



CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA DE MINAS GERAIS
DEPARTAMENTO DE COMPUTAÇÃO
CURSO DE ENGENHARIA DE COMPUTAÇÃO

PRÉ-PROJETO DE TRABALHO DE CONCLUSÃO DE CURSO:
PROPOSTA DE UMA MÉTRICA DE QUALIDADE DE CÓDIGO CSS
VISANDO A MANUTENIBILIDADE

VICTOR CARNEIRO SALVADOR

Orientador: Prof. Flávio Roberto dos Santos Coutinho
Centro Federal de Educação Tecnológica de Minas Gerais – CEFET-MG

BELO HORIZONTE
MARÇO DE 2015

VICTOR CARNEIRO SALVADOR

**PRÉ-PROJETO DE TRABALHO DE CONCLUSÃO DE CURSO:
PROPOSTA DE UMA MÉTRICA DE QUALIDADE DE CÓDIGO CSS
VISANDO A MANUTENIBILIDADE**

Pré-Projeto de Monografia apresentado ao curso de Engenharia de Computação do Centro Federal de Educação Tecnológica de Minas Gerais, como requisito parcial para obtenção do grau de Engenheiro de Computação.

Orientador: Flávio Roberto dos Santos Coutinho
Centro Federal de Educação Tecnológica de
Minas Gerais – CEFET-MG

BELO HORIZONTE
MARÇO DE 2015

Sumário

1 – Introdução	1
1.1 Objetivos	2
2 – Metodologia	3
2.1 <i>Survey</i>	3
2.2 Métricas	3
2.3 Avaliação dos resultados	4
3 – Resultados Esperados	5
4 – Conclusão	6
5 – Cronograma	7
Referências	8

1 Introdução

Após o advento da internet, nasceu a necessidade de se apresentar documentos em páginas *web*, o *Hipertext Markup Language* (HTML) foi idealizado com a função de marcar o texto de forma a mantê-lo de forma normatizada. Devido ao surgimento de vários *browsers* para apresentação do HTML, viu-se a necessidade de se determinar um padrão para escrita de elementos destes documentos. Com o intuito de padronizar todos os documentos escritos e executados na *web*, criou-se a W3C (*World Wide Web Consortium*), órgão criado por Tim Barnes, com o intuito de manter-se uma padronização e mitigar as diferenças na escrita de documentos e interpretação por parte dos *browsers*, aumentando assim a compatibilidade entre eles e os documentos.

Com o aumento da popularidade de páginas *web*, essa tecnologia passou a ser utilizada para criação de páginas mais complexas do que simples documentos, *e.g.*, portais de venda, fóruns, portais de vídeos, entre outros. Percebeu-se então a necessidade de uma apresentação mais complexa e estética para esses documentos, várias linguagens para editar esta apresentação foram desenvolvidas, entre elas se destacou o *Cascading Style Sheet* (CSS). A autoria de documento CSS é muitas vezes referido como codificação, uma vez que na maioria das vezes gasta-se mais tempo escrevendo o CSS do que criando o design.

O efeito cascata do CSS é controlado pela herança de propriedades e especificidade do seletor. Algumas propriedades de estilo são herdadas de elementos um nível acima da árvore de nós do HTML. A especificidade do seletor determina a prioridade de aplicação das propriedades, de forma tal que seletores mais específicos sobrepõem as que foram definidas em outra regra, ou herdadas de outro elemento.

Sendo um dos três padrões fundamentais da W3C para desenvolvimento de conteúdo *web*, o CSS se tornou largamente utilizado. A separação do documento de conteúdo da apresentação, foi certamente o principal motivo do CSS se tornar tão popular. Apesar das vantagens dessa separação estrutural, os códigos se tornaram complexos e de manutenibilidade onerosa (MESBAH; MIRSHOKRAIE, 2012).

