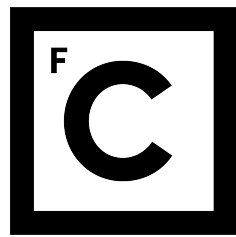


UNIVERSIDADE DE LISBOA
FACULDADE DE CIÊNCIAS
DEPARTAMENTO DE INFORMÁTICA



Ciências
ULisboa

PLACM - Prototype Large-Scale Accessibility Compliance Monitoring

Bruno Miguel Costa de Andrade

Mestrado em Engenharia Informática

Interação e Conhecimento

Dissertação orientada por:

Prof. Doutor Carlos Alberto Pacheco dos Anjos Duarte

Prof. Doutor Luís Manuel Pinto da Rocha Afonso Carriço

2020

Agradecimentos

Este ano, o mais longo de todo o caminho académico, mostrou-se muito mais complicado do que esperado. A inquietação e a ansiedade alojaram-se. Felizmente, em todo este percurso, estive acompanhado de indispensáveis, grandes pessoas.

Quero agradecer à minha mãe e ao meu pai, por todos os valores, educação, raspanetes e cuidado que me ofereceram ao longo deste par de décadas. Agradeço aos meus avós, tios e primos, pelo zelo e compreensão. Um agradecimento especial à minha madrinha, que sempre me apoiou desde o primeiro bloco de folhas, sentadinho na banca. Obrigado por estarem sempre perto.

Quero agradecer à minha namorada, Patrícia. Obrigado por todas as palavras de alento e motivação, pelos abanões e pelos abraços. Obrigado pelo sorriso.

Quero agradecer aos meus orientadores, o professor Luís Carriço e o professor Carlos Duarte, pelo enorme apoio oferecido em todo o mestrado, pela constante disponibilidade e repetido incentivo. Foi um privilégio presenciar o seu carinho e dedicação pelo trabalho nesta área. Com todo o respeito e admiração, obrigado pelo conhecimento transmitido.

Quero agradecer a todos os meus amigos, fora e dentro da universidade, por todas as boas e felizes memórias. Um especial obrigado a Estriga, “Nisco”, Alex, Diogo, “Neelo”, Nuno, Luís, “Bubas”, João, Carolina, Elsa, DT, Teresa, Bruno, Glória, Tomás, Pedro e Inês. Obrigado por tornarem o caminho mais leve.

Muito obrigado.

Resumo

Com o aumento da modernização da sociedade, a Internet já se transformou numa ferramenta utilizada todos os dias por milhões de pessoas. Mas com esta magnitude, nem todos os serviços ou aplicações disponíveis na web são acessíveis para todos.

Estima-se que cerca de um quinto da população da União Europeia tem algum tipo de incapacidade, quer física, cognitiva, visual ou auditiva. Com esta percentagem significativa, é importante que a *web* forneça alternativas acessíveis, de modo a incluir o máximo de cidadãos, promovendo a independência dos mesmos e contribuindo para uma maior equidade social.

A W3C desenvolveu um conjunto de recomendações para desenvolvedores *web* que desejam tornar o seu conteúdo acessível, sendo já consideradas como o padrão a seguir em legislações internacionais. Porém, por serem difíceis de interpretar, foi criado o projeto WAI-Tools, no qual um dos objetivos principais é o desenvolvimento de regras claras e reproduzíveis, focadas nas recomendações mencionadas. Com estas regras, é despoletada a harmonização dos resultados, ferramentas e métodos de avaliação de acessibilidade. Um outro objetivo do projeto foi a criação de um formato único de resultados de testes de acessibilidade, permitindo que várias ferramentas possam estruturar da mesma maneira os seus resultados. Um último objetivo, e foco deste trabalho, era o desenvolvimento de uma plataforma de monitorização de conformidade de acessibilidade em grande-escala onde, com a conciliação das regras desenvolvidas e o formato normalizado dos resultados daí originados, fosse possível agrupar informação de várias ferramentas e entidades e incentivar a uma constante melhoria e apreço pela acessibilidade digital.

Para alcançar este objetivo, foi realizado um levantamento de requisitos de plataformas de monitorização existentes, dentro e fora do âmbito de acessibilidade, identificando as suas necessidades e funcionalidades comuns. Tendo em conta este estudo, e pareceres de alguns parceiros do projeto europeu, foi desenhado um modelo de dados e estruturados os tipos de navegação, e correspondentes vistas gráficas, apropriados aos dados a monitorizar. Depois do processo de implementação da plataforma, foram realizados testes com utilizadores para verificar se a plataforma apresentava problemas de usabilidade, e, por fim, realizadas alterações tendo em conta os resultados dos testes.

Palavras-chave: acessibilidade web, monitorização de acessibilidade, conformidade, dados em grande escala

Abstract

With the increasing modernization of processes in society, the Internet has already become a tool used by millions of people every day. Despite this magnitude, not all services or applications available on the web are accessible to everyone.

It is estimated that about a fifth of the population in the European Union has some kind of disability, whether physical, cognitive, visual or hearing. With this significant percentage, it is important that the web provides accessible alternatives, in order to include as many citizens as possible, promoting their independence and contributing to greater social equity.

The W3C has developed a set of recommendations for web developers who want to make their content accessible and are already considered to be the standard to follow in international legislation. However, as they are difficult to interpret, the WAI-Tools project was created, in which one of the main objectives is the development of clear and reproducible rules, focused on the mentioned recommendations. With these rules, the harmonization process of results, tools and methods for assessing accessibility is triggered. Another objective of the project was to create a standardized format for accessibility test results, allowing several tools to structure their results in the same way. A final objective, and focus of this dissertation, was the development of a large-scale accessibility compliance monitoring platform where, with the conciliation of the developed rules and the standardized format of the results resulting from it, it should be possible to group information from various tools and entities and encourage constant improvement and appreciation for digital accessibility.

To achieve this goal, a survey of requirements for existing monitoring platforms was carried out, both within and outside the scope of accessibility, identifying their common needs and features. Taking this survey into account, and the opinions of some partners of the European project, a data model was designed and the types of navigation appropriate to the data to be monitored, and corresponding graphical views, were structured. After the platform implementation process, tests were carried out with users to check if the platform had usability problems, and, finally, changes were made considering the results of these tests.

Keywords: web accessibility, accessibility monitoring, conformance, large-scale data

Índice

Lista de Figuras	ix
Lista de Tabelas	xi
1 Introdução	1
1.1 Motivação	1
1.2 Objetivos	2
1.3 Metodologia	3
1.4 Contribuições	3
1.5 Estrutura do documento	3
2 Contexto	5
2.1 Legislação atual	5
2.2 Web Accessibility Initiative (WAI)	6
2.3 Projeto WAI-Tools	7
3 Trabalho relacionado	13
3.1 Monitorização de dados em grande escala	13
3.2 Monitorização de acessibilidade	19
3.3 Discussão	22
4 Desenho	25
4.1 Levantamento de requisitos	25
4.2 Modelo de dados	31
4.3 Tecnologias usadas	41
5 Implementação	45
5.1 Origem e interpretação de dados	45
5.2 Vistas de gráficos	48
5.3 Formas de navegação	49
5.4 Modificações ao Highcharts	56
5.5 Arquitetura	57
5.6 Crawler	60

ÍNDICE

6	Avaliação	61
6.1	Testes com utilizadores	61
6.2	Desempenho	63
7	Conclusão	65
7.1	Contribuições e limitações	66
7.2	Trabalho futuro	68
	Bibliografia	69
A	Relação entre regras ACT e tipos de elementos avaliados	71
B	Lista de perguntas usada nos testes com utilizadores	75
C	Avaliação do tempo de resposta dos pedidos relacionados com as métricas de critérios de sucesso	77

Lista de Figuras

3.1	Informação global do AtlantOS e gráficos relacionados com tipos de plataformas de observação	14
3.2	Estados de atividade das plataformas, agrupados por tipo de plataforma	15
3.3	Evolução temporal dos pedidos de <i>downloads</i> dos dados	15
3.4	Gráfico de linhas que representa a evolução temporal do indicador 6.1.1 por região	16
3.5	Gráfico de colunas que representa a quantidade de água em rios na região de Europa e América do Norte, a partir de dados nacionais	16
3.6	Gráfico de colunas empilhadas que representa a extensão espacial dos diferentes corpos de água na região de Europa e América do Norte, a partir de dados nacionais	16
3.7	Gráfico de radar que apresenta informação do indicador 6.5.1 relativo à região de Europa e América do Norte, onde cada linha simboliza um país dessa região	17
3.8	Gráfico de linhas que representa a evolução temporal dos valores de cobre e ferro registados no rio Guandu	18
3.9	Gráfico de dispersão onde estão representados os valores dos registos de cobre e ferro do rio Guandu	19
3.10	Amostra de registos das medições de qualidade de ar das 3 estações de monitorização, representados numa tabela	20
4.1	Exemplo de asserção originada ao avaliar uma página com a regra “Image filename is accessible name for image”	27
4.2	Modelo de dados do PLACM	33
5.1	Visualização predefinida do PLACM	50
5.2	Janela modal, no separador <i>Group by</i>	50
5.3	Vista das métricas relacionadas com os critérios de sucesso, resultante da navegação para os dados de ‘Aplicação/ <i>Website</i> ’ a partir do continente europeu	51
5.4	Vista de único gráfico resultante da comparação dos dados relativos às categorias de cultura, saúde, justiça e média	53

LISTA DE FIGURAS

5.5	Vista resultante da navegação temporal, apresentando a evolução temporal dos dados do setor privado	54
5.6	Vista resultante da navegação de conformidade dos critérios de sucesso numa aplicação/ <i>website</i> , apresentando uma listagem, por resultado, de todos os critérios de sucesso	55

Lista de Tabelas

4.1	Classes estudadas no PLACM, a origem dos seus dados e exemplos dos mesmos	28
A.1	Relações entre as regras ACT e os tipos de elemento avaliados	71
A.2	Relações entre as regras ACT e os tipos de elemento avaliados (<i>continuação</i>)	72
A.3	Relações entre as regras ACT e os tipos de elemento avaliados (2 ^a <i>continuação</i>)	73
C.1	Tempo de resposta dos pedidos de dados respetivos às classes relacionadas com as métricas de critérios de sucesso, antes das alterações	77
C.2	Tempo de resposta dos pedidos de dados respetivos às classes relacionadas com as métricas de critérios de sucesso, após alterações	77
C.3	Diminuição percentual do tempo de resposta dos pedidos de dados respetivos às classes relacionadas com as métricas de critérios de sucesso, após alterações	78

Capítulo 1

Introdução

Todos os dias, a Internet é acessada por milhões de utilizadores e cada um destes utilizadores é diferente de todos os outros. Com cada vez mais aplicações e serviços a emergirem na Web, com o desejo de facilitar o dia-a-dia da sociedade, todos os utilizadores têm o direito de os utilizar com equidade e isso, infelizmente, não acontece. Muitas páginas e aplicações Web continuam a não ser acessíveis, impedindo o acesso a utilizadores com deficiências físicas, cognitivas, visuais ou até auditivas.

Para impulsionar o alcance da acessibilidade web, a World Wide Web Consortium (W3C) elaborou as Web Content Accessibility Guidelines (WCAG), um conjunto de recomendações técnicas que auxiliam o desenvolvimento de conteúdo web acessível em qualquer tipo de dispositivo. Estas diretrizes estão divididas em quatro grandes princípios: o conteúdo web deve ser perceptível (com informação apresentada facilmente distinguível), operável (a interface não pode exigir interação que um utilizador não consegue realizar), compreensível (o conteúdo ou operação não pode ser demasiado complexa) e robusto (independentemente do utilizador ou tecnologia usada, o conteúdo continua acessível). Por sua vez, os quatro princípios agrupam dezenas de critérios de sucesso, isto é, declarações de como o conteúdo deve estar estruturado, ou como deve reagir aos utilizadores, de maneira acessível.

1.1 Motivação

Atualmente, as WCAG são internacionalmente reconhecidas e adotadas como o padrão para desenvolver conteúdo web acessível, sendo até a base de legislações existentes, tais como a diretiva relativa à acessibilidade da União Europeia [1] e respetiva transposição para a lei portuguesa [2]. Para assegurar que as legislações estão a ser cumpridas, e as diretrizes estão a ser corretamente empregues, é necessário realizar avaliações de acessibilidade ao conteúdo digital.

Já existem ferramentas de avaliação, com metodologias manuais, semiautomáticas e automáticas, usadas para verificar o nível de conformidade de conteúdo digital. A

1. INTRODUÇÃO

avaliação de acessibilidade é especialmente importante para encontrar e ajudar a corrigir possíveis falhas e, conseqüentemente, impulsionar o cumprimento das legislações atuais. Contudo, sem serem agrupados os resultados encontrados, a avaliação perde a sua relevância, por não ser possível nem estudar a evolução da acessibilidade do conteúdo avaliado nem a comparação com outro conteúdo.

Alguns governos e entidades correspondentes desenvolveram plataformas de monitorização chamadas de Observatórios de Acessibilidade. Nestes, são realizados estudos e é visualizado o estado de conformidade, por exemplo, nacional em certas categorias. Mas os dados usados são oriundos de ferramentas singulares de avaliação e os resultados, por vezes, privados.

No projeto europeu onde o trabalho desta dissertação se encontra, WAI-Tools, foi criado um formato único para os resultados de testes de acessibilidade Web. Este formato permite que, independentemente da ferramenta utilizada para avaliação, os resultados sejam estruturados seguindo o mesmo padrão. Como fruto desta padronização de resultados, uma grande quantidade de informação pode ser reunida e, assim, monitorizada eficientemente.

O desenvolvimento de uma plataforma que concentre dados de várias ferramentas e várias entidades tornou-se uma possibilidade graças a esta padronização de resultados de avaliação. Uma plataforma de monitorização centralizada, coerente e pública ajudaria a incentivar competição entre entidades para alcançar um maior nível de conformidade de acessibilidade, crescendo e melhorando em conjunto, e facilitaria a realização de estudos estatísticos, temporais ou relativos ao cumprimento legislativo.

Com a motivação de destacar a importância de monitorização de dados de acessibilidade, o trabalho desta dissertação concentra-se no desenho e implementação da plataforma PLACM.

1.2 Objetivos

O maior foco deste trabalho é a prototipagem de uma ferramenta de monitorização de conformidade de dados de acessibilidade em grande escala, de modo a permitir visualizar informação agregada de alto-nível, tal como a conformidade a um critério de sucesso num país ou resultados de testes realizados ao conteúdo web de uma entidade.

Para alcançar este propósito, os objetivos desta plataforma passam por:

- Ser usável;
- Ser acessível, suportando navegação por teclado e uso adequado de leitor de ecrã;
- Permitir a exportação de dados num formato útil para quem faz monitorização;
- Permitir uma comparação direta entre conjuntos de dados;

- Permitir aplicar filtros aos dados;
- Permitir estudar a evolução temporal dos dados.

1.3 Metodologia

Para que fosse possível começar a desenvolver uma ferramenta de monitorização de conformidade de acessibilidade, foi necessário realizar um levantamento de requisitos, juntamente com os parceiros do projeto WAI-Tools, para que fossem discutidas as necessidades e objetivos desta plataforma numa comunidade focada em cumprir o objetivo. Juntamente com o parecer dos parceiros, foram estudadas várias plataformas de monitorização de dados, algumas em grande escala, para perceber as suas funcionalidades e tipos de visualização. Com o resultado destes estudos, e com a plataforma já com os seus objetivos definidos, começou o desenvolvimento do PLACM.

Durante a implementação, foram realizadas várias iterações para melhorar a capacidade de resposta da plataforma, tendo em conta a grande quantidade de dados a apresentar. Após concluída a implementação da plataforma, foram preparados e feitos testes com utilizadores, para perceber se a usabilidade da plataforma era adequada ao utilizador comum. A identificação de problemas e melhorias possíveis na plataforma durante estes testes desencadeou novamente o processo de iteração, recomeçando o ciclo, com início no desenvolvimento.

1.4 Contribuições

Nesta dissertação, foi desenvolvida uma ferramenta de monitorização de conformidade de dados de acessibilidade em grande escala. Esta foi publicada, em código-aberto, no âmbito do projeto WAI-Tools, sendo um dos seus *deliverables*. Para que este desenvolvimento fosse possível, foram ainda identificados requisitos comuns de plataformas de monitorização existentes.

1.5 Estrutura do documento

Este documento divide-se em 7 capítulos, que seguem a seguinte estrutura:

- Capítulo 2: Contexto - Neste capítulo é introduzido o contexto do PLACM no projeto europeu onde se encontra, WAI-Tools, e as etapas anteriores necessárias para o desenvolvimento do mesmo. É ainda apresentada a legislação atual relacionada com acessibilidade e o seu emprego no projeto.

1. INTRODUÇÃO

- Capítulo 3: Trabalho relacionado - Este capítulo apresenta descrições pormenorizadas de algumas plataformas de monitorização existentes, dentro e fora do âmbito de acessibilidade, concluindo com uma discussão dos requisitos comuns das mesmas.
- Capítulo 4: Desenho - Neste capítulo são descritos os requisitos levantados após o estudo prévio de plataformas existentes combinado com o parecer dos parceiros do WAI-Tools. É ainda descrito o modelo de dados do PLACM e as tecnologias a serem utilizadas no seu desenvolvimento.
- Capítulo 5: Implementação - Este capítulo consiste na descrição dos aspetos mais importantes do desenvolvimento do PLACM, desde a introdução e interpretação de dados, cálculo das métricas apresentadas, formas de navegação e respetivas vistas gráficas e arquitetura do PLACM.
- Capítulo 6: Avaliação - Neste capítulo são discutidas a metodologia e resultados dos testes de utilizadores e algumas modificações realizadas para melhorar o desempenho do PLACM.
- Capítulo 7: Conclusão - Neste capítulo é feita uma breve conclusão, resumindo o trabalho realizado e uma análise crítica, considerando os esforços realizados e limitações encontradas. É ainda listado um conjunto de possíveis alterações futuras para uma melhoria do trabalho realizado.

Capítulo 2

Contexto

Gradualmente, a acessibilidade tem vindo a tornar-se uma preocupação para o mundo digital. A criação de leis e diretivas para a acessibilidade digital são uns dos felizes resultados dessa mesma preocupação.

De modo a contextualizar o trabalho a desenvolver, é necessário enquadrá-lo no projeto que o despoletou e nas diretivas já estabelecidas. O desenvolvimento do PLACM é vinculado ao projeto WAI-Tools, sendo um *deliverable* da sua última etapa. Como tal, todo o trabalho desenvolvido foi circunscrito aos objetivos deste projeto, à opinião e parecer da comunidade envolvida no mesmo e à atual legislação europeia.

2.1 Legislação atual

Os organismos europeus de normalização CEN/CENELEC e ETSI foram solicitados pela Comissão Europeia a produzir uma norma que correlacionasse a acessibilidade às Tecnologias de Informação e Comunicação (TIC). A norma que daí surgiu, a EN 301 549 [1], tem como objetivo definir uma diretriz à legislação sobre contratação pública na Europa, com início em 2016, mas com a ambição que a mesma se espalhe também para o setor privado, de forma a manter a competição. Com a implementação desta norma, todos os Estados Membros da União Europeia passam a ter uma única legislação sem desacordos nem discrepâncias.

A EN 301 549 [1] estabelece diretrizes que devem ser tomadas em conta no desenvolvimento das TIC e muitas das ferramentas que conhecemos são consideradas TIC, tais como as caixas multibanco, documentos digitais e conteúdo Web. Contudo, é distinguido para o caso do conteúdo Web que este deve estar de acordo com o nível de conformidade AA das diretrizes WCAG 2.1, diretrizes, descritas na próxima secção, que especificam requerimentos de acessibilidade relativos apenas à Web.

Em outubro de 2016, o Parlamento Europeu e o Conselho da União Europeia decretaram a diretiva 2016/2102 [3]. Esta, aplicando a norma EN 301 549, impõe que os Estados-Membros da União Europeia (UE) assegurem que os seus organismos públicos

2. CONTEXTO

tomam medidas para tornar os seus *websites* e aplicações móveis acessíveis, de modo a alcançar o nível de conformidade AA das WCAG 2.1. É também imposto aos Estados-Membros que os seus organismos públicos apresentem periodicamente uma declaração de acessibilidade clara. É de destacar ainda que os Estados-Membros necessitarão de apresentar periodicamente à Comissão um relatório sobre o resultado de monitorização de acessibilidade realizadas, impondo assim a responsabilidade de monitorizar e supervisionar a conformidade da acessibilidade nos seus organismos públicos.

Como diretiva, esta publicação dá o direito dos Estados-Membros de manter estes requisitos de acessibilidade mínimos descritos ou introduzir medidas mais exigentes para a sua legislação nacional, tal como aconteceu em Portugal. Foi então publicado, a 19 de outubro de 2018, o decreto-lei nº 83/2018 [2]. Este decreto-lei aplica-se, tal como a diretiva 2016/2102, a todos os conteúdos Web, independentemente do dispositivo, e aplicações móveis de organismos do setor público. Passou então a ser obrigatório (versão atual desde junho de 2018) cumprir o nível de conformidade AA definido nas WCAG 2.1. Para o cumprimento deste decreto, foi dada a responsabilidade à AMA¹ de várias ações de acompanhamento, sendo uma delas a criação e manutenção do Observatório Português da Acessibilidade dos Sítios Web e das Aplicações Móveis², uma ferramenta de monitorização de acessibilidade, detalhado no próximo capítulo.

2.2 Web Accessibility Initiative (WAI)

A WAI³ é uma iniciativa criada pela World Wide Web Consortium (W3C) com o objetivo de desenvolver diretrizes de acessibilidade para conteúdo digital, promovendo a sua harmonização internacional. Oferecendo este tipo de recursos, a WAI pretende educar os desenvolvedores Web na criação e desenvolvimento de *websites* e aplicações acessíveis para todas as pessoas. Além disso, em parceria com organizações por todo o mundo, a WAI coordena investigações e desenvolvimentos relativos a necessidades digitais de acessibilidade, tornando a educação e alcance neste campo um processo incessável.

2.2.1 Web Content Accessibility Guidelines (WCAG)

As Web Content Accessibility Guidelines (WCAG) [4] são um conjunto de diretrizes de acessibilidade desenvolvidas pela WAI com foco no conteúdo digital e atualmente encontram-se na versão 2.1. Com o aumento da informação e entendimento relativamente às limitações dos utilizadores, e com o desenvolvimento constante das tecnologias e metodologias usadas para a criação de conteúdo digital, as diretrizes têm de estar a par com estas condições. A última versão foi publicada em junho de 2018 mas já se encontra

¹<https://www.ama.gov.pt/>

²<https://observatorio.acessibilidade.gov.pt/>

³<https://www.w3.org/WAI/>

2.3 Projeto WAI-Tools

planeada a publicação da versão 2.2 durante o ano de 2021. A maior ambição destas diretrizes é abranger todas as limitações, de modo a proporcionar uma experiência igualmente adequada para toda a comunidade digital. Infelizmente, existem casos de limitações extremas que são virtualmente impossíveis de contornar.

