

FACULDADE DE TECNOLOGIA E CIÊNCIAS DE PERNAMBUCO

ISACESSÍVEL: UM SISTEMA WEB PARA AVALIAR ACESSIBILIDADE DE SITES

JOÃO GUEDES DA SILVA NETO

RECIFE

2015

FACULDADE DE TECNOLOGIA E CIÊNCIAS DE PERNAMBUCO

JOÃO GUEDES DA SILVA NETO

ISACESSÍVEL: UM SISTEMA WEB PARA AVALIAR ACESSIBILIDADE DE SITES

Trabalho de conclusão de curso apresentado a Faculdade de Tecnologia e Ciências de Pernambuco, como requisito parcial à obtenção do grau de Bacharel em Ciências da Computação, sob a orientação do professor MSc. Danilo Monteiro.

Orientador: MSc. Danilo Monteiro

RECIFE

2015

Dedico as pessoas fortes que renegam a escuridão  
da ignorância.

## **AGRADECIMENTOS**

Rendo graças a Deus, em nome de Jesus, primeiramente pelo meu professor, orientador e amigo, MSc. Danilo Monteiro, que soube reconhecer minhas limitações, ajudou-me a enxergar ideias, criar motivação e sempre me deu muita força até em assuntos de personalidade. Você não sabe, professor, quanto me ajudou a vislumbrar coisas da vida.

Sou grato, de maneira devedora, a minha esposa, não interessa a circunstância, sempre está comigo. Aos meus pais, que, mesmo em momentos que suas condições não são favoráveis, pedem ao Senhor por nós (sua prole) e nos proveem com todos os recursos que conseguirem alcançar. Aos meus irmãos, que nunca deixam de acreditar e de estender suas mãos.

Também sou grato por todos os colegas de academia, em especial Millena Regina e Mácio Matheus, pelo apoio e experiência de vida profissional que me passaram.

Ao professor e diretor Dr. Veríssimo Crescêncio e a todos os que fazem a Instituição FATECPE.

## **RESUMO**

O presente projeto destina-se a criação de uma ferramenta web que avalia acessibilidade para web, o IsAcessível. Para isto, toma-se como referencial o programa de avaliação e simulação de acessibilidade de sítios do governo federal brasileiro, o ASES, e um código fonte dele disponível na web. Para entender o que é um avaliador de acessibilidade para web, buscar-se-á compreender o que é acessibilidade, sua importância, o tema acessibilidade para web, os padrões que orientam-na, tais como WCAG e EMAG, o que são avaliadores de acessibilidade para web, a ferramenta ASES, motores de busca web, os formatos de arquivos XML e JSON para armazenamento dos dados oriundos da avaliação, e trabalhos relacionados. Com a finalidade de construir uma sequência lógica de avaliação de acessibilidade para web, os algoritmos do programa ASES serão analisados sob uma pesquisa com abordagem analítica. Serão realizados testes com IsAcessível, para diminuir o tamanho dos arquivos, por uma abordagem de pesquisa quanti-qualitativa. O projeto conclui apontando as principais dificuldades enfrentadas e os objetivos específicos alcançados além de citar trabalhos futuros.

Palavras-chave: Acessibilidade. Diversidade. Avaliação. Web. Arquivo.

## **ABSTRACT**

This project is intended to create a web tool that evaluates accessibility to web, IsAcessível. For this, it takes as a reference the assessment program and simulation accessibility of the Brazilian federal government sites, ASES, and his source code available on the web. To understand what is an accessibility evaluator for web, will seek to-understand what is accessibility, its importance, the topic accessibility to web standards that guide it, such as WCAG and EMAG, which are assessors accessibility to the web, the ASES tool, web search engines, XML and JSON file formats for storing data from the assessment, and related work. In order to build a logical assessment of accessibility to web, algorithms of the ASES program will be analyzed from a survey of analytical approach. Tests will be conducted with IsAcessível to reduce the size of files, for a quantitative and qualitative research approach. The project concludes pointing out the main difficulties faced and the specific objectives in addition to citing future work.

Keywords: Accessibility. Diversity. Evaluation. Web. Archive.

## LISTA DE FIGURAS

Figura 1 – Tela Avaliador de Acessibilidade.....	29
Figura 2 – Tela Resumo do Relatório.....	30
Figura 3 – Tela Relatório da URL.....	30
Figura 4 – Pasta temporária com o arquivo texto e xml selecionados.....	32
Figura 5 – Tela Avaliador de Acessibilidade do código fonte.....	33
Figura 6 – Tela de Resumo do Relatório do código fonte.....	34
Figura 7 – Tela de Relatório da URL do código fonte.....	34
Figura 8 – Mensagem de erro após executar a classe ASES.....	35
Figura 9 – Tela de login exibida na execução do código fonte do ASES.....	36
Figura 10 – Principais classes do ASES.....	37
Figura 11 – Trecho de código original da classe PegarPaginaWeb: construtor implementando a biblioteca Java Commons HttpClient.....	40
Figura 12 – Diagrama de caso de uso do IsAcessível.....	45
Figura 13 – Tela de entrada do IsAcessível.....	46
Figura 14 – Tela de relatório resumido.....	46
Figura 15 – Tela relatório de avaliação do IsAcessível.....	47
Figura 16 – Principais classes.....	49
Figura 17 – Arquivo XML de avaliação visualizado no navegador.....	53
Figura 18 – Arquivo XML de avaliação com tag <conteudo> expandida.....	56
Figura 19 – Arquivo XML de avaliação com a tag <listaep1> expandida.....	57
Figura 20 – Elemento <pontoverificacao> expandido.....	58
Figura 21 – Exceção lançada nos testes com os elementos <gl> e <cp>.....	59
Figura 22 – Principais classes que implementam as configurações de arquivção do IsAcessível.....	65

Figura 23 – Trecho de código FachadaArquivador.setAmbienteArquivacao.....	68
Figura 24 – Trecho de código da classe AvaliadorDeAcessibilidade.avaliar().....	69
Figura 25 – Tela de entrada com os critérios de teste.....	70
Figura 26 – Tela Relatório resumido.....	70
Figura 27 – Tela de Relatório de avaliação.....	70
Figura 28 – Pasta temporária de usuário com os arquivos gerados na Configuração 0.....	71
Figura 29 – Pasta temporária de usuário com os arquivos gerados na Configuração 2.....	72
Figura 30 – Relatório de avaliação gerado pela Configuração 2.....	72
Figura 31 – Pasta temporária de usuário com os arquivos gerados na Configuração 3.....	73
Figura 32 – Relatório de avaliação resultante da Configuração 4.....	74
Figura 33 – Pasta temporária de usuário com os arquivos gerados na Configuração 4.....	74
Figura 34 – Trecho do código do método gravaJsonNoHD da classe GerenciadorRelatorioJSON.....	75
Figura 35 – Trecho do relatório de avaliação no formato JSON, com caracteres de escape....	75
Figura 36 – Pasta temporária de usuário com os arquivos gerados na Configuração 5, com relatório de avaliação contendo caracteres de escape.....	76
Figura 37 – Trecho do código do método gravaJsonNoHD da classe GerenciadorRelatorioJSON, utilizando a classe GsonBuilder.....	76
Figura 38 – Trecho do relatório de avaliação no formato JSON, com caracteres de escape....	77
Figura 39 – Pasta temporária de usuário com os arquivos gerados na Configuração 5, com relatório de avaliação sem caracteres de escape.....	77
Figura 40 – Pasta temporária de usuário com os arquivos gerados na Configuração 7.....	78



## **LISTA DE TABELAS**

Tabela 1 – Correspondência das formas de nomenclatura normal e mínima dos elementos e atributos da estrutura XML do relatório de avaliação.....	61
Tabela 2 – Configurações de arquivção.....	64
Tabela 3 – Correspondência das configurações de arquivção com as respectivas constantes	67
Tabela 4 – Valores obtidos nos testes com as configurações de arquivção para o IsAcessível .....	78

## LISTA DE SIGLAS

ASES – Avaliador e Simulador de Acessibilidade de Sítios

API – *Application Programming Interface*

BITV – *Barrierefreie Informationstechnik-Verordnung*

CSS – *Cascading Style Sheets*

DOM – *Document Object Model*

EMAG – Modelo de Acessibilidade do Governo Eletrônico

GNU – *General Public License*

GSA – *General Services Administration*

HTML – *HyperText Markup Language*

IBGE – Instituto Brasileiro de Geografia e Estatística

IDE – *Integrated Development Environment*

J2EE – *Java Enterprise Edition*

JSF – *Java Server Faces*

JSON – *JavaScript Object Notation*

kB – kiloByte

LGPL – *Lesser General Public License*

NDA – *National Disability Authority*

OMS – Organização Mundial de Saúde

P1 – Prioridade 1

P2 – Prioridade 2

P3 – Prioridade 3

P.V – Ponto de Verificação

RIA – *Rich Internet Application*

URL – *Uniform Resource Locator*

*W3C – World Wide Web Consortium*

*WAI – Web Accessibility Initiative WAI-ARIA – Web Accessibility Initiative – Accessible Rich Internet Application*

*WCAG – Web Content Accessibility Guidelines*

*WCAG WG – Web Content Accessibility Guidelines Work Group*

*WHO – World Health Organization*

*XHTML – eXtensible HyperText Markup Language*

*XML – eXtensible Markup Language*

## SUMÁRIO

1 INTRODUÇÃO.....	14
1.1 Objetivos.....	17
1.1.1 Objetivo principal.....	17
1.1.2 Objetivos específicos.....	18
1.2 Referencial teórico.....	18
1.2.1 Acessibilidade.....	18
1.2.2 Acessibilidade para <i>web</i> .....	19
1.2.3 <i>Web Content Accessibility Guidelines (WCAG)</i> .....	20
1.2.4 Modelo de Acessibilidade do Governo Eletrônico (EMAG).....	22
1.2.5 Avaliação de acessibilidade para <i>web</i> .....	23
1.2.6 Avaliadores de acessibilidade para <i>web</i> .....	24
1.2.7 Motor de busca <i>web</i> .....	24
1.2.8 O rastreador <i>web crawler</i> 4j.....	25
1.2.9 Formato de arquivo XML.....	26
1.2.10 Formato de arquivo JSON.....	26
1.2.11 ASES – Avaliador e Simulador de Acessibilidade de Sítios.....	27
1.2.11.1 A funcionalidade Avaliador de acessibilidade.....	27
1.2.11.2 Pasta de arquivos temporários.....	31
1.2.12 Código fonte do ASES.....	32
1.2.12.1 Enfrentamentos para executar o código fonte do ASES.....	35
1.2.12.2 Abstração lógica da avaliação de acessibilidade.....	36
1.2.12.2.1 Lógica abstraída de primeiro momento.....	38
1.2.12.2.2 Lógica abstraída de segundo momento.....	39
1.2.12.3 Motor de busca do ASES.....	40

1.2.13 Trabalhos relacionados.....	41
2 DESENVOLVIMENTO.....	42
2.1 Metodologia.....	42
2.1.1 Abordagem diagnóstica.....	42
2.1.2 Tipo de pesquisa analítico.....	42
2.1.3 Abordagem quanti-qualitativa.....	43
2.2 Resultados.....	43
2.2.1 O IsAcessível.....	43
2.2.1.1 Tecnologias utilizadas.....	44
2.2.1.2 Principais restrições.....	44
2.2.1.3 Diagrama de caso de uso.....	44
2.2.1.4 Principais classes.....	48
2.2.2 Coleta de dados.....	51
2.2.2.1 Entendo o arquivo XML com dados do relatório de avaliação do ASES.....	51
2.2.2.2 Considerações sobre os arquivos de relatório de avaliação para as necessidades do IsAcessível.....	60
2.2.2.3 Testes com configurações de arquivagem para diminuir a quantidade de espaço de memória das pastas temporárias de usuário.....	61
2.2.2.4 Resumo dos testes das Configurações.....	78
2.3 Discussão.....	79
3 Conclusão.....	80
3.1 TRABALHOS FUTUROS.....	80
REFERÊNCIAS.....	82

## 1 INTRODUÇÃO

Como pode ser visto, atualmente, a utilização da *Internet*, definida, na Cartilha (2013), como grande rede de computadores, para distribuição de vários recursos e serviços, faz parte da estrutura vital da maioria das organizações. Estas, para alcançar seus clientes, querem que suas soluções computacionais facilitem, ao máximo, o acesso destes às suas interfaces de usuário, como pode ser inferido em Maciel (2012), sendo verdadeiros ambientes de relacionamento com o cliente. Assim, as soluções computacionais, demonstram sucesso quando o cliente compreende, perfeitamente bem, a mensagem da organização com quem está se relacionando. Uma das soluções computacionais para relacionamento com o cliente é o *sítio web (site)* que, segundo a Cartilha (2013, p. 18-19) define, “web, é um dos serviços oferecidos na Internet, concebido para disponibilizar permanentemente informações a todos os usuários” e *site* é “um conjunto de páginas *web* interligadas e que possuem o mesmo endereço principal e a mesma administração”. Nos *sites*, as organizações conseguem apresentar muitos recursos e serviços para seus clientes, porém, de acordo com Nicácio (2010), o acesso do cliente à mensagem no *site*, depende, não só do contato com o *site*, mas também, da forma como ele interage com o *site*, da sua capacidade de capacitação e compreensão das informações do *site*.

Observar estas dependências de acesso é valorizar a diversidade humana, pois favorece a participação de mais pessoas no uso de diversas tecnologias, principalmente, pessoas portadoras de deficiência. Dados do censo demográfico IBGE 2010 demonstram que 23,9% da população brasileira, que corresponde a 45.606.048 milhões de pessoas, tem pelos menos uma das deficiências investigadas no censo (IBGE, 2010). A Organização Mundial de Saúde (OMS) constatou que mais de um bilhão de pessoas, cerca de 15% da população do mundo, têm algum tipo de deficiência (WHO, 2015). Consoante a tecnologia e aos deficientes Soares Filho (2015, p. 11) coloca,

Nessa busca pela isonomia das pessoas com deficiência, a tecnologia tem exercido um papel fundamental, reduzindo/eliminando barreiras que as limitam de desempenhar atividades cotidianas de forma simples e independente, permitindo, ainda, que usufruam de maneira mais completa de suas capacidades intelectuais, comunicativas e físicas.

Com base nesta realidade, a promoção de utilidade das coisas sem distinção de capacidades biológicas dos humanos, conhecida por acessibilidade, levou a sociedade criar

disciplinas como *Design Universal* que, segundo Connell et al (1997 apud CHALEGRE, 2011, p. 16),

[...] diz respeito ao desenvolvimento de produtos e ambientes para serem utilizados por todas as pessoas, na maior extensão possível, sem a necessidade de adaptação ou de design especializado [...]

,estabelecendo regras, diretrizes, princípios, padrões e criando tecnologias que estão em constante aperfeiçoamento.

Desta forma, o desenvolvimento de soluções computacionais, nestes tempos de valorização da diversidade humana, conforme Soares Filho (2015), não pode ser deficiente em atender aos usuários com diferentes capacidades e em diferentes circunstâncias de interação homem-máquina. Devendo o arquiteto contar com toda a tecnologia existente e ser proativo, quanto a esta necessidade, ainda em fase de idealização do projeto.

Com relação a acessibilidade na *web*, pode-se observar os esforços de promoção de uma *web* acessível mantidos por várias organizações no mundo:

- i. Grupos de trabalho da *Web Accessibility Initiative (WAI)* (W3C, 2012);
- ii. Orientações de acessibilidade de diversos governos, tais como Modelo de Acessibilidade em Governo Eletrônico (eMAG) do Brasil (EMAG, 2004);
- iii. Disponibilizações de diversas ferramentas e *plugins* como as que o W3C propaga na sua lista de ferramentas de avaliação de acessibilidade web (W3C, 2014);
- iv. A comunidade de código aberto que também tem contribuído. Uma contribuição é a disponibilização do código fonte do programa ASES<sup>1</sup>. Que é uma ferramenta *desktop* para avaliação, simulação e correção de acessibilidade em páginas, sítios e portais, disponibilizada pelo governo federal brasileiro. O código fonte pode ser encontrado em Acessibilidade Brasil (2010).

Porém, mesmo com estas e tantas outras tecnologias para promoção de acessibilidade na *web*, pesquisa, realizada por Soares Filho (2015), demonstra a existência de

---

<sup>1</sup> ASES: Avaliador e Simulador de Acessibilidade de Sítios. Versão 1.0.2 Linux. Disponível em: <<http://www.governoeletronico.gov.br/acoes-e-projetos/e-MAG/ases-avaliador-e-simulador-de-acessibilidade-sitios>>. Acesso em: 27 de out.2015.

algumas deficiências de acessibilidade em algumas destas tecnologias. Então, há muito para ser feito e é necessário mais mentes trabalhando.

A proposta deste trabalho é criar um sistema web para avaliar acessibilidade de sites, o IsAcessível<sup>2</sup>. Utilizando-se, como referência para compreender o algoritmo de avaliação de acessibilidade para web, o programa ASES na versão 1.0.2 Linux, com foco na sua funcionalidade de avaliação de acessibilidade para web.

Este programa foi escolhido por ser código aberto, ser para sistema operacional utilizado pelo pesquisador, a versão ser compatível com o código fonte disponível e, avaliar acessibilidade conforme as orientações eMAG. Outro ponto positivo para escolha é apresentação de resultados ao usuário. Parte do estilo de resultados do ASES será o estilo de resultados do IsAcessível.

Este documento está organizado em Introdução, Desenvolvimento, Conclusão e Referências. A Introdução inicia com um panorama deste projeto. Apresenta os tópicos Objetivos e Referencial teórico.

No tópico Objetivos é elencado o objetivo principal do projeto bem como os objetivos específicos, metas para o atingimento do objetivo principal.

O Referencial teórico traz assuntos que dão base para o entendimento dos principais temas envolvidos no projeto bem como compreensão dos objetos de pesquisa do projeto, como segue:

- Acessibilidade;
- Acessibilidade para web;
- *Web Content Accessibility Guidelines (WCAG)*;
- Modelo de Acessibilidade do Governo Eletrônico (EMAG);
- Avaliação de acessibilidade para web;
- Avaliadores de acessibilidade para web;
- Motor de busca web;
- O rastreador web crawler<sup>4j</sup>;
- Formato de arquivo XML;

---

<sup>2</sup> <https://github.com/jguedes/IsAcessivel>



- Formato de arquivo JSON;
- ASES – Avaliador e Simulador de Acessibilidade de Sítios, com os subtópicos: A funcionalidade Avaliador de Acessibilidade; A pasta de arquivos temporários;
- Código fonte do ASES, com os subtópicos: Enfrentamentos para executar o código fonte do ASES; Abstração lógica da avaliação de acessibilidade; Motor de busca do ASES;
- Trabalhos relacionados.

O Desenvolvimento apresenta a Metodologia, Resultados e Discussão. Em Metodologia são apresentados as abordagens e o tipo de pesquisa utilizada no projeto.

Em Resultados são apresentados dados sobre a análise dos objetos de pesquisa deste projeto, demonstrados nos subtópicos:

- IsAcessível – são apresentadas as tecnologias utilizadas, as principais restrições, o diagrama de caso de uso e as principais classes;
- Coleta de dados – em que são listados os dados coletados e examinados nos seguintes subtópicos: Entendo o arquivo XML com dados do relatório de avaliação do ASES; Considerações sobre os arquivos de relatório de avaliação para as necessidades do IsAcessível; Testes com configurações de arquivagem para diminuir a quantidade de espaço de memória das pastas temporárias de usuário; Resumo dos testes das Configurações.

A seção Discussão traz opiniões do pesquisador sobre alguns pontos do projeto.

A Conclusão apresenta as conclusões da realização do presente projeto e trabalhos futuros.

Por fim, em Referências estão listadas as fontes referenciadoras do projeto.

## 1.1 OBJETIVOS

### 1.1.1 Objetivo principal

Desenvolver um aplicativo, denominado IsAcessível, para ambiente *web* que avalie a acessibilidade para *web*.

### 1.1.2 Objetivos específicos

1. Compreender a funcionalidade de avaliação de acessibilidade para web do programa ASES 1.0.2 Linux;
2. Abstrair o código da funcionalidade de avaliação de acessibilidade para web do programa ASES 1.0.2 Linux para ser independente de ambiente *desktop* ou web;
3. Criar interface web de usuário para o aplicativo IsAcessível;
4. Estabelecer um gerenciamento sustentável de arquivos voltado para uma aplicação web;
5. Verificar o desempenho de acesso a links utilizando motor de busca mais moderno.

## 1.2 REFERENCIAL TEÓRICO

Como base teórica para este projeto, faz-se necessário discorrer sobre acessibilidade, acessibilidade para web, as recomendações WCAG e eMAG, avaliação de acessibilidade para web, avaliadores de acessibilidade para web, o programa ASES, motor de busca web, rastreador web crawler<sup>4j</sup>, formato de arquivo XML e JSON, além de, trabalhos relacionados. Tais assuntos serão abordados nesta ordem.

### 1.2.1 Acessibilidade

No dicionário Aurélio, acessibilidade é qualidade de acessível, é a facilidade na aproximação, no trato ou na obtenção. E, acessível, é algo a que se pode chegar, de acesso fácil, que se pode alcançar, obter ou possuir, inteligível, compreensível, tratável, comunicativo (FERREIRA,1988).

