|---------------------------------------------------------------------------------------------------------------------------------|--------------------------|
| 1         | Conscientizar o usuário e a gestão sobre a importância da usabilidade                                                           | Organizacional           |
| 2         | Estabelecer objetivos claros de usabilidade                                                                                     | Planejamento             |
| 3         | Monitorar o atingimento dos objetivos de usabilidade                                                                            | Monitoramento e Controle |
| 4         | Estabelecer diretrizes de usabilidade                                                                                           | Organizacional           |
| 5         | Monitorar o uso e efetividade das diretrizes de usabilidade                                                                     | Monitoramento e Controle |
| 6         | Revisar a metodologia de desenvolvimento de <i>software</i> da organização, sob a perspectiva da usabilidade                    | Organizacional           |
| 7         | Monitorar se os projetos seguem os processos de usabilidade da metodologia de desenvolvimento de <i>software</i> da organização | Monitoramento e Controle |
| 8         | Dimensionar o esforço destinado às atividades voltadas à usabilidade                                                            | Planejamento             |
| 9         | Identificar o perfil do usuário, sob a perspectiva da usabilidade                                                               | Organizacional           |
| 10        | Identificar as tarefas e contexto de uso, sob a perspectiva da usabilidade                                                      | Planejamento             |
| 11        | Estabelecer um guia de estilo a ser usado pelo projeto                                                                          | Planejamento             |
| 12        | Monitorar se os projetos seguem um guia de estilo estabelecido                                                                  | Monitoramento e Controle |
| 13        | Identificar necessidades e requisitos de usabilidade                                                                            | Planejamento             |
| 14        | Revisar requisitos de usabilidade ao longo da construção da interface                                                           | Monitoramento e Controle |
| 15        | Construir versões iterativas, intermediárias e interativas da interface                                                         | Execução                 |
| 16        | Executar avaliações iterativas da interface ao longo de sua construção                                                          | Execução                 |
| 17        | Realizar sessão para <i>feedback</i> do usuário e lições aprendidas                                                             | Encerramento             |

Com as atividades propostas, é possível cobrir todas as práticas levantadas, de uma forma organizada. Por exemplo, conforme o PMBOK (2008), as atividades do grupo de processos de monitoramento e controle permeiam por todo o projeto. Seguindo essa abordagem, serão incorporadas nas atividades destacadas para o grupo de processos de monitoramento e controle do guia, conforme ilustrado na Tabela 9, todas as sub-atividades, técnicas e demais práticas que devem ser tratadas de forma contínua ao longo do projeto. Neste caso, temos as atividades destacadas para os diferentes tipos de monitoramento necessário, como dos objetivos, diretrizes, processos e requisitos de usabilidade, bem como o acompanhamento contínuo do guia de estilo.

Quanto às atividades de planejamento, estas devem ser tratadas o quanto antes no ciclo de vida. A atividade de conscientização do usuários e demais *stakeholders* relevantes, por exemplo, é indicada para que ocorra, quando possível, na iniciação ou bem no início do planejamento, a fim de minimizar o risco de que as atividades previstas para o planejamento tenham problemas de falta de comprometimento por parte dos envolvidos ou falta de recursos para se conduzir um planejamento adequado das atividades voltadas à usabilidade. O mesmo ocorre com a atividade proposta no guia para dimensionar o esforço a estas atividades de usabilidade. Se tal esforço não for dimensionado, corre-se o risco de termos uma estimativa irreal de prazos, custos e até mesmo escopo e qualidade do projeto.

Em relação às atividades de elaborações e prototipações de *design*, propostas no ciclo de vida da engenharia de usabilidade pela abordagem de Mayhew (1999, 2008) que os diferencia em uma evolução de três níveis, o guia consolida tais atividades em uma única atividade (construir versões iterativas, intermediárias e interativas da interface) no grupo de processos de Execução, a fim de facilitar o entendimento já que várias atividades são semelhantes. O mesmo ocorreu com a consolidação das atividades de avaliação em uma única (executar avaliações iterativas da interface ao longo de sua construção).

O próximo passo foi detalhar e refinar os objetivos de cada atividade, a fim de verificar se tais atividades garantiriam a qualidade esperada da usabilidade nos projetos, o que resultou nos objetivos descritos na Tabela 10.

Cada atividade do guia possui ainda um item de justificativa, que aponta os trechos desta dissertação em que tal atividade é tratada de forma mais aprofundada. O objetivo da justificativa é facilitar a busca de detalhes sobre uma atividade dentro do presente trabalho e, portanto, não se trata de um processo de “fichamento” da dissertação, já que o propósito é outro e as referências mencionadas nas justificativas não descrevem as seções e capítulos das obras originais, mas sim da presente dissertação.



Tabela 10 - Objetivos das atividades do guia

| Atividade | Descrição                                                                                                                       | Objetivo                                                                                                                                                                      |
|-----------|---------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1         | Conscientizar o usuário e a gestão sobre a importância da usabilidade                                                           | Garantir que haja o devido comprometimento e recursos para o atingimento da usabilidade pretendida.                                                                           |
| 2         | Estabelecer objetivos claros de usabilidade                                                                                     | Garantir que sejam definidos critérios mensuráveis que estabeleçam o grau de usabilidade pretendido.                                                                          |
| 3         | Monitorar o atingimento dos objetivos de usabilidade                                                                            | Garantir que os objetivos de usabilidade são monitorados e atingidos por meio de análise, ações preventivas e corretivas.                                                     |
| 4         | Estabelecer diretrizes de usabilidade                                                                                           | Garantir que existam diretrizes de usabilidade, claramente definidas e documentadas na organização.                                                                           |
| 5         | Monitorar o uso e efetividade das diretrizes de usabilidade                                                                     | Garantir que os projetos façam uso das diretrizes de usabilidade estabelecidas nos projetos de interface.                                                                     |
| 6         | Revisar a metodologia de desenvolvimento de <i>software</i> da organização, sob a perspectiva da usabilidade                    | Garantir que há processos com foco em usabilidade definidos na metodologia de desenvolvimento de <i>software</i> da organização.                                              |
| 7         | Monitorar se os projetos seguem os processos de usabilidade da metodologia de desenvolvimento de <i>software</i> da organização | Garantir que os projetos façam uso dos processos de usabilidade estabelecidos pela organização.                                                                               |
| 8         | Dimensionar o esforço destinado às atividades voltadas à usabilidade                                                            | Garantir estimativas mais realistas e precisas para os projetos, que considerem o esforço para os processos de usabilidade.                                                   |
| 9         | Identificar o perfil do usuário, sob a perspectiva da usabilidade                                                               | Garantir que o projeto considere características dos usuários relevantes para a usabilidade.                                                                                  |
| 10        | Identificar as tarefas e contexto de uso, sob a perspectiva da usabilidade                                                      | Garantir que o projeto considere características das tarefas e contexto de trabalho que tenham relevância para a usabilidade das interfaces.                                  |
| 11        | Estabelecer um guia de estilo a ser usado pelo projeto                                                                          | Garantir a elaboração progressiva de um documento que oriente quanto às diretrizes, processos, objetivos e atividades da usabilidade.                                         |
| 12        | Monitorar se os projetos seguem um guia de estilo estabelecido                                                                  | Garantir que o guia de estilo estabelecido para o projeto está sendo respeitado.                                                                                              |
| 13        | Identificar necessidades e requisitos de usabilidade                                                                            | Garantir que as necessidades de usabilidade sejam identificadas e incorporadas aos requisitos do projeto.                                                                     |
| 14        | Revisar requisitos de usabilidade ao longo da construção da interface                                                           | Garantir que os requisitos de usabilidade definidos sejam detalhados de forma iterativa e progressiva.                                                                        |
| 15        | Construir versões iterativas, intermediárias e interativas da interface                                                         | Garantir que a abordagem de construção da interface seja centrada no usuário, com evolução gradativa e validação dos requisitos de usabilidade.                               |
| 16        | Executar avaliações iterativas da interface ao longo de sua construção                                                          | Garantir, por meio de avaliações formais, que o <i>design</i> esteja aderente às necessidades, requisitos e objetivos de usabilidade do usuário, do projeto e da organização. |
| 17        | Realizar sessão para <i>feedback</i> do usuário e lições aprendidas                                                             | Garantir a aceitação formal da usabilidade do produto final e que lições aprendidas sejam documentadas e consideradas em fases ou projetos futuros.                           |

Uma vez definidas todas as atividades e respectivos objetivos e grupos de processos, o passo final para a elaboração do guia foi o detalhamento das técnicas e evidências objetivas. O detalhamento foi feito pela descrição das diversas técnicas, ferramentas ou formas de condução, aplicáveis a cada uma das atividades, ou ainda pela decomposição da atividades em sub-atividades e respectivas técnicas. Em seguida foram elencadas as evidências objetivas mais tipicamente utilizadas em cada atividade, que podem comprovar que a organização faz uso da respectiva atividade. Essas evidências foram extraídas das respectivas referências que deram origem às técnicas e atividades elencadas e constituem na maior parte a própria saída ou resultado gerado pela execução da atividade. Outras referências consultadas foram as tabelas de saídas dos processos (*associated work products from HCD processes*) do modelo UMM (EARTHY, 2000) e da ISO/TR 18529 (2000), bem como os produtos típicos de trabalho (*typical work products*) apresentados em cada prática específica do modelo CMMI (CHRISSIS, KONRAD e SHRUM, 2003). Alguns outros termos mencionados já são parte de uma terminologia padrão na área de desenvolvimento de sistemas (como plano de projeto, cronograma, declaração de escopo, entre outras). Assim como as técnicas e sub-atividades, as evidências podem variar tanto na nomenclatura como no seu conteúdo conforme a organização e sua metodologia de desenvolvimento de *software*. Por isso, sempre está mencionado no guia frases como “*exemplos incluem, mas não se limitam a*” de forma que a organização possa demonstrar que atende a determinada atividade por meio de uma técnica, ferramenta ou artefato alternativo.

Para finalizar o presente capítulo, será apresentada no parágrafo a seguir uma breve introdução do método proposto para utilização do guia na avaliação dos processos de uma organização. O método será descrito em detalhes no próximo capítulo.

O método contempla as seguintes etapas: sessão prévia, preparação, condução e sessão final. A sessão prévia busca elencar de forma geral e estratégica os problemas atuais da organização que podem estar relacionados à usabilidade. Essa informação será utilizada posteriormente durante a condução da avaliação. Na etapa de preparação a organização deve coletar e disponibilizar para a equipe

envolvida na avaliação todos os documentos de processos (metodologia de desenvolvimento de *software* e documentos correlacionados) e evidências de projetos (planos e outros artefatos de gestão, produtos intermediários e finais do *software* gerados). Durante a etapa de condução da avaliação, são analisadas as atividades do guia proposto pelo presente trabalho e para cada uma destas atividades são registradas as conclusões, na qual cinco questões devem ser respondidas em consenso pelos participantes e avaliador: grau atual de utilização efetiva da atividade na organização, problemas atuais e potenciais correlacionados ao grau de utilização, benefícios atuais e potenciais correlacionados ao grau de utilização, relevância desta atividade para atuar nos problemas e/ou benefícios e, por último, sugestões de ajustes e melhorias do guia para esta atividade (sendo este último um mecanismo para retroalimentação de sugestões e melhorias do guia e do processo de avaliação). A última etapa é composta de uma sessão final onde é apresentado um relatório que demonstre para cada atividade o nível de aderência da organização ao guia e eventuais recomendações de melhoria.

No próximo capítulo são apresentados os dois principais resultados do presente trabalho de pesquisa: a proposta do guia e do método de avaliação de processos utilizando-se o guia. Também é apresentada a forma de verificação da aplicabilidade desses resultados da dissertação.

## 6. Guia Proposto

Este capítulo apresenta três seções. Nas duas primeiras seções são descritos, respectivamente, os dois resultados da pesquisa desta dissertação: a proposta de um guia de referência para qualidade da usabilidade de projetos de interfaces em processos de desenvolvimento de *software* e a proposta de um método a ser adotado para a utilização deste guia em uma verificação de aderência de processos de desenvolvimento de *software* às práticas do guia proposto. Na terceira seção é descrita a forma adotada neste trabalho de pesquisa para verificar a aplicabilidade e utilidade do guia e do método produzidos nesta dissertação.

### 6.1 Atividades do Guia

Conforme mencionado no capítulo anterior, o guia está apresenta dezessete atividades relacionadas à construção de interfaces em processos de desenvolvimento de *software*. Cada atividade possui as seguintes informações: o grupo de processo em que tipicamente esta atividade é executada, o seu objetivo, as técnicas que podem ser utilizadas para a sua realização (bem como eventuais sub-atividades relevantes) e as típicas evidências objetivas que podem comprovar que tal atividade é executada na organização. Adicionalmente, são apresentadas para cada atividade as justificativas, que apontam os respectivos trechos desta dissertação correlacionados à atividade, sendo que essas justificativas não fazem parte do guia propriamente mas destinam-se a facilitar a leitura e análise do presente trabalho.

A seguir são apresentadas as atividades propostas para o guia e suas respectivas informações.

#### 6.1.1 Atividade 1

**Atividade:** Conscientizar o usuário e a gestão sobre a importância da usabilidade.

**Grupo de Processo:** Organizacional.

**Objetivo:** Garantir que haja o devido comprometimento por parte dos envolvidos e recursos adequados para o atingimento dos objetivos de usabilidade, por meio da

conscientização dos envolvidos a respeito dos benefícios do foco em usabilidade para a garantia da qualidade e gestão de riscos no projeto da interface.

**Técnicas:** Treinamento (formal, apresentação em reunião ou *workshop*). Organizações que utilizam a ISO/IEC 15504-5 (2006) podem utilizar o processo organizacional definido e aderente principalmente ao processo MAN.1 (alinhamento organizacional). Organizações que utilizam a ISO/TR 18529 (2000) podem utilizar suas práticas definidas que estejam aderentes a práticas como a HCD.2.4 e HCD.2.7 (garantir e promover a abordagem centrada no humano).

**Evidência Objetiva:** Registro da conscientização (ata de reunião, apresentação com lista de presença ou outro registro similar aplicável)

**Justificativa:** As técnicas apresentadas nesta atividade estão respaldadas no referencial teórico do presente trabalho e evidenciadas no capítulo 1 por Nielsen e Loranger (2007), Shneiderman e Plaisant (2010) e seções 2.1.1 por Prates e Barbosa (2003), 3.2.4 pela ISO 13407 (1999), 3.2.5 pela ISO/IEC 15504-5 (2006) e 3.2.6 pela ISO/TR 18529 (2000).

