

# Eduativo – Possibilidades de Interação para Pessoas com Deficiência Auditiva em Software Educacional

Cristina Paludo Santos, Cristiane Ellwanger

Universidade Regional Integrada do Alto Uruguai e das Missões – Campus de Santo Ângelo

Av. Universidade das Missões, 464 – Santo Ângelo – RS

+55 55 3313 7900

paludo@santoangelo.uri.br, cristianeellwanger@gmail.com

## ABSTRACT

Inclusive digital environments involve architectural features presentation of information, which includes requirements for accessibility and usability. This article aims to present the software Eduativo - designed from the layered architecture denominated ISA which favors the integration of interface requirements and pedagogical aspects needed for this type of application. In the validation of the prototype was verified that accessibility and usability aspects depend on the context of use, and user interaction indispensable for implementing improvements in architectural and offering activities that provide autonomy to students with hearing impairments.

## RESUMO

Ambientes digitais inclusivos envolvem características arquiteturais de apresentação de informações que contemplem requisitos de acessibilidade e usabilidade. O presente artigo visa apresentar o software Eduativo - concebido a partir da arquitetura em camadas denominada ISA a qual favorece a integração de requisitos de interface e aspectos pedagógicos necessários a este tipo de aplicação. Na validação do protótipo verificou-se que aspectos de acessibilidade e usabilidade dependem do contexto de uso, sendo a interação do usuário imprescindível para implementação de melhorias tanto em termos arquiteturais quanto no oferecimento de atividades que proporcionem autonomia a educandos com deficiência auditiva.

## Palavras-Chave

Acessibilidade, Documentos Digitais, Ferramenta de Apoio

## 1. INTRODUÇÃO

Facilitar a imersão de educandos em ambientes digitais inclusivos é um dos pressupostos que norteiam o desenvolvimento de sistemas com foco educacional. Entretanto, há uma grande distância entre as expectativas geradas pelo uso de tecnologia na educação e a realidade dos softwares educacionais disponíveis atualmente. Isto ocorre, em parte, pela concepção de softwares com *design* simplista das interfaces e de seus processos interativos. Uma interface que provenha dispositivos de interação adequados tem um efeito positivo na usabilidade do software, em sua aceitação, bem como no seu potencial em promover uma aprendizagem significativa. Por isso a concepção de software educacional e de sua interface deve estar alinhada a princípios pedagógicos adequados, atendendo desde requisitos como formato de apresentação de conteúdo e interação, até a quantidade de informação apresentada [6].

Estas considerações devem ser ainda mais enfatizadas quando se trata da concepção de softwares educacionais de cunho inclusivo, em que o processo de interação deve atender às necessidades de acesso à informação por parte de aprendizes com deficiência. No entanto, observa-se que são raros os softwares que contemplem requisitos de acessibilidade, restringindo a utilização e atuação plena dos sujeitos na construção coletiva de conhecimento. Destarte, evidencia-se a importância do desenvolvimento de recursos digitais acessíveis que garantam a plena utilização pelas pessoas com deficiência.

A reflexão acerca dessa questão incitou o Centro de Tecnologias Assistivas (CTA) da universidade, através do Núcleo de Pesquisa e Desenvolvimento de Tecnologias Assistivas (NP&D) a desenvolver um software de cunho educacional acessível para pessoas com deficiência auditiva (DAs), denominado Eduativo. As pesquisas norteadoras do desenvolvimento do software compartilham da opinião de Santoro [7] de que uma educação adaptada, utilizando diferentes recursos comunicativos que contribuam para a socialização do aprendiz com deficiência auditiva, deverá ter repercussões favoráveis na sua aprendizagem. Além disso, parte-se da premissa de que um ambiente informacional digital que se propõem a ser inclusivo necessita contemplar valores conceituais referentes às tecnologias digitais e assistivas como à acessibilidade e usabilidade, sendo importante a participação de usuários que possam contribuir para melhoria nos sistemas, de acordo com suas próprias expectativas.

Diante do exposto, o presente artigo demonstra o escopo do Eduativo, o qual tem por intuito promover diferentes formas de interação a fim de verificar o quanto os elementos agregados ao software respeitam às particularidades dos aprendizes surdos e oportunizam o acesso à informação, agregando em si elementos de acessibilidade na estruturação da arquitetura de informação que o compõe, tais como sinalizadores em Língua de Sinais e leitores em *SignWriting* para o atendimento das necessidades informacionais de pessoas surdas.

