

WEBIS: LINGUAGEM E AMBIENTE WEB ACESSÍVEL DE PROGRAMAÇÃO

Lucas Pereira da Silva

Resumo: As dificuldades encontradas no aprendizado de uma primeira linguagem de programação estão relacionadas com a própria linguagem que é objeto de estudo e também com o ambiente de programação a ser utilizado. Boa parte das linguagens não possuem regras sintáticas e semânticas simples e tão pouco oferecem um ambiente de programação simplificado e intuitivo. Isso ocorre, pois na maioria dos casos, as linguagens de programação são desenvolvidas com um foco profissional e não de aprendizado [1]. Telis é uma linguagem e um ambiente de programação que busca resolver esse problema através de uma sintaxe simples e de uma abordagem que traz de forma natural na linguagem conceitos avançados de linguagens de programação. Entretanto, Telis não resolve o problema de pessoas com deficiência visual e que dependem da utilização de leitores de tela. Este trabalho terá o objetivo de unir as principais características de Telis com uma acessibilidade mais adequada e que permita que pessoas com deficiência visual possam ter um maior poder de expressão. Essa junção resultará no desenvolvimento de uma nova linguagem e de um novo ambiente de programação *Web*, chamados de Webis. **Palavras-chave:** linguagens de programação, Web, acessibilidade, pessoas com deficiência visual.

1. INTRODUÇÃO

As dificuldades encontradas no aprendizado de uma primeira linguagem de programação estão relacionadas com a própria linguagem que é objeto de estudo e também com o ambiente de programação a ser utilizado. Boa parte das linguagens não possuem regras sintáticas e semânticas simples e tão pouco oferecem um ambiente de programação

simplificado e intuitivo. Isso ocorre, pois na maioria dos casos, as linguagens de programação são desenvolvidas com um foco profissional e não de aprendizado [1]. Em decorrência desse fato, o primeiro contato com uma linguagem de programação pode ser muitas vezes traumático, fazendo com que o programador iniciante perca o interesse.

A linguagem Telis, desenvolvida no Laboratório de Software Educacional (Edugraf), possui regras sintáticas e semânticas simples que podem ser facilmente assimiladas. Telis também é o nome do ambiente de programação oferecido para a linguagem. O presente projeto consistirá do desenvolvimento de uma linguagem baseada em Telis e de uma máquina de execução escrita diretamente em JavaScript. O projeto também deverá ter grande foco no ambiente de programação que deverá ser executado nativamente por navegadores Web. O novo ambiente e a nova linguagem de programação a serem desenvolvidos serão chamados doravante de Webis.

2. DEFINIÇÃO DA LINGUAGEM BASEADA EM TELIS

A linguagem Telis que vem sendo desenvolvida desde 1998 [2] pelo Edugraf, permite, através de construções sintáticas e semânticas simples, que o programador tenha um grande poder de expressão e possa expor de forma natural conceitos avançados como orientação a objetos, paralelismo, programação distribuída, programação de eventos e outros.

A linguagem Webis será criada a partir da linguagem Telis. As construções léxicas e sintáticas da nova linguagem serão muito semelhantes às mesmas existentes em Telis. A diferença entre as duas linguagens estará principalmente relacionada com os novos requisitos da linguagem Webis. Por exemplo, como Webis deverá ter suporte à manipulação de sons 3D, será necessário que esses recursos sejam incorporados à nova linguagem.

Um aplique Webis em execução será composto por um sistema de coordenadas cartesianas onde atores estarão distribuídos. Uma das características fundamentais do Webis será permitir que os atores possuam, além da representação gráfica, uma representação sonora equivalente. Com isso, sons poderão ser associados aos atores e sempre que um ator se movimentar no sistema de coordenadas, o sistema de som deverá refletir a

mudança de modo que seja possível perceber sonoramente a movimentação do ator. Ou seja, as primitivas de posicionamento e movimentação como `frente`, `direita`, `andarPara` e outras, deverão provocar mudanças tanto no sistema gráfico quanto no sonoro.