Acessibilidade é um direito do ser humano no exercício de sua cidadania pois, segundo apresenta Feijó (2011), o direito a acessibilidade faz parte dos direitos fundamentais na Declaração dos Direitos Humanos e do Cidadão de 1948, salientando o reconhecimento das diferenças de gênero (sexo), fases de vida (faixa etária), normalidade e excepcionalidade. Também chama a atenção para o imenso desafio de universalizar os direitos humanos firmemente para todos mostrando citações que afirmam que os direitos humanos são efetivamente universalizados com a conscientização e divulgação da população. Ainda, elucida a comparação do direito da acessibilidade com o da igualdade:

[...] os direitos dos cidadãos são os mesmos, mas as condições para exercê-los não. Daí a importância do princípio da igualdade ser aplicado nos demais direitos expostos, como o de acessibilidade, a fim de proporcionar [...] uma vida o mais natural possível (FEIJÓ, 2011).

A palavra acessibilidade está associada a *Design Universal* que, segundo Soares Filho (2015), entende-se que é um processo de projetar produtos, serviços ou ambientes incluindo a possibilidade de pessoas com restrições ou peculiaridades desfrutarem dos recursos ou das funcionalidades oferecidas. O termo acessibilidade indica a possibilidade de qualquer indivíduo usufruir plenamente os benefícios da sociedade (CHALEGRE, 2011).

Nicácio (2010, p. 21-22) exemplifica a necessidade de acessibilidade de forma interessante:

[...] nem tudo que é fácil para um adulto normal é fácil para uma criança, ou um idoso, ou um deficiente físico e assim por diante. Nem tudo que é acessível para um cego é também acessível para um paraplégico ou simplesmente para alguém com a perna engessada. Cada pessoa, independente de ser deficiente física ou não, possui um grau diferente de necessidade para poder acessar alguma coisa.

Portanto, entende-se que acessibilidade é um substantivo que representa a adequação das coisas para permitir que sejam utilizadas irrestritamente por qualquer indivíduo. É um assunto tratado pela humanidade levando em conta sua diversidade. É a sociedade dando ouvidos a uma parcela de si mesma, percebendo que a tecnologia para servir bem tem que ser usufruída por todos. É uma preocupação de quem projeta com humanidade. É dinâmica, como a sociedade, e deve ser aprendida e colocada em prática até que se torne homogênea.

### **1.2.2 Acessibilidade para web**

Antes de dispor sobre acessibilidade na web é oportuno enfatizar que a web é um serviço oferecido na Internet, foi criada por Tim Berners-Lee, tem seus padrões desenvolvidos pelo *World Wide Web Consortium (W3C)* “que é um consórcio internacional em que organizações filiadas, uma equipe em tempo integral e o público trabalham juntos para desenvolver padrões para a web”. O W3C também foi criado pelo pai da web (CARTILHA, 2013).

Conforme Nicácio (2010), entende-se que web acessível é a que permite o acesso a todos os usuários. Para promover isto, o W3C mantém a principal documentação contendo as diretrizes ou recomendações que explicam como tornar o conteúdo web acessível a todas as pessoas, o *Web Content Accessibility Guidelines (WCAG)* (EMAG, 2014). Soares Filho

(2015), explica que só as orientações do WCAG, atualmente na versão 2.0, não são capazes de resolver todos os problemas de acessibilidade da atual web devido à interatividade proporcionada por aplicações da Internet dentro do conceito *Rich Internet Application (RIA)*, e que para tentar melhorar o cenário a W3C criou o *Accessible Rich Internet Application (WAI-ARIA)* 1.0 que é um documento com diretrizes para implementar adequadamente comportamentos e informações dinâmicas da estrutura na interface do usuário.

Além do W3C, muitos governos, com foco nas suas realidades, desenvolveram, com base nas orientações de acessibilidade do W3C, suas próprias formas de orientações, tais como, Modelo de Acessibilidade em Governo Eletrônico (eMAG) do Brasil (EMAG, 2004), *Section508* programa de acessibilidade dos Estados Unidos (GSA, 2015), *National Disability Authority (NDA)* orientações de acessibilidade da Irlanda (NDA, 2012), e *BITV* orientação alemã (BITV, 2015). Muitas organizações e desenvolvedores, em vários lugares do mundo, unem-se à W3C para apoiá-la e promover suas recomendações (W3C, 2015).

Entende-se que, a acessibilidade na web é preocupação da web em si e do mundo, unindo forças para criar orientações que, diante de uma deficiência de acessibilidade a uma determinada informação, possam esclarecer esta deficiência e apresentar possíveis soluções, proporcionando uma web cada vez mais padronizada e acessível.

### **1.2.3 Web Content Accessibility Guidelines (WCAG)**

Traduzido oficialmente para Diretrizes de Acessibilidade para Conteúdo Web, são recomendações da W3C com finalidade de tornar as interfaces webs mais acessíveis. Melhorando a funcionalidade e interoperabilidade da web (W3C, 2008).

São consideradas, também, como referências, genéricas, de princípios e ideias para interfaces web. São divulgadas, pelo W3C, como documento de recomendação produzido pelo *Web Content Accessibility Guidelines Work Group (WCAG WG)*, que traduzido é Grupo de Trabalho para Diretrizes de Acessibilidade para Conteúdo Web. Este documento é revisado por membros do W3C, desenvolvedores de softwares, outros grupos do W3C e partes interessadas (W3C, 1999; W3C, 2008).

O W3C já divulgou duas versões, WCAG 1.0 e WCAG 2.0. A WCAG 2.0 é baseada na 1.0, porém, enquanto esta é voltada para questões de acessibilidade web para usuários humanos, aquela, também busca alcançar questões de acessibilidade nas tecnologias web

atuais e futuras (W3C, 1999; W3C, 2008). Outra diferença marcante é a organização das diretrizes.

Em WCAG 1.0, são 40 orientações, que podem ser reconhecidas como princípios, e estão dispostas em 14 temas. Cada orientação possui pontos de verificação (em inglês, *checkpoints*) (W3C, 1999).

Cada ponto de verificação é sinalizado com um nível de prioridade, onde nível 1 significa que a regra do ponto de verificação deve ser satisfeita, para que um ou mais usuários tenham acesso à informação. Nível 2, a regra deveria ser satisfeita, para garantir que um ou mais usuários tenham acesso à informação. E, nível 3, a regra pode ser satisfeita, para facilitar que um ou mais usuários tenham acesso à informação (W3C, 1999).

A satisfação dos pontos de verificação, com base nos níveis de prioridade, confere, às interfaces *webs*, três níveis de conformidade, a saber, nível A (todos os pontos de verificação com prioridade nível 1 satisfeitos), AA (todos os pontos de verificação com prioridade nível 1 e 2 satisfeitos) e, AAA (todos os pontos de verificação com prioridade nível 1, 2 e 3 satisfeitos) (W3C, 1999).

Em WCAG 2.0, a base da acessibilidade web é constituída por 4 princípios (perceptível, operável, compreensivo e robusto). Cada princípio engloba diretrizes (ou orientações), que fornecem os objetivos básicos a serem atingidos. Sendo um total de 12 diretrizes (W3C, 2008).

Cada diretriz contém critérios de sucesso (equivalentes aos pontos de verificação em WCAG 1.0). Cada critério de sucesso, requisita a satisfação de uma regra de acessibilidade averiguável, que descreve especificadamente o que deve ser feito diante de incompatibilidades à regra (W3C, 2008).

O critério de sucesso, quando satisfeito, incrementa um dos três níveis de conformidade, de uma interface *web*, com as orientações WCAG 2.0, a saber, nível A (mínimo possível, exige que todos os critérios de sucesso nível A, da interface web, sejam satisfeitos), nível AA (exige que todos os critérios de sucesso nível A e AA, da interface web, sejam satisfeitos) e nível AAA (exige que todos os critérios de sucesso, da interface web, sejam satisfeitos) (W3C, 2008).

A conformidade em um nível, com ausência de critérios de sucesso de nível imediatamente mais alto, implica conformidade com o nível imediatamente mais alto. Para

cada diretriz e seus critérios de sucesso há um conjunto de técnicas necessárias e outras sugeridas (W3C, 2008; W3C, 2015).

Percebe-se que a intensão deste documento é convencionar fundamentos de acessibilidade *web*. Definindo regras específicas e averiguáveis de acessibilidade nas interfaces *webs*, bem como, técnicas de implementação destas regras.

A fim de proporcionar o fomento de soluções para detecção de deficiências de acessibilidade *web*. Disponibilizando, gratuitamente, meios de promover uma *web* mais acessível. Portanto, compreende-se, por exemplo, que o interessado em promover acessibilidade numa interface *web* deve procurar enxergar os quatro princípios na interface, utilizando as técnicas necessárias e sugeridas para satisfazer, no mínimo, os critérios de sucesso nível A.

Se quiser garantir a acessibilidade, deve procurar satisfazer, no mínimo, todos os critérios de sucesso nível AA e, se quiser facilitar a acessibilidade, satisfazer todos os critérios de sucesso de nível AAA.

#### **1.2.4 Modelo de Acessibilidade do Governo Eletrônico (EMAG)**

Visando a qualidade, coerente com as necessidades brasileiras, da acessibilidade *web* nos sítios e portais do governo federal brasileiro, foram criadas recomendações de acessibilidade para serem seguidas pelos órgãos governamentais e sociedade em geral na criação de sítios e portais *web*.

Estas recomendações são disponibilizadas no Modelo de Acessibilidade do Governo Eletrônico – eMAG. O eMAG mantém suas recomendações em conformidade com os padrões internacionais. Está baseado no WCAG de forma especializada mas, não despreza as boas práticas do WCAG (EMAG, 2004).

O eMAG foi desenvolvido em 2004 mas, a primeira versão (1.4) e a segunda (2.0) foram lançadas em 2005 baseadas no WCAG 1.0. Em 2007, passou a ser obrigatoriamente observado nos sítios e portais do governo brasileiro (EMAG, 2014).

Em 2008, com o lançamento do WCAG 2.0 e revisão da versão 2.0, desenvolveu-se a versão 3.0, que foi lançada em 2011. Atualmente está na versão 3.1, lançada em 2014 (EMAG, 2014).

Até a versão 2.0 as recomendações eram organizadas em três níveis de prioridade de acessibilidade onde: nível 1, são as exigências básicas; nível 2, garantia de acessibilidade e; nível 3, facilitação de acessibilidade. Para estar em conformidade com o eMAG, os portais e sítios *webs* deveriam estar, no mínimo, observando as recomendações do nível 1 (EMAG, 2005).

A partir da versão 3.0, a organização das recomendações foi modificada e passou-se a exigir, para estar em conformidade com o eMAG, a observância de todas as suas recomendações, não existindo, desta forma, níveis de prioridade de acessibilidade (EMAG, 2014).

O eMAG 3.1 organiza as recomendações de acordo com necessidades de implementação. Que estão agrupadas em cinco situações possíveis: Marcação, Comportamento (*Document Object Model – DOM*), Conteúdo/Informação, Apresentação/*Design*, Multimídia, Formulário. Para desenvolver um sítio ou portal web em conformidade com o eMAG, todas as situações devem ser identificadas e, para cada uma, ter suas recomendações implementadas (EMAG, 2014).

#### **1.2.5 Avaliação de acessibilidade para web**

Com a finalidade de permitir que pessoas portadoras de alguma restrição tenham acesso a *web*, ferramentas assistivas, por exemplos leitores de tela, intermedeiam esta interação. Elas são implementadas para entender a estrutura da interface web, requisitada pelo usuário, decodificando-a para que este consiga compreender as informações da interface.

Como os padrões *web* estabelecem uma lógica na localização das informações nas interfaces *webs*, se a estrutura da interface *web* não estiver nos padrões de acessibilidade dificultará e até impossibilitará o acesso do usuário (SOARES FILHO, 2015).

Para colaborar com o desenvolvimento da *web* nos padrões de acessibilidade, foram desenvolvidas soluções para avaliar a conformidade das interfaces *webs* com os padrões. A exemplo da apresentada pelo governo federal brasileiro na qual são sugeridos 5 passos (EMAG, 2014):

1. Validar os códigos do conteúdo HTML e das folhas de estilo;
2. Verificar o fluxo de leitura da página;

3. Realizar a validação automática de acessibilidade utilizando o ASES e outros avaliadores automáticos;
4. Realizar a validação manual;
5. Teste com usuários reais.

Não é intenção deste trabalho estudar cada ponto desta solução ou outra solução existente. Porém, será elucidado a funcionalidade de avaliação de acessibilidade do ASES.

Resumidamente, entende-se que, para realizar avaliações de acessibilidade, deve-se criar páginas web dentro dos padrões W3C, auxiliando-se por softwares que avaliam a conformidade da página com estes padrões; seguindo orientações de acessibilidade, auxiliando-se pelos softwares avaliadores de acessibilidade; testando as páginas em ferramentas assistivas, e; realizando testes em circunstâncias reais.

### **1.2.6 Avaliadores de acessibilidade para web**

São aplicações que ajudam na avaliação da conformidade de interface web com os padrões de acessibilidade.

O governo brasileiro disponibiliza o programa ASES, como ferramenta para realizar validações das interfaces webs e ele pode ser utilizado por qualquer pessoa. O ASES é *desktop*, gratuito, específico para WCAG e eMAG, específico para idioma brasileiro, avalia página web, sítio web, arquivo, diretório, gera relatórios, mostra os pontos relevantes da avaliação na página, entre outros (BRASIL, 2008).

O W3C lista um conjunto de avaliadores de acessibilidade para web (W3C, 2014). Dentre os quais, até o momento de elaboração deste trabalho, o ASES não faz parte. Além de, não ter sido encontrado, entre eles, nenhum avaliador em idioma brasileiro que realize avaliações no padrão eMAG ou que, pelo menos, seja semelhante ao ASES na sua forma de apresentar resultados.

### **1.2.7 Motor de busca web**

Motor de busca é um mecanismo utilizado para recuperar informações na internet. Possui um software que comumente é chamado de *robot*, *crawler* ou *spider*. Este software é quem varre a web em busca de conexões a partir de um *link*, possibilitando a coleta de informações de *sites* (OLIVEIRA, 2011). Utilizar-se-á a nomenclatura *crawler*.



Existem vários crawlers de código aberto. As peculiaridades inerentes a cada um devem ser consideradas pelas necessidades do usuário, conforme demonstram Yadav e Goyal (2015). Eles realizaram um estudo comparativo entre os de código aberto, disponíveis. Apontando propriedades que devem existir neles: robustez, respeito (alguns recursos na web possuem restrições para acesso de crawlers), distribuído, escalável, performance e eficiência, qualidade, capacidade de obter dados atualizados, extensível. Além de discutirem sobre técnicas e seus efeitos sobre vários parâmetros.

Dentre os diversos crawlers da comparação realizada por Yadav e Goyal (2015), existem alguns que são voltados para linguagem Java: Apache Nutch<sup>3</sup>, Heritrix<sup>4</sup>, WebSphinx<sup>5</sup> e JSpider<sup>6</sup>.

No site Java-Source.net<sup>7</sup> são listados vários outros crawlers baseados em Java, dentre eles o Crawler4j<sup>8</sup>. Segundo Ricardo e Serrão (2013), é uma *Application Programming Interface (API)* e sua principal vantagem é a facilidade de integração com projetos Java.

O código fonte do ASES implementa uma rotina de conexão e captura de conteúdo HTML da página visitada, utilizando a biblioteca *Java Commons HttpClient*<sup>9</sup>.

### 1.2.8 O rastreador web crawler4j

É um rastreador web, de código aberto, para Java.

O site<sup>10</sup> apresenta duas classes importantes: Controller e WebCrawler. Que rastreiam todos os links em cada página que visitam e visitam as páginas, permitidas, de todos os links rastreados.

A classe Controller cria um objeto do tipo CrawlController, responsável por executar a classe que herda da classe WebCrawler, e configura os seus parâmetros. Os parâmetros necessários são:

- URLs principais (chamadas de seeds, em inglês) – são as URLs que iniciarão o rastreamento;
- Pasta onde os dados podem ser armazenados, e;

<sup>3</sup> <<http://nutch.apache.org/>>. Acessado em 26 nov.2015.

<sup>4</sup> <<https://webarchive.jira.com/wiki/display/Heritrix>>. Acessado em 26 nov.2015.

<sup>5</sup> <<https://www.cs.cmu.edu/~rcm/websphinx/>>. Acessado em 26 nov.2015.

<sup>6</sup> <<http://j-spider.sourceforge.net/>>. Acessado em 26 nov.2015.

<sup>7</sup> <<http://java-source.net/open-source/crawlers>>. Acessado em 26 nov.2015.

<sup>8</sup> <<https://github.com/yasserg/crawler4j>>. Acessado em 26 nov.2015.

<sup>9</sup> <<http://hc.apache.org/httpclient-3.x/>>. Acessado em 26 nov.2015.

<sup>10</sup> <<https://github.com/yasserg/crawler4j>>. Acessado em 26 nov.2015.

- Quantidade de threads concorrentes – utilizadas no rastreamento.

A classe *WebCrawler*, que deve ser herdada, é responsável por averiguar a permissão de visitar páginas e, por visitar as páginas. O método público boolean *shouldVisit(Page referringPage, WebUrl url)* é quem realiza os testes com os critérios de permissão. O parâmetro do tipo *Page* é a página web onde a URL, para teste, foi encontrada e o parâmetro do tipo *WebUrl* é a URL para o teste. O método público void *visit(Page page)*, é quem visita a página. O parâmetro de tipo *Page*, além do conteúdo da página, pode também, entre outros dados, trazer todas as URLs dela.

O site ainda demonstra outras configurações tais como a possibilidade de limitar a profundidade máxima de rastreamento, obedecer a políticas de restrições a visita de crawlers, realizar autenticação em proxys, etc.

### 1.2.9 Formato de arquivo XML

O *Extensible Markup Language (XML)*<sup>11</sup> é uma recomendação W3C. Apresentada por Costa (2007) como uma linguagem, suportada em *tags* ou marcas que fornece um formato para descrever dados estruturados, que facilita declarações mais precisas do conteúdo.

Abinader (2006) explica o que é marcador inicial, marcador final, elementos e seus atributos em XML para sinalizar dados estruturados. Segundo ele, marcador, ou *tag*, inicial, é um nome entre o sinal de menor (<) e maior (>), por exemplo <marcador>. Marcador, ou *tag*, final, é um nome entre o sinal de menor seguido de barra(</) e maior (>), por exemplo </marcador>. Elemento é um par ordenado de marcador inicial e final, por exemplo <marcador></marcador>. Atributo é um par ordenado nome="valor" ou nome='valor'. O atributo é opcional e deve estar no marcador inicial, por exemplo <marcador atributo='valor'></marcador>. N atributos podem ser inseridos. O dado estruturado deve estar entre marcação inicial e final do elemento, por exemplo <marcador>dados estruturados</marcador>. Metadados sobre os dados estruturados podem ser inseridos como atributos do elemento. Os dados estruturados podem ser literais ou elementos aninhados.

### 1.2.10 Formato de arquivo JSON

O *JavaScript Object Notation (JSON)*<sup>12</sup> é um formato de texto para serialização de dados estruturados, independente de linguagem, utilizado para troca de dados. Permite utilizar

<sup>11</sup> <<http://www.w3.org/TR/2008/REC-xml-20081126/>>. Acessado em 24 nov.2015.

<sup>12</sup> <<http://www.ietf.org/rfc/rfc4627.txt>>. Acesso em: 24 nov.2015.

quatro tipos primitivos de dados (*strings*, números, booleanos e *null*) e dois tipos estruturados (*Object* e *array*).

Smith (2015), coloca que a especificação de JSON define que os dados podem ser estruturados de duas formas: uma coleção de pares nome/valor e uma lista ordenada de valores.

#### **1.2.11 ASES – Avaliador e Simulador de Acessibilidade de Sítios**

É um programa distribuído em Brasil (2008), liberado para o uso da sociedade em geral. Ele é utilizado para avaliar, simular e corrigir acessibilidade em páginas, sítios e portais. Gratuito e sob licença GNU Lesser General Public License (LGPL), possui as seguintes funcionalidades:

- Avaliador de acessibilidade (e-MAG e WCAG);
- Avaliador de CSS;
- Avaliador de HTML (4.01 e XHTML);
- Simuladores de leitor de tela (tempo) e Baixa visão (daltonismo, miopia, catarata);
- Ferramenta para selecionar o DocType, conteúdo alternativo, associador de rótulos, *links* redundantes, corretor de eventos e preenchimento de formulários.

Também exibe um manual, no menu ajuda, contendo, entre outras informações, explicações sobre cada funcionalidade.

Atualmente, está em processo de atualização, mas está disponibilizado nas versões 2.0 para Windows e 1.0.2 para Windows e Linux, todas para ambiente desktop.

Este projeto utilizará a versão 1.0.2 Linux e será estudado, apenas, a funcionalidade Avaliador de acessibilidade, vista com mais detalhes na próxima seção.

##### **1.2.11.1 A funcionalidade Avaliador de acessibilidade**

Dentre as funcionalidades do ASES, é relevante, para este projeto, o Avaliador de acessibilidade. Esta funcionalidade procura, a partir de uma URL, por ocorrências de elementos, em páginas web, que ferem as recomendações de acessibilidade eMAG ou WCAG.

