

FACULDADE DE TECNOLOGIA DE SÃO PAULO

JACKELINE MELO DE LIMA

IHC – Novas tecnologias para entrada e consulta de dados

São Paulo
2012

FACULDADE DE TECNOLOGIA DE SÃO PAULO

JACKELINE MELO DE LIMA

IHC – Novas tecnologias para entrada e consulta de dados

Monografia submetida como exigência parcial
para a obtenção do Grau de Tecnólogo em
Processamento de Dados.

Orientador: Prof. Valter Yogui

São Paulo
2012

Dedicatória

À minha mãe, Maria de Fátima de Melo, por sempre me guiar, incentivar e apoiar em todas as etapas da minha trajetória de estudos.

Também ao meu irmão, José Augusto de Lima Júnior, pelo estímulo ao pensamento crítico e metódico.

Agradecimentos

À minha mãe e ao meu irmão, que sempre acreditaram e confiaram no meu potencial.

A todos os professores, pela conduta, dedicação e ensinamentos que ficarão sempre em minha memória e, especialmente, ao meu orientador Valter Yogui, pela paciência e apoio apesar das circunstâncias.

Aos amigos e colegas, pelo incentivo durante a reta final de mais esse ciclo.

Resumo

Este documento tem o intuito de descrever conceitos básicos sobre a interação humano-computacional e, através de um breve histórico de dispositivos e das interfaces usadas para viabilizar essa comunicação, apresentar um panorama do cenário atual e mencionar metodologias vigentes para a construção de projetos relacionados a esse tema.

A partir de então, tendo em vista as novas tecnologias dessa área, analisar aspectos como acessibilidade, vantagens, desvantagens e empregabilidade para, enfim, concluir estimando as tendências do mercado e o impacto na atual arquitetura de sistemas levando em consideração esse novo paradigma.

Palavras-chave: Interface, Interface Homem-Máquina, IHC, metodologias, tendências.

Abstract

This document aims to describe basic concepts of human-computer interaction and through a brief history of devices and interfaces that are used to facilitate this communication, giving an overview of the current scenario and describe existing methodologies for the construction of projects related to this theme.

Then, bearing in mind the new technologies in this area, consider aspects such as accessibility, advantages, disadvantages and usability and, finally conclude estimating the market trends and the impact on current systems architecture taking into account this new paradigm.

Keywords: Interface, Human-Computer Interaction, HCI, methodologies, trends.

Lista de Ilustrações

Figura 1 – Estrutura de prototipação.....	28
Figura 2 – Estrutura básica do ciclo de vida ágil.....	30

Lista de Tabelas

Tabela 1 - Linha do tempo da evolução dos computadores.....	20
Tabela 2 - Prototipagem: baixa fidelidade x alta fidelidade	27
Tabela 4 - Comparação entre principais metodologias ágeis	32
Tabela 5 - Comparação entre principais metodologias ágeis (cont.).....	33

Lista de Abreviaturas e Siglas

ASD	Adaptive Software Development
ENIAC	Eletronic Numerical Interpreter and Calculator
FDD	Feature Driven Development
GUI	Graphical User Interface
IBM	International Business Machine
IHC	Interface Humano-Computacional
IHM	Interface Homem-Máquina
NUI	Natural User Interface
TI	Tecnologia da Informação
WIMP	Windows, Icons, Menus, Pointing Device
XP	Extreme Programming

Sumário

1. Introdução	11
2. Conceitos básicos.....	13
2.1. Sistema	13
2.2. Interface.....	13
2.2. i. Interface humano-computacional.....	14
2.3. Dispositivos	15
2.3. i. Dispositivos de captação de dados.....	16
2.3. ii. Dispositivos de consulta de dados.....	16
2.3. iii. Dispositivos híbridos.....	16
2.3. iv. Periféricos.....	17
3. Histórico.....	18
3.1. Breve histórico da computação.....	18
3.2. Breve histórico das interfaces e seus desafios	21
3.2. i. Primeira geração.....	21
3.2. ii. Segunda geração.....	22
3.2. iii. Terceira geração	22
3.2. iv. Quarta geração.....	23
4. Cenário atual.....	24
5. Metodologias para projetos de interfaces	26
5.1. Prototipação	26
5.2. IHC e Métodos ágeis	29
5.3. Outras metodologias e semelhanças	33
6. Tipos de aplicações e exemplos	35
6.1. Interfaces multimodais	35
6.1. i. Acessibilidade e vantagens.....	36
6.1. ii. Desvantagens e desafios.....	37
6.2. Outras soluções.....	37
7. Tendências e Big Data	39
8. Considerações finais	40

1. Introdução

No cenário atual, estamos vivendo uma verdadeira revolução tanto tecnológica quanto comportamental causada pela forma como o ser humano lida com os computadores e seus dispositivos. Essa revolução vai desde a configuração das relações entre pessoas consigo mesmas, com a sociedade, com os próprios aparelhos, Internet e programas.

As redes sociais e o sucesso dos dispositivos móveis, como *smartphones*, *tablets* e, mais recentemente, os *ultrabooks* provam que levar o usuário e suas opiniões em conta são fundamentais para um projeto obter reconhecimento no mercado.

No plano de fundo, para sustentar de uma forma sólida esse sucesso, está, entre outras coisas, um bom projeto de interface. É com foco no projeto das interfaces e dispositivos de entrada e saída que esse documento foi concebido.

No primeiro capítulo, intitulado Conceitos básicos, esse trabalho procura descrever definições fundamentais relacionados ao tema.

Já os objetivos do segundo capítulo é relacionar historicamente os principais pontos da evolução tecnológica pelos quais passamos, tanto pelo ponto de vista da evolução das ideias de interfaces durante os anos, relacionando com alguns dispositivos principais, quanto da linha do tempo das invenções que culminaram na concepção do microcomputador da forma como o conhecemos hoje. Ainda nesse capítulo, são relacionados alguns desafios que cada geração de interface enfrentou.

Em seguida, temos o capítulo, chamado Cenário atual, planejado para constar logo após o histórico para seguir a linha de raciocínio e, ao levar em consideração os pontos do tópico anterior, traçar uma breve perspectiva da configuração do contexto em que estamos inseridos. Assim como no segundo capítulo, também são abordados quais obstáculos permeiam o paradigma vigente.

Em Metodologias, são apresentadas algumas abordagens comuns aos projetos de interfaces, com a descrição de suas premissas básicas e alguns pontos a se levar em consideração ao optar por um método ou outro.

No quinto capítulo, são relacionados alguns novos tipos de aplicações relacionadas às novas tecnologias de interfaces, com alguns exemplos de sua empregabilidade, vantagens e desvantagens.

No penúltimo capítulo, são relacionadas tendências dos projetos de interfaces, considerando o cenário atual e outros aspectos mencionados anteriormente, além da introdução do conceito de Big Data e os desafios que impõe aos projetos de interfaces.

Finalmente, após esses tópicos, são feitas considerações finais sobre o tema, concluindo o trabalho.

2. Conceitos básicos

Não apenas no âmbito da análise dos sistemas como também no contexto geral de análise e, inclusive, entrando na própria definição do vocábulo, para se estudar um determinado assunto é essencial que haja a decomposição em partes menores de seus elementos constituintes. Dessa forma, torna-se possível ter uma visão mais clara do tema como um todo (DESCARTES, 2005).