As diretrizes WCAG têm como finalidade orientar o desenho de conteúdo Web acessível, passando pelo texto, imagens, código ou estrutura, independentemente da tecnologia usada. As WCAG 2.1 estão organizadas em 13 diretrizes não testáveis, divididas por quatro princípios: quem quiser usar a Web deve usufruir de conteúdo perceptível (com informação apresentada facilmente distinguível), operável (a interface não pode requerer interação que um utilizador não consegue realizar), compreensível (o conteúdo ou operação não pode ser demasiado complexa) e robusto (independentemente do utilizador ou tecnologia usada, o conteúdo continua acessível).

Por sua vez, as diretrizes são compostas por critérios de sucesso. Os critérios de sucesso enunciam como o conteúdo digital deve, por exemplo, estar estruturado ou reagir à interação do utilizador, sendo por isso testáveis de forma automática, semi-automática ou manual. Os critérios estão distribuídos por três níveis de conformidade: A, AA e AAA, sendo A o nível que indica conformidade mínima de acessibilidade.

Como estes critérios de sucesso são escritos de modo a abranger qualquer tecnologia, é necessário um esforço de interpretação adicional quando é desejado cumprir estes critérios numa tecnologia específica. O processo de interpretação e aplicação destas diretrizes acaba por não ser elementar, dificultando a implementação de acessibilidade.

2.3 Projeto WAI-Tools

O projeto WAI-Tools⁴, cofinanciado pela Comissão Europeia sob o programa Horizon 2020, foi iniciado em novembro de 2017 e tem término previsto para janeiro de 2021. Este projeto tem como objetivo central desenvolver recursos para testagem e monitorização de conformidade de acessibilidade, permitindo uma interpretação irrefutável, transparente e comum dos critérios de sucesso enunciados nas WCAG. Como parceiros deste projeto, encontram-se a Faculdade de Ciências da Universidade de Lisboa (FCUL) e a Agência para a Modernização Administrativa (AMA).

O WAI-Tools está dividido em três grandes etapas, cada uma com os seus focos de trabalho. Uma das etapas focou-se no desenvolvimento de regras ACT (*Accessibility Conformance Testing*). Como cerne do projeto, este desenvolvimento manteve-se durante os três anos do mesmo. As regras ACT têm como objetivo fornecer uma interpretação clara de enfoques de critérios de sucesso oriundos das WCAG, de modo a normalizar a implementação dos testes de acessibilidade nas ferramentas de avaliação e os seus resultados. Estas regras servem de base para uma grande parte do trabalho das restantes etapas do projeto. Numa outra etapa, o objetivo era o desenvolvimento de um formato único

⁴<https://www.w3.org/WAI/about/projects/wai-tools/>

2. CONTEXTO

de resultados de testes de acessibilidade, para que, esta informação seja facilmente interpretável independentemente da ferramenta usada para avaliação. Foi ainda nesta etapa que foram aplicadas as regras ACT nos observatórios português e norueguês.

A última etapa do projeto baseou-se na integração das regras ACT na monitorização de acessibilidade. No contexto da mesma, mas ainda no primeiro ano do projeto, foram desenhados geradores de declarações de acessibilidade e de relatórios baseados no formato único de resultados de testes. Um último objetivo desta etapa era o desenvolvimento de um protótipo, baseado nos padrões e recursos existentes, que proporcione uma navegação coerente por grandes quantidades de resultados de testes de acessibilidade. A concretização do protótipo descrito é o objetivo principal desta dissertação.

O trajeto das etapas do projeto levou à necessidade do protótipo: o desenvolvimento das regras ACT permite que ferramentas diferentes obtenham o mesmo resultado quando testadas as mesmas circunstâncias no conteúdo digital e o formato único desenvolvido permite relatar os resultados dessas ferramentas, permitindo assim reunir resultados de diversas fontes de forma escalável e concisa. Atualmente, existem 4 ferramentas de avaliação automáticas e 2 metodologias manuais que já implementam as regras ACT⁵, sendo uma das ferramentas automáticas, Qualweb⁶, desenvolvida por uma equipa do LA-SIGE. Todas as 6 já fornecem os seus resultados no formato único desenvolvido, automatizando, e facilitando, cada vez mais o processo de avaliação de acessibilidade. Mas apesar da criação destes modelos, não existe nenhuma ferramenta de monitorização que lhes tire proveito.

A visualização de resultados de testes de acessibilidade em grande-escala permite trazer uma noção do alcance de implementação das regras ACT, para além de incentivar organismos a empregar o trabalho desenvolvido e melhorar a acessibilidade que proporcionam nas suas plataformas. Acima de tudo, a visualização de resultados de testes de acessibilidade agiliza a monitorização da conformidade de acessibilidade, evidenciando a evolução dos esforços feitos ao longo do tempo e facilitando o cumprimento das legislações criadas.

2.3.1 Regras ACT

Com a conformidade de acessibilidade a tornar-se legislada por todo o mundo, as entidades responsáveis por conteúdo digital necessitam de estar a par com as condições necessárias para alcançar esta conformidade; no caso da União Europeia, ditado na 2016/2102, as diretrizes WCAG. Contudo, a interpretação das WCAG, sendo estas independentes de tecnologia, obriga a um esforço adicional quando focada numa tecnologia. Foi então formada uma comunidade aberta, denominada por *ACT Rules Community Group* [5], com o objetivo de transpor interpretações das WCAG em regras ACT, se-

⁵<https://act-rules.github.io/pages/implementations/overview/>

⁶<http://www.qualweb.di.fc.ul.pt/evaluator/>

guindo o padrão *ACT Rules Format*⁷. Sendo assim, as regras ACT são uma descrição, em linguagem simples, de como testar um tipo de conteúdo tendo em conta um enfoque de requisitos de acessibilidade. Para o projeto WAI-Tools, os requisitos de acessibilidade considerados serão os critérios de sucesso concedidos pelas WCAG, apesar das regras ACT poderem ser usadas para testar requisitos que não fazem parte de um padrão W3C.

O desenvolvimento de regras permite a documentação de metodologia de testagem de critérios de sucesso de uma maneira clara e reproduzível, motivando a harmonização dos resultados, ferramentas e métodos de testagem de acessibilidade. Com esta harmonização, os testes são realizados da mesma maneira, independentemente de serem manuais ou automáticos, resultando numa maior transparência e compreensão por parte dos utilizadores de conteúdo digital e dos desenvolvedores de ferramentas de avaliação. Apesar de ser o objetivo principal do primeiro ano do WAI-Tools, o desenho de regras ACT não se limitou apenas a esse ano; é um esforço contínuo, sendo ambicionado alcançar o máximo de requisitos de acessibilidade possível.

2.3.2 Formato normalizado de resultados

Numa das etapas do projeto, o objetivo principal era desenvolver um formato normalizado de resultados de testes, tanto automáticos como manuais. Um formato deste género permite interpretar resultados de vários avaliadores, potenciando a combinação de informação de várias fontes e, conseqüentemente, o processo de monitorização de conformidade de acessibilidade.

Este formato baseia-se na serialização JSON-LD⁸, um formato baseado em JSON para serializar dados ligados entre si, da *Evaluation and Report Language* (EARL) [6], um vocabulário flexível criado para descrever resultados de testes. Para normalizar o formato de resultados de testes, foi necessário dividir e categorizar a informação pertinente das avaliações, e os seus resultados, em estruturas, que serão explicadas de seguida.

2.3.2.1 Assertion

A *assertion* é a estrutura que relaciona todas as outras, pois é a expressão do resultado de cada teste realizado.

2.3.2.2 Assertor

O *assertor* é a entidade, que pode ser uma pessoa, ferramenta de *software*, organização ou grupo (de pessoas ou de ferramentas de *software*), que realiza os testes de modo a produzir *assertions*.

⁷<https://www.w3.org/TR/act-rules-format/>

⁸<https://www.w3.org/TR/json-ld/>

2. CONTEXTO

2.3.2.3 TestSubject

O *test subject* é especificado na EARL como recurso que foi testado contra algum critério de teste. É ainda recomendado que seja especificado qual o tipo de recurso, tal como *software*, conteúdo, resposta HTTP ou documento, apesar desta especificação não ser estritamente necessária.

2.3.2.4 TestCriterion

O *test criterion* é denominado como uma declaração testável. Esta estrutura abrange todos os tipos de testes, até fora do campo de acessibilidade, apesar de serem apenas apreciadas as regras ACT no projeto desta dissertação.

2.3.2.5 TestMode

O *test mode* é a estrutura que indica como foi realizado o teste, refletindo a informação deixada pelo tipo de *assertor*. A especificação do EARL oferece 5 valores predefinidos para declarar o modo de realização do teste, sendo eles explicados de seguida:

- *earl:automatic*: quando o teste é realizado automaticamente sem intervenção humana, através de uma ferramenta de avaliação;
- *earl>manual*: quando o teste é realizado apenas através de intervenção humana;
- *earl:semiAuto*: quando o teste é realizado parcialmente por ferramentas automáticas, mas é ainda necessário opinião humana para decidir o resultado do teste;
- *earl:undisclosed*: quando é decidido não partilhar o modo de realização do teste;
- *earl:unknown*: quando o processo de testagem é desconhecido ou indeterminado.

2.3.2.6 TestResult

O *test result* é a estrutura que detalha o resultado final de um teste realizado por um *test subject*, considerando um ou mais *test criterion*. O EARL foi implementado de modo a que cada resultado de um teste seja relacionado com uma única *assertion*, mesmo se foram executados vários sub-testes. Poderão existir regras ACT que necessitem de realizar vários sub-testes para serem validadas e, neste caso, os resultados dos sub-testes são conjugados para obter um veredicto final da regra.

2.3.2.7 OutcomeValue

O valor ou expressão *outcome* descreve a condição resultante da execução do teste. Estas condições integram-se na estrutura anterior, de modo a especificar qual foi o resultado de um sub-teste e/ou teste. Na especificação do EARL, é recomendado a utilização de

2.3 Projeto WAI-Tools

5 valores predefinidos para categorizar as condições resultantes, que serão explicados de seguida:

- *earl:passed*: quando o sujeito passa o teste;
- *earl:failed*: quando o sujeito falha o teste;
- *earl:cantTell*: quando não é claro se o sujeito passa ou falha o teste;
- *earl:inapplicable*: quando o teste não é aplicável ao sujeito;
- *earl:untested*: quando não foi realizado o teste.

Capítulo 3

Trabalho relacionado

A concepção legislativa impõe a necessidade de depositar atenção neste campo por parte das entidades responsáveis por conteúdo digital, resultando numa maior quantidade de informação a monitorizar por parte, principalmente, das entidades públicas responsáveis.

Como tal, para perceber como desenhar uma plataforma de monitorização de conformidade de acessibilidade digital, foi necessário estudar plataformas existentes de monitorização de dados em grande escala e, focando na acessibilidade, plataformas de monitorização de acessibilidade digital. Relacionando as necessidades e funcionalidades de ambos os tipos de plataforma, o desenvolvimento do PLACM tornou-se mais esmerado.

3.1 Monitorização de dados em grande escala

A monitorização, em qualquer campo ou serviço, é um processo importante para promover o cumprimento de normas estabelecidas e ainda a realização de avaliação e estudo dentro do campo, de modo a possibilitar um aperfeiçoamento no desempenho e eficiência do processo. Torna-se ainda mais importante quando os dados do campo a monitorizar são em grande quantidade, dado que a leitura e interpretação desses dados torna-se impossível se não forem feitos esforços para os agrupar ou categorizar. Como esta preocupação já é reconhecida em muitos campos, desde o campo ecológico e ambiental ao campo de saúde, inúmeras plataformas de monitorização de dados têm vindo a ser desenvolvidas. Algumas destas plataformas são privadas pois, por servirem um mercado nicho, cobram a sua utilização. Foram então estudadas algumas ferramentas públicas de monitorização de dados em grande escala, para perceber quais as funcionalidades determinantes e como é disposta a informação relevante.

3. TRABALHO RELACIONADO



Figura 3.1: Informação global do AtlantOS e gráficos relacionados com tipos de plataformas de observação

3.1.1 AtlantOS Dashboard

O AtlantOS¹ foi um projeto incluído no programa EU Horizon 2020, com mais de 60 parceiros de 18 países, com o objetivo de associar várias atividades de observação oceânica no Oceano Atlântico, promovendo a importância da oceanografia operacional e a investigação e colaboração internacional neste campo. Um dos resultados deste projeto foi a criação de um *dashboard* onde é monitorizada toda a informação registada oriunda de plataformas de observação oceânica.

A primeira informação a ser mostrada são contadores de plataformas de observação ativas e conectadas, juntamente com o número de parâmetros estudados e o número de pedidos de *download* dos dados, registados num ficheiro *.csv*, como pode ser visto na figura 3.1. Esta informação é genérica mas contextualiza o resto da informação apresentada.

O resto do *dashboard* é dividido em gráficos de linhas, de barras e de mapa. O gráfico de linhas existente apresenta a evolução temporal da quantidade de plataformas de observação existentes (3º gráfico da figura 3.1). Os gráficos de barras são usados quando o objetivo dos mesmos é comparar as plataformas de observação por tipo ou por parâmetro que monitorizam (1º e 2º gráfico da figura 3.1). São também utilizados para comparar as plataformas de observação ativas e inativas existentes e para apresentar o número mensal de *downloads* dos dados, apresentados nas figuras 3.2 e 3.3, respetivamente. Apesar deste último gráfico ser uma evolução temporal, tal como acontece no de linhas, é possível que tenha sido implementado o gráfico de barras por ser tratada uma única variável, ao contrário do que acontece no anterior, facilitando a leitura da informação pretendida. O gráfico de mapa apresenta, através de um esquema de cores, o número de *downloads* dos dados por país por mês. A informação relativa ao mês é alterada através de uma *scrollbar* horizontal.

É ainda apresentada uma tabela que apresenta o número de *downloads* mensais por país, desde o mês de janeiro de 2017, facilitando assim a leitura destes dados através de um leitor de ecrã.

¹<http://www.emodnet-physics.eu/atlantos/dashboard/>

3.1 Monitorização de dados em grande escala

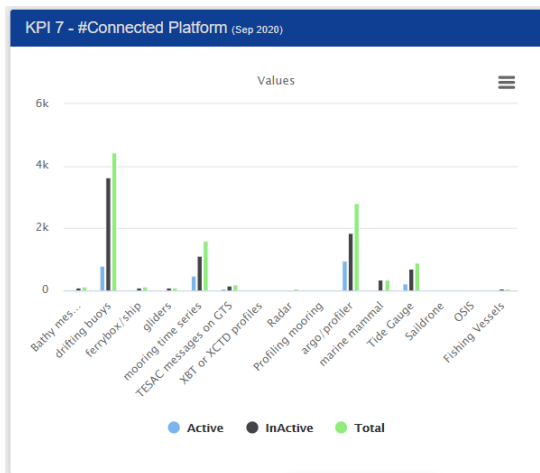


Figura 3.2: Estados de atividade das plataformas, agrupados por tipo de plataforma

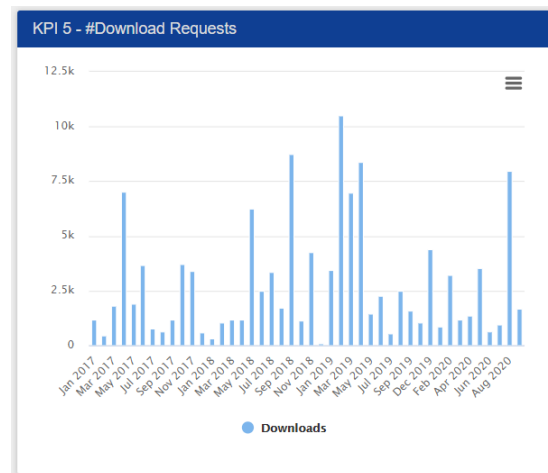


Figura 3.3: Evolução temporal dos pedidos de *downloads* dos dados

3.1.2 SDG 6 Data Portal

Em 2015, a Assembleia Geral das Nações Unidas (UN) estabeleceu 17 objetivos de desenvolvimento sustentável (Sustainable Development Goals), para estimular o seu desenvolvimento num espaço de 15 anos, sendo que o 6º (SDG 6) é focado na disponibilidade de água limpa e saneamento para todos. Como resultado desta iniciativa, foi criado um portal² que aglomera informação mundial, de modo a permitir monitorizar o cumprimento deste objetivo com rumo a 2030.

A informação apresentada em primeiro lugar é um conjunto de percentagens, de modo a apresentar as atuais circunstâncias globais das disciplinas de água potável, saneamento, qualidade da água, cooperação, etc., pois cada uma destas se relaciona com um indicador do SDG 6. As visualizações são baseadas por indicadores, logo, ao seleccionar um destes, o utilizador é redireccionado para uma *dashboard* de gráficos de linhas, barras, barras empilhadas e radar relativos a esse indicador do SDG 6.

Os gráficos de linhas tendem a apresentar informação temporal relativa a cada país ou região, juntamente com uma linha adicional a representar a média (figura 3.4). Os gráficos de barras tendem a apresentar dados relativos a países ou regiões de forma a facilitar a comparação entre estes. Os de barras empilhadas mostram a soma de valores de várias variáveis por país ou região, tal como a extensão espacial dos diferentes corpos de água por país, ilustrado na figura 3.6. O único gráfico de radar apresenta um sistema de pontuação de 0 a 100 em 4 campos diferentes relacionados com a implementação de gestão de recursos de água.

Existem ainda duas maneiras de agrupar os dados: por país ou por região, dos quais o último é a predefinição. Nas visualizações por país, a informação é fácil de reter e organizar porque é em menor quantidade. Contudo nas visualizações por região, a grande quantidade de dados aglomera-se e, por vezes, torna a sua leitura impossível, tal como

²<https://www.sdg6data.org/>

3. TRABALHO RELACIONADO

Progress over time for Indicator 6.1.1 Proportion of population using safely managed drinking water service (%) ⓘ

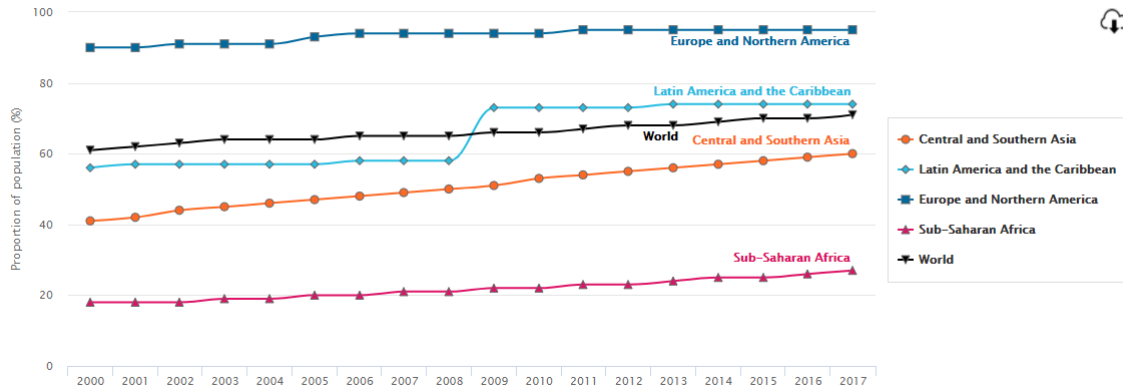


Figura 3.4: Gráfico de linhas que representa a evolução temporal do indicador 6.1.1 por região

6.6.1 Water quantity in rivers, open water bodies and aquifers in Europe and Northern America, from national data (2016) ⓘ

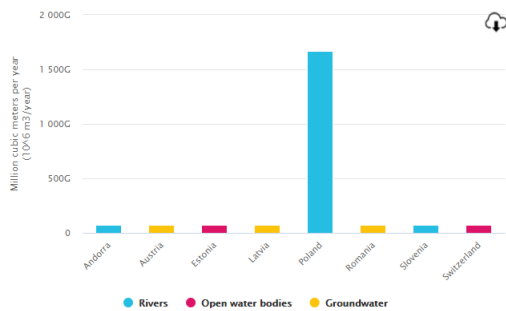


Figura 3.5: Gráfico de colunas que representa a quantidade de água em rios na região de Europa e América do Norte, a partir de dados nacionais

6.6.1 Spatial extent of wetlands, rivers and open water bodies in Europe and Northern America, from national data (2016) ⓘ

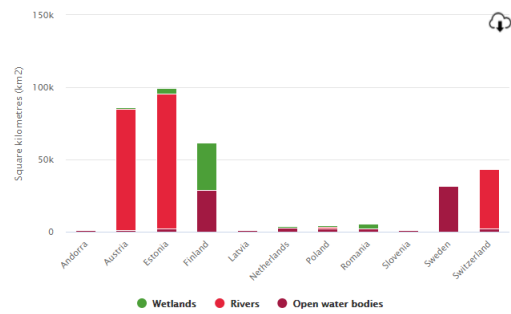


Figura 3.6: Gráfico de colunas empilhadas que representa a extensão espacial dos diferentes corpos de água na região de Europa e América do Norte, a partir de dados nacionais

3.1 Monitorização de dados em grande escala

6.5.1 Degree of integrated water resources management implementation (0-100) in Europe And Northern America, score by dimension (2017)

In the below chart, the regional value is displayed in accent colour. The values of the following countries (or areas) in the region are displayed in grey: Albania, Andorra, Austria, Belgium, Bosnia and Herzegovina, Bulgaria, Belarus, Croatia, Czechia, Denmark, Estonia, Finland, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Liechtenstein, Lithuania, Luxembourg, Malta, Monaco, Republic of Moldova, Montenegro, Netherlands, Norway, Poland, Portugal, Romania, Russian Federation, San Marino, Serbia, Slovakia, Slovenia, Spain, Sweden, Switzerland, Ukraine, North Macedonia, United Kingdom of Great Britain and Northern Ireland, Europe and Northern America

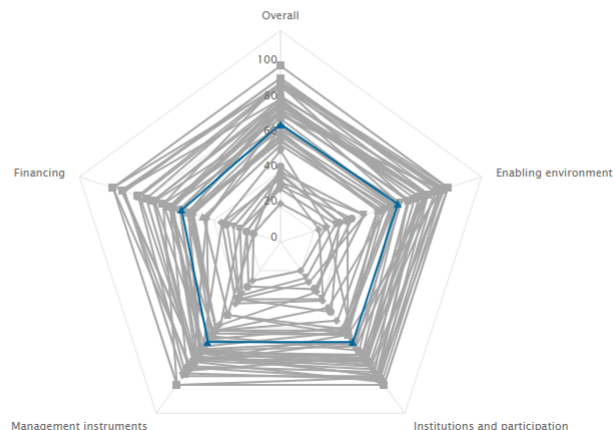


Figura 3.7: Gráfico de radar que apresenta informação do indicador 6.5.1 relativo à região de Europa e América do Norte, onde cada linha simboliza um país dessa região

acontece no gráfico de radar, exemplificado na figura 3.7, ou em alguns de linhas onde está apresentada informação relativa a todos os países da região a monitorizar. Por outro lado, em alguns gráficos de barras, não são apresentados dados relativos a todos os países, limitando informação sem qualquer critério definido. Para terminar, apenas são representadas tabelas nas visualizações por região, apresentando informação detalhada relativa ao indicador a estudar por país.