O usuário pode solicitar que a avaliação alcance apenas a página da URL, ou então, o site (com a opção de limitar a profundidade na hierarquia de diretório). Também avalia arquivo ou de diretório local. O resultado é um relatório de avaliação por página avaliada.

Este relatório exibe:

- Segundo a recomendação escolhida, erros e avisos com níveis de prioridade 1, 2 e 3. Erros, são os pontos, da página avaliada, considerados inacessíveis para o nível de prioridade, e avisos, são pontos em que a acessibilidade pode ser garantida ou facilitada no nível de prioridade. Estes erros e avisos são agrupados em casos gerais. Cada caso geral, informa a quantidade de erros ou avisos que ocorreram dele, e as linhas de ocorrência;
- Uma área chamada Saiba Mais, que exibe mais informações sobre um caso geral selecionado;
- Uma guia, chamada Código Original, contendo o documento HTML da URL, com linhas numeradas;
- Uma guia, chamada Código em Edição, com documento HTML da URL, com linhas numeradas, editável, para que o usuário possa realizar e testar as devidas recomendações no documento. O documento alterado pode ser salvo ou revertido para o formato original.
- Basicamente, funciona da seguinte forma, o usuário solicita uma avaliação, o sistema exibe uma lista de relatórios resumidos de avaliação (um para cada URL do domínio da URL de entrada). Para visualizar o relatório completo, de um item desta lista, basta dar duplo clique no relatório desejado. Nele, para visualizar, na área Saiba Mais, mais informações sobre determinado caso geral, é só clicar duas vezes sobre este. Para visualizar a linha de ocorrência, de um determinado erro ou aviso de um item da lista de casos gerais, tanto na guia Código Original quanto na Código em Edição, basta clicar duas vezes no número da linha.

Os dados, gerados da avaliação, são armazenados em arquivos numa pasta temporária chamada temp. Esta pasta é melhor detalhada na seção 1.2.11.2.

A Figura 1 mostra a tela de entrada, Avaliador de Acessibilidade, com o campo nível de avaliação expandido. Nela, o usuário deve: marcar qual tipo de avaliação; escolher o nível

de avaliação (contém as opções Nível 1, Nível 2, Nível 3, Site, Página, Arquivo e Diretório); definir os níveis de prioridade; informar a URL e; clicar no botão executar.

Figura 1 – Tela Avaliador de Acessibilidade

The screenshot shows the ASES web application interface. At the top, there is a menu bar with options: Arquivo, Editar, Avaliadores, Simuladores, Ferramentas, and Ajuda. Below the menu bar is the ASES logo and the text 'Avaliador e Simulador para Acessibilidade de Sites'. To the right of the logo is the gov.br logo and the text 'Acessibilidade Brasil'. The main form has several sections: 'Tipo de Avaliação' with radio buttons for WCAG and E-MAG; 'Níveis a Avaliar' with a dropdown menu for Nível and a list of options: Página, Nível 1, Nível 2, Nível 3, Site, Arquivo, and Diretório; 'Nível de Prioridades' with checkboxes for PRIORIDADE 1, PRIORIDADE 2, and PRIORIDADE 3; and a 'URL' input field with an 'Executar' button below it.

Fonte – O autor

Após o usuário clicar no botão executar, a avaliação é iniciada, sendo apresentada a tela Resumo do Relatório, mostrada na Figura 2.

A imagem exibe o resumo de relatório de avaliação do site da URL <http://www.fatecpe.com.br>. É possível notar, no canto inferior esquerdo e no último item da lista, que foram 201 URLs visitadas, mas apenas 188 avaliadas. Cada linha é um resumo de relatório de uma URL do domínio.

Como pode ser visto, cada resumo de relatório, apresenta: um número de ordem; a URL, e; as quantidades de erros e avisos para cada nível de prioridade. Se o usuário desejar ver detalhes da avaliação de uma URL específica, basta clicar duas vezes no resumo de relatório referente e a tela Relatório da URL é exibida. A Figura 3 mostra a tela.

Nesta tela é exibido:

- Uma área de código HTML, com linhas numeradas, nas abas Código Em Edição (que permite modificação no código desta aba) e Código Original (apenas para visualização do código sem modificação);

Figura 2 – Tela Resumo do Relatório

Arquivo Editar Avaliadores Simuladores Ferramentas Ajuda

**ASES**  
Avaliador e Simulador para Acessibilidade de Sítios  
**Avaliador e Simulador de Acessibilidade para Sítios**

Resumo do Relatório:

URL	Avisos P1	Erros P1	Avisos P2	Erros P2	Avisos P3	Erros P3
157: http://www.fatecpe.com.br/google-procura-recem-tomados-para-treinamento-remunerado/	195	34	43	25	77	0
158: http://www.fatecpe.com.br/p=303	187	34	43	25	69	0
159: http://www.fatecpe.com.br/p=288	186	34	42	25	68	0
160: http://www.fatecpe.com.br/ola-mundo/?replyto=13	198	36	47	24	81	0
161: http://www.fatecpe.com.br/ola-mundo/?replyto=15	198	36	47	24	81	0
162: http://www.fatecpe.com.br/mapa-do-site/	285	34	43	25	166	0
163: http://www.fatecpe.com.br/author/carlos-hallan/	124	18	20	1	66	0
164: http://www.fatecpe.com.br/category/noticias/page/2/	359	37	75	30	229	2
165: http://www.fatecpe.com.br/inscricao-pos-graduacao-fatec-pe/?replyto=68	213	39	55	29	91	0
166: http://www.fatecpe.com.br/inscricao-pos-graduacao-fatec-pe/?replyto=76	213	39	55	29	91	0
167: http://www.fatecpe.com.br/inscricao-pos-graduacao-fatec-pe/?replyto=136	213	39	55	29	91	0
168: http://www.fatecpe.com.br/inscricao-pos-graduacao-fatec-pe/?replyto=72	213	39	55	29	91	0
169: http://www.fatecpe.com.br/inscricao-pos-graduacao-fatec-pe/?replyto=139	213	39	55	29	91	0
170: http://www.fatecpe.com.br/microsoft-dobra-valor-do-beneficio-de-computacao-em-nuvem-para...	186	35	43	25	68	0
171: http://www.fatecpe.com.br/google-ira-realizar-primeira-conferencia-sobre-computacao-ubiqua/	186	35	43	25	68	0
172: http://www.fatecpe.com.br/internet-brasileira-completa-20-anos/	186	35	43	25	68	0
173: http://www.fatecpe.com.br/na-europa-mulheres-representam- apenas-9-dos-criadores-de-aplic...	186	35	43	25	68	0
174: http://www.fatecpe.com.br/427-2/	186	35	43	25	68	0
175: http://www.fatecpe.com.br/a-importancia-de-dominar-o-portugues-para-o-sucesso-na-carreira/	187	34	43	25	69	0
176: http://www.fatecpe.com.br/conheca-jovens-que-empresaram-e-ficaram-ricos/	189	34	43	24	69	0
177: http://www.fatecpe.com.br/guia-de-profissoes-ciencia-da-computacao/	188	36	42	26	68	0
178: http://www.fatecpe.com.br/desafio-profissao-ciencia-da-computacao/	188	36	42	26	68	0
179: http://www.fatecpe.com.br/re-matricula-2015-1/	195	35	42	25	76	0
180: http://www.fatecpe.com.br/internet-brasil-mostra-maior-crescimento-no-mundo-diz-relatorio/	187	34	43	25	69	0
181: http://www.fatecpe.com.br/importancia-da-informatica-na-empresa-nos-tempos-de- hoje/	186	34	42	25	68	0
182: http://www.fatecpe.com.br/presenca-feminina-cresce-800-em-cinco-anos-na-area-de-ti/	195	34	43	26	77	0
183: http://www.fatecpe.com.br/rotina-cientista-da-computacao/	188	36	42	26	68	0
184: http://www.fatecpe.com.br/extensao/	186	34	42	24	68	0
185: http://www.fatecpe.com.br/tecnologo-em-ciencia-da-computacao/	186	34	42	24	68	0
186: http://www.fatecpe.com.br/script/	195	35	42	25	76	0
187: http://www.fatecpe.com.br/um-novo-jeito-de-pensar-em-apresentacoes/	188	36	42	26	68	0
188: http://www.fatecpe.com.br/category/noticias/page/3/	203	31	49	18	97	0

Continuar Pausar 4 Não Avaliadas Finalizar

Avaliação Total: 100% Página Atual: 100% Velocidade: 100%

Página 291 de 291 - Avaliação terminada em: 00:06:55

Fonte – O autor

Figura 3 – Tela Relatório da URL

Arquivo Editar Avaliadores Simuladores Ferramentas Ajuda

**ASES**  
Avaliador e Simulador para Acessibilidade de Sítios  
**Avaliador e Simulador de Acessibilidade para Sítios**

Relatório da URL: http://www.fatecpe.com.br/

Código Em Edição Código Original

Código fonte:

```

0410 <script type="text/javascript" src="http://www.fatecpe.com.br/wp-content/themes/catch-kathmandu/js/catchkathmandu-slider.js?ver=20140317">
0411 <script type="text/javascript" src="http://www.fatecpe.com.br/wp-content/themes/catch-kathmandu/js/catchkathmandu-scrollup.min.js?ver=200720
0412 <script type="text/javascript">
0413 /*  */
0414 var FB_WP=FB_WP||{};FB_WP.queue={_methods:[],flushed:false,add:function(fn){FB_WP.queue.flushed?fn():FB_WP.queue._methods.push(fn)},flu
0415 /* ]&gt; */
0416 &lt;/script&gt;
0417 &lt;script type="text/javascript"&gt;(function(d,s,id){var js,fjs=d.getElementsByTagName(s)[0];if(!d.getElementById(id)){return;}js=d.createElement(s);
0418 &lt;div id="fb-root"&gt;&lt;/div&gt;
0419 &lt;/body&gt;
0420 &lt;/html&gt;
0421
</pre>
<p><b>Saiba Mais</b></p>
<p>Neste canto da tela o usuário encontrará informações sobre os erros e avisos encontrados pelo programa. Para ter acesso a estas informações, basta clicar sobre o Ponto de Verificação (P.V.) correspondente ao erro.</p>
<table border="1">
<thead>
<tr>
<th>Erros Prioridade 1</th>
<th>Avisos Prioridade 1</th>
<th>Erros Prioridade 2</th>
<th>Avisos Prioridade 2</th>
<th>Avisos Prioridade 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>P.V.</td>
<td>CASOS GERAIS</td>
<td>OCORRÊNCIAS</td>
<td colspan="2">LINHAS</td>
</tr>
<tr>
<td>1.11</td>
<td>Fornecer um equivalente textual a cada imagem (isso abrange: representações gráficas do texto, incluindo símbolos, GIFs animados, imagens utilizadas como sinalizadores de pontos de enumeração, espaçadores e botões gráficos), para tanto, utiliza-se o atributo "alt" ou "longdesc" em cada imagem. (Saiba mais)</td>
<td>18</td>
<td>28 34 58 59 63 106 383 395<br/>396 401 402 403 404 405 410 411<br/>412 417</td>
<td></td>
</tr>
</tbody>
</table>
<p>Salvar Tela Anterior Reverter</p>
<p>Página 291 de 291 - Avaliação terminada em: 00:06:55</p>
</div>
<div data-bbox="459 838 583 853" data-label="Text">Fonte – O autor</div>
```

- Uma área que exibe os detalhes dos erros e avisos inerentes a cada nível de prioridade, listando as regras em que ocorreram os erros ou avisos. Cada item da lista, contém:
  - O número da regra infringida, indicado pelo campo P.V<sup>13</sup>;
  - Descrição resumida do erro ou aviso ocorrido, indicado pelo campo CASOS GERAIS;
  - A quantidade de ocorrências do erro ou aviso, indicado pelo campo OCORRÊNCIAS, e;
  - O número das linhas onde ocorreram os erros ou avisos, indicado pelo campo LINHAS. Neste campo, ao clicar num determinado número de linha, é selecionada e exibida a linha correspondente na área de código HTML.
- A área Saiba Mais é utilizada para mostrar detalhes da recomendação inerentes ao item da lista de erros ou avisos. Basta clicar no campo P.V. ou CASOS GERAIS do item da lista;
- O botão Salvar que é exibido na visualização do código HTML em edição, serve para armazenar em arquivo o código editado;
- O botão Reverter, somente exibido com a guia Código em Edição selecionada, serve para recarregar o código original nesta guia, e;
- O botão Tela Anterior, que retorna a janela de resumo de relatórios.

#### *1.2.11.2 Pasta de arquivos temporários*

O programa tem uma pasta, temporária, para armazenar arquivos referentes a avaliação corrente. Isto é, a cada solicitação de avaliação, o conteúdo desta pasta é destruído para receber o novo. Dentre outros, são gerados:

- Um arquivo XML com as configurações da avaliação, nomeado avaliacaoConf.xml;

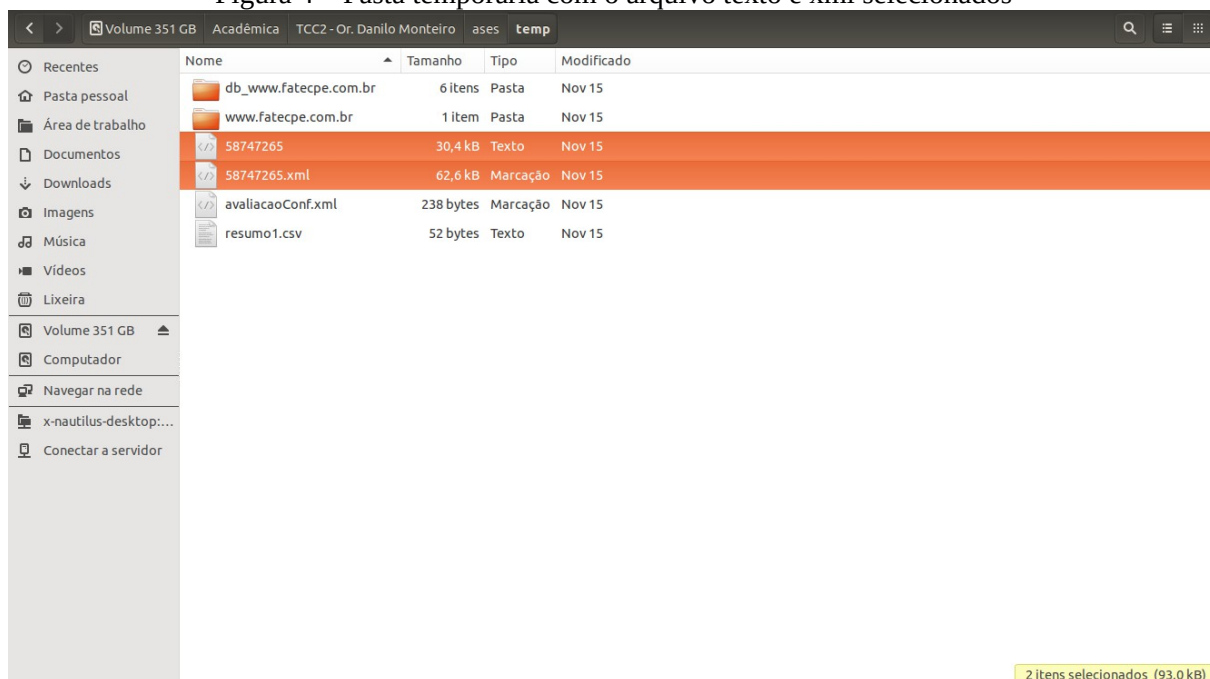
---

<sup>13</sup> Ponto de Verificação

- Arquivos Comma-Separated Values (CSV)<sup>14</sup> (um para cada 50 URLs), nomeados resumoN.csv, onde N é o número de ordem do arquivo;
- Para cada URL do domínio, dois arquivos com nomes iguais:
  - Um arquivo de texto com o conteúdo HTML capturado, e;
  - Um arquivo XML, com dados inerentes a avaliação do conteúdo capturado. Este arquivo é objeto de estudo na seção 2.2.2.1.

A Figura 4 mostra o conteúdo da pasta temporária, resultante da avaliação com uma URL, com os dois arquivos, dela, selecionados. O arquivo 58747265, de 30,4 kB<sup>15</sup> de tamanho, contendo o conteúdo HTML capturado, e o 58747265.xml, de 62,6 kB, com os dados da avaliação do conteúdo capturado.

Figura 4 – Pasta temporária com o arquivo texto e xml selecionados



Fonte – O autor

### 1.2.12 Código fonte do ASES

Existe um código fonte disponível em Acessibilidade Brasil (2010).

Porém, não há claras referências com qualquer das versões distribuídas pelo governo brasileiro. Mas, comparando a avaliação de acessibilidade com a da versão 1.0.2 Linux,

<sup>14</sup> <https://tools.ietf.org/html/rfc4180>

<sup>15</sup> Kilobyte



observa-se que as telas de entrada solicitam os mesmos dados e, os dados dos resultados são iguais e o visual, muito semelhante.

A Figura 1 pode ser comparada a Figura 5 e mostram, respectivamente, as telas de entrada da avaliação de acessibilidade da versão 1.0.2 Linux e do código fonte.

A Figura 4 pode ser comparada a Figura 6 e mostram, respectivamente, as telas de resultado, para a mesma avaliação, da versão 1.0.2 Linux e do código fonte.

A visualização da tela de relatório da URL, também é muito semelhante à do ASES 1.0.2 Linux. O que pode ser observado comparando-se a Figura 3 com a Figura 7.

Figura 5 – Tela Avaliador de Acessibilidade do código fonte

Fonte – O autor

A avaliação de acessibilidade do código fonte não foi comparada com as versões para *Windows*, por este não ser o sistema operacional utilizado pelo pesquisador.

Figura 6 – Tela de Resumo do Relatório do código fonte

Arquivo Editar Avaliadores Simuladores Ferramentas Ajuda

**Avaliador e Simulador de Acessibilidade para Sítios**

Resumo do Relatório:

URL	Avisos P1	Erros P1	Avisos P2	Erros P2	Avisos P3	Erros P3
153: http://www.fatecpe.com.br/setor-de-ti-crescimento-deve-ser-potencializado-por-solucoes-empr...	187	34	43	25	69	0
154: http://www.fatecpe.com.br/p=328	188	36	42	26	68	0
155: http://www.fatecpe.com.br/p=316	188	36	42	26	68	0
156: http://www.fatecpe.com.br/o-futuro-da-medicina-esta-no-seu-smartphone/	187	34	43	25	69	0
157: http://www.fatecpe.com.br/p=303	187	34	43	25	69	0
158: http://www.fatecpe.com.br/google-procura-recem-formados-para-treinamento-remunerado/	195	34	43	25	77	0
159: http://www.fatecpe.com.br/p=288	186	34	42	25	68	0
160: http://www.fatecpe.com.br/ola-mundo/?replyto=13	198	36	47	24	81	0
161: http://www.fatecpe.com.br/ola-mundo/?replyto=15	198	36	47	24	81	0
162: http://www.fatecpe.com.br/author/carlos-hallan/	124	18	20	1	66	0
163: http://www.fatecpe.com.br/mapa-do-site/	285	34	43	25	166	0
164: http://www.fatecpe.com.br/inscricao-pos-graduacao-fatec-pe/?replyto=72	213	39	55	29	91	0
165: http://www.fatecpe.com.br/category/noticias/page/2/	359	37	75	30	229	2
166: http://www.fatecpe.com.br/inscricao-pos-graduacao-fatec-pe/?replyto=68	213	39	55	29	91	0
167: http://www.fatecpe.com.br/inscricao-pos-graduacao-fatec-pe/?replyto=136	213	39	55	29	91	0
168: http://www.fatecpe.com.br/inscricao-pos-graduacao-fatec-pe/?replyto=139	213	39	55	29	91	0
169: http://www.fatecpe.com.br/inscricao-pos-graduacao-fatec-pe/?replyto=76	213	39	55	29	91	0
170: http://www.fatecpe.com.br/google-ira-realizar-primeira-conferencia-sobre-computacao-ubiqua/	186	35	43	25	68	0
171: http://www.fatecpe.com.br/microsoft-dobra-valor-do-beneficio-de-computacao-em-nuvem-para...	186	35	43	25	68	0
172: http://www.fatecpe.com.br/internet-brasileira-completa-20-anos/	186	35	43	25	68	0
173: http://www.fatecpe.com.br/na-europa-mulheres-representam-apenas-9-dos-criadores-de-aplic...	186	35	43	25	68	0
174: http://www.fatecpe.com.br/427-2/	186	35	43	25	68	0
175: http://www.fatecpe.com.br/a-importancia-de-dominar-o-portugues-para-o-sucesso-na-carreira/	187	34	43	25	69	0
176: http://www.fatecpe.com.br/conheca-jovens-que-empresenderam-e-ficaram-ricos/	189	34	43	24	69	0
177: http://www.fatecpe.com.br/guia-de-profissoes-ciencia-da-computacao/	188	36	42	26	68	0
178: http://www.fatecpe.com.br/desafio-profissao-ciencia-da-computacao/	188	36	42	26	68	0
179: http://www.fatecpe.com.br/internet-brasil-mostra-maior-crescimento-no-mundo-diz-relatorio/	187	34	43	25	69	0
180: http://www.fatecpe.com.br/re-matricula-2015-1/	195	35	42	25	76	0
181: http://www.fatecpe.com.br/importancia-da-informatica-na-empresa-nos-tempos-de-hoje/	186	34	42	25	68	0
182: http://www.fatecpe.com.br/pressencia-feminina-cresce-800-em-cinco-anos-na-area-de-ti/	195	34	43	26	77	0
183: http://www.fatecpe.com.br/rotina-cientista-da-computacao/	189	36	42	26	68	0
184: http://www.fatecpe.com.br/extensao/	186	34	42	24	68	0
185: http://www.fatecpe.com.br/tecnologo-em-ciencia-da-computacao/	186	34	42	24	68	0
186: http://www.fatecpe.com.br/um-novo-jeito-de-pensar-em-apresentacoes/	188	36	42	26	68	0
187: http://www.fatecpe.com.br/script/	195	35	42	25	76	0
188: http://www.fatecpe.com.br/category/noticias/page/3/	203	31	49	18	97	0

Continuar Pausar 4 Não Avaliadas Finalizar

Avaliação Total: 100% Página Atual: 100% Velocidade: 100%

Página 291 de 291 - Avaliação terminada em: 00:02:30

Fonte – O autor

Figura 7 – Tela de Relatório da URL do código fonte

Arquivo Editar Avaliadores Simuladores Ferramentas Ajuda

**Avaliador e Simulador de Acessibilidade para Sítios**

Relatório da URL: http://www.fatecpe.com.br/

Código Em Edição Código Original

Código fonte:

```

0407 var js_value = {"transition_effect":"blindX","transition_delay":"4000","transition_duration":"3000"};
0408 /* ]]> */
0409 </script>
0410 <script type="text/javascript" src="http://www.fatecpe.com.br/wp-content/themes/catch-kathmandu/s/catchkathmandu-slider.js?ver=20140317">
0411 <script type="text/javascript" src="http://www.fatecpe.com.br/wp-content/themes/catch-kathmandu/s/catchkathmandu-scrollup.min.js?ver=200720
0412 <script type="text/javascript">
0413 /* <![CDATA[ */
0414 var FB_WP=FB_WP||{};FB_WP.queue={methods:[],flushed:false,add:function(fn){FB_WP.queue.flushed?fn():FB_WP.queue_methods.push(fn),flu
0415 /* ]]> */
0416 </script>
0417 <script type="text/javascript">(function(d,s,id){var js,fjs=d.getElementsByTagName(s)[0];if(!d.getElementById(id)){return}js=d.createElement(s);
0418 <div id="fb-root"></div>
0419 </body>
0420 </html>
0421

```

**Saiba Mais**

Neste canto da tela o usuário encontrará informações sobre os erros e avisos encontrados pelo programa. Para ter acesso a estas informações, basta clicar sobre o Ponto de Verificação (P.V.) correspondente ao erro.