## 6.1.2 Atividade 2

**Atividade:** Estabelecer objetivos claros de usabilidade.

**Grupo de Processo:** Planejamento.

**Objetivo:** Garantir que sejam definidos critérios mensuráveis que permitam se estabelecer previamente qual o grau de usabilidade que se pretende atingir no projeto da interface. Objetivos tangíveis, qualitativos e/ou quantitativos são estabelecidos e priorizados e devem ser perseguidos ao longo do projeto. Devem existir critérios específicos que permitam medir a qualidade da usabilidade da interface.

**Técnicas:** Devem ser consideradas questões que podem impactar os objetivos. Exemplos dessas questões incluem mas não se limitam a: objetivos da organização, metas de negócio, *benchmarking*, perfil do usuário, contexto e ambiente de trabalho das tarefas, natureza (ou segmento vertical) da aplicação.

Exemplos de medidas gerais e atributos que podem auxiliar na definição dos objetivos de usabilidade incluem, mas não se limitam a: grau de satisfação do usuário, facilidade de aprendizado, de uso e de lembrança, produtividade, eficácia, eficiência, segurança, velocidade da operação, robustez, facilidade de recuperação, facilidade de adaptação, além de medidas relacionadas à acessibilidade, conforme a NBR ISO/IEC 25000 (2008), entre outras medidas e atributos.

As métricas que podem ser consideradas para o estabelecimento de objetivos de usabilidade incluem, mas não se limitam a: tempo para concluir uma tarefa, percentual de tarefa concluída, percentual de tarefa concluída por unidade de tempo, proporção de sucessos para falhas, tempo gasto em erros, percentual ou número de erros, percentual ou número de competidores melhor do que o sistema, número de comandos usados, frequência de uso da ajuda e da documentação, percentual de comentários de usuário favoráveis / desfavoráveis, número de repetições de comandos com falha, número de execuções com sucesso e fracasso, número de vezes em que a interface confunde o usuário ou induz a erro, número de características boas e más recordado por usuários, número de comandos disponíveis não utilizados, número de comportamentos regressivos, número de usuários preferindo o seu sistema, número de vezes em que os usuários precisam contornar um problema, número de vezes em que o usuário é interrompido em uma tarefa, número de vezes em que o usuário perde o controle do sistema e número de vezes em que o usuário expressa frustração ou satisfação, entre outros.

Objetivos podem ainda ser definidos com base na abordagem da norma NBR ISO/IEC 9126-1 (2003). De acordo com a norma, a usabilidade deve compreender as capacidades de inteligibilidade (possibilitar ao usuário compreender se o *software* é apropriado e como ele pode ser usado para tarefas e condições de uso específicas), apreensibilidade (possibilitar ao usuário aprender sua aplicação), operacionalidade (possibilitar ao usuário operá-lo e controlá-lo), atratividade (ser atraente ao usuário) e conformidade (aderência às normas, convenções, guias de estilo ou regulamentações relacionadas à usabilidade).

Esta atividade auxilia a justificar os esforços requeridos e planejados para as atividades de usabilidade.

No caso de organizações que utilizam o modelo CMMI (CHRISSIS, KONRAD e SHRUM, 2003), pode-se utilizar os processos da PA de Medição e Análise. Organizações que utilizam a ISO/IEC 15504-5 (2006) podem utilizar o processo organizacional definido e aderente principalmente aos processos MAN.4 (gestão da qualidade) e MAN.6 (medição).

**Evidência Objetiva:** Os objetivos de usabilidade do projeto podem estar definidos em um documento do projeto (termo de abertura do projeto, declaração de escopo, guia de estilo ou similar), em um documento do sistema ou produto (no caso de vários projetos de um mesmo sistema ou uma família de produtos) ou em um documento da metodologia de desenvolvimento de *software* (caso os objetivos sejam exatamente os mesmos para todos os projetos).

**Justificativa:** As técnicas apresentadas nesta atividade estão respaldadas no referencial teórico do presente trabalho e evidenciadas nas seções 2.3 pela NBR 9241-11 (2002), Preece, Rogers e Sharp (2005), Nielsen (1993) e Prates e Barbosa (2003), 3.2.1 pela NBR ISO/IEC 9126-1 (2003), 3.2.2 pela NBR 9241-11 (2002), 3.2.4 pela ISO 13407 (1999), 3.2.5 pela ISO/IEC 15504-5 (2006), 3.3 por Chrissis, Konrad e Shrum (2003) e NBR 9241-11 (2002) e 4.1.3 por Dix et al. (2004).

### 6.1.3 Atividade 3

**Atividade:** Monitorar o atingimento dos objetivos de usabilidade.

**Grupo de Processo:** Monitoramento e Controle.

**Objetivo:** Garantir ao longo do projeto que os objetivos de usabilidade definidos estão sendo perseguidos e atingidos. Buscar medidas preventivas para garantir o seu atingimento e no caso de desvios adotar ações corretivas.

**Técnicas:** Os próprios processos de monitoramento de objetivos e indicadores já existentes na organização, como os aderentes à NBR ISO/IEC 9001 (cobertos por itens como o 5.4, 5.6 e 8.2 da norma, voltados a objetivos de qualidade, análise crítica e monitoramento e medição), à norma ISO/IEC 15504-5 (cobertos pelo processo MAN.4 – gestão da qualidade, principalmente nas práticas PB4, PB5 e

PB6 e pelas práticas do processo MAN.6 - medição) ou ao CMMI (nas práticas da área de processo de MA - medição e análise) podem ser usados para esta atividade.

**Evidência Objetiva:** Registros de eventos (atas de reunião ou outro aplicável) de medição, análise e monitoramento de indicadores ou informações que possibilitem o controle dos objetivos. Registros de reuniões de análise crítica apontando ações preventivas e corretivas ou processo similar que demonstre o acompanhamento das ações relacionadas à busca dos objetivos de usabilidade definidos.

**Justificativa:** As técnicas apresentadas nesta atividade estão respaldadas no referencial teórico do presente trabalho e evidenciadas nas seções 2.3 pela NBR 9241-11 (2002), Preece, Rogers e Sharp (2005), Nielsen (1993), Prates e Barbosa (2003), 3.2.2 pela NBR 9241-11 (2002), 3.2.4 pela ISO 13407 (1999), 3.2.5 pela ISO/IEC 15504-5 (2006), 3.3 por Chrissis, Konrad e Shrum (2003), NBR ISO/IEC 9001 (2008), 3.2.5 pela ISO/IEC 15504-5 (2006) e 4.1.3 por Sommerville (2007) e pela ISO/IEC 15504-5 (2006).

#### 6.1.4 Atividade 4

**Atividade:** Estabelecer diretrizes de usabilidade.

**Grupo de Processo:** Organizacional.

**Objetivo:** Garantir que existam orientações, por meio de diretrizes (*guidelines*) e princípios (gerais e específicos) de usabilidade, claramente documentadas na metodologia de desenvolvimento de *software* da organização (ou outro documento aplicável). Essas diretrizes devem ser seguidas por todos os projetos, a fim de se garantir benefícios como uma padronização e melhor qualidade das interfaces e o uso das melhores práticas de usabilidade.

**Técnicas:** As diretrizes devem estar registradas claramente no guia de estilo de cada projeto (ou documento similar aplicável) ou no caso de diretrizes organizacionais, aplicáveis a todos os projetos, podem estar registradas em algum documento claramente identificado na metodologia (processo, procedimento, manual ou guia). No caso de exceções, como um projeto específico em que determinadas



diretrizes não sejam aplicáveis, deve-se registrar o motivo no próprio guia de estilo ou em artefato similar como um guia de adaptação do projeto.

Exemplos de fontes para a elaboração das diretrizes incluem, mas não se limitam a: lições aprendidas de projetos anteriores (diretrizes e guias de estilo), guias corporativos ou de família de produtos, livros, *journals*, fóruns, artigos e outros materiais de conferências e congressos, como as Oito Regras de Ouro de Shneiderman e Plaisant (2010), as Heurísticas de Nielsen (1993) e demais trabalhos correlatos como a pesquisa posterior de Nielsen e Loranger (2007).

Exemplos de tipos de diretrizes a serem estabelecidas pela organização incluem, mas não se limitam às citadas nos parágrafos seguintes.

Diretrizes gerais devem ser estabelecidas, como o cuidado na padronização de sequências de tarefas, uso de botões adequados conforme a opção de escolha a ser feita pelo usuário, cuidados na impressão (conteúdo e diagramação corretos), uso de pré-visualização de miniaturas de imagens (para não sobrecarregar a aplicação, no caso de arquivos de tamanho grande).

Diretrizes de plataformas, ambiente operacional e *hardware* podem ser estabelecidas como: revisar a documentação das plataformas e ambiente técnico, que podem conter informações sobre detalhes de padrões de interface para alguns componentes específicos (principalmente quando há vários projetistas ou plataformas envolvidos no projeto ou ainda se os projetistas não estão familiarizados com as capacidades e restrições da plataforma). Outro exemplo de diretriz é examinar restrições e capacidades quanto a dispositivos de entrada e saída, principalmente quando o projeto envolve dispositivos não usuais como telas muito grandes (*walls* ou monitores múltiplos) ou muito pequenas (óculos ou lentes) ou *displays* do tipo *heads-up* ou *helmet/head mounted display* (HMD) que podem requerer cuidados específicos para que a usabilidade não seja comprometida. A usabilidade também pode ser comprometida em alguns casos pela inobservância de características importantes em dispositivos de saída como *displays* e monitores como a dimensão, resolução (número de *pixels*), número de cores, luminosidade, taxas de utilização (no caso de permitir animações e vídeos), portabilidade e

privacidade ou dispositivos como impressoras (dimensão, resolução, velocidade, entre outros).

Diretrizes de mapeamento podem ser declaradas, referindo-se à relação entre os controles e seus efeitos, como, por exemplo, convenções universalmente aceitas (por exemplo, padrão de setas para movimento de cursor para cima e para baixo ou padrão de setas e símbolos para controle de áudios e vídeos como *rewind*, *play*, *pause*, *stop* e *forward*).

Diretrizes de *affordance* podem ser estabelecidas para controles como botões, barras de rolagem e ícones, por exemplo.

Diretrizes específicas para ambiente *web* estão presentes em referências como *Sun's Web Design Guide*, *The National Cancer Institute's Research-Based Web Design & Usability Guidelines*, *The World Wide Web Consortium's Web Accessibility Initiative* e *The Web Style Guide*.

Diretrizes para navegação e arquitetura de informação podem incluir orientações e regras quanto à estrutura das telas a ser usada (ou páginas) e como manter sua consistência, recomendações como evitar *designs* rebuscados, redundantes ou com caminhos duplicados, utilizar técnicas como a *breadcrumb trail* para indicar a localização do usuário no contexto hierárquico da aplicação, entre outras.

Diretrizes de textos podem especificar como usar fontes (*fonts*) com serifas e amplamente utilizadas pelos usuários, ter uma quantidade limitada de *fonts* e que sejam coerentes com a identidade visual da aplicação, entre outras.

Diretrizes de apresentação dos elementos de uma tela ou página podem estabelecer como apresentar uma estruturação (disposição dos elementos) da tela ou página considerando a prioridade dos itens, aplicando agrupamento e alinhamento apropriados, evitar páginas longas, exibir claramente a rolagem da página, entre outras.

Diretrizes de organização de layout podem definir como manter a consistência nas sequências de transações de entrada de dados, ter o mínimo (e evitar redundância) de entrada de dados pelo usuário, exigir o mínimo de carga de

memória do usuário, manter a compatibilidade entre os dados exibidos e a entrada de dados, entre outras.

Diretrizes de busca e visualização das informações podem estabelecer que deve-se avaliar o melhor tipo de pesquisa para cada atividade (busca em texto, em imagens ou outras), o melhor método de pesquisa (consulta em linguagem natural, manipulação direta, consulta dinâmica, consulta por palavra-chave, busca por filtros simples ou avançados, entre outras), o melhor tipo de visualização (exibição automática da informação ou após ação de confirmação, uso ou não de *previews*, necessidade de *zoom* e/ou 3D, entre outros) o melhor tipo de saída (lista ou tabela em tela, gráfico, relatório, impressão, arquivo para download ou outro), avaliar a necessidade de disponibilizar ordenação, classificação ou agrupamento para o próprio usuário, entre outras.

Diretrizes de obtenção da atenção do usuário podem estabelecer regras quanto ao uso de recursos aplicados nas fontes (*fonts*) como piscar ou alterar cor, tamanho ou nível de intensidade. A atenção também pode ser chamada pelo uso de recursos de áudio distintos e adequados para cada tipo de *feedback* (positivo, negativo, alerta, confirmação, entre outros).

Diretrizes quanto ao uso adequado de cores podem estabelecer que deve-se manter um esquema de cores coerente com a identidade visual da aplicação, legível, com quantidade de cores limitada e que facilite o uso, usar cores adequadas para mostrar *status* de tarefas (em esquemas de faróis ou barras de progresso, por exemplo), respeitar eventuais convenções existentes de cores na área de aplicação da interface (principalmente voltados às cores vermelho, amarelo e verde), permitir que o usuário possa (sempre que possível) alterar cores conforme suas preferências, considerar eventuais restrições por usuários com deficiências visuais ou por limitações de tecnologia (baixa fidelidade ou distorções no uso por determinados equipamentos).

Diretrizes como as Oito Regras de Ouro de Shneiderman e Plaisant podem ser usadas. Trata-se de um conjunto das seguintes heurísticas: esforçar-se para consistência (sequência de ações, terminologia comum em menus, *prompts* e *helps*, cores, layout e letras), atender a usabilidade universal (considerando a diversidade,

plasticidade e categorias de usuários), oferecer *feedback* informativo (adequado a cada ação, desde simples a complexas), projetar diálogos para produzir fechamento (prevendo sequências de ações divididas em grupos, com início, meio, fim e *feedback* ao final), prevenir erros (desabilitando opções inapropriadas, dando instruções de recuperação em casos de erro, por exemplo), permitir fácil reversão de ações (encorajando a exploração da interface pelo usuário), suportar locus interno de controle (permitindo que peritos se sintam no comando da interface, sem surpresas e sem entradas de dados entediantes) e, por fim, reduzir a carga de memória de curto prazo (evitando a necessidade de memória do usuário entre telas).