Assim, a descrição detalhada dos princípios que regem o desenvolvimento do Eduativo é apresentada nas seções subsequentes conforme seguem: a seção 2 apresenta detalhes da arquitetura ISA, proposta para concepção de softwares inclusivos, destacando possibilidades de ferramentas que podem ser utilizadas para implementação das diferentes camadas da arquitetura; a seção 3 apresenta o protótipo desenvolvido, baseado em uma instanciação da arquitetura proposta; a seção 4 descreve os resultados obtidos com a execução da turma piloto para validação do ambiente e por fim as considerações finais são apresentadas na seção 5.

## 2. A ARQUITETURA ISA

Ao se planejar um esquema de arquitetura de informação, voltada ao meio digital, deve-se contemplar preceitos relacionados à acessibilidade, a usabilidade e as tecnologias assistivas ou digitais disponíveis, na qual todos estes recursos devem ser estruturados em conformidade com o desenho universal, a fim de se garantir um ambiente informacional digital inclusivo. Esses elementos e princípios associados, quando implementados e aplicados em ambientes informacionais permitem a estruturação de uma Arquitetura da Informação Digital Inclusiva [2].

A Arquitetura ISA (*Inclusive Software Architecture*) baseia-se nesta perspectiva e direciona-se não somente a dar suporte ao desenvolvimento do software proposto (Educativo), mas também em fornecer subsídios para que outros desenvolvedores possam fazer uso das características da arquitetura em si para a concepção de softwares em contextos distintos que contemplem a interação de pessoas com deficiência auditiva.

Formada a partir de um estilo arquitetural em camadas, a arquitetura ISA abrange a estruturação de seus componentes de acordo com as peculiaridades a eles inerentes, indo ao encontro de um dos principais princípios de projetos orientados a objetos. Assim os componentes (ou módulos) que a formam são agrupados de acordo com suas funcionalidades, formando camadas distintas que podem ser analisadas separadamente, facilitando assim a sua manutenção. Conforme demonstra a Figura 1, a Arquitetura ISA constitui-se, especificamente, de três camadas: (a) Camada de Interface, (b) Camada Pedagógica e (c) Camada de Armazenamento.

texto e utilizando-se de recursos visuais que favoreçam a intermediação do percurso do usuário ao longo de seu processo interativo. Para suprir essa necessidade comunicacional, um recurso essencial para a camada de interface é a utilização de vídeos em Libras, pois através deles os usuários compreendem as funcionalidades do software. Exemplos de aplicativos que podem ser utilizados neste sentido são *Player Rybená* [5]; *Falibras MT* [9]; *SignSim* [1]; *Dicionário LIBRAS* [4] e *SigNed* [8], pois permitem a criação e/ou reprodução de vídeos em Libras.

Outra característica de suma importância à camada de interface é forma com as informações são apresentadas. Segundo Filho [3], a estruturação das informações em projetos de interfaces devem agregar características específicas relacionadas a textos, imagens, *layout* de telas, vídeos, som e feedback. Para uma melhor compreensão da abrangência destas características ou variáveis, as mesmas são elencadas e explicitadas na Tabela 1.

