

**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA  
E TECNOLOGIA DA BAHIA**  
CAMPUS VITÓRIA DA CONQUISTA  
BACHARELADO EM SISTEMAS DE INFORMAÇÃO

**Daniel de Jesus Silva**

**Saiba Mais Brasil:** Ferramenta para Auxiliar a  
Percepção de Perfis de Políticos Brasileiros

Vitória da Conquista-BA

2018

Daniel de Jesus Silva

**Saiba Mais Brasil:** Ferramenta para Auxiliar a Percepção de  
Perfis de Políticos Brasileiros

Trabalho monográfico apresentado ao Programa de Graduação em Bacharelado em Sistemas de Informação do Instituto Federal de Educação, Ciência e Tecnologia da Bahia Campus Vitória da Conquista como requisito parcial para a obtenção do grau de Bacharel em Sistemas de Informação.

Orientador: Prof. Me. Claudio Rodolfo S. de Oliveira

Coorientador: Prof. Me. Pablo F. Matos

Vitória da Conquista-BA  
2018

J586s Silva Daniel de Jesus

Saiba Mais Brasil: ferramenta para auxiliar a percepção de perfis de políticos brasileiros. / Daniel de Jesus Silva.--. – Vitória da Conquista: IFBA, 2018  
75il.; color.

Orientador: Prof. Claudio Rodolfo S. de Oliveira

Projeto Final de Curso (Graduação)- Sistemas de Informação - Instituto Federal de Educação, Ciência e Tecnologia da Bahia - Campus de Vitória da Conquista-BA, 2018

1. Democracia 2. Tecnologias da informação e comunicação 3.. Web semântica I. Oliveira, Claudio Rodolfo S. de II. T.

CDD: 303.4830981

Daniel de Jesus Silva

**Saiba Mais Brasil:** Ferramenta para Auxiliar a Percepção de  
Perfis de Políticos Brasileiros

Este Trabalho de Conclusão de Curso foi julgado adequado para a obtenção do grau de Bacharel em Sistemas de Informação, e aprovado em sua forma final pelo Programa de Graduação em Bacharelado em Sistemas de Informação do Instituto Federal de Educação, Ciência e Tecnologia da Bahia *Campus* Vitória da Conquista.

Vitória da Conquista-BA, \_\_\_\_\_ de \_\_\_\_\_ de \_\_\_\_\_:

---

**Prof. Me. Claudio Rodolfo S. de  
Oliveira**

Orientador - IFBA Campus Vitória da  
Conquista

---

**Prof. Me. Pablo F. Matos**

Coorientador - IFBA Campus Vitória da  
Conquista

---

**Prof. Me. Stenio L. Araújo**

Convidado 1 - IFBA Campus Vitória da  
Conquista

---

**Prof<sup>a</sup>. Dr<sup>a</sup>. Cátia M. B. Khouri**

Convidado 2 - UESB Campus Vitória da  
Conquista

Vitória da Conquista-BA  
2018

*Dedico este trabalho, bem como todas as minhas demais conquistas, à minha família, em especial aos meus pais pelo apoio, compreensão e por acreditarem em mim incondicionalmente.*

## AGRADECIMENTOS

Agradeço a todas as pessoas que direta ou indiretamente colaboraram com o sucesso deste trabalho.

Agradeço ao professor Claudio Rodolfo de Oliveira, responsável pela orientação deste trabalho, ao professor Pablo Matos, coorientador, à professora Críjina Chagas pelo auxílio referente à análise e modelagem da solução e aos demais que apoiaram cada etapa da pesquisa e contribuíram com as revisões do conteúdo.

Agradeço à minha família, em especial à minha esposa Deisy e aos meus pais José Carlos e Déia, que embora não tivessem conhecimentos a respeito do tema sempre me apoiaram.

Agradeço a Deus, por tudo.

*“Consagre ao Senhor tudo o que você faz,  
e os seus planos serão bem-sucedidos.”  
(Bíblia Sagrada, Provérbios 16:3)*

## RESUMO

Atualmente, no Brasil, vive-se sob um sistema democrático no qual, com a inviabilidade de todos os cidadãos participarem diretamente das decisões, estas passam a ser tomadas por seus representantes eleitos. Porém, o que se observa é um cenário onde políticos legislam em causa própria e cidadãos não se sentem representados. As sessões plenárias são normalmente abertas ao público, canais de TV transmitem as votações realizadas e os dados públicos estão abertos para consultas na Internet, mas ainda assim, é difícil para o cidadão acompanhar continuamente o trabalho de seus representantes. Com o intuito de melhorar a transparência do processo de votação das proposições do poder legislativo, este trabalho propõe o desenvolvimento de uma ferramenta capaz de cruzar as vontades dos cidadãos com as decisões de seus representantes, o que permitirá que o usuário eleitor saiba se está sendo representado. Além disso, possibilitará que o representante saiba disso e mude de postura, caso seja necessário.

**Palavras-chaves:** Democracia. Transparência. Perfil político. Dados abertos. Tecnologias da informação e comunicação. Web semântica.

## ABSTRACT

Nowadays, in Brazil, one lives under a democratic system where, with the impossibility of all citizens participating directly in the decisions, they are taken by their elected representatives. However, what is observed is a scenario where politicians legislate on their own behalf and citizens do not feel represented. Plenary sessions are usually open to the public, TV channels broadcast votes and public data are open for consultation on the Internet, but it is still difficult for the citizen to continuously monitor the work of their representatives. In order to improve the transparency of the voting process of legislative proposals, this paper proposes the development of a tool capable of crossing the will of citizens with the decisions of their representatives, which will allow the voter user to know if he is being represented. In addition, it will make it possible for the representative to know this and change posture if necessary.

**Keywords:** Democracy. Transparency. Political profile. Open data. Information and communication technologies. Semantic Web.

## LISTA DE FIGURAS

Figura 1 – Elementos RDF . . . . .	23
Figura 2 – Modelo RDF . . . . .	23
Figura 3 – Grafo RDF . . . . .	24
Figura 4 – Comparação entre as arquiteturas de aplicações móveis. . . . .	28
Figura 5 – Arquitetura do Ionic Framework. . . . .	28
Figura 6 – Exemplo de arquivo turtle. . . . .	32
Figura 7 – Estrutura de consulta SPARQL. . . . .	33
Figura 8 – Exemplo de consulta SPARQL. . . . .	34
Figura 9 – Quadro comparativo entre trabalhos relacionados. . . . .	39
Figura 10 – Diagrama de Casos de Uso do Saiba mais Brasil. . . . .	41
Figura 11 – Diagrama de Classes do Saiba mais Brasil . . . . .	42
Figura 12 – Fluxograma do Saiba mais Brasil (RF2) e (RF3) . . . . .	43
Figura 13 – Arquitetura do Saiba mais Brasil. . . . .	44
Figura 14 – Documento da coleção Deputados . . . . .	45
Figura 15 – Parte do gráfico da ontologia dos dados do Saiba mais Brasil . . . . .	46
Figura 16 – Exemplo de consulta SPARQL . . . . .	48
Figura 17 – Resultado da consulta SPARQL do exemplo. . . . .	49
Figura 18 – Tela principal. . . . .	52
Figura 19 – Tela de resultados da busca de deputados. . . . .	53
Figura 20 – Tela de informações do deputado selecionado. . . . .	54
Figura 21 – Tela de seleção de deputado e filtro de proposições. . . . .	55
Figura 22 – Tela de análise de proposições. . . . .	56
Figura 23 – Tela de resultado da análise do deputado. . . . .	57
Figura 24 – Tela de publicação do resultado no Twitter. . . . .	58
Figura 25 – Site Saiba mais Brasil (Home). . . . .	59
Figura 26 – Site Saiba mais Brasil (Passo a passo para uso do App). . . . .	60
Figura 27 – Perfil Saiba mais Brasil no Twitter. . . . .	61
Figura 28 – Gráfico completo da ontologia dos dados do Saiba mais Brasil. . . . .	66
Figura 29 – Tela de introdução 1. . . . .	68
Figura 30 – Tela de introdução 2. . . . .	69
Figura 31 – Tela de introdução 3. . . . .	70
Figura 32 – Tela de dados do usuário. . . . .	71
Figura 33 – Site Saiba mais Brasil (Home). . . . .	72
Figura 34 – Site Saiba mais Brasil (Descrição do App). . . . .	73
Figura 35 – Site Saiba mais Brasil (Passo a passo para uso do App). . . . .	73
Figura 36 – Site Saiba mais Brasil (Funcionalidades do App). . . . .	74
Figura 37 – Site Saiba mais Brasil (Galeria do App). . . . .	74
Figura 38 – Site Saiba mais Brasil (Link do repositório e download do App). . . . .	75

## LISTA DE QUADROS

Quadro 1 – Exemplo de sentença separada em partes RDF. . . . .	23
Quadro 2 – Rotas da API Saiba mais Brasil. . . . .	67

## LISTA DE ABREVIATURAS E SIGLAS

- API** *Application Programming Interface*. 18, 22, 25, 27, 29, 32, 34, 42, 43, 45, 46, 49, 50, 60
- App** *Aplicativo*. 18, 27–29, 36, 37, 44–46, 49–51, 59, 60
- CLI** *Command Line Interface*. 28, 29, 31
- CSS3** *Cascading Style Sheets 3*. 29, 51
- EC2** *Elastic Compute Cloud*. 30, 49
- GPS** *Global Positioning System*. 29
- HTML5** *Hypertext Markup Language 5*. 22, 29, 51
- HTTP** *Hypertext Transfer Protocol*. 18, 34, 50
- JSON** *JavaScript Object Notation*. 18, 31, 44, 45, 49
- LOD** *Linked Open Data*. 24
- RDF** *Resource Description Framework*. 17, 22–24, 32, 34, 40, 46, 49, 50
- SPARQL** *SPARQL Protocol and RDF Query Language*. 17, 18, 32–34, 46, 47, 49, 62
- SQL** *Structured Query Language*. 31, 33
- TIC** *Tecnologias da Informação e Comunicação*. 17, 38
- URI** *Uniform Resource Identifier*. 22, 24, 32
- W3C** *World Wide Web Consortium*. 21, 24, 32
- XML** *Extensible Markup Language*. 24, 49

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b> . . . . .	<b>14</b>
<b>1.1</b>	<b>Problema</b> . . . . .	<b>14</b>
<b>1.2</b>	<b>Justificativa</b> . . . . .	<b>16</b>
<b>1.3</b>	<b>Objetivos</b> . . . . .	<b>16</b>
1.3.1	Objetivo geral . . . . .	16
1.3.2	Objetivos específicos . . . . .	17
<b>1.4</b>	<b>Metodologia</b> . . . . .	<b>17</b>
1.4.1	Classificação da Pesquisa . . . . .	18
<b>1.5</b>	<b>Estrutura do trabalho</b> . . . . .	<b>19</b>
<b>2</b>	<b>WEB SEMÂNTICA</b> . . . . .	<b>21</b>
<b>2.1</b>	<b>HTML5</b> . . . . .	<b>21</b>
<b>2.2</b>	<b>RDF</b> . . . . .	<b>22</b>
<b>2.3</b>	<b>Linked Open Data</b> . . . . .	<b>24</b>
2.3.1	Open Government . . . . .	24
<b>3</b>	<b>TECNOLOGIAS AUXILIARES AO DESENVOLVIMENTO</b> .	<b>27</b>
<b>3.1</b>	<b>Ionic Framework</b> . . . . .	<b>27</b>
3.1.1	Apache Cordova . . . . .	29
3.1.2	AngularJS . . . . .	29
3.1.3	Ionic CLI e Ionic Pro . . . . .	29
<b>3.2</b>	<b>NodeJS</b> . . . . .	<b>30</b>
3.2.1	Modelo single thread . . . . .	30
3.2.2	Express . . . . .	30
3.2.3	Mongoose . . . . .	30
3.2.4	Heroku . . . . .	30
<b>3.3</b>	<b>MongoDB</b> . . . . .	<b>31</b>
3.3.1	Orientado a documentos . . . . .	31
3.3.2	mLab . . . . .	31
<b>3.4</b>	<b>Web Semântica</b> . . . . .	<b>32</b>
3.4.1	Turtle . . . . .	32
3.4.2	Apache Jena Fuseki . . . . .	32
3.4.3	SPARQL . . . . .	32
3.4.4	Fuseki . . . . .	34
<b>3.5</b>	<b>Twitter API</b> . . . . .	<b>34</b>

<b>4</b>	<b>TRABALHOS CORRELATOS . . . . .</b>	<b>35</b>
4.1	Operação Serenata de Amor . . . . .	35
4.2	Deliberatório . . . . .	35
4.3	Radar Parlamentar . . . . .	36
4.4	Sr. Cidadão . . . . .	36
4.5	Monitora, Brasil! . . . . .	36
4.6	Meu Deputado . . . . .	37
4.7	Eu Parlamentar . . . . .	37
4.8	Poder do Voto . . . . .	37
4.9	Using ICTs to create a culture of transparency: e-government and social media as openness and anti-corruption tools for societies . . . . .	38
4.10	E-government application in combating corruption: China's case	38
4.11	Quadro comparativo entre trabalhos relacionados . . . . .	39
<b>5</b>	<b>SAIBA MAIS BRASIL . . . . .</b>	<b>40</b>
5.1	Requisitos Funcionais . . . . .	40
5.2	Requisitos Não-funcionais . . . . .	43
5.3	Arquitetura . . . . .	44
5.4	Persistência de Dados . . . . .	44
5.4.1	Dados Semânticos . . . . .	46
5.5	API (back-end) . . . . .	49
5.5.1	Rotas da API . . . . .	49
5.6	Aplicativo (front-end) . . . . .	50
5.6.1	App . . . . .	51
5.6.2	Site . . . . .	59
5.6.3	Twitter . . . . .	60
<b>6</b>	<b>CONSIDERAÇÕES FINAIS . . . . .</b>	<b>62</b>
6.1	Contribuições . . . . .	62
6.2	Trabalhos Futuros . . . . .	62
	<b>REFERÊNCIAS . . . . .</b>	<b>63</b>
	<b>APÊNDICE A – GRÁFICO DA ONTOLOGIA DOS DADOS DO SAIBA MAIS BRASIL . . . . .</b>	<b>66</b>
	<b>APÊNDICE B – ROTAS DA API SAIBA MAIS BRASIL . . . . .</b>	<b>67</b>
	<b>APÊNDICE C – TELAS DE INTRODUÇÃO DO APP . . . . .</b>	<b>68</b>

APÊNDICE D – SITE DO SAIBA MAIS BRASIL . . . . .	72
--	----

# 1 INTRODUÇÃO

Atualmente, no Brasil, vive-se sob um sistema democrático onde, em teoria, todos os cidadãos participam, direta ou indiretamente, das propostas e das criações das leis. A palavra democracia surgiu do grego *dēmokratía* ("governo do povo"), *demos* ("povo") e *kratos* ("poder"), no século V a.C., e era utilizada para referenciar a forma de governo de determinadas cidades gregas que permitiam aos cidadãos exercerem seus privilégios de decidir, opinar e discutir assuntos relacionados à gestão de suas cidades. Embora essa democracia não tenha sido tão efetiva, pois excluía os direitos das mulheres e dos escravos, esse sistema de governo tem sido implantado ao redor do mundo de forma crescente, principalmente após a Segunda Guerra Mundial.

No dicionário Michaelis (2011), democracia é definida formalmente como um "sistema político influenciado pela vontade popular e que tem por obrigação distribuir o poder equitativamente entre os cidadãos, assim como controlar a autoridade de seus representantes".

Existem algumas variantes de democracias no mundo. Porém, duas delas são básicas: a democracia direta, onde todos os cidadãos têm participação ativa na tomada de decisões do governo e a democracia representativa (ou indireta), onde o povo permanece com o poder soberano, mas o poder político é exercido indiretamente por meio de representantes eleitos, sendo esta última, a forma mais comum das democracias modernas inclusive a democracia brasileira.

A escolha dos representantes do povo se dá através das eleições, e é quando se tem a oportunidade de mudar os representantes eleitos anteriormente ou mantê-los no exercício da função que ocupam. Em qualquer um dos casos, utiliza-se o voto, que é obrigatório, e então a vontade da maioria define quem ocupará o cargo no próximo mandato.

## 1.1 Problema

Zapater (2015, on-line) explica que:

Em Estados com centenas de milhões de habitantes ficaria difícil manter a proposta de democracia direta ateniense, na base de juntar a galera na Acrópole e perguntar quem prefere o quê na base do "levanta a mão quem concorda" (até porque o grupo de cidadãos atenienses era bem mingado, já que excluía juridicamente qualquer um que não se enquadrasse na categoria de homem livre nascido em Atenas). Além disso, o governo democrático considera a existência de diferenças individuais, e essas diferenças tornam impossível a concordância absoluta de todos os integrantes do povo sobre tudo.

Por isso, nas democracias modernas foi desenvolvido o método de verificação do consenso da maioria para possibilitar a eleição de um representante do povo que exerça o poder em seu nome. Afinal, se todos são iguais e

todas as opiniões têm igual peso, a soma das opiniões da maioria – assim foi determinado – seria o meio mais prático (ou talvez o único possível) de se chegar a um acordo sobre quem iria exercer o poder em nome do povo.

Porém, como qualquer outro sistema de governo, esse também é sujeito a falhas. "O consenso da maioria é usado não por ser o método mais justo, mas sim por ser o mais eficaz" (ZAPATER, 2015, on-line).

Com a inviabilidade de todos os cidadãos brasileiros participarem diretamente das decisões, estas passam a ser tomadas por seus representantes eleitos. Quando se elege um candidato acredita-se que ele representará seus eleitores e fará suas vozes serem ouvidas. É como se fosse dada a ele uma procuração para tomar as decisões em nome dos cidadãos, além de um cheque em branco, ambos assinados pelo povo. Porém, outra realidade é bastante comum: políticos que pouco representam seu eleitorado e que legislam em causa própria com o objetivo de continuar no poder (G1, 2018).

Em Fevereiro deste ano, o G1 divulgou uma pesquisa do Instituto Locomotiva/Ideia Big Data em sua página na Internet, apontando que "96% dos brasileiros não se sentem representados pelos políticos em exercício no país" (G1, 2018, on-line).

A pesquisa foi feita com 1.500 pessoas, com 18 anos ou mais, de 24 a 28 de janeiro de 2018 e foi desenvolvida para o projeto RenovaBR. A pesquisa diz ainda que:

94% afirmam que os políticos estão mais preocupados em se manter no poder do que governar, e 89% acreditam que os políticos não pensam na população para tomar decisões. O levantamento ainda aponta que 89% acreditam que os políticos não se preparam para desempenhar bem seu mandato (G1, 2018, on-line).

Teixidor e Pazos (2016) do El Instituto explicam em seus vídeos da série #WHYDEMOCRACY, que os programas eleitorais dos partidos deveriam ser um contrato com a sociedade, mas na prática não têm tanta importância porque não precisam ser cumpridos e não há dispositivos legais para assegurar sua implementação. Os partidos e seus políticos mudam de comportamento para atrair eleitores, muitas vezes até fabricando candidatos a fim de conquistar os votos da maior parte da população.

Na democracia, caso os cidadãos estejam insatisfeitos, a alternativa é votar em outro candidato na próxima eleição, mas isso também não dá nenhuma garantia de satisfação.

Além da corrupção, da falta de preparo para desempenhar bem seu mandato e da falta de representação, outro fator agrava esse cenário: a falta de transparência. Segundo a pesquisa: "95% dos entrevistados afirmam que os atuais políticos não são transparentes." (G1, 2018, on-line).

A Lei de Acesso a Informações Públicas Brasileiras, nº 12.527/2011, diz que

é dever do Estado garantir o direito de acesso à informação, que será franqueada, mediante procedimentos objetivos e ágeis, de forma transparente, clara e em linguagem de fácil compreensão (BRASIL, 2011, on-line).