Diretrizes como as “Dez heurísticas” de Nielsen podem ser usadas. Trata-se de um conjunto de heurísticas baseado nas seguintes diretrizes e respectivos exemplos de ações (apenas algumas estão citadas, do próprio autor e de outros) para atendê-las:

- a) Visibilidade do estado do sistema: manter o usuário informado sobre o que está acontecendo, por meio de *feedback* adequado e no tempo certo;
- b) Compatibilidade entre o sistema e o mundo real: utilizar conceitos, vocabulário e processos familiares e coerentes ao modelo mental dos usuários;
- c) Controle e liberdade do usuário: fornecer alternativas e “saídas de emergência”, como as possibilidades de desfazer e refazer ações e voltar ao ponto anterior. Outros autores mencionam ainda a necessidade de garantir o controle humano à medida que se aumente a automação;
- d) Consistência e padronização: considerar que palavras, situações e ações semelhantes devem significar conceitos ou operações semelhantes durante todo o percurso pelo sistema. O sistema deve também obedecer eventuais padrões e convenções do ambiente, plataforma ou ferramenta em uso;
- e) Prevenção de erros: evitar que o erro aconteça, informando o usuário sobre as conseqüências de suas ações ou, se possível, impedindo ações que levariam a situações de erro; deve-se buscar ainda desabilitar opções que possam causar erros, ter restrições físicas (onde a própria forma dos objetos

pode restringir movimentos errados), priorizar a seleção ao invés de digitação e possibilitar o uso de preenchimento automático, comandos únicos e macros sempre que possível;

- f) Reconhecimento ao invés de lembrança: tornar objetos, ações e opções visíveis e compreensíveis, bem como orientar as ações do usuário para que ele não tenha que acionar constantemente sua memória para tomar uma ação;
- g) Flexibilidade e eficiência de uso: permitir que os usuários customizem ações freqüentes (uso de teclas de atalhos, máscaras e navegação com “tab” em formulários, dentre outras);
- h) Estética e *design* minimalista: exibir apenas os textos e *design* que o usuário precisa ter acesso;
- i) Ajudar ao usuário para reconhecer, diagnosticar e se recuperar de erros: emitir mensagens de erro em linguagem simples, sem códigos, indicando precisamente o problema e sugerindo ao usuário o que ele precisa fazer como possível solução de contorno. Outros autores mencionam ainda que as mensagens de erro devem ser também específicas (e não genéricas ou obscuras), em tom positivo, cordial (e não hostil ou violento), claro e construtivo (com foco na resolução do problema). O estilo deve ser centrado no usuário e na orientação clara e compreensível de sua ação após o erro;
- j) Ajuda e documentação: manter acessível, com mecanismo de busca, focados no domínio e na tarefa do usuário e demonstrar os passos para a execução da tarefa.

**Evidência Objetiva:** Tipicamente as diretrizes de usabilidade do projeto devem estar documentadas (ou referenciadas) no guia de estilo do projeto, mas podem estar em outro documento do projeto como termo de abertura, plano do projeto, guia de adaptação ou similar. Podem também estar declaradas em um documento do sistema ou produto (no caso de vários projetos de um mesmo sistema ou uma família de produtos) ou em um documento da metodologia de desenvolvimento de *software* (no caso de diretrizes que devem servir a todos os projetos).

**Justificativa:** As técnicas apresentadas nesta atividade estão respaldadas no referencial teórico do presente trabalho e evidenciadas nas seções 4.1.5 por Nielsen (1993), Shneiderman e Plaisant (2010) e Nielsen e Loranger (2007), bem como nos seguintes trechos específicos desta dissertação:

- a) Diretrizes Gerais são apresentadas na seção 4.1.5 por Nielsen e Loranger (2007), Shneiderman e Plaisant (2010) e NCI (2008).
- b) Diretrizes de plataformas, ambiente operacional e *hardware* são apresentadas nas seções 3.2.5 pela ISO/IEC 15504-5 (2006) e 4.1.4 por Shneiderman e Plaisant (2010).
- c) Diretrizes de mapeamento e de *affordance* são apresentadas na seção 2.3 por Preece, Rogers e Sharp (2005).
- d) Diretrizes específicas para ambiente *web* são apresentadas na seção 4.1.5 por Nielsen e Loranger (2007), Shneiderman e Plaisant (2010), SUN (2008), NCI (2008), WAI (2008) e Lynch e Horton (2008).
- e) Diretrizes de navegação e arquitetura de informação são apresentadas na seção 4.1.5 por Nielsen e Loranger (2007).
- f) Diretrizes de textos são apresentadas na seção 4.1.5 por Nielsen e Loranger (2007).
- g) Diretrizes de apresentação dos elementos de uma tela ou página são apresentadas na seção 4.1.5 por Nielsen e Loranger (2007).
- h) Diretrizes de organização de layout são apresentadas na seção 4.1.5 por Smith e Mosier (1986).
- i) Diretrizes de busca e visualização das informações são apresentadas na seção 4.1.5 por Shneiderman e Plaisant (2010).
- j) Diretrizes de obtenção da atenção do usuário são apresentadas na seção 4.1.5 por Wickens e Hollands (2000).

- k) Diretrizes do uso adequado de cores são apresentadas na seção 4.1.5 por Nielsen e Loranger (2007) e Shneiderman e Plaisant (2010)
- l) As Oito Regras de Ouro de Shneiderman e Plaisant são apresentadas na seção 4.1.5 por Shneiderman e Plaisant (2010).
- m) As Dez Heurísticas de Nielsen são apresentadas na seção 4.2.4.3.1 por Nielsen (1993), Norman (1983), Preece, Rogers e Sharp (2005) e Shneiderman e Plaisant (2010).

### 6.1.5 Atividade 5

**Atividade:** Monitorar o uso e efetividade das diretrizes de usabilidade.

**Grupo de Processo:** Monitoramento e Controle.

**Objetivo:** Garantir que os projetos façam uso das diretrizes de usabilidade estabelecidas pela organização para o projeto.

**Técnicas:** Auditoria de processos e de produtos. Na auditoria é realizada uma verificação se o processo contempla atividades que têm o acesso às diretrizes estabelecidas (para saber em que documentos as diretrizes podem estar registradas, ver item de evidências objetivas mencionadas na atividade “estabelecer diretrizes de usabilidade” deste guia) e se os produtos estão efetivamente respeitando as diretrizes (desde produtos intermediários como protótipos e maquetes até o produto final).

A fim de manter uma institucionalização adequada das diretrizes por todos os envolvidos nos projetos, deve haver treinamento das equipes em relação às diretrizes. A fim de manter a base de diretrizes atualizada e coerente, a organização deve possuir processos que permitam revisão periódica das diretrizes, sessões de discussão sobre sua aplicabilidade e efetividade e o incentivo a iniciativas de pesquisa, teste e efetiva implementação de novas diretrizes (a partir de novas idéias e/ou tecnologias).

**Evidência Objetiva:** Registro de auditorias de processo (ata, *checklist* ou similar) e de produto (apontamentos nas avaliações intermediárias e final sobre a aderência da interface às diretrizes). Registro das atualizações das diretrizes.

**Justificativa:** As técnicas apresentadas nesta atividade estão respaldadas no referencial teórico do presente trabalho e evidenciadas nas seções 3.2.1 pela NBR ISO/IEC 9126-1 (2003), 3.2.2 pela NBR 9241-11 (2002), 3.2.4 pela ISO 13407 (1999), 3.2.5 pela ISO/IEC 15504-5 (2006), 3.2.6 pela ISO/TR 18529 (2000), 3.4 por Mayhew (1999) e por Preece, Rogers e Sharp (2005), 3.3.2 por Chrissis, Konrad e Shrum (2003) e 4.1.5 por Shneiderman e Plaisant (2010).

### 6.1.6 Atividade 6

**Atividade:** Revisar a metodologia de desenvolvimento de *software* da organização, sob a perspectiva da usabilidade.

**Grupo de Processo:** Organizacional.

**Objetivo:** Garantir que há o devido foco à usabilidade no processo definido pela metodologia de desenvolvimento de *software* da organização. Os processos de usabilidade devem estar claramente documentados e devem ser seguidos por todos os projetos.

**Técnicas:** Deve ser realizado um mapeamento por especialistas para averiguar se os processos relacionados à usabilidade (e as atividades propostas no presente guia) estão inseridos nos processos da metodologia de desenvolvimento de *software* da organização (processo, procedimento, manual ou guia). Deve-se ter claramente identificados na metodologia e no ciclo de vida do projeto em quais pontos se encontram os processos (e respectivas atividades) relacionados à garantia da usabilidade propostas no presente guia, que sejam aplicáveis à organização e coerentes com o projeto. O ideal é que se faça um registro desse mapeamento, indicando onde as práticas propostas no presente guia se encontram cobertas na metodologia. A falta de cobertura de algum processo pode gerar uma ação corretiva na organização (inclusão do processo na metodologia) ou um registro no projeto (justificativa para a não adoção de uma prática). Além das atividades previstas no presente guia, outras práticas devem ser consideradas.



Exemplos de práticas incluem, mas não se limitam às mencionadas nos parágrafos a seguir.

O atendimento às questões éticas junto aos envolvidos deve ser garantido nos processos voltados à usabilidade.

A abordagem centrada no usuário deve ser perseguida. A metodologia deve prever ações que façam com que o usuário participe ativamente da construção e evolução da interface ao longo do ciclo de vida do projeto, sendo que seu envolvimento deve estar refletido no plano do projeto (ou similar). A ISO/TR 18529 (2000) reforça a necessidade de incorporação dessa abordagem na metodologia, por meio do processo HCD.2 (planejamento e gerenciamento dos processos de desenvolvimento centrados no humano).

As diretrizes de usabilidade da organização devem ser periodicamente revistas. Ver técnicas mencionadas na atividade “Estabelecer diretrizes de usabilidade”.

O processo de solicitações de mudanças em projeto deve considerar as questões de usabilidade. As análises de impacto devem ter um item claro quanto ao monitoramento dos requisitos de usabilidade. Pode-se utilizar, por exemplo, um processo organizacional definido e aderente ao processo SUP.10 (gestão de mudança) da ISO/IEC 15504-5 (2006) ou aos processos HCD.7.1 (gerenciar as mudanças) e HCD.7.2 (determinar o impacto na organização e nos *stakeholders*) da ISO/TR 18529 (2000).

Outros processos que podem ter impacto direto ou indireto na usabilidade podem ser objeto de atenção durante as revisões da metodologia, como alguns processos aderentes à ISO/IEC 12207 (2009), como gestão de portfólio de projetos, gestão de configuração de *software*, gestão de documentação de *software*, gestão de reuso de ativos e resolução de problemas de *software*, entre outros.

**Evidência Objetiva:** Registro da revisão da metodologia (ata, *checklist* ou similar).

**Justificativa:** As técnicas apresentadas nesta atividade estão respaldadas no referencial teórico do presente trabalho e evidenciadas nas seções 3.2.3 pela NBR ISO/IEC 12207 (2009), 3.2.5 pela ISO/IEC 15504-5 (2006), 3.2.6 pela ISO/TR 18529

(2000), 3.4 por Mayhew (1999), pela ISO/IEC 15504-5 (2006) e por Preece, Rogers e Sharp (2005) e 4.2.4.5 por Avelino (2005), bem como nos seguintes trechos específicos desta dissertação:

- a) Questões éticas são apresentadas na seção 2.1.2 por Baranauskas e Melo (2006).
- b) A abordagem centrada no usuário é apresentada no capítulo 1 e nas seções 3.2.4 pela ISO 13407 (1999) e 2.1.2 por Preece, Rogers e Sharp (2005).
- c) As diretrizes são apresentadas nas referências já mencionadas anteriormente no item “justificativa” da atividade “Estabelecer diretrizes de usabilidade”.
- d) A questão de gestão de mudanças é apresentada na introdução do capítulo 4 referenciando-se à ISO/TR 18529 (2000) e ISO/IEC 15504-5 (2006) e nas respectivas seções de cada uma destas normas.

### 6.1.7 Atividade 7

**Atividade:** Monitorar se os projetos seguem os processos de usabilidade da metodologia de desenvolvimento de *software* da organização.

**Grupo de Processo:** Monitoramento e Controle.

**Objetivo:** Garantir que os projetos façam uso dos processos de usabilidade estabelecidos pela metodologia de desenvolvimento de *software* da organização.

**Técnicas:** Auditoria de processos e de produtos. O plano de projeto (ou documento similar aplicável) deverá demonstrar que o ciclo de vida utilizado pelo projeto contempla as atividades de usabilidade definidas na metodologia da organização. No caso de exceções, como um projeto específico que segue uma metodologia diferenciada em que atividades de usabilidade não sejam aplicáveis, deve-se registrar o motivo no próprio plano de projeto, guia de estilo do projeto, guia de adaptação do projeto ou artefato similar aplicável. Outros documentos de gestão do projeto (cronogramas, declaração de escopo, matriz de risco, análises de impacto de mudanças, entre outros) e artefatos produzidos durante o projeto (protótipos, registros de teste, entre outros) devem refletir o cumprimento dos processos e

atividades referentes à usabilidade, estabelecidas na metodologia. No caso de organizações que, por exemplo, utilizam o modelo CMMI (CHRISSIS, KONRAD e SHRUM, 2003), podem-se utilizar as mesmas auditorias previstas para cobrir os processos de PPQA – Garantia de Qualidade de Processos e Produtos, enquanto as organizações que utilizam a ISO/IEC 15504-5 (2006) podem-se utilizar as auditorias previstas para cobrir o processo SUP.1 (garantia da qualidade), SUP.2 (verificação) e SUP.5 (auditoria). Ações corretivas devem ser tomadas quando o uso do processo não é evidenciado.

**Evidência Objetiva:** Registro de auditorias de processo (ata, *checklist* ou similar), apontamentos nas avaliações de produto, solicitações de mudanças relacionadas à interface, registros de ações preventivas e/ou corretivas nos casos de desvios e não-conformidades ao processo definido.

**Justificativa:** As técnicas apresentadas nesta atividade estão respaldadas no referencial teórico do presente trabalho e evidenciadas nas seções 2.1.2 por Preece, Rogers e Sharp (2005), 3.2.2 pela NBR 9241-11 (2002), 3.2.3 pela NBR ISO/IEC 12207 (2009), 3.2.5 pela ISO/IEC 15504-5 (2006) e 3.3 por Chrissis, Konrad e Shrum (2003).

### 6.1.8 Atividade 8

**Atividade:** Dimensionar o esforço destinado às atividades voltadas à usabilidade.

**Grupo de Processo:** Planejamento.

**Objetivo:** Garantir estimativas mais realistas e precisas para o projeto, ao considerar nesta estimativa o esforço necessário (em horas ou percentual do projeto) para cumprir os processos e respectivas atividades voltados à usabilidade.

**Técnicas:** As características e necessidades de usabilidade devem estar refletidas nas atividades, requisitos e riscos do projeto e, portanto, fazer parte do dimensionamento destes.

**Evidência Objetiva:** Justificativa documentada do dimensionamento e registro de divulgação dessa estimativa aos responsáveis pelo dimensionamento do projeto. O

plano de projeto pode documentar a estimativa de recursos necessários, análise de custo-benefício e respectiva aprovação.