Erros Prioridade 1	Avisos Prioridade 1	Erros Prioridade 2	Avisos Prioridade 2	Avisos Prioridade 3
P.V.	CASOS GERAIS	OCORRÊNCIAS	LINHAS	
1.6	Organizar os documentos de tal forma que possam ser lidos sem recurso de folhas de estilo. Por exemplo, se um documento em HTML for reproduzido sem as folhas de estilo que lhe estão associadas, deve continuar a ser possível. (Saiba mais)	13	17 18 26 27 31 32 33 38	60 61 72 73 82
1.4	Assegurar que todas as informações veiculadas com cor estejam também disponíveis sem cor. (Saiba mais)	13	17 18 26 27 31 32 33 38	60 61 72 73 82
	Criar uma sequência lógica de tabulação para percorrer links, controles de formulários e objetos.		105 111 122 123 124 125 126 127	139 140 141 142 144 145 146 147

Salvar Tela Anterior Reverter

Página 291 de 291 - Avaliação terminada em: 00:02:30

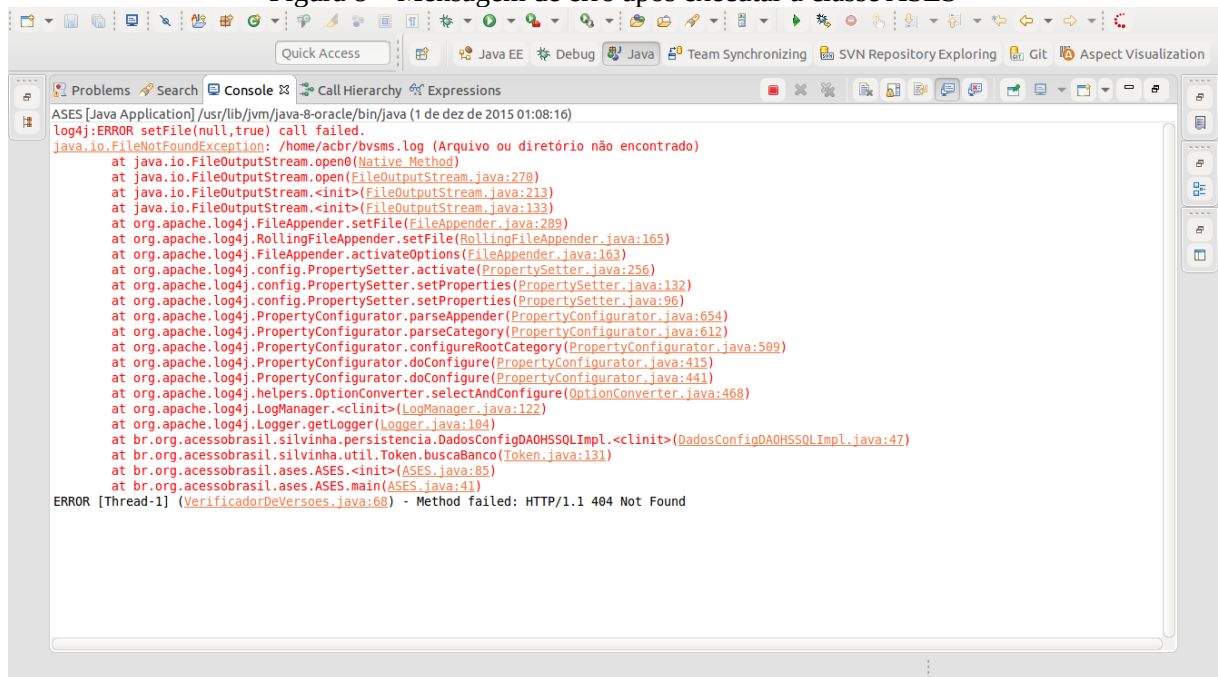
Fonte – O autor

### 1.2.12.1 Enfrentamentos para executar o código fonte do ASES

Depois de importar o código fonte para o eclipse, foi necessário convertê-lo para projeto Maven<sup>16</sup>. Ação realizada clicando-se, com o botão direito do mouse, no projeto e, em seguida, escolhendo-se o item *Configure*, clicar na opção *Convert to Maven project*.

Para iniciá-lo, executou-se, como *Java Application*, a classe ASES do pacote *br.org.acessobrasil.ases*. Uma mensagem de erro foi exibida no console, vista na Figura 8. Sendo necessário apenas desconsiderá-la e aguardar que uma tela de login seja exibida, Figura 9.

Figura 8 – Mensagem de erro após executar a classe ASES



Fonte – O autor

<sup>16</sup> <https://maven.apache.org/>

Figura 9 – Tela de login exibida na execução do código fonte do ASSES

Fonte – O autor

Nesta tela de login bastou digitar qualquer coisa nos campos usuário e e-mail, clicar, exclusivamente com o mouse, no botão salvar, aguardar uma caixa de mensagem de sucesso, clicar no botão OK.

Após isto, a tela Avaliador de Acessibilidade foi exibida, procedendo-se com a entrada dos dados, para realizar uma avaliação, e clicando no botão executar, foi mostrado uma mensagem de erro indicando falha de inicialização no banco de dados e pedindo para reinicializar o programa.

Apertando-se em OK, o programa continuou aparentemente normal mas, quando acessou-se a tela de Relatório da URL, a exibição do campo Saiba Mais não estava funcionando corretamente.

Para reiniciar o programa, bastou voltar a tela Avaliador de Acessibilidade. O que, realmente, normalizou a função do campo Saiba Mais.

#### 1.2.12.2 Abstração lógica da avaliação de acessibilidade

Realizando depuração, foi possível compreender a sequência lógica da avaliação de acessibilidade e identificar as principais classes desta funcionalidade. A Figura 10 mostra um diagrama com elas.

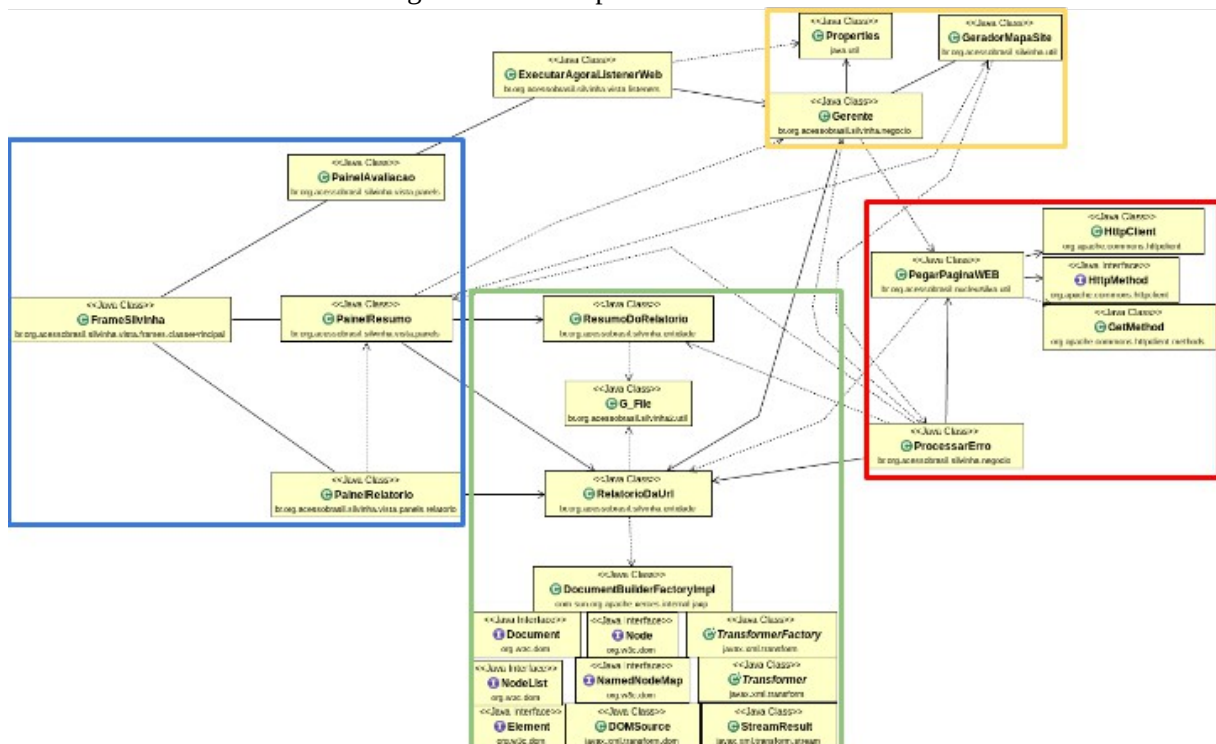
O quadro verde contém as principais classes modelos e as principais classes manipuladoras de arquivos. A classe `ResumoDoRelatorio`, é o tipo para os dados de um resumo de relatório. Para armazenar ou recuperar estes dados ela utiliza a classe `G_File`.

A classe `RelatorioDaUrl` é responsável por modelar um relatório de avaliação de uma URL. Administra o armazenamento e recuperação do conteúdo HTML em arquivo, através da classe `G_File` e, do arquivo xml, com os dados da avaliação, através do grupo de classes visualizados, logo abaixo dela, no quadro.

No quadro azul, estão englobadas as principais classes de visualização de usuário. A classe `FrameSilvinha` é um container para outras classes de visualização, das quais, as três principais, para esta funcionalidade, são:

- `PainelAvaliacao`, nela o usuário informa os dados necessários e solicita avaliação de acessibilidade;
- `PainelResumo`, exibe o resumo da avaliação;
- `PainelRelatorio`, exibe o relatório da avaliação.

Figura 10 – Principais classes do ASES



Fonte – O autor

A classe isolada, fora de todos os quadros, é a classe *ExucutarAgoraListenerWeb*. Ela é responsável por pegar os dados da classe *PainelAvaliação* e iniciar a avaliação com as classes do quadro amarelo.

O quadro amarelo, contém as principais classes responsáveis por gerenciar a avaliação. A classe *Gerente* recebe os dados para realizar a avaliação de uma URL, preparando um objeto do tipo *RelatorioDaUrl* com os critérios da avaliação. A classe *GerenciarMapaSite* administra a avaliação de um conjunto de URLs. Quando um *Gerente* termina uma avaliação *GerenciarMapaSite* lhe dá a próxima URL.

No quadro vermelho, estão as classes que acessam as páginas, capturam o conteúdo HTML e realizam a avaliação. A classe *PegarPaginaWeb* recebe da classe *Gerente* um objeto do tipo *RelatorioDaUrl* pega a URL contida nele e utiliza as classes da biblioteca *Java Commons HttpClient*<sup>17</sup> para visitar a página e capturar o conteúdo HTML que é passado para o objeto modelo.

A classe *ProcessarErro*, recebe de *Gerente*, um *RelatorioDaUrl*. Solicita, deste, o seu conteúdo HTML, neste, realiza uma busca por URLs e procede com a avaliação propriamente dita, modelando todos os dados da avaliação no objeto. Também passa para a classe *ResumoDeRelatorio* os dados necessários para que esta armazene-os.

As classes *Gerente*, *GerenciadorMapaSite* e *ProcessarErro* interagem dinamicamente com a classe *PainelResumo*, atualizando os dados na tela do usuário, automaticamente, a medida que a avaliação flui.

Assim, o fluxo de avaliação de acessibilidade do ASES, apresenta, ao usuário, o primeiro momento, onde visualiza os resumos de relatórios, e o segundo, a partir do clique em um dos resumos, em que visualiza os detalhes do relatório do resumo clicado.

#### 1.2.12.2.1 Lógica abstraída de primeiro momento

1. Carregar, manualmente, URL para avaliação, tipo de avaliação, nível de avaliação e os níveis de prioridade;
2. Identificar domínio da URL;
3. Criar lista de URLs do domínio;
4. Inserir URL na lista de URLs do domínio;

---

<sup>17</sup> <<http://hc.apache.org/httpclient-3.x/>>. Acessado em 26 nov.2015.

5. Para cada URL na lista de URLs do domínio:

5.1. Identificar URL como avaliável:

- a) Marcar URL como avaliável se, e somente se:
- b) Não está marcada como avaliada;
- c) Segue padrões de URL;
- d) Concorda com critério nível de avaliação;
- e) Está conectável.

5.2. Se a URL é avaliável:

- a) Capturar o conteúdo da página web da URL;
- b) Proceder com a avaliação no conteúdo capturado, realizando:
  - Avaliação propriamente dita, conforme os critérios: tipo de avaliação e níveis de prioridade, confeccionando relatório de avaliação de URL;
  - Identificação de URLs do domínio, inserindo-as na lista de URLs do domínio.
- c) Armazenar relatório de avaliação da URL;
- d) Gerar relatório resumido, adicionando-o na lista de relatórios resumidos;
- e) Marcar URL como avaliada.

6. Exibir lista de relatórios resumidos.

1.2.12.2.2 Lógica abstraída de segundo momento

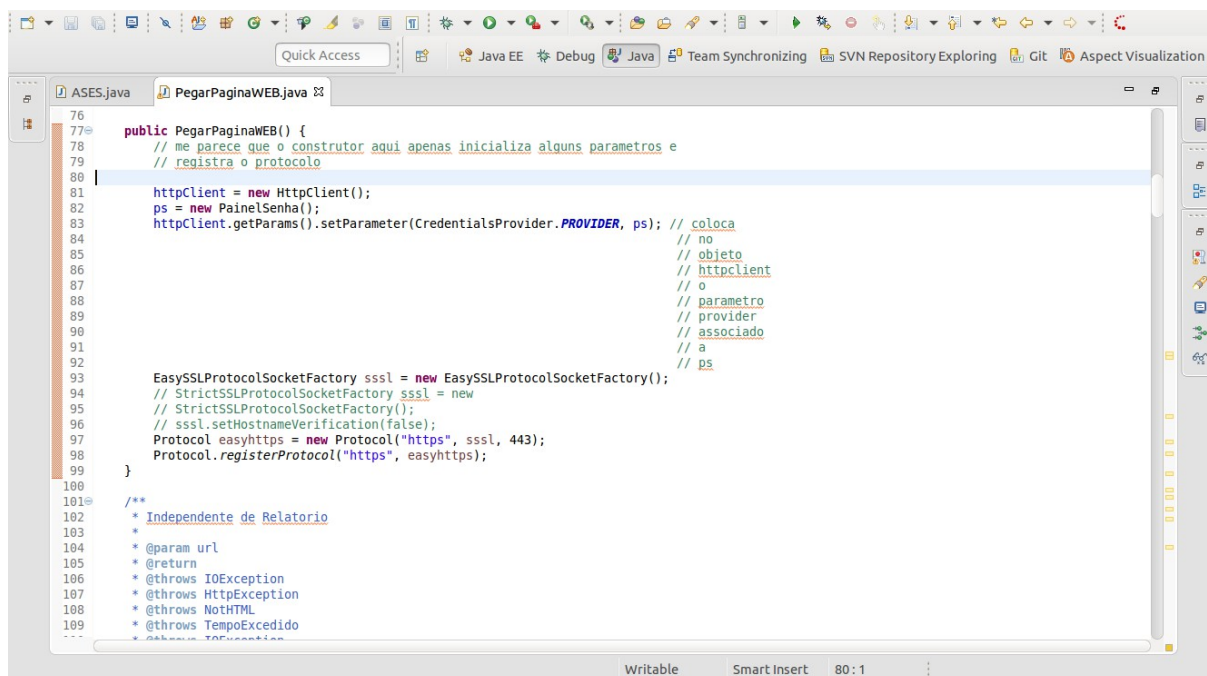
- 1. Clicar em resumo de relatório;
- 2. Buscar relatório detalhado;
- 3. Exibir relatório detalhado.

### 1.2.12.3 Motor de busca do ASES

O motor de busca do ASES é utilizado para buscar o conteúdo HTML da página de uma URL. Esta rotina encontra-se implementada na classe PegarPaginaWeb do pacote br.org.acessobrasil.nucleuSilva.util e faz uso da biblioteca *Java Commons HttpClient*<sup>18</sup>. A Figura 11 mostra, no trecho de código original, o construtor, onde esta biblioteca é inicializada.

No método getContent, o conteúdo da página é carregado por referência num objeto do tipo RelatorioDaUrl, do pacote br.org.acessobrasil.silvinha.entidade. Dentro deste método, para acessar a página da URL, o método executMethod da instância da classe HttpClient, executa um objeto do tipo GetMethod que recebe como parâmetro a URL. Este método retorna um inteiro que é utilizado, juntamente, com outros dados da instância GetMethod, para realizar testes de tipo de documento, existência de conteúdo na página, tamanho do conteúdo da página, etc.

Figura 11 – Trecho de código original da classe PegarPaginaWeb: construtor implementando a biblioteca *Java Commons HttpClient*



Fonte – O autor

Após os testes, o método getResponseBodyAsString da instância GetMethod retorna o conteúdo HTML que é armazenado, como cadeia de caracteres, no objeto do tipo

<sup>18</sup> <<http://hc.apache.org/httpclient-3.x/>>. Acessado em 26 nov.2015.



StringBuilder. Outros testes são realizados com este objeto e, então, a cadeia de caracteres é passada para o objeto do tipo RelatorioDaUrl, através do método deste, setConteudo.

*Threads* são utilizadas para controlar e testar tempo de acesso.

A busca do conteúdo HTML de várias URLs é gerenciada por grupos de *threads*, cada uma contém um objeto do tipo RelatorioDaUrl e solicita para o motor de busca individualmente.

### **1.2.13 Trabalhos relacionados**

Alguns pesquisadores têm procurado realizar melhorias em questões específicas, tais como apresenta Soares Filho (2015). Ele procura atingir a acessibilidade em softwares que gerenciam e-mails pretendendo solucionar questões levantadas a partir de entrevistas realizadas diretamente com pessoas que trabalham utilizando serviços de e-mail e que são deficientes visuais. A dissertação é de grande relevância pelo fato de apresentar detalhamento sobre as bases das teorias da acessibilidade e demonstrar casos reais de deficiências das soluções tecnológicas atuais.

## 2 DESENVOLVIMENTO

Nesta seção apresentam-se: (i) a metodologia do projeto, discorrendo sobre as abordagens e tipo desta pesquisa; (ii) os resultados, sobre o IsAcessível, citando as tecnologias utilizadas para desenvolvê-lo, as restrições, o diagrama de caso de uso, juntamente, com as telas de usuário, as principais classes; (iii) a coleta de dados, onde primeiramente, busca-se entender o arquivo xml que contém o relatório de avaliação, depois, levantar algumas considerações sobre os dados neste arquivo, então, tentar criar configurações de arquivagem para serem posteriormente testadas no IsAcessível, montando, no final, uma tabela com os dados finais; (iv) discussão, são dispostas algumas opiniões em torno de pontos do projeto.