Contudo, ainda no portal, é dada a oportunidade ao utilizador de desenhar os seus mapas, gráficos e tabelas, escolhendo os parâmetros que deseja monitorizar. Desta maneira, a informação não fica restrita às visualizações desenvolvidas e, conseqüentemente, é facilitado o desenvolvimento de estudos neste campo.

3.1.3 GEMStat water quality data

O Global Environment Monitoring System for Freshwater (GEMS/Water) foi um programa criado em 1978 sob os auspícios das Nações Unidas, juntamente com outras entidades mundiais tal como a WHO e a UNESCO. O objetivo deste programa é de compilar dados mundiais relativos à qualidade de água doce, de modo a facilitar estudos neste campo e encorajar especialmente os países em desenvolvimento a elaborar programas de monitorização da qualidade da água. Para aglomerar os dados, foi então criada a base de dados GEMStat e, conseqüentemente, o portal de visualização³ destes dados. Todos os dados guardados nesta base de dados têm como origem as redes de monitorização nacionais ou de entidades responsáveis, e são disponibilizados voluntariamente por estes.

³<http://portal.gemstat.org/>

3. TRABALHO RELACIONADO

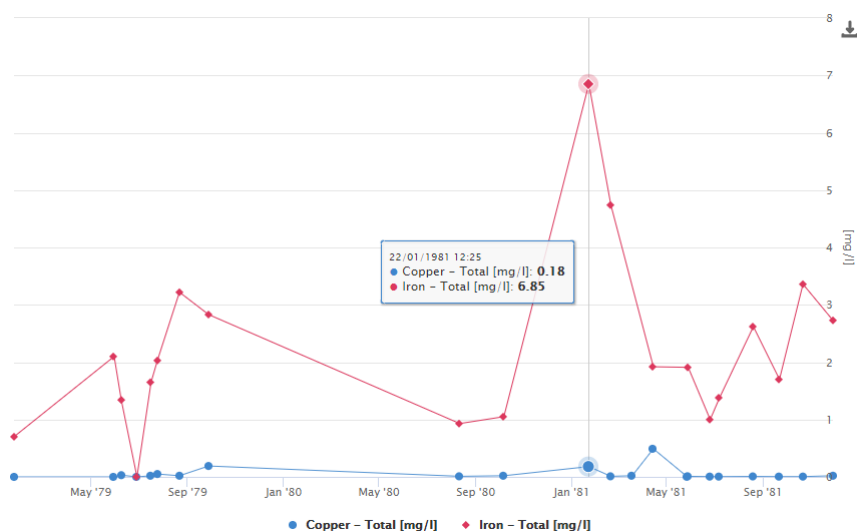


Figura 3.8: Gráfico de linhas que representa a evolução temporal dos valores de cobre e ferro registrados no rio Guandu

Atualmente, existem mais de 7 milhões de registos, permitindo a monitorização de cerca de 300 parâmetros.

Ao entrar no portal, a primeira visualização é de um mapa-mundo onde estão representadas, como predefinição, as estações de monitorização existentes. Os dados apresentados no mapa podem ser alterados entre estações, país ou bacia hidrográfica, sendo possível alterar esta seleção no canto superior direito do mapa-mundo. É ainda possível filtrar os dados apresentados através do preenchimento de um formulário lateral ao mapa. É ainda permitido substituir a visualização de dados no mapa-mundo por uma tabela, que pode alcançar os milhares de linhas.

Ao seleccionar um dos pontos no mapa ou uma linha na tabela, são listados de maneira tabelar os parâmetros monitorizados nesse ponto (quer seja estação, país ou bacia hidrográfica). Para estudar temporalmente os valores dos parâmetros, é necessário o utilizador seleccionar qual ou quais deseja monitorizar, para possibilitar a visualização desses dados em forma de tabela ou gráfico de pontos (que gera linhas quando o cursor passa pelos mesmos) (figura 3.8) ou de dispersão (figura 3.9).

De todas as ferramentas estudadas, esta é a que tem a maior quantidade de registos a monitorizar e, de modo a não dificultar a leitura destes dados, verifica-se que a maior parte das visualizações baseia-se em tabelas. Desta maneira, é apenas mostrada a informação desejada e impede a má interpretação dos dados.

3.1.4 Air Quality Monitoring Data - ACT Government

O Território da Capital Australiana (ACT) desenvolveu um portal de dados abertos⁴, de maneira a partilhar dados de uma forma renovável e acessível a todos os seus ha-

⁴<https://www.data.act.gov.au/>

3.2 Monitorização de acessibilidade

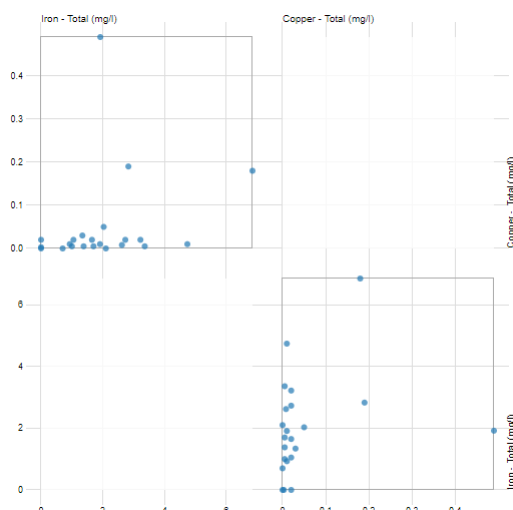


Figura 3.9: Gráfico de dispersão onde estão representados os valores dos registos de cobre e ferro do rio Guandu

bitantes. São apresentados dados relativos desde a localização de acidentes de ciclistas a localizações de semáforos. No entanto, alguns destes dados são usados para uma monitorização mais assídua, tal como a qualidade de ar⁵.

A partir de três estações de monitorização diferentes, são registados valores da qualidade de ar desses locais de hora a hora. Esta angariação de valores começou em maio de 2013, logo, para além de estudar 18 parâmetros diferentes, já estão guardados mais de 200.000 registos.

Ao contrário de todas as outras plataformas estudadas, esta não apresenta visualizações gráficas predefinidas. A única visualização predefinida é a de uma tabela que apresenta todos os valores registados, representada na figura 3.10, sendo possível fazer o seu *download*. Para visualizar qualquer tipo de dados em forma de gráfico, é necessário o utilizador configurar a sua própria visualização. Como opções de gráficos, o utilizador pode escolher entre o de barras, o de colunas, o circular, o de linhas e o de dispersão, sendo ainda permitido combinar o de colunas e de linhas. É possível configurar as variáveis a analisar e toda a apresentação do gráfico, desde o título às cores. Porém, é necessário criar uma conta na plataforma para guardar a visualização criada.

3.2 Monitorização de acessibilidade

As plataformas existentes diretamente relacionadas com a monitorização de acessibilidade digital são os observatórios. Os observatórios de acessibilidade digital são ferramentas de monitorização cujo objetivo é permitir e/ou realizar estudos neste campo. As entidades responsáveis por estas plataformas podem pertencer ao sector público ou pri-

⁵<https://www.data.act.gov.au/Environment/Air-Quality-Monitoring-Data/94a5-zqnn>

3. TRABALHO RELACIONADO

Name	GPS	Date	NO2	O3_1hr	O3_4hr	CO	PM10	PM2.5	AQI_...	AQI_...	AQI_...	AQI
Florey	(-35.2206...	23/11/20...	0	0.027	0.028	0.22	12.65	7.67	2	0	27	
Civic	(-35.2853...	23/11/20...		0.024	0.025	0	11.45	6.5	0	0	24	
Monash	(-35.4183...	23/11/20...	0	0.026	0.026	0.04	11.5	6.32	0	0	26	
Civic	(-35.2853...	23/11/20...		0.024	0.026	0	11.72	6.33	0	0	24	
Monash	(-35.4183...	23/11/20...	0	0.026	0.024	0.03	11.88	6.52	0	0	26	
Florey	(-35.2206...	23/11/20...	0	0.027	0.026	0.22	13.15	7.77	2	0	27	
Florey	(-35.2206...	23/11/20...	0	0.03	0.026	0.23	13.35	7.7	2	0	30	
Monash	(-35.4183...	23/11/20...	0	0.028	0.023	0.03	12.2	6.52	0	0	28	
Civic	(-35.2853...	23/11/20...		0.026	0.026	0	11.86	5.96	0	0	26	
Monash	(-35.4183...	23/11/20...	0	0.022	0.023	0.03	12.45	6.44	0	0	22	
Civic	(-35.2853...	23/11/20...		0.028	0.025	0	12.33	5.88	0	0	28	
Florey	(-35.2206...	23/11/20...	0	0.027	0.025	0.23	13.4	7.68	2	0	27	
Monash	(-35.4183...	22/11/20...	0	0.02	0.025	0.03	12.55	6.72	0	0	20	
Florey	(-35.2206...	22/11/20...	0	0.022	0.025	0.23	13.87	7.84	2	0	22	

Figura 3.10: Amostra de registos das medições de qualidade de ar das 3 estações de monitorização, representados numa tabela

vado, bem como os resultados dos estudos daí originados poderem ser partilhados publicamente ou não. Estas entidades podem ajudar a evoluir a acessibilidade digital no seu todo, ao permitir que os desenvolvedores das plataformas estudadas tenham acesso, pelo menos, aos resultados dos respetivos estudos, oferecendo assim um material de apoio aos desenvolvedores que não se encontram informados ou preocupados. Não existe qualquer padrão para a metodologia dos estudos de acessibilidade digital e, como tal, cada observatório tende a realizar os estudos, e a partilha dos respetivos resultados, da sua maneira particular.

A comunidade europeia impôs, através da diretiva 2016/2102, que os recursos digitais dos organismos do sector público devem ser acessíveis, acrescentando que tem de ser feita uma monitorização periódica da conformidade dos recursos digitais com os requisitos de acessibilidade, por parte dos respetivos Estados-Membros. Tendo em conta a imprescindibilidade da diretiva 2016/2102 para este projeto, os observatórios estudados pertencem aos Estados-Membros da União Europeia, sendo, de seguida, explicado o seu funcionamento.

3.2.1 Observatório português

Este observatório é um *website*⁶ disponível publicamente, de responsabilidade da Agência para a Modernização Administrativa, um dos parceiros do projeto WAI-Tools. O objetivo deste observatório é partilhar publicamente resultados de avaliações a entidades públicas.

⁶<https://observatorio.acessibilidade.gov.pt/>

3.2 Monitorização de acessibilidade

Com a centralização desta informação, a AMA, responsável pela recolha e introdução de dados, ou outras entidades interessadas podem realizar estudos focados na acessibilidade das aplicações/*websites* portuguesas. As avaliações são feitas através da utilização da ferramenta de avaliação automática QualWeb, desenvolvido pelo LASIGE, avaliando um conjunto de 13 regras ACT, 26 técnicas WCAG e 15 boas práticas.

A informação está organizada em diretórios, de modo a agrupar aplicações/*websites* testados. Na listagem de diretórios é apresentada informação bastante geral, tal como o número de páginas testadas e número de entidades registadas. A apresentação da informação relativa à aplicação/*website* baseia-se numa listagem das boas e más práticas encontradas nas suas páginas, complementada com um gráfico de barras, que representa o número de páginas por intervalo de pontuação, e um gráfico de radar, que representa as pontuações de todas as páginas testadas. Não são especificadas quais as páginas ou os elementos onde as boas e más práticas foram encontradas, mas apenas o número de páginas e elementos onde ocorreram.

Este observatório oferece ainda um sistema de pontuação baseado nos resultados dos testes, onde cada teste tem um peso predefinido para a pontuação. As pontuações são diretamente relacionadas com as aplicações, sendo que as pontuações dos diretórios são as médias das aplicações correspondentes.

3.2.2 Observatório norueguês

Este observatório encontra-se também disponível publicamente⁷, apesar de ter a sua vertente reservada. É de responsabilidade da Agência de Digitalização Norueguesa (Digitaliseringsdirektoratet), também parceiro do projeto WAI-Tools. Neste observatório, são publicados estudos em relação às barreiras digitais nos *websites* noruegueses, juntamente com informações e indicações de como podem ser desenvolvidas aplicações e *websites* considerando o design universal.

A sua metodologia de testagem baseia-se em 35 critérios dos 61 das WCAG 2.0, apenas correspondentes aos níveis A e AA. Foram criados procedimentos de teste para cada um destes critérios, de modo a serem testados manual, semiautomático ou automaticamente por profissionais da Agência e/ou por uma ferramenta de avaliação automática. As páginas que são sujeitas a estes testes são escolhidas tendo em conta o foco do estudo a ser realizado e que serviços são mais usados na aplicação/*website*. Todas as auditorias são realizadas em constante comunicação com a entidade responsável pela aplicação, de modo a partilhar especificamente os resultados da mesma. Ao fim de várias iterações, é publicado um relatório que generaliza os resultados dos testes feitos e o respetivo cumprimento dos requisitos legais.

Os relatórios publicados enunciam os critérios de sucesso, e respetivos procedimentos de teste, e páginas testadas na respetiva aplicação/*website*. São descritas as anomalias

⁷<https://uu.difi.no/tilsyn>

3. TRABALHO RELACIONADO

encontradas, tabulando a descrição dos elementos (e não um selector para os mesmos) onde os critérios de sucesso não foram cumpridos. Alguns comentários são introduzidos a estas anomalias, de modo a justificá-las e a auxiliar a correção das mesmas.

3.2.3 Observatório espanhol

Este observatório, tal como os anteriores, é um *website*⁸ de acesso público, sendo uma iniciativa do Ministério de Assuntos Económicos e Transformação Digital. O objetivo deste observatório é publicar estudos periódicos relativos à acessibilidade nos *websites* do sector público, guardando a sua evolução temporal.

A sua metodologia consiste na utilização de uma ferramenta de avaliação automática, para realizar 20 verificações, sendo estas baseadas nos critérios WCAG 2.0. Cada uma destas verificações, tendo em conta o resultado da mesma, retorna um valor entre 0 e 1. De modo a esquematizar estes valores, e estimar a conformidade do *website* testado, foram criados três tipos de indicadores de acessibilidade: pontuação média de página, pontuação média do *website* e pontuação média da verificação. A conformidade é apenas estimada pois, para depreender a conformidade real do *website*, é necessária uma revisão manual dos resultados por peritos em acessibilidade.

Por fim, baseado no foco das avaliações, os resultados das mesmas podem gerar dois tipos de relatórios: agregado e individual. Os relatórios agregados organizam os resultados de maneira pouco pormenorizada em gráficos e tabelas, sendo publicados para acesso público, em formato PDF. Os relatórios individuais detalham todos os resultados da avaliação e são apenas remetidos à entidade responsável pelo *website* testado, de modo a permitir a evolução do mesmo.

3.3 Discussão

As plataformas analisadas na secção 3.1 mostram que, de modo a viabilizar a leitura de uma grande quantidade de dados, os *dashboards* gráficos são um mecanismo comum de monitorização. Estes *dashboards* apresentam vários tipos de gráficos, mas os usados com maior frequência são os de linhas, circulares, de barras e de barras empilhadas.

Os gráficos de linhas e barras são similares em termos de comparação entre dados, pois os valores que cada ponto ou barra representam são sempre proporcionais à sua altura. Contudo, os gráficos de linhas tendem a ser usados para visualizar dados contínuos de uma única categoria, tais como variações temporais, enquanto que os de barras são usados de preferência para comparar dados de vários grupos ou categorias. Os gráficos circulares e de barras empilhadas são aplicados para representar várias categorias de dados como um

⁸https://administracionelectronica.gob.es/pae_Home/en/pae_Estrategias/pae_Accesibilidad/pae_Observatorio_de_Accesibilidad.html

todo, de modo a predispor informação global e agrupar várias categorias por diferentes pontos, respetivamente.

Ambas as plataformas que apresentam uma grande quantidade de informação imediata, descritas nas secções 3.1.3 e 3.1.4, possibilitam ainda a criação de gráficos personalizados. Em conjunto com esta funcionalidade, todos os dados são tabulados, de modo a facilitar a leitura e monitorização de subconjuntos dos mesmos.

Em relação à organização dos dados a estudar, as plataformas tendem a organizar a informação por categorias/classes, tal como acontece no SDG 6 Data Portal, descrito na secção 3.1.2, e no Observatório Português, descrito na secção 3.2.1; enquanto que a primeira organiza a sua informação pelos vários indicadores da objetivo, o Observatório Português organiza os seus dados através de uma hierarquia de classes, desde o diretório até à aplicação/*website*. Deste modo, a informação é generalizada e é possibilitado o estudo por vários níveis de granularidade, especificando a informação apenas quando o utilizador deseja.

A introdução de dados nas plataformas estudadas tem tendência a ser automática, tal como acontece em 3 das 4 plataformas descritas na secção 3.1. Destas 4, apenas a SDG 6 Data Portal (3.1.2) não utiliza este tipo de introdução de dados, pois os dados apresentados são estáticos, referentes a certos censos e estudos realizados nos diferentes países. No caso dos observatórios de acessibilidade nacionais, a informação é adicionada manualmente pelos administradores do respetivo sistema, tornando-a pública quando o desejarem. Isto possivelmente acontece porque atualmente não existe uma tecnologia que permita realizar testes de acessibilidade totalmente automáticos, sem qualquer verificação humana; a acessibilidade é um campo muito abrangente, que tem de chegar a um vasto espectro de limitações e isso traduz-se na especificidade significativa dos testes a realizar.

Capítulo 4

Desenho

Depois de estudados os diferentes tipos de plataformas, foi possível relacionar as necessidades e funcionalidades comuns e interpretá-las para os requisitos da ferramenta a desenvolver. Como parceiro do projeto WAI-Tools, foi possível realizar o levantamento de requisitos em conjunto com os restantes parceiros, apesar de pouca participação dos mesmos. Neste capítulo, é descrito o processo e resultados desse levantamento de requisitos. A partir desses requisitos, foi desenhado o modelo de dados da plataforma a desenvolver e as tecnologias a serem usadas.

4.1 Levantamento de requisitos

Para ser possível desenhar uma plataforma de monitorização de dados focada nas necessidades da comunidade europeia, foi necessário perceber junto, principalmente, das entidades monitorizadoras parceiras do projeto quais os dados relevantes a estudar e apresentar: a portuguesa AMA e a norueguesa Digdir. Ambas têm como responsabilidade acompanhar a satisfação das regulamentações orientadas à acessibilidade digital, a nível nacional. Infelizmente, e apesar de todos os parceiros do projeto terem tido a oportunidade de participar neste levantamento de requisitos, a AMA foi o único parceiro que apresentou um parecer substancial.

4.1.1 Origem de dados

Foi necessário perceber, antes de mais, quais os tipos de dados que seriam monitorizados e qual a origem dos mesmos. Como a implementação desta plataforma se encontra na meta final do projeto WAI-Tools, todo o caminho e desenvolvimento feito durante o mesmo recai na necessidade desta plataforma. Tal como descrito na secção 2.3.2, um dos objetivos do projeto era a definição de um formato padronizado de resultados de testes de avaliação de acessibilidade, de modo a uniformizar estes resultados e promover a cooperação de diferentes avaliadores. Esta padronização originou assim o primeiro tipo

4. DESENHO

de documento que a plataforma recebe como *input*, o relatório EARL.

Um outro objetivo do projeto WAI-Tools era facilitar a elaboração de declarações de acessibilidade, de modo a impulsionar uma maior dedicação à acessibilidade digital, por parte das entidades responsáveis por conteúdo digital. É também legislado, na diretiva 2016/2102, que as entidades públicas dos Estados-Membros da UE necessitam de publicar periodicamente uma declaração de acessibilidade detalhada.

Uma declaração de acessibilidade, segundo a diretiva europeia e respetiva transposição para a legislação portuguesa, deve detalhar informação sobre a acessibilidade do conteúdo da aplicação/*website* a estudar, particularmente, o nível de conformidade da mesma, tendo em conta os requisitos de acessibilidade considerados, tais como as WCAG 2.1. Deve ainda evidenciar os requisitos técnicos, quais as partes do conteúdo que não são acessíveis e, se possível, oferecer alternativas para estas. Para permitir a exposição de problemas ou incumprimentos de acessibilidade encontrados, devem também ser disponibilizados contactos para as entidades responsáveis da aplicação/*website*. Neste tipo de declaração, podem ainda ser detalhados e evidenciados os esforços e medidas realizadas para alcançar o nível de conformidade atual.

De modo a simplificar a concepção destas declarações, foi assim desenvolvido um gerador de declarações de acessibilidade¹. Neste, a entidade responsável pela aplicação/*website*, através do preenchimento de um formulário claro e prático, pode apresentar aos utilizadores os esforços feitos e respetivas evidências, as limitações de acessibilidade e possíveis alternativas e os requisitos técnicos de usabilidade. Mais tarde, de estrutura similar mas orientado para a legislação portuguesa, foi também desenvolvido um gerador de declarações de acessibilidade², por parte da AMA.

As declarações de acessibilidade tornam-se assim no segundo tipo de documento que a plataforma recebe como *input*, pela informação que explicitam e porque, ao listar as evidências dos esforços, podem disponibilizar hiperligações para relatórios de resultados de testes.

A agregação de resultados padronizados de testes, oriundos de diferentes avaliadores e entidades, permite a comparação dos mesmos, registar uma evolução temporal e incentivar a uma constante melhoria, enquanto que a informação oriunda das declarações categoriza a origem dos testes e permite uma comparação destas categorias. Foram estes os pontos fulcrais para o desenho da plataforma.

4.1.2 Agrupamento de dados

Como descrito no segundo capítulo, a padronização dos resultados de testes baseou-se numa serialização em JSON-LD do vocabulário EARL, desenvolvido para expressar resultados de testes usando RDF. Paralelamente ao EARL, a padronização tem como pedra

¹<https://www.w3.org/WAI/planning/statements/generator/>

²<https://www.acessibilidade.gov.pt/gerador/>

4.1 Levantamento de requisitos

```
{
  "@type": "Assertion",
  "test": {
    "@id": "https://act-rules.github.io/rules/9eb3f6",
    "@type": "TestCase",
    "title": "Image filename is accessible name for image",
    "description": "This rule checks that image elements that use their source filename as their accessible name do so without loss of information to the user."
  },
  "mode": "earl:automatic",
  "result": {
    "@type": "TestResult",
    "outcome": "earl:passed",
    "source": [
      {
        "result": {
          "pointer": "html > body > img:nth-of-type(1)",
          "outcome": "earl:passed"
        }
      }
    ]
  },
  "description": "This element's accessible name uses the filename which accurately describes the image",
  "date": "2020-01-07 11:12:01"
}
```

Figura 4.1: Exemplo de asserção originada ao avaliar uma página com a regra “Image filename is accessible name for image”

angular as asserções. A asserção é como que uma declaração que estabelece a ligação entre quem (ou o quê) realizou o teste (Assertor), o que foi testado (*TestSubject*), o critério de teste (*TestCriterion*), o respetivo resultado (*OutcomeValue*) e o modo de realização do teste (*TestMode*), como exemplificado na figura 4.1. Contextualizando para os requisitos desta plataforma, é um veredicto da avaliação de uma regra ACT ou critério de sucesso numa página ou aplicação, usando um avaliador, que pode ser um humano ou uma ferramenta de avaliação (ou um grupo de ambos). É o aglomerado de asserções que formaliza um relatório EARL.