Tendo esse pensamento em mente, antes de qualquer avaliação de cenário ou ponderações sobre as implicações das novas tecnologias de captação e consulta de dados, serão introduzidos conceitos básicos, a fim de ser possível entender com mais facilidade os aspectos técnicos que serão abordados posteriormente.

2.1. Sistema

Fundamental para qualquer trabalho ou estudo relacionado à área de Tecnologia da Informação (TI), um sistema, qualquer que seja, trata-se basicamente de um conjunto de componentes interligados uns aos outros realizando uma atividade a fim de atingir um determinado objetivo (BERTALANFFY, 2008).

É importante salientar que um sistema é composto por “N” outros sistemas, todos relacionados entre si, como a própria definição já explica. Essas subdivisões são chamadas de subsistemas (DESCARTES, 2005).

Dessa forma, entende-se o porquê é tão difícil discorrer sobre o estudo da Análise de Sistemas, visto que pode ser facilmente mesclado com a própria ideia de análise.

2.2. Interface

Em um âmbito geral, levando-se em consideração a definição contida em dicionários, pode-se assumir o termo interface como sendo o “[...] elemento que proporciona uma ligação física ou lógica entre dois sistemas ou partes de um sistema que não poderiam ser conectados diretamente.” (HOUAISS; FRANCO; VILLAR, 2001, p. 1.633).

Ora, seguindo esse raciocínio, pode-se dizer que a interface seria a comunicação entre sistemas heterogêneos os quais, pelo menos em um primeiro momento, não possuem a capacidade de se relacionarem entre si por eles mesmos. É nesse contexto que temos a interface, que surge para fazer o papel dessa ponte, desse “elemento de ligação” entre essas partes distintas.

2. 2. i. Interface humano-computacional

Ao recapitular a definição de sistema mencionada anteriormente e assumirmos que tratamos de dois conjuntos de elementos distintos, pessoas e máquinas, podemos chegar rapidamente ao conceito vinculado ao tema principal desse trabalho: a Interface humano-computacional (IHC), que é comumente referida em diversas fontes como interface homem-máquina (IHM) e assim também o será feito durante esse trabalho.

Seu significado é tão complexo que em seu livro, que foca justamente na questão das interfaces, Lévy (2006, p.176) as descreve objetivamente apenas nas últimas páginas de sua obra. De acordo com ele, pode-se dizer que é “[...] o conjunto de programas e aparelhos materiais que permitem a comunicação entre um sistema informático e seres humanos.”.

No entanto, para uma melhor elucidação, vamos adotar como padrão que essa interface não se refere apenas a um computador ou seus dispositivos, pois se assim o fizéssemos, estaríamos limitando demais o cenário da discussão que será levantada aqui. Por esse motivo, assumiremos a interface humano-computacional como sendo um projeto de comunicação

entre esses dois sistemas, constituída de diversas outras interfaces que possibilitam haver esse entendimento entre homens e máquinas.

Em um primeiro momento, duvida-se que esse assunto possa demandar alguma polêmica e tamanho esforço para ser compreendido, mas com o transcorrer deste documento, esses pontos de discussão serão levantados no momento propício.

2.3. Dispositivos

Continuando na mesma linha de raciocínio mantida até aqui, podemos considerar que dispositivos são interfaces, já que também são responsáveis pelo processo de comunicação. Entretanto, há uma diferença crucial que, em qualquer dicionário ou literatura básica sobre o assunto, é possível de ser percebida. Seu significado é empregado com um sentido bem mais concreto, sendo descrito sempre como um aparelho, uma peça, um mecanismo.

Muitas vezes, ao explicar sobre o processo de comunicação, há a demarcação de emissores, aqueles que transmitem a mensagem e os receptores, que decodificam a mensagem (SUANNO, 2012, p. 10).

Consultando literaturas de hardware, é possível perceber que um paradigma parecido também é aplicável na dinâmica dos dispositivos. Considerando a mensagem como um dado, é possível abstrair que os chamados dispositivos de entrada – os emissores – por onde são enviados os dados para serem codificados por um processador e os dispositivos de saída seriam, de certa forma, receptores, com a diferença que refletem a ação que está sendo executada (ALMEIDA, 2003).

Essa classificação dos dispositivos se deve porque, historicamente, as operações de entrada e saída eram fisicamente separadas (de um lado, um operador para alimentar a máquina e de outro, um operador para coletar os resultados). Esse vocabulário testemunhava essa distinção de funções. No entanto, com o passar do tempo, o avanço da tecnologia

proporcionou que esses dois extremos se aproximassem, mas a ideia geral ainda permanece (LÉVY, 2006, p. 177).

As descrições dessas classificações são relacionadas a seguir.

2. 3. i. Dispositivos de captação de dados

Não há como fugir da definição mais técnica das categorias de dispositivos, embora seus significados sejam bem intuitivos. Também chamados de dispositivos de entrada, são dispositivos que, a partir de um estímulo, interpretado por um processador e armazenado em memória, transforma esse estímulo em um dado, executando uma operação (SILVA, 2012).

A forma que isso se dá depende do dispositivo usado. Basicamente, esses dispositivos auxiliam o canal de comunicação no sentido “usuário -> computador”, pois os dados são codificados da linguagem humana para a computacional (ALMEIDA, 2003).

Pode-se citar como exemplo desse tipo de hardware o teclado, o mouse, microfone, câmeras *webcam*, *scanner*, *joystick*, *touchpad*, *dataglove*, entre outros.

2. 3. ii. Dispositivos de consulta de dados

Conhecidos também como dispositivos de saída, diferentemente da categoria anterior, permitem a comunicação no sentido oposto, ou seja, na direção “computador -> usuário”, já que exibem o dado previamente produzido pelo processador, permitindo a consulta do resultado de uma tarefa que foi executada (ALMEIDA, 2003; SILVA, 2012).

Como dispositivos dessa categoria, é válido mencionar as impressoras, os monitores, caixas de som, projetores, fones de ouvido.

2. 3. iii. Dispositivos híbridos

É a combinação das duas categorias anteriores, ou seja, são capazes de viabilizar o diálogo do sistema computacional e humano nos dois sentidos possíveis, no caso, “computador <-> usuário”.

Como avanço da tecnologia, esse tipo de dispositivo tem ganhado cada vez mais mercado, visto que é o melhor representante de uma ideia que tem estado em forte evidência nos últimos anos: a convergência. Esse assunto será retomado no capítulo sobre Tendências, visto que o objetivo por enquanto é apenas definir o conceito básico.

Exemplificando essa categoria, podemos citar os modems, discos rígidos, tablets, ultrabooks, smartphones, impressoras multifuncionais, joysticks vibratórios, câmeras fotográficas digitais, telas sensíveis ao toque, enfim.

2. 3. iv. Periféricos

Ainda com relação aos dispositivos de entrada e/ou saída, há outra denominação que pode ser citada, os chamados periféricos. De uma forma geral nesse contexto da informática, dispositivos e periféricos são sinônimos, mas vale ressaltar que os últimos são assim denominados porque se concentram instalados fora do núcleo principal de um computador, ou seja, em sua periferia e, portanto, seu vocábulo se originou dessa ideia (MONTEIRO, 2001, p. 372).

3. Histórico

Assim como uma calculadora, um computador consegue, por meio de diversas operações pré-programadas, apresentar uma resposta a uma situação que é submetida. Também é possível fazer um comparativo com a imprensa, que impulsiona e exercita a comunicação entre indivíduos (SUANNO, 2012).