As sessões plenárias são normalmente abertas ao público. Canais de TV transmitem as votações realizadas, os dados públicos estão abertos para consultas na Internet e vários outros meios de informação. Porém, ainda assim, é difícil para o cidadão acompanhar continuamente o trabalho de seus representantes. Afinal, frequentemente proposições são votadas e muitos eleitores não têm ideia se o candidato que ele ajudou a eleger realmente o está representando.

Entre os desafios que se apresentam ao governo hoje, pode-se mencionar o acesso à Internet por parte da população e utilização das tecnologias da informação e comunicação como meios para modernizar a administração pública conferindo maior transparência na relação com os cidadãos.

## 1.2 Justificativa

É necessário monitorar o trabalho do governo fornecendo ao cidadão o máximo de informações para aumentar a transparência, e capacitar o público a fiscalizar o desempenho dos políticos de forma mais próxima e eficiente.

Para Izdebski (2015), há vários casos nos setores público e privado, nos quais o acesso aos dados ajudaria a prevenir e combater a corrupção. Segundo Kim, Kim e Lee (2009), o aumento da transparência leva à diminuição da corrupção e Moon (2003) salienta que o uso da tecnologia da informação pode restaurar a confiança do público.

Rose-Ackerman (2012), Lambsdorff (1999) entre outros autores compartilham o mesmo ponto de vista e argumentam que quanto mais informações compartilhadas mais o cidadão compreende as realizações do governo, o que lhe proporciona tomar decisões políticas acertadas, melhorando a responsabilidade dos políticos.

Ter informações acessíveis e agrupadas de forma adequada das leis e projetos votados na casa parlamentar, pode tornar o sistema político brasileiro mais próximo de uma democracia de fato, onde os eleitores podem ter ciência se seus representantes têm cumprido o projeto de governo e se estão atendendo aos seus anseios, permitindo assim a tomada de decisão quanto a permanência desta representação no próximo pleito.

Segundo Tavares (2018, on-line),

Se o eleitor tivesse na mão uma ferramenta para acompanhar a atividade do deputado em quem votou, formar opinião própria sobre projetos que seu representante avalia e ainda medir a afinidade entre seu pensamento e as posturas do parlamentar, o Congresso seria melhor.

## 1.3 Objetivos

### 1.3.1 Objetivo geral

Com o intuito de melhorar a transparência do processo de votação das proposições do poder legislativo na Câmara dos Deputados, este trabalho propõe o desenvolvimento

de uma ferramenta capaz de cruzar as vontades dos cidadãos com as decisões de seus representantes, permitindo que o usuário eleitor saiba se está sendo representado. Além disso, possibilitará que o representante saiba disso e mude de postura, caso seja necessário.

### 1.3.2 Objetivos específicos

Ao considerar o objetivo geral anteriormente explicitado, se destacam os seguintes objetivos específicos necessários dentro do escopo do trabalho:

- a) definição das melhores linguagens, *frameworks* e tecnologias para o desenvolvimento da aplicação e para a manipulação e consulta dos dados semânticos;
- b) desenvolvimento de um aplicativo que inclua rotinas que permitam ao usuário acessar proposições votadas nas casas legislativas, comparando suas respostas com as de seus representantes eleitos;
- c) disponibilizar abertamente os dados em *Resource Description Framework* (RDF) de forma que possam ser utilizados em consultas semânticas no padrão *SPARQL Protocol and RDF Query Language* (SPARQL), onde será possível saber, por exemplo, qual o gênero ou a escolaridade do político que mais atende aos anseios dos eleitores;
- d) desenvolvimento de um site com a descrição da ferramenta, suas funcionalidades e link do projeto no *Github*;
- e) disponibilizar todo o código fonte abertamente no *Github*.

## 1.4 Metodologia

Este capítulo aborda a metodologia utilizada na elaboração deste trabalho. Para a apresentação dos procedimentos aplicados, considera-se o sucinto conceito de método de Moresi (2003, p. 13), expressado a partir de conceitos de outros autores discutidos em sua obra:

O método é o conjunto das atividades sistemáticas e racionais que, com maior segurança e economia, permite alcançar o objetivo - conhecimentos válidos e verdadeiros -, traçando o caminho a ser seguido, detectando erros e auxiliando as decisões do cientista.

O desenvolvimento deste trabalho pode ser segmentado nas etapas descritas a seguir:

- a) pesquisa: nesta etapa foi realizada uma revisão bibliográfica em livros e artigos disponíveis por meio da Internet com o intuito de buscar conhecimentos acerca do uso das Tecnologias da Informação e Comunicações (TICs) e da Web Semântica em conjunto com os dados abertos do Governo Federal.

Foram feitas também pesquisas sobre a representatividade dos Deputados Federais em relação aos seus eleitores e também estudos sobre as votações na Câmara dos Deputados, uso dos dados abertos, uso da *Application Programming Interface* (API) do *Twitter*, desenvolvimento de Aplicativos (Apps) móveis e consultas semânticas em SPARQL além de buscas sobre possíveis soluções semelhantes em trabalhos correlatos;

- b) projeto: nesta etapa foi feita uma análise das tecnologias mais adequadas, visando determinar a melhor forma para a implementação do projeto. Foram gerados diversos diagramas e *mockups* que foram determinantes para a definição da arquitetura. Considerando que a ferramenta proposta tem como objetivo a atuação em plataforma móvel, o Ionic Framework foi escolhido para a implementação do App e o Express Framework para a implementação da API, ambos são baseados na linguagem JavaScript e utilizam o formato *JavaScript Object Notation* (JSON) para as requisições e respostas *Hypertext Transfer Protocol* (HTTP). Para a persistência dos dados a escolha foi para um banco de dados não-relacional, e para essa finalidade o MongoDB foi eleito a melhor opção, já que além do excelente desempenho possui uma fácil integração com tecnologias baseadas em JavaScript;
- c) implementação: nesta etapa foi realizado o desenvolvimento propriamente dito da ferramenta Saiba mais Brasil, que inclui a implementação do App, da API, do site e a criação do perfil no *Twitter*. Para esta etapa foram utilizados dois cenários para testes: primeiramente em ambiente local e depois em produção com a publicação do site e da API por meio da Internet e a compilação do App Saiba mais Brasil para a plataforma Android (disponibilizado para download no site). A ferramenta passou por várias melhorias ao longo desta etapa;
- d) documentação: esta etapa compreende a redação de todos os textos elaborados que compõem este trabalho, assim como todo o conteúdo produzido referente à implementação da ferramenta, inclusive os arquivos de código-fonte que estão disponíveis no *Github*. Esta etapa esteve em prática durante todo o processo desde o início do projeto até a sua conclusão.

#### 1.4.1 Classificação da Pesquisa

A metodologia utilizada para o desenvolvimento deste trabalho, de acordo com Moresi (2003), Prodanov e Freitas (2013), é de caráter:

- a) quanto à natureza da pesquisa:

- pesquisa aplicada, com a finalidade de produzir conhecimentos para a aplicação prática na solução do problema encontrado, por meio da implementação da ferramenta Saiba mais Brasil.
- b) quanto à abordagem do problema:
- pesquisa quantitativa, considerando os resultados da pesquisa feita pelo Instituto Locomotiva/Ideia Big Data para o projeto RenovaBR, utilizados na caracterização do problema, a fim de classificá-los e analisá-los com o uso de técnicas estatísticas com o intuito de elaborar hipóteses;
  - pesquisa qualitativa, considerando comportamentos e fenômenos relacionados à representatividade dos Deputados Federais em relação aos cidadãos, identificando e analisando dados não mensuráveis numericamente.
- c) quanto aos objetivos:
- pesquisa exploratória, no sentido de buscar o domínio do estado da arte do tema proposto e também pesquisar por soluções similares na bibliografia correlata e em fontes adicionadas ao longo do trabalho;
  - pesquisa descritiva, orientada a casos de uso, em virtude do estudo, análise, registro e interpretação das informações geradas a partir das simulações e testes feitos no aplicativo Saiba mais Brasil e também das consultas semânticas.
- d) quanto aos meios de investigação:
- pesquisa de campo, considerando a inquirição e coleta de dados por meio da pesquisa do Instituto Locomotiva/Ideia Big Data entre outros resultados de trabalhos que abordam o tema;
  - pesquisa telematizada, com o auxílio da Internet, principal meio de comunicação utilizado nas buscas sobre o tema, referencial teórico e trabalhos correlatos;
  - investigação documental, a partir do estudo e manipulação das informações do Portal dos Dados Abertos do Governo Federal;
  - pesquisa bibliográfica, com estudos baseados em materiais publicados em livros e artigos referenciados ao longo deste trabalho;
  - pesquisa experimental, a partir da implementação da ferramenta Saiba mais Brasil e manipulação direta das informações relacionadas com a solução proposta.

## 1.5 Estrutura do trabalho

Este trabalho está dividido em capítulos, sendo que este, o Capítulo 1, apresenta uma introdução ao tema assim como os objetivos e a metodologia. Os Capítulos 2 e 3

---

expõem a fundamentação teórica nos quais são descritos assuntos relevantes ao trabalho. O Capítulo 4 traz um resumo de outros trabalhos que se relacionam com o tema, assim como uma análise comparativa de cada um deles com este trabalho. O Capítulo 5 descreve a modelagem e desenvolvimento da solução proposta e o Capítulo 6 apresenta as considerações finais e pontua funcionalidades e melhorias não incluídas neste trabalho, mas que podem ser implementadas futuramente.

## 2 WEB SEMÂNTICA

Neste capítulo é apresentado um estudo sobre Web Semântica, entre outros assuntos que envolvem este tema, assim como seus conceitos e aplicações.

Em outubro de 1994, com a fundação do *World Wide Web Consortium* (W3C), um consórcio mundial liderado por Tim Berners-Lee, as tecnologias e funcionalidades do ambiente Web começaram a ser padronizadas. Mais tarde, em 1998, a W3C iniciava pesquisas relacionadas ao desenvolvimento de uma nova faceta da Web capaz de dar significado aos dados de modo que as máquinas pudessem compreendê-los.

Segundo Costa Junior e Magalhães (2013, p. 2),

O conceito de Web Semântica surgiu em 2001 com a publicação do artigo, "Web Semântica: um novo formato de conteúdo para web que tem significado para computadores e que vai iniciar uma revolução de novas possibilidades.", desenvolvido em parceria por Tim Berners-Lee, James Hendler e Ora Lassila e publicado pela revista *Scientific American*. Segundo Tim Berners-Lee, criador da web e impulsionador da web semântica no W3C, "A Web Semântica é uma extensão da web atual na qual a informação recebe um significado claro e bem definido, possibilitando uma melhor interação entre computadores e as pessoas"[Berners-Lee, 2001].

Ramalho (2006, p. 38) explica que

é evidente que o objetivo final da Web Semântica é atender as pessoas e não os computadores, mas para isso torna-se necessário construir instrumentos que forneçam sentido lógico e semântico para as máquinas. Assim, pode-se verificar que a Web Semântica é uma tentativa inversa de solução, comparando-se com as tradicionalmente desenvolvidas, onde a idéia é pensar nas máquinas para que estas possam servir aos humanos de maneira mais eficiente.

Costa Junior e Magalhães (2013, p. 2) concluem que "A intenção da web semântica não é substituir pessoas e sim fazer com que as máquinas trabalhem de forma cooperativa e otimizada com os seres humanos."

### 2.1 HTML5

Para Costa Junior e Magalhães (2013, p. 3),

A criação de sites com semântica depende não apenas da utilização de *tags*, microdatas ou qualquer outra API, mas também da forma como é organizada as informações na página, ou seja, torna-se necessária a criação de uma metodologia e padrões para o uso dos novos recursos, dar início à utilização da semântica no código.

Com o objetivo de atribuir significados às seções de código das páginas, o *Hypertext Markup Language 5* (HTML5) trouxe novos elementos para a construção de sites com semântica:

- <HEADER>, <NAV>, <ARTICLE>, <ASIDE>, <SECTION>, <HGROUP> e <FOOTER>.

Esses elementos vieram para substituir as *tags* <DIV> genéricas que são utilizadas para o desenvolvimento dos *layouts* das páginas.

Outras *tags* com semântica:

- <P>, <H1 a H6>, <STRONG> e <CODE>.

Há também, com o mesmo objetivo, a API *Microdata*, que permite adicionar propriedades descritivas mais específicas para um elemento.

## 2.2 RDF

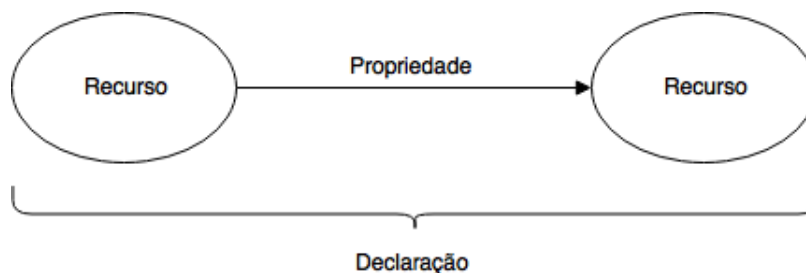
O RDF, segundo Lima e Carvalho (2005, p. 1),

é uma linguagem de propósito geral para representação de recursos na Web. Ele foi projetado para situações onde as informações necessitam ser processadas por aplicações, em lugar de somente serem visualizadas por pessoas. O RDF é baseado na idéia de que os recursos são descritos através de declarações e possuem propriedades que têm valores. Ele modela as declarações como nós e arcos em um grafo. Uma outra forma para modelar as declarações em RDF é através das chamadas “triplas RDF”. Uma tripla é dividida em três partes: sujeito, predicado e objeto.

O modelo para a descrição de dados em RDF é baseado em três tipo de elementos:

- recursos: um recurso pode ser qualquer coisa que tenha identidade. Pode ser uma página da Web, um website inteiro ou parte deste. Pode ser também um objeto não acessível via Web, como um livro, uma atriz ou uma banda de rock. Recursos são sempre especificados por *Uniform Resource Identifiers* (URIs);
- propriedades: uma propriedade é uma característica, um atributo ou uma relação utilizada para descrever o recurso;
- declarações: uma declaração é um recurso específico com uma propriedade definida mais o valor desta propriedade. Pode-se dizer que uma declaração é um recurso mais as propriedades desse recurso e mais o valor dessas propriedades.

Graficamente esses elementos podem ser representados como na Figura 1.

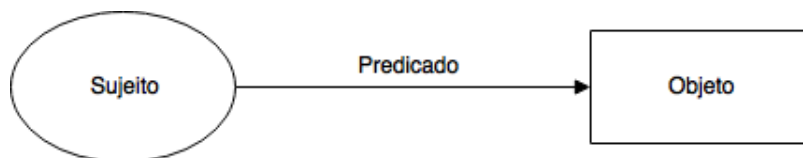
**Figura 1** – Elementos RDF

**Fonte** – Elaborada pelo autor.

Essas três partes individuais são chamadas respectivamente de sujeito, predicado e objeto. Em outras palavras, o modelo básico primitivo do RDF consiste de triplas de objeto, propriedade e valor em que:

- um modelo é um conjunto de declarações;
- sujeito é um recurso;
- predicado é um recurso;
- objeto pode ser um recurso ou um literal.

Na Figura 2 é apresentado o modelo básico do RDF.

**Figura 2** – Modelo RDF

**Fonte** – Elaborada pelo autor.

Por exemplo, considere a sentença: "John Doe é o pai de Peter Doe". Esta sentença pode ser dividida em partes, como mostrado no Quadro 1:

**Quadro 1** – Exemplo de sentença separada em partes RDF.

Parte	Valor
Sujeito	<a href="http://meudominio.com/pessoa#john_doe">http://meudominio.com/pessoa#john_doe</a>
Predicado	<a href="http://meudominio.com/propriedade#pai_de">http://meudominio.com/propriedade#pai_de</a>
Objeto	<a href="http://meudominio.com/pessoa#peter_doe">http://meudominio.com/pessoa#peter_doe</a>

**Fonte** – Elaborado pelo autor.

Também é possível representar esta sentença através de grafos rotulados como na Figura 3.

**Figura 3** – Grafo RDF



**Fonte** – Elaborada pelo autor.

Lima e Carvalho (2005, p. 1) explicam que

Para representar as declarações de forma que possam ser mais facilmente processadas por máquinas, RDF usa a Extensible Markup Language (XML). O RDF define uma linguagem de marcação XML específica, chamada de RDF/XML. Além destas características, RDF permite definir um vocabulário para ser usado nas declarações. Este vocabulário é chamado de RDF Schema. Ele é especificado como um conjunto de classes, propriedades e restrições entre seus relacionamentos .

Um exemplo de arquivo RDF é apresentado na Figura 6.

## 2.3 Linked Open Data

*Linked Open Data* (LOD) é uma faceta da Web de Dados que objetiva padronizar a forma de disponibilização de dados para que sejam facilmente referenciáveis com conteúdo acessível por software.

Segundo Izdebski (2015), um dado público verdadeiramente aberto é um conjunto de dados que está essencialmente preparado para que alguém o pegue e faça o que quiser com ele. No entanto, é necessário realizar um conjunto de práticas que fazem uso de padrões amplamente utilizados como a *Extensible Markup Language* (XML) que é uma linguagem recomendada pelo W3C para a criação de documentos com dados organizados hierarquicamente, tais como textos, banco de dados ou desenhos vetoriais, e o RDF, também recomendado pelo W3C, que permite criar um modelo simples de dados, com uma semântica formal e usar vocabulários baseados em URIs.

### 2.3.1 Open Government

Segundo (ACWG) (2015) e Iglesias (2017), Open Government é uma especificidade da LOD, com o intuito de disponibilizar dados dos governos de forma que eles se tornem mais acessíveis, receptivos, inclusivos, transparentes, responsáveis e eficientes.

BRASIL (2016, on-line) explica que,

Em vários países do mundo, as pessoas estão cada vez mais interessadas e exigentes em relação à transparência no trabalho de seus governantes e das instituições e empresas públicas. Em resposta a esse desejo, as instituições onde esse trabalho acontece têm tornado suas informações cada vez mais públicas. No Brasil, por exemplo, a Lei de Acesso à Informação (LAI), de 2011, garante o direito de solicitar e receber informações de órgãos públicos, sem ter que pagar por isso, para qualquer cidadão ou entidade.

Hacker, Alvarenga e Esfera.mobi (2011) e BRASIL (2009), na Cartilha técnica para publicação de dados abertos no Brasil demonstram que, normalmente, trabalhos nessa área focam no combate a corrupção ou na organização política.

O Dados Abertos da Câmara dos Deputados é um exemplo de iniciativa do governo brasileiro em disponibilizar as informações públicas do Poder Legislativo. "Esse acesso livre à informação permite que todos cidadãos tenham a mesma capacidade de fiscalizar, monitorar, conhecer e discutir os gastos, as ações e as decisões das entidades" (BRASIL, 2016, on-line).

BRASIL (2016, on-line) explica que "o Dados Abertos é um serviço da Câmara para fornecer dados sobre as atividades parlamentares da Casa a qualquer cidadão que possa se conectar à internet". Porém, deixa claro que a disponibilização desses dados é feita com o propósito de serem utilizados em outras aplicações por meio de APIs:

Apesar de ser um serviço prestado por meio da internet, é algo bem diferente do que é o Portal da Câmara, por exemplo. Portais são feitos para serem vistos, lidos e compreendidos por gente. Serviços de dados abertos são feitos para máquinas e programas (softwares): os dados são entregues em forma pura, sem formatações visuais, organizados para facilitar o processamento em computadores. (BRASIL, 2016, on-line)

Uso desses dados e da API de modo geral, não apenas é permitido como também é incentivado pelo governo, inclusive com a realização de concursos, *hackathons* e outros eventos. Toda a documentação necessária para a utilização da API está disponível no site dos Dados Abertos.