Como todos os dados dos testes são interligados através da asserção e esta é recolhida através dos relatórios EARL, a asserção torna-se assim na base de informação da plataforma a desenvolver. A informação a ser apresentada e monitorizada tem como base os resultados das asserções.

Mais tarde, através da necessidade da Digdir, o critério de sucesso foi adicionado como segunda base de informação. A Digdir apresenta estudos no seu observatório nacional tendo em conta critérios de sucesso, em vez de regras ACT ou técnicas WCAG e, como tal, expressaram a importância desta funcionalidade para entidades que desejem monitorizar as suas aplicações da mesma maneira. O cálculo dos resultados dos critérios será feito através dos resultados das asserções, quer estas avaliem regras quer diretamente os critérios.

4.1.3 Classes

Na discussão do capítulo anterior (secção 3.3), foi percebido que a organização da informação por classes, aumentando o nível de granularidade dos mesmos, facilita a interpretação de uma grande quantidade de dados. Como tal, esta organização foi considerada um requisito para o desenvolvimento do PLACM.

A plataforma foi assim dividida em várias classes de dados, organizando-as numa estrutura semelhante a uma hierarquia. Cada classe representa um nível de granularidade

4. DESENHO

Classe	Origem	Exemplos
Continente	Manual	África, América do Sul, Europa
País	Manual	Senegal, Argentina, Espanha
Categoria	Manual	Saúde, Cultura, Educação
Sector	Manual	Público, Privado
Entidade	DA e manual	Disney, AMA, Amazon
Aplicação/ <i>Website</i>	DA e manual	Disney Store, Portal português de acessibilidade, Twitch
Ferramenta de avaliação	RE	aXe, Alfa, Qualweb
Critério de sucesso	RE	1.2.4, 2.2.3, 4.1.1
Tipo de elemento	RE	Imagens, tabelas, elementos de vídeo
Regra	RE	Attribute is not duplicated, HTML page title is descriptive, Text has minimum contrast

Tabela 4.1: Classes estudadas no PLACM, a origem dos seus dados e exemplos dos mesmos

e, à medida que descemos nesta hierarquia, os dados apresentados são cada vez mais específicos. A tabela 4.1 apresenta quais as classes estudadas na plataforma, a origem dos dados e um conjunto de exemplos para cada classe.

Apesar do projeto WAI-Tools ser cofinanciado pela Comissão Europeia, a ambição de fazer chegar a importância da acessibilidade digital a todo o mundo mantém-se intacta. A adição das classes de ‘Continente’ e ‘País’ à hierarquia, para além de permitir uma monitorização a nível nacional, pode impulsionar uma maior competição, e consequentemente melhores resultados, entre entidades locais ou nacionais dentro e fora da Europa.

As classes de ‘Sector’ e ‘Categoria’ foram adicionadas à hierarquia porque ambos os Observatórios incluídos no projeto, o português e o norueguês, apresentam resultados de estudos tendo em conta estas classes, tornando-se um interesse para os parceiros do projeto. Apesar de pertencerem à hierarquia de classes, estas duas encontram-se no topo da mesma, paralelas ao nível da classe de ‘Continente’, para que possam ser obtidas visualizações de dados com origem nas mesmas.

Ainda que seja possível inferir o país (e respetivo continente) da aplicação testada através do domínio de topo no URL da declaração ou relatório, nem sempre é possível obter esta informação automaticamente, tal como acontece, por exemplo, quando o domínio é *.com* ou em aplicações móveis; as aplicações não têm qualquer legislação imposta em relação à localização da sua declaração e, como tal, pode não ser proporcionado um domínio de topo de código de país. Para ultrapassar esta dificuldade e introduzir pragmaticamente estes documentos na plataforma, é desejada a implementação de interpretação destes documentos em ficheiro. A informação relativa ao sector e categoria a que pertence a aplicação testada também não pode ser obtida automaticamente, logo estas 4 classes referidas são bastante dependentes da introdução manual. Contudo, foi considerado que

4.1 Levantamento de requisitos

este esforço adicional é justificado pela importância da monitorização destes dados.

Foi introduzida a classe de ‘Tipo de Elemento’ na hierarquia e não a de ‘Elemento’ porque existe uma grande quantidade de elementos diferentes avaliados nas regras e, agrupando-os, a leitura dos resultados das avaliações torna-se mais discriminável. O processo de agrupamento foi feito tendo em consideração os elementos avaliados nas regras ACT, de modo a relacionar atributos correspondentes a certos elementos ou elementos similares. Por exemplo, o tipo de elemento *Table* agrupa os elementos avaliados nas regras [d0f69e](https://act-rules.github.io/rules/d0f69e)³ e [a25f45](https://act-rules.github.io/rules/a25f45)⁴, que correspondem a células de cabeçalho e a atributos ‘headers’, respetivamente, e nunca ao elemento <table> diretamente.

Existem ainda algumas regras que realizam verificações aos atributos dos elementos, logo terão de ser avaliados todos os elementos para detetar a utilização desses atributos. Nestes casos, o nome do tipo de elemento corresponde ao conjunto de atributos avaliados ou ao tipo de elemento que utiliza esses atributos, tal como acontece nos tipos *ARIA attributes* e *Form fields*, respetivamente. Com este agrupamento, os dados relativos ao elemento são generalizados e agregados, tornando o estudo dos mesmos menos específico mas mais claro. As tabelas no anexo A relacionam as regras e os elementos e atributos avaliados com o tipo de elemento inferido.

Em conjunto com a intenção de algum tipo de competição entre entidades, a classe de ‘Ferramenta de avaliação’ foi introduzida para que se crie também alguma competição entre os desenvolvedores das mesmas. Não se ambiciona que uma ferramenta obtenha melhores resultados que outras, mas sim que haja coerência entre todas as ferramentas usadas para avaliação de acessibilidade digital. Esta classe permite ainda monitorizar e estudar se existe alguma monopolização no mercado de ferramentas disponíveis.

A informação relativa à ‘Entidade’ e à ‘Aplicação/*website*’ é obtida diretamente da declaração de acessibilidade e, como tal, são de obrigatória presença na hierarquia, tal como os critérios de sucesso e as regras que são a base de todas as avaliações de acessibilidade.

Não foi incluída a classe de ‘Página’, no fundo da hierarquia, porque, para além do seu nível de granularidade ser muito pequeno para uma plataforma de dados em grande-escala, não foi manifestada uma necessidade por parte dos parceiros do projeto de monitorizar a quantidade de resultados das asserções ou de critérios de sucesso a este nível.

4.1.4 Tipos de gráficos

Descritos na discussão do capítulo anterior, na secção 3.3, os tipos de gráficos mais vulgarmente utilizados nas plataformas estudadas são os de linhas, circulares, de barras e de barras empilhadas, sendo agora necessária uma relação entre estes tipos de gráficos com os requisitos do PLACM.

³<https://act-rules.github.io/rules/d0f69e>

⁴<https://act-rules.github.io/rules/a25f45>

4. DESENHO

Os gráficos de linha são apontados para, principalmente, visualizar dados contínuos, como acontece nas variações temporais, enquanto que os de barras são usados para comparar dados; com os dados baseados no número de asserções e critérios de sucesso por resultado, como descrito na secção 4.1.2, a comparação torna-se um ponto importante na plataforma e, conseqüentemente, o uso de gráficos de barras também.

Os gráficos circulares e de barras empilhadas são aplicados para representar várias categorias como um todo, logo apenas poderiam ser usados numa seleção de visualizações da plataforma, que será explicada na próxima secção. Contudo, os gráficos circulares foram excluídos por ser complexo comparar área de círculos, dando origem a más interpretações.

Concluindo, os gráficos de barras (simples ou empilhadas) foram percebidos como os melhores tipos a apresentar nesta plataforma e foram implementados em todas as visualizações de dados. Os gráficos de linha poderiam ter sido usados na visualização temporal dos dados mas foi decidido manter os gráficos de barras; é necessário monitorizar várias categorias de dados mas foi considerado mais importante comparar a informação combinada entre pontos temporais do que entre estas categorias.

4.1.5 Formas de visualização e navegação

Tendo em conta as bases de informação definidas e a inevitabilidade desta plataforma ser baseada em gráficos, foi pensado um conjunto de métricas, de modo a que as mesmas ocupem o eixo das abcissas dos gráficos e os seus valores o eixo das ordenadas. Como ambos os dados-base, as asserções e os critérios de sucesso, podem ter os mesmos 5 resultados, as 10 principais métricas são o número de asserções e critérios de sucesso com resultados *passed*, *failed*, *cantTell*, *innaplicable* e *untested*. Para diferenciar os dados-base, será necessário haver vistas diferentes para cada um.

Como o número de critérios de sucesso é constante, a soma dos resultados destes dados-base resultará nesse mesmo número. Deste modo, representando as várias métricas como um todo, as vistas com as métricas relativas aos critérios de sucesso podem implementar os gráficos de barras empilhadas.

Uma décima primeira métrica, o número de páginas testadas, foi também adicionada. Ao monitorizar aplicações apenas através das asserções geradas nos seus testes, não é possível deduzir o tamanho das mesmas, porque a realização de testes a muitas páginas da aplicação baseados em poucas regras (ou critérios) pode gerar uma quantidade similar de asserções se os testes foram feitos a poucas páginas mas baseados em muitas regras. Para responder a esta dificuldade, o número de páginas testadas torna-se numa métrica relevante para os dados a monitorizar na plataforma.

Com a categorização dos dados através das classes, é esperado que o utilizador possa monitorizar informação a níveis acima ou abaixo da aplicação. Para cumprir com esta funcionalidade, será necessário possibilitar uma navegação pela hierarquia. A navegação pela hierarquia deve proporcionar a oportunidade de ver os dados por classe sem qualquer

filtro, tal como os dados de todos os países ou regras testadas, mas, sendo o PLACM uma plataforma de monitorização, é bastante relevante se for possível filtrar os dados à medida que o utilizador vá mudando de nível, de modo a especificar o que quer estudar. Tomando como exemplo a classe de País, esta plataforma deve possibilitar navegar até aos dados das entidades portuguesas (filtrando por Portugal) ou aos das regras avaliadas em França (filtrando por França). Da mesma maneira, tomando como exemplo a classe de Tipo de elemento, a plataforma deve permitir monitorizar os dados dos continentes que avaliaram o tipo *Table* ou os das entidades que avaliaram o tipo *Button*.

Ao descer na hierarquia, os dados filtrados do nível inferior serão sempre um subconjunto do nível superior, logo os dados apresentados serão mais específicos. Inversamente, ao subir nas classes da hierarquia, os dados ficam mais genéricos. Para facilitar este segundo tipo de navegação, a plataforma deve oferecer a possibilidade de seleção dos dados do nível superior a apresentar, de modo a reduzir a apresentação de uma quantidade indesejada de informação ao utilizador. Com esta seleção, e mantendo o último exemplo, o utilizador pode escolher estudar os dados de, no mínimo, uma ou de todas as entidades que avaliaram o tipo *Button*.

A plataforma deve ainda fornecer uma maneira de comparar diretamente dados de uma mesma classe, tal como comparar o tipo de elemento *Table* com o *Form fields* e o *Button*, ou o sector privado com o público. Facilitando a comparação de dados, a monitorização torna-se mais instantânea e simples para qualquer utilizador, para além de facilitar possíveis estudos em relação à acessibilidade digital, impulsionando uma maior discussão e competição neste campo.

A última forma de navegação pensada é a da linha temporal dos dados. Como as aplicações web estão em constante crescimento e melhoramento, deve ser possível verificar temporalmente as melhorias (ou não) de todas as classes registadas na plataforma. Com esta linha temporal, é possível deduzir através dos dados, por exemplo, se um país tem legislada a acessibilidade digital (se apresentar bastantes resultados positivos) ou se uma promissora ferramenta de avaliação tem vindo a ser usada (tendo em conta o número de páginas avaliadas). Seria importante manter a possibilidade de filtragem, derivada da navegação para “baixo” na hierarquia mencionada anteriormente, de modo a permitir relacionar temporalmente dados específicos.

Como existem múltiplas formas de navegação, para uma grande quantidade de dados a monitorizar, um dos requisitos desta plataforma é a fácil partilha de visualização. Cada vista e cada escolha do utilizador deve ser representada no URL, para que este seja partilhável e corresponda sempre à vista desejada.

4.2 Modelo de dados

Nas secções anteriores, foi analisada a necessidade de fundamentar a grande quantidade de informação a apresentar em dois tipos de dados-base, as asserções e os critérios de

4. DESENHO

sucesso. Toda a informação a guardar é derivada de avaliações de acessibilidade ou informações relativas a aplicações, mais concretamente, relatórios EARL e declarações de acessibilidade. Sendo assim, o modelo de dados desta plataforma baseia-se nestes documentos (e nos seus geradores disponíveis), nos dados-base daí inferidos e nas classes a monitorizar. O desenho do modelo de dados está representado na figura 4.2. As 13 tabelas originadas, e respetivas colunas, serão explicadas de seguida.

4.2.1 Tabela ‘Rule’

Esta tabela surgiu da necessidade de guardar toda a informação relacionada com as regras ACT, o fundamento base para as avaliações de acessibilidade. Apesar do desenvolvimento das regras ACT ser um esforço contínuo, as regras consistem em informação estática; assim sendo, os dados relativos às regras são introduzidas nesta tabela que, por sua vez, é inicializada com a instalação do PLACM.

Esta tabela tem relação com outras três tabelas, sendo duas relações de ‘muitos para muitos’ para as tabelas ‘ElementType’, porque vários tipos de elementos podem ser avaliados numa regra e várias regras podem avaliar um tipo de elemento, e ‘SuccessCriteria’, sendo que vários critérios podem ser mapeados por uma regra e várias regras podem mapear para um critério de sucesso. A terceira relação é de ‘um para muitos’ para a tabela ‘Assertion’, pois cada asserção é o resultado da avaliação de uma regra.

As colunas desta tabela são descritas em detalhe de seguida:

- **RuleId**: Identificador único da regra para o PLACM;
- **Mapping**: Identificador único documentado da regra;
- **Name**: Nome da regra;
- **Url**: URL da documentação da regra;
- **Description**: Explicação breve, em linguagem clara, do que a regra faz

4.2.2 Tabela ‘ElementType’

Tal como explicado na organização da hierarquia, os tipos de elemento discriminam os resultados das regras por um nível de granularidade um pouco maior. A única informação pertinente de armazenar é o nome do tipo de elemento, pois, o processo de agrupamento dos diferentes elementos avaliados (e respetiva nomeação dos grupos) foi feito através de critérios informais, explicados na secção 4.1.3.

Esta tabela tem uma única relação, sendo uma de ‘muitos para muitos’ com a tabela ‘Rule’ já descrita acima. Como esta é a única relação, esta tabela também é inicializada com a instalação da base de dados e, caso sejam adicionadas novas regras, terão de ser adicionadas manualmente as relações com os tipos de elemento avaliados.

4.2 Modelo de dados

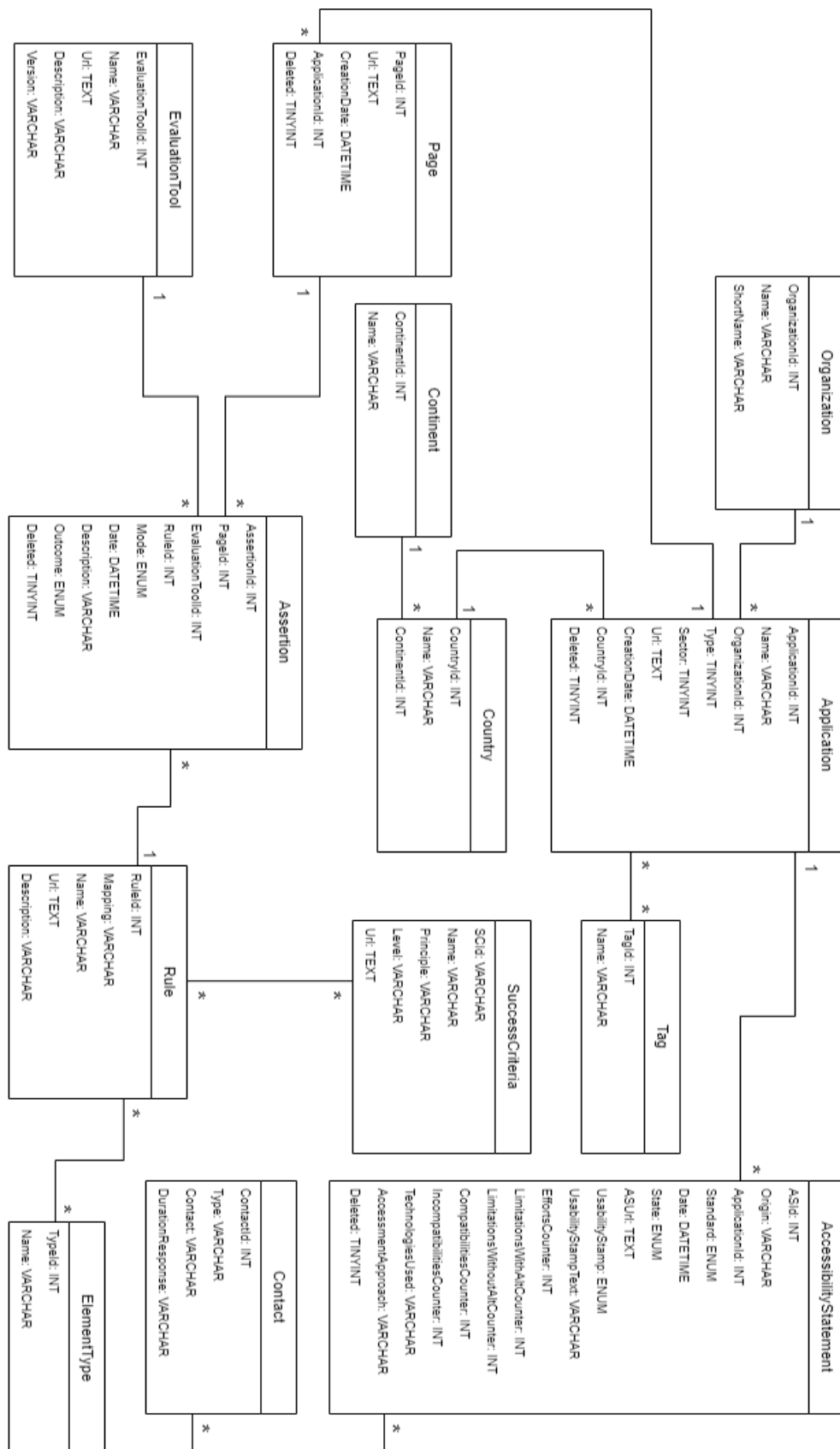


Figura 4.2: Modelo de dados do PLACM

4. DESENHO

As colunas desta tabela são descritas em detalhe de seguida:

- **TypeId**: Identificador único do tipo de elemento;
- **Name**: Nome do tipo de elemento.

4.2.3 Tabela ‘SuccessCriteria’

Esta tabela foi criada para armazenar a informação detalhada dos critérios de sucesso, permitindo a sua comparação. Como a lista de critérios de sucesso é padronizada nas diretrizes WCAG 2.1, esta informação é igualmente introduzida na instalação da base de dados e, como tal, terá de haver manutenção desta tabela quando as diretrizes WCAG forem atualizadas.

Esta tabela está relacionada com a tabela ‘Rule’, com uma relação de ‘muitos para muitos’, como já foi explicado acima.

As colunas desta tabela são descritas em detalhe de seguida:

- **SCId**: Identificador único do critério de sucesso;
- **Name**: Nome do critério de sucesso;
- **Principle**: Princípio do critério de sucesso;
- **Level**: Nível de conformidade do critério de sucesso;
- **Url**: URL da documentação do critério de sucesso.

4.2.4 Tabela ‘EvaluationTool’

A tabela ‘EvaluationTool’, guardando todos os dados relativos aos avaliadores, tem como objetivo permitir a comparação de resultados de avaliações tendo em conta quem, ou o quê, realizou a avaliação. Esta tabela não guarda apenas os dados relativos a ferramentas de avaliação automáticas, mas também todos os avaliadores manuais ou semiautomáticos, armazenando todos os ‘Assertor’ dos relatórios de avaliação. A última coluna desta tabela é utilizada para guardar informação exclusivamente relativa a ferramentas de avaliação, neste caso, a sua versão.

Esta tabela contém uma única relação com outra tabela, ‘Assertion’, de ‘um para muitos’, pois um avaliador pode originar várias asserções, mas uma asserção apenas pode ser criada por um avaliador. Esta relação possibilita um histórico de avaliações por parte do avaliador. As colunas que guardam o URL, descrição e versão são unicamente necessárias para ferramentas de avaliação automáticas ou semiautomáticas, pois nenhum destes campos é válido para caracterizar uma pessoa.

As colunas desta tabela são descritas em detalhe de seguida:

- **EvaluationToolId:** Identificador único do avaliador;
- **Name:** Nome do avaliador;
- **Url:** URL respetivo à ferramenta de avaliação ou da página informativa de uma pessoa;
- **Description:** Descrição da ferramenta de avaliação ou informação adicional de uma pessoa, tal como a sua metodologia;
- **Version:** Versão da ferramenta de avaliação.

4.2.5 Tabela ‘Tag’

Esta tabela foi originada do interesse em comparar resultados de avaliações por categoria. É uma tabela simples, guardando apenas o nome da categoria, por não existir ainda a necessidade de um agrupamento mais detalhado. A informação nesta tabela é preenchida pelo administrador que, ao adicionar os diferentes tipos de documentos, pode informar a que categoria pertence a aplicação avaliada.

Esta tabela relaciona-se à tabela ‘Application’ com uma relação ‘muitos para muitos’, porque a mesma categoria pode identificar várias aplicações do mesmo modo que uma aplicação pode pertencer a várias categorias. Esta relação é feita com a aplicação e não com a entidade, porque uma entidade pode abranger vários campos de atividade.

As colunas desta tabela são descritas em detalhe de seguida:

- **TagId:** Identificador único da categoria;
- **Name:** Nome da categoria.

4.2.6 Tabela ‘Country’

A tabela dos países surgiu, novamente, pela necessidade de comparar dados por país. Esta informação é particularmente importante para alguns dos parceiros do projeto que, sendo entidades nacionais, desejam monitorizar a conformidade nacional das diretrizes de acessibilidade. Esta tabela relaciona-se à tabela ‘Continent’ através de ‘muitos para um’ e à tabela ‘Application’ através de ‘um para muitos’.

As colunas desta tabela são descritas em detalhe de seguida:

- **CountryId:** Identificador único do país;
- **Name:** Nome do país;
- **ContinentId:** Identificador único do continente a que pertence este país.

4. DESENHO

4.2.7 Tabela ‘Continent’

Da mesma forma que é possível guardar dados correspondentes aos países, também o é relativamente a dados dos continentes. Esta tabela guarda simplesmente o nome do continente, permitindo, a um nível mais mundial, uma comparação do estado de monitorização da acessibilidade digital. Esta tabela relaciona-se simetricamente à ‘Country’ com uma relação ‘um para muitos’.

As colunas desta tabela são descritas em detalhe de seguida:

- **ContinentId**: Identificador único do continente;
- **Name**: Nome do continente.