A máquina que chamamos de computador a cada dia que passa consegue agregar mais funções e mais significados. Ponderando esse aspecto, de certa forma, quase todas as inovações tecnológicas até os dias atuais fariam parte do histórico de evolução do microcomputador da forma como conhecemos hoje.

3. 1. Breve histórico da computação

A era dos dispositivos mecânicos começa pela criação do ábaco, por volta do ano 500 a.C. que foi responsável por uma das primeiras formas de se lidar com as operações matemáticas, passa pela pascalina, mecanismo mais complexo, inventado milênios depois por Blaise Pascal e vê surgir, em 1834 a invenção responsável por tornar Charles Babbage conhecido como “Pai do Computador”: a máquina analítica que tinha memória, processador, saída para impressora e realização de programas. Esse conceito serviu como base para a concepção dos computadores atuais (MONTEIRO, 2001, p. 9-11).

A era dos dispositivos eletromecânicos testemunhou em 1889 o uso da máquina de tabulações, feita por Herman Hollerith, que utilizava cartões perfurados para armazenar dados. Com o sucesso do censo realizado, e juntamente com outras duas empresas, em 1924 houve o início da IBM (MONTEIRO, 2001, p. 11-12).

A era da evolução dos computadores eletrônicos teve início em 1946, quando houve o surgimento do ENIAC, o primeiro computador digital eletrônico de grande escala e

representante do grupo de computadores de primeira geração. Em seguida, vieram os computadores de segunda geração, que se caracterizaram por serem computadores transistorados; os de terceira geração, com computadores de circuitos integrados e os de quarta geração, com o início da miniaturização dos componentes, tornando possível o desenvolvimento dos microcomputadores pessoais, até então, inexistentes no mercado. (MONTEIRO, 2011, p. 12-18)

A seguir, uma tabela com a lista dos principais fatos que culminaram no conceito de computador que conhecemos hoje. Para mais detalhes históricos sobre o assunto, consultar Monteiro, em seu livro “Introdução à Organização de computadores”.

Tabela 1 - Linha do tempo da evolução dos computadores

Período	Evento
500 a.C.	Invenção e utilização do ábaco
1642 d.C.	Blaise Pascal cria sua máquina de somar.
1670	Gottfried Leibniz cria uma máquina de calcular que realiza as quatro operações aritméticas.
1823	Charles Babbage cria a máquina de diferenças, por contrato com a Marinha Real Inglesa.
1842	O mesmo Babbage projeta uma máquina analítica para realizar cálculos.
1889	Herman Hollerith inventa o cartão perfurado.
1890	Hollerith desenvolve um sistema para registrar e processar os dados do censo.
1924	Constituição da IBM.
1939	John Atanasoff projeta o primeiro computador digital.
1946	Término da construção do ENIAC.
1946	John von Neumann propõe que um programa seja armazenado no computador e projeta o IAS, implementando sua proposta.
1951	Termina a construção do primeiro computador comercial de propósito geral, o UNIVAC.
1956	Termina a montagem do primeiro computador transistorizado, o TX-0, no MIT.
1957	Uma equipe da IBM, liderada por John Backus, desenvolve a primeira linguagem de alto nível, Fortran, voltada para solucionar problemas matemáticos.
1958	A IBM lança o IBM-7090.
1958	Jack Kilby, na Texas Instruments, completa a construção do primeiro circuito integrado, contendo cinco componentes.
1962	Douglas Engelbart, do Stanford Research Institute, inventa o mouse.
1964	A IBM lança o IBM/360, primeiro computador a utilizar circuitos integrados.
1964	A linguagem Basic (Beginners All-purpose Symbolic Instruction Code) é desenvolvida por Thomas Kurtz e John Kennedy no Dartmouth College. Mais tarde, ela se torna popular devido ao lançamento do Altair com o interpretador desenvolvido por Bill Gates e Paul Allen, fundadores da Microsoft.
1965	Gordon Moore, diretor de pesquisa e desenvolvimento da empresa Fairchild Semiconductor, prevê que a densidade dos transistores e circuitos integrados dobraria a cada 12 meses nos 10 anos seguintes. Esta previsão foi atualizada em 1975, substituindo 12 meses por 18 meses e tornou-se conhecida como Lei de Moore.
1965	A IBM fabrica o primeiro <i>floppy disk</i> .
1967	A primeira versão do sistema operacional Unix é lançada, rodando em um computador DEC PDP-7. Este sistema foi escrito, a partir de 1969, no Bell Laboratories, por Dennis Ritchie e Ken Thompson.
1970	A linguagem Pascal é projetada por Nicklaus Wirth.
1971	A Intel lança o primeiro sistema de microcomputador, baseado no processador 4004, com desempenho de 60.000 operações por segundo e 2.300 transistores encapsulados.
1971	Dennis Ritchie, do Bell Labs, desenvolve a linguagem C.
1972	Gary Kildall escreve um sistema operacional na linguagem PL/M e o denomina CP/M (Control Program/Monitor).
1973	A Intel lança o processador 8080 de 2 MHz (primeiro lançamento em 1973), com 6000 transistores e 640.000 instruções por segundo. O CP/M é adaptado para o 8080 e a Motorola lança seu processador de 8 bits, o 6800.
1974	Na edição de janeiro da revista <i>Popular Electronics</i> é realizado o lançamento do primeiro microcomputador de 8 bits, o Altair.
1975	Steve Wozniak e Steve Job formam a Apple Computer.
1976	A DEC lança um de seus mais populares minicomputadores, o VAX 11/780.
1976	A Apple Company lança seu computador Apple II.
1976	Surge a primeira planilha eletrônica, Visicalc.
1977	A IBM anuncia o lançamento de seu primeiro microcomputador, o IBM-PC.
1979	A Apple apresenta seu primeiro computador do tipo Macintosh.
1981	A Microsoft lança sua planilha Excel, o primeiro aplicativo para o Windows.
1984	A Microsoft lança seu sistema operacional Windows para IBM-PCs.
1987	A Microsoft lança a versão 3.0 do Windows para PCs.
1989	A AMD lança seu clone do processador Intel 386.
1990	A IBM e a Motorola estabelecem um acordo para desenvolvimento do microprocessador PowerPC.
1991	A IBM lança um microcomputador portátil, o ThinkPad 700C.
1992	Linus Torvalds desenvolve o sistema operacional Linux, na Finlândia.
1992	A NCSA desenvolve o primeiro navegador para Internet, o Mosaic.
1993	O Mosaic se transforma no Netscape.
1994	A Iomega lança seus Zip drives.
1995	Inicia-se o contencioso entre a Microsoft e o governo dos EUA, que dura até os dias atuais.
1996	Aparecem no mercado os CD-RW (CD que podem ser regravados).
1998	A Compaq adquire a DEC.

3. 2. Breve histórico das interfaces e seus desafios

Considerando a evolução do computador, é claro notar que, inicialmente, a questão das interfaces era uma preocupação praticamente inexistente nos projetos concebidos. O foco era totalmente voltado para a funcionalidade do mecanismo.

Essa mentalidade era justificável principalmente porque os computadores eram usados essencialmente por usuários especialistas e seu uso não era largamente difundido. A entrada de dados nos tempos dos primórdios do computador era feita, basicamente, por cartões e fitas perfurados (SILVA, 2012).