A entidade internacional, Fundação do Conhecimento Aberto (OKFN), explica que para ser considerado aberto, um conjunto de dados precisa atender aos seguintes requisitos:

- Completos - Para os dados serem completos, todos os conjuntos de dados públicos devem ser disponibilizados, e não somente uma parte deles;
- Primários - Os dados devem ser publicados na forma coletada na fonte, com o mais alto nível de granularidade possível, e não de forma agregada ou transformada;
- Atuais - Os dados devem ser disponibilizados ainda "quentes", tão rapidamente quanto seja necessário para que sejam úteis;
- Acesso não discriminatório - Os dados devem ser disponíveis a todos os cidadãos interessados, sem que seja necessária identificação, registro ou cadastro;

- 
- Acessíveis - Os dados devem ser disponibilizados para o público mais amplo possível e para os propósitos mais variados;
  - Formatos não-proprietários - Os dados devem ser disponíveis em pelo menos um formato sobre o qual nenhum ente tenha controle exclusivo;
  - Processáveis por máquina - Os dados devem ser razoavelmente estruturados para possibilitar o seu processamento automatizado;
  - Livres de licenças - Os dados não devem ser sujeitos a regulações de direitos autorais, marcas, patentes ou segredo industrial (exceto, claro, os dados que exigem sigilo e respeito à privacidade) (BRASIL, 2016, on-line).

### 3 TECNOLOGIAS AUXILIARES AO DESENVOLVIMENTO

Neste capítulo são discutidas as principais tecnologias, *frameworks* e APIs relacionados a este trabalho, com breves descrições e demonstrações de exemplos.

#### 3.1 Ionic Framework

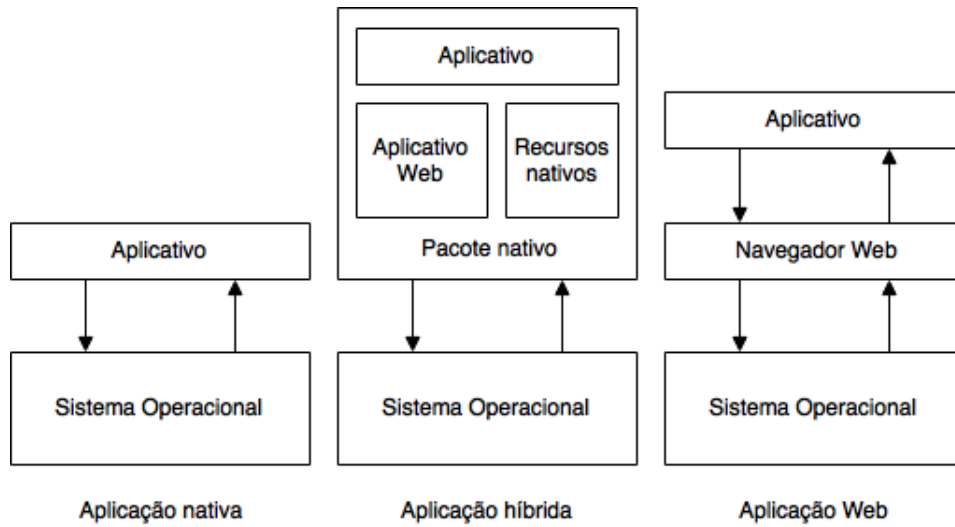
Ionic é um *framework* de código aberto para desenvolvimento de aplicações híbridas para dispositivos móveis. Foi criado no final de 2013 com o intuito de proporcionar um desenvolvimento rápido e fácil de Apps.

Existem diversos *frameworks* atualmente no mercado, cada um com suas características e propósitos. Estas particularidades ocasionaram a divisão das plataformas de desenvolvimento móvel em três categorias: Web, Nativa e Híbrida:

- Apps Web são aplicações projetadas para serem executadas nos navegadores dos dispositivos móveis e independe do sistema operacional. Normalmente são responsivas, se adequam ao tamanho reduzido das telas dos aparelhos e são hospedadas em servidores Web, o que permite uma rápida atualização porém, ao custo de uma disponibilidade limitada de recursos como armazenamento local entre outros;
- aplicações nativas são desenvolvidas diretamente para a plataforma alvo e nesse caso, linguagens específicas para cada sistema operacional, por exemplo: Swift para iOS, Kotlin para Android, etc. Apps nativos possuem um maior desempenho além de acesso facilitado aos recursos dos dispositivos. Em contrapartida cada alteração no App deve ser efetuada individualmente para cada plataforma e em seguida disponibilizada para atualização nos dispositivos;
- entre estas duas está a categoria de aplicações híbridas onde apenas um código-fonte é escrito e depois pode ser compilado para versões nativas de diversas plataformas. Apps híbridos combinam a portabilidade dos Apps Web com a disponibilidade de recursos das aplicações nativas. Existem vários *frameworks* para desenvolvimento de Apps híbridos entre eles o Ionic Framework.

Uma comparação entre as arquiteturas dessas três categorias de Apps móveis é apresentada na Figura 4.

**Figura 4** – Comparação entre as arquiteturas de aplicações móveis.

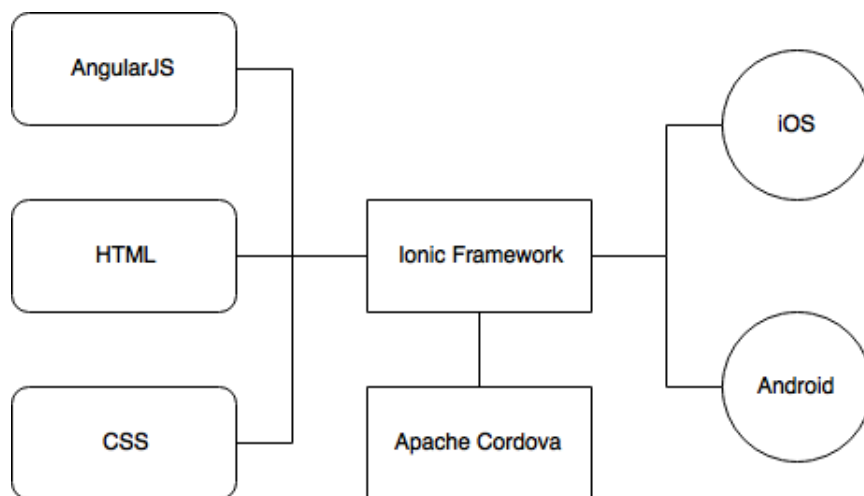


Fonte – Elaborada pelo autor.

O Ionic Framework possui uma arquitetura baseada na Figura 5 e consiste em uma pilha de componentes e outros *frameworks*:

- Apache Cordova: responsável pela integração com os recursos nativos do dispositivo;
- AngularJS: *framework* JavaScript para criação da parte Web do App;
- Ionic *Command Line Interface* (CLI) e Ionic Pro: ferramentas e componentes para criação, manutenção e compilação do projeto.

**Figura 5** – Arquitetura do Ionic Framework.



Fonte – Elaborada pelo autor.

### 3.1.1 Apache Cordova

Resumidamente, o Cordova é responsável por fazer o código JavaScript acessar recursos nativos do dispositivo, como câmera, Global Positioning System (GPS), acelerômetro, dentre outros. O Cordova provê um *framework* para construção de aplicativos para dispositivos móveis usando *Cascading Style Sheets 3* (CSS3), HTML5 e JavaScript em vez de depender das APIs específicas de plataformas como Android, iOS ou Windows Phone e assim possibilita trabalhar em uma camada superior independente do sistema operacional.

Atualmente o Cordova suporta os seguintes sistemas operacionais: Apple iOS, Bada, BlackBerry, Firefox OS, Google Android, LG webOS, Microsoft Windows Phone (7/8), Nokia Symbian OS, Tizen (SDK 2.x) e Ubuntu Touch.

### 3.1.2 AngularJS

AngularJS é um *framework* JavaScript para desenvolvimento de aplicações Web. É de código aberto e mantido pela Google.

Características que se destacam:

- é de código aberto (e, vale mencionar, mantido pela Google);
- arquitetura da aplicação em camadas bem definidas;
- permite a criação modular e de componentes reutilizáveis;
- já possui a infraestrutura para integração com back-end;
- facilita a etapa de testes de forma automatizada.

### 3.1.3 Ionic CLI e Ionic Pro

Ionic CLI é uma ferramenta de linha de comando que auxilia no desenvolvimento de aplicações Ionic. Permite ao desenvolvedor executar comandos dentro do projeto para tarefas específicas, como mostram os exemplos abaixo:

- `ionic start`: cria um novo projeto;
- `ionic serve`: executa o projeto.

Ionic Pro é um conjunto poderoso de ferramentas e serviços que garante um nível elevado de agilidade na construção e publicação de um App. Oferece, entre outros, os seguintes serviços:

- *deploy*: disponibiliza e atualiza o aplicativo remotamente em tempo real para testes ou produção;
- *view*: permite testar o aplicativo rapidamente através do App Ionic View para iOS e Android além da possibilidade de compartilhar o App com vários testadores;
- *package*: compila os binários dos aplicativos nativos na nuvem para iOS e Android;

- *monitoring*: monitora erros de execução em produção.

Cada um desses serviços é baseado em torno de um fluxo de trabalho simples baseado em *git*, que deve ser familiar para qualquer desenvolvedor de aplicativos.

## 3.2 NodeJS

O NodeJS é uma plataforma para desenvolvimento construída a partir do motor JavaScript V8 do Google Chrome. NodeJS é baseado em um modelo assíncrono de *Input/Output* (I/O) não bloqueante que o torna leve e eficiente. Além disso, o NodeJS possui um ótimo gerenciador de pacotes, o npm, que conta com um ecossistema de bibliotecas de código aberto.

### 3.2.1 Modelo single thread

Um cenário bastante comum é quando um servidor Web recebe várias requisições por segundo. Se a cada nova requisição uma nova thread for criada, como acontece em outras linguagens como Java, PHP e Ruby, essa alta utilização de recursos pode gerar a necessidade de mais servidores para suportar a demanda. Por isso, o modelo assíncrono single thread consegue processar mais requisições concorrentes com um número bem menor de recursos.

### 3.2.2 Express

Express é um *framework* para desenvolvimento de aplicações Web do NodeJS. Ele é simples, flexível, contém um conjunto de recursos robusto, um sistema de roteamento muito eficiente, um executável para geração de aplicações e muito mais.

### 3.2.3 Mongoose

Mongoose é uma biblioteca para NodeJS baseada em *Schemas*. É projetada para funcionar em um ambiente assíncrono e usada para modelagem de objetos MongoDB, já que o MongoDB não apresenta uma estrutura relacional como outros bancos de dados mais populares.

### 3.2.4 Heroku

Heroku é um *Platform as a Service* (PaaS) ou plataforma como serviço, que suporta várias linguagens de programação, inclusive NodeJS. Todos os serviços do Heroku estão hospedados na plataforma de computação em nuvem *Elastic Compute Cloud* (EC2) da Amazon.

As aplicações executadas no Heroku possuem um painel onde é possível gerenciá-lo e monitorá-lo e contam também com um domínio exclusivo (normalmente "nomedaapli-

cacao.herokuapp.com") através do qual é feito o acesso ao aplicativo. A implantação é feita através do Heroku CLI, que fornece vários comandos para o gerenciamento da aplicação. É possível também vincular um aplicativo Heroku à um repositório no GitHub ou Dropbox e assim ele será atualizado automaticamente.

### 3.3 MongoDB

MongoDB é um banco de dados de código aberto, gratuito, de alta performance, sem esquemas e orientado à documentos, escrito na linguagem de programação C++ e lançado em 2009.

#### 3.3.1 Orientado a documentos

Existem vários bancos NoSQL atualmente no mercado, visto que existem vários problemas de persistência de dados que o *Structured Query Language* (SQL) tradicional não resolve. Nesses casos, os bancos de dados não relacionais orientados a documentos são os mais comuns.

Basicamente, neste tipo de banco, temos coleções de documentos, nas quais cada documento é autossuficiente, e contém todos os dados que se pode precisar, ao invés do conceito de não repetição combinado com chaves estrangeiras do modelo relacional. Isso permite que as buscas sejam feitas sem o uso de JOINS que prejudicam muito a performance em *queries* e são necessárias no modelo relacional. Assim, é possível modelar a sua base de forma que cada *query* faça toda a busca de uma vez no banco e com apenas uma chave primária pega tudo o que for preciso. Obviamente, isto tem um custo: armazenamento em disco. Não é raro bancos MongoDB consumirem muitas vezes mais disco do que suas contrapartes relacionais.

Por ser orientado a documentos JSON, muitas aplicações podem modelar informações de modo muito mais natural, pois os dados podem ser aninhados em hierarquias complexas e ainda ser indexáveis e fáceis de buscar, igual ao que já é feito em JavaScript.

#### 3.3.2 mLab

O mLab é um *Database as a Service* (DBaaS) ou banco de dados como serviço. O mLab provê o serviço de armazenamento de dados baseado em nuvem totalmente gerenciável com uma hospedagem de bancos de dados MongoDB. Possui entre outros o plano gratuito com algumas limitações.

## 3.4 Web Semântica

### 3.4.1 Turtle

Um documento Turtle (Terse RDF Triple Language) é uma representação textual para expressar grafos em RDF. Possui uma sintaxe semelhante ao SPARQL e apresenta a informação por meio de triplas de sujeito, predicado e objeto onde cada um desses elementos é expresso como uma URI.

Na Figura 6 é descrito o relacionamento entre o Homem-Aranha e o Duende verde:

**Figura 6** – Exemplo de arquivo turtle.

```
@base <http://example.org/> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix rel: <http://www.perceive.net/schemas/relationship/> .

<#green-goblin>
  rel:enemyOf <#spiderman> ;
  a foaf:Person ;    # in the context of the Marvel universe
  foaf:name "Green Goblin" .

<#spiderman>
  rel:enemyOf <#green-goblin> ;
  a foaf:Person ;
  foaf:name "Spiderman", "Человек-паук"@ru .
```

Fonte – W3C (2014).

### 3.4.2 Apache Jena Fuseki

O Apache Jena é um *framework* Java de código aberto para desenvolvimento de aplicações de Web Semântica. Ele fornece uma API para extrair e escrever dados RDF e também possui um conjunto de classes que permitem fazer consultas simples no modelo. Para realizar consultas mais complexas é necessário utilizar SPARQL.

### 3.4.3 SPARQL

O termo SPARQL é um acrônimo recursivo que significa *SPARQL Protocol and RDF Query Language*. É uma linguagem utilizada para realizar consultas em documentos RDF padronizada pelo W3C.

Na Figura 7 é exposta a estrutura básica de uma consulta SPARQL onde é possível notar um certo grau de semelhança com a linguagem SQL, principalmente nas palavras reservadas.

**Figura 7** – Estrutura de consulta SPARQL.

```
# declaracao do prefix (abreviacao das URIs)
PREFIX foo:
...
# declaracao dos acessos ao grafos
FROM ...
# parametros a serem encontrados
SELECT ...
# query pattern
WHERE {
    ...
}
# organizador do resultado (ordenacao)
ORDER BY ...
```

**Fonte** – Oliveira (2011).

Na Figura 8 é apresentado um exemplo de consulta SPARQL simples, onde se deseja como resultado todos os recursos e suas *homepages* em que o recurso possui o valor "Apollo", em sua propriedade *foaf:name*.

**Figura 8** – Exemplo de consulta SPARQL.

```
#Seleciona o sujeito e a homepage,  
# de um sujeito onde o nome seja "Apollo 7".  
  
PREFIX foaf:  
SELECT ?craft ?homepage  
{  
  ?craft foaf:name "Apollo 7" .  
  ?craft foaf:homepage ?homepage  
}
```

Fonte – Oliveira (2011).

#### 3.4.4 Fuseki

Fuseki é uma interface HTTP que funciona através de um *endpoint* como um servidor Web para dados RDF com suporte a consultas SPARQL. É um subprojeto do Apache Jena e é desenvolvido como *servlet*.

### 3.5 Twitter API

A API do Twitter é composta por um conjunto de serviços baseados na arquitetura *Representational State Transfer* (REST) onde as mensagens trocadas são encapsuladas no protocolo HTTP. Essa API permite incorporar funcionalidades da rede social na aplicação, como: buscar *tweets* que contenham um termo ou publicar um *tweet*.

Segundo Xavier (2012), a API do Twitter pode ser dividida em três partes:

- a) REST API: Esta é a principal API. É a que possui mais serviços e destina-se à manipulação dos dados dos usuários e conexões entre eles, bem como ao envio de mensagens;
- b) *Search* API: A API de busca disponibiliza serviços para pesquisa de mensagens e usuários;
- c) *Streaming* API: Esta API possui uma arquitetura um pouco diferenciada das duas primeiras. Destina-se a gerar conexões persistentes para troca de informações de forma síncrona. Isso é útil em sistemas *desktop*, ou que necessitem atualizar o histórico de mensagens constantemente.

## 4 TRABALHOS CORRELATOS

Os trabalhos relacionados abordados neste capítulo são discutidos de forma breve e objetiva trazendo sempre uma comparação direta com a solução proposta neste trabalho.

### 4.1 Operação Serenata de Amor

A Operação Serenata de Amor, segundo Musskopf (2016, on-line), é

um projeto de tecnologia que usa inteligência artificial para auditar contas públicas e auxiliar no controle social. [...] Assim surgiu a Operação Serenata de Amor, focada em fiscalizar, com auxílio de tecnologia, os reembolsos efetuados pela Cota para Exercício da Atividade Parlamentar (CEAP) – verba que custeia alimentação, transporte, hospedagem e até despesas com cultura e assinaturas de TV dos parlamentares.

Para isso, o projeto foca nos gastos públicos, encontrando inconsistências, tais como: almoços simultâneos em estados diferentes pelo mesmo parlamentar ou refeição de R\$ 600,00 em restaurante onde o kg é R\$ 50,00. O projeto também permite o envio de mensagens via Twitter a fim de publicizar as inconsistências encontradas.

O Saiba mais Brasil explora outra vertente da política, as proposições criadas e votadas pelos parlamentares, ambas importantes. Os dois fazem uso do Twitter, contudo, no Serenata de Amor há o foco nas divergências encontradas e sempre que algo novo é encontrado isso é informado. Todavia, no Saiba mais Brasil é dado ao usuário a opção de realizar ou não a postagem com o resultado da sua análise.

### 4.2 Deliberatório

Deliberatório, segundo Brígida e Brito (2013, on-line), "é um jogo de cartas gerado a partir dos dados abertos da Câmara dos Deputados Federais". Este jogo simula o processo de discussão e deliberação das proposições, estimulando a sensibilização e mobilização social para o acompanhamento dos canais de transparência e das informações públicas. O jogo termina quando algum jogador conseguir deliberar 5 proposições.

Tanto o Deliberatório quanto o Saiba mais Brasil focam no uso das proposições. No entanto, o Deliberatório objetiva em o usuário aprovar suas proposições para ganhar o jogo, concordando ou não com elas. Já no Saiba mais Brasil há uma reflexão sobre o conteúdo das proposições e o assunto tratado por elas. Além disso, no Deliberatório, o jogador não tem ciência de como seu representante votou em determinada deliberação a fim de deduzir se está ou não sendo representado por ele, o que é possível de saber na solução proposta.

### 4.3 Radar Parlamentar

Radar Parlamentar é uma ferramenta que tem foco no desenvolvimento de um mapa dinâmico de semelhança de partidos. Ele ilustra as semelhanças entre partidos políticos com base na análise matemática dos dados de votações que ocorrem na casa legislativa. As semelhanças são apresentadas em um gráfico bidimensional, em que círculos representam partidos ou parlamentares, e a distância entre esses círculos representa o quão parecido os mesmos votam. Realiza também análises da participação de parlamentares com a intenção de observar as participações feminina e masculina. Por exemplo, mostra o perfil dos partidos de acordo com o gênero (LEITE, 2012).