4.2.8 Tabela ‘Organization’

Em ambos os geradores de declarações de acessibilidade considerados, é necessário fornecer o nome da entidade/organização responsável pela aplicação avaliada; esta tabela foi criada para guardar este nome e, conseqüentemente, possibilitar a comparação dos resultados das avaliações por entidade. É ainda dada a possibilidade de guardar a abreviatura do nome da entidade, porque esta opção está implementada no Observatório Português.

Esta tabela relaciona-se com a tabela ‘Application’ através de ‘um para muitos’, pois cada entidade pode ser responsável por múltiplas aplicações.

As colunas desta tabela são descritas em detalhe de seguida:

- **OrganizationId**: Identificador único da entidade;
- **Name**: Nome da entidade;
- **ShortName**: Abreviatura do nome da entidade;

4.2.9 Tabela ‘Application’

Esta tabela é uma das tabelas cruciais pois a classe daqui originada (aplicação) é a que interliga os dois tipos de documento de ‘input’. A tabela ‘Application’ armazena os dados relativos às aplicações/*websites* avaliados e, graças à sua posição na hierarquia, relaciona-se com a maior parte das classes. As colunas desta tabela permitem a relação com a entidade, o país de origem e o sector a que pertence (os dados da classe de sectores têm esta coluna como origem). É ainda diferenciado na coluna ‘Type’ se a tabela se refere a uma aplicação ou a um *website*, tendo em conta que as declarações de acessibilidade e os relatórios EARL se podem referir a ambos.

As relações desta tabela com as tabelas ‘Organization’, ‘Country’ e ‘Tag’ já foram explicadas, sobrando duas relações por mencionar: a ‘Application’ relaciona-se com as tabelas ‘Page’ e ‘AccessibilityStatement’ através de uma relação ‘um para muitos’, permitindo saber quais páginas e declarações pertencem à aplicação.

Foi ainda acrescentada uma coluna ‘Deleted’, que, funcionando como uma flag, dita se a aplicação já foi apagada ou não. Os requisitos atuais não apontam para a necessidade desta coluna, mas, no futuro, poderá ser uma mais valia interligar aplicações antigas com novas, tal como acontece aquando de uma mudança de nome.

A coluna ‘CreationDate’ representa a data mais antiga de que existe conhecimento desta aplicação, podendo ser a data da primeira declaração de acessibilidade introduzida, a data da primeira asserção (relativa à aplicação) introduzida ou, caso nenhuma destas seja válida, a data de introdução do documento na plataforma. De momento, não existe propósito para esta informação, sem ser o armazenamento de um detalhe, podendo vir a ser útil numa implementação futura (tal como a introdução deste filtro nos dados graficamente apresentados).

As colunas desta tabela são descritas em detalhe de seguida:

- **ApplicationId**: Identificador único da aplicação;
- **Name**: Nome da aplicação;
- **OrganizationId**: Identificador único da organização a que esta aplicação pertence;
- **Type**: Flag que representa se é aplicação ou *website*;
- **Sector**: Flag que representa o sector da aplicação;
- **Url**: URL do *website* (caso não seja aplicação);
- **CreationDate**: Data de criação da aplicação ou adição da aplicação à plataforma;
- **CountryId**: Identificador único do país a que esta aplicação pertence;
- **Deleted**: Flag que representa se a aplicação foi apagada.

4.2.10 Tabela ‘Page’

Como já foi mencionado, não existe uma classe de página porque atualmente não se deseja um nível de granularidade tão baixo na hierarquia de classes. Assim sendo, esta tabela tem como propósito armazenar as páginas avaliadas (encontradas nos relatórios EARL), para que, se necessário no futuro, seja possível comparar os resultados das asserções dependendo das páginas que as originaram. Atualmente, esta tabela tem como objetivo relacionar a ‘Assertion’ com a ‘Application’, com relações de ‘um para muitos’ e ‘muitos para um’ respetivamente.

De forma igual à tabela ‘Application’, foram introduzidas as colunas ‘CreationDate’ e ‘Deleted’, porque as páginas podem igualmente sofrer alterações de URL e originar a perda do seu caminho anterior.

As colunas desta tabela são descritas em detalhe de seguida:

4. DESENHO

- **PageId**: Identificador único da página;
- **Url**: URI da página;
- **CreationDate**: Data de adição da página à plataforma;
- **ApplicationId**: Identificador único da aplicação a que esta página pertence;
- **Deleted**: Flag que representa se a página foi apagada.

4.2.11 Tabela ‘Assertion’

A tabela ‘Assertion’ é a classe fundamental, pois serve de base para todo o modelo de dados; os dados guardados nesta tabela são o fundamento de todos os pedidos à base de dados. Cada entrada desta tabela corresponde a uma asserção, ou seja, uma declaração que materializa o resultado de um teste.

Como descrito anteriormente, uma asserção é a expressão do resultado de um teste feito a uma página (ou aplicação) por um avaliador, de um determinado modo, tendo em conta uma regra ou critério; e todas estas relações estão compreendidas nesta tabela. Descritas nas tabelas anteriores, a ‘Assertion’ relaciona-se através de ‘muitos para um’ com as tabelas ‘Page’, ‘Rule’ e ‘EvaluationTool’. Tendo em conta a linguagem EARL, o modo do teste decorrido deve ser descrito dentro de 5 opções, sendo elas *automatic*, *manual*, *semiAuto*, *undisclosed* e *unknownMode*. Como tal, apenas estes valores são permitidos na coluna ‘Mode’. De igual modo, o resultado do teste deve ser categorizado dentro de 5 opções, sendo *passed*, *failed*, *cantTell*, *inapplicable* e *untested* os valores permitidos na coluna ‘Outcome’.

As colunas desta tabela são descritas em detalhe de seguida:

- **AssertionId**: Identificador único da asserção;
- **EvaluationToolId**: Identificador único do avaliador que executou o teste correspondente à asserção;
- **RuleId**: Identificador único da regra executada no teste correspondente à asserção;
- **PageId**: Identificador único da página testada no teste correspondente à asserção;
- **Mode**: Modo do teste correspondente à asserção;
- **Date**: Data da asserção;
- **Description**: Descrição do teste correspondente à asserção;
- **Outcome**: Resultado do teste correspondente à asserção;
- **Deleted**: Flag que representa se a asserção foi apagada.

4.2.12 Tabela ‘AccessibilityStatement’

Com o mesmo intuito que a página, não é atualmente necessário o armazenamento detalhado das declarações de acessibilidade, mas como estas têm de ser interpretadas para recolher informação, é guardada esta informação para uma eventual utilidade destes dados. Esta tabela tem colunas que correspondem a dados interpretados dos resultados do gerador da W3C e do gerador português. As colunas relacionadas com os esforços e selo de usabilidade são interpretadas do gerador português, e as colunas de compatibilidades e incompatibilidades, tecnologias usadas e abordagem de avaliação são interpretadas do gerador da W3C. Em ambos os geradores, é possível interpretar qual a aplicação/*website* a ser estudado, a entidade responsável pelo mesmo e respetivos contactos, a data da declaração, o estado de conformidade e quais as diretrizes consideradas e ainda as limitações da aplicação/*website*.

De modo a recolher informação genérica, com possível utilidade futura, a informação relativa aos esforços, limitações, compatibilidades e incompatibilidades é reduzida a contadores do número de instâncias relatadas de cada tipo.

Existem ainda outros campos onde as opções são limitadas. A coluna ‘Standard’, correspondente às diretrizes consideradas, apenas guarda os valores *2.0*, *2.1*, *other* ou *unknown*. A coluna ‘State’, corresponde à conformidade da aplicação tendo em conta as diretrizes, guarda os valores 0, 1, 2 e 3, correspondentes a *não avaliado*, *não conforme*, *parcialmente conforme* e *totalmente conforme*. Estes mesmos valores são armazenados na coluna ‘UsabilityStamp’ que, descrevendo o selo de usabilidade da aplicação, correspondem a *nenhum*, *bronze*, *prata* e *ouro*. Por último, a coluna ‘AssessmentApproach’ usa um conjunto de três flags binárias para representar se a abordagem de avaliação foi interna, externa e/ou outra.

Esta tabela é relacionada com a ‘Application’ através de uma relação ‘muitos para um’ e com a tabela ‘Contact’ através de ‘muitos para muitos’, porque os mesmos contactos podem relacionar-se com diferentes declarações e a mesma declaração pode referir vários contactos.

As colunas desta tabela são descritas em detalhe de seguida:

- **ASId**: Identificador único da declaração;
- **Origin**: Identificador do gerador utilizado;
- **ApplicationId**: Identificador único da aplicação estudada na declaração;
- **Standard**: Diretrizes consideradas na declaração;
- **Date**: Data da declaração;
- **State**: Flag que representa a conformidade da aplicação;
- **ASUrl**: URL da declaração (se não for ficheiro);

4. DESENHO

- **UsabilityStamp**: Flag que representa o selo de usabilidade (apenas a partir do gerador português);
- **UsabilityStampText**: Informação adicional relacionada com o selo de usabilidade (apenas interpretado do gerador português);
- **EffortsCounter**: Contador de esforços (apenas interpretado do gerador português);
- **LimitationsWithAltCounter**: Contador de limitações com alternativas (apenas interpretado do gerador português);
- **LimitationsWithoutAltCounter**: Contador de limitações sem alternativas;
- **CompatibilitiesCounter**: Contador de compatibilidades (apenas interpretado do gerador W3);
- **IncompatibilitiesCounter**: Contador de incompatibilidades (apenas interpretado do gerador W3);
- **TechnologiesUsed**: Tecnologias usadas na aplicação (apenas interpretado do gerador W3);
- **AssessmentApproach**: Flag que representa a abordagem de avaliação (apenas interpretado do gerador W3);
- **Deleted**: Flag que representa se a declaração foi apagada;

4.2.13 Tabela ‘Contact’

Esta tabela guarda informação relativa aos contactos da entidade responsável pela aplicação. Esta é oriunda das declarações de acessibilidade e consta no tipo de contacto, o contacto em si e o tempo de resposta, sendo que o tempo de resposta é apenas interpretado nas declarações de acessibilidade resultantes do gerador da W3C. Os tipos de contacto interpretados das declarações são números de telefone, e-mails, moradas e nomes de utilizador da rede social Twitter.

Esta tabela é apenas relacionada com a ‘AccessibilityStatement’ através de uma relação de ‘muitos para muitos’.

As colunas desta tabela são descritas em detalhe de seguida:

- **ContactId**: Identificador único do contacto;
- **Type**: Tipo de contacto;
- **Contact**: Contacto em si;
- **DurationResponse**: Tempo estimado de resposta;

4.3 Tecnologias usadas

O objetivo desta dissertação é o desenvolvimento de uma plataforma de monitorização de dados em grande escala e a primeira ideia, que permaneceu na implementação, foi a de desenvolver uma aplicação web, pela facilidade de acesso às mesmas, e independente do *backend*, para que fosse facilmente publicada. Com o *backend* e o *frontend* desacoplados, a possibilidade de publicar a aplicação para o público em geral ou apenas para uma entidade específica é independente do servidor, e respetiva máquina, onde é publicada. Esta propriedade tornou-se ainda mais pertinente por inicialmente não existir um local definido para a publicação da plataforma, no âmbito do projeto.

Tendo em conta estes requisitos, em conjunto com a conveniência de desenvolver ambos os lados da aplicação em JavaScript e a familiaridade e conhecimento prévio das *frameworks*, foram escolhidos Angular para o *frontend* e Node e Express para o *backend*. Estas *frameworks* possibilitam modular tanto a camada de apresentação como a de negócio, facilitando a divisão de responsabilidades e eventuais iterações no desenvolvimento.

4.3.1 Highcharts

O Highcharts⁵ é uma biblioteca *freemium* de visualização de dados que oferece uma grande diversidade de gráficos, para além de possibilitar a gestão da informação apresentada em cada gráfico. É personalizável em vários aspetos, tais como os títulos e subtítulos, legendas, tamanho e estilo dos gráficos. Ainda possibilita a manipulação de eventos, tal como um clique na legenda ou numa coluna.

Os seus módulos de acessibilidade e exportação de dados são superiores a outras bibliotecas estudadas, tais como a Chart.js⁶ e a D3.js⁷. Permite uma navegação eficaz por teclado, possibilita a descrição do gráfico e a especificação da interpretação dos dados pelos leitores de ecrã, e uma exportação do gráfico para imagem ou ficheiro .pdf, .xls ou .csv.

Contudo, ao desenvolver gráficos nesta biblioteca, foi possível perceber que esta é orientada a gráficos estáticos. A descrição do gráfico, possibilitada pelo módulo de acessibilidade, tem de ser manualmente inserida, impossibilitando esta funcionalidade para gráficos dinâmicos. Com a constante modificação de dados dos gráficos que acontece ao navegar no PLACM, juntamente com as possibilidades de alterar a visibilidade dos dados para facilitar a leitura, é evidente que os gráficos a apresentar são dinâmicos e a API oferecida não ajuda muito nesse sentido. Quando é necessário mudar, ou esconder, apenas uma parte do conjunto de dados, é preciso alterar os dados do gráfico por completo. Este processo obriga o gráfico a renderizar por inteiro novamente, causando um problema de

⁵<https://www.highcharts.com/>

⁶<https://www.chartjs.org/>

⁷<https://d3js.org/>

4. DESENHO

desempenho considerável em utilizadores com máquinas mais fracas.

Mesmo com este contratempo, o Highcharts é uma biblioteca que oferece visualizações gráficas de grande qualidade e, especialmente, de forma acessível a todos, a custo zero.

4.3.2 Angular

O Angular⁸ é uma das *frameworks* mais conhecidas para o desenvolvimento de *frontend* de aplicações web. Esta framework, construída em TypeScript, oferece uma arquitetura baseada em componentes, o que permite dividir o desenvolvimento de componentes funcional e logicamente independentes. Apesar da sua versão mais recente ser a versão 11, esta ainda não estava disponível durante o desenvolvimento da plataforma. Como tal, foi utilizada a versão 10 do Angular.

O objetivo primordial do Angular era a criação de aplicações de página única, ou seja, páginas de conteúdo dinâmico. Para cumprir com este objetivo, o Angular oferece várias ferramentas tais como injeção de dependências, *routing* e construção de interfaces. O Angular disponibiliza ainda uma biblioteca de elementos baseados no *Material Design*, de modo a facilitar a integração de componentes de UI (*User Interface*). Estes elementos foram desenhados tendo em conta as necessidades de acessibilidade digital e, como tal, para além de melhorarem a aparência da aplicação, agilizam o acesso e navegação na mesma.

4.3.3 Node

O Node.js⁹ é um ambiente *runtime* baseado em JavaScript, permitindo assim usar a mesma linguagem para desenvolver *frontend* e *backend* de aplicações web. Graças ao paradigma *non-blocking* desta linguagem, o Node.js permite realizar múltiplas ações ao mesmo tempo numa única *thread*, melhorando o desempenho e eficiência do servidor. Para além destas vantagens, existe ainda um gestor de pacotes Node (npm¹⁰) onde, com a publicação de milhares de módulos, é incentivada a reutilização de código elaborado por outros desenvolvedores, juntamente com versionamento e gestão de dependências.

4.3.4 Express

O Express¹¹ é uma leve e flexível *framework* de servidor que complementa o Node.js, desenhado para facilitar o desenvolvimento de aplicações web. Este lida com as *routes* do servidor, encaminhando diferentes pedidos HTTP para um URL específico. É através

⁸<https://angular.io/>

⁹<https://nodejs.org/en/>

¹⁰<https://www.npmjs.com/>

¹¹<http://expressjs.com/>

4.3 Tecnologias usadas

destas *routes* que o servidor comunica com a base de dados, juntamente com o sistema MySQL. Para além desta funcionalidade, relevante para o PLACM, o Express permite a modificação de definições de aplicações web, tal como o porto de conexão, e ainda a gestão de cookies, sessões e autenticações, através de algumas bibliotecas de *middleware*¹².

¹²<http://expressjs.com/en/resources/middleware.html>

Capítulo 5

Implementação

Depois de organizadas as necessidades e estrutura de dados da plataforma, começou o processo de implementação. Neste capítulo, são descritos os aspectos principais do desenvolvimento da plataforma, tal como a interpretação dos dados a monitorizar, as diferentes formas de navegação pelos mesmos e as respetivas visualizações gráficas.

5.1 Origem e interpretação de dados

No levantamento de requisitos, percebeu-se quais seriam os documentos de dados que a plataforma iria receber de modo a obter informação para a plataforma: as declarações de acessibilidade e os relatórios EARL. Sendo o PLACM uma plataforma focada em dados em grande escala, é necessário fornecer ao utilizador alguma flexibilidade em relação à maneira de introdução de dados no sistema, de modo a agilizar o processo que pode acontecer centenas ou milhares de vezes. Foi então desenhada uma página de administração, específica para a introdução dos dois tipos de documentos estudados.

5.1.1 Página de administração

Esta página foi implementada para a introdução de dados no sistema. Dependendo da estrutura e objetivos da organização responsável pela plataforma, esta página deve ser exclusiva a um ou mais utilizadores responsáveis (administradores) do sistema, apesar de não ter sido implementada autenticação para tal.

A página está dividida em duas partes similares, para oferecer a interpretação dos dois tipos de documentos individualmente. O utilizador tem a possibilidade de fornecer as declarações e os relatórios através de ficheiros *.html* e *.json*, respetivamente, ou através de URLs, onde estes documentos estejam alojados. Ao fornecer estes documentos, é pedido ao utilizador o preenchimento de um formulário, de modo a relacionar o documento à informação útil para as representações gráficas. É neste formulário que é inserido manualmente, por exemplo, o país, as categorias e sector a que pertence a aplicação avaliada.

5. IMPLEMENTAÇÃO

No formulário relativo às declarações de acessibilidade, os campos obrigatórios são o gerador usado, para saber que elementos HTML devem ser interpretados para reconhecer os dados, o tipo de aplicação, para distinguir entre aplicação e *website*, e o sector da aplicação. Este último campo, ao contrário das categorias e país, é obrigatório porque não é desejado que fique por especificar esta informação. Quando não é oferecida informação relativa às categorias ou país a que pertence a aplicação testada, os dados introduzidos irão ser apresentados numa nova coluna nos gráficos relativos a estas classes, denominada por ‘Unspecified’. O nome da organização e aplicação testada não são obrigatórios porque podem ser interpretados das declarações, mas estes campos oferecem assim a possibilidade de modificar estes nomes. Tal como mencionado no levantamento de requisitos, as declarações de acessibilidade podem hiperligar a relatórios EARL e, quando tal acontece, não é necessário adicionar estes últimos documentos separadamente, pois é usada a informação já dada no formulário para relacionar as asserções encontradas.

No formulário relativo aos relatórios EARL é obrigatório enunciar o tipo de aplicação, sector da aplicação e ainda os nomes da organização e aplicação, por não haver esta informação disponível neste tipo de documento. Foi ainda introduzido o campo de URL da aplicação, caso esta seja um *website*.

5.1.2 Interpretação dos documentos

A interpretação das declarações de acessibilidade é baseada em classes CSS e atributos HTML específicos de cada gerador. Ambos os geradores interpretados, o português e o da W3C, evidenciam informação relevante, como por exemplo o nome da aplicação testada, em alguns elementos HTML com diferentes classes e atributos exclusivamente com o propósito de os distinguir de todos os outros elementos. É através da procura e leitura do texto destes elementos que a interpretação se baseia.

Ao implementar esta interpretação, foram encontradas declarações que não continham quaisquer classes ou atributos distinguíveis ou os nomes das classes tinham sido alterados devido a novas versões dos geradores. É importante que as entidades responsáveis pelas aplicações testadas mantenham o formato da declaração produzido pelos geradores, para além de ser necessário uma manutenção da interpretação dos resultados dos geradores na implementação do PLACM.

A interpretação dos relatórios EARL baseia-se numa função implementada na biblioteca *jsonld*¹, denominada por *frame*. Esta biblioteca corresponde a uma implementação em *Javascript* da especificação JSON-LD e foi considerada fidedigna para este projeto, porque o mapeador de implementações das regras ACT, desenvolvido no contexto do WAI-Tools, usa a mesma biblioteca com o objetivo análogo.

A função *frame* fundamenta-se no contexto EARL fornecido para retornar as asserções encontradas nos relatórios. O contexto EARL é um ficheiro que descreve a

¹<https://www.npmjs.com/package/jsonld>

5.1 Origem e interpretação de dados

estrutura de dados do relatório a estudar. O contexto considerado no PLACM teve como base o oferecido na comunidade das regras ACT². Este sofreu algumas alterações tendo em conta os resultados dos geradores de relatórios EARL desenvolvidos pelos vários parceiros do projeto. Apesar de todos estes geradores serem baseados em JSON-LD, não existe uma forma padronizada de relatórios EARL e, como tal, o contexto EARL do PLACM especifica uma maior quantidade de campos, de modo a interpretar o máximo de informação possível. Destes campos adicionados, são de realçar: o campo ‘assertedBy’ (para relacionar o *assertor* com a asserção), os diferentes resultados possíveis das asserções e os modos de avaliação possíveis.

5.1.3 Cálculo das métricas

Depois das asserções serem encontradas nos relatórios, estas são guardadas na base de dados, para que, nos pedidos à mesma, seja processado o cálculo das métricas. Ambos os conjuntos das métricas, as relacionadas com as asserções e os critérios de sucesso (descritas na secção 4.1.2), são baseadas nas asserções, mas apenas num sub-conjunto específico das mesmas. Como as páginas de uma aplicação podem ser avaliadas por diferentes critérios com diferentes avaliadores, as asserções consideradas para o cálculo das métricas são as mais recentes do tuplo [critério, página, avaliador]. Foi escolhida esta relação, em vez da simples [critério, página], para abranger os resultados de vários avaliadores; ao considerar apenas o resultado mais recente de um critério numa página, podemos estar a considerar o resultado de um avaliador mal-implementado ou desenhado para retornar apenas resultados positivos. Assim, é evitado um controlo dissimulado dos resultados das métricas, obrigando a uma coerência dos vários avaliadores usados. É de apontar que, apesar de inicialmente as regras ACT serem apontadas como os únicos possíveis critérios deste tuplo (secção 2.3.2.4), também passaram a ser considerados os critérios de sucesso, pois o gerador de relatórios EARL da WAI³ gera asserções onde estes são o *TestCriterion*.

O cálculo das métricas relacionadas com as asserções consiste na soma das asserções mais recentes por resultado, e o número de páginas correspondentes a essas asserções. Por outro lado, as métricas relacionadas com os critérios exigem um cálculo mais complexo que uma soma. Ao obter o conjunto de asserções mais recentes, estas são agrupadas por critério de sucesso. A partir do resultado das asserções agrupadas, é inferido o resultado do critério de sucesso: se existir pelo menos uma asserção falhada, o critério falha; se não existirem asserções falhadas e existir pelo menos uma asserção *cantTell*, o critério resulta em *cantTell*; se não existem asserções falhadas nem *cantTell* e existir pelo menos um asserção passada, o critério passa; se não existirem asserções com quaisquer dos resultados anteriores e existir pelo menos uma asserção inaplicável, o critério resulta em inaplicável. Se não existirem asserções relacionadas com algum critério de sucesso, ou

²<https://act-rules.github.io/earl-context.json>

³<https://www.w3.org/WAI/eval/report-tool>

5. IMPLEMENTAÇÃO

se apenas existirem asserções de resultado *untested*, o critério é considerado não-testado. Finalmente, é realizada a soma dos resultados dos critérios de sucesso.

