

Design e Ergonomia aplicados à Interface de um Compilador Front End

Grasyelle Fonseca, Frederico Coelho

Universidade Presidente Antônio Carlos – UNIPAC
Faculdade de Ciência da Computação – “Facics”

grasyfonbcena@bol.com.br
fredericocoelho@unipac.br

Resumo. *O presente texto busca agregar significado aos estudos e aplicações de designer e ergonomia aplicando as técnicas destas áreas ao planejamento, desenvolvimento e aperfeiçoamento de interfaces para um compilador Front End; desenvolvido no âmbito educacional, para a aprendizagem da Pseudolinguagem Portugol. Para tal, vincularam-se atividades de cunho prático e teórico na contextualização de técnicas para formulação de um ambiente atrativo e eficiente.*

1. Introdução

O avanço da tecnologia no desenvolvimento de softwares tem proporcionado ampliar as áreas de aplicações de sistemas, desde o âmbito comercial até o educacional. E na mesma amplitude cresce a necessidade de novas técnicas para projetar, avaliar e implementar interfaces voltadas focando o usuário final. Torna-se viável, a partir desta contextualização, a integração de profissionais especializados na busca de interatividade, usabilidade e eficiência na comunicação do usuário com a máquina. Em outras palavras, reconhecer um ambiente adequado de acordo com a área de atuação, conhecimento do usuário e as necessidades provindas da análise de requisitos de um projeto.

Conforme Preece (2005) entende-se *design* de interação a aparência de produtos interativos que fornecem suporte às atividades cotidianas das pessoas, nos diversos ambientes. Seguindo esta linha de raciocínio, é pretendido identificar as necessidades de um compilador da pseudolinguagem PORTUGOL e estabelecer os requisitos para o sistema proposto, desenvolvendo *designers* alternativos que preencham os requisitos levantados durante esta pesquisa.

O objetivo da pesquisa proposta é reconhecer características comuns na elaboração de interfaces com técnicas propícias à relação Homem-Máquina, demonstrar a importância do *design* e da ergonomia na projeção da mesma. Determinar algumas heurísticas como a de viabilidade do status do sistema, compatibilidade do sistema com o mundo real, controle do usuário e liberdade, consistência de padrões, segurança, estética, eficiência, etc.

1.1. Justificativa

Os estudos e trabalhos realizados na área de Design de Interação voltam-se cada vez mais para a elaboração de sistemas satisfatórios, agradáveis, divertidos, interessantes, úteis, motivadores, esteticamente apreciáveis, incentivadores de criatividade (incluindo modelos cognitivos), compensadores e emocionalmente adequados. Essas

características são providas da “experiência do usuário”, que por vezes não são consideradas no levantamento de requisitos ou teste de usabilidade de softwares, negligenciando a necessidade de interação entre o indivíduo e a máquina, dificultando a aceitação da ferramenta final.

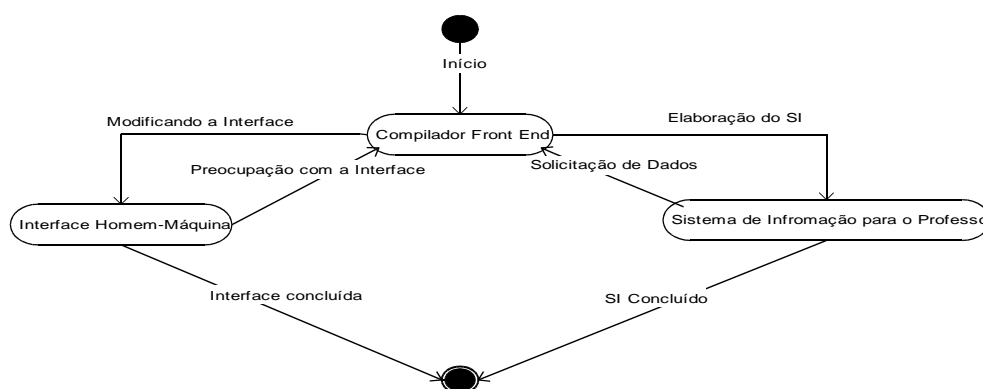
O compromisso com a qualidade e o desempenho de um software requer um bom planejamento, a aceitação do usuário final e sua usabilidade. Atualmente projetar uma Interface atrelada ao usuário é muito importante, embora seja por vezes negligenciado por parte dos projetistas. A Interface é também fator de competitividade respectivo ao produto final. Neste quadro requer a análise de viabilidade, as relações de *feedback*, restrições de uso de funções, o mapeamento de controles, a consistência e *affordances* percebidas do sistema a implementar.

No âmbito da proposta desta pesquisa consentiu-se com a necessidade de integração do estudo e aplicação de técnicas de ergonomia como uma atividade interdisciplinar, que visa o preenchimento de lacunas respectivas ao trabalho final e a variáveis deste enredo, compreendendo o grau de adequação e a maneira com a qual o indivíduo interage com o meio.

1.2.O compilador Front End

No artigo de Coelho et. al. (2005) foi proposto o desenvolvimento de um compilador Front End para o estudo da lógica de programação no contexto da pseudolinguagem Portugol. O desenvolvimento do mesmo foi dividido em três etapas: o compilador Front End, Interface Homem-Máquina e Sistema de Informação.

O primeiro módulo consiste em desenvolver as fases de análise léxica, sintática e semântica do compilador, o segundo módulo trabalha a interface com aplicação de técnicas de design para torná-la uma ferramenta com maior grau de aceitabilidade e o terceiro módulo agrega os conceitos de Sistema de Informação atribuindo ao professor o controle e acompanhamento das atividades realizadas pelos alunos. Para melhor visualização destes módulos veja a Figura 1, que mostra as perspectivas de seus idealizadores.