Escrever regras CSS não é uma tarefa trivial, as características da linguagem como herança e especificidade do seletor, colocam os desenvolvedores constantemente em situações, em que, se questionam a efetividade das associações de propriedades escolhidas. Essas características podem prejudicar o que Keller e Nussbaumer (2010, p. 116) definem como efetividade e eficiência:

"Efetividade do código: a folha de estilo é efetiva se o documento de conteúdo ao qual ele é aplicado renderiza da forma desejada. [...]

Eficiência do código: folhas de estilo que causam o mesmo efeito em um documento de conteúdo ainda pode diferir significativamente no modo em que

ela aplica a associação de propriedade. [...] Maximizar a eficiência do código CSS significa aplicar a associação de propriedades de uma forma que o esforço da autoria, manutenção e eventual reutilização seja minimizado."(Tradução nossa.)

Fica então aparente a necessidade de uma medida para qualidade do código CSS. Que dirija a autoria da apresentação de páginas *web*, em direção a um lugar comum. Para assim facilitar a leitura do código, independente do nível de experiência, mesmo dos que não tenham participado efetivamente da autoria. Ou até mesmo evitando situações em que regras e propriedades sejam sobrescritas, ou inutilizadas, e aumentem desnecessariamente o tempo de processamento e renderização das páginas.

1.1 Objetivos

Este trabalho pretende, portanto, desenvolver uma métrica de qualidade visando a manutenibilidade do código, aumentando seu tempo de vida útil mitigando as possíveis falhas. Auxiliando os desenvolvedores e designers, direcionando suas autorias de forma a reduzir repetições, possíveis efeitos colaterais, sobrecarga de propriedades, códigos inutilizados e aumentando assim o nível de efetividade das propriedades associadas.

Espera-se que com essa proposta encontre-se uma medida palpável, que possua relação com a quantidade de falhas relacionadas ao estilo e à métrica proposta. Tentando identificar a relação entre profundidade de seletores, eficiência, abstração e especificidade, para podermos basear nossas escolhas em fundamentos intrínsecos da linguagem, evitando ao máximo a subjetividade da opinião individual.

2 Metodologia

O primeiro passo da execução do trabalho consistirá na consolidação do conceito de manutenibilidade em código CSS. Pretendemos alcançar este objetivo através de referências bibliográficas e de uma pesquisa do tipo *survey* com desenvolvedores profissionais, com níveis de experiência variados. Com esta pesquisa pretendemos identificar as propriedades da linguagem e as situações mais comuns que dificultam, ou facilitam, a manutenção de códigos CSS.

Após executada a pesquisa, iremos consolidar as métricas, propondo valores individuais para as características da linguagem e definindo um índice de qualidade a partir de um agrupamento destes valores. Dessa forma teremos informações suficientes para construir uma ferramenta para verificar nossas descobertas.

Utilizando de ferramentas automatizadas, pretendemos identificar a relação entre nosso índice e a quantidade de falhas identificadas em sistemas *web* de código aberto. Analisando produções do GitHub, iremos fazer a referência cruzada, entre a pontuação alcançada pelo código CSS e o número de *issues* relacionadas ao projeto.

2.1 *Survey*

A pesquisa *survey* é uma forma de obtenção de dados ou informações sobre características, ações ou opiniões de um determinado conjunto de pessoas, indicado como representante de uma população-alvo, por meio de um instrumento de pesquisa, normalmente um questionário (FREITAS; OLIVEIRA, 2000).

O interesse de uma pesquisa desse tipo é produzir descrições quantitativas de uma população. No caso desse trabalho, temos como objetivo identificar, de forma quantitativa, as características identificadas pelos desenvolvedores como sendo as que classificam a legibilidade do código. Para tanto, será executado um *survey* exploratório, com o objetivo de identificar os conceitos do CSS que são centrais para a associação de qualidade do código.

2.2 Métricas

O objetivo desse trabalho é encontrar métricas de qualidade, com o intuito de melhorar a manutenibilidade do código. Garantindo que o código é de fácil legibilidade, entendimento e evite os possíveis efeitos colaterais dessa manutenção.