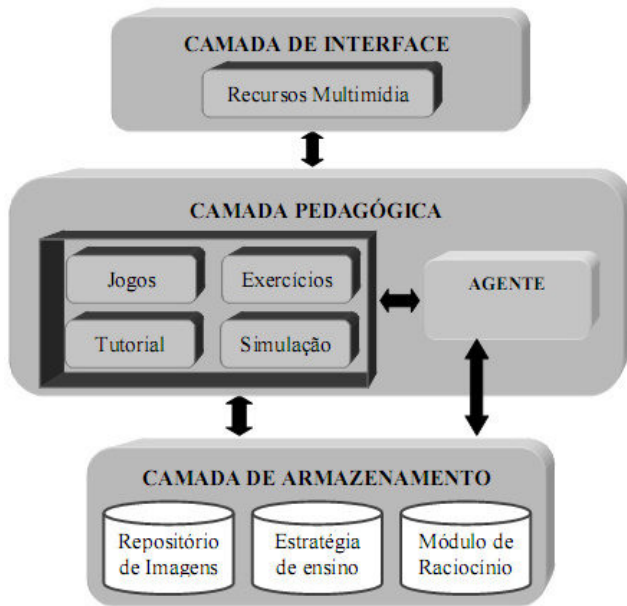


Figura 1. *Inclusive Software Architecture - ISA*

A **camada de interface** assume um papel crítico no contexto do desenvolvimento de softwares destinados às pessoas com deficiência auditiva, uma vez que o projeto de interface é de suma importância na determinação da aceitação do software pelo usuário. Por isso, esta camada deve atender às diretrizes de desenvolvimento de interfaces com recursos de acessibilidade e usabilidade adequados ao público-alvo ao qual se destina, ou seja, devem ser projetados elementos que viabilizem a comunicação do software com os usuários surdos, evitando o uso excessivo de

**Tabela 1. Variáveis a serem consideradas no projeto de interfaces**

|                       |  |
|-----------------------|--|
| <b>Textos</b>         | <ul style="list-style-type: none"> <li>• A utilização de textos na cor preta colocados sobre um fundo de cor clara permite uma leitura mais agradável do que apresentados sobre um fundo de cor mais escura ou sobre figura.</li> <li>• A utilização das palavras principais em Negrito resulta maior destaque para as palavras.</li> <li>• Os tipos de fonte podem ser utilizados as mais conhecidas, que são praticamente as mais usadas como Times New Roman, Arial e Geórgia.</li> <li>• Utilização da fonte grande e de preferência maiúscula tornando um maior destaque a palavra.</li> </ul>  |
| <b>Imagens</b>        | <ul style="list-style-type: none"> <li>• As imagens possuem um apelo visual muito forte, pois é através delas que podem motivar o usuário a executar ou ler determinada funcionalidade ou podendo ser ignorada.</li> <li>• É importante associar aos textos as imagens, assim quando ocorre à visualização da imagem com o texto ocorre uma associação melhor sobre o texto lido, deixando a leitura mais interessante.</li> </ul>   |
| <b>Layout de tela</b> | <ul style="list-style-type: none"> <li>• Esse aspecto é direcionado diretamente a aparência do software, devem ter cuidados especiais para não comprometer a usabilidade com o usuário, o tamanho, cor e a disposição dos elementos influenciam a facilidade que os usuários encontram esses elementos. Como elementos como botões, vídeos em tamanhos maiores são encontrados mais facilmente, cor deve ser agradável para que não resulte em um ambiente poluído e a disposição dos elementos é parte onde que prende a atenção do usuário, conteúdos mais importantes merecem destaque no centro do software.</li> <li>• Utilizar cores como vermelho para destacar atividades que merecem mais atenção, como perigo.</li> <li>• Na utilização de cores é preciso conhecer a faixa etária que utilizará o software, crianças preferem cores mais vivas, coloridas, já os adultos preferem cores mais neutras, discretas, através dessa característica que pode ser definida a aceitabilidade do sistema.</li> <li>• Sobre a visibilidade do software apenas as coisas necessárias devem estar visíveis, para não confundir o usuário com informações desnecessárias.</li> </ul> |
| <b>Vídeos</b>         | <ul style="list-style-type: none"> <li>• Os vídeos devem ser todos em Libras, sendo acompanhado por uma legenda em Língua Portuguesa, para que usuários que não são deficientes auditivos consigam entender.</li> </ul>  |

|                 |  |
|-----------------|--|
|                 | <ul style="list-style-type: none"> <li>• Devem estar disponibilizado opções de Usabilidade, como “Play”, ”Pause”, ”Stop” deixando a escolha do usuário sobre o que realmente deseja fazer no momento, tendo uma liberdade maior sobre o software.</li> </ul>         |
| <b>Som</b>      | <ul style="list-style-type: none"> <li>• Deve ser disponibilizado no software como uma escolha de configuração, considerando que podem existir usuários com surdez parcial e cuja presença de som é importante nas suas atividades.</li> </ul>                       |
| <b>Feedback</b> | <ul style="list-style-type: none"> <li>• É importante a cada ação que o usuário executar que seja retornado o resultado obtido, motivando o usuário a continuar executando a atividade com o objetivo de conseguir resultados melhores que os anteriores.</li> </ul> |

A **camada pedagógica** relacionada diretamente ao processo do ensino e de aprendizagem do usuário. Para ser considerado pedagógico um software deve ser classificado em, pelo menos, uma das seguintes categorias: (a) exercícios, (b) tutorial, (c) simulação e modelagem e, (d) jogos [Bertoldi 2009].

