

**UNIOESTE – Universidade Estadual do Oeste do Paraná**

**CENTRO DE CIÊNCIAS EXATAS E TECNOLÓGICAS**

**Colegiado de Ciência da Computação**

***Curso de Bacharelado em Ciência da Computação***

**Integração da Plataforma PlatMult em Carteiras e  
Lousas Digitais**

*Astério José Nuernberg Junior*

**CASCABEL**

**2013**

**ASTÉRIO JOSÉ NUERNBERG JUNIOR**

**INTEGRAÇÃO DA PLATAFORMA PLATMULTEM EM CARTEIRAS E  
LOUSAS DIGITAIS**

Monografia apresentada como requisito parcial  
para obtenção do grau de Bacharel em Ciência  
da Computação, do Centro de Ciências Exatas  
e Tecnológicas da Universidade Estadual do  
Oeste do Paraná - Campus de Cascavel

Orientador: Prof. Dr. Marcio Seiji Oyamada

CASCADEL

2013

**ASTÉRIO JOSÉ NUERNBERG JUNIOR**

**INTEGRAÇÃO DA PLATAFORMA PLATMULT EM CARTEIRAS E  
LOUSAS DIGITAIS**

Monografia apresentada como requisito parcial para obtenção do Título de *Bacharel em Ciência da Computação*, pela Universidade Estadual do Oeste do Paraná, Campus de Cascavel, aprovada pela Comissão formada pelos professores:

---

Prof. Dr. Marcio Seiji Oyamada (Orientador)  
Colegiado de Ciência da Computação,  
UNIOESTE

---

Prof. Dr. Clodis Boscaroli  
Colegiado de Ciência da Computação,  
UNIOESTE

---

Cassiano Cesar Casagrande  
Bolsista DTI do Projeto PlatMult,  
UNIOESTE

Cascavel, 17 de outubro de 2013.

## **DEDICATÓRIA**

*Dedico este trabalho a minha família, amigos, professores e a todos aqueles que participaram direta e indiretamente desta conquista.*

## **AGRADECIMENTOS**

Primeiramente, gostaria de agradecer aos meus pais Astério José Nuernberg e Jane Rosileia Esser Nuernberg pelo incentivo, força e por sempre acreditarem em mim. Sem o apoio de vocês não teria conseguido concluir esta etapa da minha vida. Muito Obrigado.

Agradecer meus tios Vanderlei Carlos Giordani e Juceli Esser Giordani com quem eu morei durante esses quatro anos, que me acolheram muito bem e me fizeram sentir em casa durante esse período.

Agradeço ao Prof. Dr. Marcio Seiji Oyamada, amigo, professor e orientador, pelo apoio e paciência durante elaboração deste trabalho. Agradeço pela confiança, pelo entendimento dos meus limites e a cada incentivo, pelas várias recomendações e conselhos, pela orientação deste trabalho e pelos três anos de Projeto.

Um grande agradecimento a todos os amigos que fiz durante esses quatro anos: Gustavo Catarino da Costa, Thiago Faino, Mateus Felipe Teixeira, Leandro Jorge Vieira da Maia, Leonardo Nardelli, Lucas Szeremeta, Gustavo Henrique Paetzold, Igor Steuck Lopes, Eduardo Vansetto, Wilson Coelho Neto, Paulo Wesley Santiago e tantos outros.

Enfim, um muito obrigado a todos que participaram direta ou indiretamente desta conquista.

## Lista de Figuras

2.1	Arquitetura Geral da Plataforma PlatMult .....	5
2.2	Mouse AFM .....	7
2.3	Esquema Global da AT-SPI .....	9
2.4	Diagrama de sequencia para fornecimento de <i>feedback</i> motor .....	12
2.5	Diagrama de sequencia para fornecimento de <i>feedback</i> auditivo .....	13
2.6	Diagrama de seqüência do funcionamento do FireDEMO .....	14
2.7	Diagrama de seqüência do funcionamento do Injetor de Eventos .....	15
3.1	Dispositivo <i>touch screen</i> inventado por E. A. Johnson .....	18
3.2	PLATO IV .....	19
3.3	HP-150 .....	19
3.4	Palm Pilot 5000 .....	20
3.5	<i>Touch screen</i> com tecnologia resistiva .....	21
3.6	<i>Touch screen</i> com tecnologia capacitiva superficial .....	23
3.7	<i>Touch screen</i> com tecnologia capacitiva projetada .....	24
3.8	<i>Touch screen</i> com tecnologia infravermelho .....	25
3.9	<i>Touch screen</i> com tecnologia superfície de onda acústica .....	26
3.10	Carteira Digital desenvolvida pelo CTI .....	27
3.11	Lousa interativa desenvolvida pelo CTI .....	28
3.12	Diagrama de blocos do sistema da carteira e da lousa .....	30
4.1	Lousa interativa com os recursos de acessibilidade da plataforma PlatMult .	35

## Lista de Tabelas

4.1	Porcentagem do uso de CPU e memória durante navegação na internet utilizando todos os recursos de acessibilidade da plataforma PlatMult . . . . .	36
4.2	Porcentagem do uso de CPU e memória durante navegação na internet utilizando todos os recursos de acessibilidade da plataforma PlatMult . . . . .	37
4.3	Atividades do Teste de Usabilidade . . . . .	39

# Lista de Abreviaturas e Siglas

ABINFO	Associação Brasileira de Informática
API	<i>Application Programming Interface</i>
ATK	<i>Accessibility Toolkit</i>
AT-SPI	<i>Assistive Technology Service Provider Interface</i>
CTI	Centro de Tecnologia da Informação Renato Archer
GAIL	<i>GNOME Accessibility Implementation Library</i>
GNOME	<i>GNU Network Object Model Environment</i>
GTK	<i>GIMP Toolkit</i>
IBGE	Instituto Brasileiro de Geografia e Estatística
ITO	Óxido de Índio e Estanho
Mouse AFM	Mouse de Acessibilidade por <i>Feedback</i> Motor
PROUCA	Programa Um Computador por Aluno
S <sub>n</sub> O <sub>2</sub>	Dióxido de Estanho
SO	Sistema Operacional
SPI	<i>Service Provider Interface</i>
TA	Tecnologia Assistiva
TIC	Tecnologia da Informação e Comunicação
IHC	Interação Humano Computador

# Sumário

<b>Lista de Figuras</b>	<b>vi</b>
<b>Lista de Tabelas</b>	<b>vii</b>
<b>Lista de Abreviaturas e Siglas</b>	<b>viii</b>
<b>Sumário</b>	<b>ix</b>
<b>Resumo</b>	<b>xii</b>
<b>1 Introdução</b>	<b>1</b>
1.1 Objetivos	3
1.2 Organização do Trabalho	3
<b>2 A Plataforma PlatMult</b>	<b>4</b>
2.1 Introdução	4
2.2 Recursos da Plataforma PlatMult	5
2.2.1 Recurso Visual	5
2.2.2 Recurso Auditivo	6
2.2.3 Recurso Tátil	6
2.3 Acessibilidade no GNOME	7
2.4 AT-SPI	9
2.5 Identificação dos Eventos usados para a Geração do <i>Feedback</i> Motor e Auditivo	10
2.6 Fornecimento de <i>Feedback</i>	12
2.6.1 <i>Feedback</i> Motor	12
2.6.2 <i>Feedback</i> Auditivo	13
2.7 Firefox na Plataforma PlatMult	14
<b>3 Tecnologia <i>Touch Screen</i>, Lousa e Carteira Digital</b>	<b>17</b>
3.1 Introdução e Histórico	17
3.2 Tipos de Tecnologia <i>Touch Screen</i>	20
3.2.1 Resistivo	21
3.2.2 Capacitivo	22

3.2.2.1	Tecnologia Capacitiva Superficial .....	23
3.2.2.2	Tecnologia Capacitiva Projetada .....	23
3.2.3	Infravermelho .....	24
3.2.4	Superfície de Onda Acústica .....	25
3.3	<i>Pen Computing</i> .....	26
3.4	Lousa e Carteira Digital .....	27
3.4.1	Arquitetura e Princípio de Operação da Lousa e Carteira Digital Indutiva .....	28
3.5	<i>Touch Screen</i> e Deficiência Visual .....	31
<b>4</b>	<b>Integração dos Recursos da Plataforma PlatMult na Lousa e Carteira Digitais</b> .....	<b>33</b>
4.1	Etapas na Integração .....	33
4.1.1	Estudo Técnico .....	33
4.1.2	Verificação de Compatibilidade .....	34
4.1.3	Teste de Integração .....	34
4.2	Problemas Encontrados .....	36
4.3	Possíveis Soluções .....	37
4.4	Teste de Usabilidade .....	38
4.5	Metodologia da Avaliação de IHC .....	39
4.5.1	Usuários Utilizados na Avaliação .....	41
4.5.2	Aplicação do Teste de Usabilidade .....	41
4.5.3	Resultados .....	42
4.5.4	Problemas Encontrados e Possíveis Soluções .....	43
<b>5</b>	<b>Considerações Finais</b> .....	<b>45</b>
5.1	Trabalhos Futuros .....	46
<b>A</b>	<b>Tutorial de Instalação da Plataforma PlatMult</b> .....	<b>47</b>
<b>B</b>	<b>Tutorial de Instalação da Carteira Digital</b> .....	<b>49</b>
	<b>Referências Bibliográficas</b> .....	<b>51</b>

# Resumo

As Tecnologias da Informação e Comunicação (TIC) têm sido importantes instrumentos e ferramentas de apoio para todas as áreas de atividade humana, especialmente para área de educação, graças à sua capacidade de processamento, armazenamento e visualização da informação. O desenvolvimento de interfaces e dispositivos de interação mais amigáveis, como o *touch screen*, que substitui o teclado e o *mouse*, permite que o usuário interaja diretamente com os elementos presentes na tela através dos dedos ou de uma caneta. Porém, essas tecnologias não são acessíveis a algumas esferas da população, como os usuários baixa visão, uma vez que encontrariam muita dificuldade na utilização desta tecnologia. Para amenizar esse problema, é necessário que recursos de acessibilidade sejam inseridos nessas tecnologias. O uso de Tecnologias Assistivas (TA) fornece essa acessibilidade, permitindo ou facilitando que usuários baixa visão utilizem ferramentas inacessíveis devido às suas limitações. É nesse ponto que se enquadra esse trabalho, que apresenta a integração da plataforma PlatMult com a carteira e a lousa digital desenvolvida no CTI – Centro de Tecnologia da Informação Renato Archer. Após a integração, um teste de usabilidade foi realizado, a fim de avaliar a usabilidade da integração e encontrar problemas decorrentes deste processo.

**Palavras-chave:** Tecnologia Assistiva, *Touch Screen*, Baixa Visão

# Capítulo 1

## Introdução

No mundo contemporâneo nenhuma sociedade está alheia à entrada das Tecnologias da Informação e Comunicação (TIC) em seu dia a dia. Nota-se que a quantidade gerada de informação por essas tecnologias traz transformações substanciais ao processo de aquisição do conhecimento pelos indivíduos.

Para se inserir na sociedade digital é necessário mais do que apenas ter acesso às TIC, é preciso, principalmente, saber utilizar essa tecnologia para a procura e a escolha de informações que possibilitem aos indivíduos resolverem seus problemas habituais, compreender o mundo e atuar na transformação de seu contexto.

Esta sociedade digital traz a necessidade da inclusão da população na Era da Informação Digital. Diante deste cenário, as escolas desempenham um papel muito importante, preparando as pessoas para essa sociedade digital. Assim, é necessário que as escolas preparem-se para a integração de tecnologias da informática ao processo educacional.

Diversos projetos e programas educacionais vêm sendo desenvolvidos com o intuito de promover tal integração. Uma das principais iniciativas é o Programa Um Computador por Aluno (PROUCA) [1], um Programa do Governo Federal que tem por objetivo ser um projeto educacional utilizando tecnologia para proporcionar inclusão digital, democratização do acesso à informação, troca de informações e experiências, compreensão crítica da realidade e desenvolvimento humano, social, cultural e educacional.

As carteiras e lousas digitais com tecnologia *touch screen* são projetos que partilham do mesmo objetivo e são o enfoque desta pesquisa. Estes projetos vêm sendo desenvolvidos pelo Centro de Tecnologia da Informação Renato Archer (CTI) em conjunto com a Associação

Brasileira de Informática (ABINFO), para a colaboração de um projeto de informatização escolar da Prefeitura de Serrana-SP.

De acordo com o CENSO/Instituto Brasileiro de Geografia e Estatística (IBGE) realizado em 2010 [2], cerca de 45,6 milhões de brasileiros possuem algum tipo de deficiência mental, motora ou visual. Desta parcela, estima-se que 60% dos indivíduos são deficientes visuais. Dentre os tipos de deficiência visual, encontra-se a baixa visão. Entende-se por visão subnormal (ou baixa visão) o comprometimento parcial, porém significativo, da visão. Não pode ser corrigida por meio de lentes de contato convencionais ou intervenção cirúrgica. A degeneração macular, a retinopatia diabética, a catarata e o glaucoma são exemplos de doenças que levam a perda parcial da visão, tornando-a subnormal [3].

Desta parcela de indivíduos com deficiência visual, uma parte é constituída de alunos que podem encontrar dificuldades no uso das novas tecnologias nas escolas. Para contornar essas dificuldades, é necessário que essas novas tecnologias possuam meios de fornecer acessibilidade, possibilitando assim que todos os estudantes possam aproveitá-las ao máximo.

Tais recursos de acessibilidade são providos por uma Tecnologia Assistiva (TA), um termo utilizado para identificar todo o arsenal de recursos e serviços que contribuem para proporcionar ou ampliar habilidades funcionais de pessoas com deficiência e consequentemente promover vida independente e inclusão [4]. De acordo com [5], a definição de TA pode ser entendida como *uma ampla gama de equipamentos, serviços, estratégias e práticas concebidas e aplicadas para minorar os problemas funcionais encontrados pelos indivíduos com deficiências*.

A TA utilizada neste estudo será proveniente de um projeto desenvolvido no Grupo de Inteligência Aplicada (GIA), no campus de Cascavel da Unioeste, a plataforma PlatMult, que é um ambiente multissensorial livre, que visa a acessibilidade em totens de informação para usuários de baixa visão. A plataforma é dita multissensorial, uma vez que seus recursos fornecem estímulos visuais, auditivos e táteis [6], que possibilitam aos usuários com baixa visão a utilização de computadores e terminais autoatendimento.

## 1.1. Objetivos

Este trabalho tem por principal objetivo a integração dos recursos de acessibilidade tátil, visual e auditivo da plataforma PlatMult na carteira e na lousa digitais desenvolvidas pelo CTI. O trabalho tem como objetivos específicos:

- Estudar a solução da carteira e lousa digital, cujas tecnologias empregadas são genuinamente nacionais;
- Possibilitar um novo meio de utilização dos recursos de acessibilidade da plataforma PlatMult.

A integração foi precedida de um estudo técnico dos dispositivos, e de uma verificação de compatibilidade entre Sistemas Operacionais e Interfaces de Programação de Aplicativos (API) a serem utilizados. Após a integração foi realizado um teste preliminar de usabilidade com dois usuários videntes, para obtenção de dados e direcionamentos de trabalhos futuros.