O sistema Radar Parlamentar analisa o perfil e mostra graficamente a similaridade das respostas das votações entre os próprios parlamentares ou partidos. O Saiba mais Brasil indica ao cidadão interessado, se o deputado selecionado na análise tem comportamento semelhante ao seu, de acordo com as respostas das votações.

### 4.4 Sr. Cidadão

O Sr. Cidadão é um aplicativo móvel para Android que agrupa vários dados públicos para o acompanhamento por parte do cidadão e oferece a funcionalidade de monitoramento do político. O usuário seleciona o estado, o partido e o deputado e então o App passa a enviar notificações relacionadas a este, como ausência nas sessões e como votou nas proposições. O aplicativo é gratuito, possui a funcionalidade de compartilhamento nas redes sociais e possui também um site com o *ranking* dos deputados com mais curtidas (SR..., 2017).

O maior diferencial do Saiba mais Brasil em comparação com o Sr. Cidadão, é a disponibilização aberta dos dados resultantes das análises para consultas semânticas. Outra diferença, é que o Saiba mais Brasil se concentra apenas nas proposições e votos apresentando assim um escopo mais definido além de ser desenvolvido para outras plataformas além do Android.

### 4.5 Monitora, Brasil!

O Monitora, Brasil!, reúne várias informações importantes como presença, gastos, projetos e contatos dos deputados federais e senadores em um aplicativo móvel com versões para Android e iOS. O App utiliza dados extraídos dos sites da Câmara dos Deputados, Senado Federal, TSE e Transparência Brasil e oferece a funcionalidade de monitorar e comparar os parlamentares além de exibir um *ranking* de deputados por gastos entre outras funções (MONITORA..., 2015).

Além da disponibilização dos dados das análises para consultas semânticas, o Saiba mais Brasil oferece a funcionalidade de comparação entre os votos do usuário e

os votos de um deputado e também a possibilidade de acompanhamento, por parte do deputado, do seu desempenho por meio das publicações no Twitter. Outra diferença do Saiba mais Brasil quando comparado ao Monitora, Brasil! é o foco, que no Saiba mais Brasil se limita às proposições e votos dos deputados federais.

#### 4.6 Meu Deputado

Meu Deputado é um aplicativo móvel para Android e iOS onde é possível navegar em uma lista com todos os deputados federais em exercício e acompanhar seus gastos detalhados por categorias, acompanhar seu posicionamento em algumas proposições de maior visibilidade e comparar estes quesitos entre dois ou três deputados. O App é gratuito e possui também um *ranking* mensal de deputados por gastos (MEU... , 2015).

Embora o App Meu Deputado traga o posicionamento dos deputados em alguns projetos, não traz todas as proposições votadas na Câmara dos Deputados, como o Saiba mais Brasil. Além disso, o Saiba mais Brasil oferece a funcionalidade em que o usuário executa a ação de votar gerando um resultado e disponibiliza abertamente esses dados organizados para consultas semânticas.

#### 4.7 Eu Parlamentar

Eu Parlamentar é um aplicativo móvel para Android com uma finalidade bem definida: listar os deputados federais em ordem de representatividade com o usuário. O App solicita o estado do usuário e exibe 10 proposições votadas na Câmara dos Deputados. A partir da comparação dos votos, o App apresenta um *ranking* com os deputados com posicionamentos mais parecidos com o usuário no topo da lista (EU... , 2018).

Enquanto o Eu Parlamentar analisa apenas 10 questões para julgar o grau de representatividade, o Saiba mais Brasil oferece para o usuário todas as proposições votadas na Câmara e oferece ainda um filtro para que o usuário analise propostas de um determinado tema.

#### 4.8 Poder do Voto

Com lançamento previsto pra maio deste ano, esse aplicativo para *smartphones* não está pronto ainda, mas está ganhando matérias pela Internet, em parte por sua relevância no cenário político brasileiro, mas principalmente porque o idealizador do projeto, o executivo Mario Mello, decidiu colocá-lo em prática ao deixar o cargo de diretor-geral do PayPal na América Latina. O App será gratuito e oferecerá a funcionalidade de acompanhamento de deputados e senadores além de permitir que o usuário manifeste como deseja que um parlamentar vote em determinada proposta (TAVARES, 2018).

Tavares (2018, on-line) explica:

Diferentemente de aplicativos similares - como o Monitora, Brasil! e o Sr. Cidadão, que também possibilitam o monitorar deputados -, o Poder do Voto terá como principal guia a pauta de votações do Congresso na semana. "A ideia é criar um debate sobre as leis do momento, para que a população atue na construção junto com deputado", diz o autor do app.

Enquanto o Poder do Voto utiliza as proposições de pautas futuras com o intuito de debater a proposta antes da votação, o Saiba mais Brasil analisa proposições já votadas além de disponibilizar abertamente os dados obtidos por meio dos resultados.

#### **4.9 Using ICTs to create a culture of transparency: e-government and social media as openness and anti-corruption tools for societies**

Este trabalho explora os impactos das TICs sobre as culturas anticorrupção e de transparência, especialmente relacionadas à implantação do Governo Eletrônico (E-government). Para isso, o trabalho foca no uso de mídia social para a publicidade de informações, como: Wikileaks, um site que permite aos usuários publicar anonimamente informações confidenciais, e um website criado pelo Instituto Nacional Democrata para ajudar os cidadãos a explorar, analisar e visualizar os dados associados à eleição presidencial de 2009 no Afeganistão (BERTOT; JAEGER; GRIMES, 2010).

Enquanto este trabalho foca na divulgação pública de dados, para auxiliar o combate à corrupção, o Saiba mais Brasil parte do pressuposto que esses dados já estejam disponíveis e faz uso desses dados. Neste trabalho indica-se que os principais desafios para o sucesso de iniciativas que usam TICs são a necessidade de alfabetização tecnológica, usabilidade e acessibilidade. O Saiba mais Brasil oferece dados filtrados e adequados aos usuários, de forma que possa auxiliá-lo em tomadas de decisão.

#### **4.10 E-government application in combating corruption: China's case**

Este trabalho propõe um sistema anticorrupção baseado em um centro de dados e quatro plataformas responsáveis por publicar, monitorar, avisar e compartilhar as informações governamentais. O objetivo é disponibilizar um tipo de rede integrada com a geração, entrega, compartilhamento, análise e feedback de informações para o suporte à tomada de decisões e à execução anticorrupção (ZHANG; ZHANG; WANG, 2011).

Este trabalho foca em disponibilizar um sistema completo e amplo que cobre desde a solução da base de dados até os sistemas periféricos de informação, monitoramento e análise. O Saiba mais Brasil explora os dados públicos já disponibilizados e analisa-os de acordo com o *feedback* do cidadão.

#### 4.11 Quadro comparativo entre trabalhos relacionados

Na Figura 9 é apresentada uma tabela comparativa entre o Saiba Mais Brasil e os trabalhos relacionados que trazem um aplicativo móvel.

**Figura 9** – Quadro comparativo entre trabalhos relacionados.

	ANO	RESULTADOS EM RDF	PROPOSIÇÕES VOTADAS	PROPOSIÇÕES A VOTAR	COMPARA DEPUTADOS	COMPARA DEPUTADO COM USUÁRIO	FEEDBACK PARA O DEPUTADO
<b>SAIBA MAIS BRASIL</b>	2018	SIM	SIM	NÃO	NÃO	SIM	SIM
<b>SR CIDADÃO</b>	2017	NÃO	SIM	NÃO	NÃO	NÃO	NÃO
<b>MONITORA, BRASIL!</b>	2015	NÃO	SIM	NÃO	SIM	NÃO	NÃO
<b>MEU DEPUTADO</b>	2015	NÃO	SIM*	NÃO	SIM	NÃO	NÃO
<b>EU PARLAMENTAR</b>	2018	NÃO	SIM*	NÃO	NÃO	SIM	NÃO
<b>PODER DO VOTO</b>	2018	NÃO	NÃO	SIM	NÃO	SIM	SIM

Fonte – Elaborada pelo autor.

## 5 SAIBA MAIS BRASIL

Neste capítulo, são detalhadas as etapas do desenvolvimento da ferramenta Saiba mais Brasil, mostrando uma visão geral sobre como o sistema foi desenvolvido, seus requisitos funcionais e não-funcionais e diagramas de casos de uso, de classes e de atividades.

A solução proposta se baseia no desenvolvimento de um aplicativo móvel capaz de gerar dados semânticos a partir de comparações feitas entre o voto de um usuário e o voto de um deputado em uma proposição que foi votada na Câmara dos Deputados recentemente.

### 5.1 Requisitos Funcionais

Um único ator se relaciona com o sistema:

- a) usuário: O usuário é a única entidade a atuar no sistema e não existe distinção de perfil ou permissões.

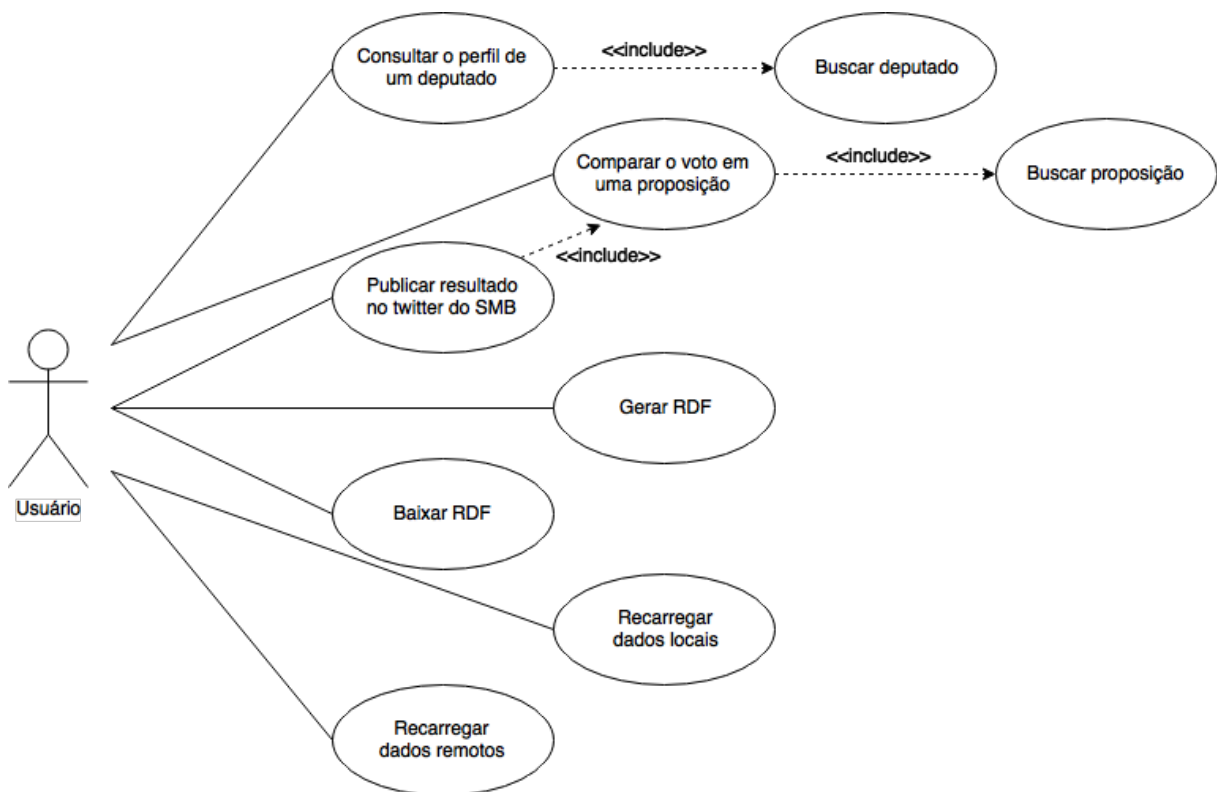
Os requisitos funcionais descrevem as funcionalidades que o sistema oferece para o usuário.

- a) consulta de deputados (RF1): O sistema disponibiliza uma consulta onde os deputados podem ser pesquisados por partido, por estado ou pelo nome. O perfil de cada deputado apresenta a sua foto, nome civil, escolaridade, data de nascimento, informações para contato entre outras;
- b) comparação do voto do usuário com o voto de um deputado (RF2): O sistema disponibiliza uma consulta de proposições que podem ser filtradas por tema e por partido, estado ou nome do autor e permite que o usuário vote e saiba o resultado da comparação;
- c) publicação do resultado no Twitter (RF3): Após o resultado, o sistema disponibiliza a opção de publicar um tweet no perfil do Saiba mais Brasil com o nome do deputado, a porcentagem de correspondência e a *hashtag* *#representa* ou *#naorepresenta*. O deputado analisado será marcado na publicação caso tenha um perfil na rede social;
- d) gerar RDF (RF4): Por meio de um *endpoint* o sistema disponibiliza uma rota para geração do arquivo RDF;
- e) baixar RDF (RF5): Por meio de um *endpoint* o sistema disponibiliza uma rota para baixar o arquivo RDF;

- f) recarregar dados locais (RF6): O sistema disponibiliza a opção de apagar os dados locais e em seguida realimenta o banco de dados local com as informações mais atuais e livre de possíveis erros;
- g) recarregar dados remotos (RF7): O sistema disponibiliza a opção de apagar os dados remotos e em seguida realimenta o banco de dados remoto com as informações mais atuais e livre de possíveis erros.

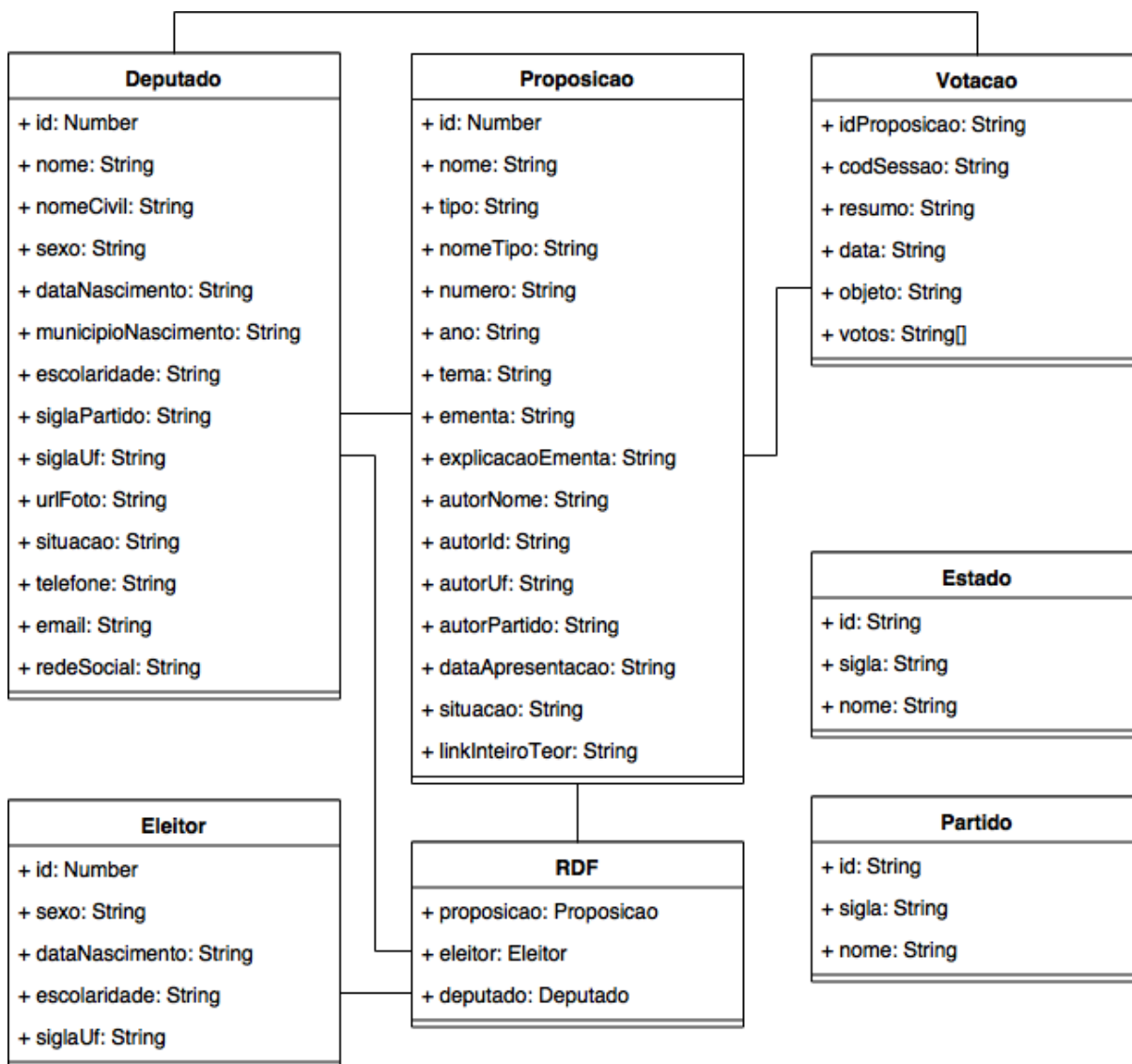
Considerando os requisitos descritos acima, é possível modelar as funcionalidades do sistema proposto tal como segue. Na Figura 10 é apresentado o diagrama de casos de uso do sistema e na Figura 11 o diagrama de classes.

**Figura 10** – Diagrama de Casos de Uso do Saiba mais Brasil.



Fonte – Elaborada pelo autor.

Figura 11 – Diagrama de Classes do Saiba mais Brasil

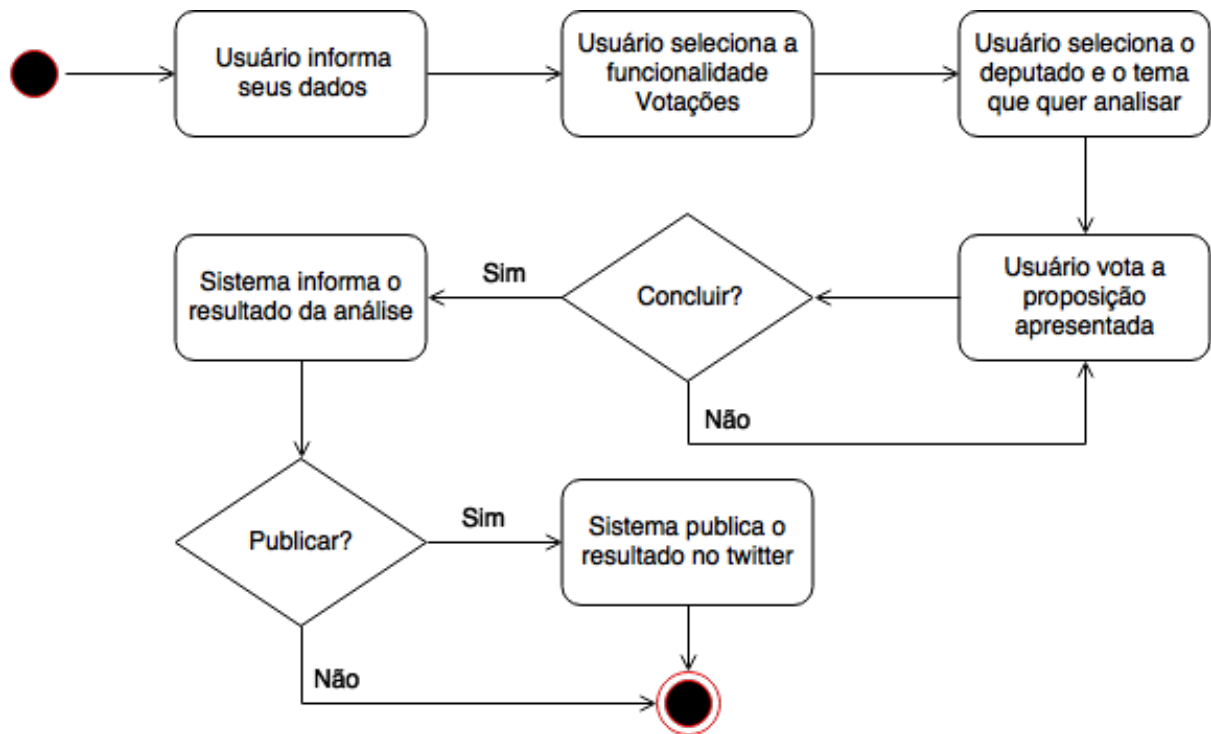


Fonte – Elaborada pelo autor.