Figura[1]. Processo de Desenvolvimento do “ComPort” – (Coelho, 2005)

Neste trabalho será realizando o *Segundo Módulo* a partir dos estudos realizados em Interação Homem-Computador (IHC), a aprendizagem cognitiva e as técnicas de design, seguindo da análise e avaliação do protótipo deste compilador educacional atrelados aos conceitos de Ergonomia e de AET (Análise Ergonômica do Trabalho).

O protótipo de compilador Front End escolhido para a realização deste trabalho permitirá que em fase de produção verifique analogias desejadas e que testes como de usabilidade e adaptação sejam realizados. Ainda usará os conceitos de semiótica sobre as cores para tornar a interface mais atraente.

O compilador Front End veiculado para esta pesquisa doravante denominado “ComPort”, este por sua vez tem como semântica da pseudolinguagem semelhante àquela adotada por de Eberspacher e Forbellone (2000), o que permite no âmbito acadêmico estar referenciando os estudos a outros autores no auxílio de estudos autônomos e pesquisas relacionais no contexto programático de cada instituição. Serão desenvolvidas alternativas de interface para o Compilador para a integração com o *Segundo Módulo* pretendido pelos programadores do mesmo, envolvendo-se o estudo da aprendizagem cognitiva. A ferramenta adotada para produção do mesmo foi o Delphi 5, atrelados aos conceitos de orientação a objetos possibilitando a modelagem em UML dos Diagramas de Caso de Uso e de Sequência.

2. Ergonomia

Em ergonomia são consideradas relevantes as características da empresa, da atividade a ser executada por uma repartição ou grupo de pessoas, e finalmente o indivíduo com suas peculiaridades. Na compreensão deste trabalho estas acepções se referem ao ambiente da instituição para qual é projetado o software, aqui se voltando ao ambiente acadêmico. Quando analisada a ótica de grupo em sala de aula formulá-se a necessidade de um melhor aproveitamento de tempo em uma grade curricular de nível técnico ou superior para a aprendizagem de lógica de programação. E referente à pessoa individualizada se correlaciona as características provindas da experiência cotidiana destes, referenciando dados como grau de escolaridade, meio de trabalho, a aquisição de experiências sociotécnicas, recursos corporais atrelados à aprendizagem cognitiva e os meios psíquicos a que ele está sujeito.

As atividades provindas destas acepções conferem em formas distintas dos indivíduos envolvidos realizarem uma mesma tarefa prescrita. Confere ainda como fatores multidisciplinares à posição pessoal, dos meios disponibilizados, das condições as quais estes são submetidos vinculados em um contexto sociotécnico singular que evoluem dinamicamente com impacto na atividade fim. A ergonomia é um mecanismo que se pode utilizar para fim de garantir o bem-estar e performance dos usuários quanto aos objetivos estabelecidos, evidenciando as diferenças de consciências, o compromisso com prescritos para produção, competência profissional e o conhecimento de limites nas possíveis funcionalidades.

A análise ergonômica do Trabalho (AET) em caráter científico permite sistematizar uma metodologia; ver a Figura 2, para agregar a personagem cliente/usuário como uma variável implícita. Nesta sistematização baseada na proposta de Guérin, não desconsiderou os riscos demandados do processo de modelização de teoria, a mesma descreve como evoluíram as atividades do estudo da inter-relação indivíduo-ambiente [Ferreira, 2006].

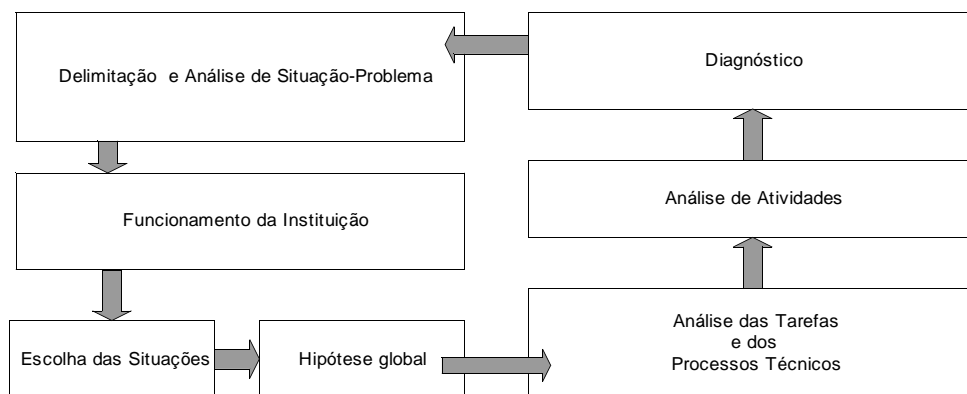


Figura [2]. AET como proposto em Ferreira (2006)

3. Metodologia

Durante a realização desta pesquisa será utilizado dados atuais, com embasamento em teoremas e técnicas propostas por especialistas na área de Design e Ergonomia. Para buscar uma IHC adequada ao usuário/ aluno, será avaliado a interface para um compilador Front End, ferramenta desenvolvida no âmbito educacional que participa de interesses do corpo docente e discente como auxílio na compreensão e ensino da pseudolinguagem Portugol, utilizando as técnicas GOMS (acrônimo de *goals, operators, methods* e *selection rules*) e CARD (*Collaborative Analysis of Requirements and Design*). E no mesmo intuito; de conceber uma interface atrelada às necessidades do usuário, será usado um questionário/ entrevista para investigar o perfil de alunos iniciantes nos estudos de informática e ou computação. Será realizado por meio de entrevistas para a análise das interfaces propostas para o “ComPort” durante e após o desenvolvimento. Correlacionando o produto provindo da pesquisa com projetos, ou melhor, sistemas, que negligenciaram a interação Homem-Máquina.