## 1.2. Organização do Trabalho

O trabalho está organizado nos seguintes capítulos:

- Capítulo 2. Apresenta a plataforma PlatMult, bem como suas características, funcionamento e seus recursos. Também introduz a biblioteca AT-SPI e sua utilização para prover acessibilidade. Descreve o funcionamento do *feedback* motor e auditivo, bem como os eventos utilizados para gera-los.
- Capítulo 3. Introduz a tecnologia *touch screen*, apresentando um breve histórico e as diferentes tecnologias utilizadas para o desenvolvimento de uma tela sensível ao toque. Também apresenta a lousa e a carteira digitais, descrevendo sua arquitetura e operação, além de abordar a situação atual das pessoas baixa visão em relação ao *touch screen*.
- Capítulo 4. Apresenta uma análise dos resultados obtidos com a integração proposta, bem como os resultados obtidos com o teste de usabilidade.
- Capítulo 5. Descreve as considerações finais do trabalho, conclusões e sugestões de trabalhos futuros.

## Capítulo 2

### A Plataforma PlatMult

Este capítulo tem por objetivo apresentar a plataforma PlatMult. As Seções 2.1 e 2.2 apresentam uma introdução sobre a plataforma, apresentando suas características, funcionamento e os recursos de acessibilidade. Na Seção 2.3 é explicado como funcionam as aplicações com acessibilidade no GNOME. Na Seção 2.4 é descrito como ocorre o relacionamento entre a biblioteca AT-SPI e as aplicações do sistema e tecnologias assistivas. Na Seção 2.5 são apresentados e descritos os eventos que ativam os recursos de *feedback* presentes na plataforma PlatMult. Nas seções 2.6 e 2.7 são explicados como funciona a geração e o tratamento dos eventos nos recursos auditivo, tátil e no *plugin* para o Firefox.

#### 2.1. Introdução

A plataforma PlatMult é uma plataforma multissensorial composta por componentes de hardware e software, que oferece recursos de acessibilidade e usabilidade, que tem por objetivo propiciar um ambiente de interação voltado, mas não somente, para usuários com baixa visão em terminais de autoatendimento. A utilização da plataforma PlatMult é voltada para locais onde há necessidade de disponibilização de informações digitais, como bancos, bibliotecas, aeroportos, empresas de serviços públicos e terminais de ônibus [6].

Um dos principais objetivos do projeto é fornecer uma solução de baixo custo, que possibilite o uso por um grande número de usuários. Para atingir esse objetivo, o hardware utilizado é constituído de computadores cedidos pela Unioeste, bem como os protótipos desenvolvidos feitos a partir de carcaças de máquinas caça-níquel doadas pelo Tribunal de Justiça de Toledo-PR. No que tange ao software, o objetivo é cumprido pela utilização de software livre. Assim, a solução executa no Sistema Operacional Linux, distribuição Ubuntu

[7], com suporte do servidor gráfico o Xorg [8] e ao gerenciador de janelas GNU *Network Object Model Environment* (GNOME) [9]. O servidor de acessibilidade utilizado é a *Assistive Technology Service Provider Interface* (AT-SPI) [10], uma camada desenvolvida pelo GNOME para prover uma *Application Programming Interface* (API) padronizada para construção de aplicações com recursos de acessibilidade.

## 2.2. Recursos da Plataforma PlatMult

A plataforma PlatMult é denominado como uma plataforma multissensorial, pois os recursos de que dispõe exploram, ao mesmo tempo, tanto os sentidos visuais, como também os auditivos e táteis do usuário que a utiliza [6]. O recurso tátil é fornecido pelo mouse *feedback*, enquanto que os recursos auditivo e visual são fornecidos pelo xLupa [11]. Tais recursos atuam em conjunto de modo a fornecer aos usuários uma melhor experiência no uso de totens de informação e computadores. A Figura 2.1 apresenta a arquitetura geral do PlatMult.

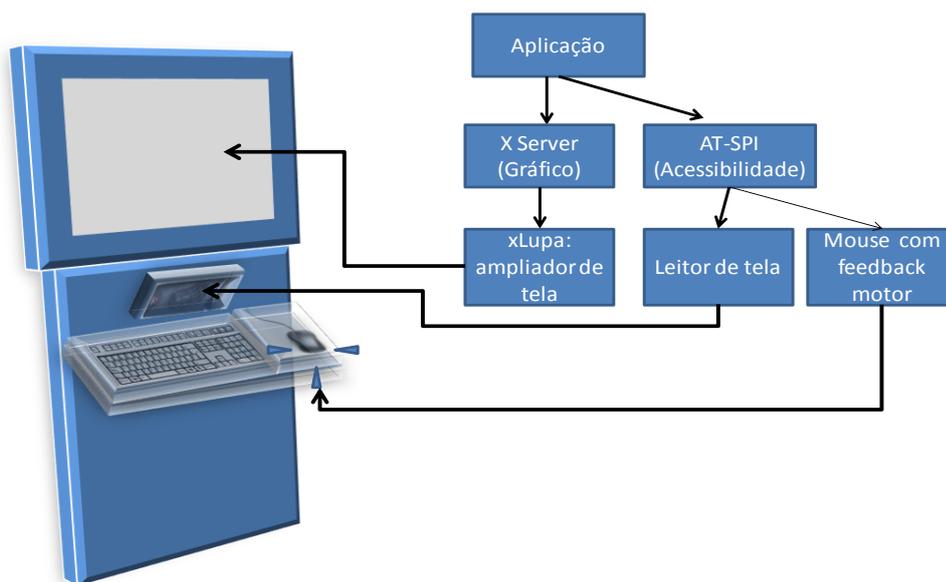


Figura 2.1: Arquitetura geral da plataforma PlatMult. Fonte: [6]

### 2.2.1. Recurso Visual

O recurso visual presente na plataforma PlatMult é fornecido pelo ampliador de tela xLupa<sup>1</sup>, um software livre que executa sob a plataforma Linux. O xLupa fornece ampliação

<sup>1</sup> <http://projetos.unioeste.br/campi/xlupa/>

de textos e imagens, além de mecanismos que possibilitam ao usuário calibrar a exibição conforme suas necessidades próprias. Esses mecanismos incluem a seleção de cor de fundo e de primeiro plano, mouse em cruz, a aplicação de algoritmos para o processamento de imagens e suavização de linhas de contorno, bem como configurações de brilho e contraste. Um dos diferenciais da ferramenta é a possibilidade da criação de perfis, identificados por um login e senha para cada usuário. Com este recurso, o usuário tem a possibilidade de salvar suas configurações de uso do software, podendo carregá-las na próxima utilização do sistema. Este recurso facilita a experiência do usuário, evitando que o mesmo execute as tarefas relacionadas à configuração da ferramenta toda vez que a utiliza [6].

### **2.2.2. Recurso Auditivo**

O recurso auditivo da plataforma PlatMult é fornecido por um leitor de tela presente no xLupa. O leitor cria uma interface de áudio com o usuário, lendo informações a respeito de componentes gráficos do sistema, como itens de menus, botões, janelas, links, bem como a inserção e remoção de caracteres em um texto, possibilitando ao usuário localizar-se durante a navegação. O leitor foi desenvolvido na linguagem de programação Python devido à facilidade de integração com outras linguagens de programação e pela possibilidade de utilização da API de acessibilidade AT-SPI, responsável por reportar os eventos que o leitor informa ao usuário. Esta API possui uma biblioteca em Python, chamada pyatspi, cuja principal função, advinda da especificação da AT-SPI, é retirar o esforço do programador em ter que criar módulos específicos para cada aplicação que esteja executando no sistema, assim o programador pode criar um módulo genérico para tratar todos os aplicativos, e evitar uma comunicação específica com o sistema ou com os aplicativos que executam sobre ele [12].

### **2.2.3. Recurso Tátil**

O recurso tátil presente na plataforma PlatMult é proporcionado por um mouse com feedback motor, que chama a atenção do usuário para informações importantes presentes na tela. Ao vibrar, o mouse indica que o usuário está diante de um tópico do menu, link, botão e demais informações presentes na tela do sistema. A solução utilizada no recurso tátil do PlatMult é uma combinação de hardware e software, diferindo dos recursos visual e auditivo, que são soluções de software [6].

O hardware da solução é o Mouse AFM (Mouse de Acessibilidade por *Feedback* Motor), um mouse adaptado que fornece vibração durante a navegação do usuário. A vibração é possibilitada por um *vibracall* de celular implantado no mouse. Na ponta deste dispositivo, há um peso ocupando metade da seção, que ao receber uma tensão contínua de 5V pela porta paralela ou pela porta USB, gera a vibração que fornece o *feedback* motor ao usuário. A Figura 2.2 demonstra um protótipo do Mouse AFM, e indica o motor vibratório nele inserido.

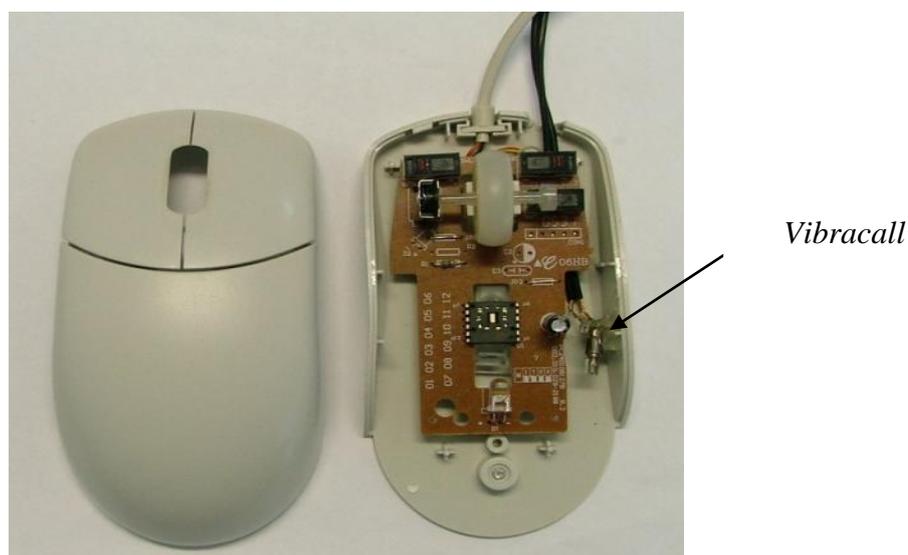


Figura 2.2: Mouse AFM

Em relação ao software usado na solução, foi desenvolvido um servidor tátil que utiliza os eventos gerados pela AT-SPI para gerar o *feedback* motor. Os princípios utilizados no desenvolvimento do servidor tátil são os mesmos do leitor de tela, com a diferença de que este foi desenvolvido na linguagem de programação C, devido à facilidade de comunicação com as portas paralela e USB para gerar a vibração do mouse.

### 2.3. Acessibilidade no GNOME

O GNOME é parte de um projeto internacional de software livre desenvolvido pela GNU que provê basicamente duas coisas: o ambiente *desktop* GNOME, intuitivo e atraente para usuários finais, e a plataforma de desenvolvimento GNOME, um *framework* para construção de aplicações que se integrem com todo o *desktop* [9].

Há outro projeto sendo desenvolvido, chamado de GNOME *Accessibility*, onde são desenvolvidas e promovidas soluções de código aberto e livre para interfaces gráficas com o

usuário. O GNOME já possui algumas tecnologias assistivas, como leitores de tela e ampliadores, para apoiar os usuários com deficiências visuais e auditivas<sup>2</sup>, e para interagir com dispositivos auxiliares comuns. A geração de acessibilidade do GNOME é base para o PlatMult, sobre o qual os recursos de acessibilidade do PlatMult foram desenvolvidos.

Para implementar aplicações com recursos de acessibilidade, o desenvolvedor deve utilizar a ATK (*Accessibility Toolkit*), uma ferramenta que o GNOME usa para habilitar acessibilidade para os usuários que necessitam de apoio extra para aproveitar ao máximo os recursos do computador. O ATK permite uma interação com o *desktop* por meio de interfaces que necessitam ser implementadas pelos componentes GUI, para que assim possam fornecer acessibilidade. Essas interfaces não dependem de nenhuma IDE específica para serem implementadas, possibilitando assim ao desenvolvedor escolher a de sua preferência.

O desenvolvimento de *widgets* (menus, janelas, botões) GTK é feito em um módulo chamado GAIL (*GNOME Accessibility Implementation Library*), uma implementação das interfaces de acessibilidade definidas pela ATK. O GTK+ é uma biblioteca para a criação de componentes gráficos (janelas, botões, menus), que utiliza a GAIL para fornecer recursos de acessibilidade para as interfaces criadas pela mesma. A GAIL é dinamicamente carregada em tempo de execução por uma aplicação GTK, e uma vez carregada, as partes das aplicações que usam *widgets* GTK padrão vão ter um nível de acessibilidade básico, sem a necessidade de modificar a aplicação. Se a GAIL não for carregada, os *widgets* GTK terão um nível de acessibilidade padrão que não retorna informações.

A maioria das tecnologias assistivas que eram executadas em outros ambientes *desktop*, necessitavam de um complexo modelo *off-screen* dos aplicativos de *desktop*, baseado na observação dos eventos do Sistema Operacional (SO), bem como do uso do SO, recursos de aplicações e API sem suporte, o que tornava o suporte à tecnologia assistiva fraco e altamente específico em relação à aplicação e ao SO. No GNOME, ao contrário, todas as informações exigidas por uma tecnologia assistiva são fornecidas pelos aplicativos em execução, via *GNOME Accessibility Framework*, para uma SPI (*Service Provider Interface*) independente de ferramentas.

---

<sup>2</sup> Orca: <https://wiki.gnome.org/Projects/Orca>

A SPI fornece um meio para que uma TA baseada em Linux possa obter informações de acessibilidade de aplicativos em execução via uma API consistente e estável, eliminando assim a necessidade de um modelo *off-screen* em muitos casos. O suporte de acessibilidade para aplicações é embutido no conjunto de ferramentas para aplicação por um kit de ferramentas apropriado das APIs (ATK, por exemplo), e exportado para a interface AT-SPI, detalhada a seguir.

## 2.4. AT-SPI

A AT-SPI (*Assistive Technology Service Provider Interface*) é a principal interface de serviços pela qual as tecnologias assistivas consultam e recebem notificações de dispositivos e aplicativos em execução. Desenvolvida pelo projeto GNOME, é um conjunto de ferramentas que fornecem uma representação lógica do conteúdo das aplicações, facilitando assim o desenvolvimento de aplicações com recursos de acessibilidade. A interface será explicada na sequência, pois é utilizada pelo leitor de tela e pelo mouse AFM para captar os eventos que estão ocorrendo no sistema e informá-los ao usuário.

A Figura 2.3 demonstra o esquema global da AT-SPI, que atua como uma conexão entre os aplicativos fornecedores de acessibilidade e os programas do sistema.

