

UNIVERSIDADE DE SÃO PAULO  
INSTITUTO DE MATEMÁTICA E ESTATÍSTICA  
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

**Ferramenta de Gestão de Selos  
de Acessibilidade na Web**

Yannick Thomas Messias

**MONOGRAFIA FINAL  
MAC 499 – TRABALHO DE  
FORMATURA SUPERVISIONADO**

Supervisor: Prof. Dr. Carlos Hitoshi Morimoto

São Paulo  
20 de novembro de 2019



# Agradecimentos

Gostaria de agradecer ao meu supervisor, Prof. Carlos Hitoshi Morimoto, por toda a ajuda e acompanhamento fornecidos durante todo o decorrer deste trabalho; aos meus colegas de faculdade Leonardo Padilha, a quem credito por grande parte do design e interação do sistema, e Pedro Sola Pimentel, grande parceiro de graduação; por fim agradeço aos meus pais, Sérgio Messias e Marthe Thomas Messias, e à minha namorada, Thais Waughan, por acreditarem em mim e me darem todo o apoio que eu poderia precisar.

Dedico este trabalho a todos que me acompanharam no meu trajeto até aqui.



## Resumo

Yannick Thomas Messias. **Ferramenta de Gestão de Selos de Acessibilidade na Web.** Monografia (Bacharelado). Instituto de Matemática e Estatística, Universidade de São Paulo, São Paulo, 2019.

Acessibilidade é uma preocupação cada vez maior por parte de qualquer organização que preze pela inclusão e, em particular, a inclusão digital. Existem diversos desafios no que concerne a fornecer acessibilidade no meio digital e na internet. Visando auxiliar desenvolvedores interessados em fornecer acessibilidade em sítios da internet, foram criadas diretrizes de acessibilidade com princípios e critérios testáveis para avaliar o nível de acessibilidade fornecida pelos sítios. Existem também ferramentas de avaliação para auxiliar os desenvolvedores verificando parte desses critérios total ou parcialmente de forma automatizada. A partir de avaliações de uma dessas ferramentas, a Secretaria Municipal da Pessoa com Deficiência da prefeitura de São Paulo avalia páginas da web e fornece selos de acessibilidade digital. Para tal, são realizadas avaliações periódicas das páginas e é feito um acompanhamento para verificar se o nível de acessibilidade mínimo requerido pela secretaria é mantido, sujeito a perda do selo caso contrário. Esse processo é bastante trabalhoso e é realizado de forma manual pelos funcionários da secretaria. Este trabalho apresenta um sistema informatizado para auxiliar esses funcionários nesse processo, automatizando parte dele e fornecendo uma interface unificada e intuitiva para o cadastro de selos e para fácil visualização do estado do sistema. São discutidos detalhes das decisões de projeto e da estrutura da implementação. Este trabalho também visa funcionar como ponto de partida para outros desenvolvedores se inteirarem no sistema e implementarem modificações ou fazerem novas adições ao sistema.

**Palavras-chave:** Acessibilidade. Internet. Avaliação. Certificado. Selo. Sistemas. IME/USP.



# **Abstract**

Yannick Thomas Messias. **Web Accessibility Certifications Managing Tool.** Capstone Project Report (Bachelor). Institute of Mathematics and Statistics, University of São Paulo, São Paulo, 2019.

Accessibility is a growing concern for any organization that seeks inclusion and, in particular, digital inclusion. There are several challenges in providing accessibility in the digital environment and on the internet. To assist developers interested in providing accessibility on websites, accessibility guidelines have been created with principles and testable criteria to evaluate the level of accessibility provided by websites. There are also evaluation tools to assist developers by verifying some of these criteria in an automated or partly-automated manner. Based on evaluations of one of these tools, the Municipal Secretariat of the Disabled for the city of São Paulo evaluates web pages and provides digital accessibility certifications. To this end, periodic page evaluations are carried out and follow-up is performed to verify that the minimum accessibility level required by the secretariat is maintained, subject to the loss of the certification otherwise. This process is quite laborious and is done manually by the secretariat staff. This paper presents a computerized system to assist these employees in this process by automating part of it and providing a unified and intuitive interface for certification registration and easy viewing of system status. Details of design decisions and implementation structure are discussed. This work is also intended to function as a starting point for new developers to get into the system and develop modifications or make new additions to it.