**Justificativa:** As técnicas apresentadas nesta atividade estão respaldadas no referencial teórico do presente trabalho e evidenciadas no capítulo 1 por Shneiderman e Plaisant (2010), seções 2.1.1 por Filho (2005), 3.3.1 pelo PMBOK (2008), 3.3.2 por Chrissis, Konrad e Shrum (2003), 3.2.5 pela ISO/IEC 15504-5 (2006) e 3.4 por Mayhew (1999).

### 6.1.9 Atividade 9

**Atividade:** Identificar o perfil do usuário, sob a perspectiva da usabilidade.

**Grupo de Processo:** Organizacional.

**Objetivo:** Garantir que o projeto esteja considerando algumas características dos usuários que tenham relevância para a qualidade da usabilidade, gerando subsídios para as tomadas de decisão que envolvam impacto na garantia da usabilidade.

**Técnicas:** Identificar categorias, características e outros atributos relacionados ao perfil dos usuários da interface que são relevantes aos aspectos da usabilidade da interface. Exemplos de técnicas incluem, mas não se limitam às apresentadas a seguir:

- a) Questionários: exemplos de recomendações incluem, mas não se limitam a: atenção na categorização e seleção de amostra dos usuários, elaboração de perguntas com clareza e especificidade, escala adequada, ordem e prioridade adequadas das perguntas, instruções de preenchimento precisas e completas, quantidade reduzida de perguntas, realização de um piloto, facilitação para envio das respostas e um processo adequado para registro, resumo e análise das informações recebidas.
- b) Entrevistas: exemplos de recomendações incluem, mas não se limitam a: definição clara do tipo (estrutura, não estruturada ou semi-estruturada), do meio (presencial, *online*, por telefone) treinamento do entrevistador e entrevistados, agendamento das entrevistas, roteiro (*script*), captura

adequada (anotação, gravação em áudio e/ou vídeo, registro de características do ambiente (físico, sociocultural, contexto do trabalho), planejamento das perguntas (questões curtas, claras, sem ambigüidade ou tendenciosas), registro das informações obtidas (sem alterações ou correções e, quando aplicável, com garantia do sigilo e não-atribuição).

- c) Grupos de foco: devem ser compostos por uma amostra representativa dos usuários e conduzidos por um facilitador com a devida experiência.
- d) Observação contextual ou estudo etnográfico: exemplos de métodos incluem, mas não se limitam a: método da coerência e *design* contextual. Requer um treinamento sobre etnografia aos envolvidos. Para a condução são aplicáveis algumas das técnicas mencionadas sobre entrevistas como as questões de agendamento prévio, captura adequada e registro das informações obtidas.
- e) Usabilidade universal: tratamento adequado das necessidades e requisitos de usabilidade quando há uma especificidade ou grande diversidade envolvida de grupos de usuários (culturas, regiões, estilos e tipos psicológicos, nível de conhecimento sobre a interface, entre outras questões), plataformas (ambientes operacionais, *hardware*, dispositivos de entrada). Esse item está contemplado também como uma diretriz na atividade “estabelecer diretrizes de usabilidade” deste guia, já que as referências bibliográficas endereçam essa questão tanto na obtenção do perfil de usuário como no estabelecimento de diretrizes.

As técnicas recomendadas para entrevistas e observações são válidas para o uso destas técnicas em outras atividades deste guia, como “Identificar as tarefas e contexto de uso, sob a perspectiva da usabilidade”, “Executar avaliações iterativas da interface” e “Realizar sessão para lições aprendidas e *feedback* do usuário”.

**Evidência Objetiva:** Registro da documentação. As informações registradas sobre o perfil do usuário podem estar documentadas no guia de estilo do projeto ou em outro documento aplicável como o plano de projeto, o registro de *stakeholders* ou similar. Outros registros incluem, mas não se limitam a: questionário e/ou pesquisa preenchidos, registro de entrevista, ata de reunião com definição do perfil do usuário

ou informações extraídas de artefatos como termo de abertura do projeto, registro de *stakeholders*, premissas, restrições e/ou fatores críticos do projeto, declaração de escopo, entre outros.

**Justificativa:** As técnicas apresentadas nesta atividade estão respaldadas no referencial teórico do presente trabalho e evidenciadas nas seções 3.2.4 pela ISO 13407 (1999), 3.4 por Holtzblatt et al. (2005) e 4.1.1 por Dix et al. (2004), pela NBR ISO/IEC 12207 (2009) e pela ISO/IEC 15504-5 (2006), bem como nos seguintes trechos específicos desta dissertação:

- a) Questionários são apresentados na seção 4.1.1 por Preece, Rogers e Sharp (2005), Mayhew (1999) e Shneiderman e Plaisant (2010).
- b) Entrevistas são apresentadas nas seções 4.1.1 por Preece, Rogers e Sharp (2005), Mayhew (1999), Shneiderman e Plaisant (2010) e 4.1.2 por Preece, Rogers e Sharp (2005).
- c) Grupos de foco são apresentados na seção 4.1.1 por Preece, Rogers e Sharp (2005).
- d) Observação contextual ou estudo etnográfico é apresentado na seção 4.1.1 por Preece, Rogers e Sharp (2005).
- e) Usabilidade universal é apresentada nas seções 4.1.1 por Shneiderman e Plaisant (2010) e pela NBR 9241-11 (2002) e 4.1.4 por Shneiderman e Plaisant (2010).

#### **6.1.10 Atividade 10**

**Atividade:** Identificar as tarefas e contexto de uso, sob a perspectiva da usabilidade.

**Grupo de Processo:** Planejamento.

**Objetivo:** Garantir que o projeto esteja considerando algumas características das tarefas e contexto de trabalho que tenham relevância para a qualidade da usabilidade, gerando subsídios para as tomadas de decisão que envolvam impacto na garantia da usabilidade.

**Técnicas:** Identificar características e atributos relacionados às tarefas que devem ser cobertas pela interface bem como os contextos de uso aos quais estão inseridas, que tenham relevância na usabilidade da interface. As atividades podem ser levantadas por meio das mesmas técnicas já apresentadas para a obtenção do perfil do usuário (ver técnicas mencionadas na atividade “identificar o perfil do usuário, sob a perspectiva da usabilidade”). Em alguns casos, esta atividade é conduzida em concomitância ao levantamento ou revisão dos requisitos (ver técnicas mencionadas na atividade “identificar necessidades e requisitos de usabilidade”). Além disso, é necessária a identificação e documentação dos atores-chave e casos de uso. Pode-se utilizar a abordagem tradicional de caso de uso ou a essencial. Outras técnicas alternativas podem ser usadas como a Análise Hierárquica de Tarefas (AHT) ou a elaboração de cenários de tarefas.

**Evidência Objetiva:** Registro da documentação. As informações registradas sobre as atividades podem estar documentadas em diagramas de caso de uso (ou similar), especificações do contexto de uso e/ou em algum modelo que identifique como as tarefas e artefatos se organizam (logicamente e em fluxo) em uma visão de hierarquia. Outros exemplos de registros incluem, mas não se limitam a: questionários e/ou pesquisas preenchidos, registros de entrevista, atas de reunião com detalhamento das atividades levantados junto ao usuário, entre outros.

**Justificativa:** As técnicas apresentadas nesta atividade estão respaldadas no referencial teórico do presente trabalho e evidenciadas nas seções 3.2.2 pela NBR 9241-11 (2002), 3.2.4 pela ISO 13407 (1999), 4.1.2 por Jacobson et al. (1992), Constantine e Lockwood (1999) e 4.1.4 por Shneiderman e Plaisant (2010).

Outras referências aplicáveis a esta atividade podem ser encontradas no item “justificativa” da atividade “identificar o perfil do usuário, sob a perspectiva da usabilidade”.

### **6.1.11 Atividade 11**

**Atividade:** Estabelecer um guia de estilo a ser usado pelo projeto.

**Grupo de Processo:** Planejamento.

**Objetivo:** Garantir que exista um documento formal de usabilidade para o projeto, progressivamente elaborado ao longo do ciclo de vida e que reúna todas as informações necessárias para se orientar os envolvidos quanto às diretrizes, processos, objetivos e demais detalhes da usabilidade.

**Técnicas:** O guia de estilo (ou outro documento aplicável) pode ser criado especificamente para o projeto, adaptado de outro projeto ou referenciado a um guia já existente que possa ser utilizado e deve possuir a devida aprovação do seu uso. O guia é o resultado da consolidação de informações de usabilidade como as diretrizes (ver técnicas da atividade “estabelecer diretrizes de usabilidade”), os objetivos (ver técnicas da atividade “estabelecer objetivos claros de usabilidade”) e os processos (ver técnicas da atividade “revisar a metodologia de desenvolvimento de *software* da organização, sob a perspectiva da usabilidade”). Os processos de usabilidade que serão utilizados no projeto podem estar opcionalmente descritos em um documento do projeto, como no plano de projeto, no guia de adaptação ou outro aplicável. O guia pode conter ainda outras informações pertinentes, extraídas do perfil de usuários, dos requisitos de usabilidade, das tarefas e contexto de uso e das próprias interfaces e avaliações de interfaces realizadas.

**Evidência Objetiva:** Documento de guia de estilo (ou outro aplicável), registro de análise e aprovação do uso do guia para o projeto específico (especialmente se for usado um guia organizacional ou de um sistema já existente para vários projetos).

**Justificativa:** As técnicas apresentadas nesta atividade estão respaldadas no referencial teórico do presente trabalho e evidenciadas nas seções 3.2.4 pela ISO 13407 (1999), introdução do capítulo 4 por Chrissis, Konrad e Shrum (2003) e seção 4.2.8 por Mayhew (1999). Além disso, como o guia de estilo é o documento que formalmente consolida as informações de diretrizes e objetivos de usabilidade para um projeto, se aplicam também as mesmas justificativas das atividades “estabelecer diretrizes de usabilidade” e “estabelecer objetivos claros de usabilidade”.

### **6.1.12 Atividade 12**

**Atividade:** Monitorar se os projetos seguem um guia de estilo estabelecido.

**Grupo de Processo:** Monitoramento e Controle.



**Objetivo:** Garantir que o guia de estilo estabelecido para o projeto está sendo respeitado.

**Técnicas:** Auditoria de processos e de produtos. Na auditoria é realizada uma verificação se o processo contempla o acesso ao guia de estilo e seu uso efetivo e se os produtos estão efetivamente respeitando o guia de estilo estabelecido (desde produtos intermediários como protótipos e maquetes até o produto final).

**Evidência Objetiva:** Registro de auditorias de processo (ata, *checklist* ou similar) e de produto (apontamentos nas avaliações intermediárias e final sobre a aderência da interface ao guia de estilo). Registro das atualizações do guia de estilo ao longo do projeto (o guia é progressivamente detalhado ao longo do ciclo de vida) e solicitações de mudança que impactem os itens que compõem o guia de estilo.

**Justificativa:** As técnicas apresentadas nesta atividade estão respaldadas no referencial teórico do presente trabalho e evidenciadas nas seções 3.2.1 pela NBR ISO/IEC 9126-1 (2003), 3.2.2 pela NBR 9241-11 (2002), 3.2.4 pela ISO 13407 (1999), 3.2.5 pela ISO/IEC 15504-5 (2006), 3.2.6 pela ISO/TR 18529 (2000), introdução do capítulo 4 por Chrissis, Konrad e Shrum (2003), seção 4.2.4.5 por Shneiderman e Plaisant (2010) e 4.2.8 por Mayhew (1999).

### **6.1.13 Atividade 13**

**Atividade:** Identificar necessidades e requisitos de usabilidade.

**Grupo de Processo:** Planejamento

**Objetivo:** Garantir que as necessidades de usabilidade sejam identificadas, transformadas em requisitos e que estes requisitos sejam incorporados à documentação de requisitos do projeto.

**Técnicas:** Elicitação de requisitos, análise de requisitos, revisão de requisitos, revisão a partir dos requisitos de segurança.

**Evidência Objetiva:** Documentação dos requisitos, declaração de escopo ou similar.

**Justificativa:** As técnicas apresentadas nesta atividade estão respaldadas no referencial teórico do presente trabalho e evidenciadas nas seções 2.1.2 por Preece, Rogers e Sharp (2005), 3.2.1 pela NBR ISO/IEC 9126-1 (2003), 3.2.4 pela ISO 13407 (1999) e 3.3 por Barbosa, Furtado e Gomes (2006), bem como nos seguintes trechos específicos desta dissertação:

- a) Elicitação de requisitos é apresentada na seção 4.1.2 por Avelino (2005).
- b) Análise de requisitos é apresentada no capítulo 4 por Mayhew (1999, 2008) e também especificamente na seção 4.1.1 pela ISO/TR 18529 (2000).
- c) Revisão de requisitos é apresentada nas seções 4.1.1 por Preece, Rogers e Sharp (2005) e 4.1.4 por Mayhew (1999, 2008).
- d) Revisão a partir dos requisitos de segurança é apresentada na seção 4.1.1 por Avelino (2005).

#### **6.1.14 Atividade 14**

**Atividade:** Revisar requisitos de usabilidade ao longo da construção da interface

**Grupo de Processo:** Monitoramento e Controle

**Objetivo:** Garantir que os requisitos de usabilidade definidos sejam detalhados, refinados e aprovados de forma iterativa e progressiva ao longo do projeto.

**Técnicas:** Ver atividade “Identificar necessidades e requisitos de usabilidade” (que trata da identificação dos requisitos) e atividade “Revisar a metodologia de desenvolvimento de *software* da organização, sob a perspectiva da usabilidade” (que aborda o tratamento adequado aos requisitos de usabilidade por meio de um processo de gestão de mudanças).

**Evidência Objetiva:** Registros de revisão e aprovação da documentação de requisitos ao longo do projeto.

**Justificativa:** As técnicas apresentadas nesta atividade estão respaldadas no referencial teórico do presente trabalho e evidenciadas nas referências já mencionadas anteriormente no item “justificativa” das atividades “identificar

necessidades e requisitos de usabilidade” e “revisar a metodologia de desenvolvimento de *software* da organização, sob a perspectiva da usabilidade”.

### 6.1.15 Atividade 15

**Atividade:** Construir versões iterativas, intermediárias e interativas da interface.

**Grupo de Processo:** Execução.

**Objetivo:** Garantir que construção da interface siga um processo que permita uma abordagem centrada no usuário e uma evolução gradativa dos requisitos de usabilidade, coerente com o processo natural de detalhamento progressivo do escopo ao longo do ciclo de vida do projeto.