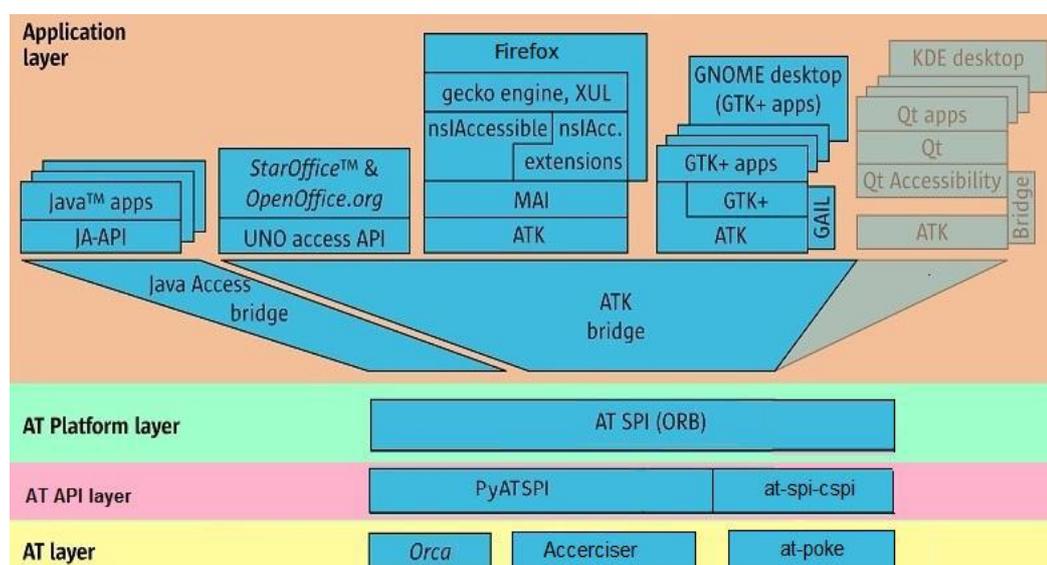


Figura 2.3: Esquema global da AT-SPI. Fonte: [13]

O esquema mostrado na Figura 2.3 é composto por quatro camadas. Na primeira camada, "Application layer", estão os programas utilizados pelos usuários e suas respectivas

bibliotecas. Esses aplicativos fornecem informações de seus componentes para a AT-SPI por meio de bibliotecas, neste caso, *ATK bridge* e *Java Access bridge*, que atuam como pontes entre as aplicações e a interface AT-SPI. A AT-SPI está situada na segunda camada, chamada de “*AT Platform layer*”, e tem como incumbência passar os eventos ocorridos na camada superior para as aplicações de acessibilidade. Sua implementação é feita em CORBA, o que possibilita o seu uso em ambientes heterogêneos em conjunto com outras linguagens de programação, como Python, C e Java.

Na terceira camada, “*AT API layer*” estão as bibliotecas utilizadas pelos programas de acessibilidade da quarta camada, chamada de “*AT layer*”. Essas bibliotecas são utilizadas para que os programas de acessibilidade se registrem na AT-SPI e possam assim receber as informações desejadas da primeira camada. Essas bibliotecas são benéficas para o desenvolvedor, pois lhe retira o esforço de ter que criar módulos específicos para cada software que esteja executando no sistema, podendo criar um módulo genérico para tratar todos os aplicativos.

## **2.5. Identificação dos Eventos Usados para a Geração do Feedback Motor e Auditivo**

Os eventos são gerados a partir das ações do usuário durante o uso do computador, como navegação por menus, itens de menus e janelas através do mouse, edição de textos, etc. Uma das formas de verificar se uma aplicação gera eventos de acessibilidade e quais são eles é utilizando o Accerciser [14]. O Accerciser é um explorador de acessibilidade interativo, desenvolvido em Python para a área de trabalho GNOME. Ele usa AT-SPI para inspecionar e controlar *widgets*, permitindo verificar se um aplicativo está fornecendo a informação correta para tecnologias assistivas e *frameworks* de testes automatizados. O Accerciser tem uma estrutura simples na qual você pode usar para criar visões personalizadas de informações de acessibilidade [14].

No recurso de acessibilidade tátil da plataforma PlatMult, os eventos do tipo *Focus* são usados para fornecer *feedback* motor ao usuário. Já para o *feedback* auditivo, são usados os eventos do tipo *Focus*, *Object:text-changed:insert*, *Object:text-changed:delete*, *Object:text-*

*caret-moved*, *Window:activate* e *Window:create*. A seguir, há uma sucinta descrição dos eventos utilizados para o fornecimento de *feedback*.

**Focus:** Este evento monitora todos os elementos gráficos da tela que podem receber foco. Este é o evento mais gerado durante a utilização do computador, pois ele acontece durante a navegação por menus, botões, janelas, etc, que o usuário faz com o *mouse*. O evento de *Focus* é utilizado no *feedback* motor devido ao ponteiro do *mouse* trazer foco aos elementos da tela que passa. O uso para ativar o *feedback* auditivo se deve ao fato de poder se obter o texto contido no elemento que recebe o foco, ajudando o usuário a se localizar na tela.

**Object:text-changed:insert:** Este evento será gerado quando ocorrer uma digitação ou inserção de textos em aplicações. Para tanto, são monitorados o teclado, a espera da inserção de texto por parte do usuário durante a edição de arquivos, bem como as inserções realizadas por aplicativos. Os eventos são utilizados no *feedback* auditivo, que lê para o usuário a edição de texto feito por ele ou por aplicativos.

**Object:text-changed:delete:** Equivalente ao evento anterior, também está ligado a edição de textos no sistema, porém só é gerado durante a exclusão de textos. O processo de aquisição do evento e seu uso por parte do leitor de tela é idêntico ao anterior.

**Object:text-caret-moved:** Similar aos eventos *Object:text-changed*, também acontece durante a edição de textos, porém é gerado quando o usuário movimenta o cursor de digitação sobre o texto (usando as setas, por exemplo). Os eventos são utilizados para ativar o *feedback* auditivo, informando assim ao usuário a posição do cursor no texto, e auxiliando-o a localizar-se no texto.

**Window:activate e Window:create:** Estes dois eventos acontecem durante a manipulação de janelas, sendo semelhantes entre si, e diferindo somente no momento em que são gerados. O *Window:activate* é gerado quando uma janela é ativada, ou seja, quando ela é posta em destaque na tela estando minimizada ou escondida anteriormente. Já o *Window:create* é gerado somente no momento em que uma janela é criada na tela. Estes eventos são utilizados pelo *feedback* auditivo para informar o nome da janela ao usuário.

## 2.6. Fornecimento de Feedback

Nas seções anteriores foi explicado como as tecnologias assistivas podem utilizar a AT-SPI para receber os eventos gerados pelo usuário e fornecer acessibilidade, assim como foram apresentados os eventos utilizados no *feedback* motor e auditivo. Nos sub tópicos desta seção tem-se a apresentação individual do esquema para fornecimento do *feedback* motor via Mouse Feedback, e do *feedback* auditivo via leitor de tela.

### 2.6.1. Feedback Motor

A ativação do *vibracall* presente no Mouse AFM é feita por um *driver* chamado Mouse Feedback. A vibração gerada pelo *vibracall* é que fornece o *feedback* motor ao usuário. O diagrama de sequência na Figura 2.4 apresenta as etapas que ocorrem para ativar o *feedback* motor da plataforma PlatMult. Na sequência, são descritos os passos presente nele.

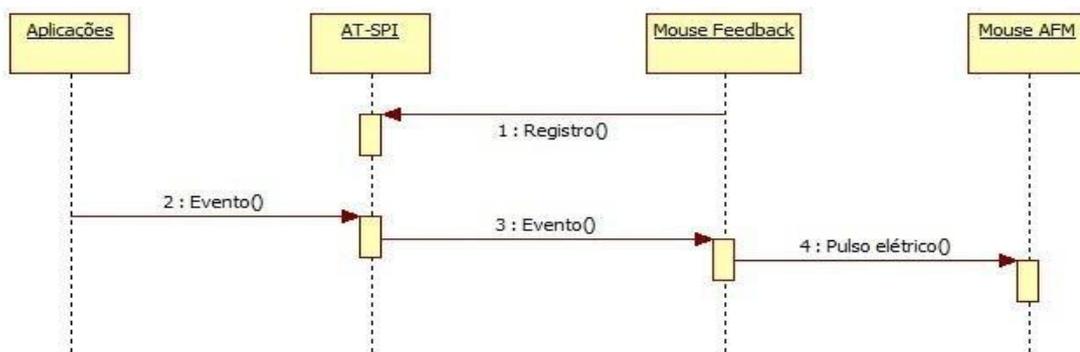


Figura 2.4: Diagrama de sequência para fornecimento de *feedback* motor.

- 1. Registro:** Quando entra em execução, o Mouse Feedback faz o registro na AT-SPI para ser notificado quando ocorrer eventos do tipo *Focus*.
- 2. Evento:** Quando um evento é gerado em uma aplicação, esse evento é reportado para a AT-SPI.
- 3. Evento:** Quando a AT-SPI recebe a informação de um evento ocorrido em uma aplicação, o Mouse Feedback é notificado do evento.
- 4. Pulso Elétrico:** Quando o Mouse Feedback é notificado sobre a ocorrência de um evento, este envia um pulso elétrico via porta paralela, para que o *vibracall* do Mouse AFM seja ativado.

Como o *driver* foi desenvolvido em C, é utilizada a biblioteca *at-spi-cspi* para realizar o registro do mesmo na AT-SPI. O *driver* se registra para receber notificação somente de eventos do tipo *Focus*, sendo que só são considerados os eventos que ocorrem após a inicialização do Mouse Feedback. Assim, sempre que ocorrer um evento após o início do *driver* o *feedback* motor presente no Mouse Feedback será ativado.

## 2.6.2. Feedback Auditivo

O leitor de tela integrado ao xLupa é o software responsável por fornecer o *feedback* auditivo para o usuário. O diagrama de sequência na Figura 2.5 apresenta as etapas da ativação do *feedback* auditivo. Na sequência são descritos os passos do diagrama de sequência.

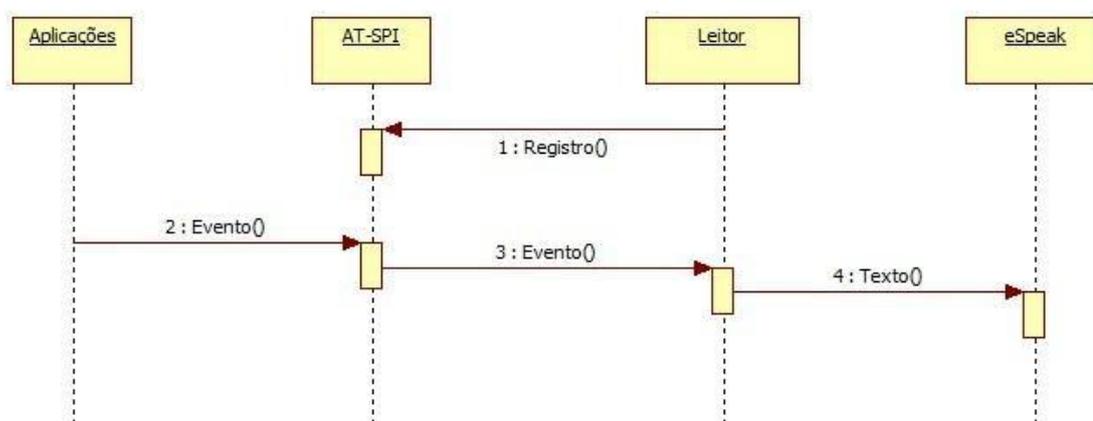


Figura 2.5: Diagrama de sequência para fornecimento de *feedback* auditivo.

1. **Registro:** Ao entrar em execução, o leitor registra-se na AT-SPI para ser notificado da ocorrência de eventos dos tipos *Focus*, *Object:text-changed:insert*, *Object:text-changed:delete*, *Object:text-caret-moved*, *Window:activate* e *Window:create*.
2. **Evento:** Quando um evento é gerado por uma aplicação, esse evento é reportado para a AT-SPI.
3. **Evento:** Quando a AT-SPI recebe a informação da ocorrência de um evento em uma aplicação, o leitor é notificado.
4. **Texto:** Quando o leitor é notificado sobre a ocorrência de um evento, este envia um texto que faz referência ao evento para o sintetizador de voz eSpeak [15], para que o mesmo leia o texto para o usuário.

Como o leitor foi desenvolvido em Python, é utilizada a biblioteca `pyatspi` para realizar o seu registro na AT-SPI. O leitor se registra para receber notificação de eventos do tipo *Focus*, *Object:text-changed:insert*, *Object:text-changed:delete*, *Object:text-caret-moved*, *Window:activate* e *Window:create*, mas que só são considerados após a inicialização do leitor. Assim, sempre que ocorrer um evento após o início do leitor o *feedback* auditivo presente no PlatMult será ativado.

## 2.7. Firefox na Plataforma PlatMult

Inicialmente, o *feedback* motor e auditivo durante a utilização da Internet só era fornecido quando o usuário percorria os menus do navegador Firefox. Não eram notificados à AT-SPI os eventos do tipo *Focus* quando o usuário passava com o ponteiro do *mouse* sobre conteúdos da página *web*, como *links*, botões, campos de entrada de dados e áreas de texto.

Essa falta de *feedback* é um problema, pois como as páginas *web* possuem muita informação, o usuário pode acabar encontrando dificuldades em localizar o conteúdo que deseja.

Como solução, foi desenvolvido o FireDEMO (Firefox Detector de Eventos de *Mouse Over*) um complemento desenvolvido em JavaScript para o navegador Firefox que capta os eventos de *mouse over*. Quando uma página é carregada, o FireDEMO faz uma busca dos campos de interesse e coloca um monitor (*Listener*) de eventos *mouse over* em cada um deles. O diagrama de sequência na Figura 2.6 apresenta o funcionamento do FireDEMO.

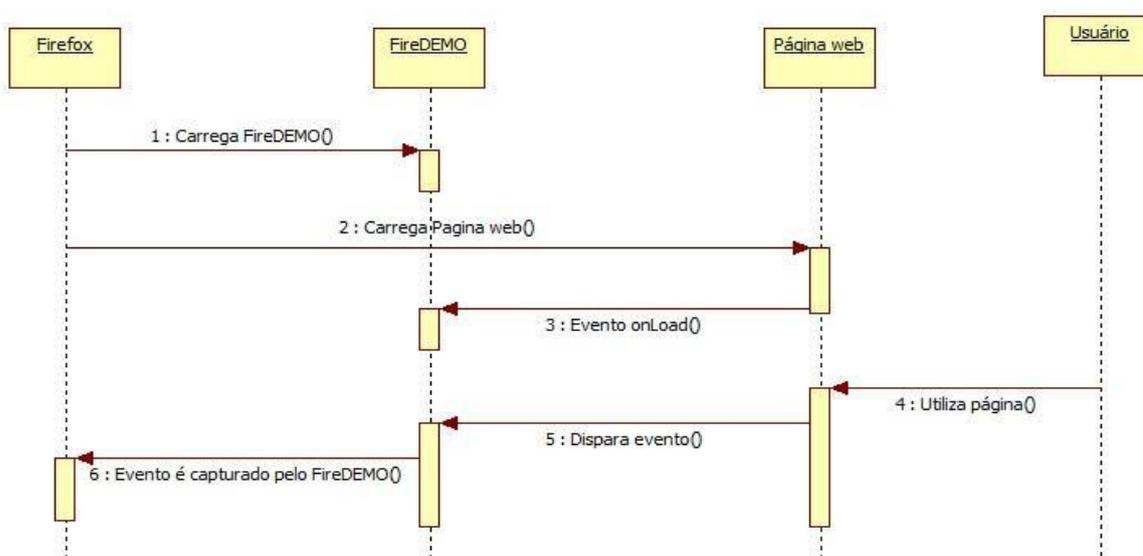


Figura 2.6: Diagrama de sequência do funcionamento do FireDEMO

1. **Carrega FireDEMO:** O complemento é carregado quando o Firefox é iniciado.
2. **Carrega Página web:** O navegador carrega uma nova página web.
3. **Evento onLoad:** O evento onLoad é ativado pelo FireDEMO após a página web ser carregada, para que sejam adicionados monitores de eventos (*Event Listeners*) nos campos de interesse da página.
4. **Utiliza página:** Navegação do usuário pela página web.
5. **Dispara Evento:** Quando o ponteiro do mouse passa sobre um campo de interesse monitorado, um evento é disparado.
6. **Evento é capturado pelo FireDEMO:** o complemento obtém as informações sobre o evento no campo de interesse.