**Keywords:** Accessibility. Internet. Evaluation. Certification. Seal. Systems. IME/USP.



# Sumário

<b>1</b>	<b>Introdução</b>	<b>1</b>
1.1	Acessibilidade na Web . . . . .	1
1.2	Ferramentas de Avaliação de Acessibilidade Digital . . . . .	2
1.3	Selos de Acessibilidade Digital da SMPED . . . . .	3
1.4	Proposta . . . . .	3
1.4.1	Motivação . . . . .	3
1.4.2	Objetivo . . . . .	4
<b>2</b>	<b>Decisões de Desenvolvimento</b>	<b>5</b>
2.1	Metodologia . . . . .	5
2.2	Tecnologias . . . . .	5
2.2.1	Git e GitHub . . . . .	5
2.2.2	Python . . . . .	6
2.2.3	Django . . . . .	6
2.2.4	HTML . . . . .	6
2.2.5	CSS . . . . .	6
2.2.6	JavaScript . . . . .	7
2.2.7	PostgreSQL . . . . .	7
2.2.8	Scrapy . . . . .	7
2.2.9	AWS . . . . .	7
2.3	Decisões de Projeto . . . . .	8
2.3.1	ASES-Web vs ASES-Desktop . . . . .	8
2.3.2	Funcionalidades . . . . .	8
2.3.3	Código . . . . .	9
<b>3</b>	<b>Estrutura da Implementação</b>	<b>11</b>
3.1	Módulos . . . . .	11
3.1.1	Configurações do Django . . . . .	11
3.1.2	Contas de Usuário . . . . .	11

3.1.3	Requerentes . . . . .	12
3.1.4	Selos . . . . .	12
3.1.5	Relatórios . . . . .	12
3.1.6	Web Crawler . . . . .	12
<b>4</b>	<b>Conclusão</b>	<b>15</b>
4.1	Resultados . . . . .	15
4.2	Próximos passos . . . . .	15
	<b>Referências</b>	<b>17</b>

# Capítulo 1

## Introdução

Acessibilidade é a qualidade daquilo que é acessível e diz respeito, em geral, ao projeto de produtos, dispositivos, serviços ou ambientes que possam ser usufruídos por uma ampla gama de usuários e, em particular, pessoas com alguma deficiência ([WIKIPEDIA, 2019](#)). Além de deficientes, que abrangem diversos níveis, incluindo deficiência visual, auditiva, física, cognitiva e neurológica, o público alvo no estudo da acessibilidade também inclui pessoas com problemas provindos de idade avançada ou mesmo classe social. Cada vez mais a acessibilidade vem se tornando uma preocupação, tanto para instituições públicas, imprescindíveis de fornecer serviços acessíveis a toda a população, quanto para instituições privadas, visando não somente uma boa imagem mas também atender uma parcela cada vez maior de clientes. Acima de tudo, pensar em acessibilidade hoje é uma responsabilidade moral de toda sociedade, interessada na inclusão democrática de todos os seus cidadãos.

A acessibilidade se dá em diversos meios e de diversas formas. É comum pensar em algo acessível do ponto de vista físico, como na presença de rampas ou elevadores em locais onde a princípio só haveriam escadas, visando o acesso de pessoas com deficiências físicas e dificuldade de locomoção. Mas a questão da acessibilidade se faz presente nos mais diversos contextos, sempre que há a necessidade de pensar em formas inovadoras de fornecer o acesso e incluir uma população que de outra forma seria excluída. Neste trabalho vamos considerar o problema de fornecer acessibilidade no contexto de sistemas digitais.

### 1.1 Acessibilidade na Web

No contexto de sistemas digitais e sítios da internet, o estudo e implementação da acessibilidade visa tornar os conteúdos presentes nestes recintos universalmente acessíveis. Isso consiste "na prática inclusiva de fazer websites que possam ser utilizados por todas as pessoas, mesmo as que não são detentoras de deficiência, englobando conteúdos gráficos e sonoros alternativos, claros, compreensíveis e capazes de garantir o controle da navegação pelos usuários, independente das suas capacidades físico-motoras e perceptivas, culturais e sociais" ([MÜLLER e CORRÊA, 2017](#)). Para atingir alguns públicos, como por exemplo o de

deficientes visuais, são utilizados navegadores especiais e outras tecnologias assistivas, como leitores de tela. Para que essas tecnologias possam funcionar corretamente, é preciso que o sítio esteja bem estruturado.