### 2.1 METODOLOGIA

O atingimento do objetivo principal e a resolução dos objetivos específicos serão orientados com abordagem diagnóstica e tipo de pesquisa analítico. Quanto a coleta de dados utilizar-se-á abordagem quanti-qualitativa.

#### 2.1.1 Abordagem diagnóstica

Segundo Marques et al (2006, p. 38), “[...] as abordagens em pesquisa referem-se aos procedimentos de aproximação do objeto de pesquisa, em termos de coletas, análise e interpretação de dados”. Uma das abordagens classificadas é chamada de abordagem diagnóstica que “[...] envolve conhecer como é o objeto de pesquisa. Implica analisá-lo minudentemente e dizer o mais detalhada e sistematicamente todas suas características” (MARQUES et al, 2006, p. 39).

Neste projeto, tem-se como objeto de pesquisa a funcionalidade de avaliação de acessibilidade do programa ASES que será analisada para compreensão e construção desta funcionalidade abstraída de ambiente *desktop* ou *web*.

#### 2.1.2 Tipo de pesquisa analítico

Tipo de pesquisa analítico é definido por Marques et al (2006, p. 52) como,

[...] tipo de estudo que visa [...] analisar uma dada situação [...], mediante procedimentos de decomposição do todo estudado, visando não apenas conhecer seus elementos constituintes, mas sobre tudo como eles se articulam entre si.

A estrutura do algoritmo do programa ASES será decomposta para compreender-se como suas camadas lógicas estão em harmonia para utilizar-se o essencial ao objetivo do projeto.

### **2.1.3 Abordagem quanti-qualitativa**

Os dados coletados serão analisados sob os aspectos quantitativos e qualitativos, com foco nos quantitativos, conforme Marques et al (2006).

O foco quantitativo prevalece para considerar um dos objetivos específicos deste projeto, a saber, o gerenciamento sustentável do espaço de memória para uma aplicação *web*, que buscará, nos dados coletados, o menor espaço. Quanto a qualidade dos dados serão observados aspectos de manutenção de dados inerentes a avaliação de acessibilidade.

## **2.2 RESULTADOS**

Nesta seção, aborda-se o IsAcessível, inicia-se listando as tecnologias utilizadas no seu desenvolvimento e suas restrições. Após, exibe-se o diagrama de caso de uso e as telas de usuário, além das principais classes e suas responsabilidades, culminando com a classe que gerencia os arquivos. Então, procura-se entender o arquivo xml de relatório de avaliação do ASES para possibilitar inferências sobre ele. A fim de criar configurações de arquivamento e testá-las, no IsAcessível, realizando comparações com os resultados que são registrados em uma tabela para tomada de decisões sobre a configuração de arquivamento mais adequada para o IsAcessível.

### **2.2.1 O IsAcessível**

O IsAcessível é projetado para ser uma aplicação web que realiza avaliações de acessibilidade para web, segundo os padrões de recomendação eMAG ou WCAG.

A ideia do projeto é permitir que qualquer pessoa acesse o IsAcessível e execute avaliações de acessibilidade. Criando uma pasta temporária para cada usuário.

Com o intuito de utilizar um motor de busca mais sofisticado, a API crawler4j é utilizada como rastreador web.

Testes com os formatos JSON e XML são realizados, a fim de maximizar a utilização sustentável dos recursos de armazenamento dos arquivos nas pastas temporárias de usuários.

### 2.2.1.1 Tecnologias utilizadas

Desenvolvido sobre plataforma *Java Enterprise Edition* (J2EE)<sup>19</sup>, utilizando *framework* de desenvolvimento *Spring*<sup>20</sup>, servidor *web* Tomcat<sup>21</sup>. Para construir a interface do usuário utilizou-se a suíte de componentes para *Java Server Faces* (JSF) *Primefaces*<sup>22</sup>. A API *crawler4j*<sup>23</sup>, para implementar o motor de busca. JSON<sup>24</sup> XML<sup>25</sup>, para estruturação e armazenamento dos dados de arquivos de avaliação. Classes e trechos de código do programa ASES (ACESSIBILIDADE BRASIL, 2008). O ambiente de desenvolvimento utilizado é a IDE Eclipse Mars<sup>26</sup> em sistema operacional Linux – Ubuntu 15.04<sup>27</sup>. Software para produzir o diagrama de caso de uso, Modelador UML Umbrello versão 2.15.3<sup>28</sup> e, para diagramas com classes, o *plugin*, para IDE eclipse, ObjectId UML Explorer<sup>29</sup>.

Para produzir as telas foi utilizado uma pré-configuração desenvolvida em outro projeto do pesquisador.

### 2.2.1.2 Principais restrições

O IsAcessível, nesta versão, avalia apenas páginas e sites online.

### 2.2.1.3 Diagrama de caso de uso

A lógica apresentada na seção 1.2.12.2 é utilizada para visualizar as regras de negócio do programa e montar o diagrama de caso de uso.

A Figura 12 mostra o diagrama de caso de uso.

<sup>19</sup> <http://www.oracle.com/technetwork/java/javaee/documentation/index.html>

<sup>20</sup> <https://spring.io/>

<sup>21</sup> <http://tomcat.apache.org/>

<sup>22</sup> <http://www.primefaces.org/>

<sup>23</sup> <https://github.com/yasserg/crawler4j>

<sup>24</sup> <http://www.ietf.org/rfc/rfc4627.txt>

<sup>25</sup> <http://www.w3.org/TR/2008/REC-xml-20081126/>

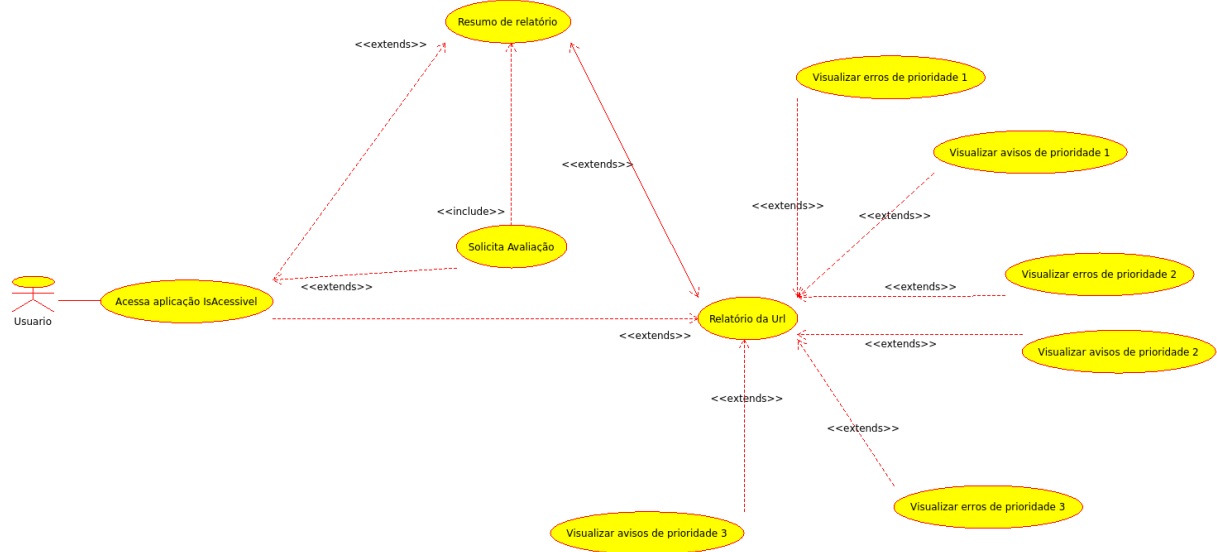
<sup>26</sup> <http://www.eclipse.org/>

<sup>27</sup> <http://www.ubuntu.com/>

<sup>28</sup> <http://umbrello.kde.org/>

<sup>29</sup> <http://www.objectaid.com/>

Figura 12 – Diagrama de caso de uso do IsAcessível



Fonte – O autor

O usuário acessa o IsAcessível, visualizando a tela de entrada de dados que pode ser vista na Figura 13. Nesta tela, são exibidos os campos:

- O menu Ferramentas:
  - Utilizado para voltar a esta tela ou sair do programa.
- Recomendações, com as opções: EMAG e WCAG;
- Profundidade, com as opções:
  - NIVEL 1, 2 e 3, são utilizadas avaliar as URLs considerando a hierarquia de diretório do domínio da URL principal, respectivamente, nos níveis de profundidade 1, 2 e 3;
  - SITE, avalia todas as URLs do domínio da URL principal;
  - PAGINA, avalia somente a URL principal.
- Prioridade 1, Prioridade 2 e Prioridade 3:
  - Definem as prioridades das recomendações que serão visualizadas.
- URL:
  - URL principal da avaliação.

- Botão avaliar:
  - Solicita a avaliação.

Figura 13 – Tela de entrada do IsAcessível

Fonte – O autor

Para realizar uma avaliação, o usuário deve fornecer os dados solicitados na tela de entrada e clicar no botão avaliar.

Após a solicitação, o sistema inicia o processo, e assim que obtém algum resultado, exibe a tela relatório resumido, vista na Figura 14.

Figura 14 – Tela de relatório resumido

Fonte – O autor

Nesta tela são exibidos:

- O menu Ferramentas;
- O botão IsAcessível – Relatório resumido:
  - Ao clicar neste botão, os dados da tela são atualizados. Por exemplo, quando é solicitado uma avaliação diferente da opção PAGINA, um conjunto de URLs, do mesmo domínio da URL principal, serão avaliadas, então, o sistema exibe as primeiras avaliações e continua avaliando. Para visualizar mais avaliações disponíveis deve-se clicar neste campo.
- O campo que informa a recomendação e a profundidade da avaliação;
- A lista de relatórios resumidos:
  - A lista está organizada em páginas, contendo, cada uma, no máximo quatro relatórios resumidos. A quantidade de páginas é proporcional a quantidade de URLs avaliadas.
- Dois campos de navegação em páginas da lista de relatórios resumos, um superior à lista e outro, inferior.

Ao clicar no botão exibir de um item, o usuário visualiza a tela de relatório de avaliação da URL do item, mostrada na Figura 15.

Figura 15 – Tela relatório de avaliação do IsAcessível

Casos Gerais	Ocorrências	Linhas
EMAG: 1.20: Assegurar que todas as páginas possam ser utilizadas mesmo que os programas interpretáveis, os applets ou outros objetos programados tenham sido desativados ou não sejam suportados. Sempre que tiver script associe logo a seguir o elemento "noscript".	Ocorrências: 17	Linhas: [28, 34, 58, 59, 63, 383, 395, 396, 401, 402, 403, 404, 405, 410, 411, 412, 417]
EMAG: 1.11: Fornecer um equivalente textual a cada imagem (isso abrange: representações gráficas do texto, incluindo símbolos, GIFs animados, imagens utilizadas como sinalizadores de pontos de enumeração, indicadores e botões gráficos).	Ocorrências: 1	Linhas: [106]

Fonte – O autor

Nela são mostrados os seguintes campos:

- O menu Ferramentas;
- O botão Voltar, que retorna para a tela relatório resumido;
- Um campo título que também informa o padrão de avaliação;
- O campo URL da avaliação;
- O campo com a quantidade total de ocorrências de erros e avisos da avaliação;
- As abas Prioridade 1, 2 e 3, cada uma informando, na guia, o seu total de erros e avisos;
- A aba Prioridade, que estiver selecionada, exibe suas abas de erros e avisos. Por exemplo, na Figura 17, a aba Prioridade 1 está exibindo suas abas Erros P1 e Avisos P1;
- Cada aba Erro P1, P2 ou P3 mostra, na guia, a quantidade de ocorrências de erros para a prioridade e, no corpo, a lista de Casos Gerais, onde, cada item, é um caso geral de recomendação, para sanar o erro, com a quantidade de ocorrências deste e o número das linhas em que ocorreram;
- Cada aba Aviso P1, P2 ou P3 mostra, na guia, a quantidade de ocorrências de avisos para a prioridade e, no corpo, a lista de Casos Gerais, onde, cada item, é um caso geral de recomendação, para satisfazer o aviso, com a quantidade de ocorrências deste e o número das linhas em que ocorreram.

#### 2.2.1.4 Principais classes

Procurou-se seguir o padrão de arquitetura *Model View Controller (MVC)*<sup>30</sup>. A Figura 16 mostra as principais classes.

Distinguindo a camada de visualização, retângulo azul, em classes responsáveis por manter a comunicação com páginas XHTML, implementadas utilizando componentes Primefaces por esta tecnologia apresentar uma visualização com um nível de organização elegante.

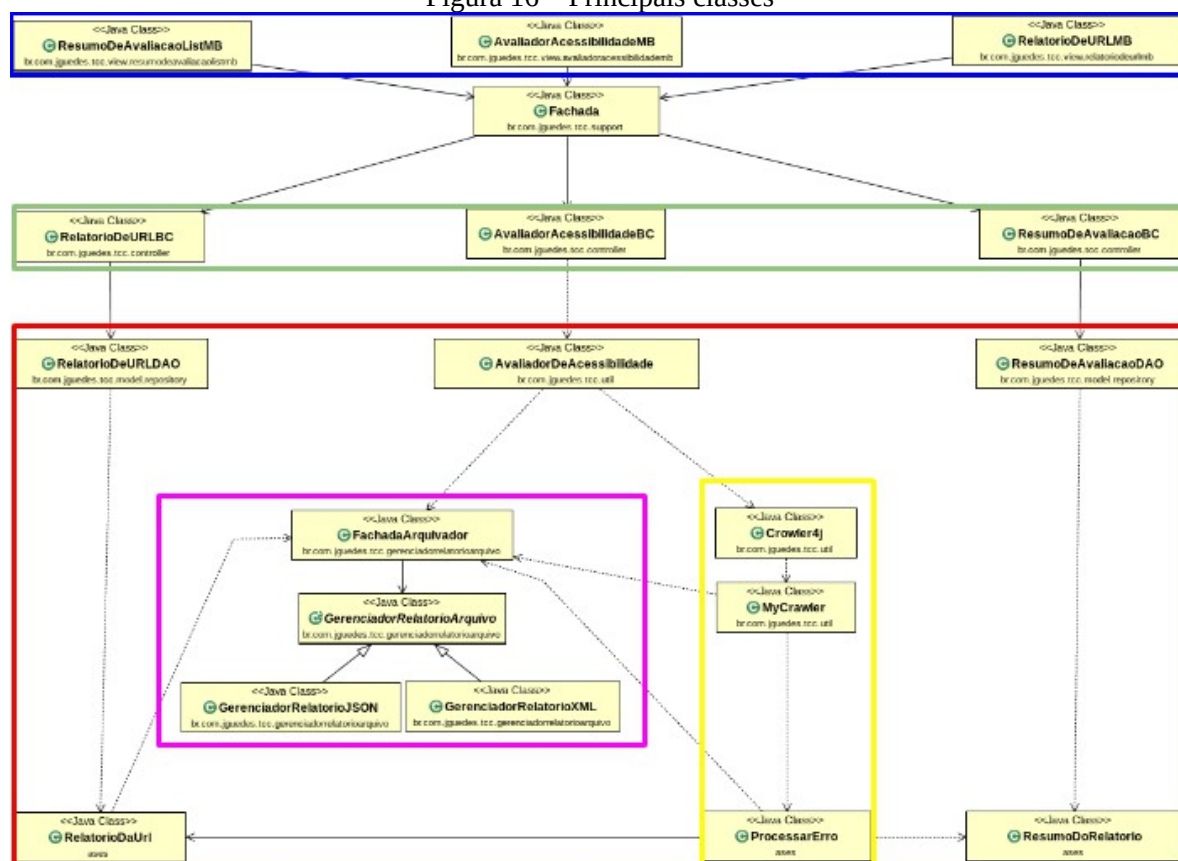
<sup>30</sup> <http://heim.ifi.uio.no/~trygver/themes/mvc/mvc-index.html>



Nesta camada, quando o usuário acessa a tela de entrada do IsAcessível, está se comunicando com a classe AvaliadorAcessibilidadeMB. Ao solicitar a avaliação, os critérios são atribuídos ao contexto do usuário, o processo de avaliação é iniciado e, após os primeiros resultados, é exibida a tela de relatório resumido.

Esta tela, se comunica com a classe ResumoDaAvaliacaoMB e exibe a lista de resumos disponibilizada pela classe. A lista é correspondente ao contexto do usuário atual. Quando o usuário clica no botão exibir, de um item da lista, a classe atribui o código do item ao contexto do usuário e aciona a tela relatório de avaliação.

Figura 16 – Principais classes



Fonte – O autor

A tela relatório de avaliação comunica-se com a classe RelatorioDaUriMB e exibe o relatório da URL entregue pela classe. A classe utiliza o código de item, do contexto atual, para recuperar o relatório.

As classes no retângulo verde são a camada de controle, a comunicação entre estas com as da camada de visual se dão, exclusivamente, por intermédio da classe Fachada.

O retângulo vermelho, engloba as classes responsáveis pelos modelos e dados.

As classes que estão dentro do retângulo amarelo realizam a avaliação propriamente dita. A classe `AvaliadorDeAcessibilidade` cria a pasta temporária do usuário, através da classe `FachadaArquivador`, e instancia a classe `Crawler4j`, passando para ela a URL principal e o caminho da pasta temporária do usuário.

A instância `Crawler4j`, organiza os parâmetros de configuração, incluindo a URL e o caminho da pasta do usuário, para a instância da classe `CrawController`. Esta, aciona a classe `MyCrawler`.

Nos testes de critérios de permissão para visitar páginas, implementados no método `MyCrawler.shouldVisit(Page p, WebUrl w)`, são testados os critérios de profundidade de URL.

No método `MyCrawler.visit(Page p)`, o conteúdo HTML é capturado, nomeado com um código e armazenado, em arquivo, pela classe `FachadaArquivador`. Uma instância da classe `RelatorioDaUrl` é carregada com este código e com os critérios de avaliação. E, com esta instância e o contexto do usuário, um objeto do tipo `ProcessarErro` é carregado. Este objeto realiza a avaliação propriamente dita no seu método `void parseWai()`.

A cada conteúdo de página capturado, a classe `ProcessarErro` é acionada para realizar a avaliação. Ela solicita, à classe `FachadaArquivador`, o conteúdo da página indicado pelo código na instância `RelatorioDaUrl` e, nesta, carrega os dados resultantes da avaliação. Também entrega, para a classe `ResumoDoRelatorio`, os dados para montar o relatório resumido.

A classe `RelatorioDaUrl` organiza os dados do relatório de avaliação. Ela é quem solicita à classe `FachadaArquivador` o armazenamento ou recuperação destes dados em arquivo. Também é responsável por entregar os dados, de uma avaliação, solicitados pela classe `RelatorioDaUrlDAO`.

A classe `ResumoDoRelatorio` realiza a organização dos dados para resumo de relatório de avaliação e armazena-os em arquivo. Também recupera os dados de, um resumo, solicitados pela classe `ResumoDeAvaliacaoDAO`.

O retângulo rosa representa as classes que são responsáveis por gerenciar, entre outros, os arquivos que armazenam os dados de avaliação. A classe `FachadaArquivador` é responsável por toda a comunicação destas classes com o programa. Ela oferece a possibilidade de arquivar os dados da avaliação nos formatos JSON e XML.

A pasta temporária de usuário é utilizada para armazenar os arquivos referentes a avaliação solicitada por este. Por exemplo, os arquivos com o conteúdo das páginas, os arquivos com os relatórios de cada avaliação, os arquivos com os resumos de avaliação, o arquivo com os critérios da avaliação.

Um ponto preocupante é a quantidade de espaço de memória para suportar estas pastas. Uma hipótese é diminuir o tamanho dos arquivos. Para isto é necessário entender como os dados estão estruturados, como se relacionam com o sistema e a relevância para a funcionalidade do IsAcessível. Assim, na próxima seção, segue-se a coleta de dados sobre o estudo e testes com arquivos para contribuir com o gerenciamento sustentável do espaço de memória do IsAcessível.

### **2.2.2 Coleta de dados**

Os dados serão coletados através da observação sistemática, que, segundo Marconi e Lakatos (2003) colocam, entende-se que é a observação de fatos em um ambiente controlado. A fim de se obter respostas a objetivos anteriormente identificados. Onde os padrões de observação devem ser flexíveis em frente as circunstâncias, objetos de observação e objetivos. O observador sabe o que busca e consegue identificar o que é relevante em cada circunstância. Eliminando a sua própria influência sobre os dados que recolhe.

Desta forma, esta seção apresenta a coleta de dados sobre a observação do comportamento de um software, o ASES, em situações forçadas, interferindo em seus recursos, arquivos da pasta temporária, controladamente, com a finalidade de reconhecer a influência destes em sua execução. Para compreender a relevância dos componentes destes recursos para serem utilizados racionalmente na questão do gerenciamento de espaço de memória do projeto IsAcessível. Tendo o cuidado de tirar conclusões apenas sobre resultados de testes.

Com foco nisto, pretende-se entender os arquivos gerados pelo ASES, bem como a relevância dos dados contidos neles, a fim de diminuir o tamanho dos arquivos sem perder a força destes.

#### ***2.2.2.1 Entendo o arquivo XML com dados do relatório de avaliação do ASES***

Executando-se o ASES e, ao mesmo tempo, monitorando sua pasta temp, é possível notar que, para cada URL, de um domínio avaliado, são criados dois arquivos de mesmo nome, um com o conteúdo HTML da página e outro, de marcação XML.