O FireDEMO é responsável somente pela detecção e notificação dos eventos. A injeção desses eventos na AT-SPI é feita pelo módulo Injetor de Eventos, um programa desenvolvido em C que recebe os eventos detectados pelo FireDEMO e os injeta na AT-SPI para serem usados pelo mouse com *feedback* motor e pelo leitor de tela. A notificação desses eventos é feita via arquivo. Ao detectar um evento, o complemento escreve as informações sobre esse evento no arquivo, enquanto que o Injetor de Eventos faz acessos constantes em busca de modificações do conteúdo em relação ao seu ultimo acesso. Ao notar uma modificação feita pelo FireDEMO no arquivo, o Injetor de Eventos chama uma função para injetar o evento na AT-SPI. O diagrama de seqüência na Figura 2.7 apresenta o funcionamento do Injetor de Eventos.

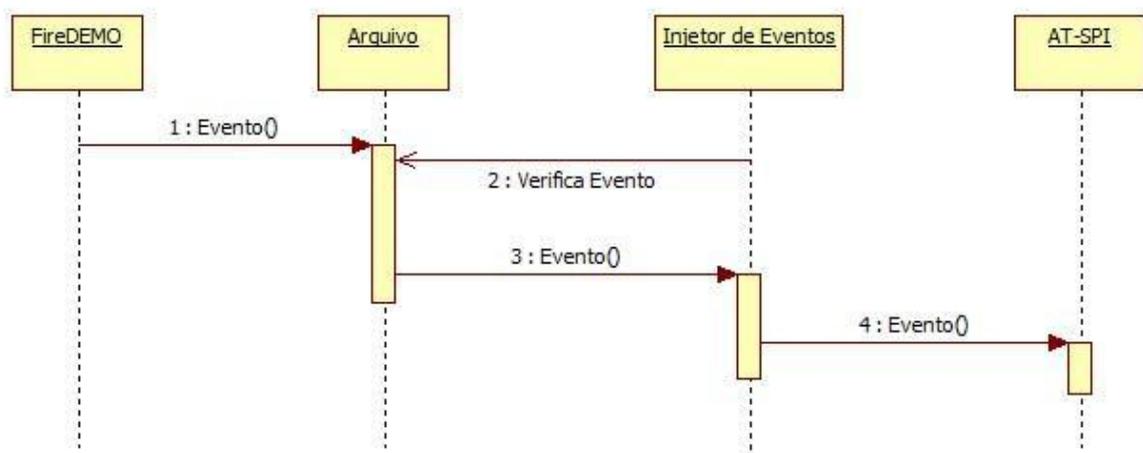


Figura 2.7: Diagrama de seqüência do funcionamento do Injetor de Eventos.

1. **Evento:** O FireDEMO capta um evento e escreve suas informações no arquivo.

2. **Verifica evento:** O Injetor de Eventos acessa o arquivo em busca de conteúdo novo.
3. **Evento:** O conteúdo do novo evento é recuperado pelo Injetor de Eventos.
4. **Evento:** O Injetor de Eventos injeta o evento na AT-SPI como um evento do tipo *Focus*.

## Capítulo 3

# Tecnologia *Touch Screen*, Lousa e Carteira Digital

Este capítulo tem por objetivo apresentar e descrever a tecnologia *touch screen*, a lousa e a carteira digital. Na Seção 3.1 há uma introdução sobre o *touch screen* e um breve histórico. Na Seção 3.2 são descritas as diferentes tecnologias utilizadas no desenvolvimento de uma tela sensível ao toque. A Seção 3.3 introduz a *Pen Computing*, que é a utilização de telas de toque por meio de uma caneta ou ponteira. Na Seção 3.4 são apresentadas a lousa e a carteira, bem como seu funcionamento, arquitetura e tecnologias empregadas. A seção 3.5 descreve o estágio atual das tecnologias *touch screen* em relação a deficiência visual.

### 3.1. Introdução e Histórico

O *touch screen* é um *display* eletrônico visual capaz de detectar a presença e localização de um toque dentro da área de exibição. Em outras palavras, uma tela sensível ao toque é qualquer monitor, com base tanto em LCD (*Liquid Crystal Display*) ou CRT (*Cathode Ray Tube*), que aceita a entrada direta na tela. A capacidade para a entrada direta na tela é facilitada por um dispositivo externo (uma caneta, por exemplo) ou um dispositivo interno (revestimento para toque e controlador), que retransmite as coordenadas *X* e *Y* para o dispositivo.

Com a tecnologia *touch screen*, o usuário pode controlar o que é exibido na tela, utilizando um ou mais de dedos, ou através de uma caneta. Isso é possível pois as telas de toque permitem que o usuário interaja diretamente com o que é exibido, em vez de controlar indiretamente com um cursor comandado por um mouse ou *touchpad*.

De acordo com historiadores, o primeiro dispositivo *touch screen* (Figura 3.1) foi inventado por E. A. Johnson, em 1965. A invenção foi publicada em [16], onde um mecanismo viável para o desenvolvimento de uma tela sensível ao toque capacitiva foi descrito. A ideia inicial de Johnson foi usada para o controle de tráfego aéreo no Reino Unido até os meados de 1990.



Figura 3.1: Dispositivo *touch screen* inventado por E. A. Johnson. Fonte: [17]

A tecnologia *touch screen* chamou a atenção do público primeiramente em 1971, por meio da invenção do “Elograph” [18]. O “Elograph” era um sensor resistivo e não possuía a tela transparente como os dispositivos atuais, no entanto, foi um marco significativo para a tecnologia de telas sensíveis ao toque. No início da década de 70, já haviam sido introduzidas inúmeras máquinas diferentes com capacidade de toque, embora nenhuma fosse sensível à pressão. Um dos primeiros que foi apresentado ao público em geral foi o terminal para o PLATO IV [19], um sistema assistido de ensino por computador. O PLATO IV (Figura 3.2) foi o precursor no uso da tecnologia infravermelho, e foi o primeiro computador com tela sensível ao toque a ser usado em uma sala de aula para permitir que os alunos tocassem na tela para responder às perguntas.



Figura 3.2: PLATO IV – Computador com tela sensível ao toque utilizando a tecnologia infravermelho.

Em 1982, o primeiro dispositivo *multi-touch* controlado por humano foi desenvolvido [20]. Constituíam-se de um painel de vidro fosco cujas propriedades ópticas locais eram de tal forma que, quando vistos por trás com uma câmera, uma mancha negra, cujo tamanho dependia da pressão do dedo, aparecia em um fundo branco. Em 1983 foi inventado o HP-150 [21], o primeiro computador pessoal com tela sensível ao toque. O HP-150 (Figura 3.3) possuía uma grade de raios infravermelhos na parte da frente do monitor que detectava o movimentos dos dedos do usuário. Um ano depois, a tecnologia *touch screen* dava um passo à frente, com o desenvolvimento do primeiro *display multi-touch* [22], o primeiro sensor capaz de capturar simultaneamente vários toques na tela. Essa descoberta ajudou a criar a tecnologia *multi-touch*, presente hoje em *tablets* e *smartphones*.



Figura 3.3: HP-150

Nos anos 1990 foram introduzidos os primeiros celulares e PDAs com telas de toque. Em 1993, IBM e Bell South se uniram para lançar o Simon Personal Communicator [23], um dos primeiros telefones celulares com tecnologia *touch screen*. Ele tinha uma tela sensível ao toque resistiva que requeria o uso de uma caneta para navegar pelos menus e dados de entrada. Em 1996, a Palm entrou no mercado de PDA com a série Pilot [24], obtendo grande sucesso na época. Também possuía tecnologia resistiva e necessitava do uso de uma caneta para navegação. A Figura 3.4 ilustra o modelo Palm Pilot 5000.

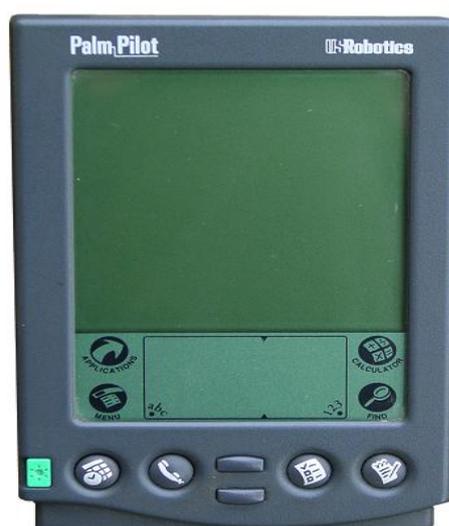


Figura 3.4: Palm Pilot 5000

Com tantas tecnologias diferentes surgindo nas décadas anteriores, os anos 2000 foram a época das tecnologias *touch screen* evoluírem e se tornarem populares. Por meio de pesquisas mais complexas, novas técnicas e barateamento no desenvolvimento, inúmeros avanços ocorreram, ajudando a popularizar a tecnologia *touch screen*. Assim, as telas sensíveis ao toque se tornaram cada vez mais presentes no dia-a-dia, em dispositivos como caixas eletrônicas, totens de informação, videogames, pontos de venda e, principalmente, *tablets* e *smartphones*.

## 3.2. Tipos de Tecnologia *Touch Screen*

Há vários tipos de tecnologias empregadas no desenvolvimento de telas de toque. Algumas utilizam luz visível ou infravermelha, algumas ondas de som, enquanto outras usam sensores de força. Todas possuem combinações individuais de características, vantagens e desvantagens, como tamanho, precisão, confiabilidade, durabilidade, número de toques

possíveis, além do custo. Na sequência são apresentadas os tipos de tecnologia: resistivo, capacitivo, infravermelho e superfície de onda acústica.

### 3.2.1. Resistivo

A tecnologia resistiva é o tipo mais comum de tecnologia *touch screen* atualmente, usada em muitos dispositivos, tais como telefones celulares, jogos portáteis, dispositivos de navegação GPS, dispositivos de monitoramento médico, pontos de venda e câmeras digitais.

Na tecnologia resistiva, as telas são feitas utilizando duas camadas separadas por pontos isolantes. Tipicamente, a camada inferior é feita de vidro e a camada superior é um filme plástico. Quando o usuário aperta o filme, este faz contato com o vidro e completa um circuito.

O vidro e plásticos são, cada um, coberto com uma grade de condutores elétricos. Estes condutores podem ser fios de metal fino, no entanto, feitos frequentemente de uma película fina de material condutor transparente. Na maioria dos casos, este material é o óxido de índio e estanho (ITO). Os elétrodos sobre as duas camadas funcionam em ângulos retos entre si: condutores paralelos correm em uma direção sobre a chapa de vidro e perpendicularmente à da película plástica. A Figura 3.5 ilustra uma tela *touch screen* resistiva, bem como as camadas que a compõe.

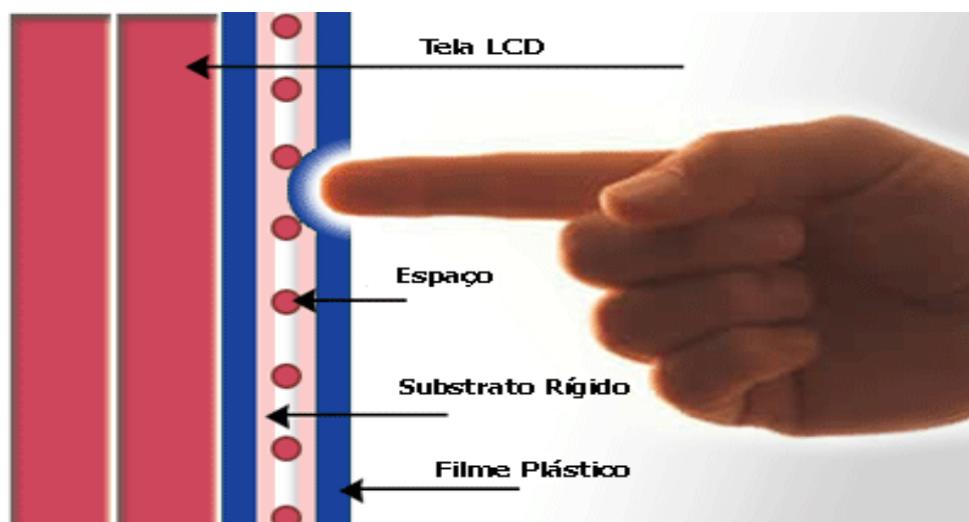


Figura 3.5: *Touch screen* com tecnologia resistiva. Fonte: [25]

Quando o usuário pressiona a tela de toque, o contato é feito entre a placa de vidro e a de filme plástico. A tensão do circuito é medida, e as coordenadas  $X$  e  $Y$  da posição de toque são

calculadas com base na quantidade de resistência no ponto de contato. Esta tensão analógica é processada por conversores analógico-digital, criando um sinal digital para que o controlador do dispositivo possa usar como um sinal da entrada feita pelo usuário.

Uma das grandes vantagens dos painéis de toque resistivo é que eles são relativamente baratos e simples de fazer. Outra, é que se pode usar quase qualquer coisa para criar um sinal de entrada, como a ponta do dedo, unha ou uma caneta. São particularmente utilizados em restaurantes e hospitais graças à sua alta resistência aos líquidos e sensibilidade à pressão, o que torna possível o toque com luvas ou outros objetos.

Esta tecnologia tem um grande número de inconvenientes, no entanto. Primeiro, o sistema analógico é suscetível de variações, de modo que o usuário pode ter que recalibrar o painel de toque ao longo do tempo. Além disso, o material de ITO utilizado para os condutores é frágil e não é adequado para dobrar. Com o tempo, o uso repetido pode levar o ITO a rachar, interrompendo o fluxo de eletricidade e, podendo resultar em um ponto morto na tela de toque. Outro problema desta tecnologia está relacionado ao espaço entre as duas camadas. Em termos funcionais, o maior problema com painéis resistivos é que podem receber apenas um toque de cada vez. Se o usuário tocar o painel em dois lugares ao mesmo tempo, o efeito combinado produzirá uma coordenada para o ponto de contato diferente de qualquer um dos dois pontos de reais.

### **3.2.2. Capacitivo**

Na tecnologia capacitiva, que é dividida em superficial e projetada, uma camada que armazena carga elétrica é colocada sobre o painel de vidro da tela. Quando um usuário toca na tela com o seu dedo, parte da carga é transferida para o usuário, diminuindo a carga na camada capacitiva. Esta diminuição é medida por circuitos localizados em cada canto da tela e as informações são enviadas para o controlador para serem processadas. O computador calcula, a partir das diferenças relativas de carga em cada canto, exatamente onde o evento de toque ocorreu, e repassa essa informação para o controlador *touch screen*.

Telas sensíveis ao toque capacitivas devem ser tocadas com o dedo, ao contrário dos painéis resistivos, onde se pode usar tanto luva, quanto caneta. Telas capacitivas possuem brilho e nitidez melhores do que telas resistivas, pois possuem menos camadas, e não há

partes móveis para desgastar. Líquidos, sujeira, gordura ou outros contaminantes também não lhes causam problemas.

