

Tradutores

Berbardo Botelho¹, Yuri Silva Paludo¹

¹Bacharelado em Ciência da Computação – Universidade do Vale do Rio dos Sinos (UFRGS)
Caixa Postal 15.064 – 91.501-970 – Porto Alegre – RS – Brazil

{yuris.plus,botelho.bernardo}@gmail.com

Abstract. *JavaScript is widely used in software development, specially for web-based application. However, it allows for a number of constructions that can generate non uniform code, that, in some cases, becomes hard to read and maintain. This work proposes a uniformization for the language by the process of improving language definitions and using only a safe subset of JavaScript.*

Resumo. *JavaScript é umas das linguagens mais utilizadas no desenvolvimento de sistemas para a plataforma web, entretanto, a sua falta uniformização permite que o desenvolvimento não seja uniforme e em alguns casos temos um código que pode se tornar ilegível. Este trabalho propõe uma uniformização de um subconjunto da linguagem JavaScript.*

1. Introdução

Este trabalho tem como objetivo a criação de um tradutor para a disciplina de Tradutores na Universidade do Vale do Rio dos Sinos no semestre de 2014/1.

Apesar da linguagem JavaScript ser largamente utilizada, não há uma forma única e uniforme de desenvolvimento. Vamos apresentar neste trabalho uma uniformização de um subconjunto da linguagem JavaScript. O objetivo principal desta uniformização é seguir as boas práticas de desenvolvimento de software, e para isso o tradutor a ser desenvolvido deve realizar as seguintes tarefas:

- Definir espaçamento na atribuição;
- Definir como o código deve ser formatado;
- Obrigatoriedade que as linhas de comando terminal em ";"
- Detecção de variáveis declaradas e não utilizadas;
- Métodos que não são chamados;

Nas sessões seguintes serão apresentadas respectivamente a linguagens de origem e a linguagem alvo, após isso teremos uma sessão dedicada aos exemplos, e para concluir a última sessão apresenta a ferramenta que será utilizada para o desenvolvimento deste trabalho.

2. Linguagem Origem

A linguagem de origem será JavaScript.

3. Linguagem Alvo

A linguagem alvo também é JavaScript, entretanto deve apresentar um JavaScript uniforme, conforme as considerações feita na sessão de introdução.

4. Exemplos

Nesta sessão vamos apresentar dois exemplos em JavaScript que mostram de forma sucinta o comportamento esperado do tradutor que será desenvolvido.

Na Figura 1 e 2 podemos ver a comparação de dois códigos em JavaScript, ambos os códigos funcionam normalmente, entretanto o apresentado pela Figura 2 é mais legível e segue algumas boas normas de desenvolvimento.

Figura 1. Exemplo 1 - Código em JavaScript não formatado

```
function myFunction()  
{  
document.getElementById("demo").innerHTML="My First JavaScript Function"  
}
```

Figura 2. Exemplo 1 - Código em JavaScript formatado

```
function myFunction()  
{  
    document.getElementById("demo").innerHTML = "My First JavaScript Function";  
}
```

Nas Figuras 3 e 4 temos novamente a comparação entre dois códigos que funcionam, porém o apresentado na Figura 4 apresenta indentação, espaçamento nas atribuições e outras características que o tornam o código mais legível.

Figura 3. Exemplo 2 - Código em JavaScript não formatado

```
function startTime()  
{  
today=new Date()  
h=today.getHours()  
m=today.getMinutes()  
s=today.getSeconds()  
m=checkTime(m);  
s=checkTime(s);  
document.getElementById('txt').innerHTML=h+": "+m+": "+s;  
t=setTimeout(function(){startTime()},500);  
}  
  
function checkTime(i)  
{  
if(i<10){  
i="0" + i;  
}  
return i;  
}
```

Figura 4. Exemplo 2 - Código em JavaScript formatado

```
function startTime()
{
    var today = new Date();
    var h = today.getHours();
    var m = today.getMinutes();
    var s = today.getSeconds();
    m = checkTime(m);
    s = checkTime(s);
    document.getElementById('txt').innerHTML=h+":"+m+":"+s;
    t = setTimeout(function() {startTime()},500);
}

function checkTime(i)
{
    if (i < 10 )
    {
        i = "0" + i;
    }

    return i;
}
```

Os exemplos utilizados nesta sessão foram extraídos de [W3Schools 2014].

5. Ferramenta utilizada

Para o desenvolvimento deste trabalho vamos utilizar a ferramenta *ANTLR* (*ANother Tool for Language Recognition*). O *ANTLR* é uma ferramenta altamente utilizada no mercado e atualmente está na versão 4.2.2.

O *ANTLR* foi escolhido pois com ele é possível realizar a análise léxica e sintática utilizando a mesma ferramenta. Além disso, apresenta uma arquitetura de funcionamento modular e amplamente documentada [ANTLR 2014].

Referências

ANTLR (2014). *ANTLR Web Page*.

W3Schools (2014). *W3 Schools JavaScript Reference*.