



CENTRO UNIVERSITÁRIO UNIVATES
CENTRO DE CIÊNCIAS EXATAS E TECNOLÓGICAS
CURSO DE ENGENHARIA DA COMPUTAÇÃO

MAURICIO OSELAME POSTAL

**FERRAMENTA GRÁFICA PARA MODELAGEM DE
AMBIENTES DE SIMULAÇÃO DE PROCESSOS
INDUSTRIAIS**

Lajeado

2011

MAURICIO OSELAME POSTAL

FERRAMENTA GRÁFICA PARA MODELAGEM DE AMBIENTES DE SIMULAÇÃO DE PROCESSOS INDUSTRIAIS

Trabalho de Conclusão de Curso apresentado ao Centro de Ciências Exatas e Tecnológicas do Centro Universitário UNIVATES, como parte dos requisitos para a obtenção do título de bacharel em Engenharia da Computação.

Área de concentração: Simulação de Processos

ORIENTADOR: Ronaldo Hüsemann

Lajeado

2011

MAURICIO OSELAME POSTAL

FERRAMENTA GRÁFICA PARA MODELAGEM DE AMBIENTES DE SIMULAÇÃO DE PROCESSOS INDUSTRIAIS

Este trabalho foi julgado adequado para a obtenção do título de bacharel em Engenharia da Computação e aprovado em sua forma final pelo Orientador e pela Banca Examinadora.

Orientador: Prof. Ronaldo Husemann, UNIVATES
Mestre em Engenharia Elétrica pelo PPGE/UFRGS,
Porto Alegre, Brasil

Banca Examinadora:

Prof. Marcelo de Gomensoro Malheiros, UNIVATES
Mestre em Engenharia Elétrica pela UNICAMP, Campinas - Brasil

Prof. Fabrício Pretto, UNIVATES
Mestre em Ciência da Computação pela PUCRS, Porto Alegre - Brasil

Coordenador do Curso de Engenharia de Computação:
Prof. Marcelo de Gomensoro Malheiros

Lajeado, junho de 2011.

AGRADECIMENTOS

Aos professores que nos transmitiram os conhecimentos necessários para a realização deste trabalho ao longo do curso e também a todos os colegas que sempre estiveram presente neste período acadêmico.

Aos membros e funcionários do Centro Universitário UNIVATES por sempre proporcionarem um bom ambiente de aprendizado.

RESUMO

Este trabalho apresenta uma proposta de desenvolvimento de uma interface gráfica para um simulador educacional de processos industriais. A interface criada possibilita a inclusão de componentes, configuração de parâmetros de operação, estabelecimento de ligação entre eles, com recursos amigáveis como copiar, recortar, colar, desfazer e refazer, possibilitando assim uma montagem simplificada de diversos ambientes de simulação de processos industriais. É também permitido o aumento e a diminuição da velocidade da simulação e a pausa da mesma. Quando a simulação está ocorrendo é possível a obtenção dos resultados e a visualização do andamento, além de salvar um histórico de todas variáveis envolvidas. A proposta dedica-se a uma solução genérica, para diversos tipos de processos industriais.

Palavras-chave: Softwares Educacionais, Simuladores, Ferramentas Gráficas.

ABSTRACT

This paper presents a proposal of development of a graphical interface for an educational simulator of industrial process. The created interface allows the inclusion of components, configuration of operational parameters, connection between components and resources like cut, copy, paste, undo and redo allowing a simplified assembly of several simulation environments of industrial process. It is also allowed to increase and decrease the speed of the simulation and to pause it. When the application is running in simulation mode it is allowed the visualization of the progress and getting the results, it also saves a historic of all variables involved. The proposal is a generic solution, for several types of industrial processes.

Keywords: Educational Software, Simulator, Graphics Tools.

LISTA DE FIGURAS

Figura 1 Exemplo de simulador de processos industriais	13
Figura 2 Software simulador de processos industriais de petróleo	14
Figura 3 Interação entre aluno e computador pelo desenvolvimento de processador de texto	18
Figura 4 Interação entre aluno e computador pelo desenvolvimento de multimídia.....	19
Figura 5 Exemplo de simulador educacional	21
Figura 6 Exemplo de simulador educacional	22
Figura 7 Exemplo do software VisIt.....	24
Figura 8 Exemplo do software EAGLE	25
Figura 9 Diagrama dos componentes da simulação	28
Figura 10 Tela Principal	29
Figura 11 Tela de edição dos itens	30
Figura 12 Software executando no idioma inglês	32
Figura 13 Inserção de um item na tela.....	33
Figura 14 Menu de opções do item	34
Figura 15 Reposicionamento dos itens.....	35
Figura 16 Ligação entre os itens.....	36
Figura 17 Mensagem de erro ao tentar executar simulação	37
Figura 18 Alteração de propriedades com simulação em andamento	38
Figura 19 Simulação em andamento	41
Figura 20 Simulação com dois tanques em série.....	42
Figura 21 Simulação paralela em andamento.....	43

LISTA DE TABELAS

Tabela 1 Formatos de imagens suportados.....	31
--	----



LISTA DE ABREVIATURAS

ASCII: Advanced Simulation and Computer Initiative

BMP: BitMaP

CSV: Comma-Separated Values

DOE: Department Of Energy

EAGLE: Easily Applicable Graphical Layout Editor

GUI: Graphical User Interface

JPG: Joint Photographic Experts Group

PDF: Portable Document Format

PNG: Portable Network Graphics

PPM: Portable PixMap

TIFF: Tagged Image File Format

XBM: X11 BitMap

XPM: X11 PixMap

SUMÁRIO

1	INTRODUÇÃO.....	11
2	DEFINIÇÕES SOBRE SOFTWARES EDUCACIONAIS, SIMULADORES E FERRAMENTAS GRÁFICAS	16
2.1	Softwares Educacionais.....	16
2.1.1	Tutorial	17
2.1.2	Programação	17
2.1.3	Processador de Texto.....	17
2.1.4	Multimídia	18
2.1.5	Desenvolvimento de Multimídia	19
2.1.6	Simulação	19
2.1.7	Jogo educativo	20
2.2	Simuladores	20
2.3	Ferramentas Gráficas.....	23
3	FERRAMENTA DESENVOLVIDA.....	26
3.1	Definição dos Componentes.....	26
3.2	Desenvolvimento do Software.....	28
3.3	Integração e Funcionamento do Software	37
4	RESULTADOS EXPERIMENTAIS	40
5	CONCLUSÃO.....	44

1 INTRODUÇÃO

Em um ambiente acadêmico, são várias as dificuldades encontradas para disponibilizar materiais e recursos didáticos atualizados para os estudantes. Por exemplo, nos cursos mais voltados às áreas de Informática e Eletrônica são problemas comuns a desatualização periódica dos equipamentos devido às evoluções constantes das tecnologias, a necessidade de suporte técnico competente, além dos elevados custos de aquisição e manutenção dos equipamentos (MENESES; SILVA, 2009).

Já nos cursos das áreas química e metalúrgica encontram-se dificuldades adicionais devido à periculosidade advinda de alguns elementos utilizados, como o fogo e algumas substâncias corrosivas. Já para cursos voltados ao meio ambiente e instrumentação, existe a dificuldade de se realizar demonstrações em salas de aulas, como por exemplo, a demonstração de funcionamento de uma estação de tratamento de água e esgoto (DE OLIVEIRA, 2009).

Estas complicações têm gerado a necessidade de alternativas para a aprendizagem prática (DE OLIVEIRA, 2009).

Com isso surgiram diversas iniciativas de se utilizar recursos de multimídia em sala de aula (imagens, áudios e vídeos), como forma de facilitar a transmissão do conhecimento e despertar a atenção dos alunos, fazendo-os ver na prática a teoria estudada. Esses recursos são bastante utilizados em cursos à distância (MENDES; FIALHO, 2004).

Apesar de contribuírem para aproximar o aluno da realidade prática, entretanto, estes recursos apresentam uma limitação crítica, que é a incapacidade de uso destes para a realização de ensaios e demonstrações interativas, impedindo alunos de desenvolver algumas capacidades práticas esperadas para os seus cursos. Como alternativa mais rápida e objetiva para atender à necessidade de treinamento prático surgiram os softwares educacionais, que possibilitam uma maior interação com o aluno. Estes softwares auxiliam os alunos no desenvolvimento de atividades práticas, permitindo uma maior interação a partir do computador. Entretanto, na maioria dos casos, o computador tem controle acerca das alterações permitindo pouca intervenção por parte dos alunos (NETO, 2006).

Uma forma de software educacional é o simulador educacional, que permite a alteração do ambiente e das variáveis relacionadas a ele, deixando que o aluno escolha, desenvolva e implemente a situação que desejar. Segundo Assis e Tannous (2005) esta solução complementa o ensino, simplifica a obtenção de resultados complexos e auxilia na difusão do conhecimento de forma mais eficaz.

De forma geral, software de simulação é um aplicativo que possibilita a representação de um sistema real, imitando o comportamento do sistema por meio de um modelo computacional construído. Com base neste modelo, podem ser montados experimentos virtuais, aumentando a compreensão do comportamento do sistema alvo, facilitando a avaliação de estratégias para operá-lo, ao mesmo tempo em que reduz também os riscos e custos quando comparado com aplicações realizadas diretamente sobre um sistema real (VIEIRA, 2006).

Com o uso dessa tecnologia em sala de aula, o professor consegue diversificar suas aulas e, alterando as condições do ambiente simulado, possibilitar aos alunos os testes de diferentes hipóteses para a obtenção rápida dos resultados (CAMPOS et al. 2008).

Assim os estudantes podem, de maneira interativa, unir teoria e prática em diversas áreas, a partir de experimentos didáticos ou mesmo profissionais, podendo realizar testes de maneiras diferentes para resolver um mesmo problema, analisando as diversas possibilidades testadas, aprendendo e enriquecendo o seu conhecimento acerca do assunto estudado (PEZZOTTA, 2001).

Os ambientes de simulação variam de acordo com o nível de interação do aluno com o sistema. A simulação pode ser fechada, quando os alunos apenas especificam alguns parâmetros, alteram opções e após isso apenas assistem as simulações, ou então pode ser aberta, quando possibilita ao aluno a montagem da própria simulação e a interação na observação dos resultados obtidos (PRETO, 2002).

De acordo com Da Silva (2003), apesar das inúmeras vantagens apresentadas, o uso das simulações, em muitos casos, tem como desvantagem a dificuldade de interpretação dos resultados gerados, pois alunos, com pouco ou nenhum contato com experimentos reais, podem interpretar as ideias passadas de forma incompleta e imprecisa. Assim para complementar a simulação é de suma importância a utilização de ferramentas gráficas. Com estas ferramentas é facilitada a compreensão do usuário, já que possibilita a visualização dos elementos do sistema de forma didática e, muitas vezes interativa, o que permite um maior controle da simulação e ajuda no entendimento do comportamento do processo. Sem o uso de interfaces gráficas o usuário, muitas vezes, ficaria limitado a interpretar tabelas ou números (DUARTE et al, 2004).

Segundo Schroeder et al. (2005) interfaces gráficas auxiliam, através de gráficos ou animações, o projeto e a análise dos ambientes simulados e permitem ao usuário a definição dos parâmetros de controle, fornecendo a visualização das informações necessárias.

Diversas soluções de simuladores gráficos podem ser encontradas como iniciativas acadêmicas ou comerciais.

Por exemplo, a Figura 1 apresenta o trabalho de Duarte et al. (2004) que descreve um simulador de processos industriais, desenvolvido em ambiente de janela, utilizando linguagem Java.

O simulador apresentado possibilita ao usuário a entrada de alguns parâmetros de configuração, visualização e controle da simulação.

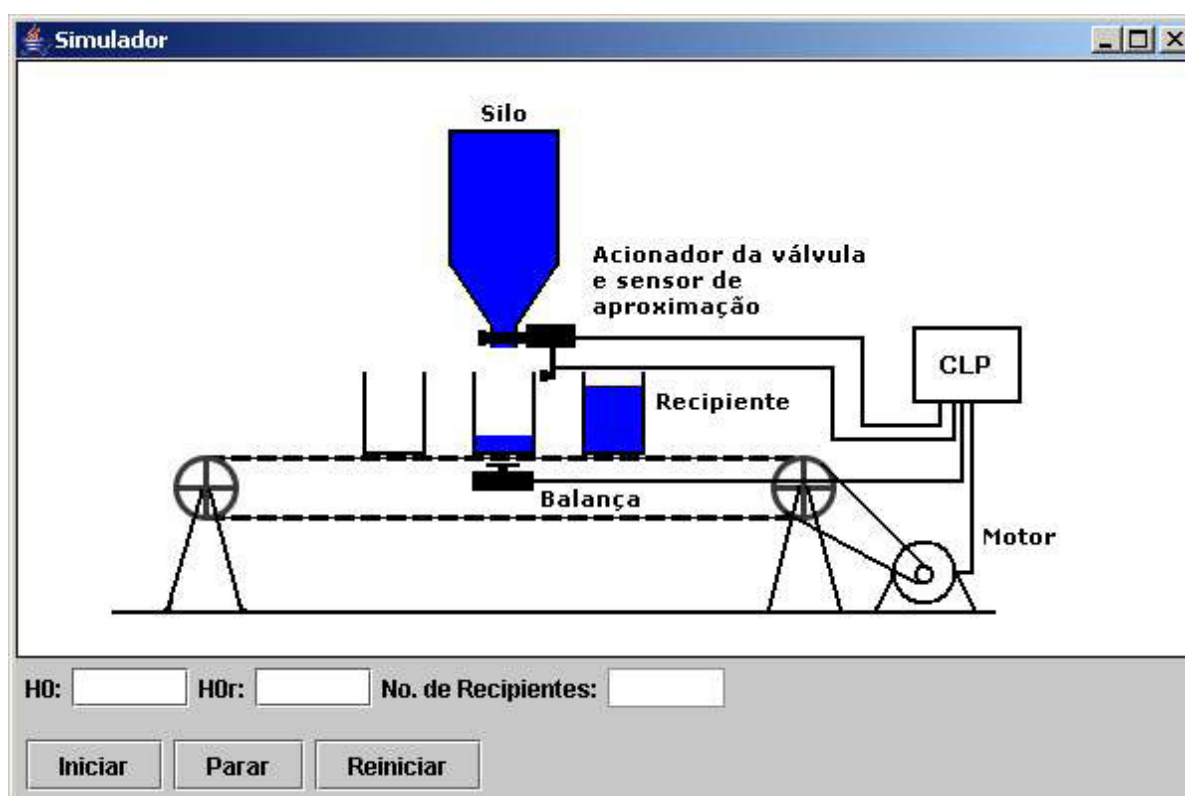


Figura 1 Exemplo de simulador de processos industriais

(DUARTE et al, 2004)

Outro exemplo de simulador gráfico de Lopes et al. (2006) que apresenta um software simulador de processos de petróleo.

A interface deste simulador (Figura 2) exibe bombas, sensores e tanques, assim como gráficos. No lado esquerdo do ambiente estão os parâmetros de controle do sistema, onde são atualizados simultaneamente gráficos de tendência (histórico de medidas de variáveis do processo).