3. 2. i. Primeira geração

A relevância da questão das interfaces passou a ser uma realidade principalmente depois da popularização dos computadores pessoais, visto que o uso da tecnologia não estava mais restrito apenas a engenheiros, matemáticos, programadores, enfim, usuários especialistas. Aos poucos, o fato de que qualquer pessoa poderia ter um computador em casa foi se consolidando e com isso, interfaces foram evoluindo para atender ao novo paradigma que foi se estabelecendo.

Nessa época, a principal forma de interação entre homem e máquina era a exibição em display única e exclusivamente de textos, através das linhas de comando em que, por meio de um teclado, eram digitados caracteres que, juntos, formavam um comando pertencente a uma linguagem de programação (OLIVEIRA, 2010; LEITE, 2012).

Entre outras palavras, invariavelmente a comunicação era prejudicada, visto que era premissa para comunicação do usuário com o computador o pleno domínio de uma linguagem de comandos para programação da máquina. Apenas dessa forma, a comunicação adequada ocorreria.

Dessa forma, ruídos nesse processo de comunicação não são raros, mesmo nos dias de hoje, e isso se deve pelo fator humano quanto pelo fator computacional, representando mais uma dificuldade na interação.

3. 2. ii. Segunda geração

Uma evolução das interfaces por linha de comando e consulta é a interface de menu simples, onde são apresentadas opções de ação executadas após a entrada de um código com a decisão do usuário. Ainda permanecia o padrão de interface textual, pelo menos em um primeiro momento (PRESSMAN, 1995; LEITE, 2012).

Nesse momento, a preocupação com as interfaces e o modo como o usuário enxergava o sistema computacional começava a ganhar destaque, já que durante sua utilização, embora houvesse menos erros por haver opções pré-determinadas para a tomada de decisão, muitas vezes, o usuário sentia a navegação por menus cansativa e tediosa, representando um obstáculo na difusão da tecnologia e na arquitetura dos projetos (PRESSMAN, 1995).

3. 2. iii. Terceira geração

Com o avanço da tecnologia e da capacidade do hardware, surgiu o padrão de interface gráfica. É nesse momento que passa a existir a base do conceito que permanece até os dias atuais: *Graphical User Interface* (GUI) ou Interface gráfica do usuário (LEITE, 2012).

O principal representante dessa fase na categoria dos periféricos é, além do teclado e monitor, o mouse, já que a interatividade se faz através de símbolos gráficos e das ideias *point and pick* (apontar e escolher), *scrolling* (deslizar), *drag and drop* (arrastar e soltar). A ideia principal é a manipulação direta de objetos (PRESSMAN, 1995; LEITE, 2012).

O desafio fundamental dessa geração de interfaces começa a girar em volta de fatores menos específicos e técnicos, já que houve a diminuição expressiva de digitação, se comparada às gerações passadas.

Com a introdução da simbologia icônica, o usuário precisou se adaptar e aprender uma nova linguagem, tanto simbólica quanto gestual, completamente diferente das formas vigentes, para poder assim, interagir com as máquinas de uma forma mais amigável.

3. 2. iv. Quarta geração

A IHC dessa geração costuma combinar o paradigma anterior, também chamado de *Windows, Icons, Menus, Pointing Device* (WIMP) – ou seja, Janelas, Ícones, Menus e Pontos de Indicação – com hipertexto e a capacidade do usuário de poder executar mais de uma tarefa cognitiva e/ou comunicativa simultaneamente. (PRESSMAN, 1995, p. 607-609).

4. Cenário atual

Considerando o capítulo anterior, podemos entender melhor a situação em que nos encontramos atualmente.

Tendo em vista o histórico da computação e, principalmente, das interfaces, é interessante perceber que todas as gerações de interfaces ainda estão presentes nos dias de hoje. Refletindo esse cenário, pode-se dizer também que os desafios encontrados por cada geração também foram herdados pelas seguintes e, na busca de melhorias de comunicação com o usuário e na preocupação com tópicos novos, como acessibilidade, mobilidade, os desafios passaram sempre a ter a tendência de se acumular.

Podemos assumir que estamos em uma quinta geração de interfaces, com um novo conceito que, aos poucos, está tomando o lugar da antiga GUI na concepção de projetos. A Natural User Interface (NUI) ou Interface Natural do Usuário visa à máxima aproximação da configuração da interação humano-máquina à configuração humano-humano (ALMEIDA, 2012).

Basicamente, se no paradigma anterior se fazia necessário entender o máximo possível a linguagem da máquina e o funcionamento dos dispositivos, nessa nova concepção, ocorre o contrário: são as máquinas que estão cada vez mais adaptadas e programadas para reconhecer o comportamento e linguagens humanos.

Dessa forma, o processo de interação entre esses dois sistemas tão distintos, ao menos sob a nossa perspectiva, está conseguindo ser menos traumática, através de produtos que facilitem essa comunicação. Um exemplo é o conceito de interação por ondas cerebrais.

Além disso, não há como deixar de mencionar que a expressividade das tecnologias móveis – representados pelos smartphones, celulares, tablets, ultrabooks e, inclusive, videogames portáteis – nos últimos anos representou uma verdadeira revolução na

configuração em que pensamos e projetamos interfaces e na forma de interação das pessoas com outras pessoas e até mesmo com um ambiente ou máquinas.

A incorporação nesses e em outros dispositivos mais clássicos de processadores de alto desempenho, sistemas de posicionamento global, conexões wireless à Internet e a outros dispositivos, telas multitoque e alta qualidade gráfica também contribuíram para a formação de um pensamento voltado à interatividade e um olhar muito mais cuidadoso à questão das interfaces, dessa vez, não limitada somente a hardware ou a software, mas ao conjunto de todas essas características mencionadas. (TRIANA GOMEZ, 2012, p. 2)

Atualmente, não é necessário estar em uma mesa que tenha obrigatoriamente acesso à corrente elétrica e espaço para comportar todos os dispositivos comuns ao se utilizar um computador. Com os dispositivos móveis, somos capazes de realizar tarefas tão complexas quanto em um computador *desktop*, com a diferença podermos estar em, praticamente, qualquer lugar. Isso representa a grande mobilidade proporcionada por essa tecnologia. (TRIANA GOMEZ, 2012, p. 3)

5. Metodologias para projetos de interfaces

Um dos grandes obstáculos para a construção de um bom projeto é a metodologia a ser adotada e embora atualmente o foco, aos poucos, esteja se voltando para a importância das interfaces e da boa comunicação do usuário com a solução oferecida, essa questão ainda encontra dificuldades ao ser abordada.

E esse é exatamente o primeiro ponto que pode cativar ou afastar o usuário da tecnologia que está sendo apresentada, muitas vezes determinando o sucesso ou o fracasso de um intenso período de planejamento.

O primeiro contato que o usuário final tem é mediado justamente pelas interfaces que,

É, de muitas maneiras, a embalagem do software de computador. Se ela for fácil de aprender, simples de usar, direta e amigável, o usuário estará inclinado a fazer bom uso daquilo que está dentro. Se ela não tiver nenhuma dessas características, invariavelmente, surgirão problemas. (PRESSMAN, 1995, p. 602).

Dentro desse contexto, ninguém melhor que o próprio usuário para opinar sobre o andamento do projeto e avaliar se está de acordo com os requisitos, ajudando a evoluí-lo. Por esse motivo, uma das metodologias mais utilizadas ainda é a prototipação.