A utilização de exercícios em softwares educacional faz-se necessária quando determinados conteúdos já são conhecidos pelo aprendiz, mas não são completamente dominados e proporcionam uma forma de assimilar melhor os conteúdos abordados enquanto que os tutoriais devem ser utilizados quando se pretende apresentar conceitos novos, quando da necessidade de novas habilidades por parte do usuário, quando se requer a aquisição de conceitos, princípios e/ou generalizações de um determinado conteúdo ou ainda existe a necessidade de se propor atividades para se verificar o quanto determinado conteúdo foi assimilado pelo educando.

A simulação e os jogos também favorecem significativamente o processo de ensino e de aprendizagem. A simulação é a representação de um objeto real, de um sistema ou evento. É um modelo simbólico e representativo da realidade que deve ser utilizado após a aprendizagem de conceitos e princípios básicos do tema em questão enquanto que os jogos, são considerados ferramentas poderosas no processo educativo ao fazerem uso do lúdico para explorar um determinado ramo do conhecimento, além favorecer no educando habilidades como destreza, associação de idéias e raciocínio lógico e indutivo, dentre outros

A aplicabilidade de mais que uma categoria no software garante certamente um maior aprendizado sobre o domínio do conhecimento. No entanto, é importante conhecer o perfil do usuário e suas preferências a fim de selecionar a(s) categoria(s) que melhor atendam os objetivos pretendidos. Contudo, independente da categoria pedagógica escolhida para ser implementada no software é necessário que recursos adicionais sejam incorporados para que o processo de ensino e aprendizagem tenha sucesso.

Na arquitetura ISA, utilizou-se o submódulo denominado Agente, para auxiliar o usuário da melhor maneira possível durante o processo de aprendizagem. A função do módulo Agente na camada pedagógica é monitorar as ações que o usuário executa no ambiente e verificar se a ação é adequada ou não. Caso for detectada uma ação inválida por parte do usuário, o agente irá interferir no processo executando uma ação predefinida e devidamente armazenada. Os agentes podem ser criados através

da utilização de linguagens específicas para este propósito (linguagens orientadas a agentes) ou linguagens de propósito geral [3]. A implementação do seu comportamento está intimamente relacionado com a definição da estratégia de ensino adotada para o desenvolvimento do ambiente e/ou software, já que cada estratégia utiliza uma tática de ensino diferente.

A utilização de agentes em um ambiente digital de aprendizagem possui algumas vantagens dentre as quais citam-se: (i) O agente se preocupa com o progresso e a interação do usuário, passa a ele a impressão de que "estão juntos" no processo de ensino-aprendizagem, encorajando assim o usuário (aluno) a se preocupar mais com seu próprio progresso; (ii) O agente pode ter a autonomia de entrar em contato com o aluno quando ele fica frustrado, recuperando o interesse do aluno antes que ele volte sua atenção para outras coisas. (iii) O agente pode mostrar entusiasmo e relação ao assunto que esta sendo tratado, incentivando o aluno a ter o mesmo ânimo na realização das tarefas e, (iv) Também a escolha de diversidade dos comportamentos podendo ser verbais, textuais, vídeos podendo ser escolhido qual é o momento certo para demonstração do mesmo e entre outras vantagens a mais que a utilização de agentes proporciona ao usuário no processo de aprendizagem. Assim, o submódulo Agente proposto serve para perceber alguma ação executada pelo usuário e realizar uma verificação no módulo raciocínio para verificar a existência de alguma regra definida para aquela ação.

Por fim, a camada de armazenamento é responsável pela manipulação da informação de forma persistente, armazenando, acessando e recuperando qualquer dado que seja utilizado no software. Essa camada deverá armazenar, dentre outros conteúdos, as regras utilizadas pela camada pedagógica, mais especificamente pelo módulo Agente. Para verificar tais regras vários são os motores de inferência que podem ser utilizados, dentre os quais citam-se CLIPS, Ral/C++, NéOpus, JESS, JEOPS e Expert Sinta, sendo que cada um apresenta diferentes estratégias de interpretação das regras.