O diagrama de classes apresentado na Figura 11 descreve, além dos relacionamentos entre as classes, os atributos de cada entidade, que são as propriedades recuperadas por meio das requisições à API dos dados abertos do governo, com exceção das classes Eleitor, que foi modelada de acordo com as informações que são solicitados ao usuário, e RDF, que possui todos os atributos relevantes que serão disponibilizados para as consultas semânticas.

Na Figura 12 são descritas as atividades que compõem o processo de comparação do voto do usuário com o voto de um deputado (RF2) e o processo de publicação do resultado no Twitter (RF3).

Figura 12 – Fluxograma do Saiba mais Brasil (RF2) e (RF3)



Fonte – Elaborada pelo autor.

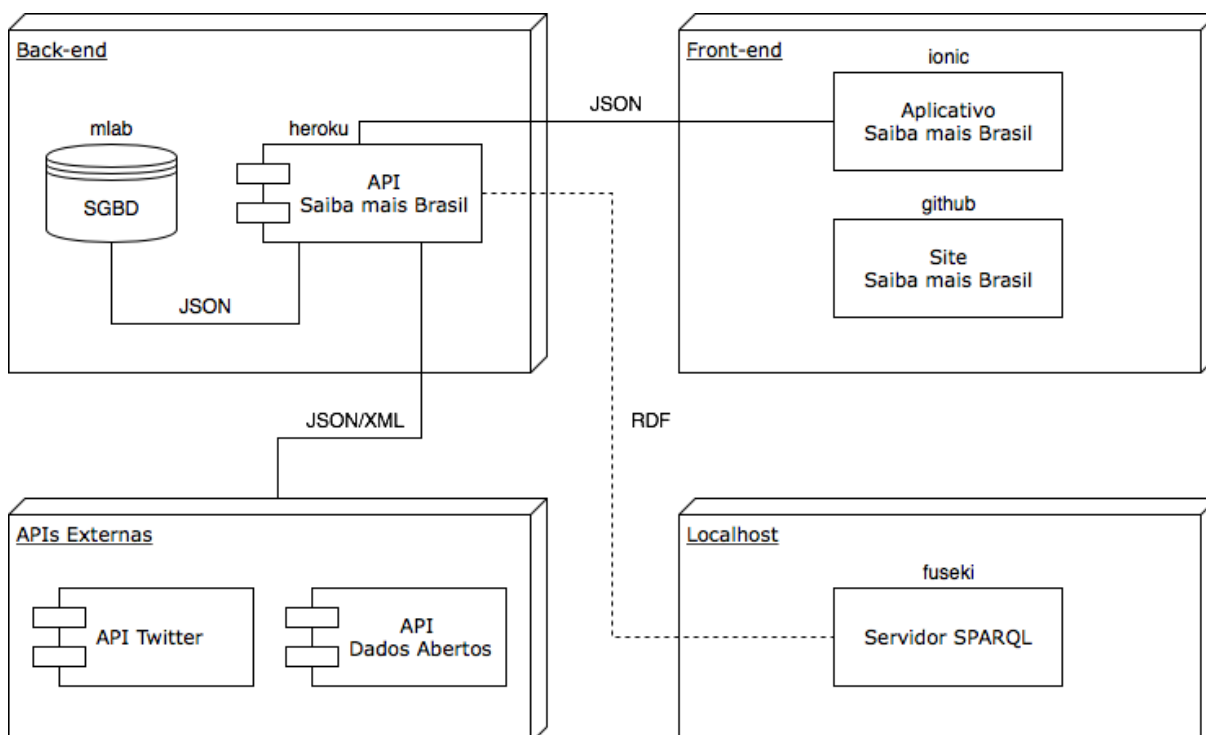
## 5.2 Requisitos Não-funcionais

- desempenho (RN1): Por se tratar de desenvolvimento para plataforma móvel onde os recursos não são abundantes, as tarefas que exigem maior processamento e memória foram delegadas para o servidor back-end que disponibiliza esses serviços por meio da API;
- segurança (RN2): Todas as APIs do sistema expostas como serviços web poderão ser acessadas por qualquer pessoa, já que os dados manipulados são abertos e as informações geradas a partir dos usuários são anônimas;
- usabilidade (RN3): Toda a aplicação foi desenvolvida de forma responsiva e projetada exclusivamente para a plataforma móvel e nenhuma adaptação para esse fim foi necessária;
- compatibilidade (RN4): Para este trabalho a aplicação foi desenvolvida para os sistemas Android e iOS. Porém o *framework* de desenvolvimento permite compilar versões para outras plataformas.

### 5.3 Arquitetura

Para que tudo funcione de forma eficiente, a arquitetura descrita na Figura 13 foi usada como base para o projeto, onde o fraco acoplamento entre o *front-end*, *back-end* e a persistência dos dados garante uma ótima escalabilidade e flexibilidade.

**Figura 13** – Arquitetura do Saiba mais Brasil.



**Fonte** – Elaborada pelo autor.

### 5.4 Persistência de Dados

Devido ao grande volume de dados indispensáveis para o funcionamento do App, que inclui informações dos deputados, partidos, proposições, votações e etc, e também por sua performance, o MongoDB foi escolhido para atuar na persistência dos dados do Saiba mais Brasil.

O MongoDB foi criado com o Big Data em mente. Ele suporta muito bem o escalonamento além de trabalhar com dados distribuídos tornando-se uma opção muito interessante para grandes volumes de dados, inclusive os desestruturados. O MongoDB é o mecanismo de persistência perfeito para uso com tecnologias que trabalham com JSON nativamente, como o NodeJS.

Além disso, do ponto de vista do desenvolvedor, o MongoDB permite criar uma pilha completa apenas com JavaScript uma vez que temos JavaScript no lado do

cliente, do servidor (com NodeJS) e do banco de dados (com MongoDB), pois todas as informações e *queries* são criadas usando JavaScript.

O trecho de código apresentado na Figura 14 mostra um documento da coleção deputados, que seria o equivalente a uma linha de uma tabela de um banco de dados relacional, apresentado na forma exata (objeto JSON) em que é armazenado no banco de dados.

**Figura 14** – Documento da coleção Deputados

```
{
  "_id": {
    "$oid": "5a907f82f56cb9001417827e"
  },
  "id": 74374,
  "nome": "GERALDO RESENDE",
  "nomeCivil": "GERALDO RESENDE PEREIRA",
  "sexo": "M",
  "dataNascimento": "1955-04-20",
  "municipioNascimento": "Córrego Danta",
  "escolaridade": null,
  "siglaPartido": "PSDB",
  "siglaUf": "MS",
  "urlFoto":
  "http://www.camara.leg.br/internet/deputado/bandep/74374.jpg",
  "situacao": "Exercício",
  "telefone": "3215-5905",
  "email": "dep.geraldoresende@camara.leg.br",
  "redeSocial": ["http://twitter.com/dep_geraldo"]
}
```

**Fonte** – Elaborada pelo autor.

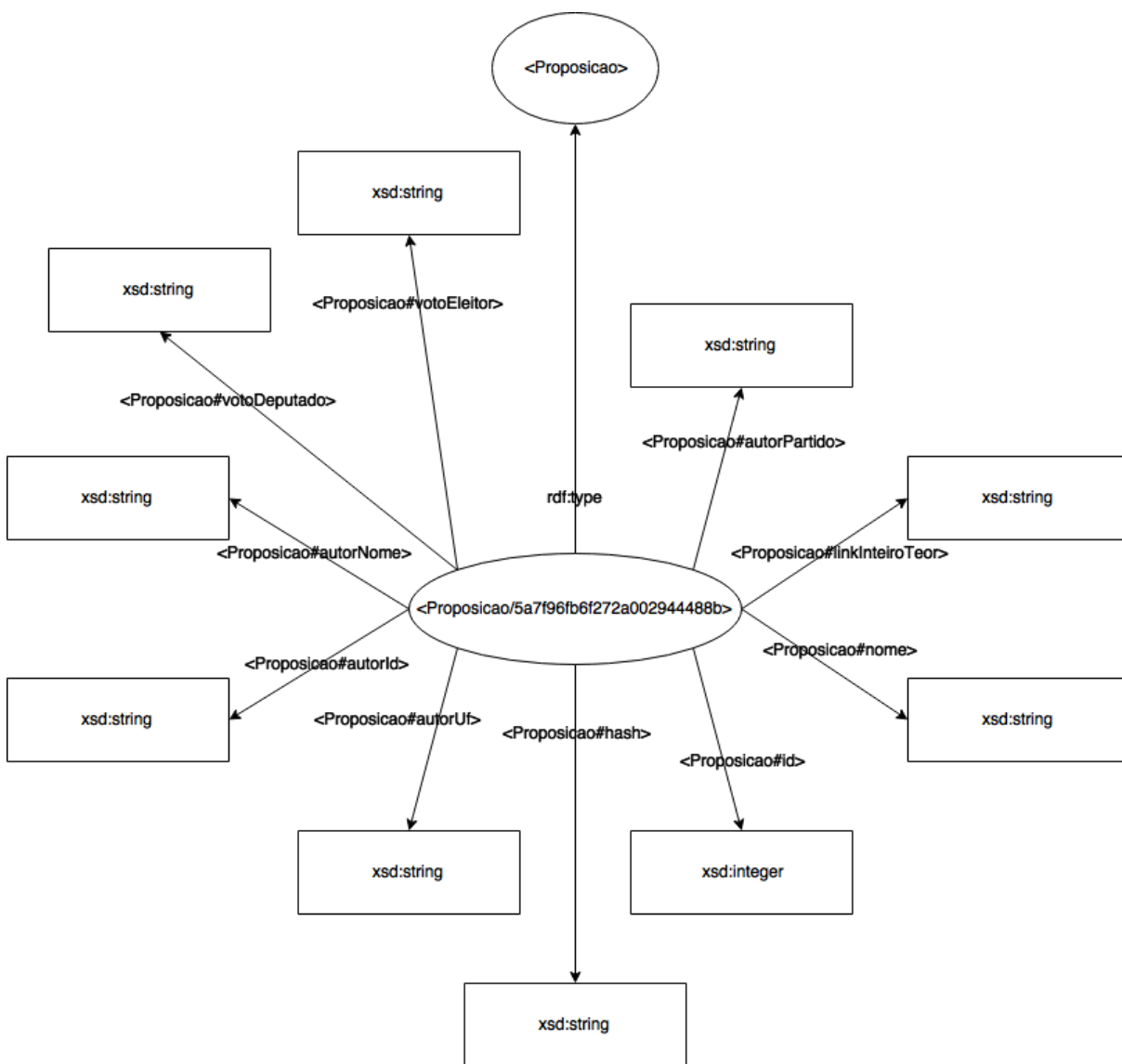
Todas as coleções e propriedades dos documentos estão descritas no diagrama de classes na Figura 11.

Com uma instância do MongoDB rodando na nuvem por meio do serviço mLab, tem-se a garantia de um alto grau de flexibilidade pois qualquer modificação nesse serviço não compromete a API ou o App. Por exemplo, é possível alterar a capacidade de armazenamento apenas da instância onde roda o MongoDB simplesmente mudando de plano ou de serviço (para o Amazon AWS, por exemplo) e assumindo os custos de acordo com a necessidade.

5.4.1 Dados Semânticos

Os dados gerados a partir das análises feitas no App do Saiba mais Brasil são disponibilizados abertamente no modelo RDF como um serviço da API. Os dados estão organizados de acordo com a ontologia descrita no gráfico exibido na Figura 28 do Apêndice A. Na Figura 15 é apresentada apenas a parte da ontologia referente às propriedades das proposições.

**Figura 15** – Parte do gráfico da ontologia dos dados do Saiba mais Brasil



Fonte – Elaborada pelo autor.

Esses dados podem ser utilizados para fazer consultas SPARQL para obter diversos resultados, por exemplo:

- a relação entre o sexo do usuário e o sexo do deputado com mais correspondências;

- a faixa etária e estado dos deputados com mais correspondências;
- a média de correspondências para eleitores com ensino superior e de um determinado sexo.

O trecho de código apresentado na Figura 16 é um exemplo de consulta SPARQL que retorna a quantidade de proposições analisadas e a quantidade de correspondências, agrupadas pelo sexo do usuário e pelo sexo do deputado.

Figura 16 – Exemplo de consulta SPARQL

```

SELECT ?sexoEle ?sexoDep (SUM(?sCount) as ?count) (SUM(?sCount2) as ?match) {
  {
    SELECT ?sexoDep ?sexoEle (COUNT(?s) as ?sCount) (0 as ?sCount2) WHERE {
      ?s a <http://smb.com/Proposicao> .
      ?s <http://smb.com/Proposicao#hash> ?hash .
      ?s <http://smb.com/Proposicao#votoDeputado> ?vDep .
      ?s <http://smb.com/Proposicao#votoEleitor> ?vEle .
      ?s <http://smb.com/Proposicao#linkInteiroTeor> ?o .
      ?deputado a <http://smb.com/Politico> .
      ?deputado <http://smb.com/Politico#sexo> ?sexoDep .
      ?deputado <http://smb.com/Politico#hash> ?hash .
      ?eleitor a <http://smb.com/Eleitor> .
      ?eleitor <http://smb.com/Eleitor#sexo> ?sexoEle .
      ?eleitor <http://smb.com/Eleitor#hash> ?hash
    }
    GROUP BY ?sexoDep ?sexoEle
  } UNION {
    SELECT ?sexoDep ?sexoEle (0 as ?sCount) (COUNT(?s) as ?sCount2) WHERE {
      ?s a <http://smb.com/Proposicao> .
      ?s <http://smb.com/Proposicao#hash> ?hash .
      ?s <http://smb.com/Proposicao#votoDeputado> ?vDep .
      ?s <http://smb.com/Proposicao#votoEleitor> ?vEle .
      ?s <http://smb.com/Proposicao#linkInteiroTeor> ?o .
      ?deputado a <http://smb.com/Politico> .
      ?deputado <http://smb.com/Politico#sexo> ?sexoDep .
      ?deputado <http://smb.com/Politico#hash> ?hash .
      ?eleitor a <http://smb.com/Eleitor> .
      ?eleitor <http://smb.com/Eleitor#sexo> ?sexoEle .
      ?eleitor <http://smb.com/Eleitor#hash> ?hash

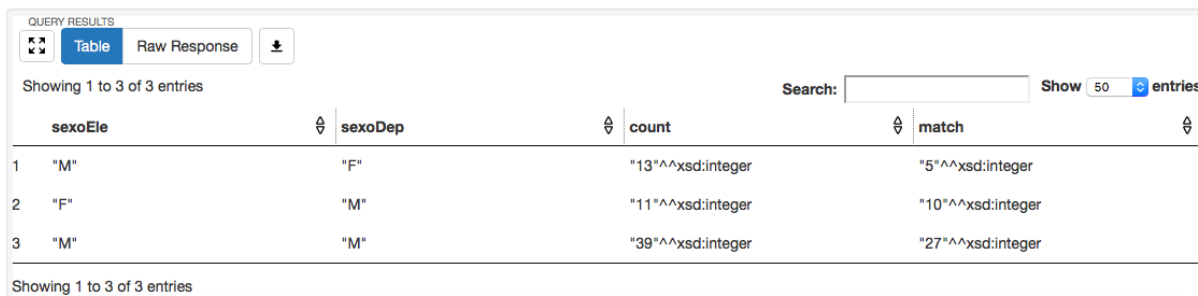
      FILTER regex(?vDep, ?vEle, "i")
    }
    GROUP BY ?sexoDep ?sexoEle
  }
}
GROUP BY ?sexoDep ?sexoEle ORDER BY ?sexoDep ?sexoEle

```

Fonte – Elaborada pelo autor.

O resultado dessa consulta é apresentado na Figura 17.

Figura 17 – Resultado da consulta SPARQL do exemplo.



	sexoEle	sexoDep	count	match
1	"M"	"F"	"13^^xsd:integer"	"5^^xsd:integer"
2	"F"	"M"	"11^^xsd:integer"	"10^^xsd:integer"
3	"M"	"M"	"39^^xsd:integer"	"27^^xsd:integer"

Fonte – Elaborada pelo autor.

Para realizar as consultas, é necessário ter os dados em RDF (disponível para download a partir da rota da API do Saiba mais Brasil especificada no Quadro 2 do Apêndice B) e um servidor SPARQL, como o Apache Jena Fuseki (disponível para download no site oficial), que roda a partir de um JAR com o Jena, o SPARQL *query engine* e outras ferramentas embutidas. O Fuseki é simples e possui uma interface web onde são feitas as consultas, as atualizações e o *upload* dos arquivos RDF diretamente para o *dataset*.

## 5.5 API (back-end)

A API Saiba mais Brasil é responsável por disponibilizar todos os serviços consumidos pelo App. Isso inclui as requisições feitas às APIs externas, consultas ao banco de dados e geração de dados RDF, tudo isso implementado em um servidor NodeJS.

A utilização do NodeJS em um projeto é facilmente justificada quando levado em consideração a integração com o banco de dados MongoDB. É tudo JavaScript!

Para esse cenário, de baixo acoplamento, existem vários serviços em nuvem para a hospedagem do *back-end* sendo difícil não mencionar o Elastic Beanstalk que faz parte do Amazon AWS, um dos pioneiros em computação em nuvem. Porém, para este trabalho, a opção escolhida foi o Heroku por possuir um plano gratuito e também porque o Heroku utiliza o EC2 do Amazon AWS para hospedar as aplicações. A URL base da API Saiba mais Brasil no Heroku é: <https://saibamaisbrasil.herokuapp.com>.

### 5.5.1 Rotas da API

As informações dos deputados, partidos, proposições e votações são extraídas da API dos Dados Abertos da Câmara dos Deputados. Algumas requisições a essa API retornam respostas em XML e outras em JSON. O servidor Saiba mais Brasil trata essas respostas ignorando dados desnecessários e alimenta o banco de dados. Sempre que necessário, é possível apagar esses dados e realimentar cada coleção da base de dados com as informações mais recentes por meio de rotas específicas da API Saiba mais Brasil.

A maioria das rotas da API Saiba mais Brasil segue um padrão simples e bastante intuitivo, onde o nome da coleção indica o modelo de documento que será retornado. Por exemplo:

- a rota `/api/deputados` retorna um *array* com todos os deputados;
- a rota `/api/deputados/:id` retorna um objeto com o deputado com o *id* passado como parâmetro;
- a rota `/api/deputadosrebase` apaga toda a coleção deputados, realimenta com os dados mais atuais e retorna `true` caso nenhum erro ocorra.

O mesmo ocorre com as rotas para as coleções partidos, estados, proposições e votações.

A coleção temas não exige uma rota de realimentação pois esta função é executada a partir da realimentação da coleção proposições.

A rota `/api/twitter` é responsável por publicar os *tweets*. Essa rota recebe dois objetos via POST: o deputado e o resultado da análise, e a partir daí elabora o texto marcando o deputado caso tenha um perfil no Twitter e publica o *tweet* na rede social.