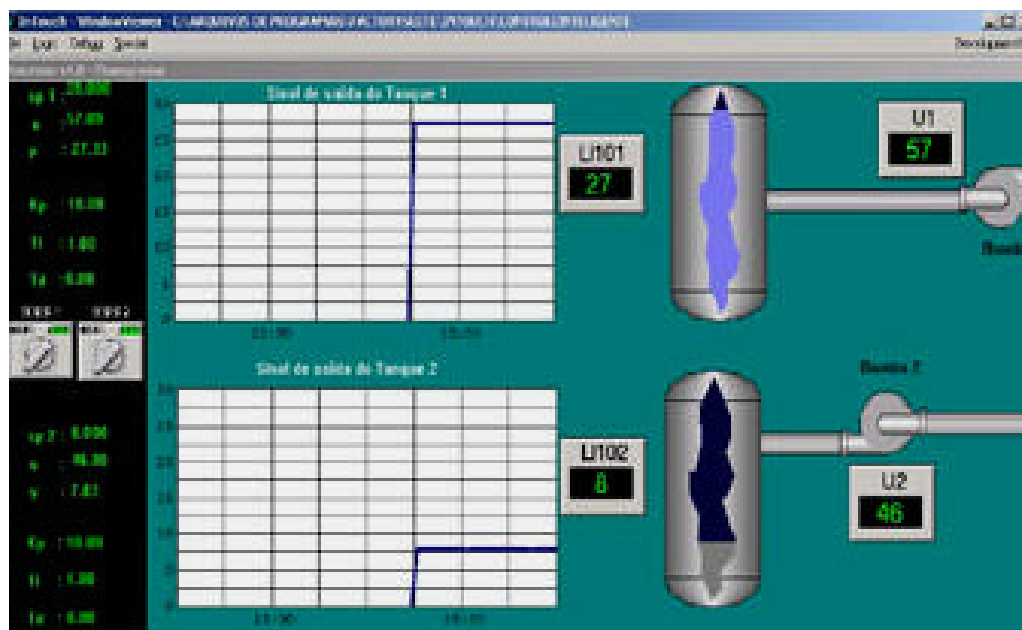


Figura 2 Software simulador de processos industriais de petróleo

(LOPES et al, 2006)

Os dois projetos citados, entretanto, são específicos para apenas um tipo de processo, apresentando pouca flexibilidade de mudança da topologia do ambiente de simulação. Com isso, estas soluções se tornam limitadas e podem não atender a necessidades de diferentes instituições de ensino.

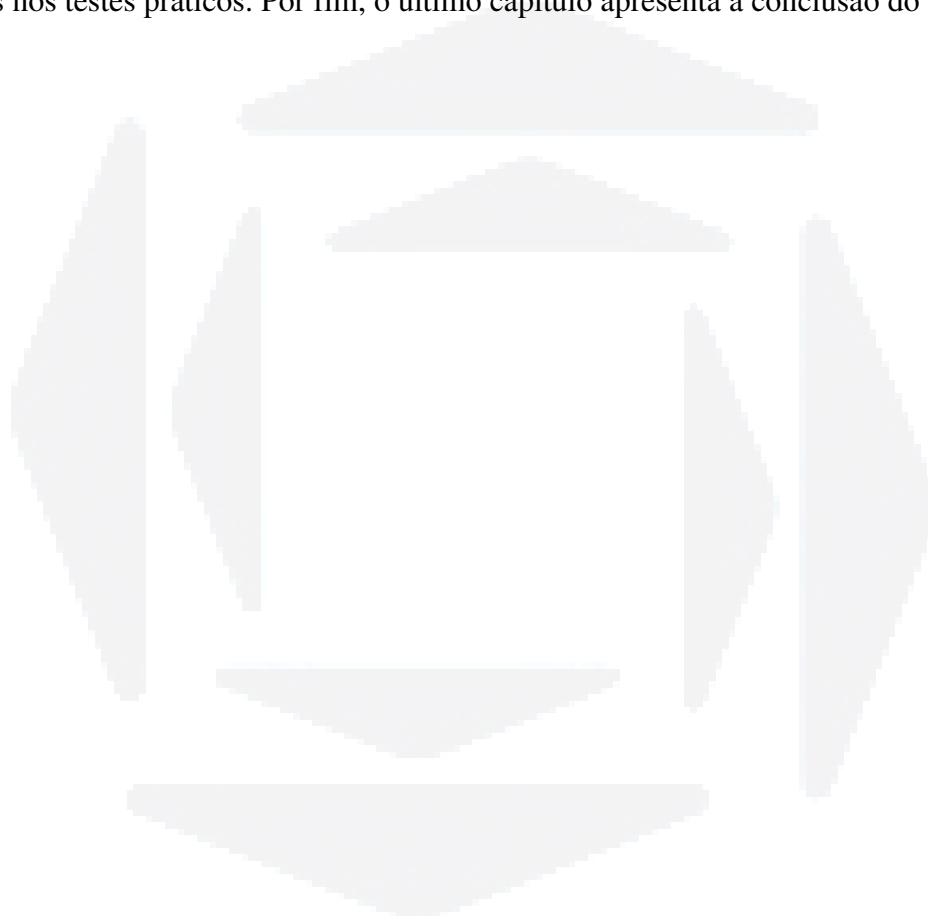
Em vista disso surgiu a motivação para a criação de uma ferramenta gráfica flexível para configuração e visualização de simuladores, e que atenda as necessidades acerca de vários tipos de processos industriais.

O software idealizado deve permitir a montagem de diversas topologias de plantas industriais em uma interface de fácil utilização, facilitando assim as tarefas de compreensão e visualização do comportamento destes processos para diferentes situações.

Para isso montou-se uma plataforma gráfica, com interação com o usuário e possibilidades de configuração, para uma planta industrial de líquidos. Essa plataforma gráfica foi programada na linguagem C++, utilizando a biblioteca gráfica QT. De acordo com Dalheimer (2002), a biblioteca QT não tem os problemas de eficiência encontrados em outras plataformas, como a SWING, do Java, pois é inteiramente programado em C++ e compilado para código nativo. Ou seja, os programas escritos em QT são mais rápidos do que outros, além de ser multiplataforma, podendo ser utilizado em Linux, Windows e Mac OS.

Cabe destacar que o QT é utilizado por alguns softwares, como por exemplo, Google Earth, Photoshop Elements, Mathematica e Skype, além de ser encontrado em aplicativos para telefones celulares e GPS (MINKE, 2009).

O próximo capítulo aborda questões teóricas para o desenvolvimento deste projeto, como os tipos e características de softwares educacionais, uso dos simuladores educacionais e utilização e importância do uso de ferramentas gráficas. O terceiro capítulo dedica-se a explicação do desenvolvimento do software, enquanto o quarto capítulo expõe os resultados obtidos nos testes práticos. Por fim, o último capítulo apresenta a conclusão do trabalho.



2 DEFINIÇÕES SOBRE SOFTWARES EDUCACIONAIS, SIMULADORES E FERRAMENTAS GRÁFICAS

2.1 Softwares Educacionais

Software educacional é um instrumento desenvolvido para atingir os objetivos educacionais estabelecidos previamente (DOS SANTOS; COSTA, 2005). É um software que possui uma proposta de ensino com um objetivo predefinido, propondo-se a auxiliar os alunos na aprendizagem de conteúdos e habilidades necessárias, utilizando uma interface computadorizada. O software deve conter situações que estimulem o aprendizado ao aluno (OLIVEIRA, et al, 2002).

Os softwares educacionais utilizados em sala de aula, segundo seus paradigmas, podem ser divididos em duas categorias: o paradigma instrucionista e o paradigma construcionista (HENDRES; KAIBER, 2005).

O paradigma instrucionista parte do princípio de que ensinar está relacionado com a transmissão da informação. Este paradigma se baseia pela postura passiva do aluno e sob esta visão considera que melhorar o ensino significa melhorar as técnicas de transmissão de informação. Neste contexto, o computador serve apenas como uma máquina de ensinar e o aluno apenas recebe as informações enviadas por ele (SANTANCHÊ; TEIXEIRA, 1999).

O paradigma construcionista, por sua vez, propõe ao aluno a construção personalizada do conhecimento dos alunos e não a transmissão do mesmo. O aluno opera o computador solicitando por comandos o que deve ser feito segundo sua visão individualizada (BURD, 1999).

De acordo com Hendres e Kaiber (2005) os softwares educacionais podem ser divididos em:

- a) Tutorial;
- b) Programação;
- c) Processador de texto;
- d) Multimídia;
- e) Desenvolvimento de multimídia;
- f) Simulação;
- g) Jogo Educativo.

A seguir se apresenta uma breve descrição de cada um destes tópicos.

2.1.1 Tutorial

Este tipo de software apresenta de maneira formal informações e conceitos, servindo de apoio ou reforço para as aulas, além de poder ser usada por alunos, no caso de perda de alguma aula ou para revisão da matéria estudada. A interação entre computador e aluno normalmente ocorre apenas pela leitura da tela ou pela escuta das informações fornecidas. Como recurso de avaliação de entendimento pode fazer perguntas ao aluno para testar a sua compreensão acerca do assunto estudado, caso a resposta seja correta, o programa prossegue para o próximo assunto, caso a resposta seja errada, o programa permite a repetição da lição passada (HENDRES; KAIBER, 2005).

Esta modalidade de software tem como principal limitação a dificuldade em entender e corrigir as questões desenvolvidas pelos alunos deixando isso a cargo de um professor (VALENTE, 1999).

2.1.2 Programação

Essa modalidade de software faz com que o aluno processe informações e as transforme em conhecimento, para assim conseguir passá-lo ao programa. O aluno tem que descrever a resolução do problema na linguagem do computador. A resposta fornecida pelo computador é fiel, ele não altera em nada o que foi informado pelo aluno, conseguindo assim reproduzir exatamente o que o aluno queria. Qualquer erro no resultado será produto do pensamento do aluno (VALENTE, 1999).

O programa representa a ideia do aluno pela relação entre os comandos inseridos e o comportamento da máquina, permitindo que o aluno consiga achar seus erros. No entanto isso não tira a responsabilidade do professor, que tem de compreender as ideias do aluno e intervir e auxiliar o aluno quando necessário.

2.1.3 Processador de Texto

É um software que foi criado com a finalidade de editar palavras e produzir textos, a interação entre o aluno e o computador é feita pelo idioma e pelos comandos do processador (alinhar o texto, grifar palavras), os processadores de texto possuem ferramentas ortográfica e gramática para fazer a verificação de palavras digitadas erroneamente e de concordância. Este, entretanto, não consegue fazer uma análise do texto e nem fazer verificações para saber se a ideia do aluno está coerente. É o professor quem tem que dar o parecer e corrigir o texto do aluno no que diz respeito ao significado, ideias e conteúdo. Após a intervenção, torna-se

mais fácil lidar com o computador do que com uma escrita com lápis e papel, pois o computador possibilita uma reescrita, modificações e correções a qualquer momento e quantas vezes forem necessárias (MENDES, 2006). Isso pode ser visto na Figura 3.

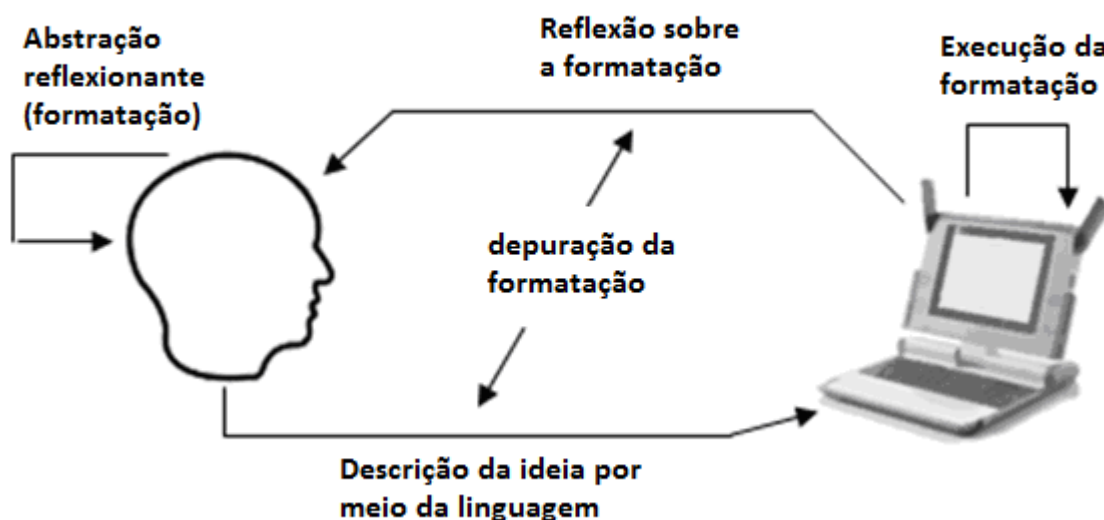


Figura 3 Interação entre aluno e computador pelo desenvolvimento de processador de texto
(CARDOSO; BARANAUSKAS, 2009)

2.1.4 Multimídia

Segundo Mendes (2006), o uso de multimídia, em sua essência, aproxima-se muito do uso de um software de tutorial, pois possibilita a escolha e utilização do que é oferecido pelo software, que tem algumas facilidades, como o uso de imagens, animações e sons que ajudam na expressão das ideias.

De acordo com Valente (1999), com o uso de softwares multimídia o aprendiz pode explorar um grande número de tópicos e navegar fundo neles, no entanto o aprendiz fica sempre preso ao que o software tem disponível, caso o software não possua a informação que o aluno deseja, ele deve buscar outro software.

Os softwares multimídia auxiliam a aquisição de informações, mas não a compreensão e construção do conhecimento, pois o aluno limita-se a escolher as opções oferecidas pelo software, sem descrever o que pensa. Após a escolha o software apresenta a informação para que o aluno reflita e, com base nessa informação, o aluno pode escolher outras opções.

2.1.5 Desenvolvimento de Multimídia

Este software permite a representação de informações por várias mídias. Isso permite ao aluno a busca pela informação, a crítica e a análise da informação apresentada. Esse sistema não registra os processos utilizados para a montagem do software, ele faz a ligação das informações, isso o torna parecido com o processador de texto, portanto, fica a cargo do professor a solicitação para que o aluno programe animações e outros efeitos. Não é necessário que o aluno descreva tudo o que ele está pensando quando ele seleciona uma determinada informação, nem explica conceitos, estratégias, etc. Isso torna o desenvolvimento de multimídia diferente da programação (VALENTE, 1999).

Na Figura 4 apresenta-se uma demonstração de Valente (1999) da interação entre o aluno e o computador no desenvolvimento de multimídia, onde o aluno inclui diferentes assuntos na multimídia e depois recebe os resultados obtidos e reflete acerca deles.

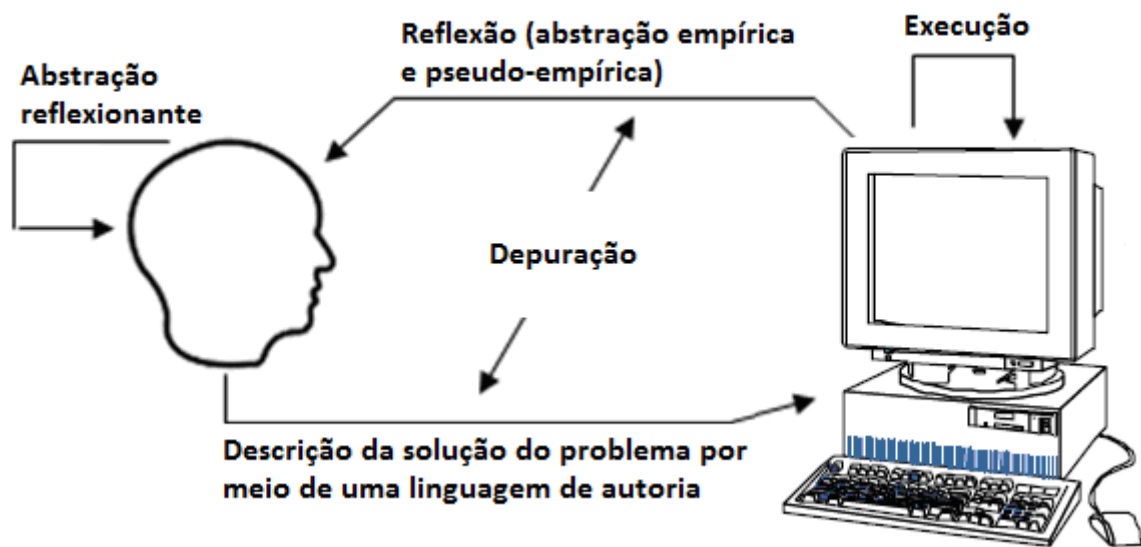


Figura 4 Interação entre aluno e computador pelo desenvolvimento de multimídia

(Baseado em VALENTE, 1999)

2.1.6 Simulação

O software de simulação possibilita ao usuário o desenvolvimento de hipóteses, testes e análise de resultados, conseguindo construir conceitos hipotéticos. Podem ser utilizados tanto como textos, quanto animações, sons e gráficos. Nesta modalidade de software, o aluno apenas altera alguns parâmetros e após isso observa o comportamento do processo, podendo

depois alterar novamente os parâmetros, o que permite assim a análise de várias formas de comportamento do processo escolhido (HENDRES; KAIBER, 2005).