A estrutura de um software de forma que este não cause atritos com o usuário deve estar vinculada a uma modelagem embasada em técnicas de designers. Tem-se como objetivo, neste ponto, a construção de perspectivas que se estruturam sob as bases dispostas no âmbito educacional como ferramenta de auxílio ao estudante.

Ao ingressar numa instituição para formação na área de computação o estudante iniciado no estudo de programação é comumente “afogado” em dificuldades que atropelam o desempenho na produção de lógica de algoritmo. A ferramenta em desenvolvimento deve ser facilitadora do ensino-aprendizagem, cuja função correspondente é a de analisar os Tokens e apresentar as estruturas léxicas e semânticas da linguagem Portugol, tornando mais proveitoso o tempo dentro da sala de aula e permitindo que o aluno visualize seus erros durante a produção do algoritmo. Atentando para a importância dos fatores motivadores dos estudantes que poderão em futuro próximo fazer uso da mesma para estudos autônomos, individualizando seus parâmetros para sanar as dificuldades deste processo. Este deverá apontar os erros em sua interface particular do Ambiente Didático Integrado exibindo telas com a identificação de símbolos, dinamizando as aulas, circunscrevendo assuntos mais relevantes, diminuindo o tempo de processo para correção de algoritmos, maior aproveitamento das atividades correlacionadas e melhorando a visualização do resultado destas teorias em aspectos de clareza e fins.

Para elaborar uma Interface que dê ao usuário a fácil manipulação e lhe seja ao mesmo tempo atrativo buscou-se aplicar as técnicas GOMS e CARD, envolvendo na estruturação deste ambiente os aspectos da semiótica.

3.1. CARD

CARD; ou *Collaborative Analysis of Requirements and Design*, é uma atividade onde o usuário se torna parceiro da equipe projetista da interface. Constitui numa atividade de Design Participativo. Este método usa cartões para representar as diversas alternativas de ícones ou componente de tela para a ferramenta a ser projetada. O usuário manipula cartões dispostos sobre uma mesa e os correlacionando em menus, botões, atalhos, etc. Durante a atividade o projetista, ou equipe envolvida deve atrelar perguntas de forma que o participante dê nomes aos menus, posições de botões e/ou atalhos, defina as tarefas a serem realizadas e este possa fornecer uma explicação clara sobre suas idéias quanto às referidas colocações. Posterior a esta atividade comparar-se-á as telas fornecidas, e/ou busca a melhor forma de agrupar as informações reunidas [Preece, 2005].

3.2. GOMS

O modelo preditivo é uma forma de avaliação que considera o usuário típico de forma a prever; a partir de heurísticas, problemas de usabilidade. A mais conhecida referente à interação homem-computador é a GOMS.

O modelo GOMS é um acrônimo para *goals* (objetivos), *operators* (operadores), *methods* (métodos) e *selection rules* (regras de seleção), é usado numa tentativa de modelar o conhecimento relacionado aos processos cognitivos envolvidos quando o usuário interage com o sistema: O que se deseja alcançar, a forma como deve ser realizada a tarefa, o que usar e a escolha de métodos. Os dados reunidos através desta técnica são quantitativos [Preece, 2005].

3.3. Uso da semiótica no desenvolvimento da IHC

As técnicas de programação visual controlam características como forma, cor, textura, dimensão, proporção e movimento no intuito de atentar aos objetos da mensagem e ao significado pretendido de acordo com o caráter e as limitações do seu meio de comunicação com o usuário. O objetivo é transmitir a mensagem do designer; emissor, para o usuário, receptor. Novamente o meio deve ser constituído de acordo com o repertório do usuário, tornando a interface consistente. Faz importante reverenciar que as cores influenciam nas emoções dos indivíduos e auxiliam no processo cognitivo [Fenner, 2000]. De acordo com Pedrosa (2005) (referência) a cor é uma sensação causada pela reflexão dos raios luminosos incidentes em um determinado objeto; percebida pelo órgão visão e interpretado pelo cérebro. Com os novos paradigmas da informação faz-se necessário respeitar a individualização no processo de aprendizagem. De forma que para o desenvolvimento deste ambiente se considerará a interdisciplinaridade, a teoria das cores, semiótica, usuário-sistema estabelecendo critérios para o uso de cores para melhor compreensão na veiculação das mensagens.

4. Perfil dos usuários

O questionário aplicado a 40 estudantes da área de informática e computação teve como intuito levantar características do público no âmbito acadêmico, vislumbrando perguntas de nível social, profissional e cultural para aplicação de técnicas ergonômicas e designer de interação. Não obstante ao trabalho realizado conjugam-se as idéias de Eberspacher e Forbellone: “Fundamental à formação do profissional de programação estão a desenvoltura lógica e a construção de algoritmos”. Pois que o intuito na elaboração de uma interface atrativa ao usuário, estudante; é propiciar formas de facilitar a compreensão dos conteúdos programáticos dos cursos que envolvem a programação com uso da pseudolinguagem PORTUGOL e ainda tornar o estudo desta disciplina menos “penosa”. O questionário contou com a participação de alunos da Universidade Presidente Antônio Carlos (UNIPAC) e da Faculdade Ubaense Ozanam Coelho (FAGOC).