A rota `/api/rdf` possui dois métodos HTTP:

- o método GET consulta todos os documentos da coleção rdfs do banco de dados, gera o arquivo RDF disponível publicamente para download e retorna `true` caso nenhum erro ocorra;
- o método POST recebe as informações da proposição analisada com os votos do deputado e do usuário, as informações do usuário e as informações do deputado, por fim salva esse documento na coleção rdfs do banco de dados.

A rota `/public/rdf.ttl` pode ser acessada por qualquer pessoa por meio de um navegador, mesmo sem o uso do App do Saiba mais Brasil. Essa rota é oferecida não para ser consumida pelo App, mas para disponibilizar os dados semânticos referentes às análises das proposições para qualquer um que queira utilizá-los, sendo essa funcionalidade uma parte importante dos objetivos deste trabalho.

Todas as rotas da API Saiba mais Brasil, assim como seus respectivos métodos HTTP, estão listados no Quadro 2 do Apêndice B.

## 5.6 Aplicativo (front-end)

O App Saiba mais Brasil é uma ferramenta que foi construída com o intuito de facilitar o acesso às informações dos deputados e votações da Câmara dos Deputados. Todos os dados estão disponíveis no portal dos dados abertos do governo, que inclusive incentiva a exploração desses dados através das TICs. Porém, além da possibilidade da interação do usuário com o sistema, o App Saiba mais Brasil traz informações organizadas de forma mais intuitiva e contribui assim para tornar o sistema político mais transparente.

O App foi desenvolvido com o Ionic Framework. Uma das principais características do Ionic é que seu desenvolvimento foi pensado em utilizar os recursos mais novos do CSS3, HTML5 e JavaScript com o objetivo de disponibilizar para o desenvolvedor uma variedade de componentes de qualidade e ótimo desempenho. Outra característica do Ionic é a produtividade que proporciona ao desenvolvimento, quando constrói versões do App para diferentes plataformas simultaneamente e sem a necessidade de escrever códigos nativos para um ou outro sistema operacional.

### 5.6.1 App

As figuras abaixo retratam como o App Saiba mais Brasil se mostra no sistema operacional iOS, que diferencia do Android apenas nos temas dos ícones e botões, pois seguem o padrão de cada plataforma e o próprio Ionic se encarrega de fazer isso.

Na Figura 18 é apresentada a tela principal, com botões para as principais funcionalidades que o App oferece para o usuário: Votações (RF2) e Deputados (RF1).

Figura 18 – Tela principal.



Fonte – Elaborada pelo autor.

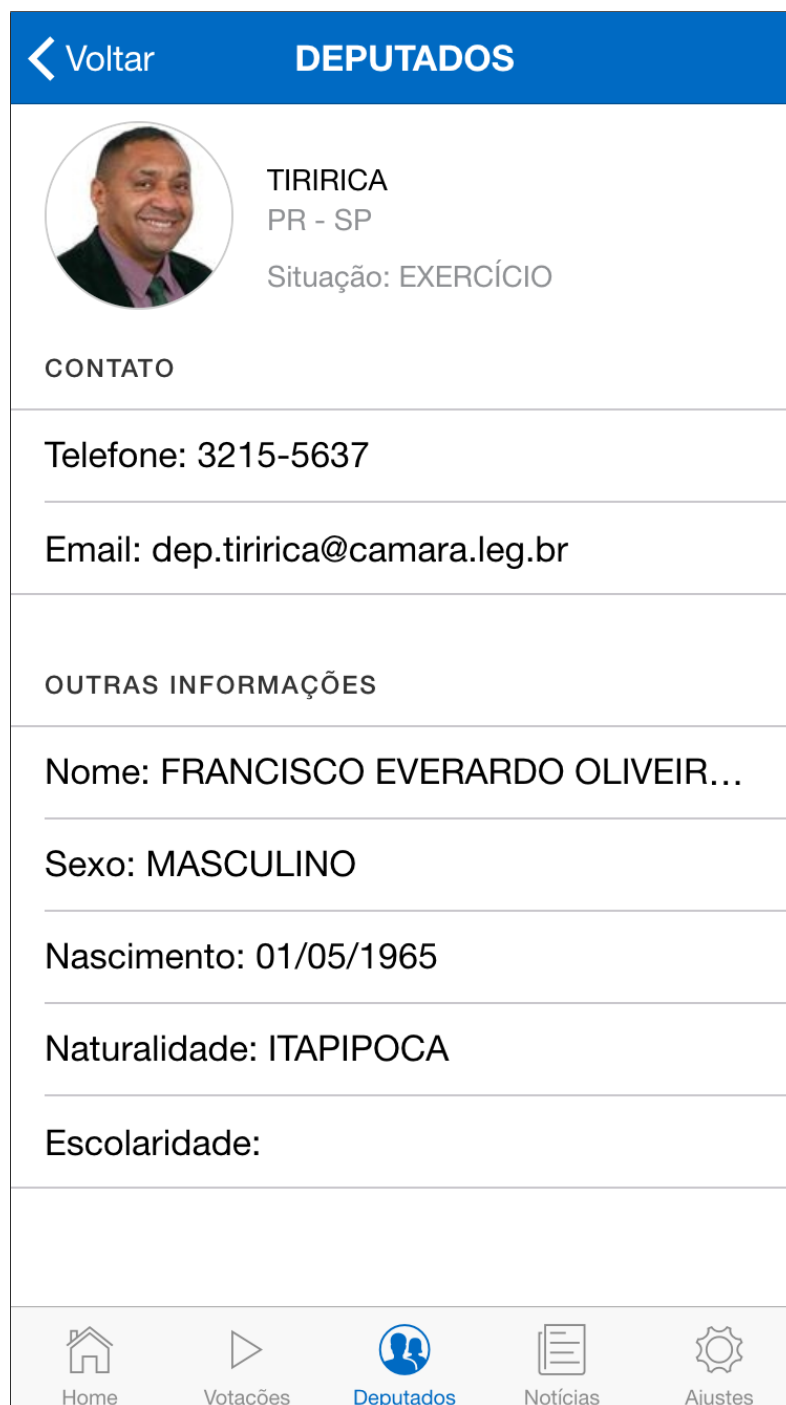
É possível observar também a barra de abas na parte inferior, que está presente em todas as telas.

Nas Figuras 19 e 20 são apresentadas as telas referentes à busca de deputados (RF1), que pode ser feita por partido, estado e por nome.

**Figura 19** – Tela de resultados da busca de deputados.








Fonte – Elaborada pelo autor.

**Figura 20** – Tela de informações do deputado selecionado.

**Fonte** – Elaborada pelo autor.

Nas Figuras 21 e 22 são apresentadas as telas referentes à busca e análise das proposições.

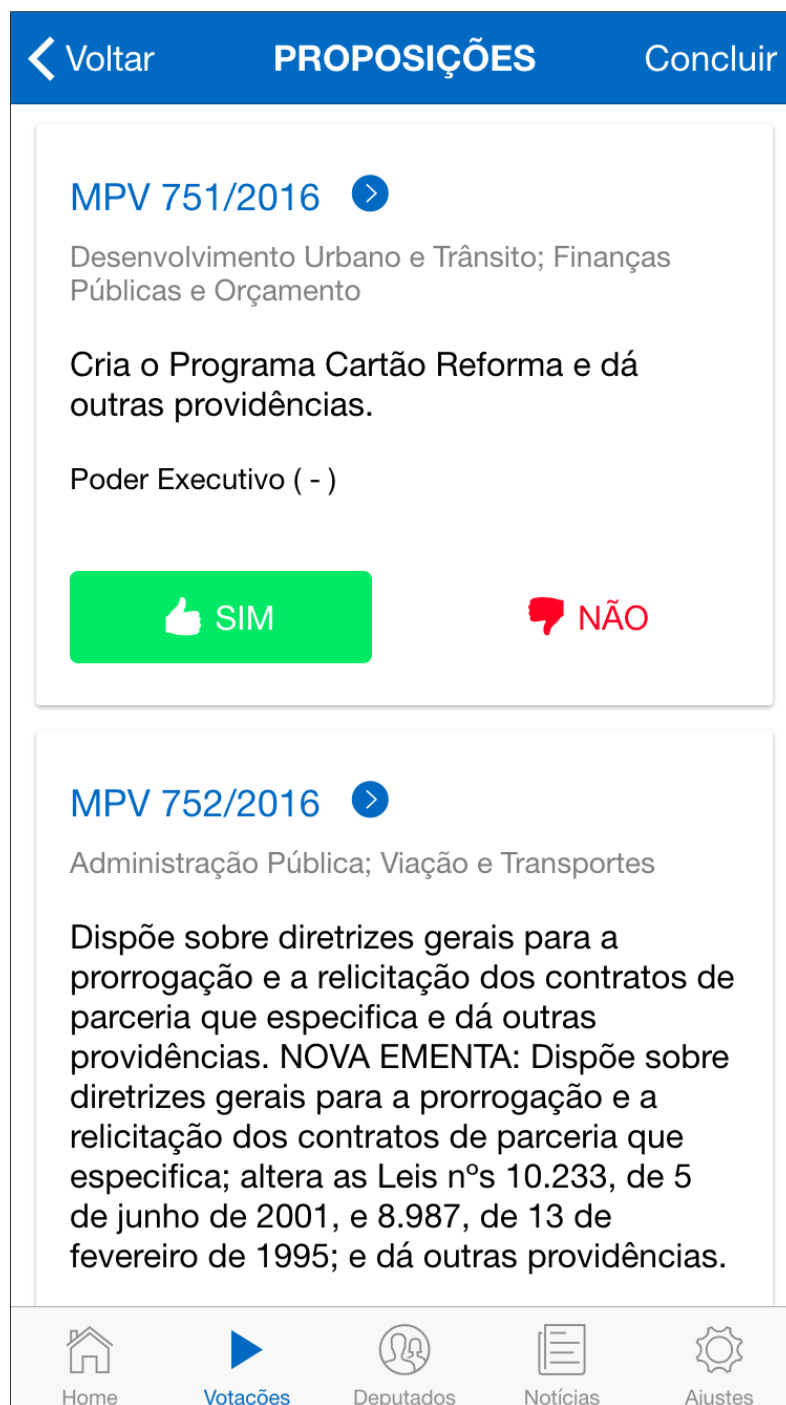
**Figura 21** – Tela de seleção de deputado e filtro de proposições.

VOTAÇÕES		Continuar
SEU REPRESENTANTE (OBRIGATÓRIO)		
Deputado	TIRIRICA (PR... ▼	
Selecione o deputado que você deseja comparar e analisar os votos nas proposições.		
PROPOSIÇÕES		
Tema	Educação ▼	
AUTOR		
Estado	São Paulo ▼	
Partido	Todos ▼	
Deputado	Todos ▼	
 Home		
 Votações		
 Deputados		
 Notícias		
 Ajustes		

**Fonte** – Elaborada pelo autor.

Na tela de seleção de deputado e filtro de proposições, mostrada na Figura 21, o usuário deve obrigatoriamente escolher um deputado para analisar. Já as proposições podem ser filtradas por tema e por autor da proposta.

Figura 22 – Tela de análise de proposições.

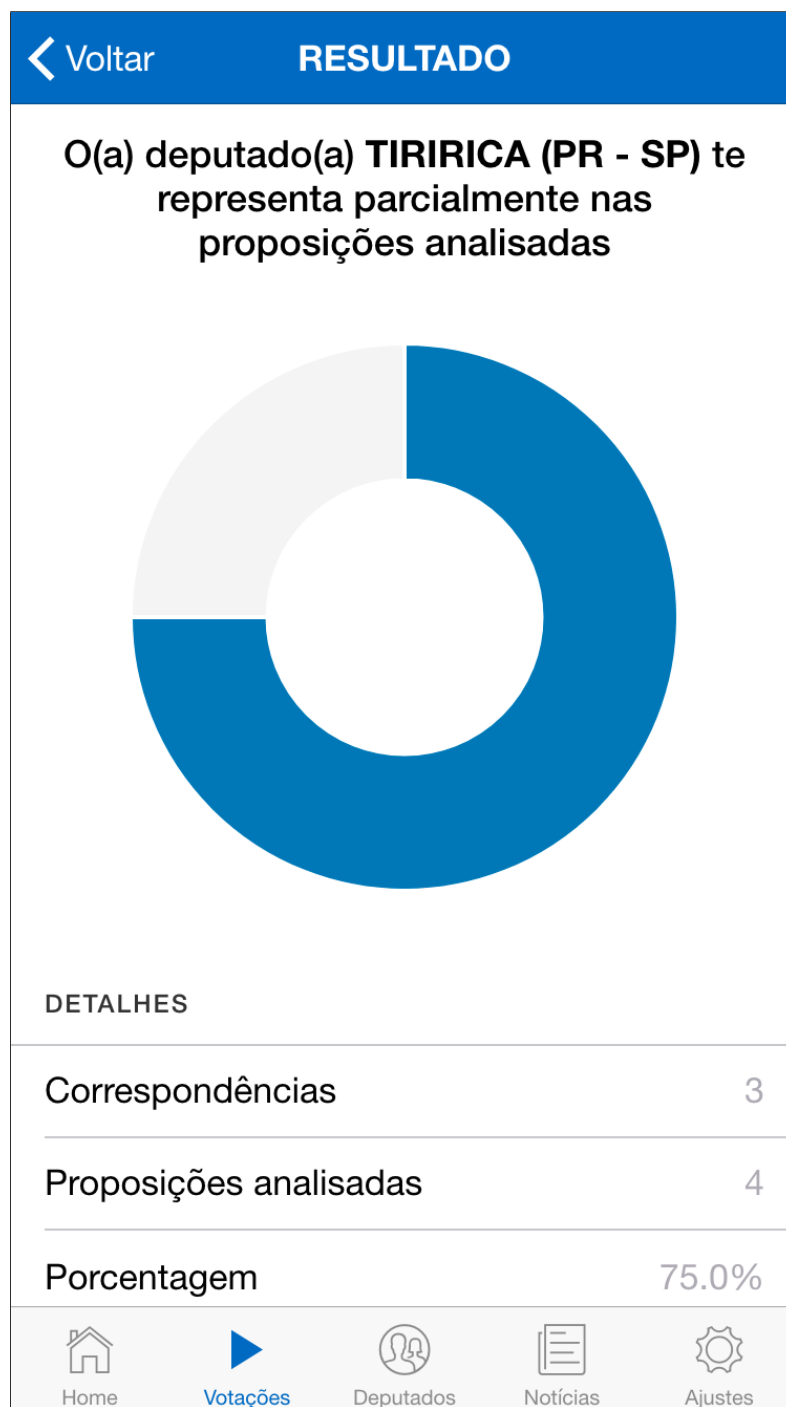


Fonte – Elaborada pelo autor.

Observa-se na tela de análise de proposições, mostrada na Figura 22, que para cada proposta são apresentados o nome, o link para o PDF com inteiro teor, os temas, a ementa e o autor.

Nas Figuras 23 e 24 são apresentadas as telas referentes aos resultados da análise do deputado nas proposições e também a tela do Twitter com a publicação.

Figura 23 – Tela de resultado da análise do deputado.



Fonte – Elaborada pelo autor.

Na tela de resultados, mostrada na Figura 23, um gráfico exibe a porcentagem da correspondência. Abaixo desse, é apresentada um resumo da análise com a quantidade de correspondências, quantidade de proposições analisadas e a porcentagem. Existe também, nessa tela, um botão para publicação do *tweet* com o resultado.

Figura 24 – Tela de publicação do resultado no Twitter.



Fonte – Elaborada pelo autor.

Na linha do tempo do Twitter, mostrada na Figura 24, cada *tweet* exibe o deputado analisado, marcado na publicação caso tenha um perfil no Twitter e também a porcentagem da representatividade. Além disso, as *hashtags* *#representa* e *#naorepresenta* são utilizadas de acordo com o resultado considerando as seguintes condições:

- *#representa*: porcentagem maior ou igual a 90%;
- *#naorepresenta*: porcentagem menor que 50%.

Algumas telas são apresentadas ao usuário apenas no primeiro acesso. Essas telas de introdução são retratadas nas Figuras 29, 30, 31 e 32 do Apêndice C.

O App na versão Android está disponível no site Saiba mais Brasil e pode ser baixado por meio do link: <https://saibamaisbrasil.github.io/site/smb.apk>.

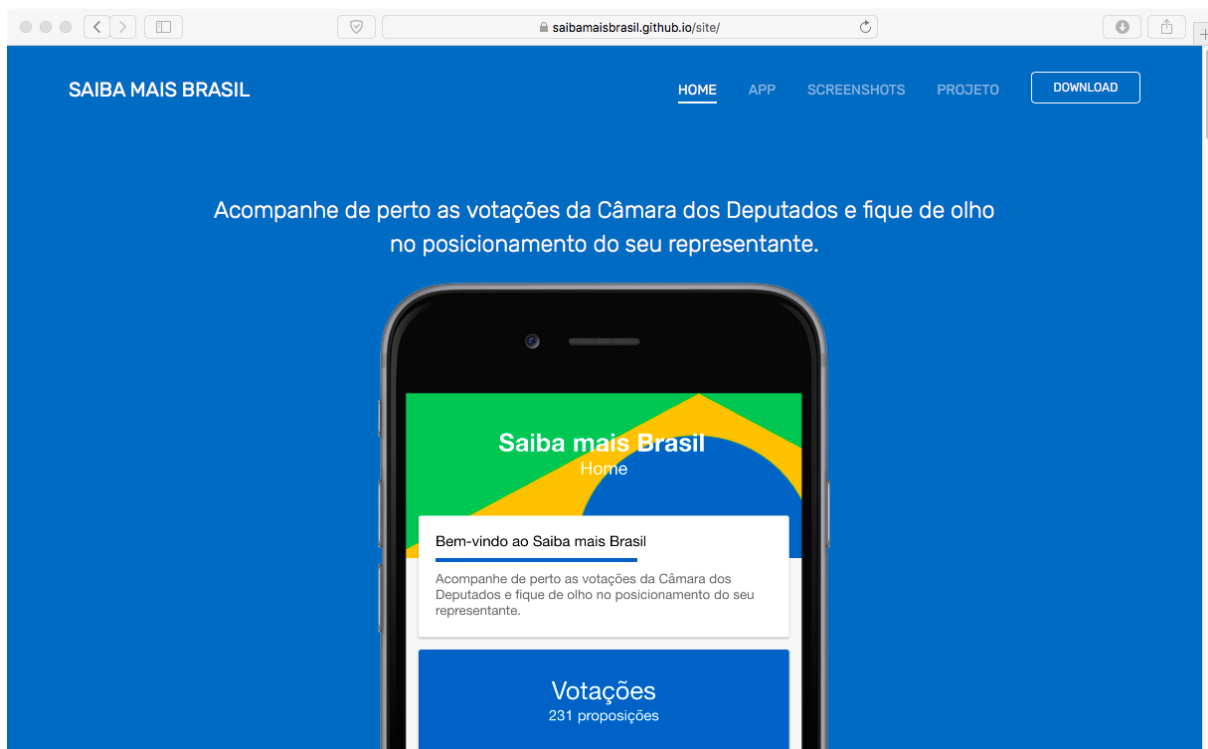
## 5.6.2 Site

Além do App Saiba mais Brasil, uma página web foi publicada através do github onde são encontrados o link para download do App, o tutorial, a galeria e também o link para o projeto no Github para eventuais contribuições por parte da comunidade.