Para avaliar se os sítios estão condizentes com o que se espera em termos de acessibilidade foram desenvolvidas diretrizes especiais de acessibilidade. O maior exemplo disso é o documento WCAG (Web Content Accessibility Guidelines), mantido pela WAI (Web Accessibility Initiative), uma subdivisão do W3C (World Wide Web Consortium). Esse documento contém princípios, recomendações e diretrizes testáveis para verificar e avaliar o nível de acessibilidade de um sítio<sup>1</sup>.

No Brasil, o governo federal utiliza um documento próprio com diretrizes de acessibilidade digital chamado e-MAG (Modelo de Acessibilidade em Governo Eletrônico). O e-MAG foi desenvolvido pelo DGE (Departamento de Governo Eletrônico) e a sua versão mais recente foi baseada no WCAG 2.0<sup>2</sup>. Desde 2007 tornou-se compulsório a todas as páginas do governo federal que se adequem às diretrizes do e-MAG.

## 1.2 Ferramentas de Avaliação de Acessibilidade Digital

Para auxiliar na avaliação de páginas web com respeito à acessibilidade, existem ferramentas de avaliação de acessibilidade digital. Essas ferramentas verificam critérios testáveis, como os do WCAG, de forma automatizada. Vale ressaltar que isso nem sempre é possível, alguns critérios, embora testáveis, dependem fortemente de verificação humana. Nesse caso, quando possível, a ferramenta pode indicar ao usuário que parte do código requer atenção para verificação do critério. No atual estado da arte, estamos muito distantes de conseguir realizar uma avaliação totalmente automatizada<sup>3</sup>.

Um exemplo de ferramenta de avaliação de acessibilidade digital é o ASEs (Avaliador e Simulador de Acessibilidade em Sítios). O ASEs<sup>4</sup> é uma ferramenta desenvolvida pelo DGE em parceria com o IFRS (Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Sul). Com essa ferramenta é possível avaliar a acessibilidade de sítios de acordo com as recomendações do eMAG. Em sua funcionalidade principal, o ASEs recebe a URL (Uniform Resource Locator) de uma página na web, faz uma avaliação da respectiva página, e devolve uma nota em porcentagem junto a um relatório que indica as falhas encontradas.

---

<sup>1</sup>Mais sobre o WCAG pode ser encontrado em <https://www.w3.org/TR/WCAG21/>

<sup>2</sup>Mais sobre o e-MAG pode ser encontrado em <http://emag.governoeletronico.gov.br/>

<sup>3</sup>Como descrito no blog da empresa de acessibilidade UsableNet (PETRI e FEDERICI, 2018)

<sup>4</sup>Disponível em [softwarepublico.gov.br/social/ases](http://softwarepublico.gov.br/social/ases)

## 1.3 Selos de Acessibilidade Digital da SMPED

A fim de atestar o nível de acessibilidade de alguns sítios da internet, a SMPED (Secretaria Municipal da Pessoa com Deficiência) fornece um selo de acessibilidade digital a empresas interessadas e entidades públicas. Essas empresas ou entidades solicitam o selo com a SMPED para um determinado domínio e um número limitado de páginas. Após análise de todas as páginas, caso atinjam o nível esperado de acessibilidade, é fornecido o selo ao requerente por um dado período, até renovação. Para fazer essa análise é utilizada a ferramenta de avaliação ASEs.

Como páginas da web são voláteis, é necessário fazer uma verificação periódica de todas as URLs associadas a um selo ativo, de forma a verificar se as páginas ainda cumprem os requisitos do selo. Esse processo é realizado manualmente todos os dias pelos funcionários da SMPED. As URLs são submetidas uma a uma à ferramenta ASEs-Web<sup>5</sup>, que dá uma nota e gera um relatório, que é então salvo como registro e enviado ao requerente do selo caso a nota fique abaixo do mínimo exigido. Caso alguma página associada a um selo ativo se mantenha por um longo período com nota abaixo do exigido o respectivo selo é então revogado.