Sabido que aspectos como sexo, idade e dimensões corporais e até características como personalidade e vivência na área são “ingredientes” de relevância no que se refere; em especial, a adequação ergonômica para as atividades que os indivíduos devem realizar, buscou-se entre os indivíduos participantes recolher dados deste enfoque. A maior parte dos alunos que ingressam na área são do sexo masculino, e as idades são variantes, dos entrevistados a idade circula dos dezoito aos trinta e cinco anos. A primeira pergunta deste questionário tentou-se evidenciar as necessidades do computador no dia-a-dia, permitindo que cada entrevistado preenchesse esta resposta com as diversas alternativas dispostas. Do uso do computador para estudos, divertimento e trabalho, foram correlacionadas diversas atividades, dentre as quais ficou evidente que grande parte destes fazem uso da máquina para lazer. O computador já faz parte do cotidiano destes indivíduos, mas o uso adequado e qualificado não pode ser apresentado em uma única pergunta, de forma que se fez necessário outras informações.

O uso rotineiro da ferramenta, computador, para estudos diários, no serviço foi especificado através da pergunta relacionada ao tempo de uso do mesmo pelos entrevistados. Através deste questionário/entrevista também foi certificado que todos no âmbito de suas raízes culturais possuíam formas de acesso e que utilizam eventual ou casualmente o computador, quando questionados sobre os hábitos de uso desta ferramenta e foi propiciado aos entrevistados elaborarem um pequeno histórico da relação com o computador, para coletar informações de caráter individual como familiaridade com ambientes diferentes, reconhecimento de hábitos quanto ao uso da internet, valorizando as experiências trazidas por cada um. A relação de casualidade e eventualidade foi justificada pela falta de tempo. Quanto ao Sistema Operacional (SO) utilizado pelos mesmos e o conhecimento dos utilitários destes ambientes, foi verificado que o Windows; ainda sendo usado por todos os entrevistados, não é uma ferramenta da qual se faça uso de todos os recursos propiciados pelo fabricante. Tal fato é verificado quando estes entrevistados afirmam saber o básico ou conhecerem conceitos intermediário deste SO.

Ainda pelo questionário pode-se verificar que 85% dos entrevistados desconhecem o significado da pseudolinguagem Portugol e 95% desconhecem ambientes e técnicas de programação, o que causa certo desconforto em relação à idéia de implementar problemas reais de forma computacional ou aplicando-se a lógica de programação na aprendizagem e construção posterior de algoritmos. Dos 5% de

estudantes que sabem programar, citaram como Linguagens de programação o Pascal, o Delphi e HTML, contudo estes alunos não reconhecem as diferenças existentes nestes ambientes, o que demonstra uma falha de informação, podendo causar certos erros na lógica aplicada aos seus algoritmos. Atenta-se para o fato de Delphi ser um ambiente de programação e não uma linguagem computacional, como sugerido por alguns entrevistados, fato que demonstra a inconsistência na transmissão de informação. Dentre os que sabem programar 16,7% conhecem ou já escreveram algoritmos em PORTUGOL. Perguntou-se aos estudantes qual o grau e relevância do professor para a aprendizagem, estes tiveram como alternativas: moderado, desnecessário e elevada. As respostas obtidas durante este questionamento foram as alternativas: elevada e moderada. Nenhum entre os entrevistados afirmou ser desnecessária a presença do mesmo para o desenvolvimento deste conteúdo. A questão referida ao profissional adequadamente habilitado para exercer a função de educador não é desmerecida por nenhum dos alunos, nem mesmo por aqueles que se consideram autodidatas. Ainda estes ressaltaram importância da presença do professor, podendo então ser afirmado que esta ferramenta será como desejado apenas um instrumento auxiliar para as disciplinas de introdução a programação.

5. Resultados

Levando em consideração o resultado do levantamento de perfil dos usuários aos quais se destina este projeto, optou-se por utilizar na formulação deste novo ambiente metáforas de interface combinado às analogias; já propostas pelo SO Windows, ao contexto da disciplina atrelada a pseudolinguagem Portugol. É pretendido tornar o trabalho de manipulação mais fácil e proporcionar uma aprendizagem cognitiva, podendo o usuário fazer referência dos conceitos então abrangidos no uso desta ferramenta em outros ambientes computacionais.

O estudo e aplicação da ergonomia; no contexto proposto, permitirá agregar as necessidades e/ou melhorias ao ambiente do protótipo analisado, tendo este o intuito de adaptar o mesmo buscando vinculá-lo ao quadro educacional de instituições privadas, públicas e/ou ainda em estudos autônomos de elaboração lógica de algoritmos na pseudolinguagem Portugol.

5.1. Interações do Usuário com o “ComPort”

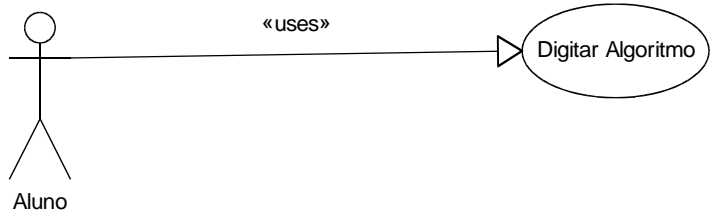
Aos alunos do 1º período de Ciência da Computação da UNIPAC, em laboratório disponibilizado pela universidade, aplicou-se uma atividade com fins de avaliar a interatividade do protótipo “ComPort”, na qual os alunos deveriam produzir um algoritmo utilizando os conhecimentos de Portugol administrados até a respectiva data.