O Webis disponibilizará algumas primitivas sonoras que permitirão o ator manipular sons. A primitiva `fixarSom` permitirá associar um som a um ator. Quando executada, essa primitiva obterá um parâmetro na pilha que indicará o endereço de um som que deverá ser associado ao ator. Uma característica importante é que ao ator executar a primitiva, o som começará a ser tocado e a execução do ator seguirá normalmente, ou seja, o ator não será bloqueado. Após a finalização do som, o ator ficará em silêncio até que outro som seja fixado. Caso o ator deseje que o som seja tocado repetidamente, então bastará utilizar a primitiva `fixarSomContínuo`.

Por padrão, um ator iniciará em silêncio e mesmo que um som seja fixado, o ator permanecerá em silêncio. Para que um som fixado seja tocado será necessário que o ator utilize a primitiva `comSom`. Essa primitiva fará com o ator passe a emitir sons, caso possua algum associado a ele. Existirá também a primitiva `semSom` que terá o objetivo de silenciar o ator. Se o ator estiver com um som fixado, então a utilização da primitiva `semSom` fará com que o ator pare imediatamente tocar o som. Já se o ator não estiver com nenhum som fixado, então a primitiva `semSom` não terá nenhum efeito imediato, porém caso o ator execute futuramente a primitiva `fixarSom` ou `fixarSomContínuo`, então o som fixado não será tocado.

3. HTML5 E JAVASCRIPT

A *Web* é baseada em três importantes pilares: o Identificador Uniforme de Recursos (URI), o Protocolo de Transferência de Hipertexto (HTTP) e a Linguagem de Marcação de Hipertexto (HTML) [3]. Neste projeto existirá um grande foco no terceiro pilar, em especial na mais nova versão do (HTML), o (HTML5).

O HTML5 não é apenas uma nova especificação para uma linguagem de marcação, mas sim um conjunto completo de tecnologias com novas funcionalidades. O objetivo das novas especificações é fazer da *Web*

uma plataforma completa e aberta, permitindo que as tradicionais aplicações *desktop* possam ser inteiramente convertidas em aplicações *Web*. A vantagem disso é a existência de um padrão aberto, onde uma aplicação poderá ser desenvolvida independentemente do dispositivo em que será executada. Tudo que o dispositivo precisará é de um navegador *Web* com suporte ao HTML5.

Dentre as principais especificações incorporadas ao HTML5, destacam-se a Folha de Estilos em Cascata 3 (CSS3) e as novas APIs do JavaScript, como a *Web Audio API* e a especificação de *Web Workers*. O JavaScript terá um papel fundamental no desenvolvimento da máquina Webis. Nesse capítulo serão discutidos conceitos importantes do JavaScript e do HTML5 para que se possa compreender melhor os mecanismos utilizados pela máquina Webis.

4. MÁQUINA WEBIS

A máquina Webis será responsável por executar os programas Webis. A máquina será escrita em JavaScript e seu ambiente de execução deverá ser em um navegador *Web*. Um programa escrito na linguagem Webis será, através do processo de geração de código, traduzido para uma lista de comandos JavaScript. A máquina Webis terá o objetivo de fornecer um ambiente de execução de modo que a entrada desse ambiente seja uma lista de comandos JavaScript que representa um programa Webis.

As etapas de análise léxica, sintática e geração do código do Webis serão realizadas através da ferramenta ANTLR. O ANTLR é um gerador de analisador léxico e sintático que pode ser utilizado para a criação de interpretadores de linguagens, compiladores e outros tradutores ([4]). No contexto desse projeto, o ANTLR será utilizado para gerar um tradutor que transformará um programa codificado na linguagem Webis em um código JavaScript. Nesse sentido, o ANTLR atuará como um *compiler-compiler*, e o código JavaScript será, então, executado pela máquina Webis. Um *compiler-compiler* é uma ferramenta que permite criar tradutores a partir de uma descrição formal de uma gramática.