A simulação pode ser fechada, ficando assim semelhante a um tutorial (o aluno apenas entra com alguns parâmetros e depois assiste a simulação feita pelo computador). A simulação pode ainda ser aberta onde o aluno descreve e implementa os aspectos do processo na própria simulação. Isso possibilita ao aluno o envolvimento com o processo, observando as variáveis atuam nele e suas influências no seu comportamento. O aluno cria hipóteses e ideias e, através da simulação, tenta validá-las (VALENTE, 1999).

2.1.7 Jogo educativo

O jogo educativo exige estratégias e habilidades do aprendiz para que ele consiga alcançar os objetivos. Conceitos e informações não são muito utilizados, tornando necessário que o aluno tenha atenção e criatividade (HENDRES; KAIBER, 2005).

Nos jogos o aluno aprende a projetar as consequências a longo prazo, também auxiliam a ver o processo como um todo e não apenas em partes. A função dos jogos é fornecer informações claras para o aluno acerca das regras e objetivos, deixar que o aluno controle a interação, assim como a continuação do desenvolvimento do jogo e possibilitar a repetição de procedimentos, além de usar efeitos auditivos e visuais. Além disso, devem fornecer informações que esclareçam o motivo da atividade, as relações entre as ações e suas consequências (KAMMER, 1999).

Os jogos educacionais assemelham-se as simulações fechadas, pois as regras do jogo já são previamente definidas e o aluno apenas deve elaborar hipóteses utilizando estratégias e conhecimentos já existentes (VALENTE, 1999). Os jogos criam condições para o aluno colocar em prática as suas estratégias. No entanto ele pode usar elas correta ou erroneamente sem saber disso, torna-se importante a intervenção do professor, documentando, discutindo e recriando as decisões dos alunos.

2.2 Simuladores

Conforme já comentado, os softwares de simulação reproduzem virtualmente uma situação real, permitindo a verificação dos efeitos causados pelos parâmetros inseridos pelos estudantes, sem a necessidade da montagem do experimento real (GUILLERMO et al. 2005).

As simulações tornam-se especialmente úteis para o caso de experiências reais são perigosas ou de difícil reprodução, como por exemplo, a manipulação de substâncias

químicas, ou quando os experimentos têm um custo muito elevado ou que precisam de muito tempo para ocorrer, como o crescimento de plantas. Também auxiliam para obtenção do conhecimento acerca de situações impossíveis de serem obtidas, como, por exemplo, desastres ecológicos (ASSIS, TANNOUS, 2004).

De acordo com Burd (1999), os simuladores auxiliam os alunos na visualização e compreensão do fenômeno estudado, além de incentivarem algumas práticas importantes, como a exigência de que o aluno cumpra alguns prazos para poder chegar ao resultado desejado.

Um exemplo de simulador é o LogicSim, software desenvolvido na linguagem Java pelo estudante Andreas Tetzl. É utilizado para melhorar a compreensão dos estudantes acerca das combinações lógicas e aritméticas presentes nos computadores (ASHRAF, 2004).

Na Figura 5, apresenta-se uma tela capturada do software LogicSim instalado em um computador pessoal (ELEKTOR, 2008). Nesta aplicação foi montado um pequeno circuito. O software possui uma barra lateral, onde se encontra a biblioteca de portas lógicas, bastando ao usuário clicar na porta desejada para selecioná-la e depois clicar na área central para colocar a porta no local desejado.

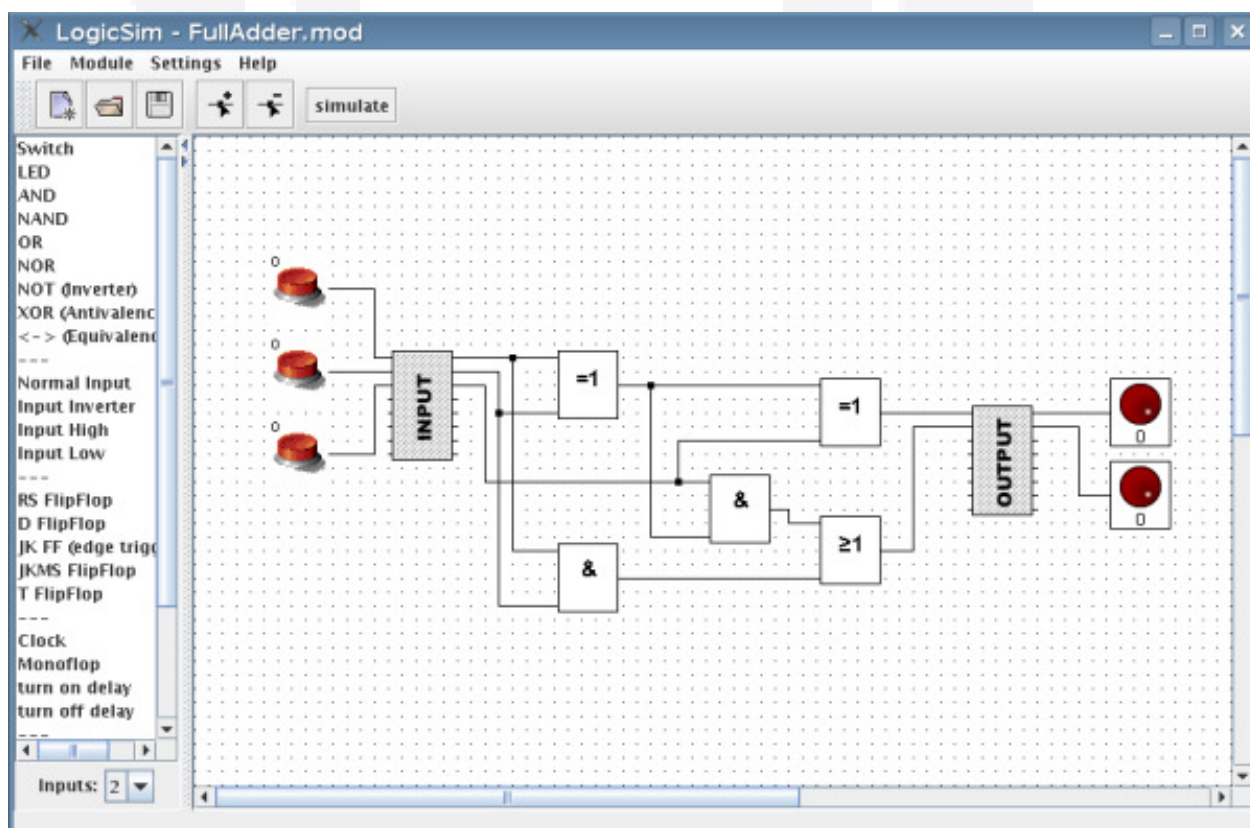


Figura 5 Exemplo de simulador educacional

(ELEKTOR, 2008)

Outro exemplo de software simulador para processos industriais é o SPS-VISU, desenvolvido pela empresa MHJ (MHJ-SOFTWARE, 2009) ele permite a inserção de elementos industriais como cilindros, esteiras, textos, displays para exibição de mensagens, entre outros. Possui uma versão gratuita de demonstração onde nem todos os tipos de simulações são suportados e uma versão paga, com suporte a todas as simulações. Ainda possui diversos exemplos, permitindo ao usuário a utilização e a alteração destes exemplos, além de possibilitar a criação de novos projetos, conforme a necessidade do usuário.

Na Figura 6 encontra-se uma tela do ambiente do SPS-VISU. Na parte direita encontra-se uma pequena tela para configuração da vazão de entrada e da vazão de saída do tanque. Na janela principal tem um tanque onde mostra a simulação de entrada e saída da água, de acordo com as configurações do usuário. Ele também possui uma barra de ferramentas com os possíveis objetos a serem inseridos, e outra com as opções de simulação.

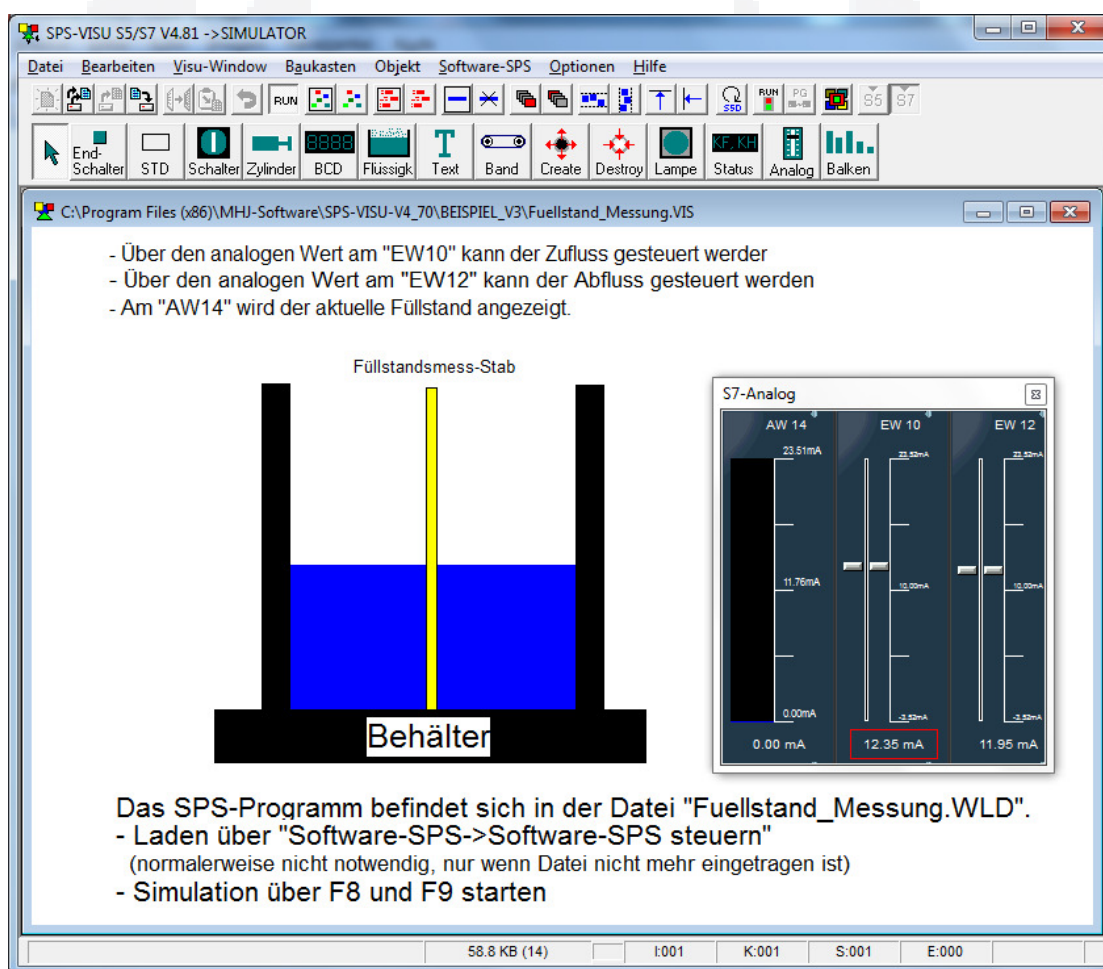


Figura 6 Exemplo de simulador educacional
(SPS-VISU, empresa MHJ-software)

2.3 Ferramentas Gráficas

A utilização das ferramentas gráficas permite a visualização dos resultados, facilitando assim a compreensão do problema por parte do usuário. Utilizando-se este recurso em sala de aula aumenta o entendimento do aluno acerca do processo em questão (MUNIZ, MARCZAC, 2001).

As ferramentas gráficas escondem detalhes da implementação do usuário, facilitando a organização das informações e também permitindo estruturas com tamanho ilimitado, entretanto é importante que o usuário tome cuidado para não criar nada com tamanho muito grande, para não sobrecarregar o software desenvolvido. Uma importante função é a de fornecer uma interface gráfica amigável para auxiliar o usuário a visualizar tudo o que está acontecendo e facilitar a sua compreensão e manipulação dos dados desejados (MUCHALUAT et al, 1996).

Existem várias ferramentas gráficas no mercado, um exemplo é o VisIt. É uma ferramenta gratuita para análises gráficas e visualizações de dados científicos. Permite a visualização de campos escalares e vetoriais, em duas ou três dimensões, ele foi desenvolvido dentro do programa ASCI (*Advanced Simulation and Computer Initiative*) do DOE (*Department Of Energy*) para visualizar e analisar os resultados de simulações (Department Of Energy, 2005). A interface gráfica com o usuário consiste em uma ou mais janelas de simulação, que mostram o andamento da simulação, e também possui uma janela de controle, que permite a inserção, remoção e edição dos itens da janela de simulação, além de informar as áreas que estão ativas e os componentes que estão presentes na simulação.

Na Figura 7 apresenta-se uma imagem do VisIt. Percebe-se do lado esquerdo a janela de controle conforme citado anteriormente, apresentando os arquivos selecionados, permitindo o início e a pausa da simulação, aumento e diminuição da velocidade, e ao lado direito a janela de visualização.

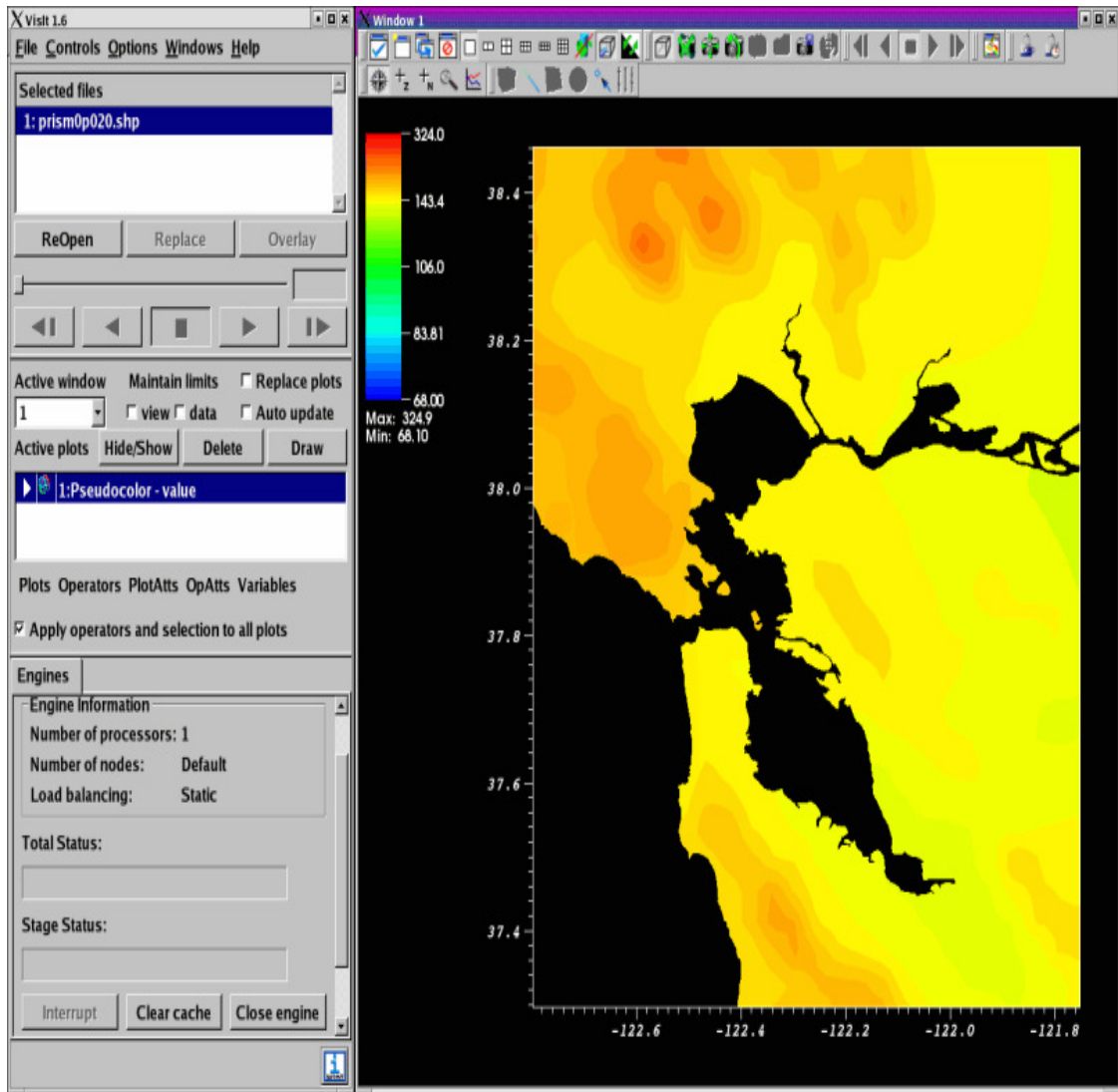


Figura 7 Exemplo do software VisIt

(DOE, 2005)

Outro software de exemplo é o EAGLE (*Easily Applicable Graphical Layout Editor*), da empresa CadSoft Computer. Conforme CadSoft (2010), o EAGLE é um software multiplataforma, para desenho de esquemáticos e placas. O usuário inicia as ações desejadas clicando nos itens de menus, na barra de ferramentas ou por comandos escritos, tanto na área reservada para isso quanto num arquivo externo que pode ser importado.