Durante a aplicação desta atividade foi verificado um grande interesse pela referida turma em relação à proposta e a oportunidade de vislumbrar de forma mais plausível a aplicação do conhecimento adquirido e a lógica referenciada pelo compilador. O mesmo atentava-os para erros de sintaxe e dúvidas adivinham da nova experiência solucionando déficits de aprendizagem que puderam ser sanados pela presença do professor que mais uma vez se faz importante no aprendizado.

5.2. Estudos referentes aos fluxos do “ComPort”

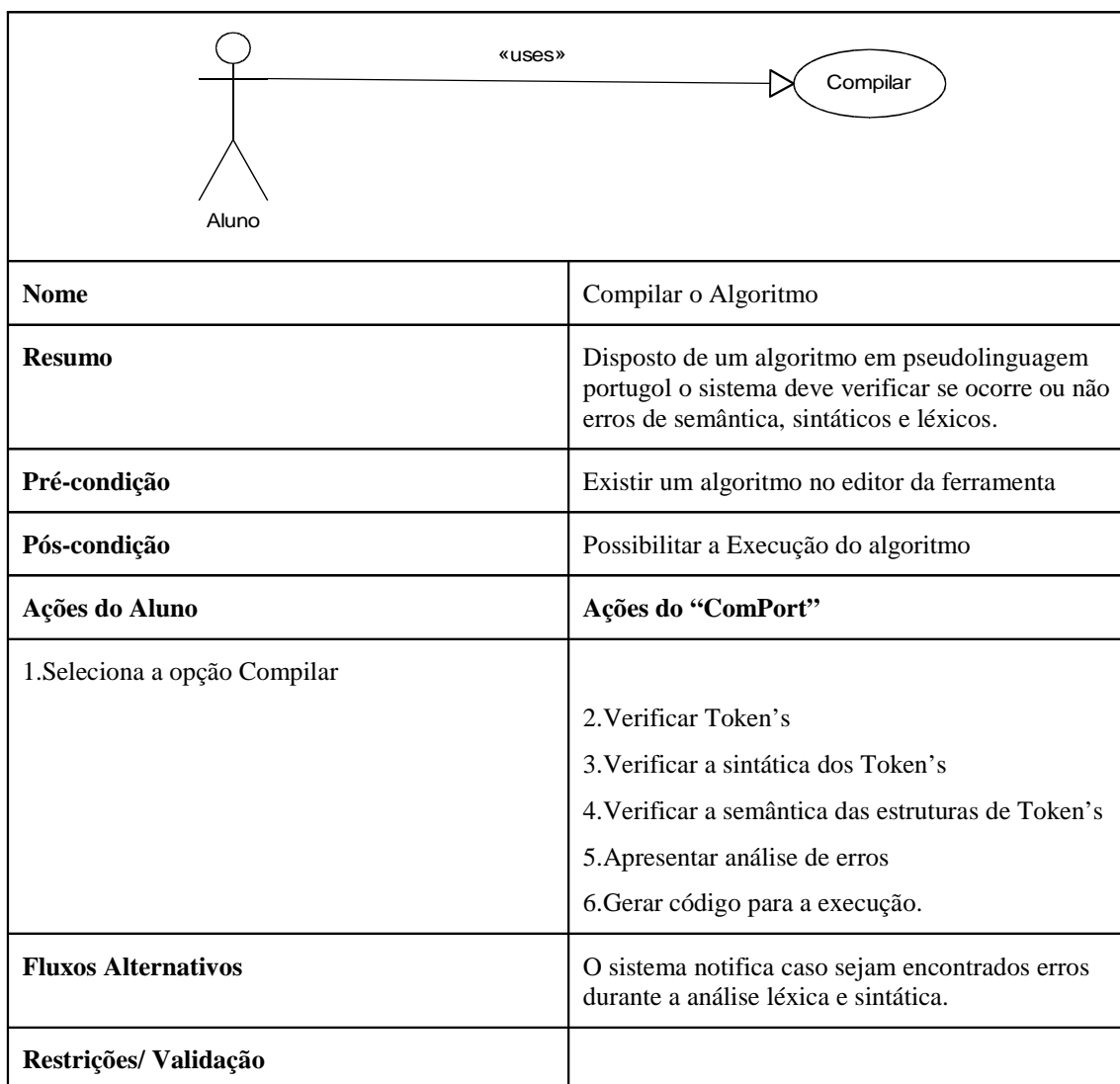
Após o levantamento de requisitos conclui-se que as atividades que o sistema deve propiciar serão três: digitar, compilar e executar o algoritmo. Utilizando os conceitos da UML (Linguagem de Modelagem Unificada) e seus diagramas; foram elaborados os diagramas de caso de uso e seqüência, os quais demonstram a interação do usuário com o sistema.

Para o caso de uso referente a digitar o algoritmo o sistema deve permitir que o usuário manipule o editor de texto fazendo uso das funções dispostas no menu Utilitários (copiar, colar, recortar e desfazer). Não havendo restrições anteriores para que este seja feito permitirá que o aluno/usuário solicite um arquivo novo, digite o texto e salve o algoritmo. Após a realização deste o mesmo poderá compilar seu algoritmo. Estes dados estão dispostos no Quadro 1.