A máquina Webis trata-se basicamente de uma máquina de pilha.

Nela os parâmetros necessários para as operações são desempilhados, enquanto que os resultados das operações são empilhados. A entrada da máquina Webis nada mais é do que uma lista que contém as instruções do programa, sendo que cada uma dessas instruções é um elemento JavaScript que saberá se executar. O sistema de execução da máquina Webis, portanto, se torna muito simples em uma primeira análise. De forma simplificada, o que o núcleo da máquina Webis faz é enviar a ordem de execução para cada uma das instruções da lista recebida como entrada. Além da ordem de execução, o sistema de execução da máquina Webis fornece à instrução a ser executada o contexto do programa. Desta forma, cada instrução terá acesso à pilha de dados e ao escopo de variáveis, podendo empilhar e desempilhar elementos da pilha de dados e alterar, remover e criar variáveis.

A máquina Webis conterá um componente denominado ecossistema. O ecossistema será o responsável por gerenciar modelos, atores e qualquer interação desde último com a página HTML ou com elementos que existam apenas no fluxo principal. Um programa Webis consistirá de uma página HTML que conterá um *script* com o ecossistema. O ecossistema assumirá aqui o papel do *script* que executa no fluxo principal. Por isso, quando for falado em execução do ecossistema estará implícito que a execução é no fluxo principal. De forma análoga, um ator executará em um *Worker* e, por isso, quando um ator for mencionado estará implícito que a execução deste é em um *Web Worker*.

5. AMBIENTE WEBIS

Além da linguagem de programação e do ambiente de execução, o Webis também será composto por um ambiente de programação. Assim como o ambiente de execução, o ambiente de programação também deverá ser inteiramente disponível através de um navegador *Web*. Para o desenvolvimento do ambiente de programação será utilizado o HTML5 em conjunto com algumas das suas mais recentes funcionalidades.

Ao acessar o Webis, qualquer usuário, estando autenticado ou não, poderá ter acesso aos aplicativos criados por outros usuários. Será permitido que qualquer usuário visualize e execute qualquer aplicativo. No

ambiente Webis será possível visualizar a listagem de todos os apliques existentes. Por padrão, os apliques são ordenados de acordo com a data da última modificação. Através dessa ordenação, os usuário poderão acompanhar os mais recentes programas Webis que estão em desenvolvimento.

O ambiente Webis permitirá que os usuários carreguem sons no sistema. Dessa forma, os sons carregados poderão ser utilizados nos programas Webis. Assim como ocorre com os apliques, os sons também poderão ser utilizados por todos usuários. O Webis disponibilizará uma listagem com os sons carregados. Para utilizar um determinado som em um programa Webis, basta que o usuário forneça a URI do som como parâmetro das primitivas `fixarSom` e `fixarSomContínuo`. Usuários autenticados terão acesso à página **Meus sons**, onde todos os sons carregados pelo usuário serão listados.

O Webis possui uma abordagem voltada para permitir a criação de um ambiente colaborativo de desenvolvimento, onde programadores possam desenvolver e compartilhar soluções. É por esse motivo que os programas Webis poderão ser visualizados por qualquer pessoa. Mais do que isso, programas Webis poderão ser reutilizados e modificados para permitir que programadores possam trabalhar em conjunto para resolver um determinado problema. Qualquer programa Webis, criado através do ambiente, será implicitamente livre. O livre compartilhamento dos programas Webis traz algumas vantagens:

6. UTILIZAÇÃO DO WEBIS NA ACIC

O Webis foi utilizado no Curso de Criação de Páginas Acessíveis e Programação *Web*, ministrado na Associação Catarinense para Integração do Cego (ACIC). O curso foi ministrado por Pablo Valério Polônia, mestre em Engenharia de Automação e Sistemas pela Universidade Federal de Santa Catarina (UFSC) e contou com a participação de cinco alunos, sendo quatro deles cegos e um com baixa visão. O curso foi dividido em duas partes, sendo a primeira parte voltada para a criação de páginas *Web* acessíveis com (HTML) e (CSS), e a segunda parte focada no desenvolvimento de

programas *Web* através da linguagem PHP. No contexto desse curso, o Webis foi utilizado com a finalidade de preparar os alunos com noções básicas de programação. Seu objetivo foi de servir como suporte para a segunda etapa do curso.