5.1. Prototipação

Um protótipo nada mais é que uma versão simplificada do sistema que será utilizado, com o objetivo de melhorar a percepção do usuário final sobre pontos importantes como funcionalidade, complexibilidade, acessibilidade, layout, enfim.

Já no primeiro levantamento de requisitos, pode-se utilizar o primeiro protótipo, geralmente, arquitetado em uma folha de papel. Segundo a professora Silvia Dotta da UFABC (Universidade Federal do ABC) em sua apresentação online, há a vantagem de ser um meio barato, acessível e prático, além disso, através dele, é possível coletar uma avaliação muito rica por parte do usuário nesse primeiro momento, já que o obriga a considerar o conteúdo, as

funcionalidades etc. Apesar do papel ser a forma mais comum, nada impede que sejam empregados outros meios; ainda mais que, com o avanço da tecnologia, ficou mais prático carregar aparelhos como smartphones, tablets, etc para essa tarefa. De toda forma, nessa etapa, o essencial é a praticidade e rapidez para simular e estimular ideias para o projeto.

Esse protótipo inicial é chamado de protótipo de baixa fidelidade. Pode ser um storyboard ou esboço e nunca é entregue como parte do projeto, servindo apenas como ferramenta para captação de novas ideias. (PREECE; ROOGERS; SHARP, 2005, p. 263).

Nada impede que protótipos de baixa fidelidade sejam utilizados em diversas etapas do projeto, no entanto, conforme a solução passe a ser mais bem definida e ficar mais clara, tomando forma, é natural que protótipos mais complexos sejam usados (AUDY; PRIKLADNICK, 2008, p. 16).

São os intitulados protótipos de alta fidelidade e se diferenciam dos primeiros por se assemelharem mais à versão final da solução projetada, simulando as funcionalidades e os aspectos físicos. Justamente por possuírem um nível de detalhe maior, demandam mais tempo e mais recursos. Abaixo, segue uma tabela comparativa entre esses dois tipos de prototipação:

Tabela 2 - Prototipagem: baixa fidelidade x alta fidelidade

TIPO	VANTAGENS	DESVANTAGENS
Protótipo de baixa fidelidade	<ol style="list-style-type: none"> 1. Custo mais baixo de desenvolvimento 2. Avalia múltiplos conceitos de design 3. Instrumento de comunicação útil 4. Aborda questões de leiaute de tela 5. Útil para identificação de requisitos de mercado 6. <i>Proof-of-concept</i> 	<ol style="list-style-type: none"> 1. Verificação limitada de erros 2. Especificação pobre em detalhe para codificação 3. Uso conduzido pelo facilitador 4. Utilidade limitada após estabelecimento dos requisitos 5. Utilidade limitada para testes de usabilidade 6. Limitações de fluxo e navegação
Protótipo de alta fidelidade	<ol style="list-style-type: none"> 1. Funcionalidade completa 2. Totalmente interativo 3. Uso conduzido pelo usuário 4. Define claramente o esquema de navegação 5. Uso para exploração e teste 6. Mesmo look and feel do produto final 7. Serve como uma especificação viva 8. Ferramenta de vendas e marketing 	<ol style="list-style-type: none"> 1. Desenvolvimento mais caro 2. Sua criação demanda tempo 3. Ineficiente para designs <i>proof-of-concept</i> 4. Não serve para coleta de requisitos.

Fonte: Adaptado de PREECE; ROOGERS; SHARP, 2005, p. 266 apud RUDD et al., 1996.

Com a premissa de que todas as abordagens de desenvolvimento se iniciam com a coleta de requisitos, é claro entender que a estrutura do modelo de prototipação é circular.

Primeiramente, a partir desse ponto, há a construção de um projeto rápido, seguindo para a criação de um protótipo de baixa fidelidade, geralmente, como dito anteriormente, logo nos primeiros contatos com o usuário.

Na próxima etapa, ocorre a avaliação desse protótipo inicial pelo usuário, entregando um *feedback* das suas impressões sobre o produto.

Com essas observações e lista de melhorias em mãos, o projetista e o analista fazem um refinamento do protótipo e uma nova coleta de requisitos (PRESSMAN, 1995, p. 35-38).

Dessa forma, entra-se em um processo de constante melhoria, com a evolução dos protótipos de baixa fidelidade para alta até não haverem mudanças a serem feitas. Nesse momento, o produto está pronto e, finalmente, poderá ser construído.

A seguir, há um esquema que resume bem as etapas da prototipação e seu sequenciamento.

Figura 1 – Estrutura de prototipação



Fonte: adaptado de PRESSMAN, 1995, p. 36

Um dos grandes desafios para projetistas que adotam essa metodologia é deixar claro que o protótipo não é o sistema final e que deverá ser descartado a fim de garantir a qualidade e o funcionamento de um bom produto (PRESSMAN, 1995, p. 38).

5. 2. IHC e Métodos ágeis

Assim como a prototipação, a filosofia ágil também possui como princípio fundamental para o desenvolvimento a interação com o cliente. Relacionando com a IHC, não é difícil entender o motivo desse conjunto de métodos, que focam na comunicação, estarem ganhando tanto espaço no desenvolvimento dos projetos de interfaces, afinal, por terem esse foco, conseguem captar melhor o que é necessário para que um produto proporcione uma boa experiência de uso e consiga atingir melhor metas como acessibilidade, mobilidade, entre outros.

Apesar de clara similaridade, há alguns pontos de atenção ao se adaptar a concepção de IHC às metodologias ágeis. Um dos principais conflitos de conceitos é com relação aos clientes e usuários. Para IHC, são papéis completamente distintos, enquanto para um conjunto de metodologias ágeis, há apenas a denominação de “cliente”. O problema disso é que, como consequência disso, a usabilidade da solução tecnológica acaba ficando em segundo plano pois se considerarmos que esses dois personagens possuem papéis distintos, é perceptível a divergência de visões tidas por cada um: enquanto o cliente está focado em obter o produto de um modo funcional, o usuário, além dessa preocupação, volta sua atenção para outros aspectos. (ROSEMBERG; SCHILLING, 2011, p. 1)

Ao estudar a metodologia ágil, termo que trataremos como o conjunto de metodologias que tem foco no desenvolvimento ágil, é possível entender, através de diversas literaturas sobre esse assunto, que fundamentalmente sua concepção acaba fazendo uma oposição e

servindo de contraponto ao método tradicional, o conhecido modelo cascata, ou waterfall (PEREIRA, 2012).

O modelo tradicional é conhecido por possuir, fundamentalmente, seis principais fases em seu ciclo de vida (Estudo, Análise, Projeto, Codificação, Testes e Implantação). Todas essas fases são percorridas de uma maneira sistemática, linear, sequencial. Uma fase não inicia enquanto a anterior não estiver concluída. Além disso, funciona como um método preditivo, e que uma análise de requisitos bem feita e um planejamento adequado são essenciais para o sucesso do projeto (PRESSMAN, 1995).

Figura 2 – Estrutura básica do ciclo de vida ágil



Fonte: adaptado de PAVÃO, 2012

Apenas com essa definição, é possível identificar qual um dos principais defeitos dessa metodologia. Caso ocorra algum atraso em uma dessas fases, ocorrerá atraso em todo o projeto. Quanto mais atrasos houverem, pior será o resultado final e a diferença entre o tempo estimado e o realizado ficará cada vez maior. Também pode-se citar como um dos pontos fracos do modelo cascata justamente seu ponto forte: embora seja focado na organização de

processos e no estabelecimento de fases bem definidas, o modelo não está preparado para um ambiente dinâmico e com intensas mudanças de escopo de projetos ou de requisitos do cliente (PEREIRA, 2012).