O arquivo XML contém elementos que estruturam:

- A URL;
- A quantidade de erros por nível de prioridade;
- A quantidade de avisos por nível de prioridade;
- Os textos de todas as linhas do conteúdo HTML. Estes elementos possuem um atributo com o número da linha;
- As permissões para exibir cada nível de prioridade;
- Os conjuntos de pontos de verificação de erros por nível de prioridade;
- Os conjuntos de pontos de verificação de avisos por nível de prioridade;

Ponto de verificação é um elemento que possui atributos para o nível de prioridade e para a recomendação. Aninha outros elementos que estruturam:

- A numeração que representa o caso geral, esta é um par de elementos;
- Um número identificador da regra;
- A quantidade total de ocorrências para o caso geral;
- O conjunto de elementos com os números das linhas onde ocorreram o caso geral. Estes elementos têm um atributo que indica se é erro ou aviso, outro que indica a coluna na linha e outro que indica o tamanho do texto na linha.

Forçando a execução do programa com arquivos modificados manualmente, a fim de entender, por tentativa e erro, a relação dos dados deste arquivo com a funcionalidade, percebe-se que estes dados não sofrem modificações no programa, os elementos mantêm correspondência coordenada com as informações visualizadas e o arquivo guarda o estado inicial da avaliação. São requisitados sempre que a tela de relatório é visualizada. A Figura 17 mostra um desses arquivos XML, visualizado no navegador.

Figura 17 – Arquivo XML de avaliação visualizado no navegador

```

▼<relatorio>
  <url>http://www.fatecpe.com.br</url>
  <errosp1>18</errosp1>
  <errosp2>1</errosp2>
  <errosp3>0</errosp3>
  <avisosp1>124</avisosp1>
  <avisosp2>20</avisosp2>
  <avisosp3>66</avisosp3>
  ▶<conteudo>...</conteudo>
  <mostrap1/>
  ▶<listaep1>...</listaep1>
  ▶<listaap1>...</listaap1>
  <mostrap2/>
  ▶<listaep2>...</listaep2>
  ▶<listaap2>...</listaap2>
  <mostrap3/>
  ▶<listaap3>...</listaap3>
</relatorio>

```

Fonte – O autor

Percebe-se o elemento raiz <relatorio> composto pelas *tags* <url>, <errosp1>, <errosp2>, <errosp3>, <avisosp1>, <avisosp2>, <avisosp3>, <conteudo>, <mostrap1>, <listaep1>, <listaap1>, <mostrap2>, <listaep2>, <listaap2>, <mostrap3>, <listaep3>, <listaap3>.

A quantidade de linhas visualizadas na imagem anterior, Figura 17, é resultante da contração de alguns elementos, por exemplo o elemento conteúdo, nele estão todas as linhas do documento HTML que foi avaliado.

A seguir, são explicados os testes realizados com a finalidade de perceber a relevância destes elementos.

O ambiente de testes:

- O console Terminal do Linux – Ubuntu 15.04 aberto, para executar o ASES e visualizar possíveis exceções lançadas;
- O programa ASES 1.0.2 Linux em execução, realizando avaliação da URL <http://www.fatecpe.com.br>, sob recomendação EMAG, em nível PAGINA e para as três prioridades;
- O programa gedit 3.10.4<sup>31</sup>, para editar arquivos;

<sup>31</sup> <http://www.gedit.org/>

- Navegador, para visualizar as estruturas XML;
- Software Captura de tela 3.14.0<sup>32</sup>, para gerar imagens;
- Rotina de teste utilizada:
  - a) Executar o ASES pelo Terminal;
  - b) Aguardar a tela Avaliador de Acessibilidade abrir;
  - c) Preencher os dados;
  - d) Solicitar a avaliação;
  - e) Aguardar a tela Resumo do Relatório exibir o resumo do relatório da URL;
  - f) Visualizar a pasta temp do ASES;
  - g) Acessar o arquivo XML com o relatório da avaliação;
  - h) Realizar as modificações de teste no arquivo;
  - i) Salvar as modificações;
  - j) Voltar ao ASES e solicitar o relatório completo da URL;
  - k) Se a tela Relatório da Url não abrir, averiguar no Terminal alguma exceção lançada e, existindo ou não, registrar o fato e as conclusões;
  - l) Pelo contrário, na tela Relatório da Url, perceber o ambiente, procurando por comportamentos não condizentes nos componentes e informações, registrando os fatos e conclusões;
  - m) Se os testes não terminaram e os fatos permitirem, voltar a tela Resumo do Relatório e iniciar o próximo teste a partir do item f desta rotina;
  - n) Se os testes não terminaram e os fatos permitirem, voltar a tela Avaliador de Acessibilidade e iniciar o próximo teste a partir do item c desta rotina;
  - o) Se os testes não terminaram e os fatos permitirem sair do programa, e reiniciar esta rotina para iniciar o próximo teste;

---

<sup>32</sup> <https://git.gnome.org/browse/gnome-screenshot/tree/>

- p) Se os testes não terminaram e os fatos permitirem encerrar o programa por linha de comando no Terminal, e reiniciar esta rotina para iniciar o próximo teste;
- q) Se os testes terminaram, revisar os dados.
- Registro dos fatos e conclusões.

#### Teste 1 – Elemento <url>

A tag <url> armazena a URL e é utilizada para atualizar a guia Código Original. Quando o usuário clica no botão Atualizar, visível apenas com a guia citada selecionada, o sistema recupera esta URL e captura novamente o conteúdo da página.

Para chegar-se a esta conclusão, modificou-se o conteúdo de um dos elementos <clinha> da estrutura do elemento <conteudo> das linhas do arquivo XML de uma determinada URL. Retirou-se o elemento url deste arquivo, salvando-o. Solicitou-se o relatório da URL correspondente ao arquivo (para se saber qual URL corresponde um arquivo, basta olhar o arquivo CSV). Na tela Relatório da Url, selecionou-se a guia Código Original. Procurou-se a modificação feita no documento manualmente. E então clicou-se em Atualizar. Percebeu-se que o documento na guia não foi atualizado. Repetiu-se o processo sem retirar o elemento url e a atualização procedeu normalmente. Os outros componentes da tela não foram afetados.

#### Testes 2 – Elementos <erros1>, <erros2>, <erros3>, <avisosp1>, <avisosp2>, <avisosp3>

Armazenam as quantidades de erros e avisos dos níveis de prioridade. Estes elementos são essenciais para exibição, na guia Código Original, das guias erros e avisos. A retirada destes elementos, manualmente, do arquivo, resultou na não exibição destas guias. Para repetir o feito, basta manter o ASES na tela Resumo do relatório, retirar qualquer um destes elementos ou todos, salvar o arquivo, solicitar o relatório completo da URL correspondente ao arquivo. Selecionar a guia Código Original. Visualizar a área das guias de erro e avisos e notar a não exibição da(s) guia(s) correspondente(s). Não percebeu-se outros componentes afetados.

#### Teste 3 – Elemento <conteudo>

É um conjunto de tags <clinha>. A Figura 18 mostra a tag <conteudo> expandida, exibindo tags <clinha>.

Figura 18 – Arquivo XML de avaliação com tag &lt;conteudo&gt; expandida

```

<avisosp3>66</avisosp3>
▼<conteudo>
  <clinha nro="0001"><!DOCTYPE html></clinha>
  <clinha nro="0002"><!--[if sssssssssssssssssIE 6]></clinha>
  <clinha nro="0003"><html id="ie6" lang="pt-BR"></clinha>
  <clinha nro="0004"><![endif]--></clinha>
  <clinha nro="0005"><!--[if IE 7]></clinha>
  <clinha nro="0006"><html id="ie7" lang="pt-BR"></clinha>
  <clinha nro="0007"><![endif]--></clinha>
  <clinha nro="0008"><!--[if IE 8]></clinha>
  <clinha nro="0009"><html id="ie8" lang="pt-BR"></clinha>
  <clinha nro="0010"><![endif]--></clinha>
  <clinha nro="0011"><!--[if !(IE 6) | !(IE 7) | !(IE 8) ]><!--></clinha>
  <clinha nro="0012"><html lang="pt-BR"></clinha>
  <clinha nro="0013"><!--<![endif]--></clinha>
  <clinha nro="0014"><head></clinha>
  <clinha nro="0015"><meta charset="UTF-8" /></clinha>
  <clinha nro="0016"></clinha>
  ▶<clinha nro="0017">...</clinha>
  ▶<clinha nro="0018">...</clinha>
  <clinha nro="0019"><!--[if lt IE 9]></clinha>
  ▶<clinha nro="0020">...</clinha>
  <clinha nro="0021"><![endif]--></clinha>
  ▶<clinha nro="0022">...</clinha>

```

Fonte – O autor

Cada linha do documento HTML é armazenada numa tag <clinha> que tem no atributo nro o número da linha. O elemento conteúdo comporta o documento HTML completo. Forçar a execução do programa, com este elemento excluído, manualmente, não apresentou alterações no seu fluxo nem na exibição de algum de seus componentes. Por conta disto, o elemento <clinha> não vai ser testado individualmente.

Teste 4 – Elementos <mostrap1>, <mostrap2>, <mostrap3>

Quando presentes, respectivamente, indicam ao programa que as recomendações das prioridades de 1, 2 e 3, na guia Código Original, devem ser exibidas. Nos testes de retirada destes elementos, averiguou-se a não exibição das guias erros e avisos correspondentes. Os passos deste teste são como os do Teste 2, apenas, substitui-se aqueles por estes.

Teste 5 – Elementos <listaep1>, <listaap1>, <listaep2>, <listaap2>, <listaep3>, <listaap3>

Estes elementos são estruturas que aninham elementos <pontoverificacao>. A Figura 19 exibe alguns elementos desses expandidos.





Figura 20 – Elemento &lt;pontoverificacao&gt; expandido

```

▼<pontoverificacao priori="1" wcagemag="2">
  <gl>1</gl>
  <cp>20</cp>
  <id_regra>0</id_regra>
  <total>17</total>
  ▼<linhas>
    <linha avisoerro="e" col="28" taglen="28">0028</linha>
    <linha avisoerro="e" col="3" taglen="31">0034</linha>
    <linha avisoerro="e" col="1" taglen="106">0058</linha>
    <linha avisoerro="e" col="1" taglen="117">0059</linha>
    <linha avisoerro="e" col="1" taglen="31">0063</linha>
    <linha avisoerro="e" col="1" taglen="31">0383</linha>
    <linha avisoerro="e" col="26" taglen="150">0395</linha>
    <linha avisoerro="e" col="1" taglen="31">0396</linha>
    <linha avisoerro="e" col="1" taglen="130">0401</linha>
    <linha avisoerro="e" col="1" taglen="143">0402</linha>
    <linha avisoerro="e" col="1" taglen="143">0403</linha>
    <linha avisoerro="e" col="1" taglen="137">0404</linha>
    <linha avisoerro="e" col="1" taglen="31">0405</linha>
    <linha avisoerro="e" col="1" taglen="138">0410</linha>
    <linha avisoerro="e" col="1" taglen="144">0411</linha>
    <linha avisoerro="e" col="1" taglen="31">0412</linha>
    <linha avisoerro="e" col="1" taglen="31">0417</linha>
  </linhas>
</pontoverificacao>

```

Fonte – O autor

### Teste 7 – Elementos <gl> e <cp>

Estruturam, conjuntamente, o número que representa um caso geral. A tag <gl> armazena o nível de prioridade da regra e, a tag <cp>, armazena o número da regra dentro do seu nível de prioridade. Estes elementos, quando submetidos a testes, individualmente ou em conjunto, como no Teste 4, ocasionaram o lançamento de uma exceção e a interrupção do programa. A Figura 21 mostra a exceção lançada.

Figura 21 – Exceção lançada nos testes com os elementos &lt;gl&gt; e &lt;cp&gt;

```
Exception in thread "AWT-EventQueue-0" java.lang.NullPointerException
at br.org.aceessobrasil.ases.regras.RegrasHardCodedEmag.getTextoRegra(Unknown Source)
at br.org.aceessobrasil.silvinha.vista.tableComponents.RelatorioTableModel.<init>(Unknown Source)
at br.org.aceessobrasil.silvinha.vista.panels.PainelRelatorioPorPrioridade.<init>(Unknown Source)
at br.org.aceessobrasil.silvinha.vista.panels.relatorio.PainelRelatorio.loadRelatorio(Unknown Source)
at br.org.aceessobrasil.silvinha.vista.panels.relatorio.PainelRelatorio.constroiPainelOriginal(Unknown Source)
at br.org.aceessobrasil.silvinha.vista.panels.relatorio.PainelRelatorio.<init>(Unknown Source)
at br.org.aceessobrasil.silvinha.vista.frames.classePrincipal.FrameSilvinha.showPainelRelatorio(Unknown Source)
at br.org.aceessobrasil.silvinha.vista.panels.PainelResumo.actionPerformed(Unknown Source)
at br.org.aceessobrasil.silvinha.vista.listeners.PainelResumoMouseListener.mouseClicked(Unknown Source)
at java.awt.AWTEventMulticaster.mouseClicked(AWTEventMulticaster.java:270)
at java.awt.Component.processMouseEvent(Component.java:6538)
at javax.swing.JComponent.processMouseEvent(JComponent.java:3324)
at java.awt.Component.processEvent(Component.java:6300)
at java.awt.Container.processEvent(Container.java:2236)
at java.awt.Component.dispatchEventImpl(Component.java:4891)
at java.awt.Container.dispatchEventImpl(Container.java:2294)
at java.awt.Component.dispatchEvent(Component.java:4713)
at java.awt.LightweightDispatcher.retargetMouseEvent(Container.java:4888)
at java.awt.LightweightDispatcher.processMouseEvent(Container.java:4534)
at java.awt.LightweightDispatcher.dispatchEvent(Container.java:4466)
at java.awt.Container.dispatchEventImpl(Container.java:2280)
at java.awt.Window.dispatchEventImpl(Window.java:2750)
at java.awt.Component.dispatchEvent(Component.java:4713)
at java.awt.EventQueue.dispatchEventImpl(EventQueue.java:758)
at java.awt.EventQueue.access$500(EventQueue.java:97)
at java.awt.EventQueue$3.run(EventQueue.java:789)
at java.awt.EventQueue$3.run(EventQueue.java:783)
at java.security.AccessController.doPrivileged(Native Method)
at java.security.ProtectionDomain$JavaSecurityAccessImpl.doIntersectionPrivilege(ProtectionDomain.java:76)
at java.security.ProtectionDomain$JavaSecurityAccessImpl.doIntersectionPrivilege(ProtectionDomain.java:86)
at java.awt.EventQueue$4.run(EventQueue.java:731)
at java.awt.EventQueue$4.run(EventQueue.java:729)
at java.security.AccessController.doPrivileged(Native Method)
at java.security.ProtectionDomain$JavaSecurityAccessImpl.doIntersectionPrivilege(ProtectionDomain.java:76)
at java.awt.EventQueue.dispatchEvent(EventQueue.java:728)
at java.awt.EventDispatchThread.pumpOneEventForFilters(EventDispatchThread.java:201)
at java.awt.EventDispatchThread.pumpEventsForFilter(EventDispatchThread.java:116)
at java.awt.EventDispatchThread.pumpEventsForHierarchy(EventDispatchThread.java:105)
at java.awt.EventDispatchThread.pumpEvents(EventDispatchThread.java:101)
at java.awt.EventDispatchThread.pumpEvents(EventDispatchThread.java:93)
at java.awt.EventDispatchThread.run(EventDispatchThread.java:82)
```

Fonte – O autor

#### Teste 8 – Elemento <id\_regra>

Este elemento existe no arquivo, porém sua exclusão em testes, como o Teste 4, não ocasionou influência e mais, o único valor, atribuído a ele, é zero em todas as situações.

#### Teste 9 – Elemento <total>

Comparando-se os valores atribuídos a este elemento com os do campo OCORRÊNCIAS das listas nas guias erros ou avisos, nota-se que corresponde a quantidade de ocorrências de caso geral. Porém, da mesma forma que o elemento <total>, sua exclusão do arquivo em nada influenciou.

#### Teste 10 – Elemento <linhas>

Este elemento é um conjunto de elementos <linha>. Em Teste, como o Teste 4, a exclusão deste elemento, ocasionou a não exibição do valor do campo LINHAS dos itens em que agregam e, ainda mais, o valor do campo OCORRÊNCIAS também não foi exibido, mesmo com o elemento <total> presente.

#### Teste 11 – Elemento <linha>

Este elemento armazena o número da linha em registra-se um erro ou aviso. Possui os atributos: avisoerro, indica se é um aviso (valor a) ou erro (valor e), col, indica a coluna da

linha em que inicia a ocorrência e, taglen, que indica o tamanho do texto na linha. Testes, conforme o Teste 4, demonstraram que a exclusão forçada deste elemento ocasionou a não exibição do número da linha, correspondente ao elemento excluído e, ainda mais, o valor do campo OCORRÊNCIAS foi decrementado a proporção destes elementos excluídos no mesmo item.

#### *2.2.2.2 Considerações sobre os arquivos de relatório de avaliação para as necessidades do IsAcessível*

Como pode ser visto na seção 1.2.11.2, os dois arquivos, gerados para cada URL da avaliação, são preponderantes na quantidade de espaço de memória ocupado pela pasta. Observando-se o tamanho dos dois, 30,4 kB do arquivo 58747265 e 62,6 kB do arquivo 58747265.xml, com o fato de que uma avaliação solicitada para site pode, por exemplo, resultar da visita de 201 URLs, como visto na seção 1.2.11.1, abstraindo-se o fato de que documentos de URLs diferentes podem ter tamanhos diferentes, multiplicando 201 URLs pela soma do tamanho destes dois arquivos produz-se 18693 kB.

Diante da necessidade do IsAcessível gerenciar a quantidade de memória utilizada para suportar as pastas temporárias de n usuários com seus arquivos de avaliações solicitadas com n URLs, isto é, gerenciar um espaço de memória que pode crescer exponencialmente é imprescindível procurar meios de diminuir o tamanho do espaço de memória utilizado por pasta temporária de usuário.

Com base nos testes realizados na seção 2.2.2.1 é possível elencar os seguintes pontos que podem colaborar:

1. O IsAcessível utiliza o arquivo com o conteúdo HTML apenas para realizar a avaliação propriamente dita, ver seção 2.2.1.4. Desta forma não é necessário mantê-lo após sua utilidade;
2. O elemento <conteudo> do arquivo xml, após a visualização da tela Relatório da Url do ASES, não tem utilidade. Pode ser excluído do documento.
3. As nomenclaturas dos elementos do arquivo xml são utilizadas somente pelo sistema, o usuário não é influenciado por elas. Podem ser diminuídas;
4. Atrelado aos pontos anteriores, utilizar outro formato de arquivo.

Os quatro pontos apresentados serão implementados e testados na próxima seção.

### 2.2.2.3 Testes com configurações de arquivção para diminuir a quantidade de espaço de memória das pastas temporárias de usuário

A geração dos arquivos estudados na seção anterior, bem como os quatro pontos, elencados, são base para 8 configurações de arquivção. Em cada uma, é observado, na pasta temporária de usuário e no campo TAMANHO, o espaço de memória utilizado por dois dos arquivos resultantes da avaliação de acessibilidade da página de uma URL. Estes arquivos são o conteúdo HTML capturado, daqui em diante chamado HTML capturado, e o arquivo com o relatório de avaliação, daqui em diante chamado relatório da avaliação.

As configurações são comparáveis entre si, pelo formato (XML ou JSON) do relatório de avaliação, por manter o HTML capturado, pela manter as linhas do documento HTML no relatório de avaliação e, quando o formato do relatório de avaliação é XML, pela quantidade de caracteres na nomenclatura dos elementos e atributos.

Este último não é aplicável ao formato JSON pelo fato da estrutura ser organizada em lista ordenada de valores, ver seção 1.2.10. Por outro lado, XML preocupa-se em descrever dados, por isto, usa tags e atributos com nomenclatura, ver seção 1.2.9.

Será utilizado, nas configurações, duas formas de nomenclatura dos elementos e atributos, do relatório de avaliação, apresentados na seção 2.2.2.1: (i) Normal – sem modificação da quantidade de caracteres, e; (ii) Mínima – onde as nomenclaturas têm apenas dois caracteres.

A Tabela 1 mostra a correspondência, dos elementos e atributos, da forma de nomenclatura normal com a mínima.