### 3. PROTÓTIPO DO SOFTWARE EDUTIVO

A arquitetura proposta apresentou uma variedade de componentes que podem ser instanciados no momento do desenvolvimento de um software. No caso do Edutivo, a arquitetura ISA foi instanciada da seguinte forma:

- **Camada de interface:** Para prover os recursos de acessibilidade pertinentes ao público-alvo para o qual se destina o Edutivo, optou-se pelo uso do módulo Player Rybená para disponibilização de vídeos em Libras e legenda da Língua Portuguesa. Para possibilitar a execução dos vídeos utilizou-se o módulo API JMF (*Java Media Framework*). Além disso, diferentes formas de apresentação das funcionalidades foram disponibilizadas no software sendo, para isto, utilizado módulos para representação de botões em Libras e botões *SignWriting*.

- **Camada pedagógica:** Foram instanciados os módulos exercícios e jogos. O primeiro é representado por 16 exercícios divididos em 2 níveis, envolvendo as operações matemáticas básicas. Já, o segundo, é composto por 3 jogos. Em ambos são utilizados os recursos da camada de interface para sua representação. Os módulos jogos e exercícios são monitorados pelo módulo Agente que tem como função monitorar as ações do usuário dentro do ambiente e executar determinadas tarefas. O Agente utilizado foi do tipo Pedagógico que servem para auxiliar

o usuário no processo de ensino e de aprendizagem. Além disso, optou-se pelo uso da estratégia de ensino por treinamento, por esta prover táticas que exploram imagens, exercícios e apresentação de problemas, pertinentes ao público-alvo em questão.

- **Camada de armazenamento:** Foi utilizado o banco de dados MySQL pelas suas vantagens como capacidade de manipular muitos registros, execução rápida de comandos, sistema de segurança simples e funcional e também por ser gratuito. No módulo de raciocínio do protótipo foi incorporado o motor de inferência JEOPS que utiliza regras de produção. A criação de tais regras ocorre pela separação explícita de todos os campos de uma regra: declarações de variáveis, condições e ações. Além disso, regras JEOPS são organizadas dentro de uma base de regras como métodos em Java dispostos dentro de classes. A Figura 2 apresenta exemplos de regras definidas para o Edutivo.

```

rule Regra1
{
  declarations
  Agente agente;

  conditions
  agente.getErrosTotal() >= 2;
  agente.getErros() == 1;
  agente.getStatus() == "exercicio";

  actions
  agente.setIndiceAgente(4);
  agente.configura(1);
}

rule Regra2
{
  declarations
  Agente agente;

  conditions
  agente.getStatus() == "jogo";
  agente.getAcertosTotal() == 9;

  actions
  agente.setIndiceAgente(-1);
}
    
```

Figura 2. Regras JEOPS definidas para o Edutivo

As regras permitem que o ambiente tenha um comportamento individualizado, reduzindo a necessidade da presença física de um professor para ensinar o usuário a realizar determinada tarefa, visto que o usuário será auxiliado pelo próprio ambiente, possibilitando a ele a realização de tarefas de maneira autônoma. Inicialmente, dez (10) regras foram definidas no software Edutivo. Entretanto, novas regras podem ser adicionadas para contemplar novas especificidades de interações com o usuário final.

A instanciação de todos os componentes ou módulos acima expostos foram contemplados na arquitetura ISA e forma a base de toda a implementação do Edutivo, conforme demonstra a Figura 3.

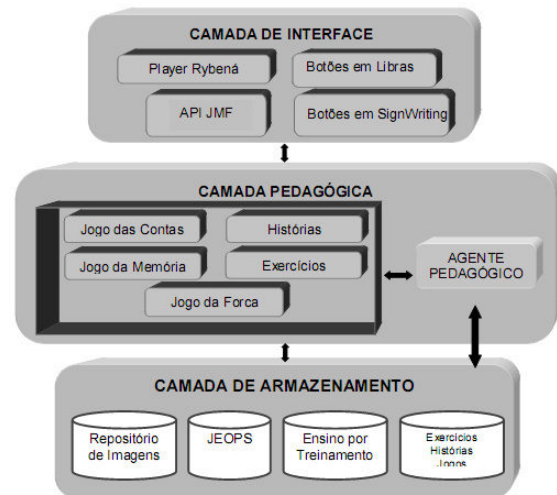


Figura 3. Arquitetura ISA e sua instanciação no Edutivo

Seguindo a arquitetura acima, o software foi desenvolvido na linguagem de programação Java, agregando suas características

específicas tais como portabilidade, facilidade de programação, alta capacidade de manutenção e reutilização de código, utilizando-se para tanto o ambiente de programação *NetBeans IDE 6.1*. Algumas interfaces implementadas, no ambiente, para a execução de atividades propostas aos educandos podem ser visualizadas nas Figuras 4 e 5.