Na Figura 8 encontra-se uma imagem de uma placa montada com o EAGLE (CADSOFT, 2010).

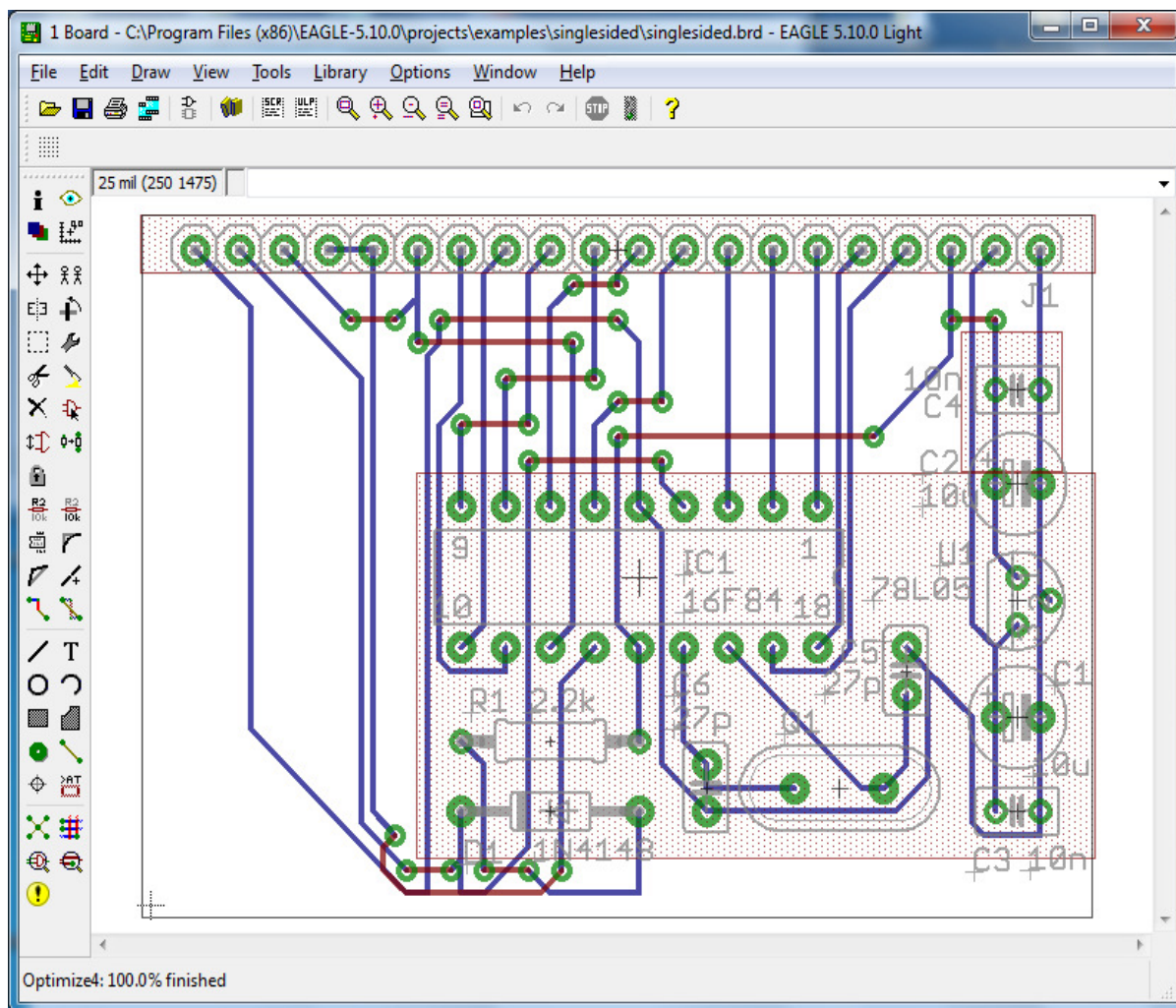


Figura 8 Exemplo do software EAGLE

(CADSOFT, 2010)

3 FERRAMENTA DESENVOLVIDA

Foi desenvolvida uma interface gráfica para montagem de uma simulação de planta industrial de líquidos. Para tanto foi projetado um ambiente para a adição de tanques, canos, válvulas e bombas de água, possibilitando ao usuário as funcionalidades de inserção e ajuste dos itens e a conexão entre eles na tela de acordo com o desejado. Para facilitar a identificação dos itens presentes na tela, o software também possibilita a adição de itens de texto, com recursos gráficos configuráveis (alteração do texto, do tamanho e da família da fonte e colocação em negrito, itálico e sublinhado).

Com o intuito de melhorar a visualização e compreensão do que ocorre durante a simulação, permite-se colocar na tela itens medidores, que são utilizados para a obtenção de informações a respeito de alguns parâmetros do item que está conectado com ele enquanto a simulação está em andamento. Por exemplo, caso ele seja conectado a um item bomba, ele permite ao usuário a escolha de leitura de vazão ou da pressão da bomba.

3.1 Definição dos Componentes

De forma resumida, pode-se dizer que, para o caso de um sistema de líquidos, a simulação deve conter entradas e saídas de nível e vazão que são alteradas dinamicamente.

Mais especificamente dentro da aplicação utilizada para validação da ferramenta foi projetada aplicação de tanques, onde uma bomba de água será utilizada para fornecer a entrada do sistema, tendo canos para transportar a água, válvulas para permitir ou impedir sua passagem e tanques que seriam os recipientes de armazenamento do líquido.

Cada um destes itens tem algumas variáveis para identificá-los:

- a) Id: identificador único dos objetos;
- b) Categoria: define a modalidade de variável física, por exemplo: vazão;
- c) Tipo físico: define o modo da variável física (analógica ou digital);
- d) Máximo_Conexões_Entrada: define o número máximo de conexões que pode possuir em sua entrada;
- e) Máximo_Conexões_Saída: define o número máximo de conexões que pode possuir em sua saída;
- f) Mínimo_Conexões_Entrada: define o número mínimo de conexões que deve possuir em sua entrada para a simulação acontecer;
- g) Mínimo_Conexões_Saída: define o número mínimo de conexões que deve possuir em sua saída para a simulação acontecer;

- h) Lista_Conexões_Entrada: armazena os itens conectados na sua entrada;
- i) Lista_Conexões_Saída: armazena os itens conectados na sua saída.

O software foi projetado de forma genérica para facilitar a inserção de novos recursos e novos componentes. Foi criada uma classe abstrata para os componentes, chamada de *component*, que possui funções utilizadas por todos os componentes da interface, como o Id, altura e largura, essa classe é estendida pela classe *meter*, que contém todos os dados acerca dos medidores, parâmetro exibido, componente a qual está conectada, também é estendida pela classe *text*, que contém o texto e informações acerca da fonte e por fim é estendida pela classe *simComponent*, responsável pelas classes e variáveis presentes em todos os componentes de simulação, contendo o número máximo de conexões de entrada, número máximo de conexões de saída, informações sobre rotação, tipo físico, categoria, entre outras. Essa classe é estendida pela classe *componentFlow* que possui funções utilizadas pelos componentes de vazão, como vazão de entrada, vazão de saída, vazão máxima de entrada e vazão máxima de saída. Para cada tipo de componente também foram criadas classes com funções específicas, que estendem a classe *componentFlow*.

Para tanques (*componentTank*), funções de volume, capacidade, que é estendida pela classe *roundedTank*, para tanques cilíndricos, com função de diâmetro, que, por sua vez, é estendida pela classe *roundedTankOneInputOneOutput*, que possui funções para permitir apenas uma entrada e uma saída.

Para canos (*componentPipe*), funções de rugosidade, velocidade de escoamento do líquido, comprimento e pressão, que é estendida pela *roudedPipe*, com funções de diâmetro, que é estendida pela classe *roundedPipeOneInputOneOutput*, que possibilita apenas uma entrada e uma saída no cano.

Para bombas (*componentPump*) com funções de pressão, que é estendida pela classe *pumpOneOutput*, que possui funções para possibilitar apenas uma saída e nenhuma entrada.

Para válvulas (*componentValve*), com funções de condutância. Essa classe é estendida por *linearValve*, para válvulas lineares que possui funções para controlar a abertura, por *quickValve*, para válvulas de abertura rápida que possui as mesmas funções da *linearValve*, por *valveOnOff*, para válvulas abre/fecha, que contém funções para abrir ou fechar a válvula e por *percentageValve*, para válvulas percentuais com funções para alterar a porcentagem de abertura da válvula e o parâmetro do projeto.

Um diagrama com essas classes pode ser observado na Figura 9.

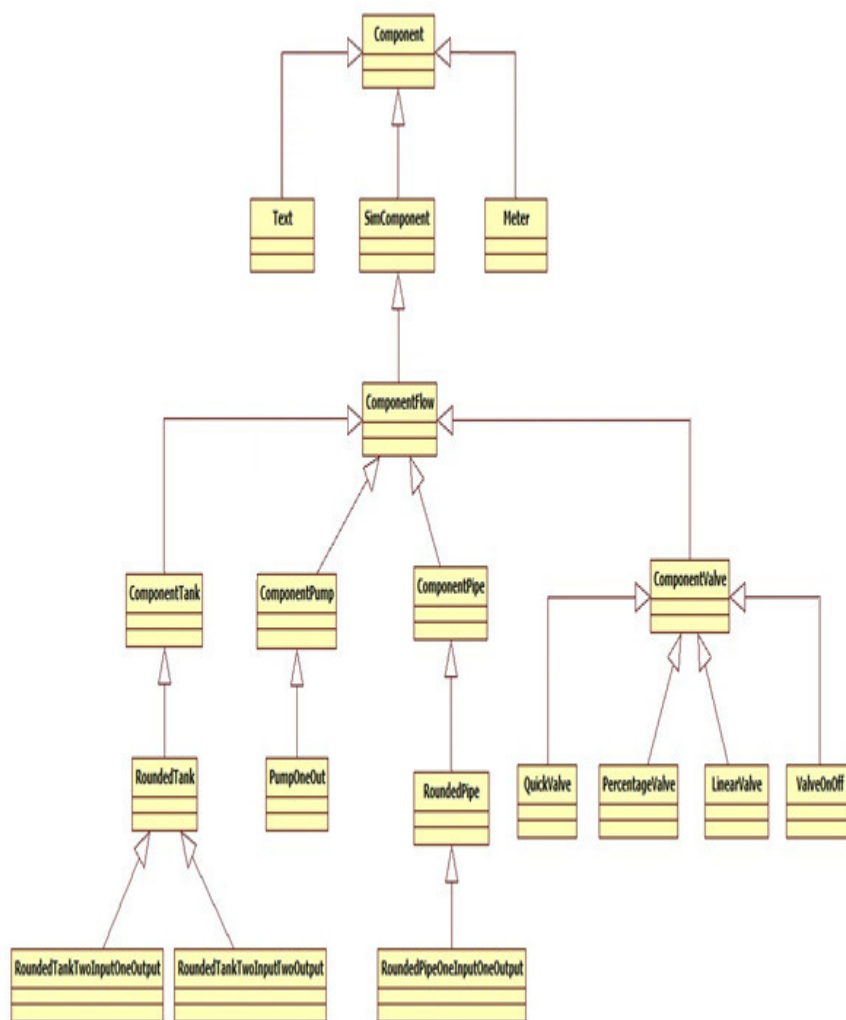


Figura 9 Diagrama dos componentes da simulação

3.2 Desenvolvimento do Software

Para desenvolvimento da interface gráfica utilizou-se a biblioteca Qt (versão 4.6.0) que utiliza linguagem C++. Optou-se pela utilização da biblioteca Qt, pois é multiplataforma, podendo ser compilada para diferentes sistemas operacionais, possui uma vasta e acessível documentação de suas funcionalidades e suporta inúmeros recursos gráficos com exemplos disponíveis na Internet (BLANCHETTE; SUMMERFIELD, 2006).

De acordo com Ezust e Ezust (2007), a biblioteca Qt fornece um número abrangente de classes GUI (*Graphical User Interface*), mais rápidas e flexíveis do que as classes SWING do Java.

A ferramenta desenvolvida nesta proposta é composta basicamente por uma tela principal e uma tela de edição.

A primeira tela (principal) é composta por uma barra de menus contendo todas as funcionalidades do software, uma barra de ferramentas, com as principais funcionalidades do software, como salvar, recortar, copiar e colar, uma área central (3), que permite a inserção dos itens, edição dos mesmos, seu reposicionamento na tela e a ligação entre eles (Figura 10). A tela também possui duas barras laterais, uma barra de itens (1), que contém todos os itens que podem ser inseridos na área central e uma barra de pré-visualização (2), que mostra uma miniatura do item clicado na barra de itens, para possibilitar ao usuário a visualização do mesmo antes de inseri-lo na área central.