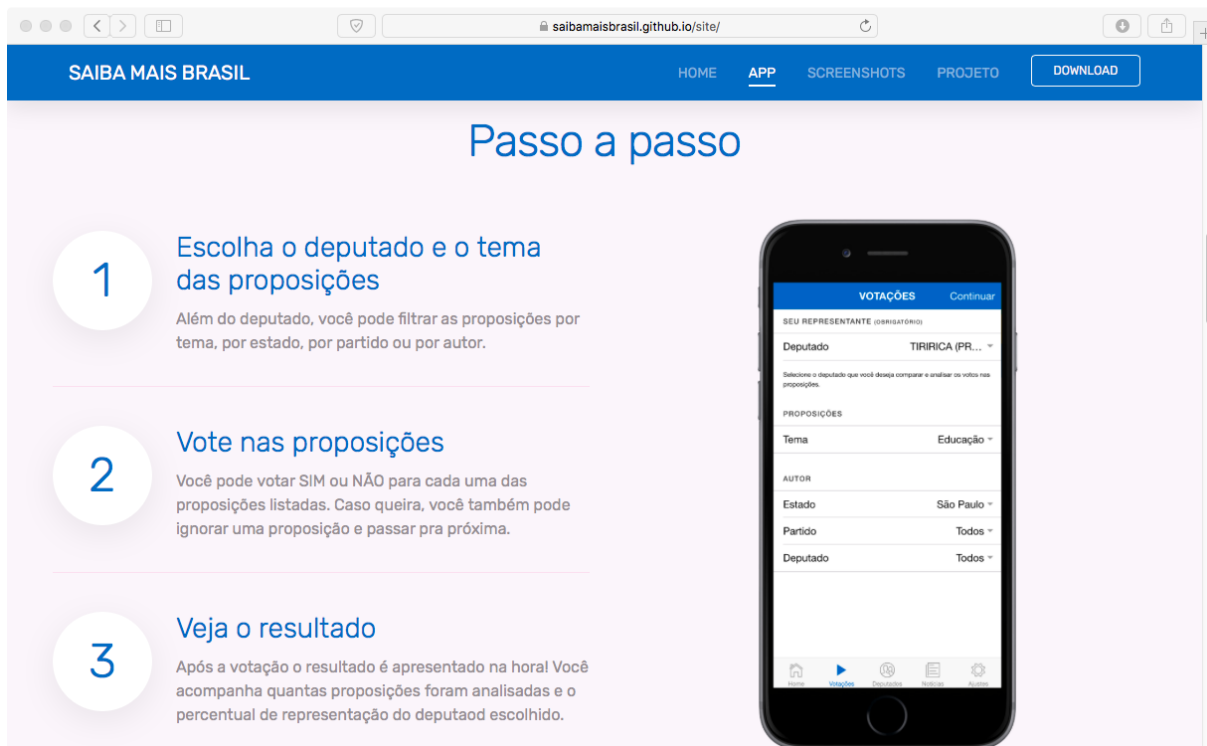
Nas Figuras 25 e 26 são mostradas algumas imagens do site no ar. Outras imagens do site estão no Apêndice D deste trabalho.

O site Saiba mais Brasil pode ser acessado através do link: <https://saibamaisbrasil.github.io/site/>.

**Figura 25** – Site Saiba mais Brasil (Home).



**Fonte** – Elaborada pelo autor.

**Figura 26** – Site Saiba mais Brasil (Passo a passo para uso do App).

**Fonte** – Elaborada pelo autor.

### 5.6.3 Twitter

O perfil do Saiba mais Brasil no Twitter foi criado exclusivamente para a publicação dos resultados (RF3). Além do perfil social, foi vinculada a esse uma conta de desenvolvedor onde foi criado o App com as devidas permissões para o acesso à API do Twitter.

O perfil do Saiba mais Brasil no Twitter pode ser acessado através do link: <https://twitter.com/saibamaisbrasil>.

Na Figura 27 é exibida uma captura de tela do perfil do Saiba mais Brasil.

Figura 27 – Perfil Saiba mais Brasil no Twitter.



Fonte – Elaborada pelo autor.

## 6 CONSIDERAÇÕES FINAIS

Pode-se dizer que o resultado deste trabalho foi positivo, considerando o êxito obtido no desenvolvimento da ferramenta proposta e nos demais objetivos que foram alcançados.

A ferramenta desenvolvida, Saiba mais Brasil, se mostrou eficiente para a finalidade a qual foi pensada. Além de atingir todos os objetivos estabelecidos, inclusive o de gerar dados para consultas semânticas em SPARQL a partir das análises, despertou ainda novas ideias que não estão no escopo deste trabalho mas que podem contribuir significativamente para aumentar ainda mais a relevância dessa ferramenta para a população.

### 6.1 Contribuições

A partir dos resultados deste trabalho, destacam-se as seguintes contribuições.

- a) Site (<https://saibamaisbrasil.github.io/site/>)
- b) Aplicativo (<https://saibamaisbrasil.github.io/site/smb.apk>)
- c) API (<https://saibamaisbrasil.herokuapp.com/>)
- d) Projeto e Documentação (<https://github.com/saibamaisbrasil>)

### 6.2 Trabalhos Futuros

As ideias que foram pensadas, e não incluídas no desenvolvimento deste trabalho, mas que podem ser implementadas futuramente são:

- a) versão Web para *desktop* do aplicativo Saiba mais Brasil com uma interface acessível pelo navegador e com mais funcionalidades, por exemplo: relatório do resultado com as proposições e os votos, *ranking* dos deputados com mais correspondências e etc;
- b) implantação de um servidor SPARQL com *endpoint* público e uma interface intuitiva para consultas semânticas onde usuários sem experiência com as linguagens de consulta possam fazer as buscas;
- c) simplificação da linguagem utilizada nas proposições, já que atualmente o texto apresentado é exatamente o da proposição original, onde muitos termos utilizados podem ser desconhecidos pelos usuários;
- d) incluir as proposições de outras casas legislativas e as votações de seus respectivos representantes, como deputados estaduais e vereadores.

## REFERÊNCIAS

(ACWG), G. A. corruption W. G. *G20 Anti-Corruption Open Data Principles*. Toronto, Ontário: Universidade de Toronto, 2015. Disponível em: <<http://www.g20.utoronto.ca/2015/G20-Anti-Corruption-Open-Data-Principles.pdf>>. Acesso em: 14 Jul 2017.

BERTOT, J. C.; JAEGER, P. T.; GRIMES, J. M. Using icts to create a culture of transparency: E-government and social media as openness and anti-corruption tools for societies. *Government Information Quarterly*, v. 27, n. 3, p. 264 – 271, 2010. ISSN 0740-624X. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0740624X10000201>>. Acesso em: 20 Jul 2017.

BRASIL. *Cartilha técnica para publicação de dados abertos no Brasil*. Brasília: Secretaria de Logística e Tecnologia da Informação e Ministério do Planejamento Orçamento e Gestão, 2009. Disponível em: <[http://www.governoaberto.sp.gov.br/wp-content/uploads/2016/02/Cartilha\\_Tecnica.pdf](http://www.governoaberto.sp.gov.br/wp-content/uploads/2016/02/Cartilha_Tecnica.pdf)>. Acesso em: 26 Jul 2017.

BRASIL. *Lei nº 12.527 de 18 de novembro de 2011. Lei de acesso a informações públicas*. Brasília: Presidência da República do Brasil, 2011. Disponível em: <[http://www.planalto.gov.br/ccivil\\_03/\\_ato2011-2014/2011/lei/l12527.htm](http://www.planalto.gov.br/ccivil_03/_ato2011-2014/2011/lei/l12527.htm)>. Acesso em: 2 Ago 2017.

BRASIL. *Pra que serve o Dados Abertos?* Brasília: Câmara dos Deputados - Palácio do Congresso Nacional, 2016. Disponível em: <<https://dadosabertos.camara.leg.br/howtouse/sobre-dados-abertos.html>>. Acesso em: 29 Abr 2018.

BRÍGIDA, L. S.; BRITO, V. S. de. *Deliberatório*. 2013. Disponível em: <<http://deliberatorio.com.br/como-jogar/>>. Acesso em: 27 Jul 2017.

COSTA JUNIOR, D. G.; MAGALHÃES, W. B. *HTML5 e Web Semântica, a web com significado*. Paranavaí: Universidade Paranaense, 2013. Disponível em: <<http://web.unipar.br/~seinpar/2013/artigos/Djalma%20Goncalves%20Costa%20Junior.pdf>>. Acesso em: 27 Jan 2018.

EU Parlamentar. 2018. Disponível em: <<https://play.google.com/store/apps/details?id=com.ionicframework.eu273435>>. Acesso em: 16 Mar 2018.

G1. *Brasileiros não se sentem representados por políticos em exercício, aponta pesquisa*. 2018. Disponível em: <<https://g1.globo.com/politica/noticia/brasileiros-nao-se-sentem-representados-por-politicos-em-exercicio-aponta-pesquisa.ghml>>. Acesso em: 3 Fev 2018.

HACKER, C. T.; ALVARENGA, E. Z.; ESFERA.MOBI. *Manual dos dados abertos: governo. Traduzido e adaptado de opendatamanual.org*. Laboratório Brasileiro de Cultura Digital e Núcleo de Informação e Coordenação do Ponto BR (NIC.br), 2011. Disponível em: <[http://www.w3c.br/pub/Materiais/PublicacoesW3C/Manual\\_Dados\\_Abertos\\_WEB.pdf](http://www.w3c.br/pub/Materiais/PublicacoesW3C/Manual_Dados_Abertos_WEB.pdf)>. Acesso em: 20 Jul 2017.

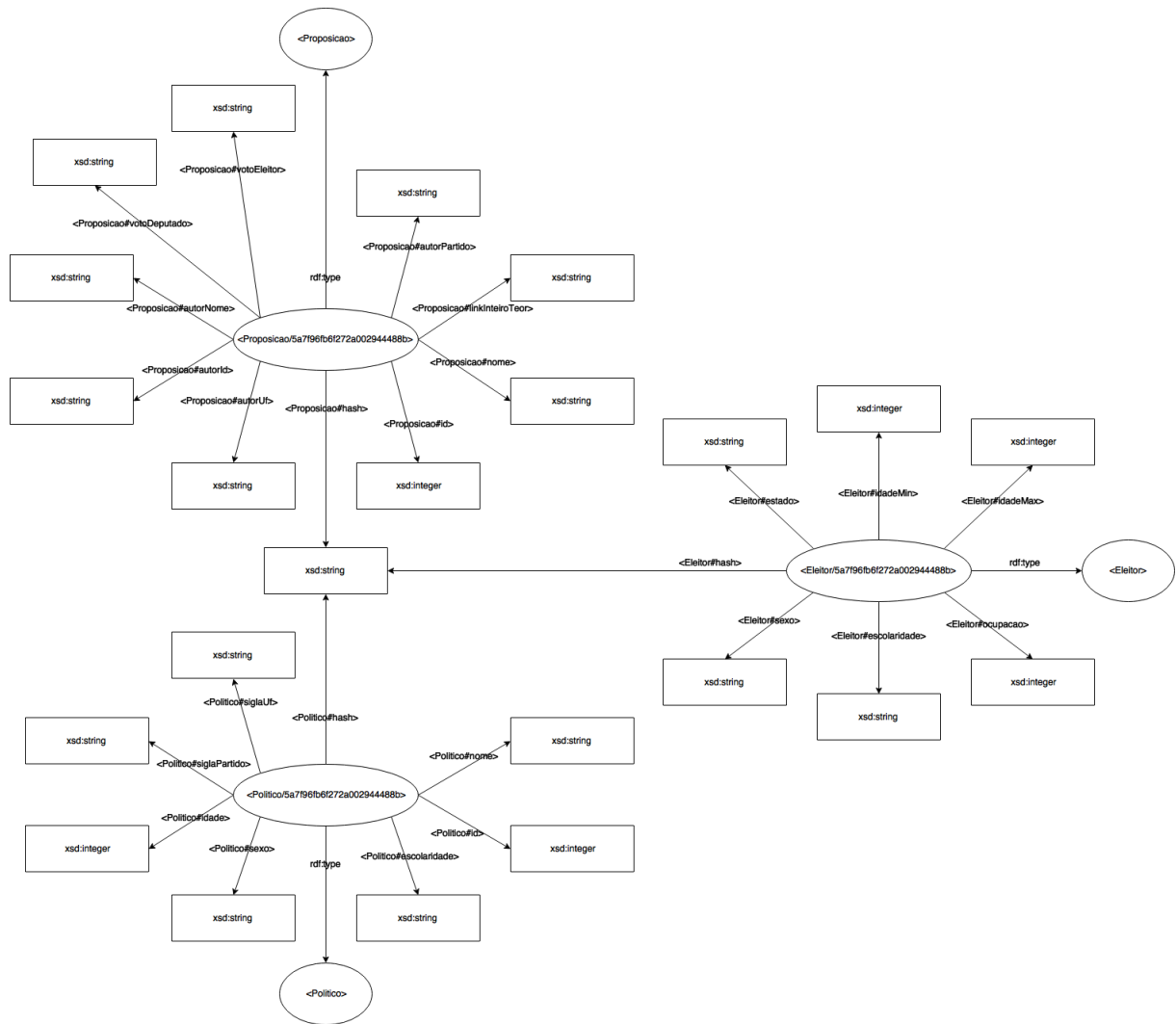
IGLESIAS, D. *Open data and the fight against corruption in Brazil*. World Wide Web Foundation, 2017. Disponível em: <[https://www.transparency.org/whatwedo/publication/open\\_data\\_and\\_the\\_fight\\_against\\_corruption\\_in\\_brazil](https://www.transparency.org/whatwedo/publication/open_data_and_the_fight_against_corruption_in_brazil)>. Acesso em: 14 Jul 2017.

- IZDEBSKI, K. *Transparency and open data principles: Why they are important and how they increase public participation and tackle corruption*. Warsaw: [s.n.], 2015. Disponível em: <<https://transparencee.org/wp-content/uploads/2015/12/open-data-principles-by-krzysztof-izdebski.pdf>>. Acesso em: 14 Jul 2017.
- KIM, S.; KIM, H. J.; LEE, H. An institutional analysis of an e-government system for anti-corruption: The case of open. *Government Information Quarterly*, v. 26, n. 1, p. 42 – 50, 2009. ISSN 0740-624X. From Implementation to Adoption: Challenges to Successful E-government Diffusion. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0740624X08001172>>. Acesso em: 23 Jul 2017.
- LAMBSDORFF, J. G. *The Transparency International Corruption Perceptions Index 1999 — Framework Document*. Transparency International, 1999. Disponível em: <[https://www.transparency.org/files/content/tool/1999\\_CPI\\_Framework\\_EN.pdf](https://www.transparency.org/files/content/tool/1999_CPI_Framework_EN.pdf)>. Acesso em: 23 Jul 2017.
- LEITE, L. *Radar Parlamentar*. São Paulo: PoliGNU - Grupo de Estudos de Software Livre da Poli/USP, 2012. Disponível em: <<http://radarparlamentar.polignu.org>>. Acesso em: 27 Jul 2017.
- LIMA, J. C. de; CARVALHO, C. L. de. *Resource Description Framework (RDF)*. Goiânia: Universidade Federal de Goiás - Instituto de Informática, 2005. Disponível em: <[http://www.inf.ufg.br/sites/default/files/uploads/relatorios-tecnicos/RT-INF\\_003-05.pdf](http://www.inf.ufg.br/sites/default/files/uploads/relatorios-tecnicos/RT-INF_003-05.pdf)>. Acesso em: 8 Mar 2018.
- MEU Deputado. 2015. Disponível em: <<https://meudeputado.mobi>>. Acesso em: 16 Mar 2018.
- MICHAELIS. *Dicionário Prático - Língua Portuguesa (Nova Ortografia)*. 2. ed. São Paulo: Melhoramentos, 2011.
- MONITORA, Brasil! 2015. Disponível em: <<https://monitorabrasil.org>>. Acesso em: 16 Mar 2018.
- MOON, M. J. Can it help government to restore public trust?: Declining public trust and potential prospects of it in the public sector. *Proceedings of the 36th Hawaii International Conference on System Sciences*, IEEE, 2003. Disponível em: <<https://www.computer.org/csdl/proceedings/hicss/2003/1874/05/187450134a.pdf>>. Acesso em: 23 Jul 2017.
- MORESI, E. *Metodologia da Pesquisa*. Brasília: Universidade Católica de Brasília - Programa de Pós-graduação Strictu Sensu em Gestão do Conhecimento e Tecnologia da Informação, 2003. Disponível em: <[https://s3.amazonaws.com/academia.edu.documents/34168313/MetodologiaPesquisa-Moresi2003.pdf?AWSAccessKeyId=AKIAIWOWYYGZ2Y53UL3A&Expires=1524161173&Signature=9Uy4Phajbc0dMCPbXfyk5xsz0mY%3D&response-content-disposition=inline%3B%20filename%3DMetodologia\\_da\\_Pesquisa\\_PRO-REITORIA\\_DE.pdf](https://s3.amazonaws.com/academia.edu.documents/34168313/MetodologiaPesquisa-Moresi2003.pdf?AWSAccessKeyId=AKIAIWOWYYGZ2Y53UL3A&Expires=1524161173&Signature=9Uy4Phajbc0dMCPbXfyk5xsz0mY%3D&response-content-disposition=inline%3B%20filename%3DMetodologia_da_Pesquisa_PRO-REITORIA_DE.pdf)>. Acesso em: 4 Abr 2018.
- MUSSKOPF, I. *Operação Serenata de Amor*. Data Science Brigade, 2016. Disponível em: <<https://serenatadeamor.org/about/>>. Acesso em: 27 Jul 2017.

- OLIVEIRA, R. *SPARQL – Parte II – Consultas básicas (select)*. Renan Oliveira [Blog], 2011. Disponível em: <<https://renanoliveira.net/2011/02/19/sparql---parte-ii---consultas-basicas-select/>>. Acesso em: 27 Nov 2017.
- PRODANOV, C. C.; FREITAS, E. C. de. *Metodologia do Trabalho Científico: Métodos e Técnicas da Pesquisa e do Trabalho Acadêmico*. 2. ed. Novo Hamburgo: Universidade Feevale, 2013. Disponível em: <<http://www.feevale.br/Comum/midias/8807f05a-14d0-4d5b-b1ad-1538f3aef538/E-book%20Metodologia%20do%20Trabalho%20Cientifico.pdf>>. Acesso em: 23 Jul 2017.
- RAMALHO, R. A. S. *Web Semântica: aspectos interdisciplinares da gestão de recursos informacionais no âmbito da Ciência da Informação*. Marília: Faculdade de Filosofia e Ciências da Universidade Estadual Paulista, 2006. Disponível em: <<http://www2.dc.ufscar.br/~gbd/download/files/meetings/meetings2009-2/Rogério.2006.WebSemantica.pdf>>. Acesso em: 27 Jan 2018.
- ROSE-ACKERMAN, S. *Corruption and Government: Causes, Consequences, and Reform*. 2. ed. Cambridge: Cambridge University Press, 2012.
- SR. Cidadão. 2017. Disponível em: <<http://site.srcidadao.com.br/#!/home>>.
- TAVARES, J. *Poder do Voto*. São Paulo: Folha de São Paulo, 2018. Disponível em: <<https://www1.folha.uol.com.br/poder/2018/03/aplicativo-quer-ajudar-eleitor-a-acompanhar-parlamentares.shtml>>. Acesso em: 16 Mar 2018.
- TEIXIDOR, B.; PAZOS, S. D. *No veas este vídeo si vas a votar hoy : WHYDEMOCRACY*. Madrid, Spain: El Instituto, 2016. Disponível em: <<https://www.youtube.com/watch?v=k8vVEbCquMw>>. Acesso em: 14 Jul 2017.
- W3C. *RDF 1.1 Turtle*. W3C, 2014. Disponível em: <<https://www.w3.org/TR/turtle/>>. Acesso em: 25 Fev 2018.
- XAVIER, O. C. *Utilizando a API do Twitter no desenvolvimento de aplicações web com PHP e cURL*. Linha de Código, 2012. Disponível em: <<http://www.linhadecodigo.com.br/artigo/3471/utilizando-a-api-do-twitter-no-desenvolvimento-de-aplicacoes-web-com-php-e-curl.aspx>>. Acesso em: 16 Nov 2017.
- ZAPATER, M. *Democracia não é vontade da maioria*. Justificando, 2015. Disponível em: <<http://justificando.cartacapital.com.br/2015/06/26/democracia-nao-e-vontade-da-maioria/>>. Acesso em: 22 Mar 2018.
- ZHANG, W.; ZHANG, Y.; WANG, B. E-government application in combating corruption: China's case. *2011 International Conference on Information Management, Innovation Management and Industrial Engineering*, IEEE, Shenzhen, v. 2, p. 246–250, 2011. Disponível em: <<http://ieeexplore.ieee.org/abstract/document/6116742/?reload=true>>. Acesso em: 20 Jul 2017.