Apesar de, nas WCAG, não serem definidas as condições de resultado *canTell* e *untested* para os critérios de sucesso, estas foram empregues no PLACM porque os resultados das asserções é que definem os resultados dos critérios, permitindo obter informação mais detalhada em relação à conformidade real de acessibilidade.

Depois do cálculo das métricas no *backend*, a apresentação destes resultados é responsabilidade do *frontend*. As funcionalidades da plataforma estão equivalentemente divididas entre o cliente e o servidor, mas o cliente é considerado “gordo” devido à implementação de uma cache de resultados. Depois da realização de um pedido ao servidor, o resultado é guardado na cache do cliente, permitindo assim tornar-se independente do servidor. A cache é apagada sempre que o cliente fecha o seu navegador ou, para os pedidos de nomes de categorias e países, necessários na página de administração, ao fim de um minuto.

5.2 Vistas de gráficos

Como plataforma de monitorização de dados em grande escala, é importante apresentar gráficos cuja informação a transmitir não deixe margens para dúvidas. Com a ajuda da *framework Highcharts*, foram implementadas várias vistas, resultantes das diferentes formas de navegação, que serão explicadas na próxima secção.

Em todas as vistas implementadas, o eixo vertical dos gráficos corresponde aos valores das métricas usadas. O eixo horizontal corresponde aos dados relativos à classe a estudar. É em cada ponto deste eixo que são introduzidas as diferentes métricas, para que seja possível induzir facilmente uma comparação dos valores das mesmas por todo o conjunto de dados.

As métricas a usar na monitorização de dados através desta plataforma foram divididas em dois conjuntos, as relacionadas com as asserções e as relacionadas com os critérios de sucesso. Segmentando as métricas a estudar, os gráficos apresentam informação sempre relacionada e em menor quantidade, tornando-os mais claros. A métrica do número de páginas testadas foi incluída apenas no primeiro conjunto devido à dificuldade de percepção do tamanho do conjunto a estudar tendo em conta o número de asserções; no conjunto avesso, o número de critérios de sucesso é estático e, como tal, não necessita de ser relativizado com o número de páginas nem asserções.

Em todas as vistas, o conjunto de métricas predefinido é o das asserções. Contudo, é dada ao utilizador a oportunidade de alterar o grupo de métricas a estudar, independentemente do tipo de navegação realizada, excepto quando a vista corresponde às últimas três classes da hierarquia. Os dados das classes ‘Critério de sucesso’, ‘Tipo de elemento’ e ‘Regra’ estão sempre relacionados a um sub-conjunto de critérios de sucesso. As vistas resultantes destas classes, estudando as métricas dos critérios de sucesso, iriam mostrar

5.3 Formas de navegação

sempre uma grande quantidade de critérios não testados, pois uma grande parte destes nem se relacionam com os dados. Para impedir esta incoerência de dados, foi desativada a possibilidade de alterar o conjunto de métricas nas vistas mencionadas, apresentando, nestes casos, as métricas relacionadas com as asserções.

O *Highcharts* permite seleccionar, através da legenda de cada gráfico, quais as métricas a apresentar no gráfico. As métricas apresentadas por predefinição em cada conjunto de métricas são o número de páginas, para o conjunto relacionado com as asserções, e o número de critérios passados e falhados, para o conjunto relacionado com os critérios. Neste segundo conjunto, são apresentadas duas métricas porque a métrica de critérios passados tende a resultar numa coluna vazia. Como descrito na secção 5.1.3, é necessário um grande esforço por parte dos desenvolvedores de conteúdo para que os critérios de sucesso passem, pois basta uma asserção com qualquer outro resultado para impedir um resultado positivo do critério. Se fosse apenas apresentada a métrica de critérios passados, o gráfico tenderia a ser inicializado sem valores, causando confusão ao utilizador.

Por não existir nenhuma funcionalidade na API do *Highcharts* que permite esconder pontos específicos do conjunto de dados do gráfico, foi implementada uma lista de *checkboxes*, onde cada uma destas representa um ponto do eixo horizontal. Estas encontram-se sempre no fundo da página, depois dos gráficos representados, e permitem alternar a visibilidade das colunas (ou dos gráficos, numa vista específica). Esta lista é visível no canto inferior esquerdo da figura 5.1.

5.3 Formas de navegação

No levantamento de requisitos, foram suscitadas 5 formas de navegação pelos dados, de modo a permitir agilizar o processo de monitorização por parte do utilizador. Mais tarde, foi implementada uma outra forma de navegação, de modo a permitir estudar detalhadamente os resultados dos critérios de sucesso específicos a uma aplicação. Exceptuando a navegação simples, as restantes formas de navegação são acessíveis através dos separadores da janela modal, apresentada na figura 5.2, que aparece ao carregar (com rato ou teclado) em qualquer coluna do gráfico. As seis formas serão explicadas de seguida:

1. **Navegação simples para os dados de uma classe:** Esta navegação baseia-se em permitir a visualização dos dados relativos a uma classe por completo, tal como os dados relativos a todas as aplicações ou tipos de elemento. A vista resultante desta navegação é a de um único gráfico onde cada ponto do eixo das abcissas representa um dado da classe escolhida. As *checkboxes* desta vista estão interligadas a esses mesmos pontos, alternando a sua visibilidade no eixo dos XX.

A vista predefinida do PLACM, representada na figura 5.1, consiste nesta mesma visualização, usando os dados da classe ‘Continente’, de maneira a representar os dados mais genéricos, o “topo” da hierarquia.

5. IMPLEMENTAÇÃO

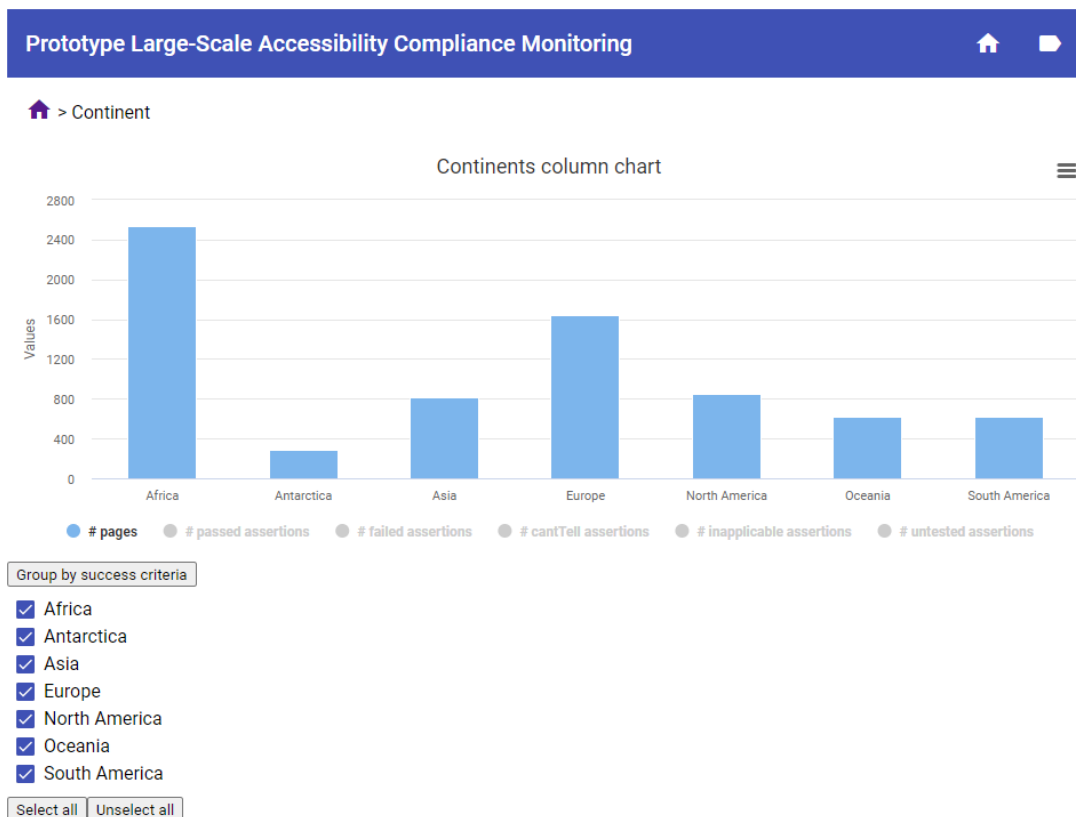


Figura 5.1: Visualização predefinida do PLACM

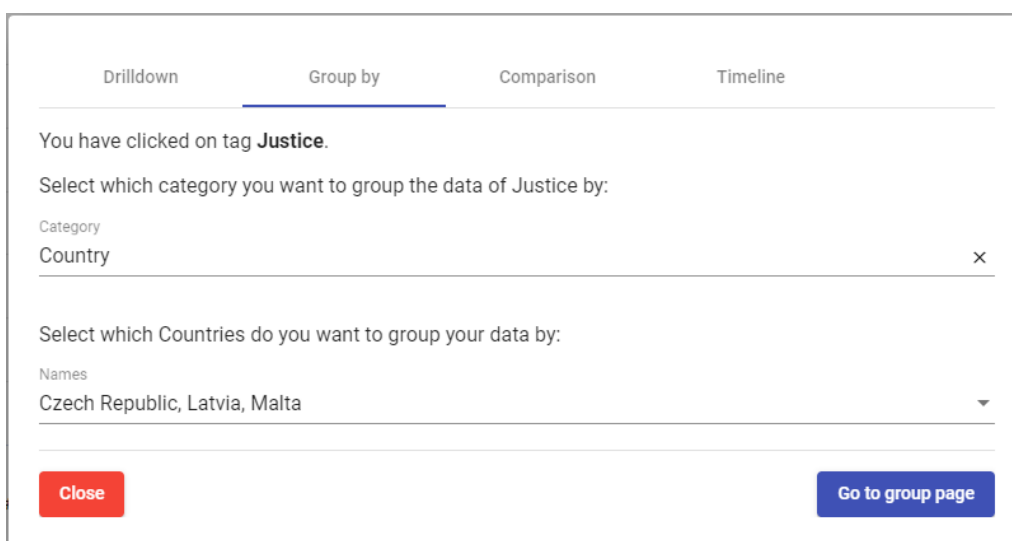


Figura 5.2: Janela modal, no separador *Group by*

5.3 Formas de navegação

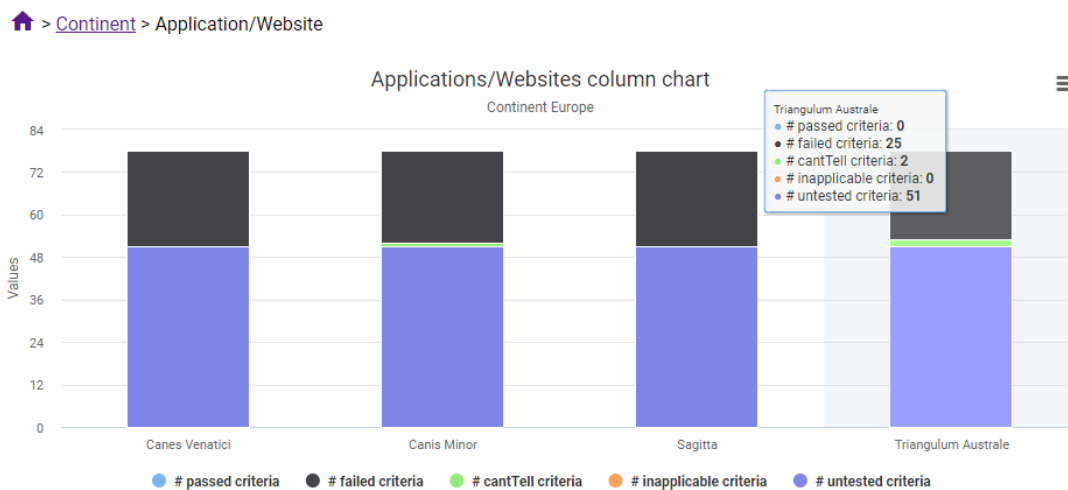


Figura 5.3: Vista das métricas relacionadas com os critérios de sucesso, resultante da navegação para os dados de ‘Aplicação/Website’ a partir do continente europeu

Na interface implementada, o utilizador pode navegar entre as classes ao carregar no botão *label*, que se encontra no canto superior direito, e seleccionar uma classe do *dropdown* daí originado.

2. **Navegação para “descer” na hierarquia:** Esta navegação baseia-se em aumentar a especificidade dos dados apresentados, para permitir uma monitorização dos dados de acessibilidade mais pormenorizada. A vista resultante desta navegação é a de um único gráfico onde, tal como acontece na navegação anterior, os pontos do eixo das abcissas representam os dados da classe escolhida e as *checkboxes* alternam a visibilidade desses pontos. Na figura 5.3 é exemplificada uma vista resultante desta navegação, cujas métricas da mesma correspondem às relacionadas com os critérios de sucesso.

Ao descer de nível na hierarquia, é acrescentado um filtro a todas as navegações posteriores, ou seja, os dados que serão mostrados terão sempre em conta as navegações deste tipo que o utilizador já realizou. Esta forma de navegação permite ao utilizador descer a qualquer classe que se encontra num nível inferior; para haver uma maior flexibilidade de filtragem, não obriga o utilizador a descer para o nível diretamente abaixo.

Pode ser feita vezes sucessivas, apenas com algumas restrições: Se as métricas da vista são as relacionadas com os resultados das asserções, a única restrição imposta é o último nível da hierarquia, a classe ‘Regra’; se as métricas são as dos resultados dos critérios de sucesso, o último nível de hierarquia passa a ser a ‘Ferramenta de avaliação’, tal como descrito na secção anterior.

É através desta navegação que o utilizador pode especificar, por exemplo, a monitorização das regras avaliadas em Portugal (filtrando o ‘País’ Portugal e des-

5. IMPLEMENTAÇÃO

cendo até à ‘Regra’) ou as aplicações europeias no campo de saúde (filtrando o ‘Continente’ Europa, a ‘Categoria’ Saúde e descendo até à ‘Aplicação/Website’).

Na interface implementada, depois de carregar em qualquer coluna do gráfico, é interpretado o ponto a que a coluna seleccionada pertence, adicionando assim um filtro. Esta forma de navegação é alcançável sempre pelo primeiro separador da janela modal, denominado por *Drilldown*. Consequentemente, as classes dos níveis inferiores à vista atual, tendo em contas as restrições já mencionadas, são dadas como opções de navegação para o utilizador.

3. **Navegação para “subir” na hierarquia:** Esta navegação baseia-se, inversamente à anterior, em diminuir a especificidade dos dados, com o intuito de comparar diretamente os dados da classe inferior tendo em conta a classe superior seleccionada. É através desta navegação que é exequível, por exemplo, a comparação dos resultados da ferramenta de avaliação A nos países Portugal e Espanha (ao clicar na ‘Ferramenta de Avaliação’ A, escolher a classe ‘País’ e os dados Portugal e Espanha).

Na interface implementada, este tipo de navegação é alcançado igualmente através da janela modal, no separador *Group by*. Neste separador, o utilizador necessita de escolher uma classe a um nível superior e quais os dados dessa classe que deseja comparar, sendo estes dependentes dos filtros existentes. Concretamente, os filtros irão restringir as opções dos dados a comparar, porque o dado correspondente ao ponto seleccionado do gráfico necessita de pertencer ao sub-conjunto de dados da classe superior escolhida.

Como predefinição, a visualização resultante deste tipo de navegação consta num único gráfico, onde o eixo horizontal corresponde às métricas a estudar e o eixo vertical o valor das mesmas. Os dados a comparar, seleccionados previamente pelo utilizador, são descritos em cada um dos pontos do eixo horizontal, permitindo assim a comparação direta dos dados por métrica. Contudo, é oferecida a oportunidade de mudar de vista e representar cada dado num gráfico separado, similar à vista da navegação anterior. Ao seleccionar a vista de múltiplos gráficos, a comparação complica-se porque os dados não estão juntos num só gráfico, originando uma comparação descontinuada e inextricável. Esta vista foi implementada para permitir a presença de similaridade entre as visualizações e aproveitar o conhecimento prévio do utilizador.

Ao contrário do *Drilldown*, este tipo de navegação apenas pode ser realizada uma vez, pois permite “subir” na hierarquia mas com o intuito de comparação de valores, não de limitação repetida dos dados apresentados.

4. **Comparação no mesmo nível de hierarquia:** Este tipo de navegação permite, de uma forma mais expressa, a comparação de dados de uma mesma classe. A comparação de dados pode ser realizada numa vista de dados de uma classe, usando

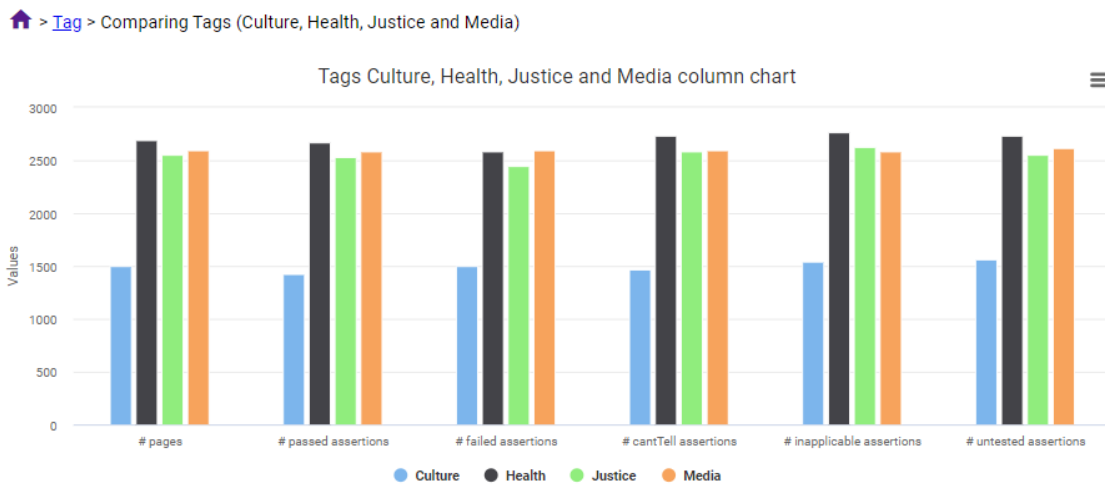


Figura 5.4: Vista de único gráfico resultante da comparação dos dados relativos às categorias de cultura, saúde, justiça e média

as *checkboxes*, mas a navegação a estudar foi implementada de modo a permitir comparação de dados através de outras duas vistas.

Como predefinição, esta navegação origina uma vista de um único gráfico onde os pontos do eixo das abcissas representam as várias métricas, e as colunas de cada ponto os dados a comparar, como representado na figura 5.4. Contudo, é dada como opção uma vista de vários gráficos (um por cada dado comparado) onde os seus eixos dos XX contêm um único ponto, que representa o dado a comparar. A vista de vários gráficos foi implementada de modo a haver continuidade gráfica com o tipo de navegação anterior, mas, como serão gerados bastantes gráficos ao comparar muitos dados, uma comparação fiável e imediata com esta vista torna-se impossível. Para responder a este problema e facilitar a leitura da informação apresentada, a vista de um único gráfico foi implementada e é a predefinida desta navegação.

Na interface implementada, este tipo de navegação é alcançado através da janela modal, no separador *Comparison*. Tal como em todos os tipos de navegação, todos os filtros até à seleção desta navegação são considerados. As *checkboxes* alternam a visibilidade de cada ponto ou de cada gráfico, dependendo da vista.

Este tipo de navegação não é possível realizar sucessivamente, porque não faz sentido comparar dados e ainda possibilitar uma comparação de um sub-conjunto desses mesmos dados. Assim sendo, é apenas realizável uma única vez, podendo ter como raiz a vista de dados de uma classe (originada pelas navegações 1 e 2) e a vista originada pela navegação 3. Os filtros anteriores continuam a ser considerados nos dados apresentados.

5. **Linha temporal:** Este tipo de navegação permite monitorizar a evolução temporal

5. IMPLEMENTAÇÃO

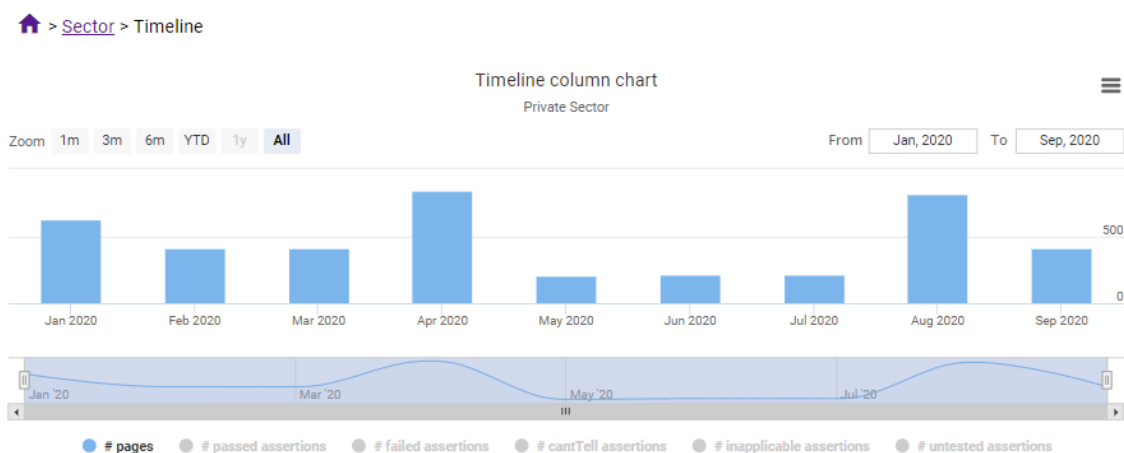


Figura 5.5: Vista resultante da navegação temporal, apresentando a evolução temporal dos dados do setor privado

dos resultados de testes registados na plataforma. A vista resultante desta navegação é um único gráfico em que os pontos do eixo das abcissas correspondem a um mês onde houve registo de resultados de testes na plataforma, representada na figura 5.5. Os dados apresentados em cada ponto simbolizam apenas os resultados registados nesse mês, e não um aglomerado de todos os resultados até esse ponto temporal. Com um aglomerado, não seria possível distinguir entre uma melhoria temporal e um aumento de testes, pois o número total de asserções poderia aumentar e os resultados permanecerem os mesmos.

A implementação deste gráfico foi através de uma outra biblioteca da família *Highcharts*, a *Highcharts Stock* que, apesar de ser focada para o mundo financeiro, pareceu mais adequado que a *Highcharts*. A *Highcharts Stock* oferece o mesmo tipo de gráfico que a *Highcharts*, com as mesmas vantagens, mas acrescenta filtros temporais programáveis e uma barra horizontal inferior ao gráfico, para permitir a navegação pelos dados temporais. Sendo uma visualização temporal, este gráfico tende facilmente a crescer e, com uma barra de navegação horizontal, os dados apresentados simultaneamente podem ser reduzidos, facilitando a leitura do mesmo. A adição dos filtros temporais agiliza a monitorização de dados de intervalos temporais recentes. Os filtros temporais implementados oferecem a filtragem pelos intervalos temporais predefinidos da biblioteca: um mês, três meses, seis meses, um ano e o ano atual.