## 1.4 Proposta

A proposta deste trabalho é apresentar uma alternativa ao modus operandi dos funcionários responsáveis pela administração dos selos de acessibilidade digital da SMPED. O que se propõe é o desenvolvimento de uma ferramenta digital para cadastro de selos e que realize as verificações periódicas das URLs de forma automatizada, mantendo também os relatórios de forma organizada e apresentando o estado do sistema de forma visualmente clara e intuitiva.

### 1.4.1 Motivação

O número de URLs que precisam ser submetidas, manualmente e todos os dias, ao ASEs-Web é razoável. Há ainda um esforço a mais para baixar e manter organizados os relatórios de todas as avaliações. Além disso, caso seja constatado que uma página não atingiu os requisitos mínimos exigidos, é preciso checar manualmente os relatórios de N datas passadas para aquela mesma página. No caso dos resultados ficarem consistentemente abaixo do exigido, a SMPED emite um alerta via email para o requerente do selo para que o problema seja resolvido. Depois de futuras análises, caso o resultado persista abaixo do exigido, aquele selo é então revogado.

Esse processo manual é muito trabalhoso e pouco eficiente, consumindo muito tempo e esforço para uma função que poderia ser feita de forma automatizada. A criação de uma ferramenta que automatize parte desse processo forneceria um ganho bastante expressivo

---

<sup>5</sup>Uma versão web do ASEs. Disponível em <http://asesweb.governoeletronico.gov.br/ases/>.

de produtividade, além de minimizar possíveis erros humanos e em última instância simplificar e facilitar o trabalho realizado pelos funcionários da SMPED. Também é interessante que haja um ambiente unificado para manter o cadastro de todas as informações pertinentes ao selo, de forma a fornecer organização e fácil acesso, bem como gerenciar os próprios funcionários que possuem acesso ao sistema.

### **1.4.2 Objetivo**

O objetivo do trabalho é então desenvolver tal ferramenta, que permita o cadastro e gestão de usuários, com a devida autenticação e níveis de permissão, requerentes do selo de acessibilidade digital e seus respectivos selos concedidos, além de realizar a checagem periódica das páginas dos selos automaticamente. O sistema deve ser simples de usar e possuir uma interface que favoreça a visualização do estado do sistema a fim de facilitar a tomada de decisões.

# Capítulo 2

## Decisões de Desenvolvimento

Neste capítulo são apresentados detalhes referentes ao desenvolvimento, incluindo metodologias empregadas, as tecnologias escolhidas para implementação, até decisões de projeto referentes a o que seria implementado, como e por que.

### 2.1 Metodologia

Durante todo desenvolvimento do projeto foi dada muita importância à qualidade do código e às soluções utilizadas, priorizou-se a manutenibilidade e a elegância. Por conta disso grande parte do tempo foi investido em estudos para construir um conhecimento sólido das tecnologias utilizadas e promover o uso das soluções mais "limpas" disponíveis. Além disso, muito esforço foi colocado na fase de planejamento, para identificar as melhores soluções, economizar tempo de implementação e principalmente minimizar mudanças custosas. Também foram realizadas diversas conversas e discussões junto ao supervisor para garantir que se mantivesse sempre em mente os propósitos do projeto e para que se criticasse o que tivesse sido desenvolvido até ali, bem como os passos seguintes. Foi utilizado prototipação para desenvolver ideias rapidamente e testar possibilidades.

### 2.2 Tecnologias

#### 2.2.1 Git e GitHub

O Git é um sistema de controle de versões, com ele é possível gerir uma árvore de modificações de todos os arquivos do projeto. Foi utilizado desde o início do trabalho para documentar o histórico de atualizações e para, eventualmente, quando necessário, facilmente retornar a um estado anterior à alguma mudança identificada como indesejada.

O GitHub é uma ferramenta web onde é possível gerir um repositório git na nuvem. Também foi utilizado desde o início junto ao git para tornar o repositório acessível indepen-

dente de um único local, permitindo um fácil acesso ao código e ao andamento do trabalho. A utilização do Git com GitHub agilizou o desenvolvimento e serviu para o supervisor ter acesso irrestrito ao código atualizado, além de, em última instância, manter o código acessível para outros desenvolvedores.