### 3.2.2.1. Tecnologia Capacitiva Superficial

Nesta tecnologia, apenas uma camada condutora é revestida de ITO. Uma pequena tensão é aplicada à camada, resultando em um campo eletrostático uniforme. Quando um condutor, a exemplo de um dedo humano, toca a superfície não revestida, um capacitor é dinamicamente formado. O sensor controlador pode determinar a localização do toque indiretamente a partir da mudança na capacitância medida a partir dos quatro cantos do painel. Como ele não tem partes móveis, é moderadamente durável, tem resolução limitada e é propenso a falsos sinais. Por isso, é mais frequentemente usado em aplicações simples, como controles industriais e quiosques. A Figura 3.6 ilustra uma tela com tecnologia capacitiva superficial e seu funcionamento.

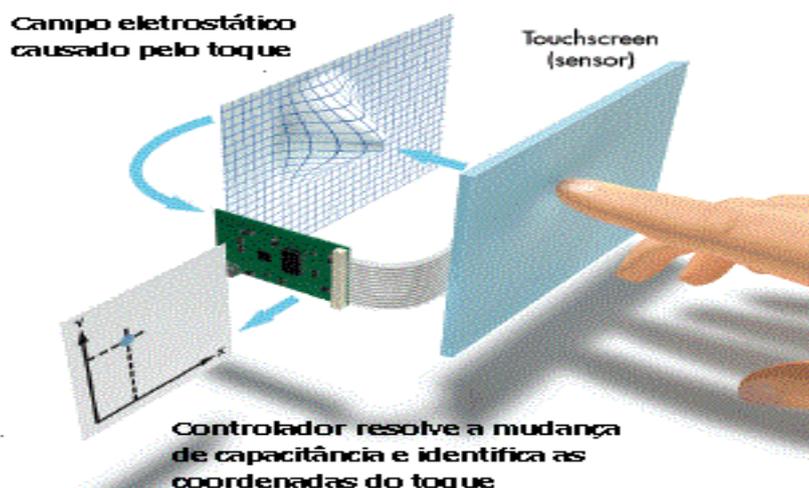


Figura 3.6: *Touch screen* com tecnologia capacitiva superficial. Fonte: [26]

### 3.2.2.2. Tecnologia Capacitiva Projetada

O projeto de telas sensíveis ao toque com tecnologia capacitiva projetada assemelha-se ao de telas sensíveis ao toque resistivas, na medida em que ambos utilizam duas camadas de ITO, com tiras condutoras de medição perpendiculares nas extremidades de cada camada, que são encaixadas entre duas camadas de vidro. Esta grade, formada pelas camadas condutoras perpendiculares, projeta um campo elétrico através da camada superior de vidro. Devido a esta projeção, quando o utilizador toca na camada superior de vidro, ele altera os valores de

capacitância medidos dos eletrodos mais próximos a ele, devido a pequena carga electromagnética contida no corpo humano. Estas alterações na capacitância são medidas e calculadas como os pontos de toque utilizando os componentes  $X$  e  $Y$ .

Telas sensíveis ao toque com tecnologia capacitiva projetada são claras, duráveis e resistentes a riscos. A Figura 3.7 ilustra uma tela com tecnologia capacitiva projetada.

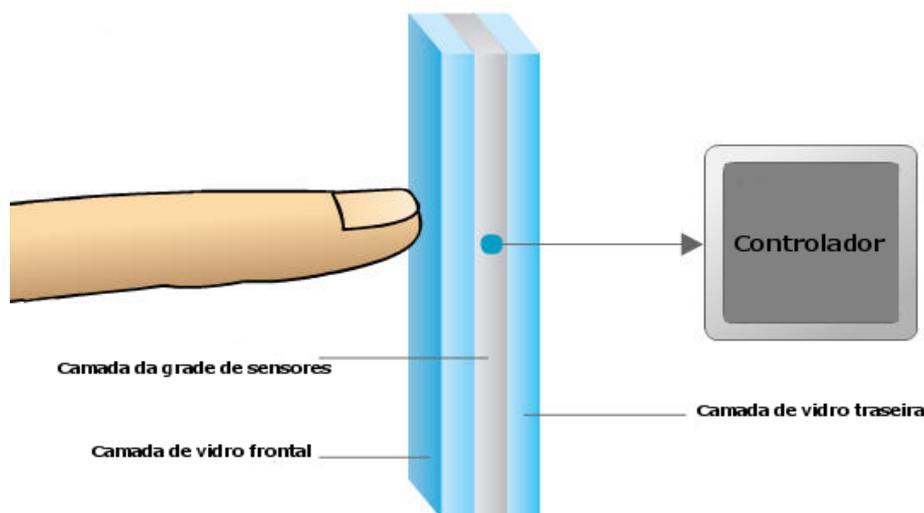


Figura 3.7: *Touch screen* com tecnologia capacitiva projetada. Fonte: [27]

### 3.2.3. Infravermelho

No *touch screen* infravermelho, há um quadro sensível ao toque que geralmente é instalado na frente da tela. Esse quadro é integrado com placa de circuito impresso que contém uma linha de LEDs infravermelhos e foto-transistores escondidos atrás do painel do quadro de toque. Cada um dos LEDs infravermelhos e foto-transistores são colocados em lados opostos para criar uma grade de luz infravermelha invisível. O painel protege as partes do ambiente de operação, enquanto permite que os feixes infravermelhos passem.

O controlador do *touch screen* infravermelho pulsa os LEDs sequencialmente para criar uma grade de feixes de luz infravermelhos. Quando um usuário toca a tela, há uma interrupção nos feixes de luz infravermelhos, assim, os foto-transistores dos eixos  $X$  e  $Y$  detectam os feixes de luz infravermelha que foram interrompidos e transmitem sinais exatos que identificam as coordenadas dos eixos  $X$  e  $Y$ . A Figura 3.8 ilustra uma tela com tecnologia infravermelho com os foto-transistores e grade formada pelos feixes de luz infravermelhos.

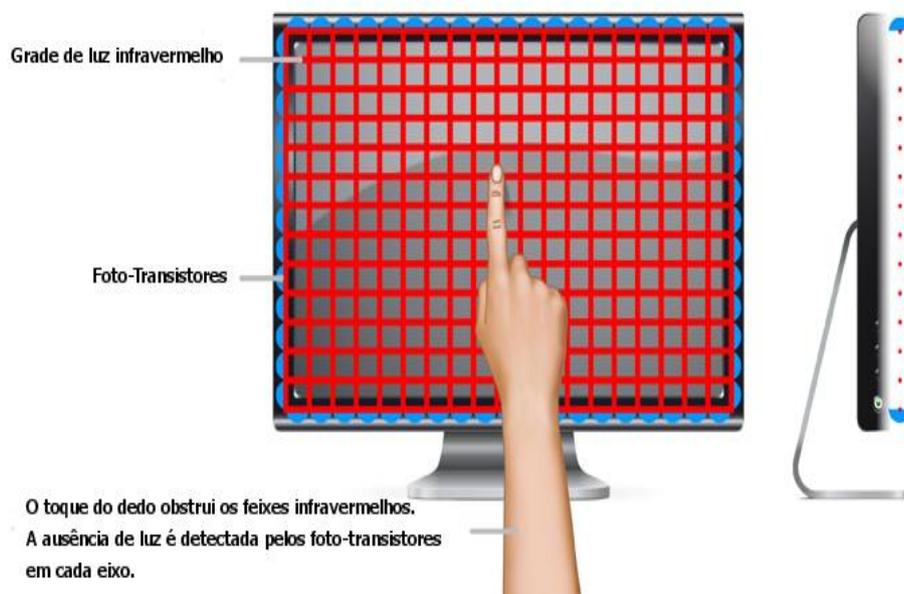


Figura 3.8: *Touch screen* com tecnologia infravermelho. Fonte: [28]

Como vantagem, o *touch screen* infravermelho é resistente, com a tela quebrando somente com impactos fortes, e também funciona com luva ou qualquer outro objeto. Como desvantagem, apesar de ter uma boa visibilidade, a luz muito forte do ambiente pode ter um impacto sobre a precisão da tela. Outro problema é que o custo desta tecnologia é relativamente elevado. O grande problema é que o painel de toque fica um pouco acima da tela, tornando suscetível a ativação precoce antes do dedo ou a caneta realmente tocar a superfície.

As telas com tecnologia infravermelha são frequentemente utilizadas em aplicações médicas e industriais, porque podem ser completamente seladas e operadas utilizando qualquer tipo de material, duros ou macios. É utilizada também em quiosques, caixas eletrônicos, pontos de venda e sistemas de controle de processos.

#### 3.2.4. Superfície de Onda Acústica

A tecnologia superfície de onda acústica é baseada em dois transdutores – um transmitindo e o outro recebendo - colocados nos eixos  $X$  e  $Y$  do painel de toque. O outro elemento importante da superfície de onda acústica é um refletor colocado sobre o vidro. O controlador envia um sinal elétrico para o transdutor transmissor, e o transdutor converte o sinal em ondas ultrassônicas e emite para refletores que estão alinhados ao longo da borda do painel. Após isso, os refletores refratam ondas para os transdutores receptores, o transdutor receptor

converte as ondas num sinal elétrico e envia de volta para o controlador. Quando o usuário toca a superfície do vidro, o dedo do utilizador absorve uma parte da energia da onda acústica e o circuito controlador mede a localização toque. A Figura 3.9 traz uma tela que possui tecnologia de superfície de onda acústica.

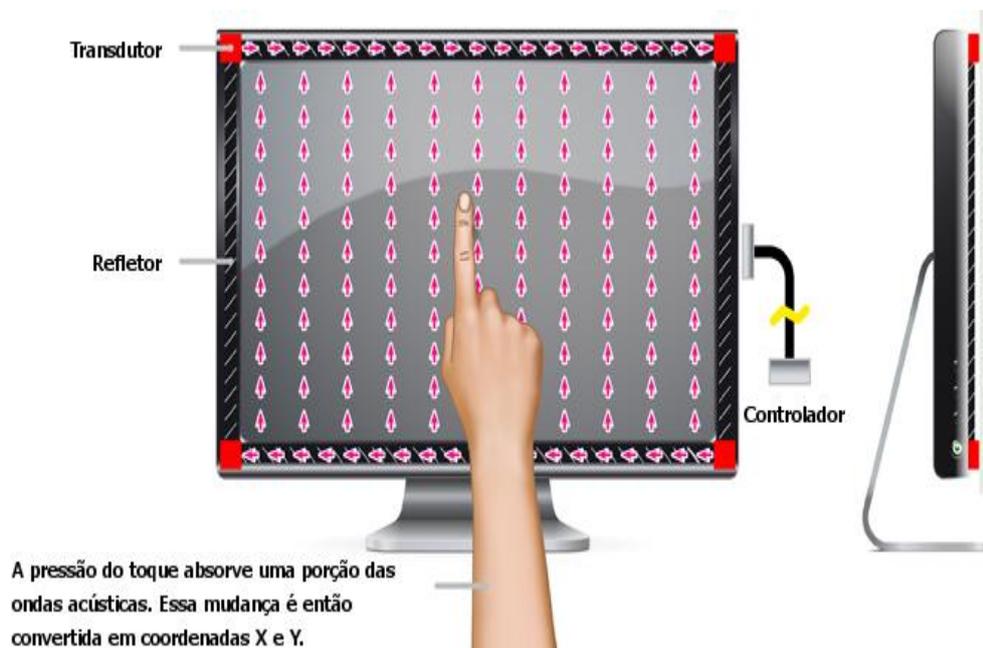


Figura 3.9: *Touch screen* com tecnologia superfície de onda acústica. Fonte: [28]

Comparado com as tecnologias resistiva e capacitiva, a tecnologia superfície de onda acústica proporciona uma superior nitidez de imagem, resolução e transmissão de luz porque utiliza em sua construção vidro puro. Como o painel é todo de vidro, não há camadas que podem sofrer desgaste, dando a esta tecnologia o maior fator de durabilidade e também a maior clareza. Desvantagens da tecnologia superfície de onda acústica incluem a dificuldade de industrialização, bem como o sinal pode ser afetado por líquidos ou outros contaminantes na superfície. O seu uso é recomendado para aplicações bancárias e financeiras, quiosques de informação pública, videogames e aplicações industriais.

### **3.3. *Pen Computing***

*Pen Computing* refere-se a computadores que utilizam uma caneta (*pen light* ou uma *stylus*) em vez de teclado e mouse como periféricos. Dispositivos como PDAs, telefones celulares, *smartphones*, *tablets* e GPS e outros dispositivos com tela *touch screen* são os que comumente utilizam *Pen Computing*.

A tela de toque usa o reconhecimento de gestos ou o reconhecimento de escrita para transmitir o item clicado ou a informação escrita na tela do computador. Reconhecimento de gestos é usado para identificar a entrada para executar um comando, enquanto o reconhecimento de escrita é usado para traduzir a entrada manuscrita em texto digital. Os usuários podem escrever com caracteres normais ou, em alguns casos, caracteres especiais. Em alguns casos, a determinação de se um caractere é usado como texto ou um comando se baseia em onde o caractere é inserido.

### 3.4. Lousa e Carteira Digital

A carteira e a lousa digital são projetos desenvolvidos no Centro de Tecnologia da Informação Renato Archer (CTI) em conjunto com a Associação Brasileira de Informática (ABINFO), para a colaboração de um projeto de informatização escolar da Prefeitura de Serrana-SP.

A carteira digital, ilustrada pela Figura 3.10 é basicamente uma mesa escolar metálica de baixo custo com um *display* integrado no topo da mesa, que consiste em uma placa sólida e transparente, como um vidro grosso. Este *display* tem uma tecnologia de *tablet* baseada em caneta, ou alternativamente, uma tecnologia *touch screen*, o que permite uma interação direta com a imagem em exibição na mesa.



Figura 3.10: Carteira Digital desenvolvida pelo CTI

A lousa digital, ilustrada pela Figura 3.11 é feita com placas de madeira compensada recobertas com fórmica branca brilhante nas faces anteriores, sendo que nas faces posteriores são aplicadas camadas recobertas por um substrato condutor. A imagem é exibida na lousa por meio de um projetor posicionado na face posterior da lousa. A lousa também possui a tecnologia de *tablet* baseada em caneta que permite interação com a imagem em exibição na lousa.

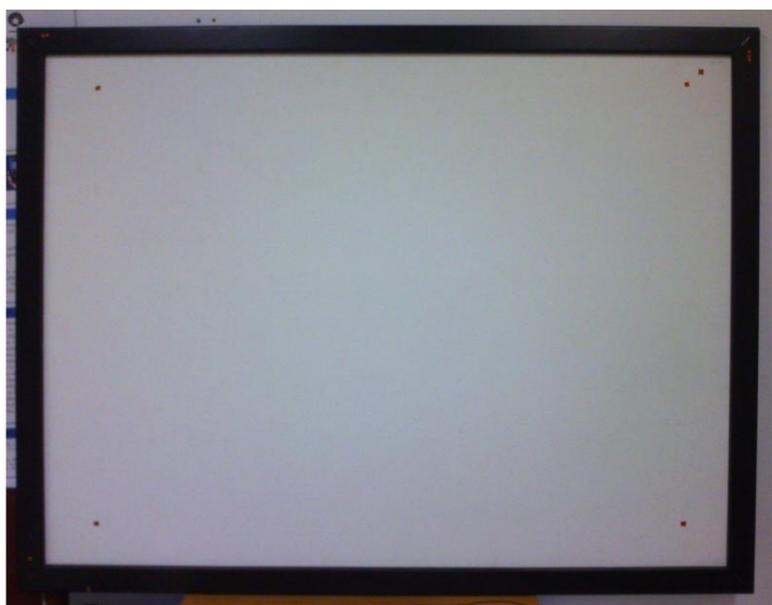


Figura 3.11: Lousa interativa desenvolvida pelo CTI

A tecnologia utilizada pela lousa e pela carteira é 100% nacional. Essa tecnologia foi desenvolvida originalmente na década de 1980, e, diferentemente da tendência internacional, empregava filmes transparentes e condutores de dióxido de estanho ( $\text{SnO}_2$ ), cuja pesquisa se iniciara na UNICAMP na década de 1970 e resultou em uma patente do processo de obtenção dos filmes [29], e noutra do processo a plasma iônico para sua foto gravação com configurações de alta resolução [30].