As métricas são identificadores numéricos baseados em características da linguagem. No caso do CSS, essas características, serão definidas a partir dos resultados do *survey*.

2.3 Avaliação dos resultados

A ferramenta proposta por [Mesbah e Mirshokraie \(2012\)](#) será utilizada para identificar-se os elementos CSS efetivos, calculando um valor de referência devido a razão de seletores inefetivos pelo número total de seletores.

Com esse valor de referência definido como peso do documento, será utilizada uma ferramenta de própria autoria para definir um índice do valor das métricas dos seletores efetivos em função desse peso.

3 Resultados Esperados

Entendemos que códigos inutilizados serão o limite inferior de nossa escala, uma vez que eles só ocuparam espaço na autoria, dificultando a leitura e identificação de efeitos colaterais, quando houverem. Por sua vez, propriedades que não sobrescrevam nenhuma outra, serão consideradas o limite superior de nossa escala. Muitas características da linguagem podem reduzir o valor dessas últimas, como especificidade do seletor, profundidade, localidade, etc. Esperamos poder identificar de forma empírica quais são qualidades de manutenibilidade que atendam ao objetivo, de forma que nossos índices possam evitar um grande número de falhas futuras na apresentação das páginas *web*.

4 Conclusão

Espera-se que a utilização das métricas propostas não influenciem de forma negativa em questões de desempenho, apesar de esta não ser uma característica vital da pesquisa aqui projetada.

Uma métrica válida seria de grande valia para a literatura, uma vez que, como identificado por [Mesbah e Mirshokraie \(2012\)](#), analisar código CSS com uma perspectiva de manutenção ainda não foi explorada em nenhum trabalho científico.

5 Cronograma

Durante a realização do trabalho, pretende-se seguir o seguinte cronograma.

Atividades	Mar 2015	Abr 2015	Jun 2015	Jul 2015	Ago 2015	Set 2015	Out 2015	Nov 2015
Revisão bibliográfica	X	X						
Elaboração e entrega da proposta	X							
Elaboração e execução do survey	X							
Análise de resultado do survey		X						
Elaboração da métrica de qualidade		X	X					
Desenvolvimento da ferramenta de análise		X	X					
Elaboração do projeto de TCC I			X	X				
Entrega do projeto de TCC I				X				
Execução dos testes				X	X			
Análise dos resultados					X	X		
Elaboração do TCC II					X	X	X	
Revisão do texto							X	
Entrega do TCC II								X
Apresentação do TCC II								X

Referências

FREITAS, H.; OLIVEIRA, M. **O Método de pesquisa Survey.pdf**. 2000. 105–112 p. Disponível em: www.rausp.usp.br/download.asp?file=3503105.pdf`\protect\begin{group}\immediate\write\@unused\def\MessageBreak`{\let\protect\edef>Yourcommandwasignored.\MessageBreakTypeI<command><return>to replace it with another command,\MessageBreakor<return>to continue without it.\errhelp\let\def\MessageBreak`(\inputenc)\def\errmessagePackageinputencError:Unicodechar\u8:âÕNot setup for use with LaTeX.``See the inputenc package documentation for explanation.`TypeH<return>for immediate help\endgroup}`. Citado na página 3.

KELLER, M.; NUSSBAUMER, M. Css code quality: A metric for abstractness or why humans beat machines in css coding. In: **Quality of Information and Communications Technology (QUATIC), 2010 Seventh International Conference on the**. [S.l.]: IEEE, 2010. p. 116–121. ISBN 978-1-4244-8539-0. Citado na página 1.

MESBAH, A.; MIRSHOKRAIE, S. Automated analysis of css rules to support style maintenance. In: **Proceedings of the 34th International Conference on Software Engineering**. Piscataway, NJ, USA: IEEE Press, 2012. (ICSE '12), p. 408–418. ISBN 978-1-4673-1067-3. Disponível em: <http://dl.acm.org/citation.cfm?id=2337223.2337272>. Citado 3 vezes nas páginas 1, 4 e 6.