### 2.2.2 Python

Python é uma linguagem de programação que tem como principais características ser uma linguagem de script, interpretada, de alto nível, com tipagem dinâmica e orientada a objetos. A linguagem Python foi escolhida para todo o desenvolvimento do trabalho por ter uma boa legibilidade, afinidade prévia e por ser ágil e bastante popular, possuindo uma ampla comunidade e diversas ferramentas desenvolvidas, como veremos a seguir.

### 2.2.3 Django

Django é um framework para desenvolvimento de ferramentas para web. Foi escolhido como base para o sistema por ser inteiramente escrito em Python e possuir funções bem poderosas e completas de antemão, fornecendo grande agilidade ao desenvolvimento. O Django utiliza o padrão MTV (Model-Template-View) e o princípio DRY (Don't Repeat Yourself) que mantém o código bem organizado e aproveita ao máximo o código escrito, respectivamente.

### 2.2.4 HTML

HTML (HyperText Markup Language) é uma linguagem de marcação utilizada na criação de páginas na web. O HTML define a estrutura de uma página, contendo todo o hipertexto<sup>1</sup> e a forma como os elementos da página se estruturam dentro dela. O código HTML pode ser gerado estática ou dinamicamente, no nosso caso parte do código é estático e parte é gerado dinamicamente pelo sistema de template do Django. A estrutura base da ferramenta, comum a todas as páginas, e dos formulários é código estático, enquanto que o conteúdo, resgatado do banco de dados do sistema ou de informações da sessão, é dinâmico. Por conter a estrutura da página, o HTML é frequentemente o principal foco das avaliações de acessibilidade, especialmente avaliações automatizadas.

### 2.2.5 CSS

CSS (Cascading Style Sheets) é uma linguagem usada para definir aspectos visuais da apresentação de documentos escritos em uma linguagem de marcação como o HTML. Com CSS é possível fazer uma separação entre a apresentação visual e o conteúdo do documento, inclusive em arquivos separados. No projeto optou-se por trabalhar dessa forma, com um

---

<sup>1</sup>Hipertexto consiste de texto agregado a outras informações, como fonte tipográfica, cor, tamanho, itálico e negrito, mas também imagens, áudios, vídeos e, principalmente, hiperlinks.

arquivo exclusivo para o CSS, porém permitiu-se incluir CSS junto ao HTML desde que a propriedade incluída fosse exclusiva da página em questão, mantendo-se o princípio DRY.

## 2.2.6 JavaScript

JavaScript é uma linguagem de programação de script e interpretada assim como o Python. É também a linguagem de programação de uso padrão na web, sendo interpretada pelos navegadores do lado do cliente, sem necessidade de requisição ao servidor. No projeto foi utilizado JavaScript pontualmente em algumas páginas para inserir elementos HTML em tempo real, a partir de eventos, a fim de tornar a interação do usuário com o sistema mais rica.

## 2.2.7 PostgreSQL

O PostgreSQL é um sistema gerenciador de banco de dados relacional. Utilizando o PostgreSQL é possível criar e gerir um banco de dados relacional utilizando código SQL. No projeto o banco é gerido inteiramente utilizando a biblioteca psycopg2 do Python e o ORM (Object-Relational Mapping) do Django, que abstrai as tabelas e instâncias do banco como classes e objetos, respectivamente.

## 2.2.8 Scrapy

Scrapy é um framework para desenvolvimento de web-crawlers escrito em Python. Web-crawling consiste em baixar automaticamente os dados de uma ou uma série de páginas web utilizando um bot que chamamos de web-crawler<sup>2</sup>. Utilizamos um desses bots para acessar a ferramenta de avaliação ASES-Web para uma série de URLs, baixando os relatórios de avaliação e registrando os resultados no banco de dados, para ser posteriormente acessado pela ferramenta de gestão.

## 2.2.9 AWS

A AWS (Amazon Web Services) é uma plataforma de serviços de cloud computing. Dentro os serviços fornecidos, foi utilizado o Elastic Beastalk para hospedar a ferramenta, junto ao banco de dados e ao web-crawler. Um outro serviço, chamado CodePipeline foi utilizado para realizar entrega contínua, isto é, o aplicativo executado na nuvem é atualizado de acordo o repositório principal no GitHub.