Tabela 1 – Correspondência das formas de nomenclatura normal e mínima dos elementos e atributos da estrutura XML do relatório de avaliação

Normal	Mínima
<relatorio>	<re>
<url>	<ur>

<b>Normal</b>	<b>Mínima</b>
<errosp1>	<e1>
<errosp2>	<e2>
<errosp3>	<e3>
<avisosp1>	<a1>
<avisosp2>	<a2>
<avisosp3>	<a3>
<conteudo>	<co>
<clinha>	<cl>
nro	nr
<mostrap1>	<m1>
<mostrap2>	<m2>
<mostrap3>	<m3>
<listaep1>	<l1>
<listaep2>	<l2>
<listaep3>	<l3>
<listaap1>	<l4>
<listaap2>	<l5>
<listaap3>	<l6>
<pontoverificacao>	<po>
priori	pr
wcagemag	wc

<b>Normal</b>	<b>Mínima</b>
<gl>	<gl>
<cp>	<cp>
<id_regra>	<id>
<total>	<to>
<linhas>	l7
<linha>	l8
col	co
taglen	ta
avisoerro	av

Fonte – O autor

As configurações de arquivagem são como segue:

Configuração 0: Formato XML, HTML capturado mantido, linhas do documento HTML presentes, nomenclatura normal. Sendo esta a configuração que gera os arquivos como os da seção 2.2.2.2;

Configuração 1: Formato XML, HTML capturado não mantido, linhas do documento HTML presentes, nomenclatura normal;

Configuração 2: Formato XML, HTML capturado não mantido, linhas do documento HTML presentes, nomenclatura mínima;

Configuração 3: Formato XML, HTML capturado não mantido, linhas do documento HTML não presentes, nomenclatura normal;

Configuração 4: Formato XML, HTML capturado não mantido, linhas do documento HTML não presentes, nomenclatura mínima;

Configuração 5: Formato JSON, HTML capturado mantido, linhas do documento HTML presentes;

Configuração 6: Formato JSON, HTML capturado não mantido, linhas do documento HTML presentes;

Configuração 7: Formato JSON, HTML capturado não mantido, linhas do documento HTML não presentes;

A Tabela 2 organiza os dados das configurações de arquivção e seus possíveis valores. Com as seguintes representações:

- Configuração: representa a configuração de arquivção, os valores são os números de ordem de cada configuração.
- Formato: representa o formato do relatório de avaliação e pode conter o valor XML ou JSON.
- HTML: representa HTML capturado mantido, os valores possíveis são SIM ou NÃO.
- Linhas: representa linhas do documento HTML presentes, os valores são SIM ou NÃO.
- Nomenclatura: representa o tipo de nomenclatura, os valores são Normal, Mínima ou Sem Suporte (para o formato JSON).

Tabela 2 – Configurações de arquivção

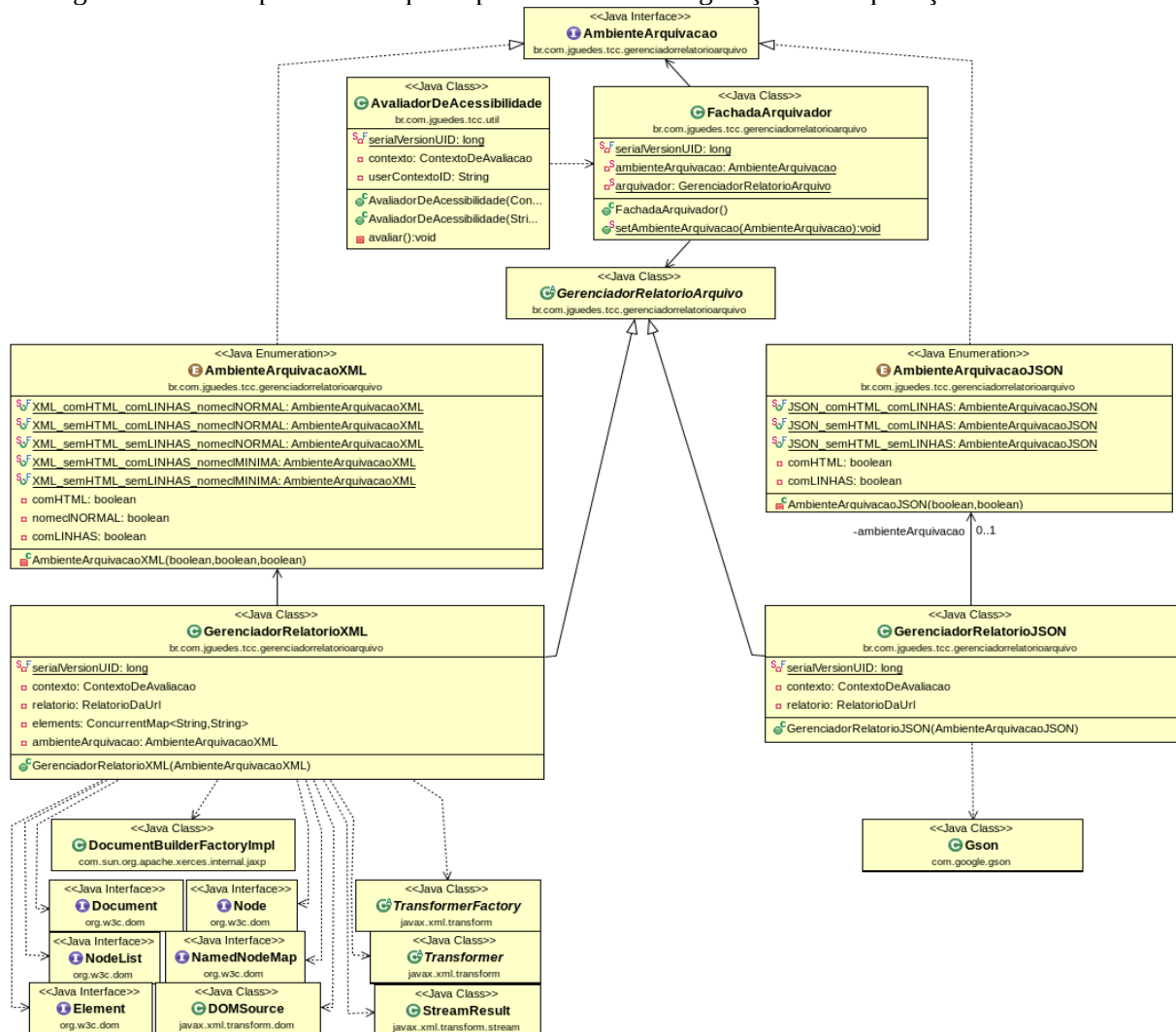
<b>Configuração</b>	<b>Formato</b>	<b>HTML</b>	<b>Linhas</b>	<b>Nomenclatura</b>
0	XML	SIM	SIM	Normal
1	XML	NÃO	SIM	Normal
2	XML	NÃO	SIM	Mínima
3	XML	NÃO	NÃO	Normal
4	XML	NÃO	NÃO	Mínima
5	JSON	SIM	SIM	Sem Suporte
6	JSON	NÃO	SIM	Sem Suporte
7	JSON	NÃO	NÃO	Sem Suporte

Fonte – O autor

As configurações de arquivção foram implementadas e realizadas com o IsAcessível. Foram criadas rotinas especializadas para cada uma. A Figura 22 mostra um diagrama com as principais classes que implementam-nas.



Figura 22 – Principais classes que implementam as configurações de arquivção do IsAcessível



Fonte – O autor

A interface `AmbienteArquivacao` é implementada pelos enumeradores `AmbienteArquivacaoXML` e `AmbienteArquivacaoJSON`. Eles encapsulam constantes que possuem os critérios das configurações de arquivção. Tais critérios são fornecidos pelos atributos:

- `comHTML` – o valor *true*, indica que o HTML capturado deve ser mantido, *false*, destruído;
- `nomeclNORMAL` – o valor *true*, indica que a nomenclatura da tags e atributos deve seguir o padrão normal, *false*, mínima;
- `comLINHAS` – o valor *true*, indica que o elemento que contém as linhas do documento HTML deve existir na estrutura, *false*, não.

As constantes enumeradas em AmbienteArquivacaoXML são:

- XML\_comHTML\_comLINHAS\_nomeclNORMAL, corresponde a Configuração 0, com os valores em atributos:
  - comHTML = true;
  - comLINHAS = true;
  - nomeclNORMAL = true.
- XML\_semHTML\_comLINHAS\_nomeclNORMAL, corresponde a Configuração 1, com os valores em atributos:
  - comHTML = false;
  - comLINHAS = true;
  - nomeclNORMAL = true.
- XML\_semHTML\_comLINHAS\_nomeclMINIMA, corresponde a Configuração 2, com os valores em atributos:
  - comHTML = false;
  - comLINHAS = true;
  - nomeclNORMAL = false.
- XML\_semHTML\_semLINHAS\_nomeclNORMAL, corresponde a Configuração 3, com os valores em atributos:
  - comHTML = false;
  - comLINHAS = false;
  - nomeclNORMAL = true.
- XML\_semHTML\_semLINHAS\_nomeclMINIMA, corresponde a Configuração 4, com os valores em atributos:
  - comHTML = false;
  - comLINHAS = false;

- nomeclNORMAL = false.

As constantes enumeradas em AmbienteArquivacaoJSON, são:

- JSON\_comHTML\_comLINHAS, corresponde a Configuração 5, com os valores em atributos:
  - comHTML = true;
  - comLINHAS = true.
- XML\_semHTML\_comLINHAS, corresponde a Configuração 6, com os valores em atributos:
  - comHTML = false;
  - comLINHAS = true.
- XML\_semHTML\_semLINHAS, corresponde a Configuração 7, com os valores em atributos:
  - comHTML = false;
  - comLINHAS = false.

A Tabela 3 resume configurações de arquivação e as respectivas constantes.

Tabela 3 – Correspondência das configurações de arquivação com as respectivas constantes

<b>Conf.</b>	<b>Constante</b>
0	AmbienteArquivacaoXML.XML_comHTML_comLINHAS_nomeclNORMAL
1	AmbienteArquivacaoXML.XML_semHTML_comLINHAS_nomeclNORMAL
2	AmbienteArquivacaoXML.XML_semHTML_comLINHAS_nomeclMINIMA
3	AmbienteArquivacaoXML.XML_semHTML_semLINHAS_nomeclNORMAL
4	AmbienteArquivacaoXML.XML_semHTML_semLINHAS_nomeclMINIMA
5	AmbienteArquivacaoJSON.JSON_comHTML_comLINHAS
6	AmbienteArquivacaoJSON.JSON_semHTML_comLINHAS
7	AmbienteArquivacaoJSON.JSON_semHTML_semLINHAS

A classe `FachadaArquivador` gerencia arquivos. Para manipular relatório de avaliação, ela possui uma referência do tipo abstrato `GerenciadorRelatorioArquivo` que é estendido pelas classes `GerenciadorRelatorioXML` e `GerenciadorRelatorioJSON`, as quais, são instanciadas de acordo com a variável de classe do tipo `AmbienteArquivacao`. A Figura 23 mostra a implementação desta rotina no método `setAmbienteArquivacao`.

Figura 23 – Trecho de código `FachadaArquivador.setAmbienteArquivacao`

```
public class FachadaArquivador implements Serializable {
    private static final long serialVersionUID = 7967232016513424533L;
    private static AmbienteArquivacao ambienteArquivacao;
    private static GerenciadorRelatorioArquivo arquivador;
    public static void setAmbienteArquivacao(AmbienteArquivacao ambienteArquivacao) {
        FachadaArquivador.ambienteArquivacao = ambienteArquivacao;
        if (ambienteArquivacao instanceof AmbienteArquivacaoXML) {
            arquivador = new GerenciadorRelatorioXML((AmbienteArquivacaoXML) ambienteArquivacao);
        } else if (ambienteArquivacao instanceof AmbienteArquivacaoJSON) {
            arquivador = new GerenciadorRelatorioJSON((AmbienteArquivacaoJSON) ambienteArquivacao);
        }
    }
}
```

Fonte – O autor

`GerenciadorRelatorioXML` é a classe que manipula o HTML capturado e o relatório de avaliação estruturado em XML. Ela gera os arquivos obedecendo aos critérios encapsulados no parâmetro do tipo `AmbienteArquivacaoXML`.

Na imagem da Figura 22, todas as classes, que estão em baixo da classe `GerenciadorRelatorioXML`, fazem parte das APIs utilizadas para manipulação de XML.

A classe `GerenciadorRelatorioJSON` manipula HTML capturado e o relatório de avaliação estruturado em JSON. Nela, `AmbienteArquivacaoJSON` é o tipo do parâmetro que encapsula os critérios que são obedecidos para gerar os arquivos.

`Gson`<sup>33</sup> é a API utilizada para manipular JSON.

A classe `AvaliadorDeAcessibilidade` é responsável por passar para `FachadaArquivador` as configurações de arquivção. A Figura 24 exhibe o código do método `avaliar` onde `FachadaArquivador` esta recebendo a constante da Configuração 0 (linha 40).

<sup>33</sup> <https://github.com/google/gson>

Figura 24 – Trecho de código da classe AvaliadorDeAcessibilidade.avaliar()

```

34 private void avaliar() {
35
36     contexto.setTotLinks(0);
37
38     FachadaArquivador.inicializarDiretorioAvaliacao(contexto.getFolderTemp());
39
40     FachadaArquivador.setAmbienteArquivacao(AmbienteArquivacaoXML.XML_comHTML_comLINHAS_nomeClnORMAL);
41
42     String url = contexto.getCritério().getUrl();
43
44     String path = contexto.getUserFolder();
45
46     try {
47         new Crawler4j(url, path);
48     } catch (Exception e) {
49     }
50 }

```

Fonte – O autor

O ambiente de observação é como segue:

1. Via programação, passar os parâmetros de arquivção para a classe FachadaArquivador;
2. Executar ou reiniciar o servidor;
3. Realizar avaliação no IsAcessível com os seguintes dados de entrada:
  - URL da instituição acadêmica deste trabalho, a saber, <http://www.fatecpe.com.br>;
  - Padrão de recomendação EMAG;
  - Três níveis de prioridade selecionados;
  - Profundidade PAGINA.
4. Acessar a pasta temporária do usuário do teste;
5. Perceber os resultados e registrá-los no quadro de comparações.

Todas as solicitações de avaliação, nos testes destas configurações, exibem as mesmas telas para o usuário. A tela de entrada com os critérios de testes é mostrada na Figura 25. A tela de relatório resumido é mostrada Figura 26. A tela de detalhamento da avaliação pode ser vista na Figura 27.

Figura 25 – Tela de entrada com os critérios de teste

IsAcessível - Avaliador de Acessibilidade para web

Recomendações: EMAG Profundidade: PAGINA

☒ Prioridade 1 ☒ Prioridade 2 ☒ Prioridade 3

URL: http://www.fatecpe.com.br/

avaliar

Fonte – O autor

Figura 26 – Tela Relatório resumido

IsAcessível - Relatório resumido

Recomendação: EMAG, Profundidade: PAGINA

1: http://www.fatecpe.com.br/

Erros P1	Avisos P1	Erros P2	Avisos P2	Erros P3	Avisos P3	exibir
18	124	1	20	0	66	

Fonte – O autor

Figura 27 – Tela de Relatório de avaliação

IsAcessível - Relatório de avaliação de acessibilidade no padrão EMAG

URL: http://www.fatecpe.com.br/

229 ocorrências (19 erros | 210 avisos)

Prioridade 1: 142 ocorrências Prioridade 2: 21 ocorrências Prioridade 3: 66 ocorrências

Erros P1: 18 ocorrências Avisos P1: 124 ocorrências

Casos Gerais	Ocorrências	Linhas
EMAG: 1.20: Assegurar que todas as páginas possam ser utilizadas mesmo que os programas interpretáveis, os applets ou outros objetos programados tenham sido desativados ou não sejam suportados. Sempre que tiver script associe logo a seguir o elemento "noscript".	Ocorrências: 17	Linhas: [28, 34, 58, 59, 63, 383, 395, 396, 401, 402, 403, 404, 405, 410, 411, 412, 417]
EMAG: 1.11: Fornecer um equivalente textual a cada imagem (isso abrange: representações gráficas do texto, incluindo símbolos, GIFs animados, imagens utilizadas como sinalizadores de pontos de enumeração, legendas e botões gráficos).	Ocorrências: 1	Linhas: [106]

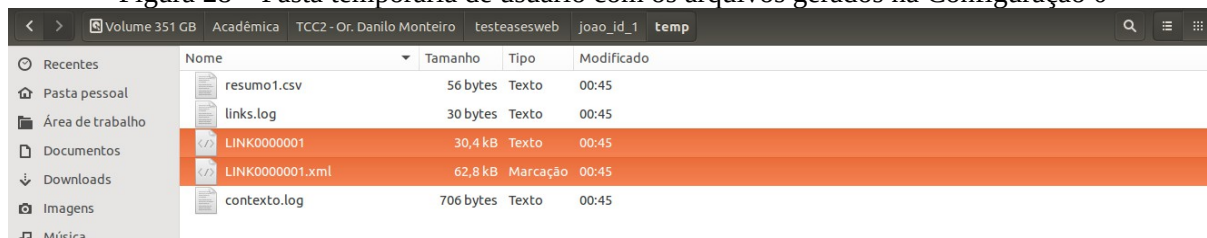
Fonte – O autor

## Teste da Configuração 0

Esta configuração é a que gera os arquivos como os da seção 2.2.2.2. A classe `FachadaArquivador` deve ser configurada com a constante, do enumerador `AmbienteArquivacaoXML`, `XML_comHTML_comLINHAS_nomeclNORMAL`.

A pasta temporária de usuário com os arquivos gerados, após a avaliação, pode ser vista na Figura 28.

Figura 28 – Pasta temporária de usuário com os arquivos gerados na Configuração 0



Fonte – O autor

Nota-se os arquivos `LINK0000001` e `LINK0000001.xml` selecionados, com 30,4 kB e 62,8 kB de tamanho, respectivamente. Onde, o primeiro é o HTML capturado e o segundo, o relatório da avaliação.

Espaço total de memória desta configuração: 30,4 kB + 62,8 kB = 93,2 kB.

## Teste da Configuração 1

A constante que configura a classe `FachadaArquivador` é `XML_semHTML_comLINHAS_nomeclNORMAL` do enumerador `AmbienteArquivacaoXML`.

Após a avaliação, dos dois arquivos estudados, somente houve o relatório de avaliação com 62,8 kB.

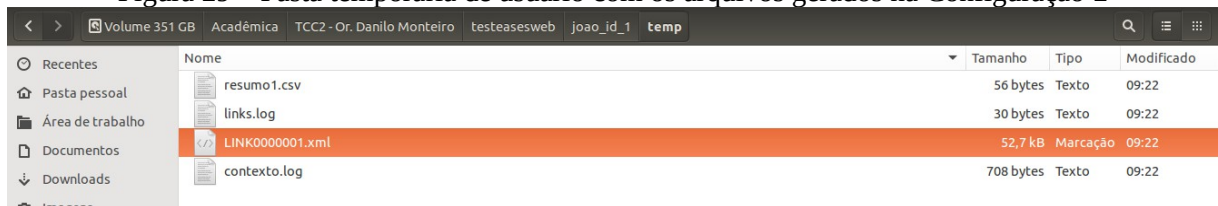
Espaço total de memória desta configuração: 62,8 kB.

## Teste da Configuração 2

A constante `XML_semHTML_comLINHAS_nomeclMINIMA`, do enumerador `AmbienteArquivacaoXML`, configura a classe `FachadaArquivador`.

Após a avaliação, a pasta temporária de usuário está como na Figura 29 e, o conteúdo do relatório de avaliação é exibido na Figura 30, com o elemento <co>, que é o elemento <conteudo> com seu nome na forma MINIMA, contraído.

Figura 29 – Pasta temporária de usuário com os arquivos gerados na Configuração 2



Nome	Tamanho	Tipo	Modificado
resumo1.csv	56 bytes	Texto	09:22
links.log	30 bytes	Texto	09:22
LINK0000001.xml	52,7 kB	Marcação	09:22
contexto.log	708 bytes	Texto	09:22

Fonte – O autor

Figura 30 – Relatório de avaliação gerado pela Configuração 2

```

<re>
  <ur>http://www.fatecpe.com.br/</ur>
  <e1>18</e1>
  <e2>1</e2>
  <e3>0</e3>
  <a1>124</a1>
  <a2>20</a2>
  <a3>66</a3>
  <co>...</co>
  <m1/>
  <l1>...</l1>
  <l4>
    <po pr="1" wc="2">...</po>
    <po pr="1" wc="2">...</po>
    <po pr="1" wc="2">...</po>
    <po pr="1" wc="2">...</po>
    <po pr="1" wc="2">...</po>
    <po pr="1" wc="2">...</po>
    <po pr="1" wc="2">...</po>
    <po pr="1" wc="2">
      <gl>1</gl>
      <cp>7</cp>
      <id>0</id>
      <to>6</to>
    </po>
    <l7>
      <l8 av="a" cl="314" ta="314">0314</l8>
      <l8 av="a" cl="3" ta="134">0320</l8>
      <l8 av="a" cl="3" ta="132">0326</l8>
      <l8 av="a" cl="3" ta="132">0335</l8>
      <l8 av="a" cl="3" ta="134">0341</l8>
      <l8 av="a" cl="3" ta="134">0347</l8>
    </l7>
  </po>
  <po pr="1" wc="2">...</po>
  <po pr="1" wc="2">...</po>
  <po pr="1" wc="2">...</po>
</l4>
<m2/>
<l2>...</l2>
<l5>...</l5>
<m3/>
<l6>...</l6>
</re>

```

Fonte – O autor



Nota-se a não presença do HTML capturado, que seria o arquivo com o mesmo nome do relatório de avaliação, LINK0000001.

Como pode ser visto, o tamanho do relatório de avaliação, LINK0000001.xml, é 52,7 kB;

Espaço total de memória desta configuração: 52,7 kB.

### Teste da Configuração 3

Utiliza-se a constante do enumerador AmbienteArquivaçãoXML, XML\_semHTML\_semLINHAS\_nomeclNORMAL, para configurar a classe FachadaArquivador.

A Figura 31 mostra a pasta com o arquivo XML de resultado de avaliação do IsAcessível com a Configuração 3.