Figura 4. Interface de exercícios no EduTivo

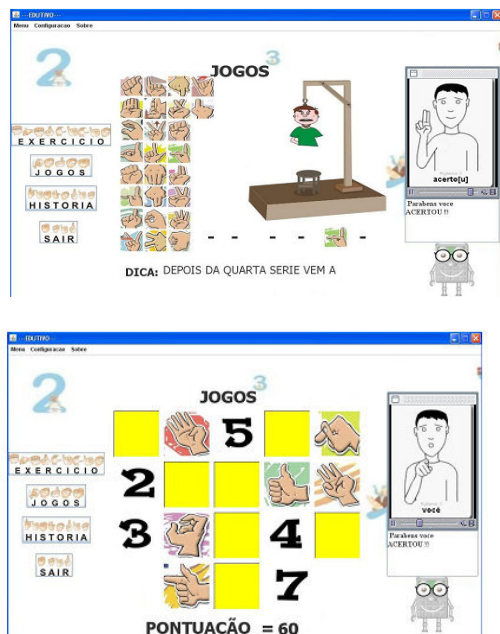


Figura 5. Interface de jogos disponibilizados no EduTivo

Conforme demonstrado, nas figuras acima, as interfaces concebidas contemplam os recursos de acessibilidade propostos na arquitetura, a partir do uso de botões utilizando símbolos em

Libras, vídeo em Libras e também um avatar representado pelo agente. Outro recurso de acessibilidade disponibilizada no EduTivo é utilização do sistema SignWriting na representação do menus do software, conforme demonstra a Figura 6.



Figura 6. Sistema SignWriting utilizado na representação dos menus no EduTivo

Além disso, estão contemplados no protótipo princípios de usabilidade como:

- (a) Visibilidade do estado do sistema: Os usuários são mantidos informados sobre o progresso do sistema com *feedback* apropriado durante realização de suas tarefas;
- (b) Compatibilidade do sistema com o mundo real: Toda informação apresentada pelo software utiliza os recursos de acessibilidade adequados ao público-alvo, utilizando a Língua Brasileira de Sinais (Libras) que é sua linguagem familiar que possuem um maior domínio, a representação em Libras é em imagens e em vídeos;
- (c) Controle do usuário e liberdade: A interface disponibiliza formas do usuário escolher o que realmente deseja executar, provendo possibilidades de desfazer ações indesejadas;
- (d) Consistência e padrões: Os botões disponíveis são padronizados tendo a mesma ação e significado em todas as interfaces onde aparecem;
- (e) Design estético e minimalista: Todas as informações apresentadas para o usuário são importantes e necessárias;
- (f) Ajuda e documentação: É disponibilizada uma central de ajuda, com algumas perguntas e respostas em Libras pré-definidas.

A fim de avaliar o quanto os recursos de acessibilidade e heurísticas de usabilidade propostos no EduTivo contribuem efetivamente para facilitar a interação do usuário e aquisição do conhecimento, foram realizados testes em uma turma piloto. Os resultados da validação realização são apresentados na seção a seguir.

#### 4. VALIDAÇÃO DO SOFTWARE

Para ser possível validar os recursos propostos pelo EduTivo, o protótipo foi disponibilizado para por aprendizes com deficiência auditiva do 5º ano da Escola Esther Schroder no município de Santo Ângelo. A turma é composta por 4 aprendizes, dois deles com surdez profunda e dois com surdez severa, todos bilaterais e com a faixa etária de 10 a 18 anos, todos alfabetizados na Língua Portuguesa e na Língua de Sinais (Libras). O protótipo foi utilizado em 4 aulas, totalizando um tempo de uso de 8 horas. Para a avaliação do protótipo foram elaborados questionários para serem aplicados aos aprendizes e à educadora, buscando averiguar a satisfação dos aprendizes no uso do software e também das estratégias pedagógicas propostas no EduTivo. A Figura 7 mostra os aprendizes utilizando o protótipo.