**Técnicas:** Esta atividade visa garantir que o processo de construção da interface seja baseado em uma execução de forma gradativa por meio de ciclos de iteração, com detalhamento progressivo, resultando em versões intermediárias de *design* que permitam a visualização, a manipulação e a conseqüente validação da interface pela interação com o usuário. *Designs* alternativos podem ainda ser apresentados ao usuário durante o desenvolvimento, para auxiliar no detalhamento, *redesigns* e validação das soluções técnicas propostas.

A seguir são apresentadas as sub-atividades e respectivas técnicas desta atividade, elencadas a partir do próximo parágrafo. Essas atividades seguem uma sequência natural, baseada na evolução do projeto (*design*), partindo de um nível de abstração maior da interface (que tipicamente ocorre nas primeiras atividades de desenvolvimento, testes e prototipação) e seguindo para um nível cada vez mais detalhado e evoluído da interface (que tipicamente ocorre nas atividades finais). As sub-atividades ocorrem em ciclos de iteração, sendo cada ciclo marcado por uma avaliação do *design*.

Uma análise de melhorias no processo atual deve ser feita primeiramente. Pode-se avaliar a documentação de análise de contexto das tarefas e verificar se há necessidade de eventual reengenharia no processo atual, que permita maior usabilidade no sistema a ser desenvolvido.

A definição da priorização e hierarquia de apresentação da interface é uma sub-atividade a ser realizada. Em um modelo conceitual orientado a produto, produtos primários devem ser exibidos nos primeiros níveis de interação (podem ser minimizados e ter janelas concorrentes em paralelo) enquanto os secundários devem ser exibidos em caixas de diálogo ou menus posteriores (não podem ser minimizados, ocultados ou fechados sem que a ação seja concluída). Já em um modelo orientado a processo, deve-se seguir a hierarquia dos processos, simplificando o entendimento dos passos de navegação por parte do usuário. Para aplicações em que o usuário deve acessar informações de diversas fontes ou janelas, pode-se usar técnicas como a de janelas múltiplas coordenadas (*coordinated windows*) ou de “salas” de janelas (*room approach*).

Uma identificação das principais telas e caminhos de navegação deve ser realizada. Nesta sub-atividade, identificam-se os possíveis diferentes caminhos que podem ser seguidos pelo usuário, verificando se a hierarquia da aplicação está sendo respeitada. Nesse estágio, pode-se documentar à mão, sem necessidade de ferramentas.

Técnicas alternativas para a obtenção desta primeira versão do *design* podem ser aplicáveis. Exemplos de técnicas incluem mas não se limitam ao uso de *design* participativo (com uso de ferramentas de *software* ou técnicas em papel como maquetes, PICTIVE ou CARD) e de paradigmas de interação (GUI ou WIMP, computação ubíqua, computação pervasiva, computação “vestível”, realidade aumentada e 3D. Outra técnica é o *design* alternativo, em que soluções técnicas alternativas de projeto são apresentadas ao usuário (pode-se usar como referência para essa técnica a área de processo de Solução Técnica do CMMI).

Um esboço inicial do *design* deve ser preparado. Para esta sub-atividade, deve-se selecionar as funcionalidades que necessitam de um esboço inicial (maquete ou protótipo) conforme um critério definido (funcionalidades novas, que sofreram alterações, de missão crítica, de relevância ou outro critério). Detalhes de telas não são o foco do esboço nesse estágio inicial de evolução do *design*. Maquetes ou protótipos devem ser construídos (em papel ou por meio de ferramentas de *software*) para as funcionalidades selecionadas.

A avaliação inicial do *design* é a próxima sub-atividade (tratada em detalhes na atividade “executar avaliações iterativas da interface ao longo de sua construção” deste guia). Após esta avaliação é feita uma verificação de atingimento dos objetivos de usabilidade definidos que são pertinentes e aplicáveis a esta versão. Se os objetivos de usabilidade são atingidos o guia de estilo é atualizado com base nos resultados obtidos até então. Caso contrário, um novo ciclo de iteração é iniciado.

A padronização do *design* de telas é a próxima sub-atividade. Exemplos incluem, mas não se limitam a padrões de uso para os controles como botões e caixas de diálogo (campos obrigatórios e opcionais, editáveis ou não, entre outros), caixas de mensagens (erro, alerta, *status*, posicionamento, formato, , entre outros), *feedback* (tarefa em andamento ou concluída, opção selecionada/ativada ou não, “faróis” , entre outros), indicadores de *status* ou progresso (mensagens, barras, entre outros).

A prototipação é a próxima sub-atividade a ser realizada. Deve-se selecionar quais as funcionalidades necessitam de protótipo, conforme um critério definido (funções mais usadas, críticas, com maior potencial de problemas, funções com várias sequências, ou outros critérios) e preparar um esboço esboço *design* detalhado das telas reais incluindo janelas, controles de ações, caixas de diálogo, mensagens, menus, interações e caminhos, bem como rótulos, disposição e organização dos mesmos. A partir do esboço são construídos os protótipos, que podem ser de baixa fidelidade (*storyboards*, esboços, maquetes ou fichas) ou de alta fidelidade (por ferramenta de *software*, e que podem ser evolutivos ou descartáveis).

Uma avaliação intermediária do *design* deve ser então realizada (tratada em detalhes na atividade “executar avaliações iterativas da interface ao longo de sua construção” deste guia). Uma boa prática é utilizar novos testadores, diferentes dos utilizados na avaliação inicial do *design*, sempre que possível e aplicável. Após a avaliação é feita uma verificação de atingimento dos objetivos de usabilidade definidos que são pertinentes e aplicáveis a esta versão. Quando os objetivos de usabilidade são atingidos o guia de estilo é atualizado com base nos resultados obtidos até então. Caso contrário, um novo ciclo de iteração é iniciado a partir da sub-atividade subsequente à ultima avaliação do *design*.

A conclusão do *design* detalhado é então a próxima sub-atividade. Além dos elementos de *design* já descritos nas atividades anteriores, todos os demais detalhes devem ser concluídos até esse estágio, como, por exemplo, definições de menus e de dispositivos de entrada. Exemplos de definições relacionadas aos menus podem ser quanto aos seus tipos (único ou múltiplo, *pull-down/toolbar/ribbon*), suas características (lista pequena ou longa, a ser exibida em *display* pequeno ou grande, menu em áudio), sua exibição (explícita ou implícita como os *embedded* menus e *hotlinks*), complexidade (estáticos, dinâmicos, simultâneos, organizados em árvores) e sua organização e acesso a partir de outros locais, como em mapas de *websites*, por exemplo. Quanto aos dispositivos, devem ser definidos quais serão usados. Exemplos incluem mas não se limitam a: teclado (considerando teclas de atalho e aceleradores), mouse (detalhando o uso de *single-clicks*, *double-clicks*, *drag-and-drops*), tela sensível ao toque (*touchscreen* ou *touchpad*), voz, caneta (*light pen* ou *smart pen*), *trackball*, *trackpoint*, controles para mãos (luvas) ou pés, joystick, sensores, papel digital, teclados diferenciados (coletores ou *smartphones*), mesas (*surface*) e/ou outros dispositivos compatíveis com o aplicativo criado.

A avaliação final do *design* é a última sub-atividade desta atividade. É tratada em detalhes na atividade “executar avaliações iterativas da interface ao longo de sua construção” deste guia e já mencionada nas sub-atividades anteriores de avaliação do *design* inicial e intermediária. Após a avaliação é feita uma verificação de atingimento dos objetivos de usabilidade definidos que são pertinentes e aplicáveis a esta versão e se todas as funcionalidades foram atendidas. Se os objetivos de usabilidade são atingidos o guia de estilo é atualizado com base nos resultados obtidos até então e o *design* é encerrado. Caso contrário um novo ciclo de iteração é iniciado a partir da sub-atividade subsequente à última avaliação do *design*. Caso as funcionalidades tenham sido atendidas, prossegue-se para a implantação do sistema e para a condução da sessão de lições aprendidas e *feedback* do usuário, caso contrário, pode ser necessário um novo ciclo de iteração a partir do início da construção.

**Evidência Objetiva:** Registro que comprove a elaboração dos *designs* dos produtos gerados (intermediários e final), de forma iterativa e com a participação do usuário.

Registro de utilização efetiva desses produtos gerados (protótipo, maquete ou qualquer similar mencionado nas técnicas) por parte do usuário.

**Justificativa:** As técnicas apresentadas nesta atividade estão respaldadas no referencial teórico do presente trabalho e evidenciadas na seção 3.2.4 pela ISO 13407 (1999) e nas sub-seções da seção 4.2 por Mayhew (1999, 2008) e Shneiderman e Plaisant (2010), bem como nos seguintes trechos específicos desta dissertação:

- a) Análise de melhorias no processo atual é apresentada na seção 4.2.1 por Mayhew (1999, 2008).
- b) Definição da priorização e hierarquia de apresentação da interface é apresentada na seção 4.2.2 por Mayhew (1999).
- c) Identificação das principais telas e caminhos de navegação é apresentada na seção 4.2.2 por Mayhew (1999) e por Preece, Rogers e Sharp (2005).
- d) Técnicas alternativas para a obtenção da primeira versão do *design* são apresentadas nas seções 3.3 por Chrissis, Konrad e Shrum (2003) e 4.2.2 por Mayhew (1999), Preece, Rogers e Sharp (2005) e Shneiderman e Plaisant (2010).
- e) Prototipação é apresentada na seção 2.1.2 e capítulo 4 por Preece, Rogers e Sharp (2005) e Mayhew (1999) e nas seções 3.3 por Barbosa, Furtado e Gomes (2006), 3.4 por Cybis, Betiol e Faust (2007), Holtzblatt et al. (2005) e 4.2.6 por Cybis, Betiol e Faust (2007) e Preece, Rogers e Sharp (2005).
- f) *Storyboarding* é apresentado nas seções 3.3 por Barbosa, Furtado e Gomes (2006) e 3.4 por Holtzblatt et al. (2005).

#### **6.1.16      Atividade 16**

**Atividade:** Executar avaliações iterativas da interface ao longo de sua construção.

**Grupo de Processo:** Execução.

**Objetivo:** Garantir, por meio de avaliações formais, que o *design* esteja aderente às necessidades, requisitos e objetivos de usabilidade do usuário, do projeto e da organização.

**Técnicas:** As avaliações são realizadas durante a construção da interface, ao longo do ciclo de vida, nos produtos intermediários e finais. A técnica recomendada, portanto, é a de avaliação formativa (ou construtiva) e não somativa (ou conclusiva) que são realizadas em produtos já terminados. As avaliações ocorrem ao longo da construção, conforme previsto nas sub-atividades da atividade “construir versões iterativas, intermediárias e interativas da interface” do presente guia.

Exemplos de algumas decisões que devem ser tomadas quanto às avaliações incluem, mas não se limitam às destacadas nos próximos parágrafos.

A escolha do método de avaliação é uma primeira decisão a ser tomada. Pode-se usar o método analítico (não envolve usuários, mas sim especialistas que avaliam a interface) ou empírico (envolve usuários nos testes da interface, normalmente em ambientes controlados como laboratórios). Exemplos de tipos de avaliação analítica incluem: avaliação heurística, percurso cognitivo, percurso pluralista ou pluralístico, conformidade com diretrizes e padrões e inspeções de consistência, padrões e formais.

A escolha do método de coleta dos dados é outra decisão a ser tratada. Podem ser utilizadas técnicas como a coleta de opinião de usuários (por meio de questionários, entrevistas, avaliações “rápidas”, *wikis*, *newsgroups*), observação direta usuários (por meio de etnografia, anotações do observador, gravação de vídeo e/ou áudio, laboratórios) ou indireta dos usuários (ferramentas de *software*, *logs*, gravações de interações, monitoramento remoto). Algumas destas técnicas são mencionadas na atividade “identificar o perfil do usuário, sob a perspectiva da usabilidade” deste guia.

A seguir são apresentados nos próximos parágrafos diversas técnicas que podem ser utilizadas para a avaliação iterativa.

Questionários do tipo “*user survey*” podem ser usados como técnica de avaliação. Exemplos de modelos prontos e padronizados incluem, mas não se



limitam a:, QUIS – *Questionnaire for User Interface Satisfaction*, PUEU – *Perceived Usefulness and Ease Use*, NAU – *Nielsen's Attributes of Usability*, NHE – *Nielsen's Heuristic Evaluation*, CSUQ – *Computer System Usability Questionnaire*, ASQ – *After-Scenario Questionnaire*, PHUE – *Practical Heuristics for Usability Evaluation* e PUTQ – *Purdue Usability Testing Questionnaire*.

A análise preditiva é outra técnica que pode ser utilizada. Por meio desta técnica os avaliadores verificam os dados coletados de especialistas e tentam prever que tipo de problemas os usuários enfrentarão (por meio de inspeção da interface ou técnicas de modelagem), o que permite a obtenção de medidas de desempenho sem testes. Um exemplo desta técnica é a GOMS (*Goals, Operators, Methods, Selection Rules*), que consiste na decomposição de metas em ações e estas em métodos. Os usuários aplicam seleção de regras para escolher entre os métodos a fim de alcançar as metas.

A abordagem baseada em teoria explanatória é uma outra técnica que pode ser usada e prevê a descrição da sequência de eventos, identificação de causas e efeitos e de eventuais intervenções necessárias. Um exemplo é a abordagem de Norman, baseada em um modelo cíclico composto de sete estágios de ação: formando o objetivo, formando a intenção, especificando a ação, executando a ação, percebendo o estado do sistema, interpretando o estado do sistema e avaliando o resultado.

A avaliação heurística é outro exemplo de técnica. Trata-se de um tipo de avaliação analítica que visa identificar problemas de usabilidade a partir de um conjunto de princípios (também chamados de heurísticas, diretrizes ou *guidelines*), baseados nas melhores práticas definidas por profissionais experientes e especialistas em IHC. Tipicamente uma avaliação heurística é composta de três estágios: sessão breve e preliminar (na qual se diz aos especialistas o que fazer), período de avaliação (no qual cada especialista inspeciona independentemente o produto, utilizando as heurísticas como guia, e para cada heurística violada define a sua localização na interface e sua gravidade conforme uma tabela de grau de severidade quanto à usabilidade) e sessão de resultados (na qual os especialistas de reúnem para discutir as descobertas, priorizar os problemas e sugerir soluções).

Uma das técnicas é a abordagem das Dez Heurísticas de Nielsen (detalhada na atividade “estabelecer diretrizes de usabilidade”).