---

<sup>2</sup>Também conhecido como spider ou mesmo robô

## 2.3 Decisões de Projeto

### 2.3.1 ASES-Web vs ASES-Desktop

No processo de planejamento e também durante o desenvolvimento, foram realizadas algumas decisões que merecem menção. Primeiramente em respeito à avaliação realizada pelo sistema ASES em sua versão web, que é utilizada pelos funcionários da SMPED. O primeiro impulso foi utilizar a versão desktop para realizar as avaliações, mantendo assim todas as dependências em um mesmo servidor. Essa ideia foi descartada ao verificar-se que a versão desktop funciona a partir de uma interface gráfica, dificultando qualquer processo de automatização. Sendo assim, em uma segunda tentativa de usar a versão desktop, foi estudado o código fonte (que é aberto) a fim de identificar os módulos de interesse e construir uma API<sup>3</sup> a ser utilizada pela ferramenta. Porém após exaustivas tentativas de estudar o código (em Java) constatou-se que o esforço era desproporcional e improdutivo, devido a características do próprio código legado como alto acoplamento e baixa legibilidade. Por fim, a criação de um web-crawler em Python que realizasse a mesma requisição que os funcionários da SMPED, utilizando o ASES-Web foi identificado como sendo a melhor solução, de maior clareza e com maior viabilidade de produção.

### 2.3.2 Funcionalidades

Referente às funcionalidades que estariam presentes no sistema, foi realizada uma seleção do que seria mais importante de imediato e do que poderia ser deixado para atualizações futuras. Constatou-se antes de tudo que seria fundamental automatizar as requisições de avaliação ao ASES-Web, visto que é o processo mais trabalhoso e maior agregador de valor do sistema, e que, em um segundo momento, essa função deveria estar atrelada a um sistema informatizado que reunisse todas as informações dos selos, inclusive os resultados obtidos das avaliações. Ficou então estabelecido que essas seriam as principais frentes, o que ditou as funcionalidades tidas como fundamentais para a primeira versão do sistema, apresentada no presente trabalho.

Dentre as funcionalidades presentes na ferramenta, primeiramente temos a gestão e autenticação de usuários do sistema. É necessário possuir um usuário e fazer login para utilizar a ferramenta. Usuários possuem identificador e senha para login, informações pessoais e de contato como RG e Email, e possuem permissões para o uso da ferramenta, que podem ser de somente leitura, leitura e escrita ou permissão de administrador. Somente um usuário com permissão de administrador pode gerir usuários, isto é, cadastrar, editar informações, trocar senhas ou inativar usuários. Qualquer usuário pode pesquisar (por nome, identificador ou email) usuários e visualizar as informações pessoais e permissões dos usuários.

Além de usuários, também é possível cadastrar e gerir requerentes do selo de acessibilidade digital. Usuários com permissão de leitura e escrita podem cadastrar e editar requerentes. É possível ainda adicionar um representante legal ou associar um novo

---

<sup>3</sup>Application Programming Interface

selo cadastrado ao requerente. Qualquer usuário pode pesquisar (por nome/razão social, CPF/CNPJ, email, telefone ou representante legal) requerentes e visualizar suas respectivas informações e de seu representante legal.

Cadastrado um requerente, é possível cadastrar também um selo de acessibilidade digital associado a este requerente. O selo possui suas próprias informações, como domínio, data de concessão e até 20 URLs cadastradas, que podem ser editadas por um usuário que possua permissão de leitura e escrita. Apesar de possuir um prazo cadastrado para renovação, como não foi implementado um sistema de notificações, optou-se por não inativar o selos de forma automática. Os selos devem ser inativados manualmente para garantir que o gestor esteja ciente da inativação. Futuramente espera-se que a ferramenta realize essa operação automaticamente, notificando o gestor, ou que gere um aviso para que o gestor possa realizá-la manualmente. Qualquer usuário pode pesquisar (por domínio, identificador ou nome/razão social do requerente) um selo e visualizar suas respectivas informações.