A tecnologia empregada nos protótipos utilizados neste trabalho será apresentada na sequência, bem como sua arquitetura e princípio de operação.

### **3.4.1. Arquitetura e Princípio de Operação da Lousa e Carteira Digital Indutiva**

A tecnologia empregada na lousa e na carteira digital utiliza o princípio indutivo, bem como os equipamentos utilizados neste trabalho. O display da carteira digital é composto por uma placa retangular de vidro sódio-cálcico transparente, recoberto por um filme fino de

dióxido de estanho (SnO<sub>2</sub>), que fornece grande transmitância<sup>3</sup> de luz. O display da lousa digital é feito de materiais não transparentes (folha de madeira compensado, por exemplo), sobre o qual se justapõe uma folha de material condutor de várias naturezas. Neste caso, a lousa opera com um projetor digital frontal que projeta a imagem na face condutiva da lousa. O dióxido de estanho (SnO<sub>2</sub>) acaba sendo o material condutor mais utilizado, devido as propriedades químicas e mecânicas, que permitem com que a tela seja tocada diretamente por ponteiros, como as metálicas ou as feitas de grafite [31].

A interação do usuário com a tela *touch screen* é feita somente por meio de uma ponteira (caneta), que simula um mouse, e permite ao usuário realizar as operações do mesmo. A ponteira opera como uma antena, que lê os sinais enviados a partir de um circuito de controle e sensoramento. A leitura das coordenadas *X* e *Y* é efetuada aproximando-se a ponteira diretamente sobre a camada indutiva transparente. Uma diferença de potencial aplicada entre as extremidades (horizontais ou verticais) da camada condutora se distribui uniformemente, e cada ponto corresponde a uma tensão que é lida pela ponteira.

Para que a relação entre as tensões lidas e as coordenadas do ponto de toque seja linear, as linhas de corrente na direção *x* e na direção *y* devem ser paralelas, o que requer que a tensão seja aplicada através de contatos distribuídos ao longo das bordas da placa, alternadamente em *X* e *Y*. Assim, empregaram-se comutadores eletrônicos para alimentar independentemente o resistor na direção *X* e na direção *Y*, isolando-se os contatos horizontais e verticais aplicados nas bordas da tela com chaves constituídas por diodos semicondutores uniformemente distribuídos ao longo das bordas [31].

A tensão lida pela ponteira é então enviada para um circuito análogo-digital de múltiplos canais. Sua saída alimenta um microcontrolador, responsável por executar as tarefas de aquisição, comutação, transmissão serial e pré-processamento de dados. O microcontrolador também faz a filtragem digital de séries de leituras de coordenadas, eliminando componentes de alta frequência que são gerados pela comutação das coordenadas *X* e *Y* ou pela ponteira no instante em que o toque acontece [31].

A ponteira tem por objetivo emular um *mouse*, para tanto, os dados das coordenadas devem ser codificados e empacotados, seguindo o protocolo definido pelo periférico. Assim,

---

<sup>3</sup> Transmitância é a quantidade de luz visível que pode passar por uma lente.

os dados são empacotados utilizando-se um *byte* para delimitar o início, dois *bytes* representam o valor de *X*, um *byte* de separação e dois *bytes* representando o valor de *Y*, com os valores de *X* e *Y* variando entre 0 e 255 [31]. Após serem empacotados, os dados são enviados ao microcomputador, na forma serial, no padrão RS232 [32]. A Figura 3.12 mostra o diagrama de blocos do sistema que compõe a carteira e a lousa digital.

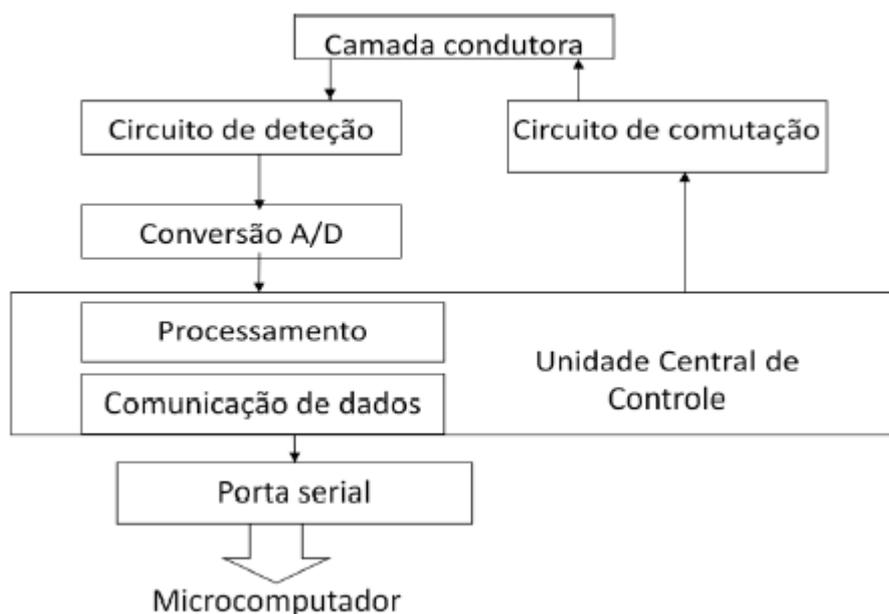


Figura 3.12: Diagrama de blocos do sistema da carteira e da lousa. Fonte: [31]

A arquitetura foi concebida para que o dispositivo pudesse ser conectado a qualquer microcomputador que dispusesse de uma interface serial padrão RS232, sendo sua conexão efetuada por meio de um cabo comum com conector de 9 pinos (DB9).

Em relação aos componentes de software empregados no Projeto, foi desenvolvido um software para operação do microcontrolador em Assembler, que emprega funções e um empacotamento de dados que simplificam o desenvolvimento de aplicações. Na comunicação com o computador, feita via porta serial, há um *device driver* implementado em C que recebe as informações advindas do microcontrolador e obtém os valores das coordenadas *X* e *Y*. Há também um software chamado Fercal, que é utilizado para calibração e correção das não linearidades da superfície resistiva.

O uso da tecnologia indutiva traz algumas vantagens sobre a tecnologia resistiva usada no início, como a não necessidade de contato entre a ponteira e a tela, evitando a exposição da tela resistiva e dos circuitos que a alimentam, também há a possibilidade de ser construída com camadas condutoras menos exigentes quanto às propriedades elétricas e mecânicas, bem como a possibilidade de detectar a pressão da ponteira sobre a tela, à semelhança do lápis sobre o papel [31].

A tecnologia indutiva apresenta um problema de deriva do cursor exibido na tela. A deriva é observada enquanto a ponteira é aproximada da tela, ou seja, quando a ponteira sofre deslocamento na coordenada Z até tocá-la, condição em que o cursor sofre um deslocamento indesejável nas demais direções. Este efeito compromete seriamente a utilização dos dispositivos por dificultar a função de apontar que antecede todas as demais funções como arrastar, marcar, traçar, transladar e copiar [31]. Este problema acaba ocorrendo principalmente na lousa digital, devido ao tamanho da tela *touch screen*. A tensão presente na camada condutora não fica perfeitamente uniforme, fazendo com que alguns pontos apresentem valores maiores que outros, acabando por dificultar a leitura por parte da caneta. Pesquisadores do CTI estão estudando algumas soluções para esse problema, como a troca de material condutor ou dos componentes responsáveis por gerar a tensão na superfície condutora.

### **3.5. Touch Screen e Deficiência Visual**

A difusão das telas sensíveis ao toque tem apresentado vantagens de usabilidade para muitos usuários como, capacidade de apresentar um conteúdo interativo variado num único espaço, possibilidade de controlar diretamente o conteúdo exibido na tela por meio dos dedos, entre outras vantagens. No entanto, pessoas com deficiência visual muitas vezes experimentam desafios significativos ao interagir com interfaces *touch screen* [33]. A característica principal de telas sensíveis ao toque é a sua capacidade de permitir aos usuários manipular diretamente as informações com as pontas dos dedos [34], mas esta capacidade muitas vezes apresenta desafios para usuários com deficiência visual, que não podem ver ou sentir a informação visual apresentada.

Esses usuários podem encontrar vários desafios ao utilizar telas *touch screen*, como:

- Falta de *feedback* auditivo e tátil enquanto executam suas ações;
- Falta de capacidade para determinar o estado atual do dispositivo *touch screen*;
- Dificuldade em selecionar o item desejado na tela.

Esses problemas fazem com que os usuários portadores de deficiência visual fiquem dependentes de uma pessoa sem deficiência para a utilização dessa tecnologia, ou necessitem de uma interface acessível alternativa, ou até mesmo serem incapazes de usar tal tecnologia [35].

Atualmente poucos dispositivos com tecnologia *touch screen* fornecem algum suporte de acessibilidade para usuários com deficiência visual. Enquanto alguns projetos de investigação tentaram aumentar a acessibilidade dos sistemas baseados em tela de toque, estes sistemas exigiam tanto uma tela sensível ao toque e hardware adicional, como um teclado externo ou uma sobreposição tátil [35]. Uma vez que este equipamento não está sempre disponível e é caro, estes sistemas não são seguramente acessíveis.

Outra alternativa acessibilidade em telas sensíveis ao toque é por meio do fornecimento de *feedback* tátil ou auditivo. Uma solução que vem sendo empregada é o uso de leitores de tela, principalmente em *smartphones*. Essa solução, no entanto, ainda apresenta algumas limitações [36]. Primeiramente, estas soluções são tipicamente limitadas a fornecer *feedback* de áudio, não fornecendo assim *feedback* tátil. Em segundo lugar, essa solução é focada principalmente no fornecimento de acessibilidade para telefones móveis, assim, dispositivos como mesas interativas ou quiosques *touch screen*, podem apresentar desafios adicionais.

Este trabalho tem por objetivo propor uma solução para tornar a tecnologia *touch screen* acessível para usuários com deficiência visual. Para isso, serão integrados recursos de acessibilidade da plataforma PlatMult aos dispositivos *touch screen* (lousa e carteira) desenvolvidos pelo CTI. Os resultados obtidos com esta integração serão apresentados no próximo capítulo, assim como os problemas encontrados e outras observações.

## **Capítulo 4**

# **Integração dos Recursos da Plataforma PlatMult na Lousa e Carteira Digitais**

Este capítulo apresenta os resultados obtidos com a integração, através dos estudos técnicos e de compatibilidade feitos na carteira e na lousa, bem como por meio de testes realizados com os recursos de acessibilidade da plataforma PlatMult integrados na lousa e carteira digitais.

### **4.1. Etapas na Integração**

O processo de integração dos recursos tátil, auditivo e visual fornecidos pela plataforma PlatMult nas soluções desenvolvidas pelo CTI foi dividido em três etapas: estudo técnico, verificação de compatibilidade e teste de integração.

#### **4.1.1. Estudo Técnico**

Inicialmente, foi realizado um estudo técnico da carteira e da lousa digital, de modo a conhecer sua arquitetura, princípios de operação e familiarizar-se com os dispositivos. Este estudo foi realizado através da leitura de documentos disponibilizados pelo CTI, bem como por meio de testes operacionais feitos com os dispositivos. Ao utilizar a lousa interativa, pode-se notar um deslocamento do cursor quando a ponteira se aproximava da tela. Para melhor entender este problema, procurou-se observar estes deslocamentos aproximando e afastando a ponteira em vários pontos da tela. Notou-se que durante o movimento de aproximação da ponteira, o cursor aproximava-se do ponto de toque, até coincidir com ele no

instante do toque. Ao afastar a ponteira, a partir do ponto de toque estes deslocamentos resultavam em um desvio da posição do cursor na tela em relação à ponteira.

#### **4.1.2. Verificação de Compatibilidade**

Outra etapa realizada no processo de integração, foi a verificação de compatibilidade entre SO, pacotes e APIs utilizadas na plataforma PlatMult e nas soluções do CTI. Ambos utilizam Sistema Operacional Linux, distinguindo em relação à distribuição utilizada. A plataforma PlatMult executa sobre a distribuição Ubuntu 10.10, e com a possibilidade de ser portado para outras distribuições, como Debian, Fedora e Linux Educacional. Os softwares necessários para o funcionamento (*driver* e Fercal) da carteira e da lousa não foram desenvolvidos para uma distribuição específica, podendo assim serem utilizados em qualquer distribuição Linux.

No que tange a plataforma PlatMult, os problemas estão presentes nos softwares responsáveis pelos três recursos de acessibilidade. Com o xLupa, o problema está relacionado ao servidor gráfico XORG, já que as versões mais atuais destas distribuições utilizam outro ambiente de trabalho (Unity, por exemplo) ou versões mais novas do XORG. Já os recursos auditivo e tátil, que utilizam a AT-SPI como base para o seu funcionamento, encontram problemas em relação a versão da AT-SPI utilizada nas distribuições mais atuais, ficando assim impossível sua execução. Em relação à carteira e à lousa, o problema ocorre com o *driver*, pois o mesmo faz uso de algumas bibliotecas relacionadas ao pacote XFree86, que não estão presentes nas versões atuais das distribuições utilizadas. Assim, foi necessário utilizar distribuições mais antigas para que a integração ocorresse, e os testes pudessem ser realizados.

#### **4.1.3. Teste de Integração**

Os testes de integração na carteira foram realizados utilizando a distribuição Debian 5 (Lenny) e Ubuntu 10.10 (Maverick Meerkat), enquanto que na lousa foram utilizadas as distribuições Fedora 13 (Goddard) e Ubuntu 11.04 (Natty Narwhal). Versões atuais destas distribuições não foram utilizadas devido a problemas de compatibilidade entre APIs, pacotes e bibliotecas necessários para o funcionamento tanto da plataforma PlatMult quanto dos dispositivos do CTI.

Os testes foram realizados utilizando computadores com as seguintes especificações:

- **Plataforma:** x86;
- **Memória RAM:** 2 GB;
- **Processador:** Intel Atom CPU 230 @ 1.60GHz;
- **Disco Rígido:** 320 GB.

A Figura 4.1 traz um exemplo de uso da lousa interativa já com os recursos integrados. A ponteira possui um mecanismo de vibração que é acionado quando o cursor do mouse passa por cima de uma imagem, link ou caixa de texto em uma página web. Nestes mesmos casos, o leitor de tela é ativado, e o texto do link é lido para o usuário.



Figura 4.1: Lousa interativa com os recursos de acessibilidade da plataforma PlatMult

A Tabela 4.1 apresenta a porcentagem de uso da CPU e de uso da memória, enquanto eram utilizados os recursos tátil, auditivo e visual, além do Firefox. A aplicação Injetor é responsável por injetar os eventos ocorridos no Firefox na AT-SPI, e o Pulseaudio é o servidor de som do Linux.