## APÊNDICE A – GRÁFICO DA ONTOLOGIA DOS DADOS DO SAIBA MAIS BRASIL

Figura 28 – Gráfico completo da ontologia dos dados do Saiba mais Brasil.



Fonte – Elaborada pelo autor.

## APÊNDICE B – ROTAS DA API SAIBA MAIS BRASIL

**Quadro 2** – Rotas da API Saiba mais Brasil.

<b>Método HTTP</b>	<b>Rota</b>	<b>Retorno</b>
GET	/	
GET	/api/deputados	deputados
GET	/api/deputados/:id	deputado
GET	/api/deputadosrebase	boolean
GET	/api/partidos	partidos
GET	/api/partidos/:id	partido
GET	/api/partidosrebase	boolean
GET	/api/estados	estados
GET	/api/estados/:id	estado
GET	/api/estadosrebase	boolean
GET	/api/proposicoes	proposicoes
GET	/api/proposicoes/:id	proposicao
GET	/api/proposicoesrebase	boolean
GET	/api/temas	temas
GET	/api/temas/:id	tema
GET	/api/votacoes	votacoes
GET	/api/votacoes/:id	votacao
GET	/api/votacoesrebase	boolean
POST	/api/twitter	tweet
GET	/api/rdf	boolean
POST	/api/rdf	rdf
GET	/public/rdf.ttl	rdfs

**Fonte** – Elaborado pelo autor.

## APÊNDICE C – TELAS DE INTRODUÇÃO DO APP

Essas telas são apresentadas ao usuário apenas no primeiro acesso.

**Figura 29** – Tela de introdução 1.



**Fonte** – Elaborada pelo autor.

Figura 30 – Tela de introdução 2.



Fonte – Elaborada pelo autor.

**Figura 31** – Tela de introdução 3.



**Fonte** – Elaborada pelo autor.

**Figura 32** – Tela de dados do usuário.

The image shows a mobile application screen titled "Sobre você" (About you). It features a list of user profile fields, each with a label on the left and a value on the right, followed by a downward-pointing triangle indicating a dropdown menu. The fields are: "Faixa etária" (Age range) with the value "18 a 24 an..."; "Sexo" (Gender) with the value "Feminino"; "Escolaridade" (Education) with the value "Ensino Mé..."; "Ocupação política" (Political occupation) with the value "Nenhuma ..."; and "Estado de votação" (Voting state) with the value "Bahia". At the bottom of the screen, there is a blue button labeled "Continuar" (Continue).

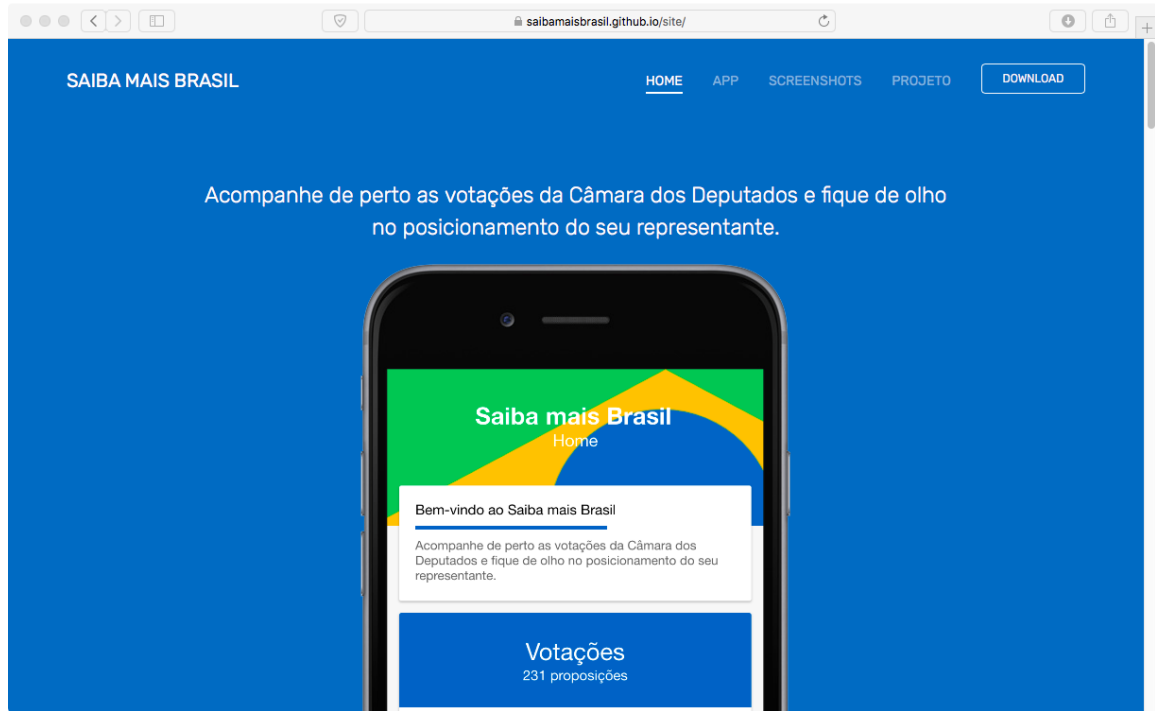
Sobre você	
Faixa etária	18 a 24 an... ▼
Sexo	Feminino ▼
Escolaridade	Ensino Mé... ▼
Ocupação política	Nenhuma ... ▼
Estado de votação	Bahia ▼

[Continuar](#)

Fonte – Elaborada pelo autor.

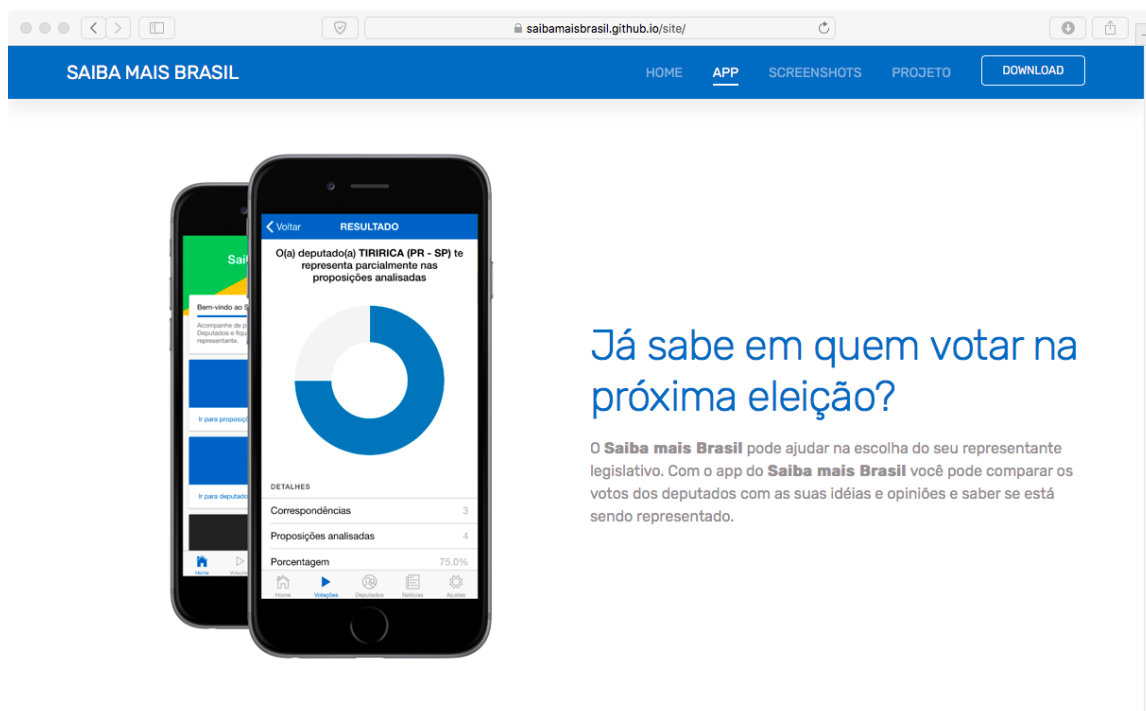
## APÊNDICE D – SITE DO SAIBA MAIS BRASIL

**Figura 33** – Site Saiba mais Brasil (Home).



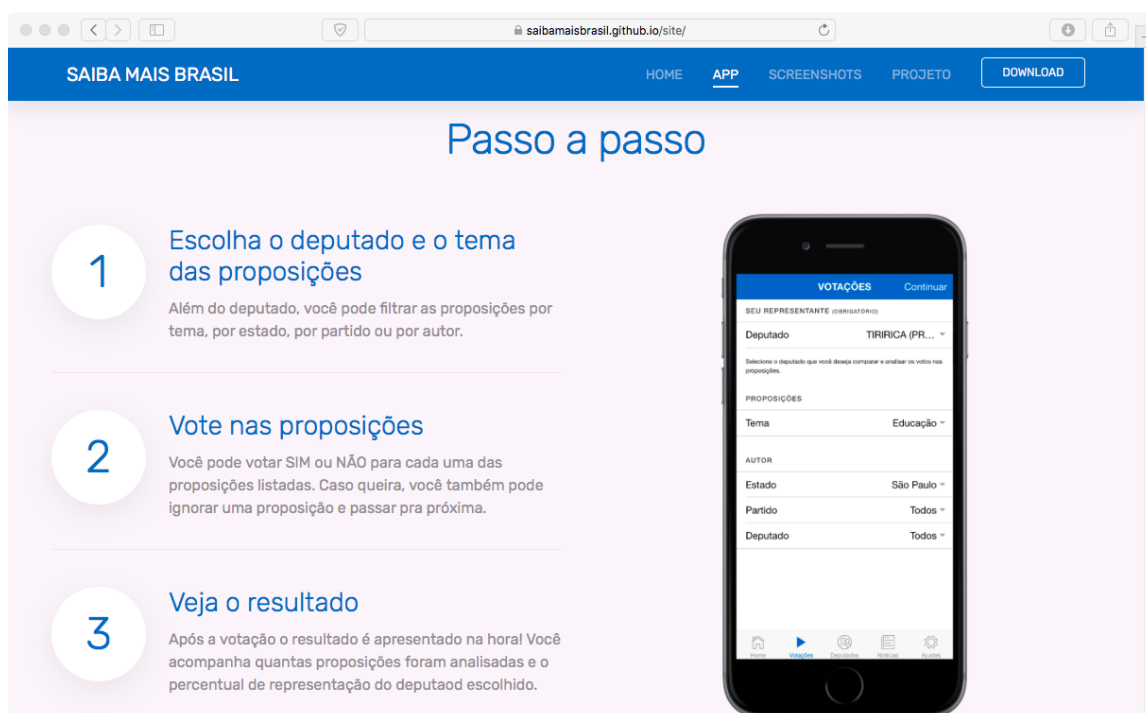
**Fonte** – Elaborada pelo autor.

Figura 34 – Site Saiba mais Brasil (Descrição do App).



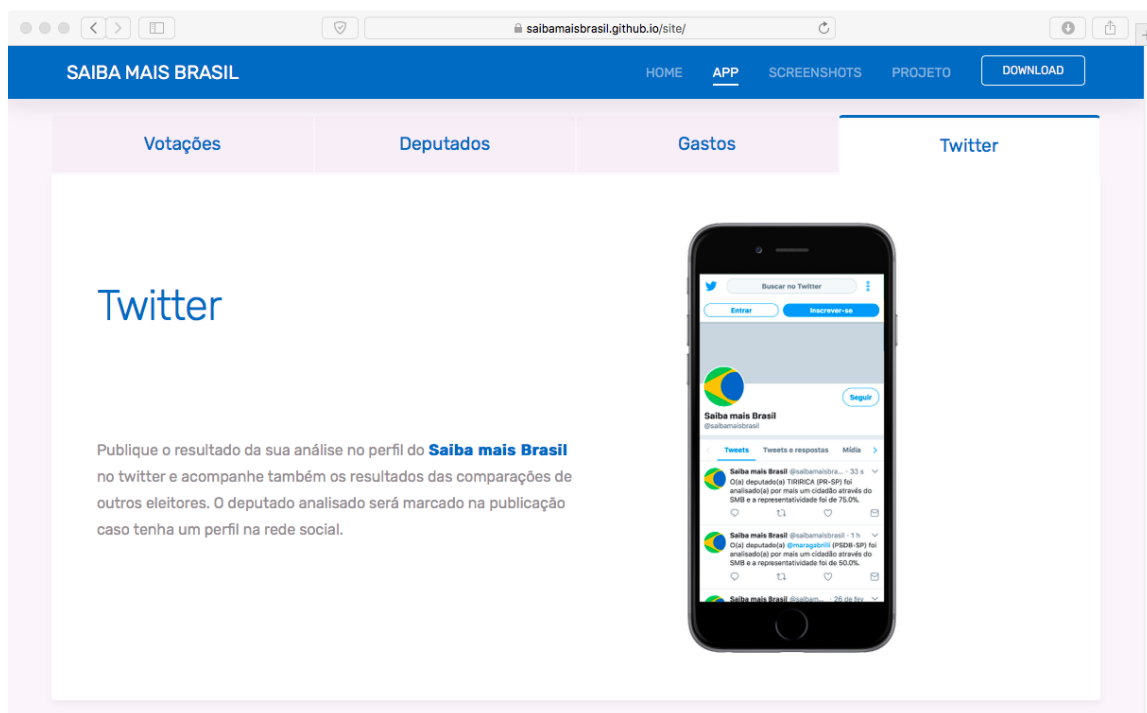
Fonte – Elaborada pelo autor.

Figura 35 – Site Saiba mais Brasil (Passo a passo para uso do App).



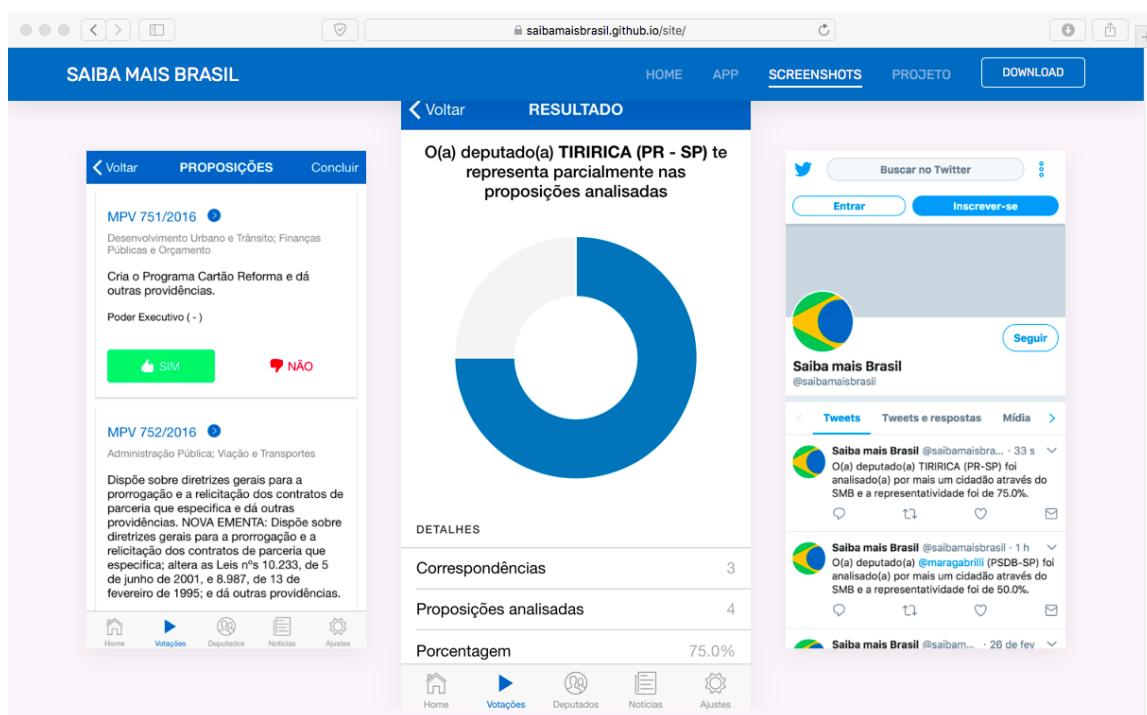
Fonte – Elaborada pelo autor.

Figura 36 – Site Saiba mais Brasil (Funcionalidades do App).



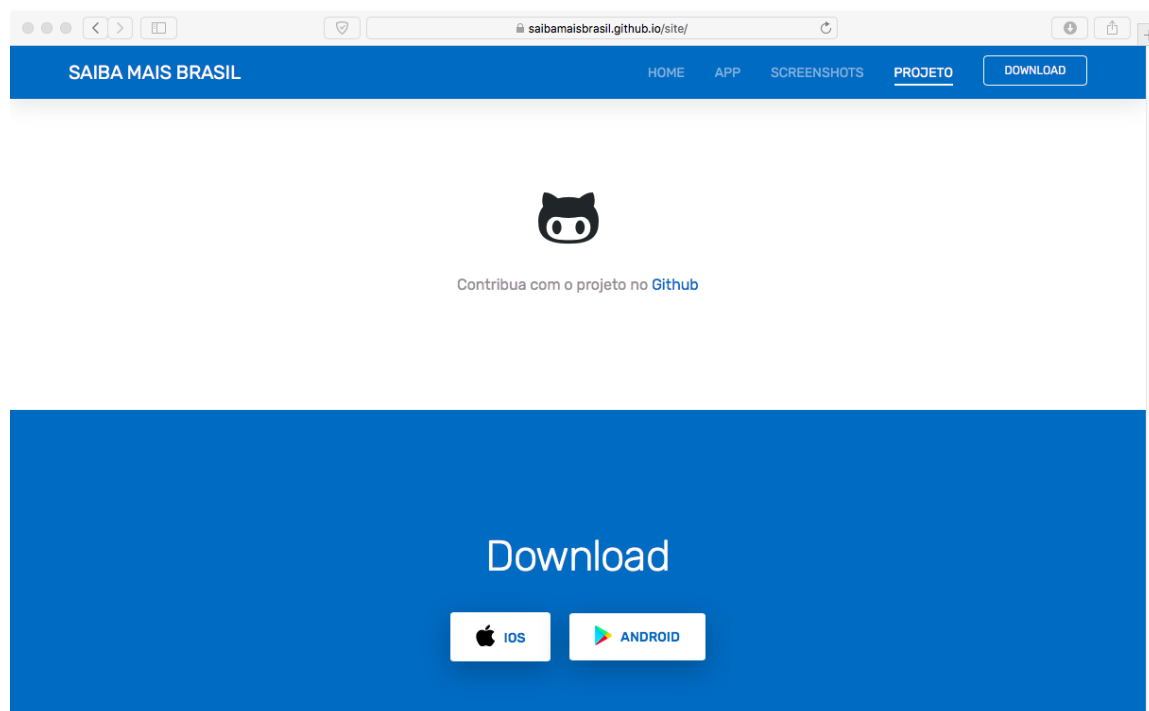
Fonte – Elaborada pelo autor.

Figura 37 – Site Saiba mais Brasil (Galeria do App).



Fonte – Elaborada pelo autor.

**Figura 38** – Site Saiba mais Brasil (Link do repositório e download do App).



**Fonte** – Elaborada pelo autor.