Na interface implementada, este tipo de navegação é alcançado através da janela modal, no separador *Timeline*. Novamente, todos os filtros até à seleção desta navegação são considerados.

- 6. Conformidade dos critérios de sucesso numa aplicação/website:** Este tipo de navegação é o único que permite analisar diretamente as asserções guardadas. Na

5.3 Formas de navegação

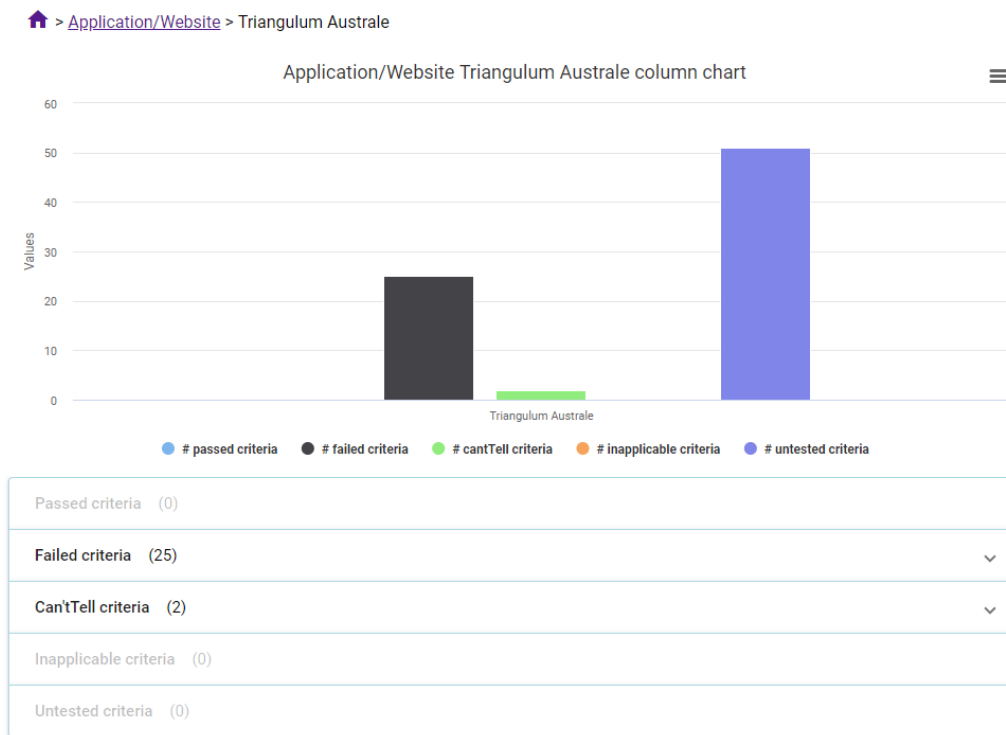


Figura 5.6: Vista resultante da navegação de conformidade dos critérios de sucesso numa aplicação/*website*, apresentando uma listagem, por resultado, de todos os critérios de sucesso

interface implementada, é alcançado através da janela modal, no separador *Details*. Este separador é apenas visível quando o utilizador está perante uma vista dos dados da classe de ‘Aplicação/*Website*’ e com as métricas relativas ao critério de sucesso.

A vista resultante desta navegação é a de um único gráfico de barras, exemplificado na figura 5.6, com um ponto no eixo das abcissas a apresentar as métricas dos critérios de sucesso da aplicação/*website* seleccionada, similar à visualização gráfica simples; foi aplicado o gráfico de barras simples em vez do de barras empilhadas, para discriminar os valores das diferentes métricas. Sob o gráfico, estão listados todos os critérios de sucesso WCAG 2.1, agrupados pelo seu resultado, num menu acordeão. A seleção de um critério apresenta uma listagem das asserções relacionadas com o mesmo, mostrando o resultado, a regra usada, a página testada, o avaliador e a descrição da mesma.

Este tipo de navegação foi implementado para diminuir o nível de granularidade até à classe mais específica de dados, a asserção. Contudo, foi implementada ao nível da aplicação para reduzir ao máximo a informação apresentada e ser de mais fácil leitura e interpretação.

5. IMPLEMENTAÇÃO

5.4 Modificações ao Highcharts

Com a utilização da *framework* Highcharts para gerar os diferentes tipos de gráficos, especialmente com a adição dos módulos de acessibilidade e exportação, foi possível oferecer alternativas à interação e leitura dos dados apresentados, providenciando navegação por teclado e leitura através de leitores de ecrã. Ainda assim, esta *framework* não disponibiliza API para suportar a interação simultânea em múltiplos gráficos, tornando custosa a tarefa de apresentar múltiplos gráficos numa vista. São de seguida explicadas algumas medidas aplicadas para alcançar uma maior acessibilidade e usabilidade, especialmente importantes nestas últimas vistas:

- **Tabela:** O Highcharts oferece a funcionalidade de apresentar uma tabela com os dados do gráfico, acessível através do menu gerado pelo módulo de exportação. Contudo, este gera apenas uma tabela por gráfico, tornando a comparação de dados impossível quando a vista é de múltiplos gráficos. Foi então implementada à parte, com o uso do elemento HTML `<table>`, renderizado entre o gráfico e as *checkboxes*, uma tabela que aglomerada a informação de todos os gráficos apresentados. A opção predefinida de apresentação da tabela no menu de exportação foi substituída para corresponder à ação de alternar a visibilidade desta tabela conjunta.
- **Tooltips:** Por pré-definição, no seu módulo base, o Highcharts permite que seja apresentado um *tooltip* quando o cursor do rato passa por cima das colunas, para facilitar a leitura dos dados do gráfico, como é visível na figura 5.3. Contudo, esta funcionalidade não existe quando a navegação pelos dados é feita através do teclado. Manipulando os eventos DOM de *mousemove*, *mouseleave*, *touchmove*, *touchstart* e *keyup*, foi possível implementar a apresentação das *tooltips* em diversos gráficos ao mesmo tempo, quer através do cursor ou do teclado. Assim é possível comparar os dados na mesma posição em gráficos diferentes diretamente pela informação apresentada na *tooltip*.
- **Legendas:** O Highcharts trata da seleção das legendas dos gráficos automaticamente, apesar de disponibilizar a opção de programar esta tarefa. Na maior parte das vistas do PLACM, as legendas correspondem às métricas a estudar, logo é importante que sejam visíveis, principalmente nas vistas de múltiplos gráficos, as mesmas legendas em todos estes, originando coerência em toda a vista. Para isso, foi programada uma cascata para que, quando a legenda é carregada num dos gráficos, quer por cursor do rato ou teclado, as legendas seleccionadas sejam sempre as mesmas.
- **Colunas:** A API do Highcharts oferece deteção de eventos de clique nas colunas dos gráficos. Contudo, a escala dos gráficos pode dificultar o clique em colunas que representam valores demasiado baixos. Para evitar este problema, pode ser usada

a navegação por teclado oferecida pelo módulo de acessibilidade, mas foi também implementada a deteção do clique fora da coluna, de modo a simular o clique na mesma.

5.5 Arquitetura

Juntamente com o requisito de elaborar uma aplicação web cujo desenvolvimento do *frontend* e do *backend* é desacoplado, descrito na secção 4.3, foi aplicado o estilo de arquitetura REST. Neste estilo, o cliente envia pedidos ao servidor para receber ou modificar informação e este último limita-se a responder aos pedidos, possibilitando o desenvolvimento de uma parte da aplicação independente da outra. Nesta secção são expostos os pontos mais importantes na implementação do cliente (em Angular) e servidor.

5.5.1 Cliente

Tal como explicado na secção 4.3.2, o Angular fornece várias ferramentas para desenvolver conteúdo web dinâmico. Para além dos componentes, que permitem separar os conceitos da aplicação, o Angular oferece ainda a possibilidade de criar serviços. Deste modo, os componentes Angular ficam responsáveis pela parte de apresentação, enquanto os serviços tratam do processamento de informação ou dos pedidos de dados ao servidor. De seguida, são descritos os aspetos e decisões importantes dos componentes e serviços implementados.

5.5.1.1 Componentes

Beneficiando da funcionalidade do Angular de desenvolver partes da aplicação em diferentes componentes, a arquitetura do *frontend* baseou-se na implementação dos componentes por tipos de navegação. Os componentes principais são explicados de seguida:

- *graphic-header*: Este componente é fundamentalmente um cabeçalho da aplicação, que se mantém visível independentemente dos gráficos apresentados;
- *graphic-breadcrumbs*: Neste componente, é calculado o texto a apresentar nos *breadcrumbs* da aplicação, de modo a permitir que o utilizador se recorde das navegações (e escolhas) realizadas. Este componente é inicializado nos componentes relacionados com as vistas dos tipos de navegação, pois o cálculo dos *breadcrumbs* tem de ser feito sempre que os gráficos são carregados, relacionando a navegação feita com a visualização apresentada;
- *graphic-display*: Este componente trata da apresentação originada pelas navegações 1 e 2, originando um único gráfico;

5. IMPLEMENTAÇÃO

- *graphic-compare*: Neste componente, é tratada a visualização das navegações 3 e 4, apresentando uma quantidade de gráficos dependente dos dados a apresentar;
- *graphic-timeline*: Neste componente, é implementada a apresentação da visualização da navegação 5, apresentando um único gráfico mas usando a *framework Highcharts Stock*;
- *app-sclist*: Este componente apresenta o gráfico e lista de critérios de sucesso, originados na navegação 6;
- *admin*: Este componente corresponde à página de administração, inicializando os dois componentes relativos à submissão de declarações de acessibilidade e relatórios EARL;
- *drilldown-dialog*: Este componente corresponde à janela modal que permite a escolha entre os diferentes tipos de navegação implementados.

5.5.1.2 Router

Um ponto que se veio a verificar fulcral para o desenvolvimento do *frontend* foi o requisito de todas as vistas gráficas serem partilháveis através do URL das mesmas. Para implementar este requisito, foi necessário usar a funcionalidade embutida de encaminhamento e navegação do Angular, denominada de *Router*, de modo a alterar os componentes carregados e, conseqüentemente, a vista a apresentar, tendo em conta o URL.

O *Router* permite escolher uma de duas estratégias de encaminhamento, sendo que apenas uma, a *PathLocationStrategy*, permite a utilização descomplicada de parâmetros, ao mesmo tempo que mantém o URL claro (por não acrescentar '#', como a *HashLocationStrategy*). A estratégia implementada foi a primeira porque é necessário guardar as escolhas do utilizador em cada vista e isso pode ser facilmente feito através dos parâmetros de URL.

Como cada navegação e seleção por parte do utilizador teriam de se traduzir no URL, enquanto este se mantém facilmente interpretado, os caminhos são declarados com a ordem tipo de visualização, variáveis-base e classe. Por exemplo, na vista de comparação entre países, tendo as asserções como variáveis-base, o caminho é */compare/assertions/country*, sendo de fácil leitura para o utilizador comum. Esta declaração é feita no ficheiro *app.module.ts*, de modo a definir o módulo *Router* e permitir a sua utilização por todos os componentes.

Com o reaproveitamento de código alcançado pela definição de componentes Angular, alguns caminhos acabam por inicializar o mesmo componente, sendo necessária uma interpretação do URL em cada um deles, de modo a apresentar os dados corretos. A interpretação do caminho é bastante direta graças à API fornecida pelo módulo *Router*, enquanto que para a interpretação dos parâmetros do URL foi necessário um maior

esforço “manual”, para uma correta apresentação das *checkboxes* e legendas e tipos de gráficos.

5.5.1.3 Serviços

Como a comunicação entre a aplicação e o servidor deve ser feita através de serviços, foram criados serviços correspondentes a cada classe de dados, de modo a realizar pedidos de informação distintos ao servidor. Para que os vários componentes não tenham de chamar funções de serviços diferentes, os serviços relacionados com as classes foram agrupados num único, denominado por *CombinedService*. Foram ainda desenvolvidos serviços para a lógica relacionada com a submissão de declarações de acessibilidade e relatórios EARL.

Os pedidos de informação baseiam-se numa operação HTTP e o caminho pretendido para o recurso onde a operação deve ser feita. Estes caminhos foram desenvolvidos de modo a permitir que o cliente perceba facilmente qual a operação a ser realizada. Na implementação do PLACM, os pedidos de informação contêm adicionalmente o nome da base de dados correspondente à instância da aplicação, pois foram desenvolvidas várias instâncias da aplicação para que os parceiros do WAI-Tools pudessem testar a adição e monitorização dos seus próprios dados de maneira privada. Deste modo, os pedidos feitos ao servidor são desenvolvidos e realizados de maneira igual, sendo apenas diferenciados na base de dados a utilizar.

5.5.1.4 Cache

O PLACM foi orientado para dados em grande-escala logo, para ser competente na gestão de uma grande quantidade de informação, foi utilizado um mecanismo de cache, aplicando a *Web Storage API*⁴. Esta API oferece duas possibilidades de armazenamento: a *localStorage* e a *sessionStorage*, onde a diferença entre ambas é o tempo de expiração de informação. Enquanto que, ao aplicar a *localStorage*, a informação não tem tempo de expiração, a *sessionStorage* apenas permite o armazenamento de informação até o término de uma sessão do *browser*, isto é, até este ser fechado.

Como a informação do PLACM tende a crescer, foi aplicada a *sessionStorage*, de modo a permitir uma atualização constante da informação apresentada, sem repetir excessivamente pedidos inalterados ao servidor. Com este mecanismo de cache, a informação a apresentar apenas precisa de ser pedida uma vez por sessão, melhorando o desempenho de ambos os lados da aplicação, especialmente relevante se o utilizador visitar informação previamente pedida.

⁴https://developer.mozilla.org/en-US/docs/Web/API/Web_Storage_API/Using_the_Web_Storage_API

5. IMPLEMENTAÇÃO

5.5.2 Servidor

Para uma melhor modularidade, o servidor deveria estar estruturado por *routers*, controladores e modelos, de modo a receber o pedido, tratar o pedido e introduzindo lógica adicional se necessário e realizar o pedido à base de dados. Tendo em conta que no PLACM o servidor tem a responsabilidade de apenas guardar e devolver informação da base de dados, a estrutura deste baseia-se em *routers* e modelos, ignorando a lógica nos controladores.

Mantendo a organização dos serviços no cliente, foram criados *routers*, e respetivos modelos, para cada classe de dados a ser tratada. Em todos os modelos, foram implementados pedidos MySQL para o cálculo das métricas, separando as métricas relacionadas com as asserções das relacionadas com os critérios de sucesso. Foram ainda, tal como nos serviços do cliente, implementados modelos relacionados com a inserção de informação derivada das declarações de acessibilidade e relatórios EARL.

Como não foi determinado onde seria publicado o servidor e como este tem apenas a pequena responsabilidade de responder a pedidos, foi considerado que uma única instância do servidor iria servir diferentes instâncias da aplicação. Para cumprir com este objetivo, foram criadas várias bases de dados, e correspondentes ficheiros de configuração, para que as instâncias das aplicações apresentassem apenas os seus devidos dados.

5.6 Crawler

No contexto de uma plataforma de monitorização de dados em grande escala, é importante encontrar e incorporar dados prontamente para uma evolução da informação apresentada. Tendo em conta as fontes de dados da plataforma, foi implementado um *crawler* simples que encontra declarações de acessibilidade, dado um único domínio ou uma lista de domínios. Este *crawler*, desenvolvido com a biblioteca *Apify*⁵, procura pelos *links* no primeiro nível da página inicial do domínio e, tendo em conta as classes que os geradores cuja interpretação foi implementada, percebe se algum desses links é uma declaração. É ainda adicionado manualmente o URL *domínio-a-pesquisar/acessibilidade* a esta procura.

A implementação do *crawler* foi ditada pelo ponto 2, do 8º artigo, do 2º capítulo do decreto-lei nº 83/2018, que menciona que as declarações de acessibilidade devem ser publicadas numa página cujo URI acabe em */acessibilidade* e esta tem de estar hiperligada à página inicial do *website*.

Sendo simples, o resultado do *crawler* é apenas acessível através do pedido HTTP para a *route /proto/findASLinks*. Foi ainda incluída a biblioteca de interpretação de ficheiros *.xlsx* e *.csv convert-excel-to-json* para facultar outra possibilidade de *input* de domínios.

⁵<https://www.npmjs.com/package/apify>

Capítulo 6

Avaliação

Depois da primeira iteração de desenvolvimento, foram realizados testes com utilizadores, com o objetivo de detetar possíveis problemas de usabilidade inerentes a uma grande quantidade de dados a monitorizar e diferentes tipos de navegação implementados.

Durante o desenvolvimento, ainda foi percebido que seria necessário melhorar o desempenho dos pedidos feitos à base de dados e ao processamento de dados na visualização de gráficos. Qualquer demora na resposta pode causar problemas de acessibilidade no *frontend*, para além do desconforto que causa a qualquer utilizador.

6.1 Testes com utilizadores

Foram realizados testes com utilizadores para avaliar a existência de problemas de usabilidade para o utilizador comum da plataforma. No total, foram realizadas duas rondas de testes pois, como numa primeira ronda foram detetados problemas e melhorias necessárias, foram implementadas as devidas alterações e repetido o processo de testagem até não haver problemas a corrigir.

Os testes constavam na resposta a 20 perguntas, apresentadas na lista B em anexo, relacionadas com os dados apresentados, onde cada pergunta levaria o utilizador a realizar tarefas que incluem tipos de navegação e filtrações específicas. De modo a simular casos de uso verídicos, a base de dados do PLACM foi preenchida com dados aleatórios, correspondendo a 20760 asserções, 4152 páginas, 22 aplicações/*websites*, 22 entidades, 4 categorias e 243 países. Todos os testes foram realizados ou presencialmente ou remotamente recorrendo à partilha de voz e ecrã por parte do utilizador.

Numa primeira ronda, foram realizados testes com 5 utilizadores, um presencialmente e os restantes através de conferência web, com idades compreendidas entre os 22 e 35 anos, sendo que 3 deles têm formação no campo de informática. Os maiores problemas encontrados, direta ou indiretamente comunicados pelos utilizadores, foram:

- Demora na apresentação de dados relativos a critérios de sucesso;

6. AVALIAÇÃO

- Dificuldade em comparar visualmente dados de múltiplos gráficos, pois as escalas eram diferentes;
- Incapacidade de seleção de todas as opções de dados da classe seleccionada, no separador relacionado ao tipo de navegação 3 da janela modal;
- Dificuldade em concretizar as tarefas iniciais relacionadas com o tipo de navegação 3 (tarefas 9a, 14 e 15).

A dificuldade descrita no último ponto foi mais acentuada nas tarefas 9a, 14 e 15 por serem precisamente as primeiras tarefas relacionadas com o tipo de navegação 3. Para chegar à resposta respetiva à tarefa 9a, o utilizador necessita de escolher um tipo de navegação que nunca usou até este ponto, o *Comparing different categories*. No separador da janela modal correspondente a esta navegação, é-lhe pedido que escolha a classe em que deseja basear os dados (tal como acontece com o *Drilldown*) e, pelo menos, dois dados relativos a essa classe. Esta limitação foi introduzida porque era considerado que o resultado deste tipo de navegação era uma comparação, necessitando assim de mais que um dado.

Os utilizadores, em geral, não conseguiam entender qual seria o resultado deste tipo de navegação ao ler a explicação do mesmo, para além de que ficavam confusos com a necessidade de selecção de dois dados em vez de um. Como resultado, ignoravam este separador da janela modal, por incompreensão de leitura (e, conseqüente, desprezo) ou por tentarem escolher os dados e, à custa da limitação introduzida, interpretarem incorretamente que estavam no separador “errado”.

Apenas um utilizador conseguiu concretizar estas tarefas sem qualquer ajuda, 2 precisaram de ajuda direta e os restantes 2 chegaram ao separador correto por tentativa-erro. Depois de realizarem estas duas tarefas, todos os utilizadores sabiam como usar este tipo de navegação e não apresentaram quaisquer problemas. Contudo, para tornar o uso da plataforma mais intuitivo, foram alterados os títulos, textos de explicação e ordem dos separadores da janela modal, especialmente o separador relativo à navegação 3, que foi renomeado de *Comparing different categories* para *Group by*.

As correcções aos primeiros três pontos da lista anterior basearam-se em alterações nos pedidos à base de dados (explicado na secção abaixo), em igualar as escalas dos gráficos e em acrescentar botões que permitem a selecção instantânea de todas as opções daquele menu. Ainda neste menu, foi removida a necessidade de escolher dois dados, passando a ser necessário escolher apenas um.

Adicionalmente a estes problemas, 4 dos 5 dos utilizadores utilizaram instintivamente as *checkboxes* para comparar dados, em vez de usufruírem do tipo de navegação 4. Após ter sido pedido a um utilizador para realizar a mesma tarefa mas com a utilização da janela modal, este comentou que “com as *checkboxes* é mais rápido, para além de que seleccionar os dados na janela (modal) é mais difícil; a janela é mais pequena”. Apesar

deste comentário, os restantes utilizadores não manifestaram qualquer problema e, como tal, não houve alterações neste tipo de navegação e respetivo separador da janela modal.

Preparadas estas alterações, foram realizados, com a mesma lista de tarefas, testes com mais 2 utilizadores, distintos dos primeiros 5. Esta segunda iteração de testes teve como objetivo perceber se os problemas identificados anteriormente tinham sido mitigados ou totalmente corrigidos. Os utilizadores desta segunda iteração tinham idades de 22 e 24 anos, um deles com estudos na área de informática e outro não.

Estes testes provaram que as alterações na estrutura dos separadores da janela modal foram eficazes em melhorar a usabilidade. Ambos os utilizadores sentiram alguma dificuldade em descobrir o tipo de navegação necessário para realizar tarefa 9a, por ser a primeira utilização do *Group by*, mas a percepção do tipo de navegação necessário nas tarefas 14 e 15 já foi quase instantânea. Com a mitigação deste problema, surgiu o infortúnio de desconsideração pelo tipo de navegação 4; ambos os utilizadores decidiram utilizar exclusivamente as *checkboxes* quando desejavam comparar dados. Curiosamente, ambos os utilizadores mencionaram ainda a necessidade de uma barra de pesquisa para a lista de *checkboxes* ao executar a tarefa 10, algo que nunca foi reportado pelos primeiros 5 utilizadores.

6.2 Desempenho

6.2.1 Queries

O problema de demora na resposta era específico a *queries* relacionadas com o cálculo das métricas dos critérios de sucesso, por ser necessário recolher todas as asserções, agrupá-las por critério de sucesso numa tabela temporária na base de dados, e inferir assim, a partir do resultados das asserções, os resultados dos mesmos.

A solução atual passou por evitar uma modificação adicional de campos da tabela temporária usada para calcular o número de critérios não testados, passando a fazer um pedido inicial para saber a quantidade de critérios de sucesso existentes e realizar uma simples subtração a este valor com a quantidade dos restantes resultados.