É justamente para evitar esses tipos de problemas que os métodos ágeis se desenvolveram. Diferentemente do modelo tradicional, possui uma base adaptativa, receptiva às mudanças.

A metodologia ágil se baseia principalmente na interação entre a equipe, na entrega rápida de um produto funcional que esteja em conformidade com os requisitos, no cliente e na capacidade de absorver variações de escopo ou alterações de requisitos (KENT et al, 2012).

Segundo Pereira (2012), os agilistas, termo que os próprios desenvolvedores e projetistas que seguem a metodologia ágil se atribuem, possuem alguns princípios básicos, descritos no manifesto ágil, transcrito a seguir:

- [...] satisfazer o cliente, através da entrega adiantada e contínua de software de valor.
- Aceitar mudanças de requisitos, mesmo no fim do desenvolvimento. Processos ágeis se adequam a mudanças, para que o cliente possa tirar vantagens competitivas.
- Entregar software funcionando com frequência, na escala de semanas até meses, com preferência aos períodos mais curtos.
- Pessoas relacionadas à negócios e desenvolvedores devem trabalhar em conjunto e diariamente, durante todo o curso do projeto.
- Construir projetos ao redor de indivíduos motivados. Dando a eles o ambiente e suporte necessário, e confiar que farão seu trabalho.
- O Método mais eficiente e eficaz de transmitir informações para, e por dentro de um time de desenvolvimento, é através de uma conversa cara a cara.
- Software funcional é a medida primária de progresso.
- Processos ágeis promovem um ambiente sustentável. Os patrocinadores, desenvolvedores e usuários, devem ser capazes de manter indefinidamente, passos constantes.
- Contínua atenção à excelência técnica e bom design, aumenta a agilidade.
- Simplicidade: a arte de maximizar a quantidade de trabalho que não precisou ser feito.

- As melhores arquiteturas, requisitos e designs emergem de times auto-organizáveis.
- Em intervalos regulares, o time reflete em como ficar mais efetivo, então, se ajustam e otimizam seu comportamento de acordo. (KENT et al, 2012)

É importante ressaltar que os agilistas não desprezam a documentação, mas pregam que seja não produzida de forma tão abrangente quanto o modelo clássico e sim, que seja coerente com o que é realmente consultado pelos *stakeholders*.

Seguindo essas premissas fundamentais, diversas metodologias surgiram, dentre as quais, segue um breve o comparativo descritivo entre as principais: XP (eXtreme Programming), SCRUM, FDD (Feature Driven Development) e ASD (Adaptive Software Development).

Tabela 3 - Comparação entre principais metodologias ágeis

Métodos / Fases	XP	SCRUM	FDD	ASD
Requisitos Iniciais	User stories escrita pelo cliente	Os requisitos são listados originando o Product Backlog	Os requisitos são listados, definidos e documentados através de casos de uso (diagramas de classe e sequência)	Sessões de JAD com a presença de representantes dos clientes
Atribuir Requisitos aos incrementos	Definição das "User Stories" que serão desenvolvidas a cada iteração	Os requisitos definidos no Product Backlog são alocados às Sprints durante a Reunião de Planejamento da Sprint.	Requisitos agrupados e priorizados conforme importância e dependência	É definido a quantidade de iterações e requisitos que serão implementados em cada uma delas
Projetar Arquitetura do Sistema	Paralelo ao desenvolvimento da User Stories	Projeto geral baseado no Product Log	Diagrama de Classes para representar a arquitetura do sistema	Nenhuma atividade relacionada ao Projeto de Arquitetura.
Desenvolver Incremento de Sistema	Implementação das user stories que fazem parte da iteração corrente por dupla de programadores	Implementação dos requisitos contemplados no Sprint Backlog Reuniões diárias de 15 – 30 minutos	Acorre após análise da documentação existente, criação de diagrama de sequência e revisão do modelo gerado durante levantamento de requisitos e projeto de arquitetura	Requisitos desenvolvidos dentro de suas respectivas iterações e simultaneamente

Tabela 4 - Comparação entre principais metodologias ágeis (cont.)

Metodos / Fases	XP	SCRUM	FDD	ASD
Validar Incremento	Programadores: testes de unidade Cliente: testes de aceitação Ambos os testes são escritos antes da codificação e executados após	Acontece ao final da Sprint	Testes efetuados pelos programadores ao final de cada iteração	Grupo de clientes são definidos para revisar e testar a aplicação. Programadores verificam a qualidade do código.
Integrar Incremento	Código é integrado a medida que vai sendo desenvolvido	Acontece ao final de cada Sprint	Ocorre no final de cada iteração após os testes e inspeções	Nenhuma atividade sugerida
Validar Sistema	Sistema é validado pelo Cliente	Sistema é validado no último dia de cada Sprint (Revisão da Sprint)	Validação acontece através das inspeções e dos testes de integração	Nenhuma atividade específica é sugerida
Entrega Final	Cliente Satisfeito	Todos os itens do Product Backlog desenvolvidos	Todos os requisitos devem passar por todas as fases de Projeto e construção	Todos os requisitos desenvolvidos

Fonte: LIBARD, BARBOSA, 2010, p. 19

5.3. Outras metodologias e semelhanças

Além das duas tecnologias citadas, há diversas outras com um forte apelo a projetos de IHC, todas com essas mesmas similaridades. A seguir, uma breve descrição de algumas, baseadas nos conceitos expostos por Barbosa e Silva (2010) de algumas dessas outras abordagens.

- **Modelo estrela:** A ideia central desse modelo é a avaliação. Todas as outras fases (Implementação, prototipação, Análise de tarefas/funcional, Definição de requisitos/especificações e Representação formal do design) podem passar a qualquer momento pela Avaliação, por esse motivo, esse modelo foi chamado assim e é representado como uma estrela. Problema: difícil gerenciamento.

- ***Engenharia de usabilidade Mayhew***: Baseado em três etapas essenciais: Análise de requisito, Projeto/desenvolvimento/teste e instalação, facilita o colaborador que não está familiarizado com questões de usabilidade, já que envolve diversos detalhes complexos da solução.
- ***Design baseado em cenários***: Análise de problema (utilizando seu respectivo cenário), projeto de solução IHC (através dos cenários de atividade, informação e interação), Prototipação e avaliação da solução proposta são as atividades básicas dessa abordagem.

Um importante ponto a se refletir em uma metodologia com foco em IHC é que ela deve ser iterativa, ou seja, ter diversos ciclos. Dessa forma, o projetista ou analista em a oportunidade de evoluir a solução concebida e resolver problemas o mais rápido possível.

Para um projeto de interfaces não é viável adotar um modelo cascata, por exemplo. Se houver algum erro ou divergência no entendimento dos requisitos logo no início do projeto, muito tempo será gasto em um projeto fadado a morrer por não atender o usuário.

Nesse ponto, para a IHC é imprescindível o *feedback* do usuário tão logo quanto possível pois é fator determinante para o sucesso da tecnologia uma boa receptividade, seja hardware ou software.