Figura 31 – Pasta temporária de usuário com os arquivos gerados na Configuração 3



Nome	Tamanho	Tipo	Modificado
resumo1.csv	56 bytes	Texto	01:00
links.log	30 bytes	Texto	01:00
LINK0000001.xml	17,0 kB	Marcação	01:00
contexto.log	708 bytes	Texto	01:01

Fonte – O autor

Nota-se o arquivo selecionado com apenas 17,0 kB de tamanho.

Espaço total de memória desta configuração: 17,0 kB.

### Teste da Configuração 4

Com a classe FachadaArquivador configurada com a constante XML\_semHTML\_semLINHAS\_nomeclMINIMA, do enumerador AmbienteArquivacaoXML, a avaliação resultou no relatório de avaliação exibido na Figura 32. E a Figura 33 mostra a pasta temporária de usuário.

Deve-se atentar para a ausência do elemento que deveria conter as linhas do documento HTML, neste caso, seria o elemento <co>.

O relatório de avaliação, arquivo LINK0000001.xml, tem 10,7 kB de tamanho.

Espaço total de memória desta configuração: 10,7 kB.

Figura 32 – Relatório de avaliação resultante da Configuração 4

```

▼<re>
  <ur>http://www.fatecpe.com.br/</ur>
  <e1>18</e1>
  <e2>1</e2>
  <e3>0</e3>
  <a1>124</a1>
  <a2>20</a2>
  <a3>66</a3>
  <m1/>
  ▶<l1>...</l1>
  ▼<l4>
    ▶<po pr="1" wc="2">...</po>
    ▶<po pr="1" wc="2">...</po>
    ▶<po pr="1" wc="2">...</po>
    ▶<po pr="1" wc="2">...</po>
    ▶<po pr="1" wc="2">...</po>
    ▶<po pr="1" wc="2">...</po>
    ▶<po pr="1" wc="2">...</po>
    ▶<po pr="1" wc="2">...</po>
    ▼<po pr="1" wc="2">
      <gl>1</gl>
      <cp>7</cp>
      <id>0</id>
      <to>6</to>
      ▼<l7>
        <l8 av="a" cl="314" ta="314">0314</l8>
        <l8 av="a" cl="3" ta="134">0320</l8>
        <l8 av="a" cl="3" ta="132">0326</l8>
        <l8 av="a" cl="3" ta="132">0335</l8>
        <l8 av="a" cl="3" ta="134">0341</l8>
        <l8 av="a" cl="3" ta="134">0347</l8>
      </l7>
    </po>
    ▶<po pr="1" wc="2">...</po>
    ▶<po pr="1" wc="2">...</po>
    ▶<po pr="1" wc="2">...</po>
  </l4>
  <m2/>
  ▶<l2>...</l2>
  ▶<l5>...</l5>
  <m3/>
  ▶<l6>...</l6>
</re>

```

Fonte – O autor

Figura 33 – Pasta temporária de usuário com os arquivos gerados na Configuração 4

	Nome	Tamanho	Tipo	Modificado
Recentes	resumo1.csv	56 bytes	Texto	01:07
Pasta pessoal	links.log	30 bytes	Texto	01:07
Área de trabalho	LINK0000001.xml	10,7 kB	Marcação	01:07
Documentos	contexto.log	709 bytes	Texto	01:07
Downloads				
Imagens				

Fonte – O autor

## Teste da Configuração 5

Agora, o enumerador AmbienteArquivacaoJSON, contém a constante, JSON\_comHTML\_comLINHAS, para configurar a classe FachadaArquivador.



Figura 36 – Pasta temporária de usuário com os arquivos gerados na Configuração 5, com relatório de avaliação contendo caracteres de escape

Nome	Tamanho	Tipo	Modificado
resumo1.csv	56 bytes	Texto	10:28
links.log	30 bytes	Texto	10:28
LINK0000001	30,4 kB	Texto	10:28
LINK0000001.json	53,1 kB	Programa	10:28
contexto.log	669 bytes	Texto	10:29

Fonte – o autor

A outra forma, sem substituir os caracteres especiais, é instanciando o objeto do tipo Gson a partir do método `disableHtmlEscaping().create()` da classe `GsonBuilder`.

A Figura 37 mostra o mesmo trecho de código da implementação anterior, agora com as modificações percebidas nas linhas 126 e 128.

A Figura 38 exibe o trecho do relatório de avaliação gerado. Nota-se os caracteres especiais, tais como, os delimitadores de tags.

A Figura 39 apresenta a pasta temporária com o HTML capturado, com 30,4 kB, e o relatório de avaliação com 43,8 kB, menor que os 53,1 kB da implementação sem `GsonBuilder`.

Adotar-se-ão os valores produzidos com o uso do `GsonBuilder`. E, os próximos testes utilizarão a instância `Gson` desta classe.

Assim, a soma dos tamanhos dos arquivos é, 30,4 kB + 43,8 kB = 74,2 kB.

O espaço total de memória desta configuração: 74,2 kB.

Figura 37 – Trecho do código do método `gravaJsonNoHD` da classe `GerenciadorRelatorioJSON`, utilizando a classe `GsonBuilder`

```

119
120     File outFile = new File(contexto.getFolderTemp() + fileName + ".json");
121
122     try {
123
124         FileWriter writer = new FileWriter(outFile, true);
125
126         Gson gson = new GsonBuilder().disableHtmlEscaping().create();
127
128         gson.toJson(relatorioBean, RelatorioBean.class, writer);
129
130         writer.flush();

```

Fonte – O autor

[illegible]

Fonte – O autor

Figura 39 – Pasta temporária de usuário com os arquivos gerados na Configuração 5, com relatório de avaliação sem caracteres de escape

Fonte – O autor

### Teste da Configuração 6

A classe FachadaArquivador deve ser configurada com a constante `JSON_semHTML_comLINHAS`, do enumerador `AmbienteArquivacaoJSON`.

Logicamente, a ausência do HTML capturado, resulta para este teste o tamanho do relatório de avaliação, que manteve o tamanho 43,8 kB.

O espaço total de memória desta configuração: 43,8 kB.

## Teste da Configuração 7

Neste teste, a classe FachadaArquivador é configurada com a constante JSON\_semHTML\_semLINHAS, do enumerador AmbienteArquivacaoJSON.

A Figura 40 mostra a pasta temporária de usuário contendo o relatório de avaliação no formato JSON com 7,5 kB.

O espaço total de memória desta configuração: 7,5 kB.

Figura 40 – Pasta temporária de usuário com os arquivos gerados na Configuração 7



Fonte – O autor

#### 2.2.2.4 Resumo dos testes das Configurações

Os testes com as configurações de arquivção foram implementados e realizados na seção anterior, 2.2.2.3. Os resultados obtidos estão organizados na Tabela 4. Que herda os atributos da Tabela 2 acrescentando os resultados dos testes.

Tabela 4 – Valores obtidos nos testes com as configurações de arquivção para o IsAcessível

Conf.	Formato	HTML	Linhas	Nomenclatura	Espaço em memória utilizado
0	XML	SIM	SIM	Normal	93,2 KB
1	XML	SIM	SIM	Mínima	62,8 KB
2	XML	NÃO	SIM	Normal	52,7 kB
3	XML	NÃO	SIM	Mínima	17,0 KB
4	XML	NÃO	NÃO	Mínima	83,5 kB
5	JSON	SIM	SIM	SEM SUPORTE	74,2 kB
6	JSON	NÃO	SIM	SEM SUPORTE	43,8 kB
7	JSON	NÃO	NÃO	SEM SUPORTE	7,5 kB

Fonte – O autor

Vê-se que a Configuração 7, é a que menos ocupou espaço de memória.

Em relação, a Configuração 0, representa uma diminuição de, aproximadamente, 92%.

Outra observação possível é a influência das linhas do documento HTML no tamanho dos arquivos, em ambos os formatos, representando cerca de 41% do espaço de memória em relação a Configuração 0.

## 2.3 DISCUSSÃO

Pelos dados solicitados, nas telas de entrada, para realizar avaliação de acessibilidade, percebe-se que, tanto a versão do ASES, utilizada neste projeto, quanto o do código fonte, não estão realizando avaliações com as recomendações eMAG e WCAG mais recentes, a saber, eMAG 3.1 e WCAG 2.0. Tais dados são os níveis de prioridade 1, 2 e 3. Os quais eram relevantes até a versão eMAG 2.0 e não mais existem na versão atual. E, na versão WCAG, atual, é utilizado a ideia de níveis de conformidade A, AA e AAA, relacionados com critérios de sucesso. Não foi possível nenhum esforço para atualizar, porém o reconhecimento do problema contribui para motivar a continuidade do projeto.

Nota-se os primeiros desafios observando os arquivos de resultados de avaliação do ASES: arquivos de texto e XML, resultantes do processo de avaliação, são gerados e utilizados para realizar a visualização do usuário. Porém, o espaço de memória ocupado por estes arquivos, num ambiente web, não é o ideal, pois serão n usuários. Por isso, que a diminuição do espaço de armazenamento foi de vital importância para o sucesso deste trabalho, isso ocorreu com a troca de tecnologia de armazenamento e organização do arquivo.

Outro ponto, observado no código fonte do ASES, é a utilização de bibliotecas antigas para realizar a busca dos conteúdos HTML das páginas. Segundo o site que hospeda a biblioteca *Java Commons HttpClient*<sup>34</sup>, esta já foi substituída por outra. Desta forma, foi feita a troca desta rotina pelo API *crawler4j*<sup>35</sup> visando a construção de um sistema mais atual. Foi notada uma certa melhoria na performance, no entanto, como não faz parte do escopo deste trabalho, ela não foi avaliada.

Por fim, a análise do código fonte do ASES mostrou-se bastante difícil diante das muitas rotinas, espalhadas pelo código, para suprir as funcionalidades *desktop*, controladas por várias *threads*. Outro ponto que dificultou, foi o alto acoplamento entre classes. Estes enfrentamentos, exigiram modificação de código e exclusões de classes, levando considerável tempo com testes e depurações. Mesmo assim, foi possível a análise e abstração da lógica de avaliação de acessibilidade.

---

<sup>34</sup> <http://hc.apache.org/httpclient-3.x/>

<sup>35</sup> <https://github.com/yasserg/crawler4j>

### 3 CONCLUSÃO

Para desenvolver o IsAcessível, foi necessário conhecer as bases teóricas e práticas do tema acessibilidade na web. O que foi possível, tendo-se a oportunidade de conhecer o software ASES. Pois foi relevante no entendimento da funcionalidade Avaliador de Acessibilidade, antes obscura.

O levantamento dos conceitos de acessibilidade para web, juntamente, com o estudo do código fonte do ASES, possibilitou um maior vislumbre das tecnologias utilizadas para realizar avaliação, colaborando com a abstração da funcionalidade e criação de uma lógica voltada para o ambiente web.

Para produzir a interface web para o IsAcessível, contou-se com uma pré-configuração, anteriormente produzida.

Para estabelecer um gerenciamento sustentável do espaço de memória utilizado pelo IsAcessível, os testes com as configurações de arquivção demonstraram que o formato de arquivo JSON é o mais adequado em comparação com XML.

Para verificar o desempenho de acesso a links, foi integrado, com sucesso, a API crawler4j no projeto.

O IsAcessível vem para contribuir, sendo uma ferramenta voltada para web, realizando a avaliação de acessibilidade de uma forma inovadora: gerando relatórios de acessibilidade de várias páginas para o usuário em ambiente web.

No princípio, houve muitas dificuldades para compreender o tema acessibilidade na web. Também, conhecer o funcionamento interno das tecnologias estudadas e, ao mesmo tempo, aprender os fundamentos delas.

#### 3.1 TRABALHOS FUTUROS

Portanto, deixa-se como trabalhos futuros realizar testes com outras tecnologias de rastreamento na web.

Realização de um estudo mais aprofundado sobre formas de arquivos que propiciem melhorar a forma de armazenamento de arquivos.

Desenvolver um ambiente web mais confortável a proposta deste trabalho.



Pesquisar novas formas de promover acessibilidade web para buscar atualizações junto as necessidades da sociedade, como recomendações estrangeiras.

Implementar as tecnologias mais recentes visando entrar nos conceitos mais atuais da acessibilidade para web, principalmente as recomendações WCAG.

## REFERÊNCIAS

- ABINADER, J. A; LINS, R. D. **Web Services em Java**. Rio de Janeiro: Braspor, 2006. Disponível em: <[https://books.google.com.br/books?id=MUdQSavIynEC&printsec=frontcover&hl=pt-BR&source=gbg\\_summary\\_r&cad=0#v=onepage&q&f=false](https://books.google.com.br/books?id=MUdQSavIynEC&printsec=frontcover&hl=pt-BR&source=gbg_summary_r&cad=0#v=onepage&q&f=false)>. Acesso em: 24 nov.2015.
- ACESSIBILIDADE BRASIL. Fabio Issamu Oshiro; Guilherme De Azambuja Lira; Haroldo Veiga Gomes; Renato Tomaz Nati. **ASES – AVALIADOR E SIMULADOR DE ACESSIBILIDADE DE SÍTIOS**. BR n. PI 09267-1, 23 dez. 2008, 09 fev.2010. Disponível em: <<https://code.google.com/p/ases-acessibilidade/>>. Acesso em: 14 set.2015.
- BITV. *BIK BITV Test*, 2015. Disponível em: <[http://www.bitvtest.eu/about\\_us.html](http://www.bitvtest.eu/about_us.html)>. Acesso em: 09 nov. 2015.
- BRASIL. Ministério do Planejamento, Orçamento e Gestão. Secretaria de Logística e Tecnologia da Informação. Governo Eletrônico. **ASES: Avaliador e Simulador de Acessibilidade de Sítios versão 1.0.2 Linux**. Governo eletrônico, [2008]. Disponível em: <<http://www.governoeletronico.gov.br/acoes-e-projetos/e-MAG/ases-avaliador-e-simulador-de-acessibilidade-sitios>>. Acesso em: 27 de out.2015.
- CARTILHA de acessibilidade na web W3C Brasil. W3C Brasil. **GT Acessibilidade**, 2013. Disponível em: <<http://acessibilidade.w3c.br/cartilha/fasciculo1/>>. Acesso em: 05 nov. 2105.
- CHALEGRE, V. C. Uma metodologia de teste de acessibilidade para usuários cegos em ambientes Web. 2011. 194f. Dissertação (Mestrado em Ciência da Computação) – Universidade Federal de Pernambuco. Recife.
- COSTA, C. J. **Desenvolvimento para Web**. Lisboa: Lusocrédito, 2007. Disponível em: <[https://books.google.com.br/books?id=Jn6dTDF-wcsC&printsec=frontcover&hl=pt-BR&source=gbg\\_summary\\_r&cad=0#v=onepage&q&f=false](https://books.google.com.br/books?id=Jn6dTDF-wcsC&printsec=frontcover&hl=pt-BR&source=gbg_summary_r&cad=0#v=onepage&q&f=false)>. Acesso em: 24 nov. 2015.
- EMAG. Modelo de Acessibilidade em Governo Eletrônico. Governo Eletrônico, 2004. Disponível em: <<http://www.governoeletronico.gov.br/acoes-e-projetos/e-MAG>>. Acesso em: 14 set.2015.
- EMAG. Modelo de Acessibilidade em Governo Eletrônico – versão 2.0. Governo Eletrônico, 2005. Disponível em: <<http://www.governoeletronico.gov.br/biblioteca/arquivos/e-mag-versao-2.0/view>>. Acesso em: 12 nov. 2015.
- EMAG. Modelo de Acessibilidade em Governo Eletrônico – versão 3.1. Governo Eletrônico, 2014. Disponível em: <<http://www.governoeletronico.gov.br/acoes-e-projetos/e-MAG>>. Acesso em: 14 set.2015.
- FEIJÓ, A. R. A. O direito constitucional da acessibilidade das pessoas portadoras de deficiência ou com mobilidade reduzida. **Portal de e-governo, inclusão digital e sociedade do conhecimento**, 05 mar. 2011. Disponível em: <<http://www.egov.ufsc.br/portal/conteudo/o-direito-constitucional-da-acessibilidade-das-pessoas-portadoras-de-defici%C3%Aancia-ou-com-mob>>. Acesso em: 07 nov. 2015.
- FERREIRA, A. B. de H. **Dicionário Aurélio escolar da língua portuguesa**. Rio de Janeiro: Nova Fronteira, 1988.
- GSA. **GSA Section508.gov**, 2015. Disponível em: <<http://www.section508.gov/>>. Acesso em: 03 nov.2015.

IBGE. **Censo demográfico 2010**. Características gerais da população, religião e pessoas com deficiência. Rio de Janeiro: IBGE, 2010. Disponível em:

<[http://biblioteca.ibge.gov.br/visualizacao/periodicos/94/cd\\_2010\\_religiao\\_deficiencia.pdf](http://biblioteca.ibge.gov.br/visualizacao/periodicos/94/cd_2010_religiao_deficiencia.pdf)>. Acesso em: 4 de nov.2015.

MACIEL, A. M. A. Investigação de um ambiente para o desenvolvimento integrado de interface de voz. 2012. 145f. Tese (Doutor em Ciência da Computação) – Universidade Federal de Pernambuco. Recife.

MARCONI, M. A.; LAKATOS, E. M. **Fundamentos de metodologia científica**. 5. ed. São Paulo: Atlas, 2003.

MARQUES, H. R. et al. **Metodologia de pesquisa e do trabalho científico**. Campo Grande: UCDB, 2006.

NDA. *National Disability Authority*, 2012. Disponível em: <<http://nda.ie/>>. Acesso em: 03 nov.2015.

NICÁCIO, M. J. **Técnicas de acessibilidade**: criando uma web para todos. Maceió: EDUFAL, 2010.

OLIVEIRA, G. A. de. **O uso de Meta tags no processo de recuperação de informação na web**: uma análise de busca dos periódicos eletrônicos da área de CI. João Pessoa: UFPB, 2011. Disponível em: <[https://books.google.com.br/books?id=vzs6xuxDgfmC&printsec=frontcover&hl=pt-BR&source=gbg\\_summary\\_r&cad=0#v=onepage&q&f=false](https://books.google.com.br/books?id=vzs6xuxDgfmC&printsec=frontcover&hl=pt-BR&source=gbg_summary_r&cad=0#v=onepage&q&f=false)>.

Acesso em: 26 nov.2015.

RICARDO, A. SERRÃO, C. Comparison of existing open-source tools for Web crawling and indexing of free Music. [S.I]: **Journal of Telecommunications**, v. 18, issue 1, 31 Jan. 2013. Disponível em: <<http://pt.scribd.com/doc/123153248/Comparison-of-existing-open-source-tools-for-Web-crawling-and-indexing-of-free-Music#scribd>>. Acesso em: 26 nov. 2015.

SMITH, B. **JSON básico**. São Paulo: Novatec Editora, 2015. Disponível em:

<[https://books.google.com.br/books?id=aGOsCQAAQBAJ&printsec=frontcover&hl=pt-BR&source=gbg\\_summary\\_r&cad=0#v=onepage&q&f=false](https://books.google.com.br/books?id=aGOsCQAAQBAJ&printsec=frontcover&hl=pt-BR&source=gbg_summary_r&cad=0#v=onepage&q&f=false)>. Acesso em: 24 nov.2015.

SOARES FILHO, A. G. T. Abordagem dos elementos da experiência do usuário aplicada ao design de uma interface de webmail acessível. 2015. 232f. Dissertação (Mestrado em Ciência da Computação) – Universidade Federal de Pernambuco. Recife.

W3C. **Current Members**, 2015. Disponível em:

<<http://www.w3.org/Consortium/Member/List>>. Acesso em: 11 nov. 2015.

W3C. **Diretrizes de Acessibilidade para Conteúdo Web (WCAG) 2.0**, 11 dez. 2008.

Disponível em: <<http://www.w3.org/Translations/WCAG20-pt-br/>>. Acesso em: 10 nov. 2015.

W3C. **Understanding WCAG 2.0: A guide to understanding and implementing Web Content Accessibility Guidelines 2.0**, 26 fev. 2015. Disponível em:

<<http://www.w3.org/TR/UNDERSTANDING-WCAG20/Overview.html>>. Acesso em: 11 nov. 2015.

W3C. **WAI Working Groups and Interest Groups**, 2012. Disponível em:

<<http://www.w3.org/WAI/groups>>. Acesso em: 03 nov.2015.

W3C. **Web Accessibility Evaluation Tools List**, 18 dez. 2014. Disponível em:

<<http://www.w3.org/WAI/ER/tools/index.html>>. Acesso em: 28 out.2015.

W3C. **Web Content Accessibility Guidelines 1.0**, 05 mai. 1999. Disponível em: <<http://www.w3.org/TR/WAI-WEBCONTENT/>>. Acesso em: 10 nov. 2015.

WHO. *Media centre. Fact sheets. Disability and health*. Geneva: World Health Organization, 2015. Disponível em: <<http://www.who.int/mediacentre/factsheets/fs352/en/>>. Acesso em: 05 nov.2015.

YADAV, M.; GOYAL, N. *Comparison of Open Source Crawlers – A Review*. [S.I.]: **International Journal of Scientific & Engineering Research**, v. 6, issue 9, Set. 2015. Disponível em: <<http://www.ijser.org/researchpaper%5CComparison-of-Open-Source-Crawlers--A-Review.pdf>>. Acesso em: 26 nov. 2015.