**Figura 7. Testes de usabilidade com o usuário**

A partir da avaliação realizada pode-se perceber que os recursos de acessibilidade utilizados estavam adequados, visto que todos os aprendizes utilizaram com facilidade o software. No entanto, constatou-se que o uso do sistema *SignWriting* não é necessário, pois poucas são as pessoas com deficiência auditiva que têm conhecimento deste sistema. Assim, somente a Língua dos Sinais (Libras) é suficiente para representar as funcionalidades do Edutivo.

Exceto esta consideração as demais avaliações se mostraram favoráveis quanto ao projeto do software. Por parte da educadora, houve as seguintes sugestões:

- (a) “Uso de sinais completos correspondentes a cada palavra e não soletrando como está representado”;
- (b) “Modificar a parte de visualização dos resultados, deixando como está a parte da escrita e acrescentando a mais um desenho representando o sinal de acerto e erro” e,
- (c) “Desenvolver exercícios de alguma palavra desconhecida, porque têm algumas palavras que não são conhecidas por eles e se não tiver alguma professora para explicar podem ficar perdidos. Um exemplo seria o exercício “Ache a soma do número 1.810 com seu sucessor”, a palavra “sucessor” é desconhecida para muitos, poderia ser interessante aplicar alguns exercícios como associar a essa palavra como, depois, ao sinal + mostrando alternativas de como representar a palavra”.

Tais solicitações sugerem pequenos ajustes, ainda necessários, no protótipo do Edutivo.

## 5. CONSIDERAÇÕES FINAIS

Este artigo apresentou a arquitetura ISA e sua utilização para implementação de softwares educacionais destinados a aprendizes com deficiência auditiva, enfatizando os requisitos de interação que devem estar contemplados nos softwares dedicados a este público.

Acredita-se que as pesquisas que subsidiaram a proposta do Edutivo promovam contribuições no sentido de minimizar as dificuldades de interatividade entre o usuário com deficiência auditiva e as tecnologias computacionais, resultando em um passo importante na inclusão digital e social de pessoas com deficiência no contexto escolar. Neste cenário, a arquitetura contribui para que outros softwares sejam desenvolvidos em diferentes áreas do conhecimento, agregando acessibilidade e usabilidade adequadas.

Decorrente da arquitetura proposta, o protótipo Edutivo fornece subsídios para que avaliações em contextos reais de uso sejam possíveis e permite que fatores essenciais para viabilizar a qualidade de acesso e aceitabilidade do software sejam analisados na prática. Além disso, favorece a possibilidade de ajustes no intuito de tornar tanto a arquitetura, quanto o software, ferramentas efetivas na produção de novas tecnologias.

## 6. REFERÊNCIAS

- [1] Borba, M.; Giraffa, L.M.M. SIGNSIM: Uma Ferramenta para Auxílio à Aprendizagem da Língua Brasileira de Sinais. Faculdade de Informática, PUCRS, 2007.
- [2] Corradi, J.A. Ambientes Informacionais Digitais e Usuários Surdos: Questão de Acessibilidade – Dissertação de Mestrado. Ciência da Computação, São Paulo, 2007.
- [3] Filho, T. A. G.; A Tecnologia Assistiva: de que se trata? In: MACHADO, G. J. C.; SOBRAL, M. N. (Orgs.). Conexões: educação, comunicação, inclusão e interculturalidade. 1 ed. Porto Alegre: Redes Editora, p. 207-235, 2009.
- [4] Libras. Dicionário da Língua Brasileira de Sinais. Disponível em <http://www.acessobrasil.org.br/libras>. Último acesso mar.2013.
- [5] Rybená Player – Disponível em [www.rybena.org.br](http://www.rybena.org.br). Último acesso em abril de 2013.
- [6] Santarosa, L.M.C; Conforto, D.; Passerino, L.M; et al. Tecnologias Digitais Acessíveis. JSM Comunicação Ltda: Porto Alegre, 2010.
- [7] Santoro, B. M.R Contando Histórias, programando o ensino: a literatura na pré-escola com alunos surdos. Campinas: PUCCAMP, 2006.
- [8] Signed Corradi, J.A.M. Ambientes Informacionais Digitais e Usuários Surdos: Questão de Acessibilidade, pág. 89, 2007.
- [9] Tavares, O. L; Coradine, L.; Breda W.L. Falibras MT – Autoria de tradutores automáticos de textos do português para Libras, na forma gestual animada: Uma abordagem com memória de tradução para a Comunicação. Dissertação de Mestrado. Universidade do Paraná. Curitiba, 2007.