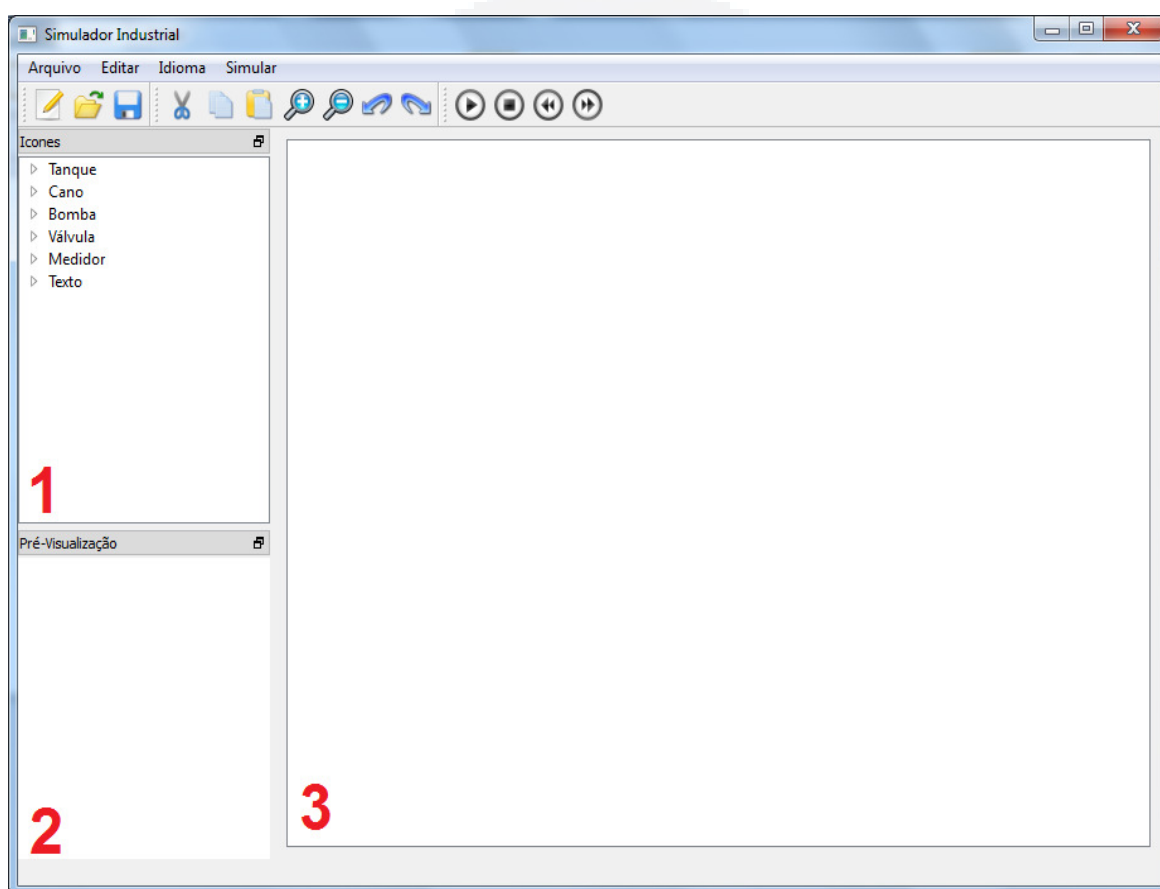


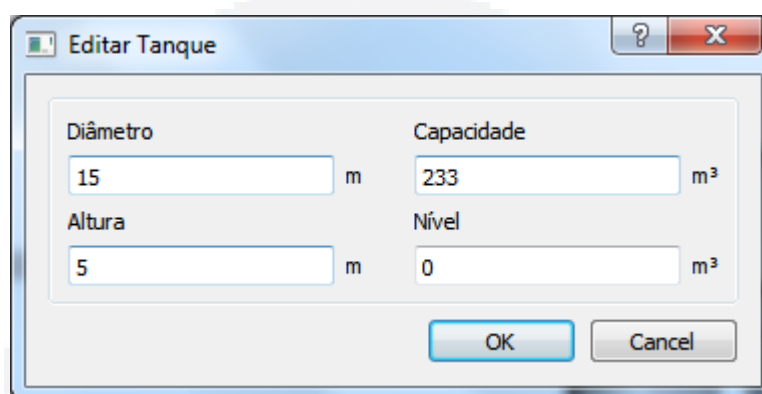
Figura 10 Tela Principal

O ambiente também possui uma tela de edição que é responsável pela configuração de parâmetros próprios de cada um dos itens presentes na aplicação. Mais especificamente, este recurso permite a alteração das dimensões do item selecionado, além de configurações próprias de cada um.

Por exemplo, para itens do tipo tanque pode-se editar o nível inicial de líquido dentro dele e a sua capacidade. Já para canos permite-se a alteração da rugosidade e da velocidade máxima de escoamento do líquido, para bombas pode-se alterar pressão e vazão máxima, para

válvulas abre/fecha pode-se indicar se ela está aberta ou fechada entre outros. Para itens do tipo texto permite-se a edição do tamanho e do tipo da fonte, a alteração do texto inserido e permite colocá-lo em negrito, itálico ou sublinhado e para itens do tipo medidor permite-se apenas a alteração da grandeza que será medida.

Na Figura 11 se apresenta uma tela de edição de um tanque, onde se observam à esquerda as configurações das dimensões (diâmetro e altura) do item e à direita os parâmetros importantes para simulação (capacidade e nível).



A imagem mostra uma janela de diálogo intitulada "Editar Tanque". Ela contém quatro campos de entrada organizados em duas colunas. A primeira coluna, sob o rótulo "Diâmetro", tem um campo com o valor "15" e a unidade "m". A segunda coluna, sob o rótulo "Capacidade", tem um campo com o valor "233" e a unidade "m³". A terceira linha, sob o rótulo "Altura", tem um campo com o valor "5" e a unidade "m". A quarta linha, sob o rótulo "Nível", tem um campo com o valor "0" e a unidade "m³". No canto inferior direito da janela, há dois botões: "OK" e "Cancel".

Diâmetro	Capacidade
15 m	233 m³
Altura	Nível
5 m	0 m³

Figura 11 Tela de edição dos itens

Procurou-se durante o desenvolvimento da ferramenta produzir uma interface amigável de operação. Por exemplo, para inserir um item novo na tela principal, o usuário deve clicar no item desejado (barra lateral) e então arrastá-lo para a área central, ele será posicionado no local onde o botão do mouse for solto. Ao dar-se um duplo clique com o mouse sobre algum item presente na área central, a tela de edição do mesmo é aberta. Este tipo de interface intuitiva deve facilitar sua utilização, mesmo para usuários que não têm muita experiência com montagem de simulações gráficas.

A ferramenta permite para qualquer item inserido na interface de trabalho as funcionalidades de remoção, reposicionamento na tela, ligação com outros itens e edição das propriedades.

Também foram previstas opções comuns de tratamento gráfico como recortar, copiar e colar o item selecionado, além de aumentar ou diminuir a visualização (*zoom*) em até oito vezes, para uma melhor visualização e organização da simulação e os comandos de desfazer e refazer, possibilitando ao usuário maior facilidade na montagem da simulação, para alguns itens é proporcionada a opção de rotação de 90 graus.

Também é possibilitado ao usuário salvar a simulação montada e abrir a simulação salva anteriormente. Para o salvamento gera-se um arquivo de texto, sem cabeçalho, que armazena todas as informações sobre os itens presentes na simulação.

Para itens do tipo texto são salvas informação sobre a fonte (tamanho, família, negrito, itálico), e o texto que estava escrito. Para itens de simulação (bombas, canos, tanques e válvulas) salva-se o número de itens conectados em sua entrada, o número de itens conectados em sua saída e seus respectivos números de identificação (ID), o número de medidores conectados a ele, a grandeza lida pelo mesmo e seus números de identificação (ID), os graus de rotação e todos os parâmetros que podem ser editados na tela de edição, além de salvar o tipo do objeto (tanque, válvula, texto, etc.) e sua posição na tela.

Caso o usuário deseje imprimir a simulação desenvolvida ou salvá-la para visualização em computadores sem o software instalado, é permitida a exportação da visualização da simulação para PDF (*Portable Document Format*) e para os formatos de imagem suportados pelo software, PNG (*Portable Network Graphics*), JPG (*Joint Photographic Experts Group*), BMP (*Windows Bitmap*), PPM (*Portable Pixmap*), TIFF (*Tagged Image File Format*), XBM (*X11 Bitmap*) e XPM (*X11 Pixmap*). Eles foram escolhidos devido a limitações da ferramenta, conforme pode-se observar na Tabela 1. Alguns formatos suportam apenas leitura e não escrita, portanto não é permitida a exportação para estes formatos na simulação.

Tabela 1 Formatos de imagens suportados

Formato	Descrição	Suporte
BMP	Windows Bitmap	Leitura/Escrita
GIF	Graphic Interchange Format	Leitura
JPG	Joint Photographic Experts Group	Leitura/Escrita
PNG	Portable Network Graphics	Leitura/Escrita
PBM	Portable Bitmap	Leitura
PGM	Portable Graymap	Leitura
PPM	Portable Pixmap	Leitura/Escrita
TIFF	Tagged Image File Format	Leitura/Escrita
XBM	X11 Bitmap	Leitura/Escrita
XPM	X11 Pixmap	Leitura/Escrita

Fonte: Nokia (2001)

Além disso, o software suporta a alteração do idioma da ferramenta em tempo de execução, o software permite a troca do padrão (português) para o inglês, pode-se observar na Figura 12 o software em funcionamento no idioma inglês.

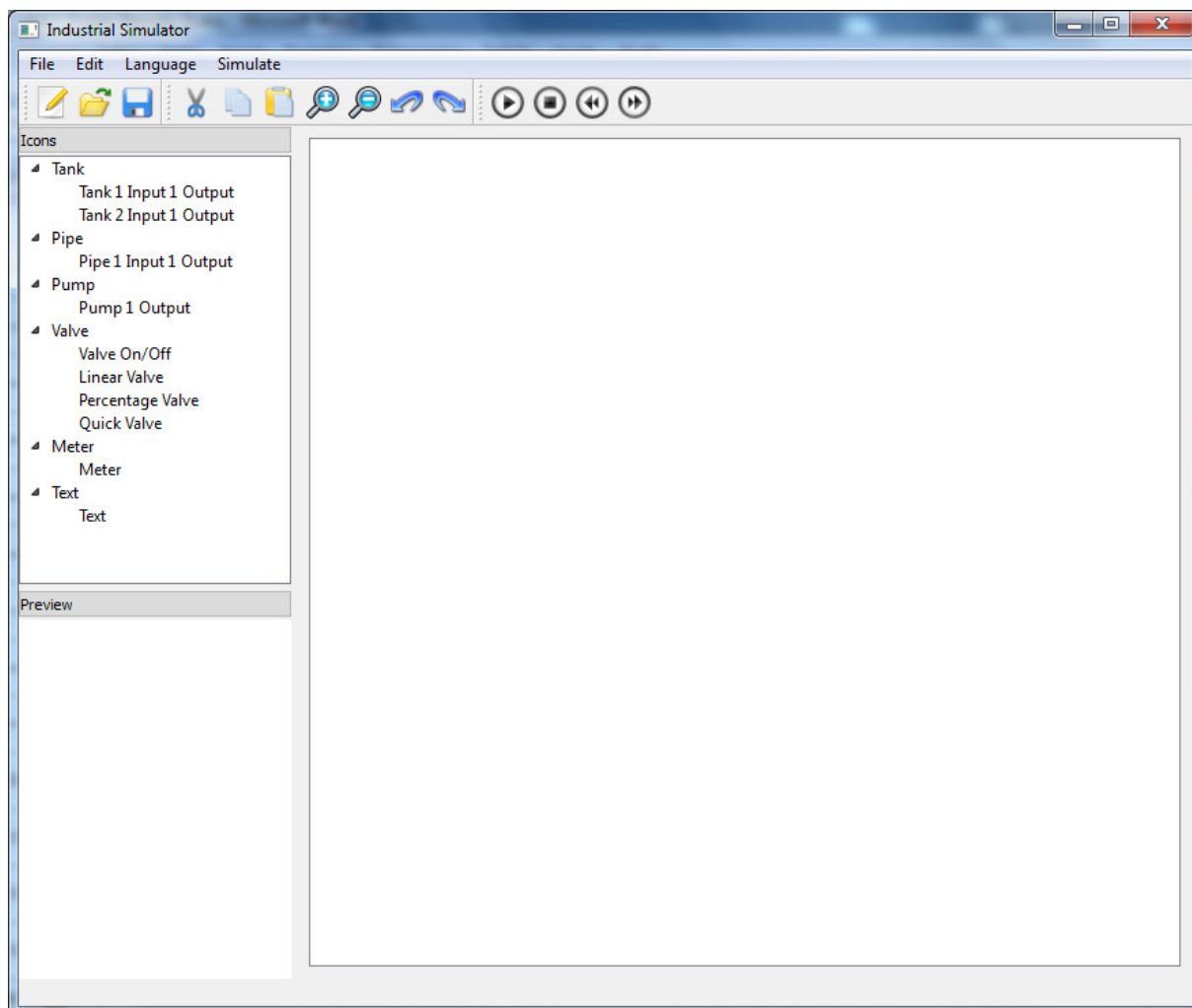


Figura 12 Software executando no idioma inglês

Os componentes a serem simulados devem ser colocados na área de edição à direita. Conforme pode-se observar na Figura 13, um exemplo do componente tanque. O tanque possui uma área no centro, para que o usuário possa visualizar o nível de líquido dentro do mesmo, de forma gráfica (barra colorida possui altura proporcional ao valor determinado do nível).

O nível inicial do tanque pode ser também configurado (tela de edição).

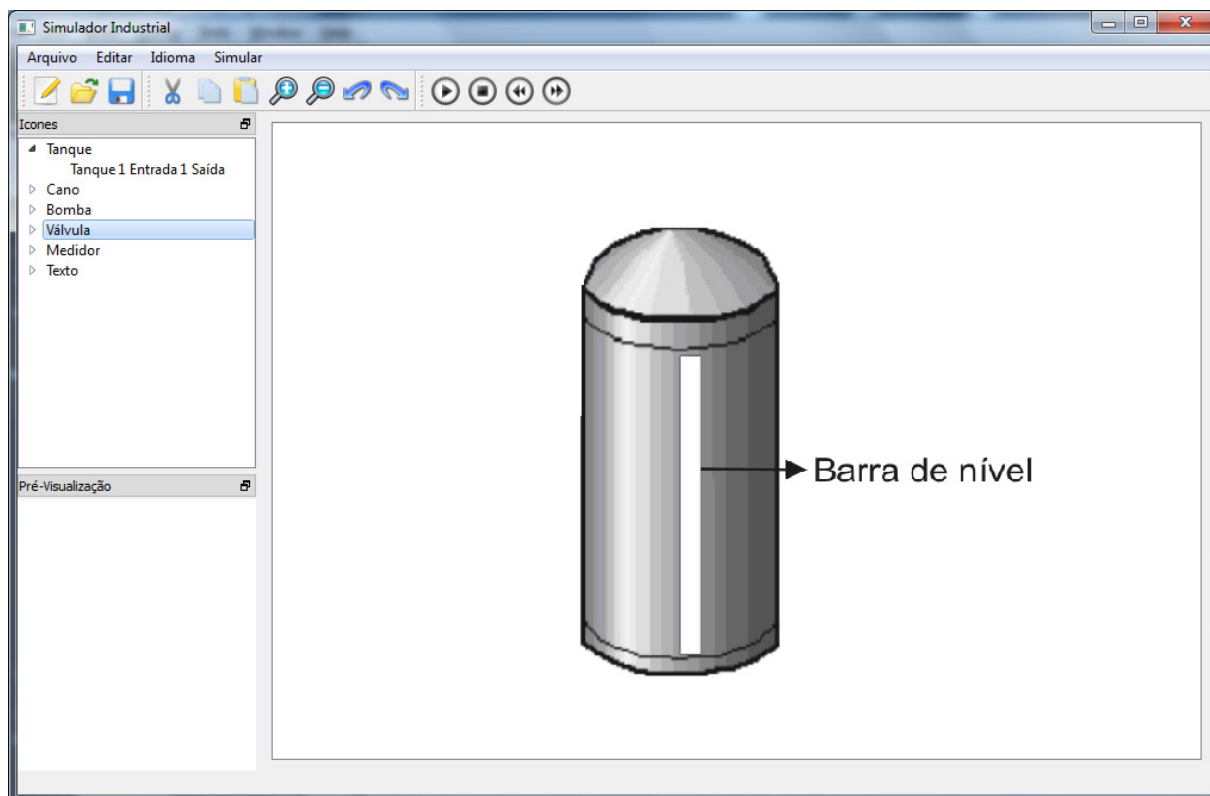


Figura 13 Inserção de um item na tela

Ao clicar com o botão direito do mouse sobre algum item, um menu de opções é aberto, permitindo ao usuário recortar, copiar, colar, excluir, editar ou girar o item selecionado.

Na Figura 14 pode-se observar esse menu após ter-se clicado com o botão direito do mouse sobre um item e automaticamente o menu de opções do mesmo foi aberto, permitindo ao usuário a escolha da opção desejada.