O percurso cognitivo é outro exemplo de técnica aplicável às avaliações. Trata-se de um tipo de avaliação analítica que busca simular um processo de solução de problemas a cada passo da interação entre o usuário e a interface, verificando se é possível assumir que os objetivos do usuário e sua memória para as ações conduzam a uma próxima ação correta. Envolve atividades como: definição das hipóteses sobre os usuários e sobre o conhecimento que eles têm a respeito da tarefa e da interface proposta, definição dos cenários de tarefas (e da seqüência “correta” de ações para concluir cada tarefa), proposta de *design* em papel ou protótipo (ilustrando cada passo e indicando o estado da interface antes/depois de cada um), construção de “histórias plausíveis” sobre a interação de um usuário típico com a interface (com base nos cenários de tarefas selecionados), simulação da execução da tarefa (efetuando uma série de perguntas sobre cada passo, buscando descobrir problemas em potencial, ou seja, que poderiam ocorrer durante a interação de usuários reais com o produto final implementado conforme aquela proposta de *design*) e anotação final dos pontos-chave (como o que o usuário precisa saber antes de realizar a tarefa e o que o usuário deve aprender ao realizar a tarefa).

O método empírico é outra opção como técnica de avaliação a ser utilizada. Trata-se de um tipo de avaliação que envolve os usuários nos testes da interface, normalmente em ambientes controlados como laboratórios. Uma avaliação realizada por este método deve seguir um processo que envolve várias atividades.

O planejamento no método empírico envolve primeiramente uma série de atividades para a avaliação como:

- a) Revisar as condições de testes, certificando-se de que estas são as mesmas para todos os participantes;
- b) Selecionar as tarefas que terão foco no teste. O critério pode ser o mesmo já mencionado anteriormente para selecionar funcionalidades para prototipação, por exemplo;

- c) Popular corretamente o banco de dados (com dados próximos da realidade);
- d) Definir as medidas a serem utilizadas, baseadas nos indicadores da tarefa e nos objetivos de usabilidade;
- e) Elaborar um plano de teste;
- f) Selecionar os participantes, que representem usuários típicos e com representatividade quanto à categorias e amostragem;
- g) Garantir as questões éticas, como prover informação clara e precisa aos participantes, garantia de anonimato quando aplicável, explicar as condições acordadas para o teste, obter consentimento documentado do usuário, entre outras;
- h) Preparar os materiais necessários à execução como instruções, questionário pré-teste, materiais de treinamento, documento para armazenamento dos dados coletados, identificação do material, questionários pós-teste, entre outros;
- i) Decidir o local ou ambiente de teste. Pode ser um ambiente controlado como um laboratório ou o próprio ambiente real no caso de ambientes incomuns ou de alta complexidade para reprodução com fidelidade;
- j) Projetar o local ou ambiente de testes no caso de laboratórios (móvel ou fixo, uma ou duas salas, *hardware*, câmeras, equipamentos específicos como eye-tracking e outros de apoio, posicionamento adequado de todos os equipamentos para observação, gravação ou filmagem);
- k) Planejar as técnicas a serem usadas no ambiente de teste como *think-aloud* ou *retrospective think aloud* por exemplo;
- l) Documentar tudo isso em um plano de avaliação, encerrando o planejamento.

A execução no método empírico envolve uma série de atividades para a avaliação como:

- a) Seguir e manter atualizado o plano de avaliação;
- b) Executar teste-piloto (opcional);
- c) Executar a avaliação em si (o teste é realizado e os dados são coletados e sumarizados);
- d) Analisar e interpretar os dados (os problemas observados são registrados, classificados conforme a severidade e comparando o resultado aos objetivos pré-estabelecidos);
- e) Formular conclusões e sugestões de mudanças (os problemas e as soluções são priorizados).

Outra técnica para avaliação é a avaliação remota, que pode ser realizada de diversas formas. Exemplos incluem, mas não se limitam a: acompanhamento por meio de *softwares* de emulação, vídeo-conferência, *softwares* de monitoramento com captura de dados e imagens durante o uso da interface pelo usuário (avaliação remota instrumentada) e por meio de logs de uso do sistema ou aplicativos que permitam com que o usuário identifique e comente os pontos da interface que apresentaram problemas (avaliação remota semi-instrumentada).

Outras técnicas também podem ser usadas como apoio às avaliações como: percurso pluralista ou pluralístico (usuários, desenvolvedores e especialistas em usabilidade percorrem uma interface juntos discutindo questões de usabilidade associadas a elementos de diálogo envolvidos nos passos de cenário), inspeção de consistência (entre uma nova interface de uma família de produtos com as demais interfaces dessa família, verificando-se terminologia, *fonts*, cores, *layout*, formatos de entrada e saída de dados, bem como documentação e ajuda *online*), inspeção padrão (um especialista em determinado padrão relevante da interface checa a aderência da interface a esse padrão), inspeção formal de usabilidade (semelhante a uma inspeção de código mas voltada a problemas de usabilidade), Metáforas do Pensamento Humano (MOT), *Bird's-eye view* (vista aérea, para se apurar inconsistências e, no caso do uso de múltiplos desenvolvedores, verificar se um padrão foi seguido), testes *Can-you-break-this* (abordagem destrutiva em que usuários percorrem a interface em busca de falhas fatais e vulnerabilidades) e

ferramentas para avaliação automatizada (quando há um grande número de interfaces a ser verificado, por exemplo) como o *Webtango*.

Um framework que pode auxiliar na orientação de avaliações é o DECIDE (*determine, explore, choose, identify, decide, evaluate*).

**Evidência Objetiva:** Deve haver um registro da análise e da decisão quanto ao tipo e técnica de avaliação e de coleta de dados. As demais evidências dependem da técnica de avaliação escolhida. A seguir são elencados exemplos típicos de evidências para as técnicas mais amplamente descritas nesse guia. Se a técnica adotada foi o uso de questionários, as evidências principais são os questionários preenchidos e a análise posterior com as devidas ações corretivas e de melhorias. Se a técnica usada foi a análise preditiva, um exemplo de evidência são os resultados documentados da utilização do método GOMS. Para a técnica de avaliação heurística ou outras inspeções, exemplos de evidência são a ata da sessão inicial, o relatório de descobertas ou de conformidade aos critérios (heurísticas violadas e severidades) e a ata da sessão de resultados. Se foi utilizada a técnica de percurso cognitivo, exemplos de evidências podem ser o documento de hipóteses definidas, a definição dos cenários de tarefas, a proposta de *design* (com os passos e estados da interface), as relação de perguntas e o relatório de considerações finais. Para a técnica de avaliação pelo método empírico, as principais evidências são o plano de avaliação (que deve evidenciar todos os resultados do planejamento) e respectivos documentos relacionados ao plano, registro dos dados coletados e sumarizados, registro dos problemas observados e classificados e registro das conclusões e sugestões.

**Justificativa:** As técnicas apresentadas nesta atividade estão respaldadas no referencial teórico do presente trabalho e evidenciadas nas seções 3.2.2 pela NBR 9241-11 (2002), 3.2.3 pela NBR ISO/IEC 12207 (2009), 4.2.4.1 por Rocha e Baranauskas (2003), 4.2.4.2 por Prates e Barbosa (2003) e Preece, Rogers e Sharp (2005), bem como nos seguintes trechos específicos desta dissertação:

- a) Questionários do tipo “*user survey*” são apresentados na seção 4.2.4.2 por Avelino (2005).

- b) A Análise preditiva é apresentada na seção 4.2.4.2 por Shneiderman e Plaisant (2010).
- c) A abordagem baseada em teoria explanatória é apresentada na seção 4.2.4.2 por Norman (1988).
- d) A avaliação heurística é apresentada na seção 4.2.4.3.1 por Nielsen (1993) e Nielsen e Mack (1994).
- e) O percurso cognitivo é apresentado na seção 4.2.4.3.1 por Nielsen e Mack (1994) e Prates e Barbosa (2003).
- f) O método empírico é apresentado na seção 4.2.4.4 por Nielsen (1998), Prates e Barbosa (2003), Dumas e Redish (1999), Nielsen (2000), Preece, Rogers e Sharp (2005) e Shneiderman e Plaisant (2010).
- g) Outras técnicas são apresentadas na seção 4.2.4.4 por Mayhew (1999), Preece, Rogers e Sharp (2005) e Shneiderman e Plaisant (2010).
- h) A técnica GOMS é apresentada na seção 4.2.4.2 por Shneiderman e Plaisant (2010) e a técnica DECIDE é apresentada na seção 4.2.4.5 por Preece, Rogers e Sharp (2005).

### **6.1.17      Atividade 17**

**Atividade:** Realizar sessão para *feedback* do usuário e lições aprendidas.

**Grupo de Processo:** Encerramento.

**Objetivo:** Garantir que haja um entendimento e aceitação formal junto ao usuário, quanto ao cumprimento da usabilidade acordada para o produto final e garantir que as lições aprendidas sejam documentadas para as devidas considerações nos futuros projetos ou fases do projeto.

**Técnicas:** A usabilidade do sistema deve ser formalmente aceita. As lições aprendidas devem ser coletadas e documentadas ao longo de todo o ciclo de vida do projeto. Ao final do projeto pode haver uma reunião para análise das lições registradas e elaboração de planos de melhorias para as atividades de usabilidade

previstas no processo da organização. Organizações que utilizam a ISO/IEC 15504-5 (2006) podem utilizar o processo organizacional definido e aderente principalmente ao processo RIN.3 (gestão do conhecimento), à prática MAN.4.PB7 (coletar *feedback* para a gestão da qualidade) e ao processo PIM.3 (melhoria de processo).

Quanto à obtenção do *feedback* do usuário, exemplos de técnicas incluem mas não se limitam ao uso de: questionários, testes de usabilidade, entrevistas, grupos focais, abordagem automatizada e estudos de utilização.

As técnicas de questionários, entrevistas e grupos focais se encontram mencionadas na atividade “identificar o perfil do usuário, sob a perspectiva da usabilidade”. A técnica de testes de usabilidade é mencionada na atividade “executar avaliações iterativas da interface ao longo de sua construção”. Para a técnica de abordagem automatizada pode utilizar o *design* participativo, mencionada na atividade “construir versões iterativas, intermediárias e iterativas da interface”.

Para a técnica de estudos de utilização, primeiramente define-se a forma de coleta de dados, que pode ser por observação randômica (presencial) ou por meio de monitoramento remoto por *software*. Os estudos devem ser executados em diferentes horários, pois os resultados e o trabalho podem variar conforme o horário. Os dados devem ser coletados e analisados, buscando-se perceber as funções que são usadas com baixa frequência ou por pequenos períodos e tentando-se apurar os motivos. As conclusões devem ser documentadas, relatando-se os eventuais problemas na usabilidade que causaram o baixo uso das funções identificadas.

**Evidência Objetiva:** Registro de aceitação da usabilidade, lições aprendidas e *feedback*, conforme a técnica utilizada e registro de plano de melhoria do processo baseado nas informações obtidas.

**Justificativa:** As técnicas apresentadas nesta atividade estão respaldadas no referencial teórico do presente trabalho e evidenciadas nas seções 3.2.5 pela ISO/IEC 15504-5 (2006), 3.2.6 pela ISO/TR 18529 (2000), 3.3.2 por Chrissis, Konrad e Shrum (2003), 3.3.1 pelo PMBOK (2008), 3.4 por Cybis, Betiol e Faust (2007) e 4.3.4 por Mayhew (1999, 2008).

Essas são as atividades previstas para o guia objeto desta dissertação. A seguir será apresentado o método de utilização proposto para a aplicação desse guia em uma avaliação dos processos de uma organização.

## **6.2 Método para verificação de aderência ao guia**

A seguir serão apresentados os passos para a verificação de aderência de processos de desenvolvimento de *software* às práticas do guia proposto.

É importante salientar que como o objetivo da verificação com base no guia não é atribuir um nível ou pontuação à organização, mas sim proporcionar uma visão do grau de aderência atual da organização às práticas do guia, o método foi elaborado de forma a refletir esse caráter de orientação e não de certificação ou conformidade como resultado final.

A primeira atividade a ser realizada é uma sessão prévia de curta duração com os envolvidos, a fim de se relacionar os principais problemas estratégicos e gerenciais atuais na organização, relacionados à usabilidade. O objetivo é elencar de forma geral (quando possível com menção a indicadores quantitativos) problemas de alto nível como falta de atingimento de resultados dos projetos, impacto nos objetivos de qualidade da organização, reclamação ou perda de clientes, entre outros. Esse levantamento deve ser breve, mas é fundamental para direcionar a atividade de sessão de consenso, que será vista mais adiante nesta seção. O objetivo é que essa atividade contribua para a identificação do quanto os objetivos de usabilidade estão alinhados aos objetivos estratégicos e coletar informações que serão úteis para as atividades seguintes. Essa necessidade está embasada no mesmo propósito de avaliação de processo da ISO/IEC 15504-5 (2006), que é definido como *“determinar o quanto que os processos padrões da organização contribuem para o atendimento de seus objetivos estratégicos e apoiar o foco da organização na necessidade de melhoria contínua de processo”* (SALVIANO, 2006).

O próximo passo é a preparação da verificação. A verificação deve permitir que se apure o quanto uma atividade está definida (prevista e documentada nos processos da organização) e o quanto está institucionalizada (conhecida pelos envolvidos e executada na prática nos projetos). Esse tipo de análise é baseado no método adotado para avaliações pelo CMMI (CHRISSIS, KONRAD e SHRUM,



2003). A fim de se verificar se uma atividade está definida, devem ser disponibilizados aos avaliadores a metodologia de desenvolvimento de *software* da organização e os documentos relacionados a esta (políticas, processos, procedimentos, guias, *templates*, manuais, entre outros). A fim de se verificar se a atividade está institucionalizada, devem ser disponibilizados os artefatos e documentos de gestão de projetos (como especificações, diagramas, relatórios, planos, atas, entre outros), além de produtos intermediários e finais dos *softwares* gerados (protótipos e manuais de sistema, por exemplo).

Em seguida, inicia-se a verificação dos pontos de aderência e as lacunas entre as atividades do guia proposto e as atividades desempenhadas pela organização. Com esta finalidade, são verificados tanto os processos (para se apurar o grau de definição dos processos) como as evidências objetivas de projetos e organizacionais (para se apurar o grau de institucionalização dos processos). Para isso, procede-se da maneira descrita a seguir. Para cada atividade do guia, o processo da organização é verificado para se identificar se este cobre ou não as recomendações desta atividade e com que profundidade. Isto permite verificar o grau de aderência do processo definido pela organização às atividades do guia. Para cada atividade do guia, são também analisadas as evidências objetivas coletadas dos projetos (documentos produzidos durante o projeto) e da organização (documentos produzidos fora do âmbito dos projetos, no caso das atividades organizacionais). Isto permite verificar se as atividades, além de estarem definidas e documentadas no processo da organização, são também executadas de fato pelos respectivos responsáveis na organização.