Apesar destas *queries* ainda demorarem mais tempo que as relacionadas exclusivamente a asserções, foi possível obter uma melhoria de em média 37% no tempo de resposta, listado na tabela C.3 em anexo. Para a realização deste teste, a base de dados estava preenchida com os mesmos dados aleatórios utilizados nos testes de utilizadores: 20760 asserções, 4152 páginas, 22 aplicações/*websites*, 22 entidades, 4 categorias e 243 países. Os tempos de resposta obtidos estão tabulados no anexo C, onde os cabeçalhos das colunas destas tabelas correspondem à classe a monitorizar (em que os dados se baseiam), enquanto os cabeçalhos horizontais correspondem aos filtros adicionados ao pedido, de modo a complexificar o mesmo. Os filtros são ignorados quando são relativos à classe de dados a monitorizar; nestes casos, todos os dados relativos a essa classe são obtidos e,

6. AVALIAÇÃO

mesmo assim, foi obtida uma melhoria notória.

6.2.2 Processamento *frontend*

Complementando os testes com utilizadores, foram realizados testes de carga para perceber o desempenho da plataforma em situações onde é necessário apresentar simultaneamente uma grande quantidade de informação. Quando era pedida uma grande quantidade de dados, ou quando nesses dados era pedida uma modificação de visualização, a aplicação bloqueava, não apresentando *feedback* algum ao utilizador. Este problema era especialmente notável nas vistas de múltiplos gráficos.

Estudadas as hipóteses, percebeu-se que o problema maior estava relacionado com a API do Highcharts. O Highcharts oferece funções para alterar os dados dos gráficos, sem ter de os criar novamente. Contudo, estas funções acabam por demorar mais tempo, especialmente no processo de *rendering*, do que apagar e criar os gráficos novamente, tendo em atenção a necessidade de manter os elementos HTML onde os gráficos seriam introduzidos.

De modo a evitar este tempo adicional para a recriação dos gráficos, o comportamento das *checkboxes* foi alterado na vista originada pelo tipo de navegação 4, quando os dados estão dispostos em gráficos separados, alterando a visibilidade dos elementos HTML que contêm os gráficos, usando JavaScript. Este processo é instantâneo, tornando esta alteração imperceptível ao utilizador. Apesar disso, não foi possível estender esta alteração para as restantes vistas, porque as *checkboxes* correspondem a dados do gráfico e não aos gráficos em si, sendo assim necessário continuar a recriar os gráficos após a seleção destas.

Como em todas as alterações gráficas, o processo que demorava mais tempo era o de *rendering*, foi ainda testada a incorporação do módulo de *boost*¹ do Highcharts. Este módulo prova-se bastante eficiente quando existem milhares de dados no mesmo gráfico, permitindo a aceleração de *rendering* ao processar apenas as informações fundamentais de cada ponto. No entanto, no caso do PLACM, este módulo tem um resultado contrário, pois desformata os pontos dos gráficos sem mostrar melhorias de tempo no processo de *rendering*. Este resultado é expectável porque as vistas do PLACM a melhorar constam em múltiplos gráficos com pouca informação, ao invés das situações-alvo para a utilização deste módulo: poucos gráficos com muita informação.

¹<https://www.highcharts.com/docs/advanced-chart-features/boost-module>

Capítulo 7

Conclusão

Este trabalho baseou-se no desenho e implementação de uma ferramenta de monitorização de dados de acessibilidade em grande escala, denominada como PLACM, como um dos últimos objetivos do projeto WAI-Tools. Este projeto, cofinanciado pela União Europeia, concretizou dois objetivos que provocaram a utilidade deste trabalho: desenvolvimento de regras ACT, permitindo uma interpretação clara das diretrizes WCAG, e padronização dos resultados da execução de testes de acessibilidade. Esta padronização possibilita uma igual interpretação de resultados de avaliações de diversas fontes e, como tal, a aglomeração e, conseqüente, monitorização desses dados.

O PLACM vem preencher a falta de plataformas de monitorização de acessibilidade, de modo a complementar as inúmeras ferramentas de avaliação já existentes. No contexto e projeto em que foi desenvolvida, o PLACM contactou com os parceiros do WAI-Tools para, com a ambição de chegar a todos, perceber quais os requisitos e pareceres dos mesmos. De forma a ser facilmente acessível e partilhável, o PLACM manifestou-se numa aplicação web, baseada em *dashboards* de gráficos tais como outras plataformas de monitorização existentes.

A informação a ser apresentada nesta plataforma baseou-se em dois tipos de documentos, as declarações de acessibilidade, para obter informação relativa às entidades ou aplicações a monitorizar, e os relatórios EARL, para obter dados sobre os resultados das avaliações realizadas. Os dados inferidos destes documentos foram agrupados por classes, desde ‘Continente’ até ‘Regra’, de modo a estruturar uma hierarquia baseada em granularidade de informação e, conseqüentemente, possibilitar uma navegação pelas mesmas.

Para que fosse possível uma avaliação e estudo detalhado sobre a conformidade das aplicações a monitorizar, foram introduzidos conjuntos de métricas baseadas em asserções, bases fundamentais dos relatórios de formato padronizado de resultados de testes de acessibilidade, e critérios de sucesso, enunciados de como o conteúdo digital deve estar estruturado ou reagir à interação do utilizador. Estas métricas foram aplicadas em gráficos de barras e barras empilhadas para facilitar a leitura e comparação de, possivelmente, milhares de dados.

7. CONCLUSÃO

A fim de navegar por toda esta informação, foram desenvolvidos seis diferentes tipos de navegação:

- Navegação para os dados de uma classe, que consta na visualização dos dados totais de uma classe;
- Navegação para “descer” na hierarquia, onde, filtrando através da seleção de dados, é possível especificar informação a monitorizar;
- Navegação para “subir” na hierarquia, que permite realizar uma comparação de dados tendo em conta uma seleção de dados de uma classe inferior na hierarquia;
- Comparação no mesmo nível de hierarquia, onde a visualização originada permite a comparação direta de uma seleção de dados da mesma classe;
- Linha temporal, de modo a estudar a evolução temporal de uma classe;
- Conformidade dos critérios de sucesso numa aplicação, que se baseia numa listagem dos resultados dos critérios de sucesso aplicados numa aplicação e as asserções que determinaram esses resultados.

Como alguns destes tipos de navegação necessitam de algum esforço para compreender o seu funcionamento, foram realizados testes, em duas etapas, para perceber o nível de usabilidade da plataforma. Ao fim da primeira etapa, foram reconhecidos alguns problemas, o que levou a uma iteração na nomenclatura e estrutura do PLACM e na melhoria de desempenho da mesma.

Esta plataforma obteria uma maior utilidade ao ser aplicada publicamente e tornando-a responsabilidade de uma entidade mundial competente, de modo a permitir a monitorização e aglomeração de dados mundiais e, conseqüente, emulação entre entidades. Contudo, com a improbabilidade desta hipótese, esta plataforma pode ser usada para proveito próprio das entidades que desejam monitorizar a sua acessibilidade. Na altura da escrita desta dissertação, a Accessibility Foundation, parceiro holandês do WAI-Tools, estava a considerar a implementação desta plataforma para emprego próprio.

7.1 Contribuições e limitações

O desenvolvimento do PLACM baseou-se em alguns requisitos levantados através do estudo de plataformas já existentes e de alguns pareceres das entidades parceiras responsáveis por observatórios de acessibilidade nacionais, mesmo depois de ter sido partilhada a indispensabilidade de comentários por parte de todos os parceiros do WAI-Tools. A pobre participação dos parceiros, exceptuando a AMA, mostrou-se um obstáculo na evolução do PLACM, ao ser perdida a partilha direta de conhecimento de especialistas de acessibilidade. Porém, no início de outubro, foi realizada uma demonstração do PLACM,

7.1 Contribuições e limitações

juntamente com outros resultados do projeto WAI-Tools, a 120 especialistas, que apresentaram palavras de apreço em relação à funcionalidade do mesmo.

Um dos maiores interesses no desenvolvimento de uma plataforma de monitorização de dados de acessibilidade, é, igualmente, a sua acessibilidade, sobretudo ao ser baseada no cenário volúvel de *dashboards* gráficos. A *framework Highcharts* foi utilizada por fornecer um módulo dedicado a acessibilidade, mas que apresentava ainda grandes limitações para os requisitos do PLACM, tais como apresentar e interagir com vários gráficos simultaneamente. Devido à falta de suporte nativo da *framework* para os problemas encontrados, estes exigiram um esforço adicional, através da captura de eventos e modificação do documento HTML, para que fossem corretamente resolvidos.

Outro cuidado considerado no desenvolvimento do PLACM, juntamente com tornar toda a plataforma acessível, foi também a possibilidade de partilhar todas as visualizações. Todas as escolhas do utilizador, como os tipos de navegação realizados e as legendas e pontos dos gráficos visíveis, são traduzidas no URL de modo a que, especialmente importante nas realizações e publicações de estudos, todas as visualizações sejam partilháveis.

Como o PLACM se encontra no contexto do projeto WAI-Tools e existe um esforço contínuo de desenvolvimento de regras ACT, é necessário que esta informação seja corretamente mantida na base de dados do PLACM, para preservar a coerência dos dados. Assim sendo, os administradores das implementações do PLACM devem ter em mente a manutenção desta informação, especialmente nas tabelas relativas às regras e, consequentemente, tipos de elementos avaliados. Esta limitação tem de ser corretamente tratada pois a informação a guardar, apesar de estática, correlaciona todos os dados da plataforma.

Uma outra limitação do PLACM é a consideração das asserções mais recentes no cálculo das métricas. A implementação atual considera as asserções mais recentes por tuplo [critério, página, avaliador], onde o critério está relacionado com o *TestCriterion*. O avaliador é considerado para evitar que as asserções do tuplo [critério, página] sejam facilmente manipuladas por um avaliador malicioso. Neste último caso, um avaliador poderia gerar apenas asserções passadas para que os critérios de sucesso resultassem sempre positivos. Infelizmente, com o tuplo considerado na implementação, também persiste o problema de existir um avaliador malicioso que, inversamente ao anterior, basta criar uma asserção falhada e inferir a falha de critérios de sucesso. Ambos os tuplos estudados trazem o problema de manipulação de dados, mas foi considerado o segundo por originar mais facilmente falsos positivos, oposto a falsos negativos; foi considerado preferível declarar que o conteúdo digital não é acessível quando o é do que declarar que o conteúdo é acessível quando não o é.

Uma alternativa possível para a implementação atual seria um sistema de votação, de modo a ditar o resultado de um critério de sucesso considerando, por exemplo, o resultado que existe na maioria das asserções mais recentes do tuplo [critério, página, avaliador], de modo a resolver o problema de consenso e, consequentemente, diminuir o impacto de avaliadores maliciosos.

7. CONCLUSÃO

Complementando esta alternativa, poderia ainda ser acrescentada uma limitação temporal à escolha das asserções, isto é, em vez de serem consideradas as asserções mais recentes do tuplo [critério, página, avaliador], seriam apenas consideradas as asserções mais recentes mas com menos de, por exemplo, 2 anos. Esta limitação temporal, que poderia ser alterável entre implementações do PLACM, garantia uma avaliação periódica dos dados a monitorizar e, conseqüentemente, a uma atualização e manutenção constante dos dados.

7.2 Trabalho futuro

Depois de concluído o desenvolvimento do projeto, os requisitos do mesmo foram implementados mas ainda ficam algumas melhorias para futuras versões do PLACM. Algumas destas são:

- Otimização do código - reestruturação e simplificação do código escrito para um melhor desenvolvimento futuro;
- Otimização do desempenho - apesar de terem sido realizadas algumas operações durante o desenvolvimento, ainda há espaço para melhorar na geração e modificação dos gráficos;
- Melhoria de documentação - complementando o primeiro ponto, para que seja facilitada a interpretação do código para futuros desenvolvedores;
- Implementação de autenticação - para que seja possível existir múltiplos administradores responsáveis pela adição e manutenção de informação.

Bibliografia

- [1] CEN/CENELEC, ETSI, “EN 301 549 v3.1.1,” Acedido em: 19/11/2020. [Online]. Available: https://www.etsi.org/deliver/etsi_en/301500_301599/301549/03.01.01_60/en_301549v030101p.pdf
- [2] Presidência do Conselho de Ministros, “Decreto-Lei n.º 83/2018 de 19 de outubro,” Acedido em: 19/11/2020. [Online]. Available: <https://dre.pt/application/conteudo/116734769>
- [3] Parlamento Europeu, “Diretiva (UE) 2016/2102 do Parlamento Europeu e do Conselho, de 26 de outubro de 2016, relativa à acessibilidade dos sítios web e das aplicações móveis de organismos do setor público,” Acedido em: 19/11/2020. [Online]. Available: <https://eur-lex.europa.eu/legal-content/PT/TXT/PDF/?uri=CELEX:32016L2102>
- [4] W3C, “Web Content Accessibility Guidelines (WCAG) 2.1,” Acedido em: 19/11/2020. [Online]. Available: <https://www.w3.org/TR/WCAG21/>
- [5] ACT Rules Community Group, “About Us,” Acedido em: 20/11/2020. [Online]. Available: <https://act-rules.github.io/pages/about/>
- [6] W3C Working Group, “Evaluation and Report Language (EARL) 1.0 Schema,” Acedido em: 20/11/2020. [Online]. Available: <https://www.w3.org/TR/EARL10-Schema/>

Apêndice A

Relação entre regras ACT e tipos de elementos avaliados

Regra	Nome	Tipo de elemento
ff89c9	ARIA required context role	ARIA attributes
bc4a75	ARIA required owned elements	ARIA attributes
6a7281	ARIA state or property has valid value	ARIA attributes
5c01ea	ARIA state or property is permitted	ARIA attributes
5f99a7	aria-* attribute is defined in WAI-ARIA	ARIA attributes
6cfa84	Element with 'aria-hidden' has no focusable content	ARIA attributes
4e8ab6	Element with 'role' attribute has required states and properties	ARIA attributes
674b10	'role' attribute has valid value	ARIA attributes
2ee8b8	Visible label is part of accessible name	ARIA attributes
e6952f	Attribute is not duplicated	Attributes
3ea0c8	'id' attribute value is unique	Attributes
e7aa44	audio element content has text alternative	Audio
2eb176	audio element content has transcript	Audio
afb423	audio element content is media alternative for text	Audio
80f0bf	audio or video avoids automatically playing audio	Audio, Video
4c31df	audio or video that plays automatically has a control mechanism	Audio, Video
aaa1bf	audio or video that plays automatically has no audio that lasts more than 3 seconds	Audio, Video
97a4e1	Button has non-empty accessible name	Button
46ca7f	Element marked as decorative is not exposed	Decorative
80af7b	Focusable element has no keyboard trap	Focusable
ebe86a	Focusable element has no keyboard trap via non-standard navigation	Focusable

Tabela A.1: Relações entre as regras ACT e os tipos de elemento avaliados

A. RELAÇÃO ENTRE REGRAS ACT E TIPOS DE ELEMENTOS AVALIADOS

Regra	Nome	Tipo de elemento
a1b64e	Focusable element has no keyboard trap via standard navigation	Focusable
oj04fd	Element in sequential focus order has visible focus	Focusable
36b590	Error message describes invalid form field value	Form fields
e086e5	Form control has non-empty accessible name	Form fields
cc0f0a	Form control label is descriptive	Form fields
ffd0e9	Heading has non-empty accessible name	Heading
b49b2e	Heading is descriptive	Heading
b33eff	Orientation of the page is not restricted using CSS transform property	HTML Document
7677a9	Device motion based changes to the content can also be created from the user interface	HTML Document
c249d5	Device motion based changes to the content can be disabled	HTML Document
ffbc54	No keyboard shortcut uses only printable characters	HTML Document
cae760	'iframe' element has non-empty accessible name	iframe
4b1c6c	'iframe' elements with identical accessible names have equivalent purpose	iframe
23a2a8	Image has non-empty accessible name	Image
e88epe	Image not in the accessibility tree is decorative	Image
9eb3f6	Image filename is accessible name for image	Image
7d6734	'svg' element with explicit role has non-empty accessible name	Image
qt1vmo	Image accessible name is descriptive	Image
59796f	Image button has non-empty accessible name	Image, Button
73f2c2	'autocomplete' attribute has valid value	Input
de46e4	Element with 'lang' attribute has valid language tag	lang attributes
b5c3f8	HTML page has 'lang' attribute	lang attributes
5b7ae0	HTML page 'lang' and 'xml:lang' attributes have matching values	lang attributes
bf051a	HTML page 'lang' attribute has valid language tag	lang attributes
ucwvc8	HTML page language subtag matches default language	lang attributes
c487ae	Link has non-empty accessible name	Link
5effbb	Link in context is descriptive	Link
fd3a94	Links with identical accessible names and context serve equivalent purpose	Link
b20e66	Links with identical accessible names have equivalent purpose	Link

Tabela A.2: Relações entre as regras ACT e os tipos de elemento avaliados (*continuação*)

Regra	Nome	Tipo de elemento
bc659a	'meta' element has no refresh delay	Meta
b4f0c3	'meta' viewport allows for zoom	Meta
8fc3b6	Object element rendering non-text content has non-empty accessible name	Object
0ssw9k	Scrollable element is keyboard accessible	Scrollable
a25f45	'headers' attribute specified on a cell refers to cells in the same table element	Tables
d0f69e	Table header cells have assigned data cells	Tables
afw4f7	Text has minimum contrast	Text
59br37	Zoomed text node is not clipped with CSS overflow	Text
efbc7	Text content that changes automatically can be paused, stopped or hidden	Text
2779a5	HTML page has non-empty title	Title
c4a8a4	HTML page title is descriptive	Title
eac66b	'video' element auditory content has accessible alternative	Video
f51b46	'video' element auditory content has captions	Video
ab4d13	'video' element content is media alternative for text	Video
c5a4ea	'video' element visual content has accessible alternative	Video
1ea59c	'video' element visual content has audio description	Video
f196ce	'video' element visual content has description track	Video
1ec09b	'video' element visual content has strict accessible alternative	Video
1a02b0	'video' element visual content has transcript	Video
c3232f	'video' element visual-only content has accessible alternative	Video
d7ba54	'video' element visual-only content has audio track alternative	Video
ac7dc6	'video' element visual-only content has description track	Video
ee13b5	'video' element visual-only content has transcript	Video
fd26cf	'video' element visual-only content is media alternative for text	Video

Tabela A.3: Relações entre as regras ACT e os tipos de elemento avaliados (2ª continuação)

Apêndice B

Lista de perguntas usada nos testes com utilizadores

1. Quantas asserções passaram no tipo de elemento ‘table’?
2. O PLACM já estuda dados de quantos países europeus?
3. Quais as categorias estudadas no país de Malta?
4. Quais foram as ferramentas de avaliação usadas para avaliar a aplicação ‘Leo’?
5. Qual foi o tipo de elemento avaliado em mais páginas da aplicação ‘Leo’?
6. Quais foram as ferramentas de avaliação usadas para avaliar o setor privado da categoria de saúde do país de Malta?
7. Quantas são as regras mapeadas ao critério de sucesso ‘1.4.2 – Audio Control’?
8. Qual foi a aplicação com mais critérios de sucesso aprovados?
9.
 - (a) Qual é o sector que apresenta mais asserções falhadas em relação ao tipo de elemento ‘button’?
 - (b) E apenas em relação ao sector privado, qual o tipo de elemento que tem mais asserções ‘cantTell’: ‘audio’, ‘button’, ‘css’ ou ‘form’?
10. Entre as regras ‘Attribute is not duplicated’ e ‘Text has minimum contrast’, qual é a que originou mais asserções inaplicáveis?
11. Em relação à organização ‘Cattle Org.’, qual foi o critério de sucesso que foi avaliado em mais páginas dessa organização: ‘1.3.4 – Orientation’ ou ‘1.3.5 – Identify Input Purpose’?
12. Qual foi a ferramenta de avaliação usada pela única aplicação croata?

B. LISTA DE PERGUNTAS USADA NOS TESTES COM UTILIZADORES

13. Qual é o sector australiano com mais critérios de sucesso passados?
14. Qual é a organização que apresenta mais asserções passadas usando a ferramenta de avaliação 'Earth'?
15. Entre o continente europeu e africano, qual é o que apresenta um maior número de asserções inaplicáveis em relação à regra 'All table header cells have assigned data cells'?
16. Qual é a aplicação que, usando a ferramenta de avaliação 'Jupiter' para correr a regra 'HTML page has a title', obteve um maior número de asserções falhadas?
17. Quais são as categorias da aplicação 'Hydrus'?
18. Quantas asserções passaram quando a aplicação Hydrus foi avaliada no mês de agosto de 2020?
19. Ainda em relação à aplicação Hydrus, qual foi o mês onde as avaliações originaram menos critérios de sucesso falhados?
20. Qual o mês onde foram avaliadas mais páginas usando a ferramenta de avaliação 'Earth'?

Apêndice C

Avaliação do tempo de resposta dos pedidos relacionados com as métricas de critérios de sucesso

Filtro usado	Tempo de resposta por classe (ms)					
	Cont.	País	Cat.	Sector	Org.	Apl.
continentIds=1,2,3	3417	3291	3183	3215	3173	3300
countryIds=27,51,139	1019	1203	1035	1040	1001	1037
tagIds=1,2,3	3755	3752	3613	3810	3815	3929
sectorIds=1,2	-	-	2420	2451	2441	2580
orgIds=1,2,3	-	-	-	1010	979	1024
appIds=1,2,3	-	-	-	991	989	1169

Tabela C.1: Tempo de resposta dos pedidos de dados respetivos às classes relacionadas com as métricas de critérios de sucesso, antes das alterações

Filtro usado	Tempo de resposta por classe (ms)					
	Cont.	País	Cat.	Sector	Org.	Apl.
continentIds=1,2,3	1936	1987	1963	1984	2130	1998
countryIds=27,51,139	794	574	765	593	759	776
tagIds=1,2,3	2277	2271	2225	2284	2272	2230
sectorIds=1,2	-	-	1579	972	1648	1572
orgIds=1,2,3	-	-	-	605	580	850
appIds=1,2,3	-	-	-	625	771	565

Tabela C.2: Tempo de resposta dos pedidos de dados respetivos às classes relacionadas com as métricas de critérios de sucesso, após alterações

C. AVALIAÇÃO DO TEMPO DE RESPOSTA DOS PEDIDOS RELACIONADOS COM AS MÉTRICAS DE CRITÉRIOS DE SUCESSO

Filtro usado	Diminuição de tempo por classe (%)					
	Cont.	País	Cat.	Sector	Org.	Apl.
continentIds=1,2,3	43.34	39.62	38.33	38.29	32.87	39.45
countryIds=27,51,139	22.08	52.29	26.09	42.98	24.18	25.17
tagIds=1,2,3	39.36	39.47	38.42	40.05	40.45	43.24
sectorIds=1,2	-	-	34.75	60.34	32.49	39.07
orgIds=1,2,3	-	-	-	40.10	40.76	16.99
appIds=1,2,3	-	-	-	36.93	22.04	51.67

Tabela C.3: Diminuição percentual do tempo de resposta dos pedidos de dados respetivos às classes relacionadas com as métricas de critérios de sucesso, após alterações