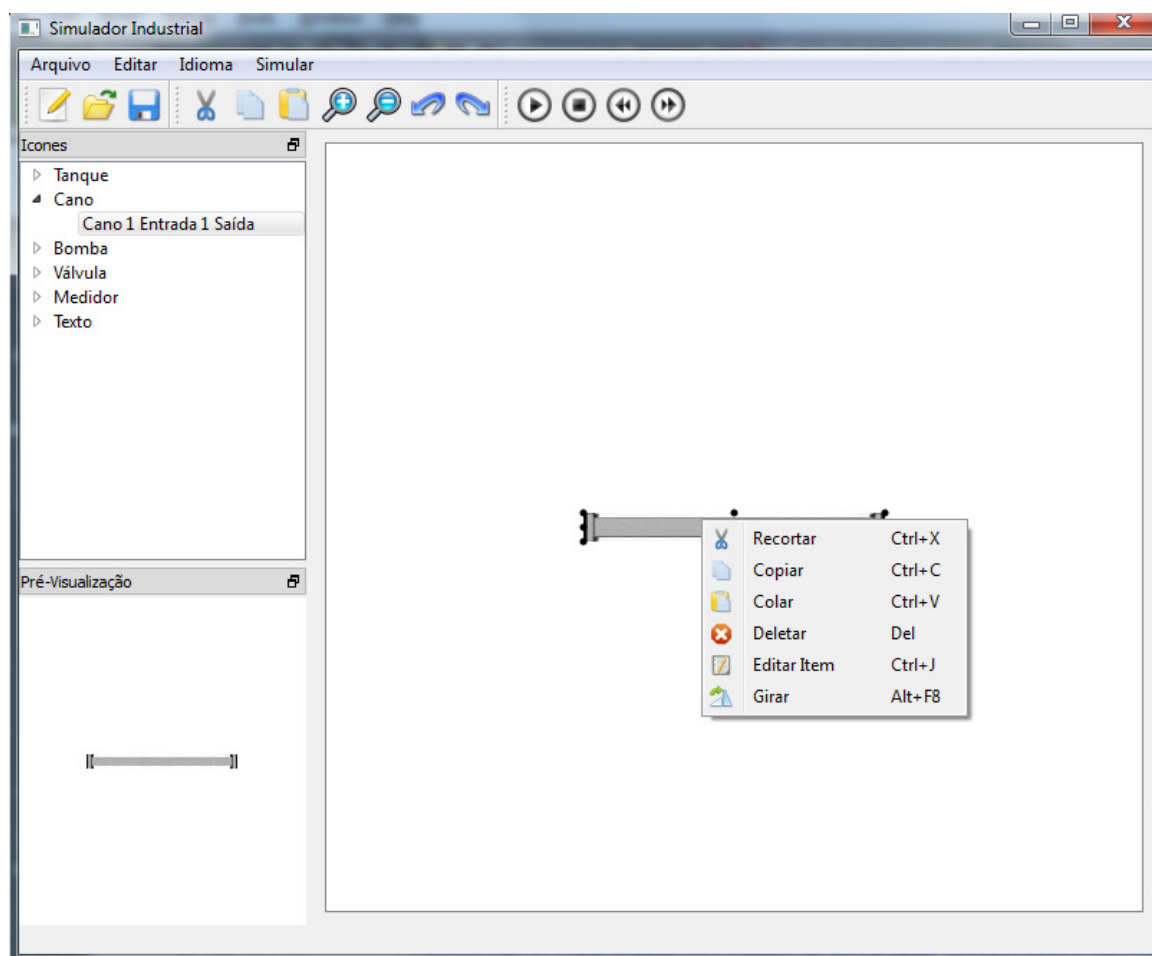


Figura 14 Menu de opções do item

Cada item, exceto itens de texto e medidores, possui pontos de entrada e pontos de saída, que indicam a posição das entradas e das saídas do item. Estas informações servem para o reposicionamento dos itens na tela em caso de colisão e uma lista de entrada e uma lista de saída, contendo todos os itens que estão conectados nestes pontos de entrada e saída, respectivamente.

Se um item A é solto na área central, o software verifica se existe colisão entre ele e algum outro item. Caso a colisão aconteça, presume-se que o usuário deseja conectar os itens. A ferramenta verifica se o item A está mais próximo do ponto de entrada ou do ponto de saída do item com o qual foi conectado (B), caso esteja mais perto de seu ponto de entrada, o item A será reposicionado para o ponto de entrada do item B. No entanto, caso o item B já possua itens conectados em todas as suas entradas o item A será conectado a saída do item B.

Após o reposicionamento e a conexão dos itens na tela, com o item A conectado no ponto de saída do item B, conforme pode-se observar na Figura 15, insere-se o item A na lista de saídas do item B e o item B na lista de entradas do item A.

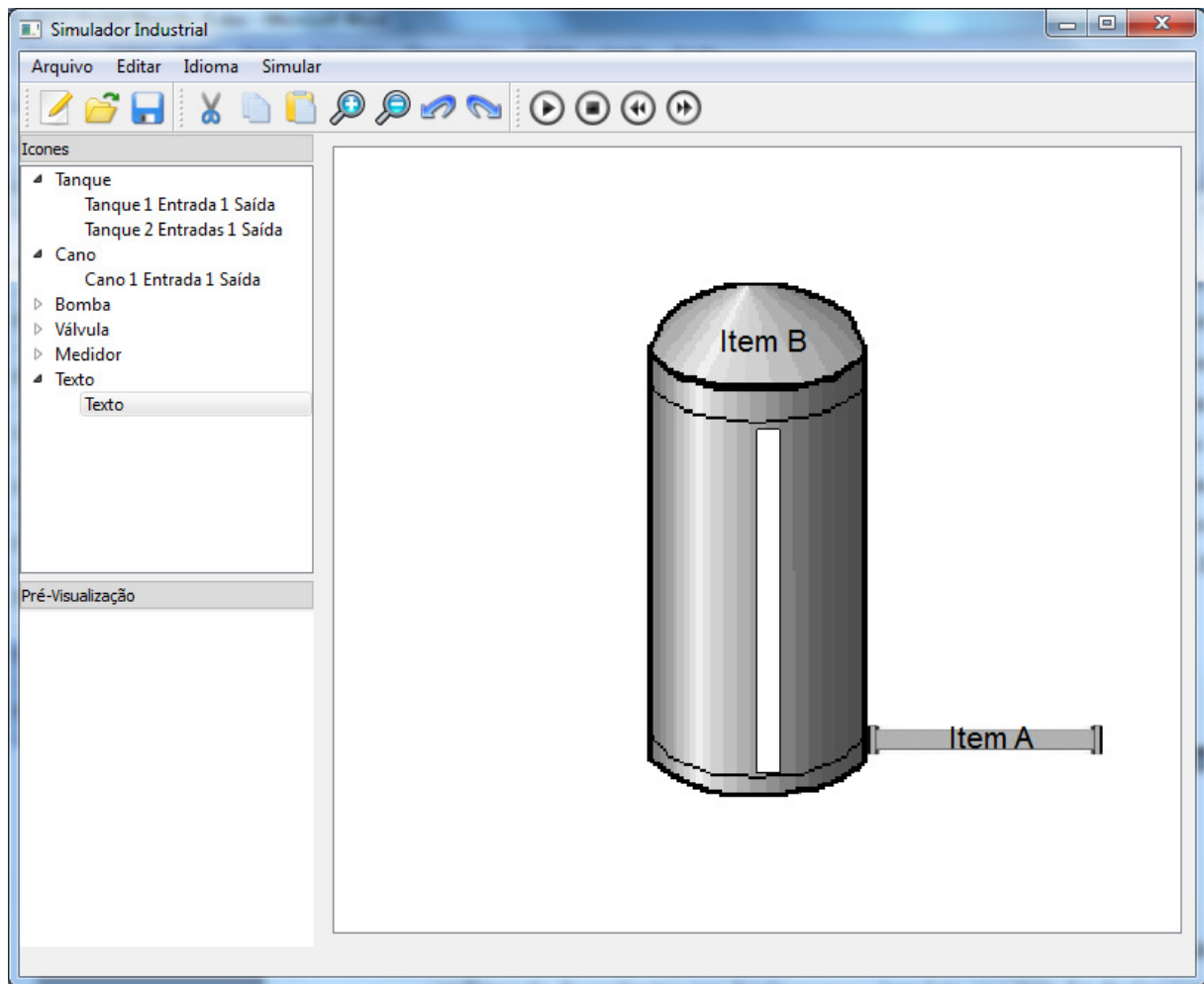


Figura 15 Reposicionamento dos itens

Caso a colisão aconteça com itens medidores, no entanto, uma tela é aberta para a configuração do parâmetro que deseja-se que o item leia. No caso de uma sobreposição ocorrer com um item do tipo texto, nenhuma verificação é feita e o item texto mantém sua posição e fica sobre o item com o qual ele colidiu.

Na Figura 16 pode-se verificar a conexão entre itens distintos. No centro da tela se observa uma bomba, um cano e um tanque conectados. Durante sua montagem a ferramenta verificou automaticamente os itens sobrepostos e alterou suas posições na tela, de acordo com as ligações e com seus pontos de entrada e saída. Ou seja, a saída da bomba está ligada na entrada do cano (ligação 1). Já a saída do cano está ligada na entrada do tanque (ligação 2). Os itens de texto foram adicionados e observa-se que não foram reposicionados, eles ficam sobrepostos aos outros itens.

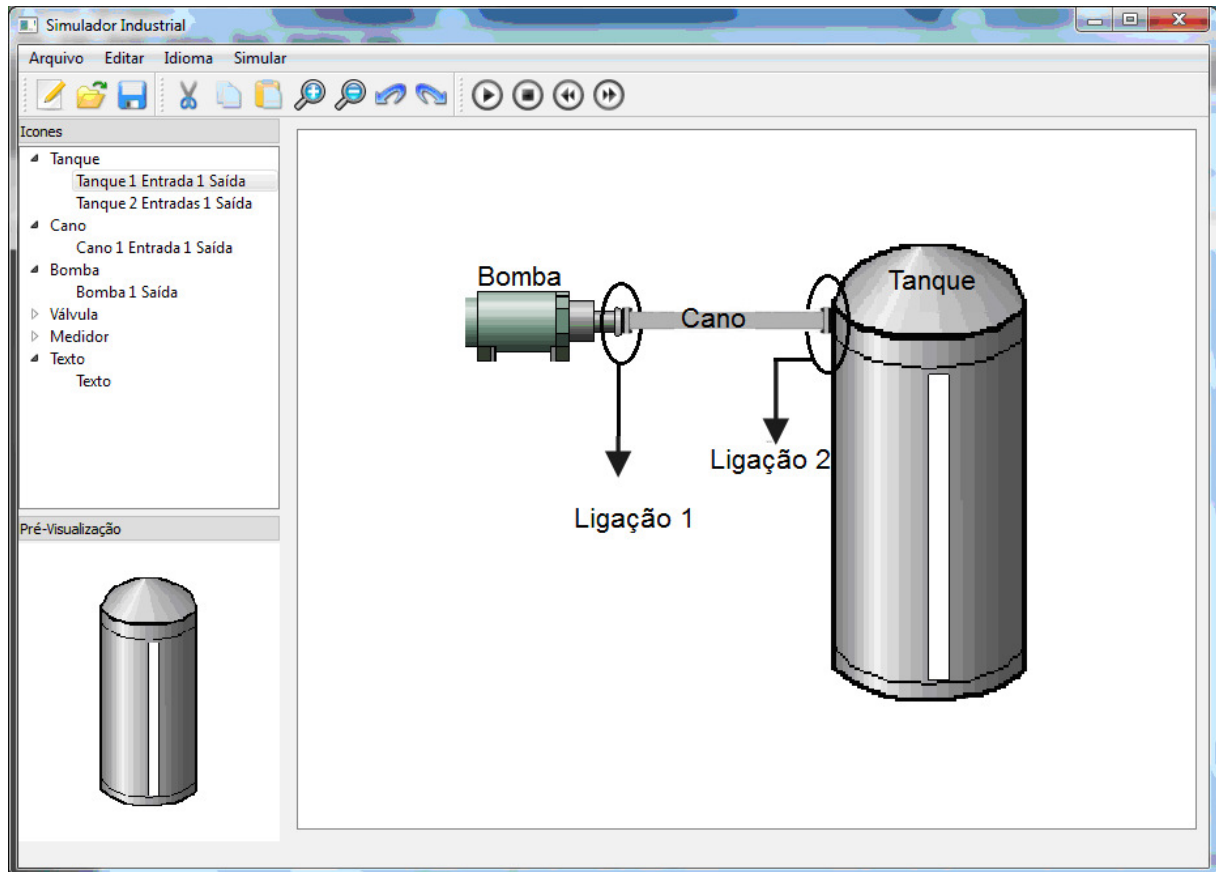


Figura 16 Ligação entre os itens

Em alguns casos, a ligação entre os componentes não é feita, em vista disso, ao tentar iniciar a simulação o software verifica o número de conexões de entrada e saída de todos os objetos presentes na área de simulação, e compara com o número mínimo de conexões de entrada e saída que o objeto deve possuir, caso o objeto contenha um número menor de conexões do que o número mínimo necessário uma mensagem de erro é exibida ao usuário informando que os objetos devem conter conexões para que a simulação possa ter início.

A Figura 17 mostra esse exemplo, a tela com um cano sem nenhuma entrada e uma bomba sem nenhum item conectado a sua saída, após o usuário solicitar o início da simulação é mostrada a mensagem de erro para informá-lo que devem ser feitas as ligações entre os itens.

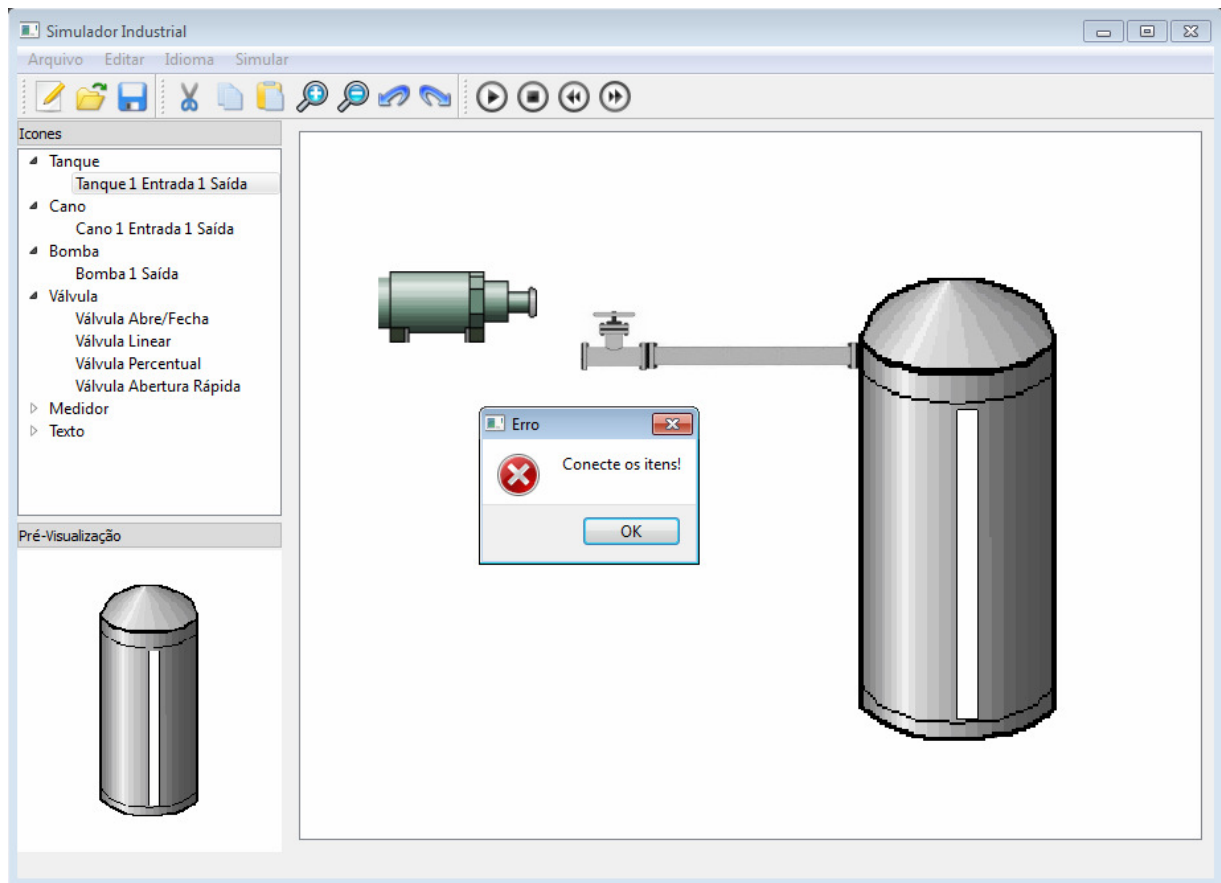


Figura 17 Mensagem de erro ao tentar executar simulação

Caso as listas de entrada e de saída estejam de acordo com as restrições verificadas a simulação tem início. O funcionamento da simulação pode ser visto na seção 3.3.