	
Nome	Digitar o algoritmo
Resumo	Digitar um algoritmo no editor de texto da ferramenta em pseudolinguagem Portugol ministrada pelo professor no conteúdo da disciplina
Pré-condição	
Pós-condição	
Ações do Aluno	Ações do “ComPort”
1.Seleciona a opção NOVO 2.Digita o algoritmo 4.Salvar o algoritmo em arquivo *.pas	3.Mostrar a posição cursor no texto, em linha e coluna, como uma matriz.
Fluxos Alternativos	Abrir um arquivo já existente para a alteração
Restrições/Validação	

Quadro [1]. Caso de Uso: Cenário Digitar Algoritmo

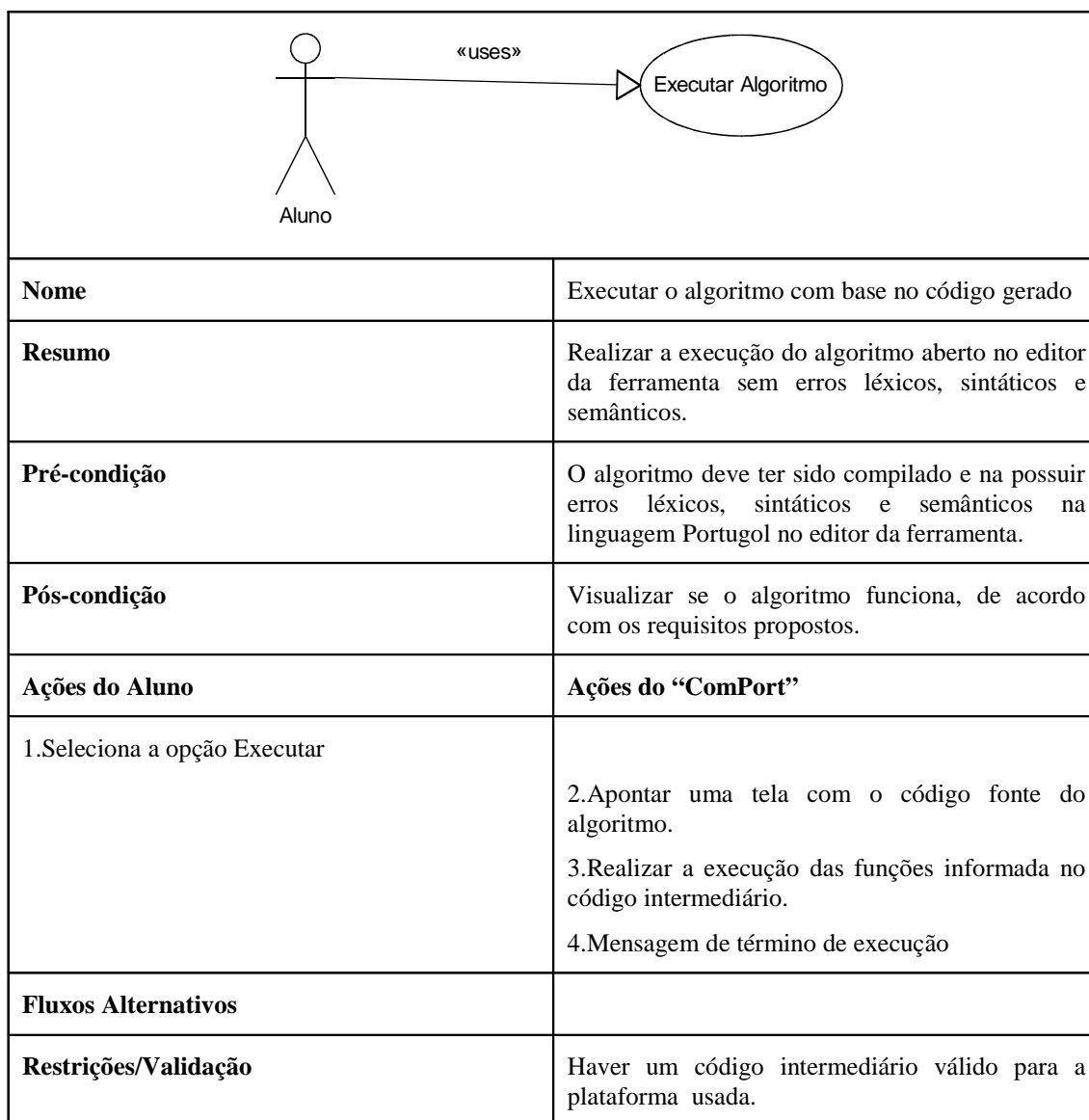
Para o segundo cenário, compilar o algoritmo, é necessário que um algoritmo esteja aberto no editor do “ComPort” ou que o usuário tenha previamente digitado, escrito em pseudolinguagem Portugol. Através deste fluxo o usuário terá disposto em telas paralelas a identificação dos Tokens e posições onde são encontrados na tela. O que se verifica no Quadro 2 é que este caso permite a geração de código intermediário para a sua posterior execução, esta última apenas será executada se o mesmo não apresentar erros semânticos, sintáticos e léxicos.



Quadro [2]. Caso de Uso: Cenário compilar algoritmo

No Quadro 3 apresenta-se o caso de uso do cenário executar algoritmo, no qual nota-se que é necessário a compilação do algoritmo e a criação de um código intermediário para a execução através da ferramenta. Com a execução do código os erros serão apontados, e caso este funcione adequadamente ele envia uma mensagem ao usuário informando que a tarefa foi realizada com sucesso.