6. Tipos de aplicações e exemplos

Aspecto tão importante quanto a metodologia a ser utilizada para a criação de um projeto de interface humano-computacional é a forma que a comunicação entre o mundo do usuário e o mundo da informática será feita. Atrelado a isso, temos que levar em conta características do público-alvo da aplicação, bem como, qual o dispositivo ideal que terá o mecanismo necessário para proporcionar a esse usuário uma experiência que seja o mais agradável e amigável possível.

A maioria da população possui diversas maneiras de perceber e se comunicar com o mundo a sua volta. Focando nisso, diferentes soluções podem ser desenvolvidas tendo como alvo uma ou mais dessas vias de comunicação, inerentes ao ser humano.

6.1. Interfaces multimodais

Basicamente, são interfaces que utilizam uma ou mais modalidades sensoriais, conhecidos como “sentidos”. Dessa forma, as soluções tecnológicas dessa categoria costumam combinar um ou mais canais de comunicação de uma vez, como linguagem escrita, linguagem verbal, expressões faciais, gestos, olhares, toques para a entrada e consulta de dados (FONSECA; JORGE, 2012).

Dependendo da quantidade de canais de comunicação utilizados, uma interface multimodal pode ser considerada Exclusiva – constituem aquelas em que, embora existam diversas modalidades disponíveis, apenas uma é utilizada – ou Sinérgica – grupo de produtos que utilizam mais de uma modalidade sensorial tanto na captação de dados quanto na consulta e saída (FONSECA; JORGE, 2012).

Seu objetivo é se aproximar o máximo possível da linguagem natural dos seres humanos e com isso, diminuir substancialmente o tempo de aprendizagem e a necessidade de perícia para a interação com a máquina.

Isso representa um importante passo para o aumento do acesso de determinados grupos de pessoas, geralmente excluídas, ao mundo da tecnologia.

6. 1. i. Acessibilidade e vantagens

Um importante ponto a se citar sobre tecnologias que utilizam interfaces multimodais é seu enorme poder de inclusão digital. Há diversas aplicações que são voltadas desde a área médica, com foco nos pacientes ou foco nos profissionais de saúde, até a área educacional, onde consegue promover a conscientização de grupos de pessoas que possuem acesso restrito a recursos como escolas, bibliotecas, universidades, enfim.

Diferente do paradigma comum atual, GUI e WIMP, as interfaces multimodais possuem a vantagem de, se comparadas, representarem obstáculo bem menor aos usuários principiantes que, segundo PRESSMAN (1995, p. 611), não possuem conhecimento do know-how necessário para utilizar um sistema computacional ou aplicação.

Isso pode ser percebido principalmente porque, como foi mencionado anteriormente, estão mais próximas à linguagem natural humana. Para navegar ou fazer uso de uma interface multimodal, basta saber falar, se expressar, gesticular, tatear, enfim. Além disso, dependendo da aplicação, não se faz necessário dominar todas as modalidades sensoriais.

Para as pessoas com mobilidade reduzida ou algum tipo de deficiência, é um novo mundo ao qual passam a ter acesso podendo, inclusive, estimular na sua recuperação ou no modo como se comunicam com o mundo.

Exemplo é o caso do físico Stephen Hawking, que possui uma doença neurodegenerativa e está quase totalmente paralisado. Para se comunicar com o mundo e

continuar ativo na comunidade científica, sempre testa diversos meios de se comunicar e, inevitavelmente, já utilizou diversas aplicações as interfaces multimodais.

Atualmente, a comunicação se dá através de um dispositivo infravermelho, acoplado a um par de óculos, que capta pequenas contrações abaixo do olho direito do físico, única região que ainda possui movimentos, e as transforma em conjuntos de caracteres para que ele consiga passar sua mensagem (LANGE, 2012).

Funcionando em conjunto com esse mecanismo, há também um sintetizador de voz, que reproduz as mensagens decodificadas pelos óculos. Também há uma *webcam* e outros monitores especiais acoplados que conseguiram devolver a Hawking, mesmo que de maneira artificial, a capacidade de se comunicar e transmitir suas ideias (DUNCAN, 2012).

6. 1. ii. Desvantagens e desafios

A maioria de aplicações de interfaces multimodais ainda precisa de diversos ajustes e personalizações, sendo restrita a reconhecer uma quantidade limitada de tipo de entrada de dados e conseguir processar, na maioria dos sistemas, a entrada de dados apenas de forma sequencial. Esse conjunto de restrições pode tornar a solução maçante e, embora se aproxime mais da linguagem natural humana que digitar caracteres. Dependendo da maneira como a interface for programada, pode se tornar pouco intuitiva se for necessário memorizar muitos comandos pré-determinados, por exemplo (FONSECA; JORGE, 2012).

6. 2. Outras soluções

O estudo das interfaces representa uma vasta área a ser explorada, não só atrelada à análise e desenvolvimento de sistemas, mas também a outras vertentes do conhecimento. É um tema muito rico e ainda possui diversas descobertas a serem feitas. Algumas dessas

descobertas, como as novas tecnologias de aplicações, já estão revolucionando o modo como compreendemos o mundo. Entre os tipos de aplicações, pode-se citar ainda interação cérebro-computador, realidade aumentada e a holografia.

Embora soluções como as de Stephen Hawking ainda não estejam facilmente acessíveis a todo o mercado, há diversos estudos que procuram encontrar soluções para problemas de acessibilidade, mobilidade ou interatividade.

7. Tendências e Big Data

Em tipos de aplicações, através do caso exemplificado, pode-se perceber também que é necessário obter formas melhores de se relacionar e reconhecer padrões de entrada de dados e a relação existente entre eles, podendo influenciar positivamente em casos como o de Hawking.

No capítulo de metodologias foram citadas algumas abordagens que buscam acompanhar esse processo tão dinâmico, mas mesmo assim ainda deve-se ter em mente que os dados produzidos hoje com a Internet, o uso de smartphones, tablets, notebooks e qualquer tecnologia similar são gigantescos.

De acordo com levantamento da Cisco (2012), estima-se que em 2015, esse número chegará aos incríveis 966 exabytes (cada exabyte equivale a um milhão de terabytes).

O conceito de Big Data surge a fim de tornar esses dados acessíveis e trabalháveis em menos tempo. Hoje em dia, o processo de arquitetura e engenharia de software não está preparado para abraçar essa mudança de paradigma (COLLET, 2012).

O Big Data está atrelado a três pontos básicos: variedade, englobando dados estruturados e não estruturados, volume, pela grande quantidade de dados sendo processados, analisados e mapeados e velocidade, ou seja, rapidez de processamento das informações.

Diante do chamado "Big Data", um grupo emergente de tecnologias de mineração de dados, aliada à supercomputação, está transformando o armazenamento, a manipulação e a análise de dados em tarefas mais baratas e mais rápidas. E tornando grandes massas de dados disponíveis para consulta pelas empresas, mudando a maneira como muitas delas fazem negócios (COLLETT, 2012).

Levando em conta essa informação, muitas oportunidades para explorar esses dados surgirão, mas será que as empresas estão preparadas para lidar com todos esses dados? Esse é um dos desafios e tendências de mercado para os próximos anos.

8. Considerações finais

Na era em que estamos vivendo, a preocupação com a interação humano-computacional ocupa cada vez mais o cotidiano da sociedade como um todo, não só dos estudiosos acadêmicos, que sempre buscam novos conceitos e aplicações, como também do usuário comum, o qual ainda esbarra em problemas e limitações já conhecidos, como Interface não amigável, dificuldades de aprendizagem para manipulação dos recursos disponíveis, pouca praticidade, restrições quanto à acessibilidade, entre outros.