3.3 Integração e Funcionamento do Software

Para organizar melhor o software foi criado um módulo de simulação dedicado, desenvolvido pelo estudante Anderson Giacomolli, aluno do curso de Engenharia de Controle e Automação, de forma colaborativa com este trabalho. Para facilitar a integração este módulo também foi desenvolvido em linguagem C++ utilizando a biblioteca Qt (GIACOMOLLI, 2011). A idéia geral é que qualquer projeto projeto de simulação de processos industriais tenha sua solução dividida em dois ambientes complementares: módulo de simulação e ferramenta de interface gráfica. Se não for realizada a integração dos dois ambientes pode-se apenas montar a interface da simulação, pois todos os comandos para que a simulação possa acontecer estão presentes na biblioteca. Para os testes os dois módulos são conectados e se atualizam sincronamente.

Em vista disso, para todos os itens que podem ser inseridos na área gráfica, foram criados objetos referentes para cada item, tanto no módulo de simulação quanto na ferramenta gráfica. Na inserção de um item na interface, cria-se um objeto deste item na ferramenta e este objeto possui um objeto referenciando o item correspondente do módulo de simulação e qualquer alteração sofrida na ferramenta é comunicada para o módulo de simulação, atualizando ambos ao mesmo tempo.

Quando o usuário solicita o início da simulação, o software se comunica com o módulo de simulação, que faz todos os cálculos necessários, atualiza os itens e após o término disso envia sinais que são recebidos pela interface, através de *slots*, e então o software atualiza os itens da interface de acordo com os valores recebidos.

Visando facilitar a depuração por parte do usuário, os itens presentes na interface podem ter suas propriedades alteradas durante a simulação, não necessitando a pausa da mesma para efetuar as alterações, por exemplo, pode-se abrir ou fechar uma válvula, ou alterar as características físicas de um componente qualquer. Na Figura 18 pode-se observar a mudança das configurações de um cano.

No entanto as funcionalidades de inserção, exclusão e reposicionamento na tela podem ocorrer com a simulação em andamento apenas para itens medidores e texto, para o resto dos componentes essas alterações não são permitidas.

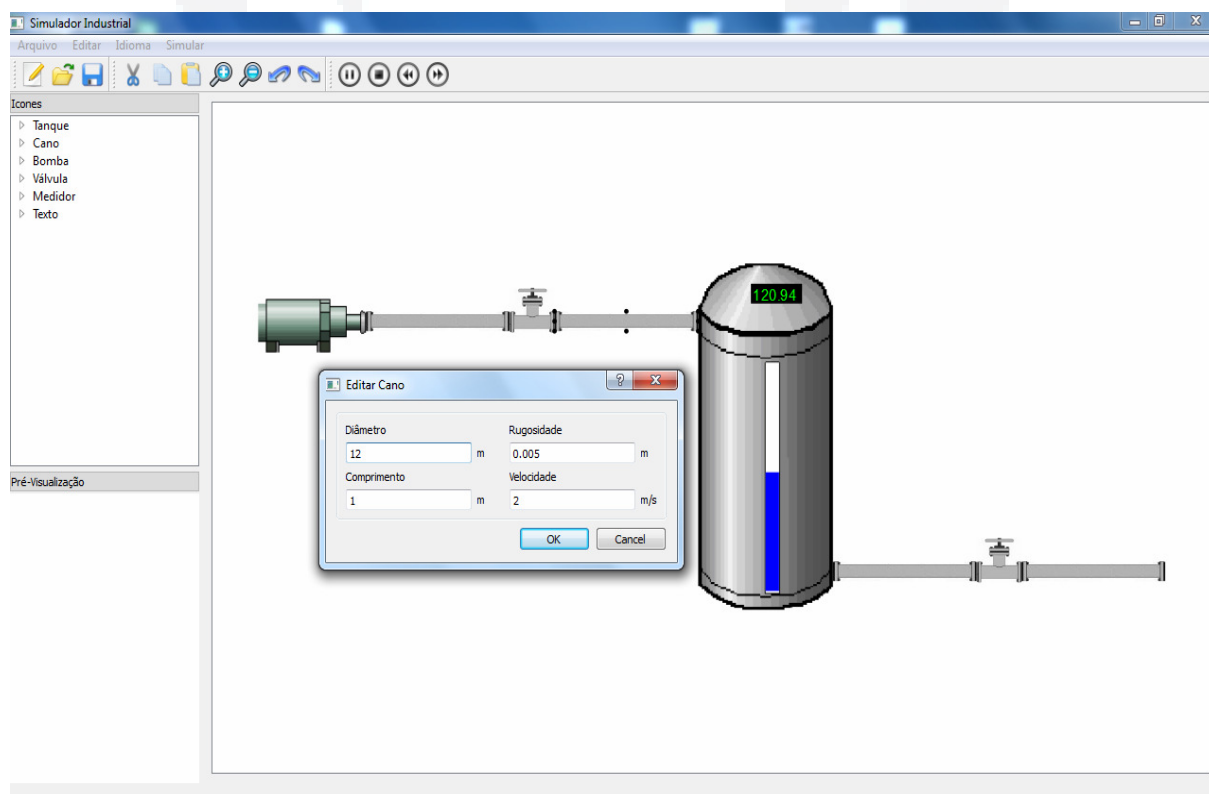


Figura 18 Alteração de propriedades com simulação em andamento

O software possui quatro botões de simulação, um é reservado para iniciar e pausar a simulação. Quando o usuário clica para iniciar a simulação, o botão tem o seu ícone alterado para um ícone correspondente a pausa e a simulação tem início, ao clicar-se novamente neste botão a simulação é pausada, mantendo os dados da simulação que estava em andamento, permitindo ao usuário a continuação da simulação no momento em que foi parada e tendo seu ítem novamente alterado para o ítem de iniciar simulação.

Ao iniciar a simulação é gerado um arquivo de histórico no formato *CSV (Comma-Separated Values)*, contendo informações sobre todos os componentes da simulação em relação ao tempo. Cada componente tem informações diferentes para ser salvas. Os tanques, por exemplo, possuem dados de vazão de entrada, vazão de saída e nível. Os canos possuem dados de vazão. Bombas contêm vazão e pressão e válvulas salvam a vazão. Com este formato de arquivo podem ser gerados gráficos, facilitando-se assim fazer uma análise da simulação montada.

Outro botão importante é o de parar simulação. Ao clicar-se nele após a simulação ter início, ele para a simulação e volta os dados referentes aos valores iniciais, deixando a tela da mesma forma que estava antes da simulação ter início.

A simulação também possui um botão para aumentar a sua velocidade e um para diminuir a sua velocidade, fazendo com que a simulação do sistema aumente ou diminua de velocidade, conforme a necessidade do usuário.

4 RESULTADOS EXPERIMENTAIS

Para verificar o funcionamento do software desenvolvido, o mesmo foi integrado com o módulo de simulação anteriormente citado.

A partir daí diferentes processos puderam ser simulados. Inicialmente montou-se uma simulação básica, inserindo uma bomba, quatro canos, duas válvulas (uma válvula abre/fecha e uma válvula linear) e um tanque.

A bomba é a entrada do sistema, fornecendo o líquido para o sistema funcionar. Ela possui a sua saída ligada na entrada de um dos canos, responsável por transportar o líquido. A saída do cano está ligada na entrada da válvula abre/fecha, que permite ou impede a passagem do líquido. A válvula tem sua saída ligada em outro cano, que possui um medidor conectado a ele medindo a sua vazão. Esse cano transporta a água para o tanque, e possui sua saída ligada na entrada do tanque que vai enchendo conforme a água vai caindo do cano. O tanque também possui um medidor conectado exibindo o seu nível. Entretanto o tanque tem ligado em sua saída a entrada de outro cano, que está também ligado a um medidor responsável pela exibição de sua vazão, recebendo água do tanque e passando para uma válvula linear que está conectada a sua saída, onde é permitida a alteração da porcentagem de abertura.

A fim de ter maior controle da quantidade de líquido que sai do tanque, a válvula linear tem sua saída conectada com um cano que despeja o líquido para fora do sistema. O tanque possui uma barra de nível responsável por mostrar ao usuário a quantidade de água presente no tanque, permitindo ao usuário um maior controle da simulação.

Após a montagem deste exemplo uma simulação foi iniciada e após alguns segundos capturou-se a tela ilustrada na Figura 19, onde percebe-se que a barra de nível do tanque está mostrando que o tanque possui líquido dentro, para saber o valor exato desse nível um medidor foi conectado ao tanque para informar esse nível, também possui medidores em um cano conectado a entrada do tanque e em outro cano conectado a saída do tanque, ambos lendo a vazão do seu cano correspondente.

Pode-se observar que as vazões dos dois canos são diferentes. Isso se deve aos diferentes valores de abertura de cada uma das válvulas do processo simulado.

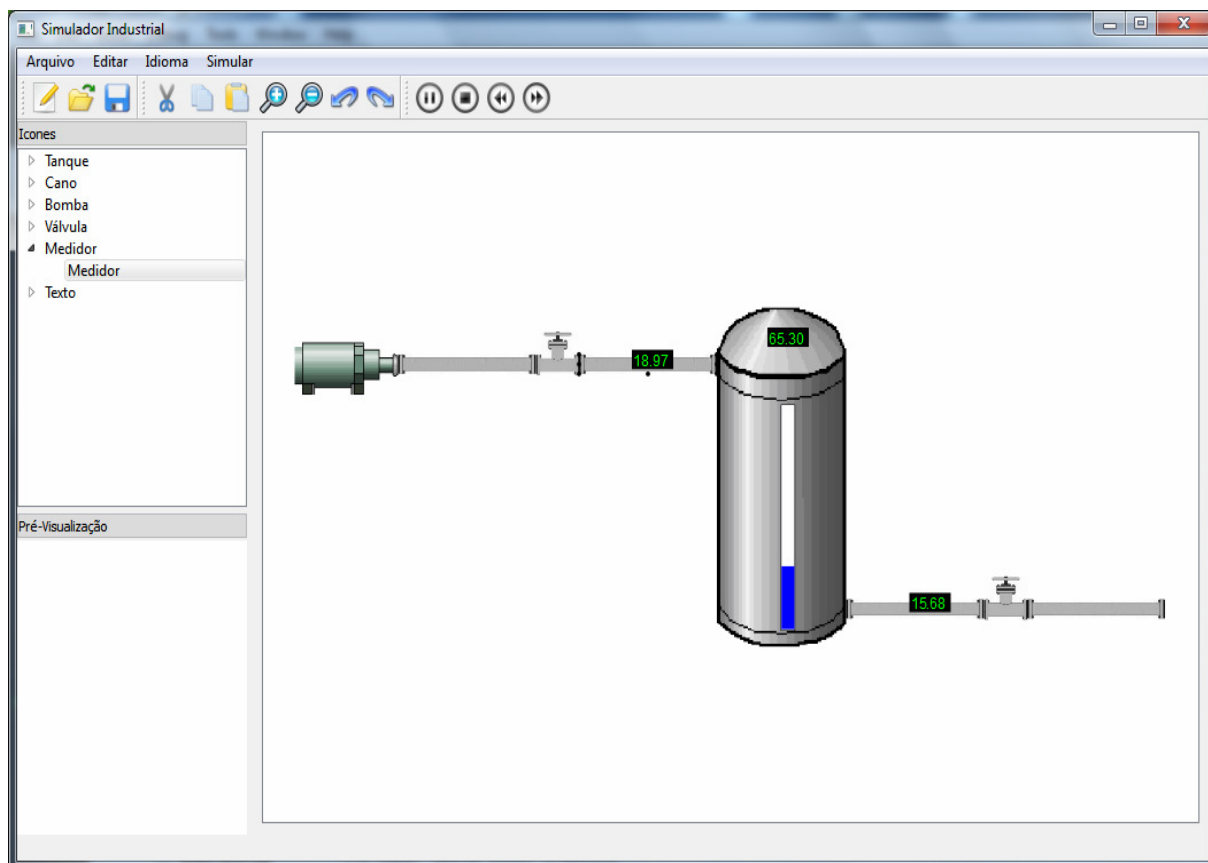


Figura 19 Simulação em andamento

Além do exemplo anterior, projetou-se outra simulação com dois tanques em série.

Este novo sistema simulado é composto por uma bomba, cinco canos, duas válvulas, uma válvula percentual e uma válvula abertura rápida, e dois tanques.

A diferença principal entre os dois ambientes montados (simulação anterior e nova) é que o líquido que sai do primeiro tanque, neste caso, não é jogado para fora do sistema e sim para dentro de outro tanque.

Conforme o líquido vai saindo do primeiro tanque ele vai entrando no segundo. Isso pode ser observado na Figura 20.

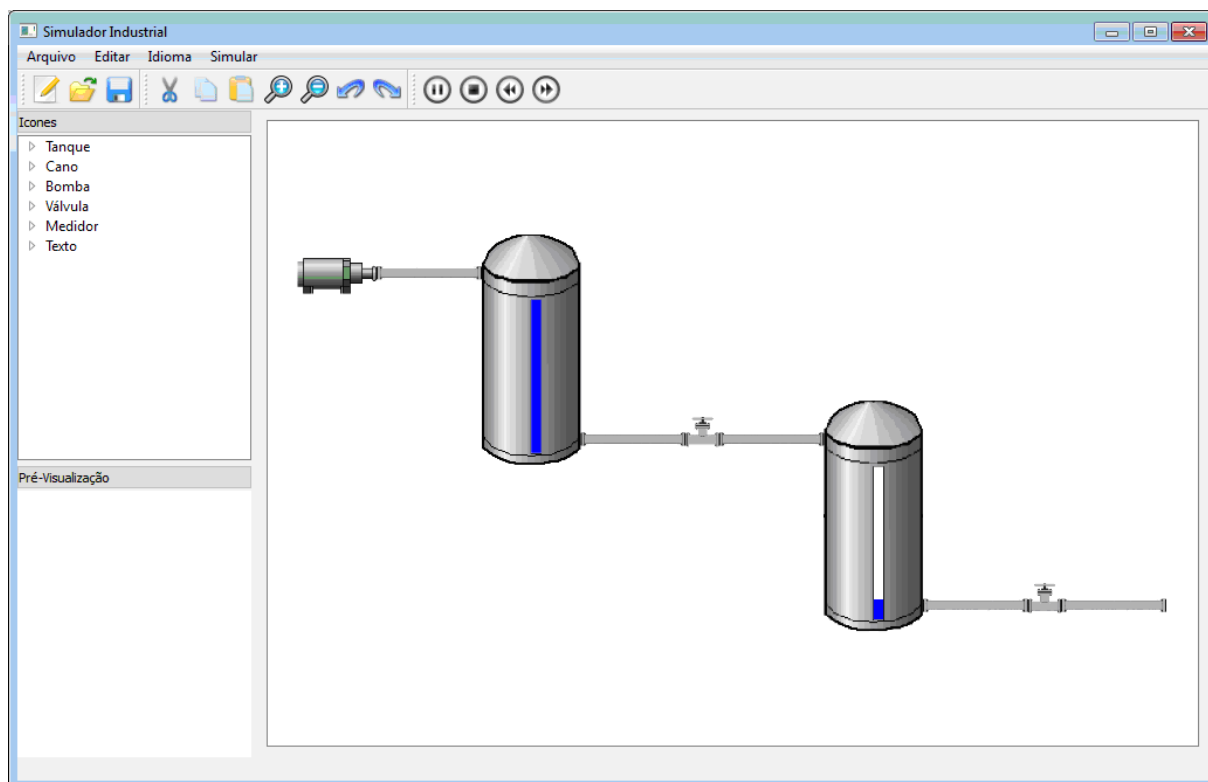


Figura 20 Simulação com dois tanques em série

Em outro experimento montado, inseriu-se na simulação anterior, novos componentes, no entanto sem conexão nenhuma com os já existentes, montou-se basicamente duas simulações rodando em paralelo, conforme pode-se observar na Figura 21.