Somado ao cadastro de requerentes e selos, o sistema tem como principal função a avaliação automática de todas as páginas de cada selo. Na página principal da ferramenta, com a lista dos selos concedidos e ativos, é possível verificar rapidamente se algum selo está aquém das exigências mínimas de acordo com a última avaliação. Para informações mais detalhadas, o gestor pode inspecionar um selo específico e verificar uma tabela com os resultados das avaliações para N dias passados referentes a cada URL do selo, além de ter acesso a cada um dos respectivos relatórios completos salvos pelo sistema ao passo de um clique.

### 2.3.3 Código

A respeito do código foram tomadas algumas decisões. Primeiramente evitou-se o uso excessivo de comentários, evitando que se justificasse código ruim com explicações. Em contrapartida, houve um esforço para manter o código o mais claro possível por si só. Isso foi alcançado parcialmente pela própria linguagem Python possuir essa premissa, mas também graças a funções curtas e pela escolha de bons nomes para as variáveis e funções. Nomes curtos não foram prioridade, em vez disso priorizou-se o uso de nomes altamente significativos. Outra decisão foi a de manter o código fonte inteiramente em inglês. Essa decisão foi tomada para manter uma coerência linguística com a linguagem Python, mas também com o código do próprio Django. Por fim, optou-se por separar o código em módulos, reunindo classes e funções fortemente relacionadas em um mesmo módulo, e separando-as quando possível em módulos diferentes. Cada módulo será mais aprofundado a seguir onde será coberta toda a estrutura da implementação.



# Capítulo 3

## Estrutura da Implementação

### 3.1 Módulos

Como mencionado anteriormente, o sistema foi dividido em módulos para estruturar a implementação de forma organizada e facilitar a sua manutenção. A seguir apresentamos cada módulo, citando suas funções dentro do sistema assim como alguns detalhes e particularidades referentes à respectiva implementação. Os módulos têm a estrutura padrão de qualquer app do Django<sup>1</sup>, exceto quando indicado. A estrutura e características gerais do Django não serão comentadas aqui, mas podem ser verificadas diretamente consultando a documentação do próprio framework (DJANGO, 2019).

#### 3.1.1 Configurações do Django

O primeiro módulo é aquele que contém as configurações básicas do aplicativo Django e está contido na pasta que se chama simplesmente *django app*. Aqui é definido o banco de dados a ser utilizado pela ferramenta, as pastas utilizadas para os templates e arquivos estáticos, assim como diversas outras definições do framework.

#### 3.1.2 Contas de Usuário

O módulo de contas de usuário é onde se encontra todo o código referente ao sistema de autenticação, cadastro e gestão de contas de usuário e se encontra na pasta *accounts* do aplicativo Django. A implementação foi feita utilizando e se baseando fortemente nas ferramentas fornecidas pelo framework Django e contidas em *django.contrib.auth*<sup>2</sup>. Como

---

<sup>1</sup>No Django, app é o nome dado aos diferentes módulos de uma aplicação. Geralmente apps do Django possuem certa independência, podendo ser inseridos em mais de uma aplicação sem problemas de dependência. No nosso caso os módulos estão mais inseridos no contexto, possuindo interdependências, mas apresentando a estrutura de um app. Utilizar vários apps foi uma decisão de desenvolvimento visando modularização do código.

<sup>2</sup><https://docs.djangoproject.com/en/2.2/ref/contrib/auth/>

particularidade foi utilizado um modelo de usuário customizado para suprir as necessidades do sistema no que se refere a informações adicionais de usuário, autenticação de usuário por um identificador não padrão e às definições de permissão também diferentes do padrão. Apenas usuários com permissão de administrador podem atuar livremente neste módulo, podendo cadastrar, editar informações, inativar e até mesmo deletar usuários, restando aos demais usuários somente a leitura das informações dos usuários cadastrados.

### 3.1.3 Requerentes

O módulo de requerentes contém todo o código referente ao cadastro e gestão de requerentes ao selo de acessibilidade digital e respectivos representantes legais, e está contido na pasta *applicants* do aplicativo Django. É o módulo de implementação mais simples do sistema, possui basicamente a função de CRUD<sup>3</sup> para os objetos de requerente e representante legal. É dependente do módulo de contas de usuário para trabalhar com as permissões, somente usuários com permissão de leitura e escrita podem cadastrar novos requerentes e representantes legais, ou editar os já cadastrados. O opção de exclusão é reservada aos administradores.