A utilização do Webis no curso foi voltada à apresentar os conceitos básicos existentes nas linguagens de programação, como estruturas de controle de fluxo, laços de repetição, estruturas de dados e outros. Percebeu-se no decorrer das aulas com Webis que a sintaxe, bastante simples, facilitou os alunos na hora de escreverem seus programas. Essa facilidade ficou bastante clara ao analisar as dificuldades encontradas pelos alunos em relação às sintaxes do HTML e do PHP. O principal problema enfrentado pelos alunos com a sintaxe do Webis foi em relação às acentuações. As primitivas do Webis são acentuadas e em algumas ocasiões os alunos esqueceram de utilizar os acentos. A primitiva `fixarSomContínuo`, por exemplo, era escrita algumas vezes pelos alunos como `fixarSomContínuo`, gerando, assim, erros de execução.

7. CONCLUSÃO

A implantação do Webis através da linguagem JavaScript executando em um navegador *Web* se mostrou bastante plausível. Graças às novas funcionalidades provenientes do HTML5, como *Web Workers* e *Web Audio API*, foi possível implementar adequadamente o modelo de execução e o sistema sonoro do Webis. A implementação foi dificultada em alguns aspectos devido às diferenças entre o modelo de execução do Webis e do JavaScript, entretanto essas dificuldades foram contornadas com estratégias apropriadas para a modelagem da arquitetura da máquina Webis. Se por um lado existiram dificuldades para a implementação do Webis devido ao uso do JavaScript, por outro lado o modelo de orientação a objetos baseados em protótipos unido com os aspectos funcionais da linguagem facilitaram bastante o processo de programação.

O sistema de sons do Webis também se mostrou viável para permitir que pessoas com deficiência visual possam, além de executar programas mais acessíveis, desenvolver seus próprios programas.

O Webis ainda pode evoluir bastante no sentido de facilitar o aprendizado de programadores iniciantes e de se tornar uma plataforma mais completa de desenvolvimento. Isso abre a possibilidade de trabalhos futuros virem a melhorar algumas características do Webis e trazerem novas funcionalidades. A ampliação do sistema gráfico e sonoro, a comunicação direta entre atores, o servidor de estímulos para permitir que apliques conversem entre si, a adição de novas funcionalidades no editor de programas e no ambiente de execução e a adição de novas primitivas são apenas algumas das melhorias que podem ser adicionadas a eventuais futuras novas versões do Webis. O Webis desenvolvido através deste trabalho possibilitou a criação de uma base sólida tanto para o ambiente de programação quanto para o ambiente de execução. Entretanto, melhorias e modificações são necessárias para permitir a criação de uma ambiente mais completo de desenvolvimento, abrindo, assim, a possibilidade de trabalhos futuros virem a contemplar essas melhorias e modificações.

8. REFERÊNCIAS BIBLIOGRÁFICAS

- [1] GOMES, Anabela; MENDES, A. J. **Learning to program**: difficulties and solutions. International Conference on Engineering Education, Coimbra, Portugal, 2007.
- [2] PIERI, Giovani. **Projeto e implementação de uma linguagem de programação**. Curso de Ciências da Computação, Departamento de Informática e Estatística, Universidade Federal de Santa Catarina, Florianópolis, Santa Catarina, Brasil, 2007.
- [3] FERREIRA, Elcio; EIS, Diego. **HTML5**. W3C escritório Brasil, 2010. Disponível em: <http://www.w3c.br/pub/Cursos/CursoHTML5/html5-web.pdf>. Acesso em: maio de 2013.
- [4] PARR, Terence. **The definitive ANTLR reference**: Building domain-specific languages. The Pragmatic Bookshelf, 2007. 358 p.