Esses fatores representam verdadeiros desafios a serem superados, principalmente, pelos projetistas e equipes de desenvolvimento, que tem a importante missão de fazer a interligação entre esses dois mundos: teórico e real.

Além disso, elaborar um projeto de IHC adequado, usando dispositivos de fácil aprendizagem e metodologias coerentes para um bom produto de software são fatores determinantes no sucesso ou fracasso da disseminação de uma nova solução tecnológica.

Dentro dessa perspectiva, em que os dados são consultados a todo instante através de diversas formas de interação, é necessário que os desenvolvedores, gestores e *stakeholders* em geral questionem se as metodologias existentes atualmente para arquitetura de sistemas estão acompanhando ou, ao menos, se adaptando a esse avanço.

Interface não é só componente ou software. Um projeto de interface deve ser trabalhando levando em consideração diversos aspectos, principalmente os aspectos de transformações do ambiente, dados e interação humana.

Referências

1. DESCARTES, René. **O discurso do método**. Porto Alegre: L&PM, 2005.
2. BERTALANFFY, Ludwig Von. **Teoria Geral dos Sistemas**: fundamentos, desenvolvimento e aplicações. Petrópolis: Vozes, 2008.
3. HOUAISS, Antônio; FRANCO, Francisco Manoel de Mello; VILLAR, Mauro de Salles. **Dicionário Houaiss da língua portuguesa**. Rio de Janeiro: Objetiva, 2004.
4. LÉVY, Pierre. **As tecnologias da inteligência**: o futuro do pensamento na era da informática. São Paulo: 34, 2006.
5. MONTEIRO, Mario A. **Introdução à organização de computadores**. 4ª ed. Rio de Janeiro: LTC, 2001.
6. PRESSMAN, Roger. **Engenharia de Software**. 3ª ed. São Paulo: Makkron Books, 1995.
7. PREECE, Jeeny; ROOGERS, Yvonne; SHARP, Helen. **Design de interação**: Além da interação homem-computador. Porto Alegre: Bookman, 2005.
8. AUDY, Jorge; PRIKLADNICK, Rafael. **Desenvolvimento distribuído de software**: Desenvolvimento de software com equipes distribuídas. Rio de Janeiro: Campus/Elsevier, 2008.
9. BARBOSA, Simone Diniz Junqueira; SILVA, Bruno Santana. **Interação Humano-computador**. Rio de Janeiro: Campus/Elsevier, 2010.
10. LEITE, Jair C. **Design de Interfaces de Usuário**. Disponível em: <<http://www.dimap.ufrn.br/~jair/ES/c6.html>>. Acesso em: 23 mai. 2012.
11. BECK, Kent. **Manifesto para o desenvolvimento ágil de software**. Disponível em: <<http://manifestoagil.com.br/principios.html>>. Acesso em: 9 jun. 2012.
12. LIBARDI, Paula L. O; BARBOSA Vladimir. **Métodos ágeis**. Limeira, 2011. Disponível em: <http://www.ft.unicamp.br/liag/Gerenciamento/monografias/monografia_metodos_ageis.pdf>. Acesso em 12 jun 2012.

13. PAVÃO, Felipe. **Desenvolvimento ágil e a Toplab**. Disponível em: <<http://www.toplab.me>>. Acesso em 12 jun. 2012.
14. ROSEMBERG, Carlos; SCHILLING, Albert. **Integrando IHC e Métodos ágeis**. Fortaleza, 2011. Disponível em: <http://www.cin.ufpe.br/~ihc_clihc2011/docs/tutoriais/clihc2011_submission_183.pdf>. Acesso em 12 jun. 2012.
15. CISCO Visual Networking Index: Forecast and Methodology 2011-2016. Disponível em: <http://www.cisco.com/en/US/solutions/collateral/ns341/ns525/ns537/ns705/ns827/white_paper_c11-481360_ns827_Networking_Solutions_White_Paper.html>. Acesso em: 5 jun. 2012.
16. TRIANA GOMEZ, Edwin Miguel. **Reconhecimento de fala para navegação em aplicativos móveis para português brasileiro**. 2011. Dissertação (Mestrado em Ciência da Computação) - Instituto de Matemática e Estatística, Universidade de São Paulo, São Paulo, 2011. Disponível em: <<http://www.teses.usp.br/teses/disponiveis/45/45134/tde-19032012-211153>>. Acesso em: 13 jun. 2012.
17. PEREIRA, Alessandro Francisco. **Metodologia de desenvolvimento de software tradicional x ágil**. Disponível em: <<http://www.gonow.com.br/blog/2011/10/19/diferencas-entre-metodologia-tradicional-e-agil-de-desenvolvimento-de-software-na-semana-de-tecnologia-de-itapetininga/>>. Acesso em 15 jun. 2012.
18. SUANNO, Marilza Vanessa Rosa. **Novas Tecnologias de Informação e Comunicação: reflexões a partir da Teoria Vygotskyana**. Disponível em: <<http://www.abed.org.br/seminario2003/texto16.htm>>. Acesso em: 25 mai. 2012.
19. ALMEIDA, Carlos Cândido. Novas Tecnologias e Interatividade: além das interações mediadas. **DataGramZero - Revista de Ciência da Informação**, vol. 4, n. 4, ago/2003. Disponível em: <http://www.dgz.org.br/ago03/Art_01.htm#Nota01>. Acesso em: 30 mai. 2012.
20. SILVA, Nuno. **Dispositivos de Entrada e Saída de dados**. Disponível em: <<http://www.dei.isep.ipp.pt/~nsilva/ensino/ti/ti1998-1999/dispintrsaida/dispintrsaida.htm>>. Acesso em: 18 mai. 2012.

21. DOTTA, Silvia Cristina. **Projeto de Interface com o usuário**. Disponível em: <<http://www.slideshare.net/silviadotta/>>. Acesso em : 12 jun 2012.
22. OLIVEIRA, Igor Aguiar. Interface de Usuário: A Interação Homem-Computador Através dos Tempos. **Revista Olhar Científico**, vol 1, n. 2, ago-dez/2010. Disponível em: <<http://www.olharcientifico.kinghost.net/index.php/olhar/article/viewFile/28/29>>. Acesso em: 5 jun. 2012.
23. FONSECA, Manoel João; JORGE, Joaquim Armando. **Eddy: Um Editor Gráfico Multimodal com Reconhecimento de Fala e Gestos**. Disponível em: <<http://virtual.inesc.pt/8epcg/actas/c4/index.html#ind>>. Acesso em: 5 jun. 2012.
24. DUNCAN, David Ewing. A Little Device That's Trying to Read Your Thoughts. **The New York Times**. San Diego, 2 abr. 2012. Disponível em: <<http://www.nytimes.com/2012/04/03/science/ibrain-a-device-that-can-read-thoughts.html#>>. Acesso em: 23 mai. 2012.
25. LANGE, Catherine. **The man who saves Stephen Hawking's voice**. Disponível em: <<http://www.newscientist.com/article/dn21323-the-man-who-saves-stephen-hawkings-voice.html>>. Acesso em: 16 jun. 2012.
26. COLLET, Stacy. **Big Data: é um grande negócio?** Disponível em: <<http://computerworld.uol.com.br/tecnologia/2011/09/02/big-data-e-um-grande-negocio/#ir>>. Acesso em: jun. 2012.