Tabela 4.1: Porcentagem do uso de CPU e memória durante navegação na internet utilizando todos os recursos de acessibilidade

	<b>Uso CPU</b>	<b>Uso Memória</b>
<i>Driver</i>	1-2	0-1
<i>Mouse Feedback</i>	5-10	0.5-1
<b>Injetor</b>	20-30	0.5-1
<b>Firefox</b>	15-35	5-7
<b>xLupa</b>	30-45	1.5-3
<b>XORG</b>	30-45	1.5-3
<b>Pulseaudio</b>	15-25	0.5-1

Nos testes, o uso total da CPU fica entre 85% quando os eventos não estão sendo gerados e 95% quando os eventos são gerados. Deste total, o xLupa juntamente com o XORG é responsável por 60 a 70% do processamento, independentemente dos eventos estarem ou não sendo gerados. A utilização da CPU se torna ainda maior quando é feita a utilização do Firefox e a captação dos eventos gerados pelo mesmo.

Os resultados de desempenho apresentados na Tabela 4.1 mostraram-se iguais durante a utilização da lousa e da carteira.

## 4.2. Problemas Encontrados

Nos testes realizados, identificou-se um problema de desempenho que prejudica a utilização do sistema pelo usuário. O problema ocorre devido ao grande uso da CPU pelos aplicativos de acessibilidade, o que acaba por gerar uma lentidão do sistema e a alguns momentos de travamento. O recurso auditivo também apresenta alguns problemas, como a não leitura de alguns eventos e lentidão na fala.

Outro problema identificado com a integração acontece durante a utilização do xLupa. Ao escolher um fator de ampliação, problemas em relação a calibração da ponteira ocorrem, fazendo com que o deslocamento entre o cursor e a ponteira se agravem. Quanto maior o fator de ampliação, maior esse deslocamento fica, prejudicando assim a utilização da lousa e da carteira por parte do usuário.

Os problemas identificados durante as etapas da integração, também ocorreram tanto na lousa quanto na carteira, pois elas utilizam o mesmo conjunto de *driver* e tecnologia, diferindo apenas no tamanho da tela.

### 4.3. Possíveis Soluções

No que diz respeito ao desempenho, uma solução foi a utilização de um computador com um processador mais potente, possibilitando que os eventos possam ser gerados e tratados corretamente. Os testes foram realizados utilizando computadores com as seguintes especificações:

- **Plataforma:** x86;
- **Memória RAM:** 2 GB;
- **Processador:** 2x Intel(R) Core(TM)2 Duo CPU E8200 @ 2.66GHz;
- **Disco Rígido:** 80 GB.

A Tabela 4.2 apresenta a porcentagem de uso da CPU e de uso da memória, enquanto eram utilizados os recursos tátil, auditivo e visual, além do Firefox. A aplicação Injetor é responsável por injetar os eventos ocorridos no Firefox na AT-SPI, e o Pulseaudio é o servidor de som do Linux.

Tabela 4.2: Porcentagem do uso de CPU e memória durante navegação na internet utilizando todos os recursos de acessibilidade

	<b>Uso CPU</b>	<b>Uso Memória</b>
<i>Driver</i>	1-2	0-1
<i>Mouse Feedback</i>	1-5	0.5-1
<b>Injetor</b>	3-10	0.5-1
<b>Firefox</b>	5-15	3-5
<b>xLupa</b>	15-25	1.5-3
<b>XORG</b>	10-20	1.5-3
<b>Pulseaudio</b>	1-5	0.5-1

Nos testes, o uso total da CPU fica entre 60% quando os eventos não estão sendo gerados e 70% quando os eventos são gerados, uma diminuição de 20% em relação aos dados apresentados na Tabela 4.1. Esta maior capacidade de processamento melhorou a utilização

do computador em conjunto com os recursos de acessibilidade ativados, já que não ocorreram travamentos e lentidão durante a utilização do sistema.

Em relação ao desvio do ponteiro com o uso de diferentes fatores de ampliação, algumas tentativas foram feitas para corrigir a calibração do *driver* enquanto o xLupa era executado. Após alguns testes foi verificado que durante a utilização do xLupa, os valores de coordenadas de tela que a ponteira do *driver* lia e as coordenadas do ponteiro do mouse na tela eram diferentes. Para tentar corrigir este problema, foi adotado uma estratégia em que os valores das coordenadas do *driver* eram gravados em um arquivo e lidos pelo ampliador de tela. Ao ler esses dados, o xLupa ainda aplicava o fator de ampliação usado no momento. Alguns testes foram feitos com esse método, mas os resultados não foram satisfatórios. Outra tentativa foi aplicar essa estratégia em uma versão mais nova do *driver* da carteira e da lousa, porém os resultados também foram os desejados.

#### **4.4. Teste de Usabilidade**

O teste de usabilidade avalia a usabilidade de sistemas interativos por meio da observação de usuários reais realizando tarefas típicas e pré-definidas, no intuito de melhorar a facilidade de uso do sistema interativo, avaliar o grau de sucesso que um usuário atinge ao realizar uma tarefa, o número de erros cometidos pelo usuário e classificar estes erros, além de determinar o grau com que o sistema se ajusta e apoia as necessidades dos usuários [37].

Através deste método procura-se quantificar o desempenho do usuário. Por isso, para cada medida a ser observada, deve-se definir quais são os limites mínimos aceitáveis, os máximos possíveis e também o valor almejado para a medida no projeto. Alguns exemplos de medidas comumente utilizadas no teste de usabilidade são: número de tarefas completadas, tempo gasto para se executar uma tarefa, número de erros cometidos, porcentagem de usuários a conseguirem se recuperar de um erro, porcentagem de usuários a se dizerem satisfeitos com a aplicação, entre outras.

Para realizar um teste de usabilidade e avaliar a experiência de uso de um sistema por meio da observação, Silva & Barbosa [38] recomendam um série de atividades planejadas a fim de obter resultados mais eficazes, conforme Tabela 5.2 (adaptada de [38]).

Tabela 4.3: Atividades do Teste de Usabilidade

<b>Etapa</b>	<b>Descrição</b>
Preparação	<ul style="list-style-type: none"> <li>- definir tarefas para os participantes executarem;</li> <li>- definir o perfil dos participantes e recrutá-los;</li> <li>- preparar material para observar e registrar o uso;</li> <li>- executar um teste-piloto.</li> </ul>
Coleta de Dados	- observar e registrar a performance e a opinião do participantes durante sessões de uso controladas.
Interpretação e Consolidação de Resultados	- reunir, contabilizar e sumarizar os dados coletados dos participantes de acordo com seu contexto.
Relato dos Resultados	- relatar a performance e a opinião dos participantes.

## 4.5. Metodologia da Avaliação de IHC

O método de avaliação escolhido para avaliação da integração foi o Teste de Usabilidade (descrito na Seção 5.4.1). As métricas para o Teste de Usabilidade foram definidas pelo avaliador, que também foi responsável por coordenar e aplicar os testes, além de acompanhar a realização das tarefas próximo aos usuários. A avaliação foi feita utilizando-se a carteira digital devido à presença de um protótipo na Universidade.

Durante as sessões de teste, os usuários foram instruídos a descrever o que estavam pensando e quais as opções tomadas por eles para a realização das tarefas. Com isso, foi possível fazer anotações sobre como os usuários estavam vendo o sistema e como se comportavam diante das diversas situações.

O roteiro aplicado no experimento foi elaborado de forma a refletir tarefas cotidianas aos usuários e a proposta de uso da carteira digital. Durante os testes, os usuários realizaram uma série de tarefas em um cenário de navegação na internet, onde para cada tarefa foi possível avaliar o grau de sucesso, identificar problemas na utilização da plataforma e erros cometidos pelo usuário. Na sequência é feita uma descrição das tarefas definidas para avaliação:

- **Tarefa 1 – Iniciar o Firefox e navegar no Facebook:** nesta tarefa o usuário deve iniciar o navegador *Firefox* por meio ícone que o representa posicionado ao lado direito da palavra *Sistema* no menu superior, ou seguindo o endereço *Aplicativos* → *Internet* → *Navegador Web Browser*. Após iniciar o navegador, o usuário deverá inserir na barra de endereços o endereço especificado ([www.facebook.com.br](http://www.facebook.com.br)) e prosseguir (com a tecla *Enter* ou botão *Ir* do navegador). Na sequência o usuário deve inserir seus dados nos campos *E-mail ou telefone* e *Senha* e clicar no botão *Entrar*. Após entrar em sua página pessoal, o usuário poderia fazer qualquer ação.
- **Tarefa 2 – Iniciar o Firefox e buscar um CEP:** nesta tarefa o usuário deve iniciar o navegador *Firefox* por meio ícone que o representa posicionado ao lado direito da palavra *Sistema* no menu superior, ou seguindo o endereço *Aplicativos* → *Internet* → *Navegador Web Browser*. Após iniciar o navegador, o usuário deverá inserir na barra de endereços o endereço especificado ([www.correios.com.br](http://www.correios.com.br)) e prosseguir (com a tecla *Enter* ou botão *Ir* do navegador). Na sequência o usuário deve selecionar a opção *Busca CEP* na parte superior do site, e após a página carregar, deve inserir o CEP desejado no campo *Busca CEP – Endereço* e clicar no botão *Buscar*.

Ao término dos testes o participante respondeu a um questionário de pós-teste indagando sobre a utilidade do ambiente – ou de recursos específicos – em situações cotidianas, as dificuldades de interação, seus motivos e formas alternativas para facilitar a utilização. Abaixo, seguem as perguntas presentes no questionário pós-teste:

1) *Você considera que em uma situação real, a presença dos recursos de acessibilidade seriam úteis?*

- *Quais são esses recursos e por que seriam úteis?*

2) *Em quais momentos sentiram mais dificuldades ao interagir com a carteira? Quais foram essas dificuldades?*

3) *Consideram que o recurso 'x' presente no carteira auxilia ou atrapalha na interação?*

- *Se auxilia, por que o recurso 'x' auxilia?*
- *Se atrapalha, por que o recurso 'x' não auxilia ou atrapalha?*

4) *Faça outros comentários se desejar.*

O objetivo desse questionário foi medir o grau de satisfação dos usuários mediante os recursos apresentados e receber dos usuários o *feedback* relativo às suas experiências de uso com a plataforma.

As características do ambiente computacional utilizado nas sessões de teste são as seguintes:

- **Sistema Operacional:** Debian versão 5.0.10, Kernel Linux 2.6.26-2-686 (i686), GNOME 2.20;
- **Processador:** 2x Intel(R) Core(TM)2 Duo CPU E8200 @ 2.66GHz;
- **Memória:** 2Gb;
- **HD:** 80Gb;

#### **4.5.1. Usuários Utilizados na Avaliação**

Inicialmente, a ideia era realizar o teste de usabilidade com usuários baixa visão, porém, como não foi conseguido solucionar em tempo hábil o problema de perda de calibração ao utilizar o xLupa, optou-se por utilizar-se usuários sem deficiência visual. Essa alteração foi feita, pois os usuários com deficiência visual utilizam o ponteiro do mouse para guiar sua navegação pelo Sistema, e com este problema de calibração, sua experiência ficaria muito prejudicada. Assim, foram escolhidos dois alunos do curso de Ciência da Computação para participar da avaliação. Os dois usuários não possuem deficiência visual e com relação às noções de Informática, os dois usuários que realizaram os testes já estão habituados ao uso de computadores, navegação na Internet, mas não tem muita experiência com sistemas Linux. Ambos os usuários possuem grande experiência com dispositivos *touch screen* através de *tablets* e *smartphones*, porém sem a utilização de canetas.

#### **4.5.2. Aplicação do Teste de Usabilidade**

As atividades se iniciaram com um diálogo em torno da avaliação. A ideia era definir não só a sua dinâmica, mas também os propósitos de sua aplicação, onde os participantes puderam sanar suas dúvidas.

Na sequência, os usuários foram conduzidos à carteira digital juntamente ao avaliador para o acompanhamento do teste. Durante a observação do teste, o avaliador anotou os diálogos e fenômenos ocorridos durante a sessão. Além de coletar dados da avaliação, o avaliador também se apresentou como o meio de consulta para os usuários nos momentos que o mesmo não sabia como proceder em alguma atividade. O roteiro de atividades foi entregue aos usuários, para leitura e retirada de dúvidas. Após a leitura do roteiro os usuários iniciaram os testes e ao fim desses testes, foram instruídos a responder o questionário pós-teste (descrito na Seção 4.5.).

Os testes foram realizados somente com o leitor de tela e o mouse com *feedback* motor habilitados. O xLupa não foi utilizado pois os usuários não possuem deficiência visual.

### **4.5.3. Resultados**

O objetivo do teste era avaliar a integração de recursos de acessibilidade na carteira digital, a partir da análise do desempenho de usuários na execução de tarefas com base em um roteiro predefinido. E assim, através de dados obtidos em observações e entrevistas pós-teste, encontrar dificuldades relacionadas ao uso da plataforma, e se possível, posteriormente citar soluções para tais problemas.

Abaixo se encontram alguns relatos descritos pelos participantes no questionário pós-teste, sobre suas experiências.

*1) Você considera que em uma situação real, a presença dos recursos de acessibilidade seriam uteis?*

Os dois participantes responderam que tanto o recurso auditivo quanto o tátil seriam uteis em situação de uso real.

*2) Em quais momentos sentiram mais dificuldades ao interagir com a carteira? Quais foram essas dificuldades? Por que tiveram dificuldades?*

Os dois participantes citaram dificuldades com a utilização da caneta. Eles relataram que a caneta não foi precisa em alguns poucos momentos, como quando desejam selecionar algo na tela (um clique) ou abrir algo (dois cliques). Ainda foi relatado que algumas palavras não foram faladas pelo leitor de tela durante a navegação na Internet.

3) *Consideram que o recurso 'x' presente no carteira auxilia ou atrapalha na interação?*

Os dois participantes citaram que tanto o recurso auditivo quanto o tátil os ajudaram durante a realização das tarefas propostas. Esse auxílio se mostrou muito útil, principalmente durante a navegação na Internet.

4) *Faça outros comentários se desejar.*

Ambos os usuários relataram que gostaram da experiência e da possibilidade de interagir diretamente com os elementos da tela através da caneta, apesar de alguns problemas enfrentados.

#### **4.5.4. Problemas Encontrados durante os testes de usabilidade**

Durante a realização do teste foram identificados dois problemas principais: falta de precisão da caneta e ausência de *feedback* auditivo em alguns momentos.

##### ***Precisão da Caneta***

Sobre a precisão da caneta, primeiramente deve-se fazer um estudo a fim de verificar se o problema é ocasionado pelo *driver* da caneta, ou por causa de problemas durante a leitura da tensão que passa sobre a superfície condutora da tela. No caso do *driver*, uma possibilidade seria fazer alguns testes com a coordenada Z lida pela ponteira. A coordenada Z possui o valor da distância entre a ponteira e tela e indica se houve um toque na tela. Esses testes devem ser feitos a fim de encontrar um valor mais adequado para indicar o toque do usuário na tela. O problema também pode estar relacionado a tela e a superfície, com o defeito podendo estar ocorrendo em algum componente que envia a tensão para a tela, ou até mesmo um defeito na superfície condutora.