“Programar é um processo complicado e, como é feito por seres humanos, freqüentemente conduz a erros. Por mero capricho, erros em programas são chamados de **bugs** e o processo de encontrá-los e corrigi-los é chamado de **depuração** (*debugging*)” [Soares, 2006]. Como os erros de sintaxe e léxicos, também existem os erros que ocorrem em tempo de execução (*runtime errors*), quando se faz a distinção destes torna-se mais rápida a sua localização. Durante a execução ao através de uma ferramenta específica deve-se verificar estes erros. No “ComPort” esta função se encontra no menu “Utilitários” de forma modular, mas ainda não foi implementado. Por este fato não incluído no estudo de caso de uso, ficando aqui o comentário como uma observação para análise correta da ferramenta e dos dados informados a seguir.



Quadro [3]. Caso de Uso: Cenário Executar o algoritmo

Para visualizar o fluxo de dados no sistema “ComPort” foi desenhado o Diagrama de Seqüência o qual se referi a ação do usuário acionar o dispositivo compilador, do evento “onClick” da opção “compilar” no menu “Utilitários”. O mesmo seguiu na Figura 3 identificada como uma mensagem que instancia outros objetos e denominada onClickCompilar2. A escolha desta representação é em razão da sua importância diante do que se propõe o programa e pelo número de fluxos instanciando diferentes objetos das várias classes existentes.

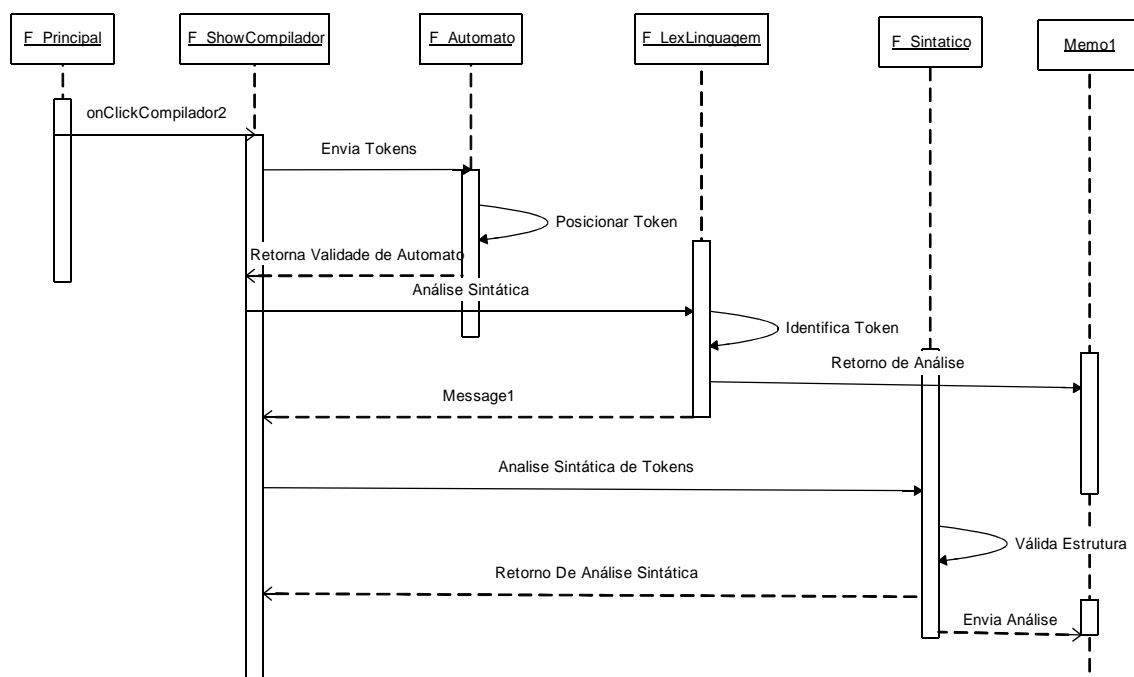


Figura [3]. Diagrama de Seqüência para compilar o algoritmo

5.3. Alterações propostas

É importante salientar que alguns componentes e uma nova modelagem se fazem necessárias para melhor visibilidade, atratividade e adequar ao âmbito da aprendizagem cognitiva. A partir dos estudos realizados optou-se por elaborar uma interface híbrida de modelagem conceitual para o compilador em análise; ou seja, tornar possível mudar pelas técnicas de designer e ergonomia adotadas que o usuário desta ferramenta interaja, manipule e navegue pelo ambiente realizando atividades fins sem desorientação, com confiabilidade e segurança.

Uma dentre as propostas sugeridas pelo estudo é tornar mais visível a principais funções do sistema, controlando o número de informações dispostas para o usuário na tela principal para não desviar sua atenção das tarefas a serem executadas, nem causar uma “poluição” visual. Trazendo desta forma para a “barra de ferramenta” do “ComPort” um atalho para a função “Compilar”, até então escondida no menu “Utilitários” e retirando o botão “Fechar” aproximando a interface de ambientes comumente utilizados, como Microsoft Word, e o ambientes que surgirão no âmbito dos estudos de programação, tal qual o Delphi. A Figura 4 mostra as mudanças propostas.



Figura [4]. O “ComPort” anterior e as mudanças sugeridas

Na Figura 5, é apresentada a importância de se agregar ao texto figuras ilustrativas de forma a tornar mais clara a função a ser executada por cada botão e/ou

atalho construindo desta forma uma interface cognitiva que assegura que o usuário lembrará de qual ação deve ser selecionada em um “Menu”. É proposto o uso de teclas de atalhos para as diversas funções da ferramenta, permitindo que o usuário mais habilidoso tenha alternativa, optando pela forma que melhor convier para ativar as funções.