Ao final da verificação de cada atividade, procede-se à sessão de consenso, que tem como objetivo consolidar as informações coletadas e apurar o grau de aderência da organização ao guia. O objetivo não é conferir uma certificação de aderência, nível de maturidade, pontuação, percentual geral ou *score* de cobertura, mas sim, um relatório final que demonstre para cada atividade o nível de aderência atual da organização e a recomendação para eventual melhoria desse nível.

Nessa sessão, para cada atividade são consolidadas as evidências e passa-se ao registro das conclusões das atividades. Nesse registro, para cada atividade do guia, cinco questões devem ser respondidas: grau atual de utilização efetiva da

atividade na organização, problemas atuais e potenciais correlacionados ao grau de utilização, benefícios atuais e potenciais correlacionados ao grau de utilização, relevância desta atividade para atuar nos problemas e/ou benefícios e, por fim, sugestões de ajustes para esta atividade. Cada questão será explicada nos parágrafos a seguir.

A questão sobre o grau atual de utilização da atividade tem como objetivo apurar se a atividade realmente está definida e institucionalizada da forma devida. Para o preenchimento das respostas será usada a seguinte escala quanto à utilização: não aplicável, nenhuma, baixa, alta ou plena. Foram consideradas as escalas do CMMI (CHRISISS, KONRAD E SHRUM, 2003) e da ISO/IEC 15504-5 (2006) como base para a elaboração desta escala. O critério para preenchimento não é subjetivo. A resposta deve ser baseada nas evidências objetivas coletadas durante a avaliação e apurada da seguinte forma: deve ser respondido “não aplicável” quando a implementação da atividade não faz sentido no contexto da organização (devido à natureza da operação, metodologia, tecnologia utilizada ou outro motivo justificável), “nenhuma” quando é aplicável, mas a organização não utiliza as práticas dessa atividade, “baixa” quando as sub-atividades ou técnicas aplicáveis estão definidas, mas não estão institucionalizadas, “alta” quando as sub-atividades ou técnicas aplicáveis estão definidas e institucionalizadas e “plena” quando todas as sub-atividades ou técnicas aplicáveis são plenamente cobertas pelo processo, estando definidas e institucionalizadas.

A questão sobre os problemas atuais e potenciais é dissertativa e tem como objetivo a identificação de problemas, pontos fracos, vulnerabilidades e riscos (atuais), além de ameaças (futuras) que tenham relação direta com o grau de implementação desta atividade específica. A falta de utilização efetiva de uma atividade pode trazer riscos não apenas para a usabilidade como para outros aspectos da qualidade do produto, da imagem da organização ou dos resultados dos seus projetos. Por outro lado, a implementação inadequada, exagerada ou excessiva de determinada atividade pode da mesma forma trazer problemas como alto custo, atrasos, resistência por parte dos envolvidos, entre outros.

A questão sobre os benefícios atuais e potenciais é dissertativa e tem como objetivo a identificação de pontos fortes e diferenciais (atuais) ou oportunidades

(futuras) da organização, que tenham relação direta com o grau de implementação desta atividade específica. Um alto grau de implementação pode render bons resultados nos projetos, aumentar a satisfação do cliente, diminuir o número de defeitos e retrabalho, entre outros fatores. Por outro lado, essa questão pretende levantar também se eventualmente a organização pode enxergar o baixo grau de implementação de determinada atividade como um benefício. A não utilização de uma atividade pode ser enxergada pela organização como, por exemplo, uma forma de redução de custos que lhe traz um diferencial competitivo de preço junto aos concorrentes (o que deve ser avaliado com cautela já que pode representar riscos).

A questão sobre a relevância da utilização da atividade tem como objetivo apurar qual o grau de contribuição que a utilização desta atividade traz (quando já está implementada na metodologia de desenvolvimento de *software*) ou pode trazer (caso fosse implementada) para impactar de maneira positiva os objetivos de usabilidade da organização (atuando sobre a probabilidade e/ou impacto dos benefícios e problemas elencados nas questões anteriores). Novamente, essa necessidade está embasada no mesmo propósito de avaliação de processo da ISO/IEC 15504-5 (2006) para atingimento dos objetivos estratégicos. Para o preenchimento das respostas será usada a seguinte escala quanto à relevância: nenhuma, baixa, alta ou imprescindível. A resposta deve ser baseada na correlação de causa e efeito estabelecida, junto aos especialistas, entre as atividades e os problemas e benefícios elencados e deve ser apurada da seguinte forma: deve ser respondido “nenhuma” quando a relação custo/benefício não é favorável ou simplesmente inviabiliza sua implementação (por exemplo, devido a restrições como custo, complexidade ou prazo para sua implementação), “baixa” quando pode atuar de alguma forma reduzindo os problemas e/ou aumentando os benefícios elencados, “alta” quando pode atuar de forma significativa, influenciando fortemente a probabilidade e impacto dos riscos positivos e negativos associados aos problemas e benefícios elencados, e finalmente, “imprescindível” quando a ausência de implementação da atividade inviabiliza ou compromete de forma significativa o cumprimento dos objetivos de usabilidade estabelecidos para os projetos e/ou para a organização.

A última questão, sugestões de ajustes, é dissertativa e tem como objetivo obter recomendações de correções e melhorias no guia, identificadas a partir do seu

uso efetivo na verificação dos processos da organização. Este é um mecanismo de retroalimentação para a melhoria progressiva e contínua do guia, aderente ao objetivo de referências como a ISO/IEC 15504-5 (2006) que busca melhoria do processo (prática PIM.3) e aos níveis mais altos de maturidade do CMMI (CHRISSIS, KONRAD E SHRUM, 2003) que buscam a melhoria contínua dos processos. Desta forma, espera-se que a cada avaliação executada sejam levantados pontos de melhoria para o guia. Estes serão analisados e eventualmente implementados os devidos ajustes para atender às sugestões apontadas. Exemplos de sugestões de ajustes podem ser a inclusão, exclusão ou alteração de atividades e técnicas, inserção ou atualização de bibliografias de referência mencionadas nas técnicas, atividades e diretrizes, alteração do grupo de processos de atividades e alteração das evidências objetivas, entre outras.

### **6.3 Uso e perspectivas de aplicabilidade do guia**

Para avaliar a aplicabilidade do guia e do respectivo método de aplicação, obtidos como resultado deste trabalho, estes foram submetidos a uma análise por parte de uma equipe de especialistas em usabilidade de uma organização brasileira de grande porte que atua com o desenvolvimento de sistemas de gestão empresarial, com presença consolidada no mercado do Brasil e na América Latina. Essa organização possui um departamento focado em usabilidade, que conta com processos, laboratório e infraestrutura voltados à garantia da usabilidade dos *softwares* produzidos.

A fim de avaliar a aplicabilidade e utilidade do guia, os especialistas das áreas de desenvolvimento de soluções, inovação e usabilidade da organização analisaram todas as atividades propostas, seus objetivos, grupos de processos e suas técnicas relacionadas. Para avaliar a aplicabilidade do método de verificação proposto, a equipe conduziu uma verificação de aderência dos processos da organização, utilizando o próprio método e guia apresentados neste trabalho. Como o próprio método de verificação proposto já prevê a coleta de recomendações para ajuste do modelo, estas foram utilizadas tanto para a melhoria do guia inicialmente proposto como para a conclusão sobre a aplicabilidade do guia e método de verificação propostos.

Primeiramente, foram apresentados à equipe da organização o propósito da criação do guia (descrito na introdução deste trabalho), o seu contexto (projeto de pesquisa acadêmico), o referencial teórico que serviu de embasamento para sua elaboração, o objetivo deste procedimento empírico de validação da sua aplicabilidade e utilidade e a estrutura do guia (subdivisão em atividades, técnicas, justificativas e demais itens). Em relação a esses pontos apresentados não houve nenhuma recomendação de ajustes.

A partir desse entendimento, o processo de verificação seguiu exatamente os passos mencionados no método proposto por esse trabalho. Foi entregue pelo autor deste trabalho à organização uma lista de todos os itens da metodologia de desenvolvimento de *software* e as típicas evidências objetivas que poderiam ser necessárias durante o processo de verificação para comprovar a utilização na organização das práticas que constam no guia. Em poucos dias a equipe disponibilizou toda a documentação necessária, selecionou um projeto relevante e recente da empresa reunindo todos os artefatos gerados que estavam correlacionados à lista apresentada. A única recomendação levantada pela organização sobre os passos de preparação da verificação é que fosse contemplada a realização de um treinamento prévio sobre o guia para a equipe que o utilizará, de forma a facilitar o entendimento da equipe em relação às atividades e agilizar a execução, já que pode haver diferença de nomenclatura entre os termos apresentados no guia e os nomes dos itens da metodologia e artefatos da organização, dificultando a coleta destes. Em seguida procedeu-se à execução da verificação. Cada atividade do guia foi explorada e, para cada uma, foram analisados os processos e evidências correlacionados. Ao final desta análise, uma sessão de consenso foi realizada para se chegar às recomendações de melhorias e correções em cada atividade.

Ao final, com base nas recomendações obtidas, foi feita uma análise por parte do autor deste trabalho e dos especialistas que participaram da verificação. Foram discutidos ajustes quanto ao guia, suas atividades e ao método de verificação. A seguir serão apresentadas as principais conclusões e recomendações obtidas.

Em relação às atividades, todas foram entendidas pela equipe da organização como aplicáveis e úteis. A quantidade de atividades e seus objetivos foram

considerados adequados para cobrir os aspectos fundamentais da usabilidade. Em relação à relevância das atividades, é importante primeiramente lembrar o critério adotado pelo método de verificação proposto. Segundo o método, a organização classifica a relevância de cada uma das atividades do guia usando a seguinte escala: baixa relevância (quando atua de alguma forma sobre a usabilidade, reduzindo os problemas e/ou aumentando os benefícios do uso da atividade), alta relevância (quando atua de forma significativa, influenciando a probabilidade e/ou impacto dos riscos) ou imprescindível (quando a ausência da atividade inviabiliza ou compromete de forma significativa o cumprimento dos objetivos dos projetos ou até mesmo da organização). Das dezessete atividades, quatro foram classificadas como baixa relevância (algum impacto na usabilidade), onze foram classificadas como alta relevância (alto impacto na usabilidade) e duas como imprescindíveis (obrigatórias para o sucesso dos projetos). Este resultado mostra que a seleção das atividades está coerente. Foi sugerida a separação da atividade dezessete (realizar sessão para *feedback* do usuário e lições aprendidas) em duas atividades: uma especificamente voltada ao tratamento das lições aprendidas dos projetos e outra à obtenção do *feedback*. Esta sugestão é coerente pois, embora sejam ambas atividades típicas do encerramento de fase ou projeto, as técnicas e as evidências são bem distintas. Não houve sugestão de inclusão ou exclusão de atividades.

Quanto às técnicas do guia, foram propostas sugestões como as apresentadas a seguir. Para a atividade de número um (conscientizar o usuário e a gestão sobre a importância da usabilidade), foi sugerido incorporar técnicas como *e-learning*, oficinas e palestras. Para a atividade de número quatro (estabelecer diretrizes de usabilidade), foi sugerido incluir diretrizes que tratem de forma específica os critérios ergonômicos. Na atividade de número cinco (monitorar o uso e efetividade das diretrizes de usabilidade), sugeriu-se incluir como técnica o uso de comunidades (*wikis*) já que demonstraram bons resultados na organização. No caso da atividade de número sete (monitorar se os projetos seguem os processos de usabilidade da metodologia de desenvolvimento de *software* da organização) foi comentado que além das auditorias formais, técnicas como o uso de recompensas e/ou premiações para incentivo ao uso dos processos podem ter bons efeitos. Para a atividade de número quinze (construir versões iterativas, intermediárias e interativas da interface) foi recomendada a inclusão da ferramenta Balsamiq Mockup

como técnica alternativa para geração de maquetes e protótipos e realização de testes com o usuário, já que esta ferramenta tem demonstrado maior facilidade, agilidade e praticidade do que o uso de maquetes e protótipos em papel. Tais recomendações demonstram que a aplicação prática do guia por meio de verificações podem contribuir efetivamente para que se mantenha um conjunto amplo, corroborado e atualizado das técnicas em cada atividade.

Em relação aos grupos de processo associados à cada atividade, a estrutura e nomenclatura baseada nos grupos de processo do PMBOK (2008) foi aceita como adequada. Foi sugerido que as atividades de número dois (estabelecer objetivos claros de usabilidade) e três (monitorar o atingimento dos objetivos de usabilidade) fossem organizacionais e não restritas a um grupo de processo. A sugestão é válida (já que normalmente não há tempo hábil para executar tais atividades dentro do projeto) desde que exista uma área responsável e um processo formal destacados para tais atividades.

Um ponto importante observado na validação, em relação ao método de verificação proposto, é que este permite a visibilidade de eventuais incompatibilidades entre o grau de utilização efetiva de uma atividade pela organização e o grau de relevância percebida (pela equipe envolvida na verificação de aderência) para esta atividade, ou seja, quando o seu grau de utilização é maior ou menor do que o grau de aderência às práticas do guia. Tal fato não retrata necessariamente a falta ou excesso de práticas para uma determinada atividade por parte da organização já que técnicas alternativas às mencionadas no guia podem estar sendo utilizadas, porém, é objeto de atenção por parte da equipe. Ao final, foram solicitadas as percepções e considerações gerais da equipe envolvida a respeito do guia e do método de verificação. De acordo com os depoimentos, o guia foi considerado efetivamente aplicável e útil para organizações que pretendem utilizar processos voltados à garantia da usabilidade.

A seguir será apresentado um breve resumo das alterações realizadas no guia e no método de verificação propostos pelo presente trabalho, após a análise das considerações recebidas e destacadas acima pela equipe envolvida nessa primeira utilização do guia e do método.

### 6.3.1 Ajustes no guia e no método de verificação

Em relação ao método de verificação, o primeiro ajuste incorporado foi a inclusão de um breve treinamento sobre o guia para a equipe envolvida, entre a sessão prévia e a preparação da verificação. O objetivo desse treinamento é fazer com que a equipe se familiarize com as nomenclaturas e técnicas descritas no guia e que isso facilite a coleta das evidências de processos e de projetos por parte da equipe. Esse entendimento pode agilizar também a execução da verificação, uma vez que eventuais dúvidas sobre as técnicas e divergências de terminologias já tenham sido esclarecidas.