### 3.1.4 Selos

O módulo de selos contido na pasta *certifications* do aplicativo Django é o módulo principal da ferramenta, ele inclui o código referente aos selos de acessibilidade digital, as respectivas páginas web e também os resultados e relatórios das avaliações periódicas. Aqui se reúnem informações de todos os módulos, havendo dependência de todos eles. Permissões de usuário funcionam de forma análoga ao que ocorre no módulo de requerentes.

### 3.1.5 Relatórios

Único módulo da ferramenta que não é um app do Django, é o responsável por manter os relatórios em dia. A pasta *reports updater* do aplicativo Django contém um único arquivo com as funções de atualização dos relatórios, que consistem de uma função que executa o web-crawler e outra que inicializa um agendador de tarefas responsável por periodicamente realizar a execução da primeira função. Esta última função, cuja chamada se encontra no módulo de selos, é executada junto com a inicialização do sistema e utiliza a biblioteca apscheduler do Python.

### 3.1.6 Web Crawler

Único módulo separado da ferramenta em Django, localizado na pasta *web crawler* corresponde ao bot desenvolvido com o framework Scrapy voltado para web-crawling.

---

<sup>3</sup>Create, Read, Update and Delete

## 3.1 | MÓDULOS

Funciona como um serviço e pode ser executado de forma independente. Quando acionado, o programa busca as URLs de selos ativos diretamente do banco de dados, realiza as avaliações pelo ASEs-Web, baixa e salva os relatórios pdf em uma organizada estrutura de pastas e salva objetos com data e nota em porcentagem no banco de dados utilizando o ORM do próprio Django.



# Capítulo 4

## Conclusão

### 4.1 Resultados

O software desenvolvido atinge os objetivos estabelecidos no início do trabalho. O sistema possui todas as funções básicas e essenciais previstas para a gestão dos selos de acessibilidade digital da SMPED e deve trazer um enorme ganho de eficiência no trabalho realizado pelos funcionários da secretaria. É bastante simples de interagir com o sistema e ele propicia uma forma de trabalho dinâmica e direta, na qual o funcionário pode verificar rapidamente quais ações deve tomar e pode realizá-las facilmente por meio do sistema. Também espera-se que, graças ao cuidado com o código, seja fácil de se compreender e de estender o que foi feito, viabilizando a criação de novas funcionalidades na medida em que estas forem necessárias ou desejáveis para uma melhor experiência.

### 4.2 Próximos passos

Algumas funções não essenciais foram deixadas de fora nessa primeira versão do sistema, mas devem ser incluídas em futuras atualizações. Funções previstas incluem uma validação mais rígida dos dados inseridos no sistema<sup>1</sup>, um registro de todas as atividades do sistema, acessível pela própria ferramenta, e um sistema de notificações, voltado tanto para os gestores quanto para os requerentes. Também é esperado que, com o uso constante da ferramenta, identifiquem-se novas necessidades ou possíveis melhorias no sistema.

---

<sup>1</sup>Atualmente o sistema possui uma validação primária dos dados, que previne a entrada de dados ilegais ou nulos no sistema, mas não possui validação no sentido de padronizar o formato dos dados inseridos.



# Referências

- [DJANGO 2019] Django The web framework for perfectionists with DEADLINES. *Django documentation*. URL: <https://docs.djangoproject.com/en/2.2/> (acesso em 12/11/2019) (citado na pg. 11).
- [MÜLLER e CORRÊA 2017] Vanessa MÜLLER e Jordana Freire CORRÊA. *Deficiência e Acessibilidade: Um Panorama da União Europeia*. 2017. URL: <http://coral.ufsm.br/congressodireito/anais/2017/6-2.pdf> (acesso em 10/11/2019) (citado na pg. 1).
- [PETRI e FEDERICI 2018] Giacomo PETRI e Christian FEDERICI. *Automated testing is not enough for web accessibility ada compliance*. 2018. URL: <https://blog.usablenet.com/automated-testing-is-not-enough-for-web-accessibility-ada-compliance> (acesso em 13/09/2019) (citado na pg. 2).
- [WIKIPEDIA 2019] Wikipedia The Free ENCYCLOPEDIA. *Accessibility*. URL: <https://en.wikipedia.org/wiki/Accessibility> (acesso em 10/11/2019) (citado na pg. 1).