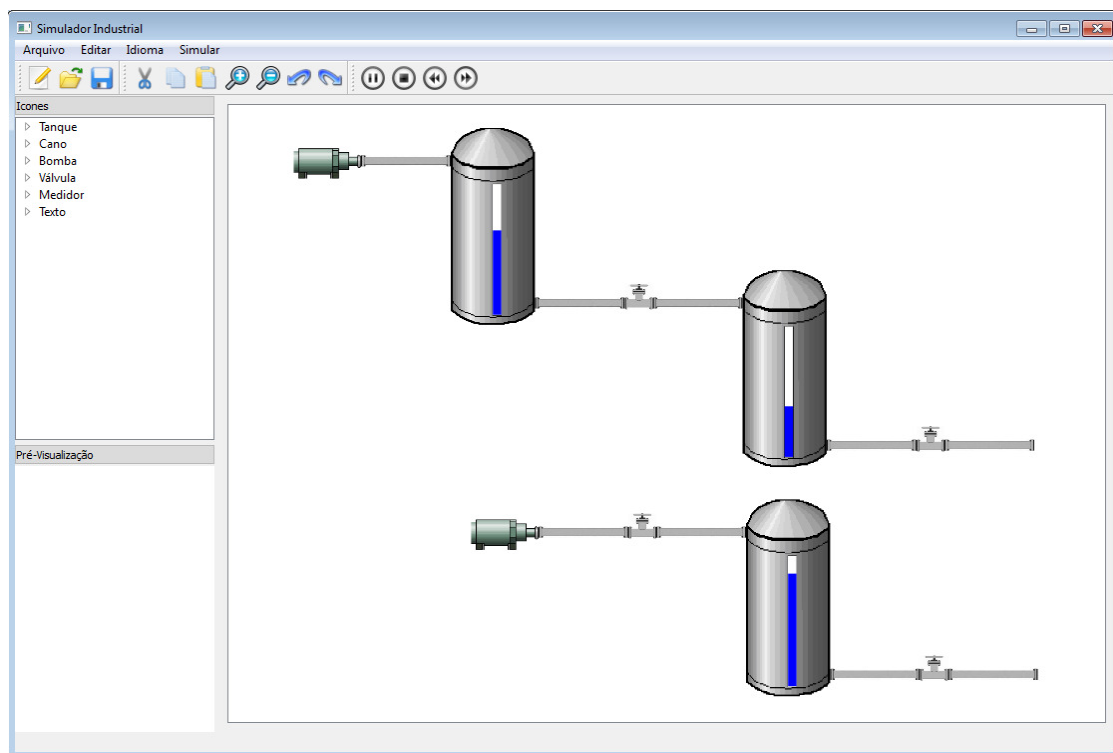


Figura 21 Simulação paralela em andamento

5 CONCLUSÃO

Este trabalho apresentou uma ferramenta gráfica para modelagem de ambientes de simulação de processos industriais. Inicialmente foram descritos conceitos sobre softwares simuladores, feita uma análise das soluções existentes e identificou-se a necessidade de um sistema para tal propósito.

Para a implementação do sistema procurou-se desenvolver uma interface amigável com o usuário para a exibição dos resultados e montagem da simulação desejada. O software foi desenvolvido de forma genérica para que novos componentes e novas funcionalidades possam ser inseridas com facilidade.

O software desenvolvido faz parte de uma iniciativa de um projeto para simular processos industriais em ambiente acadêmico, os testes de integração com o simulador que está sendo desenvolvido indicam que o ambiente é satisfatório e permite-se a continuação do desenvolvimento do software por parte de outros alunos.

Como trabalhos futuros pretende-se a implementação das seguintes funcionalidades:

- Inserir novos componentes: possibilitar suporte para novos componentes na interface, como temperatura e esteiras;
- Inserir funções de animação: com isso serão mais facilmente visualizados os novos componentes, como a esteira, e melhorará a exibição do enchimento do tanque facilitando a visão e compreensão do usuário acerca do que acontece na simulação;
- Área para adição de novos componentes: Criação de uma área para que o usuário possa inserir novos componentes por uma interface gráfica. Bastando colocar a imagem do componente, definir alguns parâmetros para configurá-lo e logo em seguida o componente torna-se parte da interface, contendo todas as funcionalidades desejadas.

REFERÊNCIAS

ASHRAF, A. M. **Design and Use of Instruments for the Measurement of Software Usefulness**. 2004, 46 p. Dissertação (Mestrado) – Department of Computer Science - Michigan Technological University, Michigan.

ASSIS, D. M.; TANNOUNS, K. **Desenvolvimento de Software Simulador Aplicado em Reatores de Leito Fluidizado**. In VI Congresso Brasileiro de Engenharia Química em Iniciação Científica, 2005, Campinas – SP. Anais . . . Campinas, São Paulo, 2005. 7 p. Disponível em < <http://www.feq.unicamp.br/~cobeqic/top02.pdf> >. Acesso em; 10 ago. 2010.

BLANCHETTE, J.; SUMMERFIELD, M. **Getting Started. C++ GUI Programming with Qt 4**, p 3-12, 2006. ISBN 0-13187-249-4.

BURD, L. **Desenvolvimento de Software para Atividades Educacionais**. 1999, 241 p. Dissertação (Mestrado) - Departamento de Engenharia de Computação e Automação Industrial - Universidade Estadual de Campinas, Campinas, 1999.

CADSOFT. EAGLE version 5.10.0: EAGLE Manual, Version 5, 7th Edition.

CAMPOS, F. O. *et al.* **Softwares em ambientes educacionais**. In XII Congresso Regional de Informática e Telecomunicações, 2008, Cuiabá – MT. Anais . . . Cuiabá, Mato Grosso do Sul, 2008. 15 p. Disponível em <<http://www.sucesumt.org.br/mtdigital/anais/files/Softwaresemambienteseducacionais.pdf>> Acesso em: 10 ago. 2010.

CARDOSO, D. G.; BARANAUSKAS M. C. C. **Um Caderno Digital nos laptops educacionais: proposta conceitual e ferramenta**. Revista Brasileira de Informática na Educação, v. 17, n. 2, p. 59-73, mai 2009. Disponível em: <http://www.br-ie.org/pub/index.php/rbie/article/viewFile/99/86> Acesso em 04 jun. 2011.

DA SILVA, V. O. **Um Modelo para Avaliação e Implantação da Mudança de um Sistema de Produção Convencional para Manufatura Celular Empregando Simulação Discreta**, 2003, 125 p. Dissertação (Mestrado em Engenharia) - Programa de Pós-Graduação em Engenharia de Produção, Universidade Federal de Santa Catarina, Florianópolis, 2003.

DALHEIMER, M. K. **A Comparison of Qt and Java for Large-Scale, Industrial-Strength GUI Development**. Klarälvdalens Datakonsult AB, 2002, 12 p.. Disponível em <http://www.dca.fee.unicamp.br/portugues/pesquisa/seminarios/2010/artigos/santos_martino.pdf>. Acesso em: 04 ago. 2010.

DE OLIVEIRA, R. C. **Química e Cidadania: Uma Abordagem a Partir do Desenvolvimento de Atividades Experimentais Investigativas**. 2009, 150 p. Dissertação (Mestrado em Educação) – Programa de Pós-Graduação em Educação e Ciências Humanas, Universidade Federal de São Carlos, São Carlos, 2009.

DEPARTMENT OF ENERGY. VisIt version 2.1.0: VisIt User's Manual, Version 1.5, 2005.

DOS SANTOS, R. P.; COSTA, H. A. X. TBC-AED e TBC-AED/WEB: **Um Desafio no Ensino de Algoritmos, Estruturas de Dados e Programação**. In: IV Workshop em Educação em Computação e Informática do Estado de Minas Gerais, 2005, Varginha – MG. Anais . . . Varginha, Minas Gerais, 2005, 15 p. Disponível em <<http://www.cos.ufrj.br/~rps/pub/completos/2005/WEIMIG.pdf>>. Acesso em: 19 out. 2010.

DUARTE, A. O. S. M. *et al.* **Modelagem e Simulação de um Processo Industrial**. In XXXII Congresso Brasileiro de Ensino de Engenharia, 2004, Brasília – DF. Anais . . . Brasília, Distrito Federal, 2004. 12 p. Disponível em <http://www.deec.ufpa.br/~peteletrica/artigos/mod_sim_proc_ind.pdf>. Acesso em: 14 ago. 2010.

ELEKTOR, LogicSim: Um simulador multi-plataforma gratuito, Revista Elektor, Janeiro de 2008. Disponível em <<http://www.elektor.com.pt/revistas/2008/janeiro/logicsim.379335.lynkx>>. Acesso em: 19 nov. 2010.

EZUST A.; EZUST P. **Introduction to QT. An Introduction to Design Patterns in C++ with Qt 4**, p 81-94, 2007. ISBN 0-13187-905-7

GIACOMOLLI A. A. **Simulador de Processos Industriais com Interface para CLP**. 2011, 76 p. Trabalho de Conclusão (Etapa 1) – Curso de Engenharia de Controle e Automação, Centro Universitário UNIVATES, Lajeado - RS, 2011.

GUILLERMO, O. E. P. *et al.* **O Poder das Simulações no Ensino de Hidráulica**. V Ciclo de Palestras Inovações em Tecnologia na Educação - CINTED/UFRGS, 2005, Porto Alegre – RS. Anais . . . Porto Alegre, Rio Grande do Sul, 2005, 10 p. Disponível em <http://www.cinted.ufrgs.br/renote/maio2005/artigos/a67_hidraulica.pdf>. Acesso em: 06 set. 2010.

HENDRES, C. A.; KAIBER, C. T. **A Utilização da Informática como Recurso Didático nas Aulas de Matemática**. Revista de Ciências Naturais e Humanas, v. 7, n. 1, p. 25-38, jan 2005. Disponível em <<http://www.ulbra.br/actascientiae/edicoesanteriores/Acta%20Scientiae%20v7%20n1%202005.pdf>>. Acesso em: 02 set. 2010.

KAMMER, R. **Desenvolvimento de um Tutorial de Pascal Utilizando o Learning Space**. 1999. 71 p. Trabalho de Conclusão de Curso – Universidade Regional de Blumenau – Curso de Ciência da Computação – Blumenau, SC, 1999.

LOPES, J. S. B. *et al.* **Controle Avançado de Processos Industriais de Petróleo Utilizando um Sistema de Inferência Neuro-Fuzzy para Geração Dinâmica de Set-Points**. In V Congresso Rio Automação, 2009 , Rio de Janeiro – RJ. Anais . . . Rio de Janeiro, Rio de Janeiro, 2009, 6 p. Disponível em <<http://www.sigaa.ufrn.br/sigaa/verProducao?idProducao=480&key=31787f063fbe19ee5867d0fd15124211>>. Acesso em: 06 ago. 2010.

MENDES, M. A.; FIALHO F. A. P. **Avaliação de Simuladores Aplicados na Educação Tecnológica a Distância**. In XI Congresso Internacional de Educação a Distância, 2004, Salvador – BA. Anais . . . Salvador, Bahia, 2004. 11 p. Disponível em <<http://www.abed.org.br/congresso2004/por/pdf/036-TC-B1.pdf>>. Acesso em 11 ago. 2010.

MENESES, R. C.; SILVA, G. C. **Um Ambiente Computacional de Apoio à Aprendizagem de Instruções Assembly**. In: I Encontro Nacional de Informática e Educação, 2009, Cascavel – PR. Anais . . . Cascavel, Paraná, 2009. 10 p. Disponível em <<http://www.inf.unioeste.br/enined/anais/enined/A34.pdf>>. Acesso em: 11 ago. 2010.

MHJ-SOFTWARE, SPS-VISU S5/S7, Demo Version, Benutzerhandbuch, 132 p. 2009 Disponível em < <http://www.winplc7.com/Download/SPS-VISU-V4-80.EXE>>. Acesso em 19 nov. 2010.

MINKE, J. **Developing a generic OpenGL Qt Viewer**. 2009, 27 p. Trabalho de Conclusão de Curso - University of Zurich – Suíça, 2009.

MUCHALUAT, D. C. *et al.* **Editores Gráficos para Estruturas com Composições**. In: X Simpósio Brasileiro de Engenharia de Software, 1996, Florianópolis – SC. Anais . . . Florianópolis, Santa Catarina, 1996. 12 p. Disponível em < ftp://ftp.telemidia.puc-rio.br/~lfgs/docs/conferencepapers/1996_10_MUCHALUAT.pdf>. Acesso em: 28 out. 2010.

MUNIZ, A. R.; MARCZAC, L. D. F. **Uso do Software Maple no Ensino de Transferência de Calor**. In XXIX Congresso Brasileiro de Ensino de Engenharia, 2001, Porto Alegre – RS. Anais . . . Porto Alegre, Rio Grande do Sul, 2001. 9 p. Disponível em <<http://www.pp.ufu.br/Cobenge2001/trabalhos/NTM097.pdf>>. Acesso em: 14 ago. 2010.

NETO, C. C. B. C. **Uma Ferramenta de Modelagem e Simulação de Aplicações Interativas. Monografia** (Graduação em Ciência da Computação) - Departamento de Ciência da Computação, Instituto de Matemática, Universidade Federal da Bahia, Salvador, 2006.

NOKIA, Qt Reference Documentation. QPixmap Class Reference. Disponível em <<http://doc.qt.nokia.com/4.7/qpixmap.html>> Acesso em: 14 jun. 2011.

OLIVEIRA, R. A. *et al.* **Desenvolvimento e Avaliação de Sistema Multimídia para Ensino e Aprendizado de Irrigação**. Revista Brasileira de Engenharia Agrícola e Ambiental, v. 6, n. 3, p.553-556, fev. 2002. Disponível em <<http://www.scielo.br/pdf/%0D/rbeaa/v6n3/v6n3a30.pdf>>. Acesso em: 19 out. 2010.

PEZZOTA, C. A. **Emprego da Orientação a Objetos para Caracterização de Recursos em um Ambiente de Simulação de Instrumentos**, 2001, 82 p. Dissertação (Mestrado em Metrologia) – Programa de Pós-Graduação em Metrologia Científica e Industrial, Universidade Federal de Santa Catarina, Florianópolis, 2001.

PRETO, T. M. **Considerações Sobre uso de Ambientes para Ensino a Distância. Tuiuti: Ciência e Cultura**, n. 9, p 85-96, abr 2002. Disponível em <<http://www.utp.br/tuiuticienciaecultura/FACET/FACET%2029/PDF/Art%206.pdf>>. Acesso em: 13 ago. 2010.

SANTANCHÈ, A.; TEIXEIRA, C. A. C. **Integrando Instrucionismo e Construcionismo em Aplicações Educacionais através do Casa Mágica**. In Workshop de Informática na Escola, 1999, Rio de Janeiro – RJ. Anais . . . Rio de Janeiro, Rio de Janeiro, 1999. 13 p. Disponível em < <http://www.lis.ic.unicamp.br/~santanch/publications/WIE99-CasaMagica.pdf>>. Acesso em: 19 nov. 2010.

SCHROEDER, G. N. *et al.* **Simulador Gráfico para Controle de Robôs Móveis Omnidirecionais**. Journal of Computer Science, v 4, n. 4, p 38-47, ago 2005. Disponível em <<http://www.dcc.ufla.br/infocomp/artigos/v4.4/art05.pdf>>. Acesso em 09 ago. 2010.

VALENTE, J. A. **Análise dos Diferentes Tipos de Softwares Usados na Educação. O Computador na Sociedade do Conhecimento**, p 71-86, 1999. Disponível em <<http://rxmartins.pro.br/teceduc/computador-sociedade-conhecimento.pdf>>. Acesso em: 03 set. 2010.

VIEIRA, G. E. **Uma Revisão sobre a Aplicação de Simulação Computacional em Processos Industriais**. In XIII Simpósio em Engenharia de Produção, 2006, Bauru – SP. Anais . . . Bauru, São Paulo, 2006. 10 p. Disponível em <http://www.simpep.feb.unesp.br/anais/anais_13/artigos/676.pdf>. Acesso em: 04 ago. 2010.