Em relação ao guia, serão elencadas a seguir as alterações incorporadas. A atividade dezessete (realizar sessão para *feedback* do usuário e lições aprendidas) pode ser desmembrada em duas atividades: a primeira voltada especificamente à obtenção do *feedback* do usuário enquanto a segunda, uma nova atividade para o guia (atividade dezoito), voltada à obtenção e organização das lições aprendidas dos projetos. Nesse caso, os objetivos e evidências devem ser separados para as duas atividades. Em relação às técnicas, para a atividade dezoito são aplicáveis os processos da organização aderentes principalmente ao processo RIN.3 (gestão do conhecimento) e PIM.3 (melhoria de processo) da ISO/IEC 15504-5 (2006) enquanto as demais técnicas permanecem como aplicáveis à atividade dezessete. A atividade de número cinco (monitorar o uso e efetividade das diretrizes de usabilidade) deve ter uma nova técnica: o uso de comunidades (*wikis*) e redes sociais para apurar se as diretrizes são de conhecimento dos envolvidos. A atividade de número sete (monitorar se os projetos seguem os processos de usabilidade da metodologia de desenvolvimento de *software* da organização) deve contemplar uma nova técnica: o planejamento e uso de recompensas e premiações como incentivo ao uso dos processos de usabilidade. A atividade de número quinze (construir versões iterativas, intermediárias e interativas da interface) deve listar a ferramenta Balsamiq Mockup como técnica alternativa para geração de maquetes e protótipos.

Em relação aos grupos de processos das atividades, as atividades de número dois (estabelecer objetivos claros de usabilidade) e três (monitorar o atingimento dos objetivos de usabilidade) passam a ser organizacionais.



## 7. Conclusão

O presente trabalho concluiu o guia proposto a partir da identificação das atividades relacionadas ao ciclo de vida do desenvolvimento de sistemas, que visam buscar maior qualidade da usabilidade das interfaces a partir do foco ao atendimento destas atividades no processo de desenvolvimento de *software* de uma organização.

A definição das atividades foi baseada em uma análise que considerou os diferentes tipos de referências disponíveis (normas, padrões, modelos e referencial teórico) e as diversas fontes em cada tipo de referência, mantendo-se, contudo, sempre o foco na perspectiva de usabilidade e na visão do processo. A organização e estrutura do guia permitiram a criação de um conjunto coeso, prático e integrado de atividades, baseado não somente na corroboração das melhores práticas como também trazendo contribuições como o tratamento à questão da gestão de mudanças, da classificação das atividades em grupos de processo, da consolidação de várias atividades com objetivos comuns (como as atividades de revisão da metodologia, de monitoramento da metodologia e de construção de versões iterativas, intermediárias e iterativas).

A partir da aplicação prática do guia em uma organização, os envolvidos no processo concluíram que o guia e o método podem ser considerados válidos, que os resultados encontrados por meio de sua utilização permitem uma análise efetiva dos processos de usabilidade da organização e que o guia pode contribuir de forma significativa para a melhoria dos produtos de *software* por meio do foco na usabilidade.

Como consideração para trabalhos futuros, pode-se estender a aplicação do guia para outros tipos de organizações (diferentes da que foi utilizada neste trabalho para a verificação da aplicabilidade do guia) a fim de verificar a adequação do guia a outros tipos de cenário e, conseqüentemente, refinar sua qualidade. Esse refinamento progressivo a cada utilização do guia já é previsto no método de avaliação proposto pelo trabalho, por meio do seu mecanismo de retroalimentação e melhoria contínua. Outra sugestão de trabalho futuro é uma revisão e possível adaptação do guia de forma a estar mais aderente aos ciclos de vida de metodologias ágeis e a empresas de outros segmentos como as empresas de mídia.

## 8. Referências

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **NBR 9241-11: Requisitos Ergonômicos para Trabalho de Escritórios com Computadores: Orientações sobre Usabilidade**. Rio de Janeiro, 2002. 21 p.

\_\_\_\_\_. **NBR ISO/IEC 9001: Sistemas de Gestão da Qualidade – Requisitos**. Rio de Janeiro, 2008. 41 p.

\_\_\_\_\_. **NBR ISO/IEC 9126-1: Engenharia de software - Qualidade de produto**. Rio de Janeiro, 2003. 21 p.

\_\_\_\_\_. **NBR ISO/IEC 12207: Engenharia de Sistemas e Software - Processos de ciclo de vida de *software***, Rio de Janeiro, 2009. 121 p.

\_\_\_\_\_. **NBR ISO/IEC 25000: Engenharia de software - Requisitos e avaliação da qualidade de produtos de software (SQuaRE) - Guia do SQuaRE**, Rio de Janeiro, 2008. 42 p.

AVELINO, V. F. **MERUSA: Metodologia de Especificação de Requisitos de Usabilidade e Segurança orientada para Arquitetura**. 2005. 277p. Tese (Doutorado em Engenharia de Computação e Sistemas Digitais). Escola Politécnica, Universidade de São Paulo, 2005.

BARANAUSKAS, M. C. C.; MELO, A. M. **Design para a Inclusão: Desafios e Proposta**. Anais do IHC 2006 – Natal, Brasil. 2006.

BARBOSA, D.F.; FURTADO, E.S.; GOMES, A.S. **Uma Proposta de Institucionalização da Usabilidade Alinhada com Práticas do Modelo CMMI e Foco nas Necessidades da Organização**. In IHC 2006 – VII Simpósio Sobre Fatores Humanos em Sistemas Computacionais. Natal, Novembro, 2006.

BARBOSA, S.D.J.; SILVA, B.S. **Interação Humano Computador**. Rio de Janeiro: Elsevier, 2010.

BENINI, M. J. S. **Uma agenda de Intervenção para a Ergonomia da Atividade na Concepção de Sistemas Computacionais Interativos**. 2006. 113p. Dissertação (Mestrado). Escola Politécnica, Universidade de São Paulo, 2006.

BOEHM, B. W. **A spiral model of software development and enhancement**. IEEE Computer, 21(5), 61-72, 1988.

CARD, S.; MORAN, T.; NEWELL A. **The Psychology of Human-Computer Interaction**. Hillsdale, NJ: Laurence Erlbaum Associates, 1983.

CARROLL, J. M. **Introduction to the special issue on “Scenario-Based Systems Development”**. Interacting with Computers. 13(1), 41-42, 2000.

CHRISSIS, M. B.; KONRAD M.; SHRUM, S. **CMMI – Guidelines for Process Integration and Product Improvement**. Boston: Addison-Wesley Pearson Education, 2003.

CONSTANTINE, L. L.; LOCKWOOD, L. A. D. **Software for use**. Harlow, UK: Addison-Wesley, 1999.

CYBIS, W.; BETIOL, A. H.; FAUST, R. **Ergonomia e Usabilidade - Conhecimentos, Métodos e Aplicações**. São Paulo: Novatec Editora, 2007.

DIX, A. et al. **Human-Computer Interaction**. 3rd Ed. London: Pearson Prentice Hall, 2004.

DUMAS, J. S.; REDISH, J. C. **A Practical Guide to Usability Testing (Revised Edition)**. Exeter, UK: Intellect, 1999.

EARTHY, J. **Usability Maturity Model: Processes**, Technical Report, 2000. Disponível em: <<http://www.idemployee.id.tue.nl/g.w.m.rauterberg/lecturenotes/Usability-Maturity-Model%5B2%5D.pdf>>. Acesso em 22 de Maio de 2012.

FILHO, A. M. da S. **O papel de protagonistas no desenvolvimento de sistemas interativos**. Artigo publicado na revista Espaço Acadêmico, num. 47, Abril de 2005.

HAMMERSLEY, M.; ATKINSON, P. **Ethnography: principles in practice**. London: Tavistock, 1983.

HIX, D.; HARTSON, H.R. **Developing User Interfaces: Ensuring Usability through Product and Process**. New York: John Wiley, 1993.

HOLTZBLATT et al. **Rapid Contextual Design: A How-To Guide to Key Techniques for User-Centered Design**. Morgan Kaufmann, San Francisco, 2005.

International Organization for Standardization. **ISO 13407. Human-centred design processes for interactive systems**. Geneve, Switzerland, 1999.

International Organization for Standardization. **ISO/TR 18529. Ergonomics of human system interaction – human centered lifecycle process descriptions**. Geneve, Switzerland, 2000.

International Organization for Standardization and the International Electrotechnical Commission. **ISO/IEC 15504-5. Information Technology - Process Assessment – Part 5: An exemplar Process Assessment Model**. Geneve, Switzerland, 2006.

JACOBSON, I. et al. **Object Oriented Software Engineering - A Use Case Driven Approach**. Harlow, UK: Addison-Wesley, 1992.

JUNIOR, S.L.D et al. **Integração IHC e ES: Processo de Planejamento da Reengenharia de Software Guiado pela Avaliação de Usabilidade – PPR-U**. VI Simpósio sobre Fatores Humanos em Sistemas Computacionais — Mediando e Transformando o Cotidiano. Curitiba, UFPR, CEIHC—SBC. 2004.

KOTONIA, G.; SOMMERVILLE, I. **Requirements engineering: processes and techniques**. Chichester, UK: John Wiley & Sons, 1998.

LACERDA, G. S.; BARBOSA, A.; RIBEIRO, V. G. **Adoção do CMMI e Metodologias Ágeis em Empresas Brasileiras**. Revista Avances en Sistemas e Informática, V. 8, n. 3, Medellin, Colômbia, 2011.

LYNCH, G.; PALMITER, S. **Design and Rapid Evaluation of Usable Web Sites**, CHI2002 tutorial notes, 2002.

LYNCH, P.; HORTON, S. **Web Style Guide: Basic Design Principles for Creating Web Sites**, 3. ed., Yale University Press, New Haven, CT, 2008.

MAYHEW, D. J. **The Usability Engineering Lifecycle**. San Francisco: Morgan Kaufmann, 1999.

\_\_\_\_\_. **Requirements specifications within the usability engineering lifecycle**. In: SEARS, A.; JACKO, J. A. *The Human-Computer Interaction Handbook - Fundamentals, Evolving Technologies and Emerging Applications*, 917-926, 2. ed. New York, 2008.

MORAN, T. **The Command Language Grammars: a representation for the user interface of interactive computer systems**. In: *International Journal of Man-Machine Studies*, Academic Press, 1981.

NCI, National Cancer Institute, U.S. Department of Health and Human Services, **Research-Based Web Design & Usability Guidelines**, 2008. Disponível em: <<http://www.usability.gov/pdfs/guidelines.html>>. Acesso em 22 de Maio de 2012.

NIELSEN, J. **Cost of User Testing a Website**, 1998. Disponível em: <<http://www.useit.com/alertbox/980503.html>>. Acesso em 22 de Maio de 2012.

\_\_\_\_\_. **Usability Engineering**. Academic Press, 1993.

\_\_\_\_\_. **Why You Only Need to Test with 5 Users**, 2000. Disponível em: <<http://www.useit.com/alertbox/20000319.html>>. Acesso em 22 de Maio de 2012.

NIELSEN J.; LORANGER H. **Usabilidade na Web – projetando Websites com qualidade**. Rio de Janeiro: Elsevier, 2007.

NIELSEN, J.; MACK, R.L. **Usability Inspection Methods**. New York, NY: John Wiley & Sons, 1994.

NORMAN, D. A. **Design rules based on analyses of human error**, *Communications of the ACM*, 1983.

NORMAN, D. A. **The Design of Everyday Things**. New York: Basic Books, 1988.

OLIVEIRA NETTO, A. A. de. **IHC – Interação Humano-Computador – Modelagem e Gerência de Interfaces com o Usuário**. São Paulo: VisualBooks, 2004.

PAULA, M.G. **Projeto da Interação Humano-Computador Baseado em Modelos Fundamentados na Engenharia Semiótica: Construção de um Modelo de Interação**. 2003. 76f. Dissertação (Mestrado em Informática). Departamento de Informática. PUC-RJ, 2003.

PRATES, R. O.; BARBOSA, S. D. J **Avaliação de Interfaces de Usuário - Conceitos e Métodos**. In: Jornadas de Atualização em Informática – JAI 2003, Rio de Janeiro, 2003.

PREECE, J. **Online Communities: Designing Usability, Support Sociability**. Chichester, UK: John, Wiley & Sons, 2000.

PREECE, J; et al. **Human-Computer Interaction**. Nova Jersey-USA: Addison-Wesley, 1994.

PREECE, J.; ROGERS, Y.; SHARP, H. **Design de Interação: Além da Interação Homem-Computador**. Porto Alegre: Bookman, 2005.

PROJECT MANAGEMENT INSTITUTE. **PMBOK: A Guide to the Project Management Body of Knowledge**. Pennsylvania, EUA. 2008.

REBELO, Irla Bociansoki. **Interação entre Homem e Computador e procedimentos de Avaliação**. Centro Euroamericano UNIEURO, 2009.

ROCHA, H. V. da; BARANAUSKAS, M. C. C. **Design e Avaliação de Interfaces Humano-Computador**. Campinas: NIED/UNICAMP, 2003.

SALVIANO, C. F. **Uma proposta orientada a perfis de capacidade de processo para evolução da melhoria de processo de software**. 2006. 246p. Tese (Doutorado em Engenharia Elétrica). Faculdade de Engenharia Elétrica e de Computação, Universidade Estadual de Campinas, 2006.

SHNEIDERMAN, B.; PLAISANT, C. **Designing the user interface** 5. ed. Boston-USA: Addison Wesley, 2010.

SMITH, S. L.; MOSIER, J. N. **Guidelines for Designing User Interface Software**, Report ESD-TR-86-278, Electronic Systems Division, MITRE, Bedford, MA, 1986.

SOMMERVILLE, I. **Engenharia de Software**, 8.ed. Sao Paulo: Addison Wesley, 2007.

SOUZA, C.S. de; PRATES, R. O.; BARBOSA, S. D. J. **A Method for Evaluating Software Communicability**. Rio de Janeiro, Brasil. PUC-RJ. Inf MCC 11/99, 1999.

SUCHMAN, L. A. **Plans and Situated Actions**. Cambridge: Cambridge University Press, 1987.

SUN Microsystems, Inc. **Web Design Standards**, 2008. Disponível em: <<http://www.sun.com/webdesign/>>. Acesso em 22 de Maio de 2012.

WAI, Web Accessibility Initiative, **World Wide Web Consortium**, 2008. Disponível em: <<http://www.w3.org/WAI/>>. Acesso em 22 de Maio de 2012.

WATANABE, R. H; DUDUCHI, M.. **Método para Aplicações Web Focado em Usabilidade Aderente a um Processo de Software Convencional**. In: The Interaction 09 - South America, São Paulo, 2009.

WHARTON, C.; et al. **The Cognitive Walkthrough Method: A Practitioner's Guide**. In NIELSEN, J.; MACK, R.L. (Eds.), Usability Inspection Methods, John Wiley & Sons, New York, NY, 1994.

WICKENS, C. D.; HOLLANDS, J. G. **Engineering Psychology and Human Performance**, Prentice-Hall, Englewood Cliffs, NJ, 2000.