##### ***Ausência de Feedback Auditivo***

Esse problema ocorre na identificação dos campos de interesse no Firefox. Como explicado na Seção 2.6.3, o FireDEMO identifica os campos de interesse ao carregar uma página e os informa ao Injetor de Eventos quando o usuário passa com o mouse sobre um deles. Alguns desses eventos identificados acabam por ser passados de maneira incorreta pelo FireDEMO ou são tratados incorretamente pelo Injetor de Eventos, ocasionando assim a ausência de feedback. Como solução, pode-se realizar um estudo mais detalhado sobre os

eventos presentes em uma página do Firefox, para identificá-los e reportá-los com mais precisão, e assim fornecer o *feedback* auditivo corretamente ao usuário.

## Capítulo 5

### Considerações Finais

Este trabalho tem como objetivo principal fornecer acessibilidade para as carteiras e lousas digitais desenvolvidas pelo CTI a partir dos recursos tátil, visual e auditivo presentes na plataforma PlatMult. Para atingir esse objetivo, primeiramente foi necessário realizar um estudo técnico para conhecer e familiarizar-se os dispositivos, bem como fazer uma verificação de compatibilidade a fim de encontrar problemas que pudessem prejudicar ou impossibilitar a integração.

Após essas duas etapas, os recursos de acessibilidade foram implementados na carteira e na lousa, e testes de integração foram realizados. A partir dos testes, pode-se constatar o funcionamento dos recursos em conjunto com os dispositivos e descobrir alguns problemas decorrentes do processo. O principal problema encontrado foi a perda de calibração da caneta ao utilizar a ampliação.

Outra etapa do trabalho foi a avaliação de usabilidade da integração. Essa avaliação baseou-se em um teste de usabilidade. Dois usuários foram recrutados para realização dos testes, cujos relatos possibilitaram obter um *feedback* sobre a integração, bem como encontrar problemas que tinham passados despercebidos nos testes de integração. Nesta etapa pode-se verificar problemas com o leitor de tela e com a precisão da ponteira em alguns momentos.

A integração de recursos de acessibilidade nas carteiras e lousas digitais trouxe vários benefícios. Para o Projeto PlatMult, proporcionou um novo meio de empregar seus recursos de acessibilidade, enquanto que para os Projetos desenvolvidos pelo CTI, possibilitou a adição de recursos de acessibilidade tátil e auditiva a seus dispositivos.

## **5.1. Trabalhos Futuros**

Como trabalhos futuros, pretende-se realizar novas avaliações de IHC sobre a integração. Um novo teste de usabilidade deverá ser aplicado utilizando usuários baixa visão, a fim de avaliar todos os recursos implementados na carteira e na lousa, além de se obter resultados sobre a usabilidade com base no usuário alvo da pesquisa. Também deverá ser realizada uma avaliação empregando o método proposto por Brajnik [39]. Este método é uma adaptação da Avaliação Heurística utilizada para investigações de acessibilidade, onde as diretrizes são substituídas por barreiras.

Outra necessidade é a correção dos problemas de calibração ocorridos quando o xLupa é utilizado. Para isso, deve ser implementado um método de correção da calibração a fim de solucionar esse problema, e possibilitar assim o uso da carteira e da lousa de maneira satisfatória por parte do usuário com baixa visão.

# Apêndice A

## Tutorial de Instalação da Plataforma PlatMult

Instalação física:

- Conectar o mouse à porta paralela do computador.

Instalação dos softwares:

- Instalar o Sistema Operacional Debian 5 (Lenny).
- Criar usuário *root*.
  - No terminal digite: *sudo passwd root*
  - Coloque a senha de sua escolha.
- Configurar para fazer *login* como *root* no sistema.
  - Edite o arquivo: *vi /etc/gdm/custom.conf*
  - E deixe altere as seguintes linhas: *AutomaticLoginEnable=true* e *AutomaticLogin=root*.
- Configurar os repositórios para download dos pacotes necessários.
  - Abrir o arquivo *sources.list* no caminho */etc/apt/*, apagar o conteúdo e adicionar o seguinte:

```
deb http://archive.debian.org/debian/ lenny main non-free
contrib
deb-src http://archive.debian.org/debian/ lenny main non-free
contrib
deb http://archive.debian.org/debian-security/ lenny/updates
main contrib non-free
deb-src http://archive.debian.org/debian-security/
lenny/updates main contrib non-free
deb http://archive.debian.org/debian-volatile lenny/volatile
main contrib non-free
deb-src http://archive.debian.org/debian-volatile
lenny/volatile main contrib non-free
deb http://archive.debian.org/debian-backports lenny-backports
main contrib non-free
```

- Atualizar o gerenciador de pacotes.
- Instalar pacotes relacionados ao At-Spi.
  - Buscar por at-spi no Synaptic e instalar os seguintes pacotes: at-spi, at-spi-doc, libatspi-dev, libatspi1.0-0, libatspi-dbg, python-pyatspi.
- Realizar *checkout* com o *SVN* dos arquivos do PlatMult.
  - Endereço: <https://code.google.com/p/platmult/>
- Instalar o xLupa 4.1, que se encontra dentro da pasta xLupa dos arquivos baixados anteriormente.
- Instalar Mozilla Firefox 3.6.28.
  - Endereço: <https://ftp.mozilla.org/pub/mozilla.org/firefox/releases/3.6.28/>
- Instalar o *plugin* para o Firefox, responsável por identificar os eventos durante a navegação na Internet encontra-se no caminho MouseFeedback2.0→FireDEMO.
- Os demais programas que necessitam ser somente executados encontram-se nos seguintes caminhos:
  - Injetor de eventos, responsável por receber os eventos lidos no Firefox e injetá-los na At-Spi encontra-se no caminho MouseFeedback2.0→Injetor.
  - MouseFeedback, responsável por acionar o *vibracall* no mouse ao ser notificado de um evento da At-Spi encontra-se no caminho MouseFeedback2.0→MouseFeedback.

# Apêndice B

## Tutorial de Instalação da Carteira Digital

Instalação física:

- Conectar microcontrolador à fonte de energia;
- Conectar ponteira ao microcontrolador;
- Conectar o cabo que sai da tela *touch screen* da carteira digital ao microcontrolador;
- Conectar cabo serial no computador e a outra ponta no microcontrolador;
- Conectar cabo VGA do monitor da carteira digital no computador;
- Conectar cabo de força do monitor da carteira digital à energia;
- Conectar mouse, teclado e caixas de som ao computador;
- Conectar fonte à energia.

Instalação dos softwares:

- Instalar o Sistema Operacional Debian 5 (Lenny).
- Criar usuário *root*.
  - No terminal digite: *sudo passwd root*
  - Coloque a senha de sua escolha.
- Configurar para fazer *login* como *root* no sistema.
  - Edite o arquivo: *# vi /etc/gdm/gdm.conf*
  - E altere a linha: *AllowRoot=false*
  - Para true: *AllowRoot=true*
- Configurar os repositórios para download dos pacotes necessários.
  - Abrir o arquivo *sources.list* no caminho */etc/apt/*, apagar o conteúdo e adicionar o seguinte:

```
deb http://archive.debian.org/debian/ lenny main non-free
contrib
deb-src http://archive.debian.org/debian/ lenny main non-free
contrib
deb http://archive.debian.org/debian-security/ lenny/updates
main contrib non-free
```

```
deb-src http://archive.debian.org/debian-security/ lenny/updates
main contrib non-free
deb http://archive.debian.org/debian-volatile lenny/volatile
main contrib non-free
deb-src http://archive.debian.org/debian-volatile lenny/volatile
main contrib non-free
deb http://archive.debian.org/debian-backports lenny-backports
main contrib non-free
```

- Atualizar o gerenciador de pacotes.
- Instalar o *driver*:
  - No diretório raiz do *driver* digite os comandos:
    - *./configure*
    - *sudo make*
    - *sudo make install*
  - Durante a execução do *./configure* pode ocorrer algum erro devido a falta de pacotes. Para solucionar basta instalar o pacote que está faltando.
- Calibração do *driver*.
  - No diretório raiz do software de calibração digite:
    - *./fercal*
    - Aponte a caneta sobre cada ponto mostrado na tela, tecle *Enter* e na sequencia *Space*.
    - Ao fim, tecle *A* para confirmar a calibração e gravar os dados.

## Referências Bibliográficas

- [1] UCA – Um Computador por Aluno. Disponível em: <http://www.uca.gov.br/institucional/index.jsp>. Acesso em: 25 mar. 2013.
- [2] IBGE. Censo Demográfico 2010. Disponível em: <http://www.censo2010.ibge.gov.br/>. Acesso em: 28 mar. 2013.
- [3] O que é Visão Subnormal. [S.l.]: Sociedade Brasileira de Visão Subnormal. Disponível em: <http://www.visaosubnormal.org.br/oquee.html>. Acesso em: 14 jun. 2013.
- [4] BERSCH, Rita. Introdução à Tecnologia Assistiva. Disponível em: <http://www.assistiva.com.br/tassistiva.html>. Acesso em: 28 mar. 2013.
- [5] COOK, A. M.; HUSSEY, S. M. Assistive Technologies: Principles and Practices. 3. ed. Reading: Mosby, 1995.
- [6] BIDARRA, J.; OYAMADA, M. S. Development of an interactive kiosk with screen amplifier targeting low vision and old-aged people. In: AIRTECH - Accessibility, Inclusion and Rehabilitation using Information Technologies. Havana-Cuba: 2011. p. 13-14.
- [7] UBUNTU. Disponível em: <http://www.ubuntu.com/>. Acesso em: 27 mar. 2013.
- [8] XORG. Disponível em: <http://www.x.org/wiki/>. Acesso em: 27 mar. 2013.
- [9] GNOME. Disponível em: <http://www.gnome.org/>. Acesso em: 27 mar. 2013.
- [10] AT-SPI. Disponível em: <http://directory.fsf.org/wiki/At-spi>. Acesso em: 27 mar. 2013.
- [11] BIDARRA, J.; BOSCARIOLI, C.; RIZZI, C. B. xLupa - Um Ampliador de Tela com Interface Adaptativa para Pessoas com Baixa Visão. In: Amanda Meincke Melo; Lara Schibelsky Godoy Piccolo; Ismael Mattos Andrade Ávila; Cláudia de Andrade Tambascia. (Org.). Usabilidade, Acessibilidade e Inteligibilidade Aplicadas em Interfaces para Analfabetos, Idosos e Pessoas com Deficiência. Campinas - São Paulo: CPqD - Centro de Pesquisa e Desenvolvimento em Telecomunicações, 2009, p. 23-30.
- [12] BALANSIN, C. Especificação e Implementação de um Leitor de Tela. Trabalho de Conclusão de Curso: Ciência da Computação. Unioeste Cascavel, 2011.
- [13] LEE, S. Python Powered Accessibility. Disponível em: <http://live.gnome.org/Accessibility/PythonPoweredAccessibility>. Acesso em: 14 jun. 2013.

- [14] ACCERCISER. Disponível em: <https://wiki.gnome.org/Accerciser>. Acesso em: 14 jun. 2013.
- [15] ESPEAK. Disponível em: <http://espeak.sourceforge.net/>. Acesso em: 27 mar. 2013.
- [16] JOHNSON, E. A. Touch Display – A novel input/output device for computers. In: Electronics Letters. 1965. p. 219-220.
- [17] Some Milestones in Computer Input Devices: Na Informal Timeline. Disponível em: <http://www.billbuxton.com/inputTimeline.html>. Acesso em: 14 jun. 2013.
- [18] ELOGRAPH. Disponível em: <http://www.elotouch.com/AboutElo/History/>. Acesso em: 14 jun. 2013.
- [19] HACK, R. L. Interfacing a Plato IV Display Terminal to a PDP-10 Computer. 1. ed. Reading: Defense Technical Information Center, 1973.
- [20] NIMISH, M. A Flexible Machine Interface. Tese (Tese de Doutorado em Engenharia Elétrica) - University of Toronto, 1973.
- [21] HP-150. Disponível em: <http://www.columbia.edu/cu/computinghistory/hp150.html>. Acesso em: 14 jun. 2013.
- [22] MULTI-TOUCH. Disponível em: <http://m.npr.org/story/144185699>. Acesso em: 14 jun. 2013.
- [23] Simon Personal Communicator. Disponível em: <http://cims.ncsu.edu/the-simon-personal-communicator/>. Acesso em: 14 jun. 2013.
- [24] Palm Pilot. Disponível em: <http://www.palminfocenter.com/news/8493/>. Acesso em: 14 jun. 2013.
- [25] Touch Screens. Disponível em: <http://www.screentekinc.com/resource-center/touch-screens.shtml>. Acesso em: 10 jul. 2013.
- [26] Please Touch! Explore The Evolving World Of Touchscreen Technology. Disponível em: <http://electronicdesign.com/components/please-touch-explore-evolving-world-touchscreen-technology>. Acesso em: 10 jul. 2013.
- [27] Touch Screens – Capacitive Touch Screens. Disponível em: <http://www.venturautomation.com/product.php?pid=97>. Acesso em: 10 jul. 2013.
- [28] Optical Touch: How it differs from other technologies. Disponível em: <http://www.nextwindow.com/optical/comparison.html>. Acesso em: 10 jul. 2013.
- [29] MAMMANA, A. P. Obtenção de filmes transparentes condutores. Patente INPI No. 8.002.761, 30/4/80.
- [30] MAMMANA, C. I. Z.; BRAGA, E. S.; MAMMANA, A. P. Foto gravação de materiais condutores transparentes por plasma iônico. Patente INPI no. 8.005.488, 29/8/80.

- [31] ROMANELLI, C. B. Contribuições ao Desenvolvimento de um Tablet Indutivo. Dissertação (Dissertação de Mestrado do Programa de Desenvolvimento Humano e Tecnologias) – Universidade Estadual Paulista, Rio Claro – SP, Abril 2013.
- [32] RS232. Disponível em: <http://www.tscm.com/rs-232.pdf>. Acesso em: 10 jul. 2013.
- [33] KANE, S.K.; BIGHAM, J.P.; WOBROCK, J.O. Slide Rule: Making mobile touch screens accessible to blind people using multi-touch interaction techniques. In: ACM SIGACCESS Conference on Computers and Accessibility. 2008. p. 73–80.
- [34] SHNEIDERMAN, B. Touch screens now offer compelling uses. IEEE Software, New York, v. 8, n. 2, p. 93–94, March 1991.
- [35] FULLY ACCESSIBLE TOUCH SCREENS FOR THE BLIND AND VISUALLY IMPAIRED. Disponível em: <http://faculty.washington.edu/wobbrock/pubs/nish-09.pdf>. Acesso em: 6 jul. 2013..
- [36] KANE, S. K., MORRIS, M. R., PERKINS, A. Z., WIGDOR, D., LADNER, R. E. and WOBROCK, J. O. Access Overlays: Improving nonvisual access to large touch screens for blind users. Proc. UIST'11, ACM Press, 2011, 273–282.
- [37] PRATES, R. O.; BARBOSA, S. D. J. Avaliação de interfaces de usuário - conceitos e métodos. In: Anais do XXIII Congresso Nacional da Sociedade Brasileira de Computação. [S.l.]:SBC, 2003.
- [38] BARBOSA, S. D. J.; SILVA, B. S. da. Interação Humano-Computador. São Paulo, SP, Brasil: Elsevier Editora Ltda., 2010. ISBN 978-85-352-3418-3.
- [39] Barrier Walkthrough. Disponível em: <http://sole.dimi.uniud.it/~giorgio.brajnik/projects/bw/bw.html>. Acesso em: 16 out. 2013.