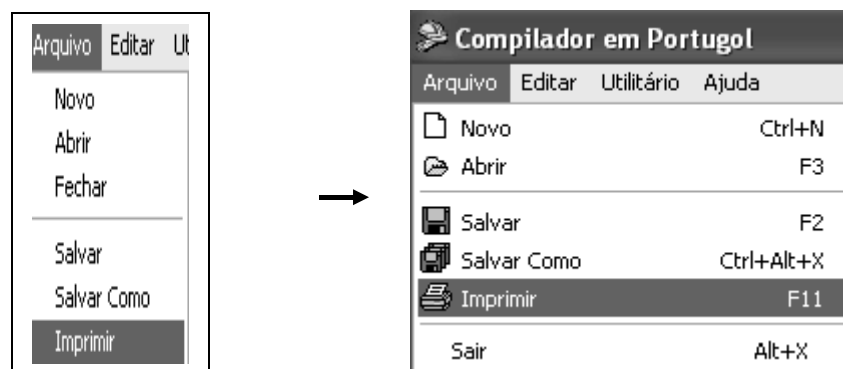


Figura [5]. Menu Arquivo “ComPort” anterior aos estudos realizados seguido das alterações

6. Conclusão

Ao fim desta pesquisa, nota-se mudar que apesar das dificuldades implícitas no envolvimento do usuário na modelagem de uma interface, o mesmo é um método que tem como benefícios claros o custo no que dispensará em remodelagem, correção de erros por falhas em levantamentos de requisitos, terá maior aceitabilidade e permitirá um menor gasto com treinamento de usuários, dentre outros inúmeros benefícios que são hoje a maior razão de credibilidade e sucesso de um software.

Quanto à referida aprendizagem da pseudolinguagem Portugol vislumbra-se uma grande perspectiva ao desenvolvimento do contexto educacional da ferramenta. Fica aqui como sugestões para trabalhos futuros a produção do compilador para o ambiente Linux, o desenvolvimento do Sistema de Informação para a interação de alunos e professores, a conclusão do módulo de autômato e do módulo de “Debugar” (tratamento do algoritmo durante a execução do código gerado pelo compilador) do “ComPort”.

Referências

- Dazzi, R.L.S., Santiago, R. (2006). “Interpretador de Portugol”, http://www.niee.ufrgs.br/cbcomp/cbcomp2004/html/pdf/Algoritmos/t170100165_3.pdf, abril.
- Eberspacher, H.F., Forbellone, A.L.V. (2000). “Lógica de Programação: A construção de algoritmos e estruturas de dados”, MAKRON Books.
- Esmin, A. A. A., (1998). “Portugol/Plus: Uma ferramenta de apoio ao ensino de Lógica de Programação baseado no Portugol”, <http://www.niee.ufrgs.br/ribie98/TRABALHOS/118.PDF>, março.
- Fenner, R.C. (2000). “Contribuições do Design na produção de software educacional”, <http://teses.eps.ufsc.br/defesa/pdf/3088.pdf>, dezembro.

- Ferreira, M.C.(2006). “O Sujeito forja o ambiente, o ambiente “forja” o sujeito: Inter-Relação Indivíduo-ambiente em ergonomia da atividade”, <http://www.unb.br/ip/labergo/sitenovo/mariocesar/artigos3/OSujeito.PDF>, fevereiro.
- Issing, L.J. (2005). “Conceitos básicos de Didática para Multimídia” (in portuguese), <http://penta.ufrgs.br/edu/teleduc/tdimult.htm>, novembro.
- Martins, J.,Vargas, K.S. (2004). “Ferramenta para Apoio ao Ensino de Introdução à Programação”, <http://www.inf.furb.br/seminco/2005/artigos/107-vf.pdf>, março.
- Oliveira Netto, A.A.(2004), “IHC – Interação Homem Computador: Modelagem e Gerência de Interfaces com o Usuário”, VisualBooks.
- Coelho, F., Cândido, A., Hermano, G., Padilha, J. (2005). “Um compilador Front End para o estudo da Lógica de Programação: Pseudolinguagem PORTUGOL”, II Simpósio Mineiro de Sistema de Informação – SMSI 2005, outubro.
- Pedrosa, T.M.C e Toutain, L.B.(2005). “O uso das cores como informação na Interfaces digitais”, http://www.cinform.ufba.br/vi_anais/docs/TaisPedrosaLidiaToutain.pdf, dezembro.
- Perry, G.T., Neto, A. Amaral, F.G. (2004). “Integrando Conhecimentos, Aproximando disciplinas: A importância do Design e a Ergonomia no Projeto e no Desenvolvimento de Softwares Educacionais”, http://www.cinted.ufrgs.br/renote/maio2005/artigos/a18_designergonomia.pdf, novembro.
- Preece,J., Rogers, Y., Sharp., H. (2005). “Design de Interação: Além da interação homem-computador”, Bookman.
- Santos, N. (2005) “Interface de Ambientes Educacionais: Diretrizes de Projeto”, <http://www.ime.uerj.br/~neide/Interfaces.htm>, novembro.
- Soares, C.F.B (2005). “Como Pensar como um Cientista de Computação”, <http://pensarpython.incubadora.fapesp.br/portal/documentos/Como%20Pensar%20como%20um%20Cientista%20da%20Computacao.pdf>, dezembro.
- Sociedade Brasileira de Informação-SBC. (2006). “Modelo para submeter artigos”<http://www.sbc.org.br>, junho.